

Applying Graph Theory to Conservation Documentation

Ana Tam

A Thesis submitted in partial fulfillment of the
requirements for the degree of Doctor of Philosophy
at the University of the Arts London

July 2024

Volume 1 - Thesis

Table of Contents - Volume 1

<i>Table of Contents</i>	<i>i</i>
<i>Table of Figures</i>	<i>vi</i>
<i>Table of Tables</i>	<i>x</i>
<i>Acknowledgements</i>	<i>xiii</i>
ABSTRACT	1
1.0 INTRODUCTION	
1.1 The Problem Statement	2
1.2 Research Hypothesis	3
1.3 Research Aims and Motivations	4
1.3.1 Aligning Conservation with Advances in Information Management and Data Science	4
1.3.2 Achieving Data Integration via Graphs and Semantic Standards	6
1.3.3 Applying Graph Theoretic Analysis to Conservation Networks	6
1.4 The Research Questions	7
1.5 Structure of the Thesis	7
2.0 CONSERVATION DOCUMENTATION	
2.1 Content and Variety	8
2.2 Capturing Complexity	11
2.3 The Role of Documentation in Conservation Epistemology	12
2.4 The Role of Documentation in Conservation Inference	14
2.4.1 Rules-based Inference and Decision Trees	16
2.4.2 Heuristics	17
2.4.3 Embodied Knowledge	19
2.5 Siloed Documentation Systems vs FAIR Data Practices	21
2.6 Extending Documentary Practice via a Computational Thinking Framework	26

3.0 GRAPHS	
3.1 Mathematical Graphs and Graph Theory	34
3.2 Diagrammatic Graphs in Cognition and Philosophy	39
3.2.1 Structure Mapping Theory	44
3.2.3 Deleuze and Guatarri's Rhizomes	45
3.3 Graphs in Knowledge Representation and Semantic Networks	46
3.4 Knowledge Graphs	54
3.4.1 Definition of a Knowledge Graph	54
3.4.2 Building a Knowledge Graph for a Specific Domain	56
3.5 Existing Knowledge Graphs Relevant to Conservation	59
3.5.1 The CIDOC CRM	59
3.5.2 Cultural Heritage Thesauri	63
3.5.3 Linked Conservation Data	63
3.6 Examples of Graph-Based Applications and Analysis	64
3.6.1 Graphs applied to reasoning, prediction, and discovery	64
3.6.2 Graph models for monitoring places and spaces	64
3.6.3 Graphs for workflows and risk assessments	65
3.6.4 Graph models for flexibility at scale	67
3.6.5 Graph-based research in cultural heritage and archival sciences	68
3.6.6 Graph-based measures for domain analysis	70
3.7 Summary	71
4.0 A GRAPH REPRESENTATION METHOD FOR CONSERVATION	
4.1 Overview	74
4.2 Tools	75
4.2.1 The Limitations of RDF and the Rationale for Using Neo4j/LPG	75
4.2.2 Notes to the reader regarding Neo4j configuration and Cypher syntax	78
4.3 Modelling Principles for a Graph Representation Method for Conservation	81
4.3.1 The Representational Basis for the Data Model(s)	81
4.3.2 Sets, Tuples, and Subgraphs	83
4.3.3 Categorical Representation and Graph Enrichment	92
4.3.4 Attributes and Relationships: Disambiguating "Property" with Analogical Reasoning	96
4.3.5 Star schema	102

4.4 Graph Theoretic Analysis	103
4.4.1 Order and Size	104
4.4.2 Density/Sparsity, Degree, and Clustering	104
4.4.3 Triangles, Graphlets and Motifs	108
4.4.4 Paths, Distance, Shortest Path and Diameter	110
4.4.5 Planarity and K3,3 Bipartite Graphs	111
4.4.6 Eigenvector Centrality	112
4.5 Query-based Analysis and Inference	115
4.6 Verification, Validation and Calibration	115
4.6.1 Code Verification	115
4.6.2 Model Validation	115
4.6.3 Model Calibration	117
4.6.4 Integrating Verification, Validation and Calibration (VVC)	118
4.7 Summary of Method	118
5.0 CIDOC CONCEPTUAL REFERENCE MODEL (CRM)	
5.1 Background	120
5.1.1 The Versions	121
5.2 ETL: Importing the CIDOC CRM RDFS Models into Labelled Property Graph	122
5.2.1 Standard Transformation Procedure (ETL1)	122
5.2.2 Modified Transformation Procedure (ETL2)	123
5.3 Results of Graph Theoretic Analysis	127
5.3.1 Order and Size	127
5.3.2 Density/Sparsity	127
5.3.3 Global Triangle Count	131
5.3.4 Diameter	132
5.3.5 Planarity and K3,3 Bipartite Graph	133
5.3.6 Undirected Motifs Frequency	133
5.3.7 Eigenvector Centrality	138
5.4 Query-based Analysis and Inference	139
5.5 Verification, Validation, Calibration	143
5.6 Summary Findings	144

6.0 LINKED CONSERVATION DATA	
6.1 Background	146
6.2 ETL: Importing the LCD RDF models into Labelled Property Graph	151
6.3 Results of Graph Theoretic Analysis	152
6.3.1 Order and Size	153
6.3.2 Density/Sparsity	153
6.3.3 Global Triangle Count	159
6.3.4 Diameter	159
6.3.5 Planarity and K3,3 Bipartite Graph	160
6.3.6 Undirected Motif Frequency	161
6.3.7 Eigenvector Centrality	167
6.4 Query-based Analysis and Inference	170
6.4.1 Graph Exploration	170
6.4.2 Analysing for Objects, Materials and Types	178
6.4.3 Analysing for Treatment Events	184
6.4.4 Analysing for Techniques	188
6.4.5 Analysing for Trends Over Time	193
6.5 Verification, Validation and Calibration	200
6.6 Summary Findings	201
7.0 THE REVISED LPG MODEL	
7.1 Introduction	204
7.2 Applying the Method to Case Study Components	205
7.2.1 The TNA CCD Dataset	207
7.2.2 The NLP-derived Dataset	212
7.2.3 The TNA Discovery Catalogue Data	215
7.2.4 Conservation and Art Materials Encyclopedia Online (CAMEO)	218
7.2.5 Overview of the Resulting P3-LPG Schema	221
7.3 Results of Graph Theoretic Analysis	224
7.3.1 Order and Size	224
7.3.2 Density/Sparsity	224
7.3.6 Undirected Motif Frequency	224
7.3.7 Eigenvector Centrality	225
7.4 Transforming the Phase 3 LPG Graph to CRM-mapped RDF graph	225
7.5 Verification, Validation and Calibration	227
7.6 Summary Findings	228

8.0 DISCUSSION	
8.1 Thesis Summary	229
8.2 Reflections and Interpretations	231
8.2.1 (RQ1) How to build a conservation knowledge graph?	232
8.2.2 (RQ2) How can knowledge graph construction clarify the nature of complexity in conservation?	234
8.2.3 (RQ3) What are the affordances of graph-based analysis for conservation?	236
8.3 Implications	237
8.3.1 The Craft of Modelling	237
8.3.2 Verification, Validation and Calibration (VVC)	240
8.3.3 Understanding the CIDOC CRM	243
8.3.4 Implications on Conservation Practice	243
8.3.5 Challenges to Implementation	244
8.3.6 Cultural Assumptions	245
8.4 Limitations	246
8.5 Recommendations	247
8.5.1 Recommendations for Implementation	247
8.5.2 Recommendations for Implementing Graph Theoretic Analysis	248
8.6 Further Work	249
8.6.1 Further integration to enhance cross-searching across datasets	250
8.6.2 Other Graph Theoretic Measures	250
8.6.3 Bibliographic or Topic Networks	251
8.6.4 Machine Learning and Deep Learning Models	252
8.6.5 The Future of Conservation Practice	252
9.0 CONCLUSIONS	253
10.0 BIBLIOGRAPHY	257

List of Figures

Images and diagrams are the author's own work unless otherwise stated.

- Figure 1. 1 *"Graph representing the metadata of thousands of archive documents, documenting the social network of hundreds of League of Nations personals" (Grandjean 2013).*
- Figure 2. 1 *A diagram of the folder structure of artists' archive" (Barok, Thorez, et al 2019)*
- Figure 2. 2 *Schema illustrating the Map of Interactions that are involved in the making of the David Lamelas' Time (2018)" (Lawson et al 2019)*
- Figure 2. 3 *Diagram of the relationships between recording, documentation, and information management practices. (Letellier, Schmid, and LeBlanc 2007)*
- Figure 2. 4 *Deduction and Induction*
- Figure 2. 5 *Diagram of deductive reasoning from premises to conclusion*
- Figure 2. 6 *Oddy Tests at The British Museum.*
- Figure 2. 7 *Visualisation of a RDBMS database structure (from Gupta et al 2014)*
- Figure 2. 8 *The Taxonomy of Computational Thinking Practices by Weintrop et al (2016)*
- Figure 2. 9 *Current computational practices in the field of conservation at large*
- Figure 3. 1 *Example of a simple directed graph*
- Figure 3. 2 *Example of a simple graph with five vertices and five edges.*
- Figure 3. 3 *Euler's diagram of the 'Seven Bridges' problem. (Paoletti 2013)*
- Figure 3. 4 *Euler's diagram representing how his methodology. (Paoletti 2013)*
- Figure 3. 5 *The Seven Bridges of Königsberg problem represented as a graph.*
- Figure 3. 6 *Map of the Complexity Sciences (Castellani and Gerrits 2021)*
- Figure 3. 7 *The graph schema for the construction of book endleaves including choice variations by Campagnolo (2015)*
- Figure 3. 8 *Haeckel's original (1866) conception of the three kingdoms of life as a tree (acyclic graph).*
- Figure 3. 9 *Marine food web network (cyclic graph) in the pelagic zone (Choy, Haddock and Robison 2017)*
- Figure 3. 10 *Structure mapping diagram from Gentner (1983)*
- Figure 3. 11 *Example of a conceptual graph of the phrase "a person is between a rock and a hard place" by John Sowa*
- Figure 3. 12 *The Subject - Predicate - Object "triple" structure of RDF.*
- Figure 3. 13 *The LOD Cloud graph*
- Figure 3. 14 *The DBpedia data graph produced by the LodLive project expanding from the English keyword node for "Paraloid B-72"*

- Figure 3. 15 *An overview of the structures and functions of KOS [knowledge organization systems] from (Zeng 2008)*
- Figure 3. 16 *The DIKW model based on the diagram by Bellinger et al 2004.*
- Figure 3. 17 *Stokman and de Vries' (1988) workflow diagram for the Knowledge Integration and Structuring System (KISS) for knowledge graph construction.*
- Figure 3. 18 *The CIDOC CRM top level categories. From Bruseker et al 2017,*
- Figure 3. 19 *Conceptual modelling of event data using the CRM. From Sanderson, R. (2020)*
- Figure 3. 20 *Graph representation of the classes tree for E22 Man-Made Object (CRM v. 7.1.1)*
- Figure 3. 21 *Apgar Score.*
- Figure 3. 22 *"Workflow of pattern creation and application" by Kesper et al (2020).*
- Figure 3. 23 *Graphlet Permutations. from Espejo et al 2020*
- Figure 3. 24 *Basic descriptive statistics of all analyzed datasets from Zloch et al 2021.*
- Figure 3. 25 *A Revised Taxonomy of Knowledge Organisation Systems.*
- Figure 4. 1 *Contributions and influences on the GQL specification (from Neo4j 2019).*
- Figure 4. 2 *Directed graphs and multigraphs.*
- Figure 4. 3 *Comparisons between RDF and LPG structures.*
- Figure 4. 4 *An example of systems of information in conservation (not exhaustive).*
- Figure 4. 5 *Hypothetical representation of datasets connected as subgraphs.*
- Figure 4. 6 *Conceptualisations of sets*
- Figure 4. 7 *Example of further type decomposition using the example sets from Figure 4.6.*
- Figure 4. 8 *Modelling of the sets from Figure 4.6 and how they interrelate.*
- Figure 4. 9 *Examples of subgraphs of Figure 4.8.*
- Figure 4. 10 *A demonstration of connectivity across representational levels.*
- Figure 4. 11 *The systems of information in conservation from Figure 4.4 annotated by conceptual representational level.*
- Figure 4. 12 *Illustration of the Titanic example statements as RDF triples (s-p-o).*
- Figure 4. 13 *Illustration of the Titanic example statements in LPG.*
- Figure 4. 14 *Generic star schema diagrams for illustrative purposes.*
- Figure 4. 15 *Diagram to illustrate node degrees (from Bales and Johnson 2006).*
- Figure 4. 16 *Illustration of the structure of graph clusters (from Roy and Chakrabarti 2017).*

- Figure 4. 17 Motif patterns in sequence after Abuoda, Morales, and Aboulhage (2020).
- Figure 4. 18 Isomorphic graphs.
- Figure 4. 19 Isomorphs of K4
- Figure 4. 20 Representations of K3,3 and K5 graphs
- Figure 5. 1 Image of CIDOC CRM in LPG after ETL1 transformation procedures demonstrating an example of “modified” Relationships (RDF properties) for class E55_Type
- Figure 5. 2 Graph visualisation of the CIDOC CRM RDF Schema using ETL1 shows the CIDOC CRM is a highly-connected cyclic graph.
- Figure 5. 3 Two classes from ETL1 model of CIDOC CRM v.6.2.1
- Figure 5. 4 Excerpt from the CIDOC CRM v.6.2.1 RDFS presenting how ‘domain’ and ‘range’ are encoded.
- Figure 5. 5 The standard transformation procedure (ETL1) of the RDFS for CIDOC CRM
- Figure 5. 6 Example of the declared semantic triple $E18 \rightarrow P49 \rightarrow E39$ and its reciprocal statement $E39 \rightarrow P49i \rightarrow E18$.
- Figure 5. 7 The ETL2 modified transformation procedure adds a reciprocal $xDOMAIN$ and $xSCO$ relationship to reassert the semantic intention as in Figure 5.6
- Figure 5. 8 Graph visualisation of the CIDOC CRM RDF Schema using ETL2 shows the CIDOC CRM remains a highly-connected cyclic graph
- Figure 5. 9 The top 20 results from each CRM version presented side by side.
- Figure 5. 10 Visualisation of E18_Physical_Thing as a hub node surrounded by its properties (:Relationship)
- Figure 5. 11 Bar graph of results from table 5.3.8 in motif identifier order after Abuoda et al 2020
- Figure 5. 12 Bar graph of results from Table 5.3.9 which re-orders the results in line with ascending node-to-edge ratios of each $k=3,4,5$ motif
- Figure 5. 13 Visualisation of the “E1 CRM Entity” cluster of immediate neighbours (cluster on the left) and its connection with the “E2 Temporal Entity” cluster (on the right) from CRM v5.0.4
- Figure 5. 14 Visualisation of the “E1 CRM Entity” cluster of immediate neighbours (on the left) and its connection with the “E2 Temporal Entity” cluster (on the right) from CRM v7.1.1 (ETL2)
- Figure 5. 15 Searchable dropdown navigation menu from the Classes & Properties Declaration for v.7.1.1
- Figure 5. 16 Screenshots of the documentation for P10, P86, and P89.
- Figure 5. 17 Visualisation of results to the Cypher query for finding CRM property “P172”.
- Figure 5. 18 The results of searching for [:Relationship] nodes with “contains” as the label,

- Figure 5. 19 The visualised FOL statements as graphs.
- Figure 6. 1 LCD project transformation pipelines (Image source: Lieu and Campagnolo 2022)
- Figure 6. 2 LCD project data model. (Image source: Lieu and Campagnolo 2022)
- Figure 6. 3 Excerpt in TriG from *tna-data-2020-12-31.trig*
- Figure 6. 4 Visualisation of a *E52_Time-Span* node with CRM properties *P82a* and *P82b* transformed into node properties.
- Figure 6. 5 Windmill Graphs
- Figure 6. 6 The windmill-like expanded graph...
- Figure 6. 7 The LCD-BOD results of the Label Propagation analysis...
- Figure 6. 8 The LCD-TNA results of the Label Propagation analysis...
- Figure 6. 9 Visualisation of *m5.16* motif (Limit 1) from each LCD dataset.
- Figure 6. 10 Visualisation of *m5.16* motifs (limit 5) from each LCD dataset.
- Figure 6. 11 Bar graph of results from table 6.3.8 in motif identifier order after Abuoda et al 2020.
- Figure 6. 12 Bar graph of results from Table 6.3.9 which re-orders the results in line with ascending node-to-edge ratios of each *k=3,4,5* motif
- Figure 6. 13 ...the node for "ply" in the BOD dataset...
- Figure 6. 14 Examples from LOC and TNA original .trig files exhibiting encodings of *rdfs:label*
- Figure 6. 15 Visualisations of *E22_Man-Made_Object* nodes in each LCD dataset and their immediate neighbours.
- Figure 6. 16 The objects graph (*E22_Man-Made_Object*) for each LCD dataset.
- Figure 6. 17 Two object graphs, each representing a book from the LOC dataset.
- Figure 6. 18 The "Book Graph" for "Book (The National Archives, ADM 139/1023)" and its parts from the TNA dataset.
- Figure 6. 19 View of two SUL object graphs showing *E22* and *E19* nodes...
- Figure 6. 20 The material graphs for each LCD dataset.
- Figure 6. 21 The Type graphs demonstrate the distribution of categorical type nodes in each dataset.
- Figure 6. 22 Visualisation of the treatment events in the BOD dataset
- Figure 6. 23 Visualisations of *E11_Modification* and neighbouring nodes (of length 1) for each LCD dataset.
- Figure 6. 24 Visualisation of the LOC *E11* graph
- Figure 6. 25 Visualisation of the TNA *E11* graph
- Figure 6. 26 Partial detailed view of the SUL *E11* and neighbours graph.
- Figure 6. 27 Visualisation of where *E11_Modification* nodes are situated relative to *E57_Material* nodes.

- Figure 6. 28 Visualisation from each LCD dataset showing instances matching the list of 15 board reattachment techniques identified and used for queries by Velios and St. John (2022).
- Figure 6. 29 Visualisation of distance between E52_Time-Span (large, dark green) nodes to E57_Material (light green) nodes are of length 2 in LOC
- Figure 6. 30 Visualisation of distance between E52_Time-Span "decade" node to E57_Material are of length 3 in LOC
- Figure 6. 31 Timeline/Gantt chart visualisation of Table 6.4.9.
- Figure 6. 32 Relative position of E52_Time-Span to P32_used_general_technique relationships...demonstrates how proximity correlates with relevance in the SUL graph.
- Figure 6. 33 Relative position of E52_Time-Span to P33_used_specific_technique relationships...demonstrates how proximity correlates with relevance in the SUL graph.
- Figure 7. 1 Building a prototype conservation knowledge graph, using a hypothetical representation.
- Figure 7. 2 TreatmentEvent node with full row of data content mapped as properties to the node.
- Figure 7. 3 Visualisation of 10 treatment events represented as star schema.
- Figure 7. 4 Materials Graph. Visualisation of the network of treatment event (pink) nodes and material type (green) nodes.
- Figure 7. 5 Annotated Materials Graph (manually annotated by the author).
- Figure 7. 6 Visualisation of NLP-derived nodes as instances matching to Cameo (type) hub nodes.
- Figure 7. 7 Connecting an NLP-derived mention of "Tyvek" to the categorical (:Cameo) node for "Tyvek".
- Figure 7. 8 An excerpt of the results of a filter query...
- Figure 7. 9 The TNA Discovery catalogue identifiers...
- Figure 7. 10 The TNA Discovery catalogue tree hierarchy...
- Figure 7. 11 Representation of the specific to more general semantic relationships between data nodes, CAMEO nodes and CRM node.
- Figure 7. 12 The P3- LPG Schema
- Figure 7. 13 Example of a (:Vocab) node...
- Figure 7. 14 Example mapping for transformation to RDF.
- Figure 8. 1 Computational thinking practices employed in this research...
- Figure 8. 2 The iterative nature of knowledge graph development.
- Figure 8. 3 View of a random sample of the TNA CCD dataset
- Figure 8. 4 Annotated version of Figure 8.3 to highlight visually diagnostic features.
- Figure 8. 5 Visualisation of a spanning tree...

List of Tables

- Table 2. 4. 1 *Condition Categories. An example of a heuristic in conservation.*
- Table 2. 6. 1 *Example Research Areas Supported by a Computational Thinking Framework*
- Table 2. 6. 2 *Examples of quality dimensions for research data as defined by Kesper et al 2020 (after Laranjeiro et al 2015)*
- Table 3. 4. 1 *Selected definitions of “knowledge graph” as collated by Ehrlinger and WöB (2016)*
- Table 4. 2. 1 *List of Tools*
- Table 4. 2. 2 *Examples of Cypher Syntax*
- Table 4. 3. 1 *Angles’ (2018) definition of a property graph*
- Table 4. 3. 2 *Francis et al’s (2018) definition of a labelled property graph*
- Table 5. 1. 1 *The CIDOC CRM Versions*
- Table 5. 3. 1 *Order and Size Results for CIDOC CRM Group*
- Table 5. 3. 2 *Density/Sparsity Results for CIDOC CRM Group*
- Table 5. 3. 3 *Local Clustering Coefficient Results for CIDOC CRM Group*
- Table 5. 3. 4 *Degree Centrality Results for CIDOC CRM Group*
- Table 5. 3. 5 *Global Triangle Count Results for CIDOC CRM Group*
- Table 5. 3. 6 *Diameter Results for CIDOC CRM Group*
- Table 5. 3. 7 *K3,3 Bipartite Graph Results for CIDOC CRM Group*
- Table 5. 3. 8 *Undirected Motifs Frequency Results for the CIDOC CRM Group*
- Table 5. 3. 9 *CIDOC CRM Group Motif Node:Edge Ratios*
- Table 5. 3. 10 *Eigenvector Centrality Results for CIDOC CRM Group*
- Table 6. 1. 1 *List of the 15 board reattachment techniques*
- Table 6. 1. 2 *Summary of LCD datasets for secondary analysis*
- Table 6. 3. 1 *Order and Size Results for Linked Conservation Data Group*
- Table 6. 3. 2 *Density/Sparsity Results for Linked Conservation Data Group*
- Table 6. 3. 3 *Local Clustering Coefficient Results for Linked Conservation Data Group*
- Table 6. 3. 4 *Degree Centrality Results for Linked Conservation Data Group*
- Table 6. 3. 5 *Global Triangle Count Results for Linked Conservation Data Group*
- Table 6. 3. 6 *Diameter Results for Linked Conservation Data Group*
- Table 6. 3. 7 *K3,3 Bipartite Graph Results for Linked Conservation Data Group*
- Table 6. 3. 8 *Undirected Motifs Frequency Results for the Linked Conservation Data Group*
- Table 6. 3. 9 *LCD Group Motif Node:Edge Ratios*
- Table 6. 3. 10 *Eigenvector Centrality Results for Linked Conservation Data Group*

- Table 6. 4. 1 Count and Percentage of Nodes by Label per LCD Dataset*
- Table 6. 4. 2 Relationships in Order of Frequency Count and Percentage of LCD Dataset*
- Table 6. 4. 3 Colour-code key for Table 6.4.1. and Table 6.4.2*
- Table 6. 4. 4 List of Distinct Node Labels for Leaf Nodes for each LCD Dataset*
- Table 6. 4. 5 Example of composite string values for rdfs:labels and their classes*
- Table 6. 4. 6 Matches to the LCD techlist*
- Table 6. 4. 7 Techniques identified via strategies 1-3 from each LCD dataset*
- Table 6. 4. 8 Number and Percentage of E52_Time-Span nodes per LCD dataset*
- Table 6. 4. 9 Material Type Usage Over Time*
- Table 6. 4. 10 Results of E52_Time-Span datetimes and general technique types (via P32).*
- Table 6. 4. 11 Results of E52_Time-Span datetimes and specific technique types (via P33).*
- Table 7. 2. 1 Simulated Mappings to CIDOC CRM via paths.*
- Table 7. 3. 1 Order and Size Results for the Revised LPG Model*
- Table 7. 3. 2 Density/Sparsity Results for the Revised LPG Model*
- Table 7. 3. 4 Eigenvector Centrality Results for the Revised LPG Model*
- Table 8. 2. 1 Examples of tasks undertaken during the research that matches each computational thinking practice category and sub-category.*

Acknowledgements

My deepest thanks to the following persons and organisations, without whom this body of work would not have been possible.

I am grateful to the Arts and Humanities Research Council (UK) and the Techne Doctoral Training Partnership for funding this research.

To my supervisory team: Dr. Athanasios Velios, Dr. John Howse, and Dr. Malcolm Quinn. Thank you for your expertise, generosity of spirit, and steadfast encouragement through all these years.

The team at The National Archives (UK), particularly Sonja Schwoell, Sarah VanSnick, and David Underhill.

The CIDOC CRM Special Interest Group and staff at FORTH-ICS with special thanks to Martin Doerr, George Bruseker, Eleni Tsoulouha, Chryssoula Bekiari, Elias Tzortzakakis, and Anastasia Axaridou.

The Linked Conservation Data (LCD) project team, especially Kristen St. John, Ryan Lieu, and Alberto Campagnolo.

The Techne DTP administrative team: Jane Nobbs, Emma Molyneux, and Carol Hughes. The Research Management and Postgraduate Administration teams at UAL and the UAL library staff, especially Cait Peterson.

A special thank you to the wider ecosystem of individuals who helped sustain this journey through cross-pollination, especially the *Mercator* exhibition and *Cultivate* exhibition teams.

My family and friends. My patient and understanding co-pilots: Maya, Lily and Hugh.

Abstract

Conservation is the management of change (UNESCO et al 2013). Heritage assets—which include tangible and intangible objects and places—are recognised as non-renewable resources. As conservators, we are merely stewards of these shared assets and the notes and records we create contribute to their long-term care and understanding. Therefore, it is imperative these records are both human- and machine-readable. This thesis leverages mathematical graph theory to identify and examine networks captured in conservation documentation. It demonstrates how the use of existing graph-based technologies, such as semantic web technologies (RDF) and property graph (PG) databases, can be used to build and inform computational models for conservation through the creation and analysis of graph-based metamodels and knowledge graphs.

Conservation treatment data provided by The National Archives (UK) was used to develop a labelled property graph model and database that was also convertible to CIDOC CRM-mapped RDF triples. To further inform the development, investigations were conducted on existing conservation graphs, including the CIDOC CRM RDFS serialisation and RDF graphs produced by the Linked Conservation Data (LCD) project. The modelling decisions and investigations made during this process contributed to a suite of verification, validation and calibration (VVC) practices for graph model creation, assessment, and refinement, including the use of graph theoretic algorithms. The outcome is a graph representation method for conservation data which includes modelling principles to aid queryability and avoid common modelling pitfalls.

Of the graph theoretic measures employed, leaf node detection, triangle count, motif frequency, diameter and eigenvector centrality measures were found to be diagnostic for comparing or contrasting data collection practices as evidenced in the datasets across institutions. Eigenvector centrality is also a strong candidate for systematic model validation due to its usefulness in identifying modelling errors. Furthermore, results demonstrate that the conservation graphs from each study case exhibit recurrent bipartite $(k_{3,3})$ subgraphs, an indicator of non-planarity. This higher dimensionality speaks to an intrinsic characteristic of conservation data and may explain why tabular and traditional relational data models, while able to capture facets of conservation, have been so difficult to use to capture and model across conservation's more complex nature. These results signal a promising new means for conservation to capture, encode, study, and discuss complex conservation events and practices.

1.0 Introduction

1.1 The Problem Statement

To manage and preserve heritage objects and places, the conservation process requires the creation and use of a wide variety of records and documentation. This documentation provides the basis for making crucial judgements in caring for heritage and is itself an invaluable resource for scholarship. The expository nature of conservation records lends itself to being highly document-based (e.g. treatment reports, condition surveys, etc.) or tabular (e.g. spreadsheets). Velios (2016) has examined the difficulty of processing free-text or unstructured, document-based, conservation records and its problematic impact on information retrieval and information extraction. The simple structure of tabular data is often considered only to be a semi-structured data representation given the lack of consistent and explicit schemas to support automatic machine processing and requires general or domain-specific knowledge to interpret the semantic contents (Ristoski and Paulheim 2016). Thus, a disproportionately large amount of conservation information stored as spreadsheets and/or published on websites and relational databases is not accessible via searching or through automatic machine-readable means.

Documentation is intrinsic to conservation and the evolution of documentation practices will be essential to the evolution of the profession (Marchese 2011). Besides the ethical reasons for documentation, there are also significant economic considerations in collecting, storing, managing and disseminating information, including outright costs for staff and technology infrastructure to indirect costs and/or savings associated with informed planning and decision-making (Letellier, Schmid, and LeBlanc 2007). Inaccessible information is a two-fold problem consuming resources at point of data entry and again when a researcher seeks to retrieve it. A commissioned report by the Foundation of the American Institute for Conservation of Historic and Artistic Works (FAIC) (Zorich 2016) has identified three key challenges facing the conservation profession in today's digital world:

- fragmented resource materials across various platforms and locations,
- a lack of data standards,
- and a tendency for redundant efforts to create local solutions without wider collective gains.

The fact that the three key issues highlighted by this report all relate to information access and management is a resounding call to action for focused research and development. However, insufficient skills in information technology amongst existing conservation practitioners (Aitchison 2013) has been a likely hindrance. Nevertheless, the impact of digitisation on conservation cannot be ignored and its impact on documentation will only intensify (Roy, Folster, and Rudenstine 2007; Moore 2001).

1.2 Research Hypothesis

Knowledge graphs offer a versatile method for modelling connected data and semantic context. However, a comprehensive study on the application of knowledge graphs to model conservation documentation has not been undertaken despite its uses elsewhere in cultural heritage informatics (Arns 2016) and its status as a standard information modelling methodology (Hayes and Patel-Schneider 2014; Cyganiak, Wood, and Lanthaler 2014; Giutierrez 2008).

Graph theory is the area of mathematics that studies the interconnectedness of things. Graphs, in this sense, are made up of nodes and edges where the node, for example, is a piece of data and the edge is the relationship between two nodes resulting in a diagrammatic model (Diestel 2017), also known as a Knowledge Graph (Stokman and de Vries 1988). Graph analysis remains a largely under-utilised method in conservation research and practice.

The challenge of data fragmentation begs for resources and solutions in data integration. The application of graph theoretic approaches for knowledge representation and semantic mapping can address the three key issues raised in the FAIC report. This approach provides a data model that is more conducive to data integration and data interoperability and has been successfully deployed elsewhere (as Chapter 3 will detail).

Encoding is itself a documentary practice (Scifleet et al 2009). The expectation based on these precedents is that graph modelling and graph-based analysis provides a method for encoding conservation knowledge that is both human- and machine-readable, and can support the identification and inference of new knowledge in conservation. This 'new knowledge' includes the identification of existing gaps in knowledge and inferring what is missing. Therefore, a graph-based approach would support the computational turn in conservation.

1.3 Research Aims and Motivations

To address the problems stated above, the hypothesised graph-based modelling approach allows for the flexible encoding of data (i.e. discrete elements of meaning and interest) with data schema (i.e. frameworks that enable sense-making and informs how discrete elements are related and are meaningful, or in other words, the contextualisation of specialist knowledge). There is a strong potential for the resulting graph(s) (i.e. network models) to enable and improve both human- and machine-readability. The scope for capturing highly-structured networks in conservation and their degree of meaningfulness remains to be explored, therefore, this work presents a critical first step. The aims of this thesis are threefold:

1. to align conservation with advances in information management and data science using graph data models and graph technologies,
2. to demonstrate data integration using graph-based approaches and semantic standards, and
3. to apply graph theoretic analysis to existing and case study conservation networks to assess the level of effectiveness of such an approach in terms of methodology and new knowledge.

The following subsections provide the context and motivations for deriving each of these aims.

1.3.1 Aligning Conservation with Advances in Information Management and Data Science

Since the FAIC (2016) report, Otero (2022) has further emphasised the challenges facing the future of heritage conservation by situating the locus of the problem with a misalignment between existing data practices in conservation and advances made in data management and data science:

[To] date, there has not been a single work on any macroperspective analysis or data science applied to the understanding and management of the conservation data from heritage. This is surprising especially for three reasons:

- i. studies of the heritage conservation are incredibly data-rich and spread in a vast number of sources;*

- ii. *current research is still progressing without macroperspective directions;*
- iii. *most excellent scientific findings lack nowadays the adequate dissemination and are rarely transferred into practice (ibid.).*

There is an ongoing trend for information management standards and methodologies, particularly those for the Semantic Web, to align towards a graph-based model. This signals an increasing relevance for the conservation sector to attain knowledge of graph-based approaches. However, explorations into its use to model conservation-specific information is limited in the literature. The benefit of a graph-based approach is the flexibility to handle structured and unstructured (or schema-free) data. Beyond a model for data storage, it has enabled new ways to engage with information (Mugnier and Chein 1998), bringing information management and interrogation together into one system (Robinson, Webber, and Eifrem 2015).

The widespread application of graphs in information management spans fields from bioinformatics (Pavlopoulos et al 2011), to ecological research (Dale 2017), communication networks (Kumar, Wainright and Zecchina 2015) to investigative journalism (Hunger and Lyon 2016), and serves as an underpinning technology of search engines (Uyar and Aliyu 2015). To paraphrase Grandjean (2014) who applied a graph-based approach to his investigation of League of Nations documents: the network is already present in the object itself [the object here being the archive], this approach only serves to reveal and give the researcher another way to analyze it.



Figure 1.1 Graph representing the metadata of thousands of archive documents, documenting the social network of hundreds of League of Nations persons (Grandjean 2013).

Otero's essay (2022) was a call to action to further highlight how the experience and knowledge of conservation professionals add value and as such should be documented and disseminated. A graph-based approach provides the tools and frameworks to achieve this.

1.3.2 Achieving Data Integration via Graphs and Semantic Standards

A graph-based model enriches data with more detail and context and lends itself to Tim Berners-Lee's vision of the Semantic Web:

to have a common and minimal language to enable to map large quantities of existing data onto it so that the data can be analysed in ways never dreamed of by its creators (Berners-Lee, Hendler, and Lassila 2001).

Furthermore, a graph-based approach to information modelling and management supports algorithmic analyses methods, such as network analysis and artificial intelligence methods. In essence, this transforms a passive information storage system into a dynamic analysis engine.

Graphs follow an established deductive framework for mapping data as a series of interconnected nodes, akin to trains of thought (Summers-Stay 2017). Graphs have been demonstrated to be a good fit for modelling cultural heritage data and have been directly used with the CIDOC CRM framework (Bogdanova, Todorov, and Noev 2016; Brushke and Wacker 2014), an internationally recognised data integration standard for modelling cultural heritage information. It has also been demonstrated (Mantegari, Matteo and Vizzari 2010) that mapping cultural heritage data using graphs can generate new knowledge.

1.3.3 Applying Graph Theoretic Analysis to Conservation Networks

This research has been influenced and inspired by professor Mark Dale's *Applying Graph Theory in Ecological Research* (2017). Dale's work draws on an extensive existing body of literature in the ecological sciences where graph theoretic approaches have not only been trialed but also tested with very clear patterns and methods that aid the understanding of ecological systems, resulting in a very accessible volume with many applied examples of mathematical graph theory.

However, unlike the ecological sciences, currently, to the best of this author's knowledge, there is a dearth of applications and published literature on the use of graph theory for the conservation of cultural assets. Hence, there is a clear challenge in how to adopt graph theoretic-based methods and apply them to heritage conservation.

1.4 The Research Questions

In order to address and assess the challenges of understanding and adopting a graph-based approach to data management and data science for conservation, this research focuses on the following research questions:

RQ1. How do we build a conservation knowledge graph?

RQ2. How can knowledge graph construction clarify the nature of complexity in conservation?

RQ3. What are the affordances of graph-based analysis for conservation?

For clarity, this research does not seek to extend graph theory itself, rather it aims to apply graph theory for the purposes of an empirical investigation on conservation documentation.

1.5 Structure of the Thesis

Chapters 2 and 3 will provide the contextual review for this research, firstly, in terms of the purposes of conservation documentation, the state of its data landscape and related challenges. Secondly, a brief history of graphs will be presented, including how they have been utilised in knowledge representation and information management. Chapter 4 sets out the method in principle used in this research. Chapters 5 to 7 recount the implementations of the method upon the case study datasets, provide examples of encodings, and the results. The datasets include the CIDOC CRM ontology as a structure to be assessed, data from the Linked Conservation Data project, and data provided by The National Archives (UK). Chapter 8 will discuss the results and recommend trajectories for future research and, finally, chapter 9 will provide summary conclusions.

2.0 Conservation Documentation

The challenges identified by the FAIC report (Zorich 2016) place digital data management at the forefront of concerns in the conservation profession. The challenges associated with data management are not only a matter of technological infrastructure, but have wider implications on conservation knowledge organisation and epistemology. Therefore, to assess the appropriateness and feasibility of the proposed knowledge graph-based solution, first, this chapter will begin with a summary review of the role and significance of conservation documentation in conservation decision-making. In Chapter 3, graphs and graph theory will be introduced along with examples of how “knowledge graphs” are utilised in knowledge representation, semantic technologies, and analysis across a variety of fields, particularly in health and medicine, which have often been compared with conservation (Ashley-Smith 2016; Smith and Přikryl 2007 amongst others). Finally, a summary of findings will be presented that strongly supports the need to investigate how graphs can be applied to conservation documentation.

2.1 Content and Variety

The definition of a cultural heritage asset spans all that can be valued by humans as contributors to their culture and heritage. This includes everything from prehistoric rock art to industrial plastics, spacecraft and ephemera. Many different specialists, from engineers, surveyors, archaeologists, curators, historians, scientists, and even the general public, for example, by means of oral history projects, can shed light on how best to care for heritage objects and sites and contribute to understanding the significance of the heritage assets. This diversity of sources means that records can vary in their methods, scope, and levels of detail (Letellier, Schmid, LeBlanc 2007).

A brief demonstration of the complex, multi-faceted nature of the conservation knowledge base can be found in the work of Zorich and Fuentes (2014) and Barok, Thorez, et al (2019). The former examined approximately 500 online resources used by the professional conservation community to establish a baseline understanding of the digital conservation “information space”. Barok, Thorez, et al (2019), on the other hand, identified a host of multi-leveled, intra-institutional resources and living artists’ archives (see Figure 2.1) that conservators must consult and manage in order to inform future re-installation of “complex digital artworks” (also known as time-based media). This diversity of collaborative sources of information is typical to heritage management at large. One only needs to consider the cross-cultural and interdisciplinary information

sources (Sloggett 2009) and the technology-driven collaborative working resources (Marty 1999) at play to get a sense of the ever-growing corpus of reference materials. Therefore, it is not possible to provide an exhaustive list of the various types of conservation documentation here due to the immense variety, but as Green and Mustalish (2009) have put it, conservation documentation must encompass:

All the information that is generated by conservation, preservation, and scientific activities, including texts of examination records, treatment reports, analytical results, and accompanying images in digital format (ibid, 7).

They also hasten to say in their survey of digital technologies and conservation documentation management systems:

No single system currently exists that can successfully accommodate the full scope of requirements for the broad range of media and institutional workflows represented by the entire conservation profession (ibid, 7).

As work practices evolve with increasing expectations to access databases across multiple computer points, to publish collections and object information online, and to improve public access to such information, cloud-based database solutions have gained interest (M&G NSW, n.d.). As a result of more public-facing information related to conservation stewardship, the conservator's role has expanded to that which Barok, Noordegraaf, and de Vries (2019) have re-contextualised in digital media terms:

Conservators are content creators.

The conservator as content creator and the conservator as steward are not opposing positions, but rather, content creation is an expansion of the conservator's stewardship role through the medium of publicly-facing data/digital resources. Stewardship is not a role isolated from the public. It requires broad understanding and willingness to engage and examine multiple values and perspectives¹. While the role's direct proximity to heritage assets is a privileged position, this privilege comes with a deep responsibility to the wider public to sustain and improve access and engagement. Documentation as content for communicating conservation is one means of addressing this responsibility.

¹ An example of this is given on p.27 in regards to the *Digital Repatriation of Biocultural Collections: Rio Negro, Amazonia* project.

The *Map of Interactions* (see Figure 2.2) developed by Lawson et al (2019) highlights institutional and extra-institutional contributors, and both human and non-human elements involved in a record for performance-based artwork. Re-installation, or further performances, at subsequent times and locations, still must express an authenticity and a critical sense of being “live” (ibid.) in connection with the original. Identifying and tracking intangible aspects and qualities is crucial and it serves to emphasise how heritage conservation is not limited to objects and artefacts that are physical and material, but also to those which are intangible and give meaning (UNESCO 2003; Bedjaoui 2004). Documentation can often be the only physical evidence. Hence, there is no limit as to the form, manner or content of conservation documentation, but the overriding intention is that the resulting content will be used by other conservators (whom some will be far removed both geographically and temporally to when the record is made) to inform ongoing care.

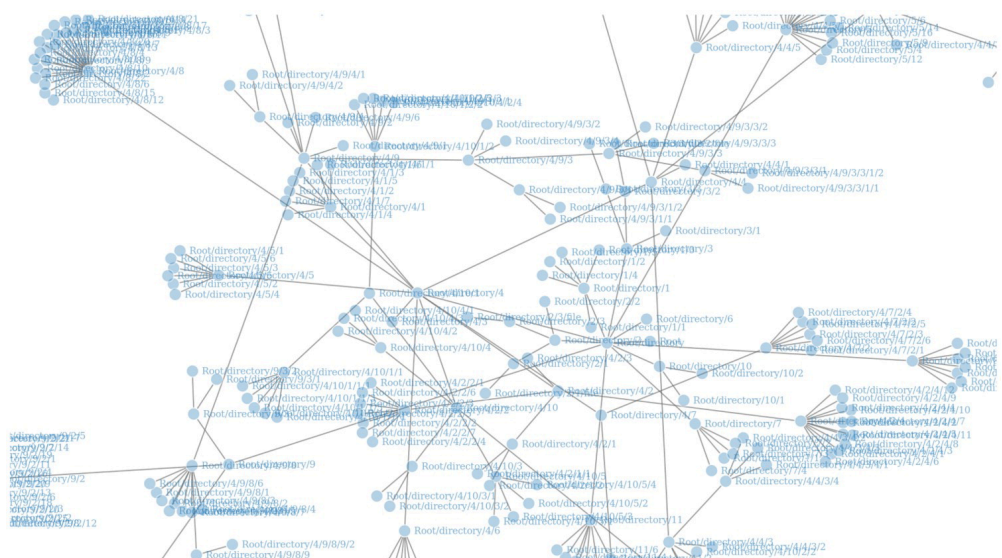


Figure 2.1 “A diagram of the folder structure of artists’ archive”. (Barok, Thorez, et al 2019, Fig. 2). Although the labels are illegible (as is the case in the original image), the graph structure is visibly obvious.

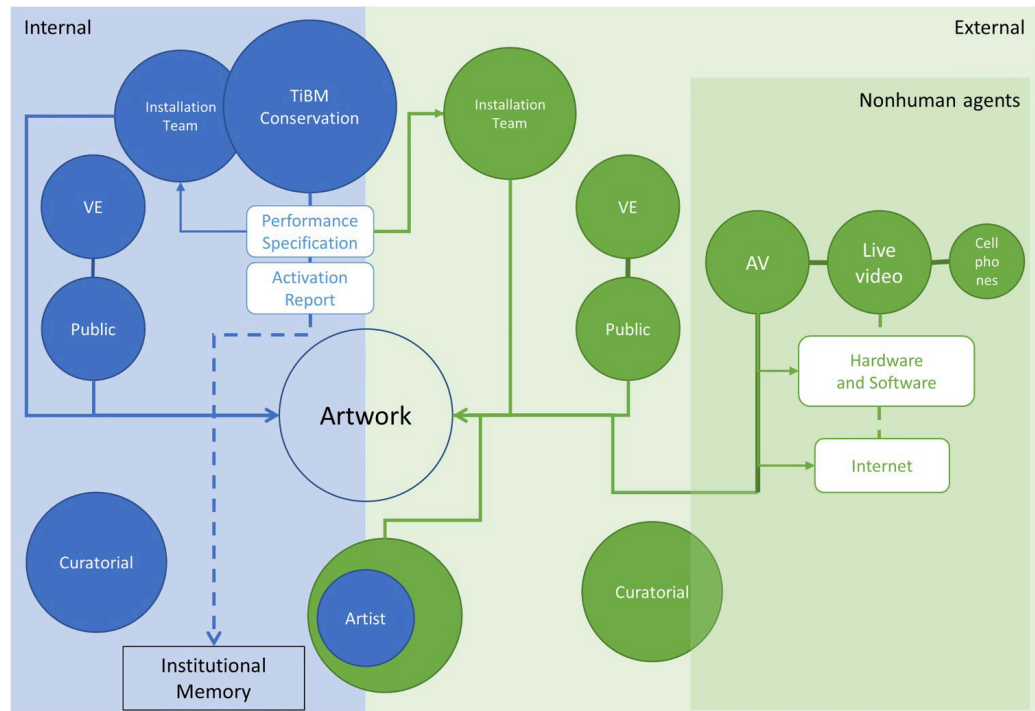


Figure 2.2 "Schema illustrating the Map of Interactions that are involved in the making of the David Lamelas' Time (2018)" (Lawson et al 2019, Fig. 2).

2.2 Capturing Complexity

The broad scope and high variability to be found in conservation documentation reflects the complex nature of conservation and the challenges of what can be captured in a record. Complexity in a conservation context refers to the myriad of considerations necessary to progress towards a suitable solution for a given problem. That is, to take the direct, literal definition of the word "complexity", conservators must always consider:

a group or system of different things that are linked in a close or complicated way; a network" [or that which is] "not easy to analyse or understand [and can be] complicated or intricate (Pocket OED, 2023).

Dealing with complexity is inherent to the conservation profession (Ashley-Smith 2000; Henderson 2018) to the end in which "coping with complexity" is a category in itself when assessing professional competence, according to the UK Institute of Conservation's Professional Accreditation of Conservator-Restorers (PACR) Handbook (2016). In fact, the words "complex" and "complexity", in the context of "complex

conservation problems” or “situations”, appear 21 times within the 42-page handbook. Yet, the nature of this complexity has not been studied directly beyond an attribution to chaotic systems (Ashley-Smith 2000) and the recognition that complexity is synonymous with uncertainty (Ashley-Smith 2000; Taylor 2018; Henderson 2018). Other characteristics of complexity, such as emergent properties, where the overall behaviour of a system cannot be reduced to the behaviours or intentions of specific components (Sturmberg and Martin 2013; Cilliers 2013, 31), have not been investigated nor specifically verified.

As will be discussed in Chapter 3, graph representation and graph theoretic approaches serve as an underlying approach to the study of complexity itself. Graph theory offers a way of understanding complexity with a language and discernible features to help describe components and contributors to complexity at any scale. Therefore, the advantages of investigating using a graph-based approach is not only beneficial for diagrammatic insights but can provide insights into patterns within a wider network of activity and system of communication.

2.3 The Role of Documentation in Conservation Epistemology

Contemporary conservation theory and policy identifies cultural heritage as a non-renewable resource (Holtorf 2001; Europa Nostra 2014). Conservation policy and legislation in the UK is framed within the context of resource management and sustainability (English Heritage/Historic England 2008). Decision-making in conservation practice correlates with risk management (Ashley-Smith 2000) and best practice standards further emphasise ‘impact on significance’ as a key consideration (Historic England 2015; Russell and Winkworth 2009). Applebaum (2009) frames conservation treatment as a goal-oriented activity, where minimising risk to the asset is one such goal, but not necessarily the only potential goal. Other goals may include aesthetic reasons or culture-driven reasons. Therefore, conservation methodology (Applebaum 2009; Ashley-Smith 2000; Munoz-Vinas 2012) recognises the need to assess many avenues of potential activities, including taking no action at all (Ashley-Smith 2018). To communicate these complex histories (i.e. sequences of events and activities and evolving or multi-stranded narratives), record-keeping serves as an intrinsic part of the conservation profession with a direct impact on present day and future decision-making. Producing documentation is firmly situated within conservation professional standards

and codes of ethics (ICON 2020; AIC/FAIC 1994). It is recognised across international conventions (UNESCO 1970; UNESCO 1972; UNESCO 2001; UNESCO 2003). The criticality of documentation cannot be overstated. A record is evidence and implies an ongoing process (Caple 2012, 70). The heritage asset itself serves as a record. It is a historic document as it contains evidence and information of the past (Letellier, Schmid, and LeBlanc 2007; Cronyn 2003). Where the heritage asset is no longer available, whether through decay, damage, loss or ephemerality, sometimes the documentation is the only evidence available.

The point of departure or initiator of the decision-making process “usually begins due to a particular question, an interest, or a specific situation” (Giebeler et al 2021, after Fischer and Funke 2016, 217–229). Muñoz-Viñas also refers to the “kinds of questions a practicing conservator typically needs to answer when working” (2022, 177). These many forms of documentation are a key element in the inferential process to understanding the past and identifying suitable courses of action. The role of questions in decision-making are bound up with the role of documentation. It is the recording and tracking of progress towards finding and applying answers. Hence, documentation is intrinsic to the epistemology of conservation. In fact, Caple (2012, 70) attributes the practice of record-keeping and records creation to have marked the “crucial” transition in conservation “from a craft to a profession”.

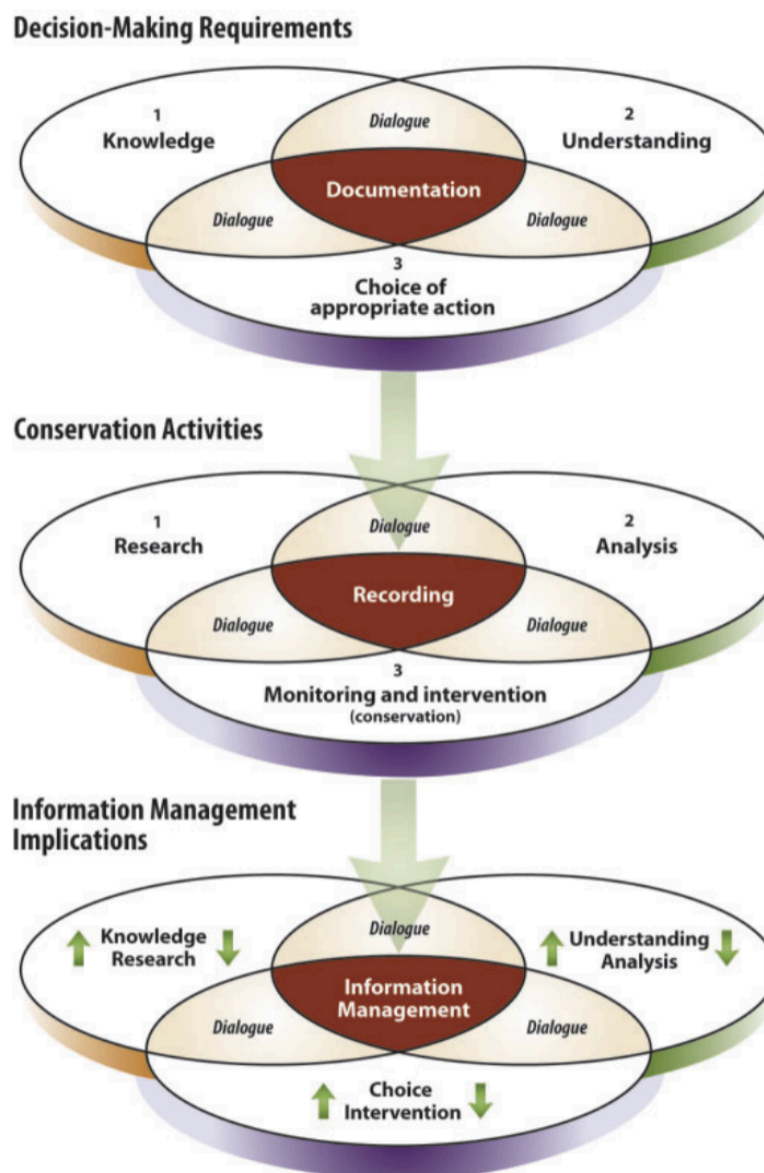


Figure 2.3 “Diagram of the relationships between recording, documentation, and information management practices. Well-managed heritage information is a powerful communication tool through which understanding is traded and shared” (Letellier, Schmid, and LeBlanc 2007, 28, Fig. 18).

2.4 The Role of Documentation in Conservation Inference

The nature of inference in conservation has been expounded by Ashley-Smith (2000), Henderson (2018), Taylor (2018), and Caple (2012). Inference is the process of deductive or inductive reasoning. The former derives specific conclusions from general premises and the latter derives general principles from specific instances (Figure 2.4).



Figure 2.4. Deduction and Induction

There is no formal system of logic specific only to conservation. Conservation is an applied discipline, like medicine or engineering, and therefore utilises systems of logic and reasoning derived from various frameworks, including the scientific method and analogical reasoning.

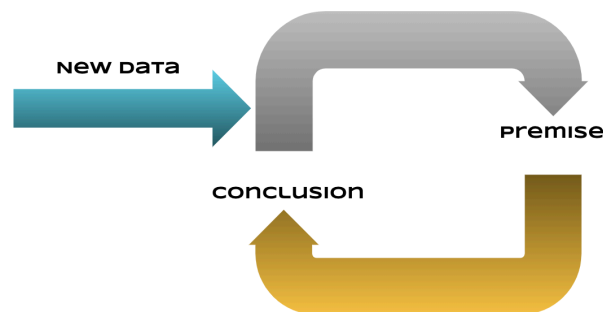


Figure 2.5. Diagram of deductive reasoning from premises to conclusion

The content of documentation serves as premises not only in the conservation decision-making process but can also serve as premises for understanding the historic and cultural past. Premises can be weighted, with some bearing greater consideration than others. For example, a risk value to the physical stability of an object and a risk value to preserving the significance of an object may be prioritised over other concerns such as aesthetics, cost, or accessibility, although there are circumstances where aesthetics, cost, and accessibility considerations are pushed to the fore (Henderson 2019). Inferring from documented premises can derive further documentation, such as treatment proposals, where several avenues of consideration are discussed based on the evidence and proposed projections. Furthermore, the results of a survey by Lindsay (2018) on the uses of collections care documentation (CCD) for wider operational decision-making found the following regular use-cases:

- *to promote the impact of preservation activities,*
- *encourages greater understanding of collections management processes,*

- *can act as a prompt for annual reviews of data,*
- *reduces the loss of institutional knowledge when staff leave or retire,*
- *allows patterns of risk to be identified and analysed,*
- *prevents duplication of activities,*
- *provides data for funding applications,*
- *provides information for annual reports and other management functions (ibid, S176).*

In practice, the conceptual schemas or cognitive frameworks for inferencing in conservation include:

1. rules-based inference and decision trees,
2. heuristics, and
3. embodied knowledge.

The next three subsections will expound upon these cognitive inference frameworks in the context of conservation decision-making.

2.4.1 Rules-based Inference and Decision Trees

Rules-based inference is deterministic and uses explicit criteria, which once met, precipitates explicit actions to follow, which in turn lead to specific conclusions. Rules of inference in law can come into play, for example in conservation instances dealing with historic building modifications or new town planning initiatives where progress of such undertakings must meet the tests set out under the conservation clauses (nos. 184 - 202) within the *National Planning Policy Framework* (Ministry of House, Communities and Local Government 2019) and arguments on precedents must maintain consistency in decision-making. Therefore, the conservation process within the historic environment can play out somewhat differently from a museum or archive context where a treatment decision for one object will not necessarily set up a context that influences or creates a formalised precedent that must be adhered to by subsequent instances. Legislation, professional standards, and best practice guidelines are examples of such formalised decision-making considerations, albeit some are more explicit in their influence than others.

Caple's (2012) influential volume, *Conservation Skills: Judgment, Method and Decision Making*, expounds the nature of decision-making to involve being confronted with a

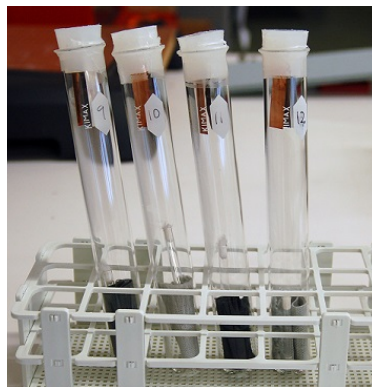
range of options and possessing agency to make appropriate selections (ibid, 170). He cites Moore and Thomas' *The Anatomy of Decisions* (1976) which itself drew significantly from mathematical logic and probability, including the formal logic constructs of *decision trees*. The application of statistical methods to conservation problem-solving and collections management was first discussed and demonstrated by Reedy and Reedy (1988). Statistical methods can help define and refine the rules for decision-making, such as the use of weightings or calculable thresholds for differentiating one course of action over another. However, implementation of statistical approaches in day-to-day operations have neither been systematic nor widespread. A renewed call to arms to use statistical methods was raised by Suenson-Taylor et al (1999) who drew parallels between conservation treatments and clinical trials and emphasised the need for conservation to leverage the statistical analysis of existing data for predictive means and to inform progress in data collection. Caple (2012) also included statistical methodologies alongside his advocacy of decision trees. However, challenges in skills (Aitchison 2013) and resource shortages (Zorich 2016) have seen this avenue of analysis in conservation remain largely under-developed. Chapter 3 on graphs will speak specifically to how graphs are used in knowledge representation such as to encode rules-based schema, perform statistical analysis of data, and applied to predictive tasks.

2.4.2 Heuristics

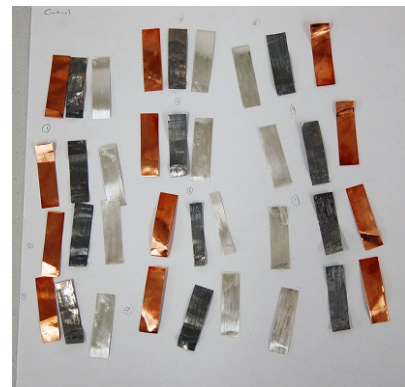
Heuristics are short-cuts or 'rules of thumb' used in the decision-making process to reduce the cognitive load of managing uncertainty, that is, the potential for large decision trees associated with countless options and subsequent outcomes (Dietrich 2010; Cioffi 1997). Heuristics are a type of rules-based inference strategy that attempts to truncate the potential breadth and scope of a decision tree. A common heuristic employed is *segmentation* where a framework of a few categories stand in for a broader, sometimes infinite, spectrum of options. Segmentation heuristics employ explicit criteria for each category, albeit some of these can be subjective in their interpretation, and are used to help order and prioritise groupings (i.e. sets). Examples of segmentation heuristics include the use of traffic-light systems and condition categories in assessment practices.

Figure 2.6 below is an example of heuristic-based decision-making used to categorise the results of accelerated materials tests, also known as 'Oddy tests' (Thickett and Lee 2004), where a 'pass', 'fail', or 'temporary' value is assigned to the tested materials based on the extent of corrosion visible on the metallic test coupons at the end of the

test. A ‘pass’ (green) indicates the assessor is confident the material can be used within a museum display environment with a very low expectation of adverse effects to the collection. A ‘fail’ (red) indicates the assessor is confident of a high expectation the material will off-gas harmful compounds that can decay or damage items in the collection. And ‘temporary’ (yellow) indicates the assessor has identified off-gassing of potentially harmful compounds but the deterioration effects are of a low enough range that limited exposure, for example, for use in a temporary exhibition as opposed to a long-term installation, is possible under certain conditions and considerations of which the assessor would detail in the notes column of the spreadsheet.



(a)



(b)

(c)

Figure 2.6. Oddy Tests at The British Museum. (a) Test tubes prepared for accelerated materials testing. (b) Resulting test coupons of copper, lead, and silver with varying degrees of corrosion. (c) Screenshot of the spreadsheet “Oddy Test Results Database 2014-2018”, British Museum (2018).

Table 2.1 is an example of condition categories assigned by the conservator to a heritage asset to indicate the urgency in which intervention is required and this is correlated with the asset's condition in terms of chemical-physical stability.

Table 2.4.1 Condition Categories. An example of a heuristic in conservation.

Indicator	Category	Description
A	Good	No work needed.
B	Fair	Low conservation priority. (In stable condition, but some work is desirable when other priorities and/or resources permit.)
C	Poor	Medium conservation priority. Not in immediate danger but needs essential work.
D	Unacceptable	High conservation priority. (e.g. active deterioration)

Another heuristic in conservation is the “classification of material and collection type based on gross similarities” (Ashley-Smith 2000), a common practice in heritage management based on the premise that similarly made assets of similar materials will likely deteriorate in the same manner and therefore require similar preservation conditions.

However, uncertainties also arise from heuristic-related biases (Henderson 2018) where incorrect inferences can result from over-simplified or imprecise heuristic frameworks. The potential to capture rules-based schema as a graph extends to capturing heuristics with the possibility for identifying and tracking when and where short-cutting can be successful and where it may not.

2.4.3 Embodied Knowledge

In *Contemporary Theory of Conservation* (2012), Muñoz-Viñas speaks of “microdecisions” and how conservation activities can be the result of sequences of countless, almost imperceptible, inferences made by the conservator:

The conservator applies a certain amount of a solvent to the tip of a swab; the amount is important: too much and the solvent will very likely run down the painting; too little and it might do nothing at all or, worse still, the swab might erode the painting. This amount of solvent simply cannot be measured using a scientific method (e.g. weighing the amount of solvent absorbed by the swab or applying it with a graduated pipette...)

So, each time a swab is imbibed with a solvent, the conservator is judging whether or not the swab contains an adequate amount of solvent. After deciding that the amount of solvent is adequate (a microdecision based upon previous experiences in varnish removal), the swab is applied, usually with a rubbing motion... (Muñoz-Viñas 2012, 133-134).

Such is an example of what Merleau-Ponty (1962) called “embodied knowledge”. In this age of information and big data, there remains a recognition and respect for experience. Not all that is worth knowing is explicit, easily recordable, or transferable. How to communicate tacit knowledge remains of great interest to academia and the commercial sectors. The “idea that embodied knowledge is as important in science and medicine as it is in any other area, such as craftsmanship” (Kneebone 2019a) was explored during two multi-disciplinary symposia². The predominant aim of both symposia was to address a “crisis in skills and understanding of the embodied and material world (Kneebone 2019b).”

Any research into conservation knowledge must appreciate the tacit or embodied ways of knowing within the profession. However, it is unlikely that detailed recorded instances of inferencing from embodied knowledge will feature strongly in the case study datasets in this research as articulating, quantifying and recording of embodied knowledge remains elusive, imprecise and, presently, uncommon in practice. Nevertheless, there is the potential for a graph-based approach to contribute towards the capture of embodied knowledge as the flexible and associative nature of the graph data model (Homan and Kovacs 2009) can support narrative representations.

²*Encounters on the Shop Floor: Embodiment and the Knowledge of the Maker* (symposium), 26-28 June 2019, Victoria & Albert Museum, Andrew W. Mellon Grant;
Picturing the Invisible (symposium), 7-8 November 2019, Chelsea College of Art, University of the Arts London, AHRC Network Grant

2.5 Siloed Documentation Systems vs FAIR Data Practices

The following challenges for conservation documentation have been identified (Zorich 2016; Zorich and Fuentes 2014; Green and Mustalish 2009; Velios 2016a; Velios 2016b; Suenson-Taylor et al 1999; Aitchison 2013):

- Systems legacy issues
- Data format compatibility and multi-format management
- Data entry (including free-text compatibility)
- Query issues (including limited queries in existing user interfaces)
- New end-user demands (e.g. web integration, support for new data file types)
- Limited resources and technical support
- Terminology

Addressing these challenges requires aligning conservation with advances in information management and data science (i.e. the first aim of this thesis, see section 1.3.1 above). The solution will not be found in a one-size-fits-all, siloed documentation system, but through FAIR data practices. FAIR stands for “Findable, Accessible, Interoperable and Reusable” and is a set of principles put forth by 53 co-authors from the international scientific community in 2016 that set out what characteristics data resources, tools and infrastructures should possess in order to support discoverability and reuse (Wilkinson et al 2016).

For example, a machine may be capable of determining the data-type of a discovered digital object, but not capable of parsing it due to it being in an unknown format; or it may be capable of processing the contained data, but not capable of determining the licensing requirements related to the retrieval and/or use of that data. The optimal state—where machines fully ‘understand’ and can autonomously and correctly operate-on a digital object—may rarely be achieved. Nevertheless, the FAIR principles provide ‘steps along a path’ toward machine-actionability (ibid).

Immediately, the locus of practice is reframed in both human and machine readable and actionable terms. FAIR compliance practices include rich metadata, transparent provenancing of data sources and data cleaning pipelines, and interoperability via general purpose, open technologies. While records and their medium for recording (i.e. the documentation system) tend to be inextricably linked in practice, we must

disambiguate “documentation systems” by delineating documentation *practice* from documentation *procedures* and documentation *tools* (particularly the use of technologies) as these can become conflated, which obfuscates and limits what data management and data science processes are actionable by human agents and/or machines. The sheer volume and diversity of data to store and process for research purposes alone (not including general operational purposes, etc.) necessitate computational (i.e. machine) assistance.

Documentation practice encompasses the creation and use of records while *documentation procedures* are formalised methods, often institution-specific and/or work-flow-specific, for record creation and use. Nevertheless, it is often conflated in the literature. For example, in the book *Museum Documentation Systems: developments and applications* (Light, Roberts and Stewart, first edition 1986 and 2014), which has long informed informatics in museums, libraries and archives, “museum documentation system” is defined as:

the procedures used by museums to manage information concerning their collections or of relevance to their curatorial functions. The primary aims of such a system include aiding the control and use of collections and ensuring the preservation of information about the cultural and environmental heritage.

There have been numerous reviews, studies, and critical appraisals of tools and technologies for cultural heritage management (Salvatore 2018 is a key reference), particularly museum documentation systems (Light, Roberts and Stewart, 1986; Sledge 1999; Carpinone 2010; M&G NSW, n.d., to name a few). The topic remains in regular discussion within heritage domain forums such as the Museums Computer Group³ and the Global Conservation Forum⁴ (formerly Conservation Online DistList). While another comprehensive assessment of museum documentation systems is outside the scope of this study, it is pertinent to contextualise for the reader how computational methods of analysis (i.e. data science) can be carried out, agnostic of the data repository system, to add value to the domain.

The first electronic data repository system was the database management system (DBMS) developed by the General Electric Company in the USA in 1964 (McLeod and Schell 2001). Adoption of such systems by the museum sector saw two decades of

³ <https://www.jiscmail.ac.uk/cgi-bin/webadmin?A0=mcg>

⁴ <https://www.culturalheritage.org/publications/online-publications/global-conservation-forum> (requires subscribing and creating a login and profile in order to review posts)

transitions from the 1960s to 1970s from paper-based records, to the use of punch cards and then mainframes (Carpinone 2010) with legacy records and formats persisting. Since the 1980s, local PC-based, then networked, and now cloud-based systems continue to contribute to institutional documentation systems that equate such systems with a *CMS*. The acronym, CMS, has been used interchangeably to mean “collection management *system*” and “collection management *software*” (Museums & Galleries of New South Wales, n.d.) in reference to a database system. Again, the focus of documentation in this manner limits knowledge organisation down to the scope of a technological platform, which is often specific and proprietary. For example, two well-known collections management systems are TMS by Gallery Systems⁵ and EMu^{6,7} by Axiell⁸.

Efforts to standardise practice and technologies have yielded process-based solutions with documentation standards in museums often aligning to Spectrum (Collections Trust, n.d.). Spectrum is a process standard for museum functions (Sledge 1999). It outlines procedures and information content requirements for how something, such as taking inventory or lending objects out, should be done to meet the standard (Collections Trust, n.d.). Spectrum, as it is published and maintained by the Collections Trust, is not formally encoded (i.e. machine-readable) but rather a recognised framework or policy to be adopted for use. Nevertheless, Spectrum has been incorporated into the design of proprietary museum information systems through a license⁹ with the software reviewed and validated by the Collections Trust to meet specific grades or levels of compliance¹⁰ (ibid.).

If software is Spectrum Compliant it has a place for every unit of information you might need to record for any procedure. There might not always be an exact one-to-one match between Spectrum units and system fields, but the developer will have shown Collections Trust how they map across. More importantly, the developer will be able to explain to you how to record any Spectrum unit using the system. Only systems validated by Collections Trust can call themselves Spectrum Compliant.

⁵ <https://www.gallerysystems.com/>

⁶ EMu was purchased by Axiell from KE Software in 2014 but may still appear as KE EMu in the literature (Collections Trust, n.d.)

⁷ <https://emu.axiell.com/support/documentation/the-database-engine/emu-s-database-documentation>

⁸ Other Axiell products used in museums, galleries and libraries which may not be Spectrum-compliant but readers may be familiar with include MIMSY XG, AdLib Collections Management System and CALM for library collections.

⁹ <https://collectionstrust.org.uk/spectrum/spectrum-licensing/>

¹⁰ <https://collectionstrust.org.uk/software/>

Both of the proprietary CMS examples given above (TMS and EMu) are Spectrum compliant. However, neither the aforementioned CMS platforms nor Spectrum compliance attest to the compliance of the data content to being both human- and machine-readable and actionable to the same level. This risks failing interoperability from a machine-readable perspective and, therefore, will continue to contribute to the challenges listed above.

As a process standard, Spectrum is similar to BIM (Building Information Modelling), a collaboration process standard for the production and management of electronic information in the architecture, engineering and construction (AEC) industries¹¹. This includes Historic BIM guidance produced by Historic England (2017). The AEC industry accepts that various actors and stakeholders may use different in-house information systems for their own purposes, however, the demands for digital collaborative working necessitates adoption of FAIR principles in enterprise contexts, hence BIM Level 3 is a step towards FAIR compliance where a project is run using general purpose, machine-readable, open technology formats at the point of data exchange and collaborative working. In-house documentation systems need not be dismantled or superseded, only the working practices are modified, for example, copies of datasets are extracted or exported into more general purpose formats. These data procedures, in turn, are incorporated into the overall data management process (which can be automated).

Despite there being numerous proprietary systems available (as noted in the above cited reviews and studies), the underlying backend data model for these systems is based on the relational model by Codd (1970). These relational database management systems (RDBMS) run on the SQL query language¹² and are generalised as SQL databases. However, limitations of the relational database model were acknowledged from its inception. According to Homan and Kovacs (2009, 211), "E.F. Codd, the father of the relational database" [wrote]:

a "relational database is best suited to data with a rather regular or homogeneous structure" and that more research is needed to determine if an

¹¹ <https://www.thenbs.com/knowledge/bim-levels-explained>

¹² SQL (pronounced "sequel") was developed based on Codd's relational model (1970) and became a recognised standard by the International Organization for Standardization (ISO) in 1987 (ISO 9075:1987). The latest version is ISO/IEC 9075-11:2023 (<https://www.iso.org/standard/76586.html>)

RDBMS can sufficiently handle “heterogeneous data” such as “images, text, and miscellaneous facts.” (Homan and Kovacs 2009, 211; after Codd, 2007).

Relational databases are tabular (see Figure 2.7) and tabular data (i.e. structured like a table, e.g. a spreadsheet) is considered only to be a semi-structured data representation due to the lack of consistent and explicit schemas to support automatic machine processing. Each discrete piece of data and its attributes (i.e. a row in a table) are locked into a table and the relationships (i.e. the joins, represented by arrows) are generalised between tables, not between rows. It requires general or domain-specific knowledge to interpret the semantic contents (Ristoski and Paulheim 2016).

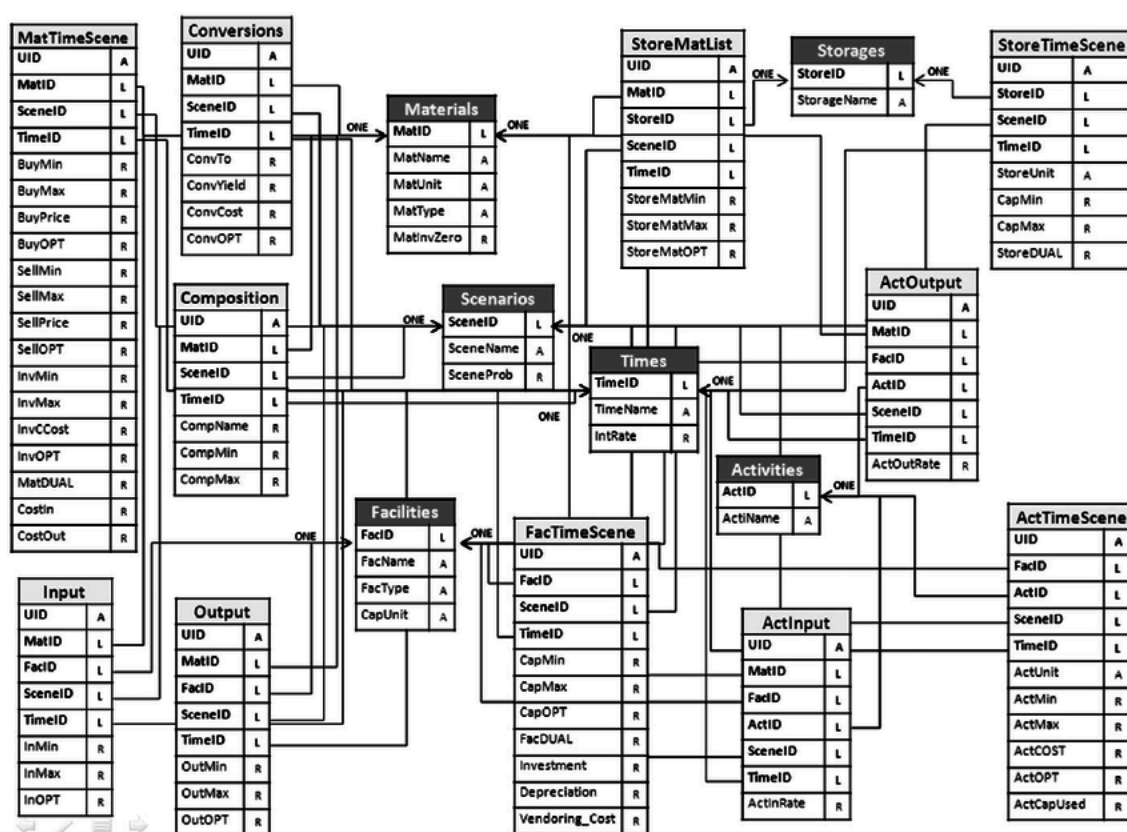


Figure 2.7. Visualisation of a RDBMS database structure (from Gupta et al 2014, Fig. 1). Note the primary tabular structure with joins (relationships) generalised between whole tables and not between rows or cells of different tables.

A SQL relational database is the backend technology that supports both open source database systems, such as PostgreSQL, and proprietary systems, such as those built on Microsoft Access, Microsoft SQL Server, MySQL (Roberts 2019) and Oracle systems (Carpinone 2010). For example, the publicly accessible Natural History Museum (NHM) Data Portal uses the open source PostgreSQL database in its technology stack (Scott et al 2019) while previously, the NHM in-house collection management system used EMu

by Axiell (Blagoderov et al 2017). EMu itself is based on Texpress which is similar to SQL in that it is an object-oriented data model but uses TexQL as the query language and runs on a Linux/Unix server. According to Blagoderov et al (2017), the Axiell system was considered to be “unsuited for rapid data entry and includes few tools for data processing” which led to the development of a separate in-house database named iCollection using Microsoft SQL Server database with an interface created via MS Access 2010.

Other query languages and their corresponding database technologies include XQuery for XML (eXtended Markup Language) databases, SPARQL for RDF (Resource Description Framework), and GQL¹³ (graph query language) for graph databases, to name a few. These do not use a tabular data model and are instead examples of *NoSQL* databases where schemas and relationships between discrete data entities can be more explicitly expressed. XML databases are document-oriented databases (or document store) while RDF bears characteristics of both XML and graph databases. In terms of FAIR compliance, XML, RDF, JSON, and CSV are general purpose formats that are automatically machine-readable (Wilkinson et al 2016). This research will utilise a property graph database (see more on Neo4j in Chapter 4, section 4.2) with data and metadata extracted from XML/RDF data sources (see Chapters 5 and 6) and tabular data sources (see Chapter 7).

The next and final section in this chapter will identify how undertaking data management and data science using a computational thinking framework can add value to documentation practice and the wider conservation domain. Chapter 3 will then present how graphs specifically can aid data management and data science by serving as a metamodel for this framework.

2.6 Extending Documentary Practice via a Computational Thinking Framework

The context of collections care can necessitate the aggregation of data over time and the recording of multiple semantic layers or semantic dimensions to represent plural perspectives with regard to an object or a collection itself (Pringle et al 2022). The recording of nonlinear, multifaceted, and even paradoxical information may arise as perspectives on cultural materials can change over time or parallel meanings and significances may come to light which can have direct practical implications for conservation, such as who can or cannot handle an object, how to handle or store it, and

¹³ <https://www.gqlstandards.org/>

what is or is not an appropriate use for the object, e.g. such as for public view or loan. An example case would be where an object is considered sacred, culturally sensitive, or 'active', and continues to have direct significance and influence upon their source communities. Frameworks and policies, such as *The Policy for the Care of Culturally Restricted Objects* at the Great North Museum: Hancock, Newcastle upon Tyne, England, are developed in collaboration with plural stakeholders to manage such sensitive considerations and both create and yield additional dimensions for documentation and conservation practice.

Cosmological or existential perspectives on reality may differ between originators of cultural materials, subsequent collectors, and the viewing public, thereby altering our understanding of such materials. For example, the *Digital Repatriation of Biocultural Collections: Rio Negro, Amazonia* project is a collaborative effort to foster a more symmetric relationship between indigenous and non-indigenous researchers and "build a pilot digital resource of biocultural collections, guided by the objective to promote the quality of life and integrity of the Indigenous territories in Amazonia" (Martins et al, n.d.; Martins 2021) The project's participating organisations include the Royal Botanic Gardens, Kew, Birkbeck, University of London, the Instituto Socioambiental, the Rio de Janeiro Botanical Garden, the Federation of Indigenous Organisations of Rio Negro, and the Berlin Ethnological Museum. Housed at Kew are cultural materials sourced from the Rio Negro communities and collected by the 19th c. botanist Richard Spruce. During a visit by project participants to the Royal Botanic Gardens, Kew, members of the Rio Negro communities clarified and informed fellow participants of points of divergence between the communities' perspectives and cataloguing practice, such as, that certain objects are considered as paired and therefore should be housed together (Sekulowicz 2022). This follows on from the view that cultural materials and natural entities are paired (Santos-Granero 2009). A further example was where an object categorised as a musical instrument in the collection was not perceived as such by members of the community (Sekulowicz 2022). Hence, it is recognised that how an object is catalogued or classified (i.e. encoded) depends on the context available to the cataloguer or classifier. Scifleet et al (2009) assert that encoding is itself a documentary practice:

Documentation is a highly contextualised activity that is intimately tied to the practitioner.

Therefore, encoding of conservation information by conservation practitioners is not only appropriate and necessary, it is imperative that the computational turn in

conservation empowers practitioners in the recording of multidimensional semantics as limited practitioner involvement remains an area vulnerable for contextual loss.

Such are the many conservation considerations which demand a flexible documentation system that can allow and support disambiguation as well as plurality, to ensure semantic representation can be accurately captured over time. Reframing conservation documentation with a conceptual thinking framework after Marciano et al (2019) allows the conservation profession to tackle the volume of content and variety of our documentation, while facilitating the study of ascribed complexity in conservation practice, knowledge, and decision-making.

Marciano et al (2019) demonstrated how a computational thinking framework can be applied to archival science, specifically, in six areas of archival practice (see Table 2.2, first column). It is not a far leap to apply a similar framework to conservation documentation practices (see Table 2.2, second column). But, firstly, what is computational thinking?

Table 2.6.1. Example Research Areas Supported by a Computational Thinking Framework

Applicable Areas in Archival Practice (source: Marciano et al 2019):	Example Adaptation for Application in Conservation Practice:
<ol style="list-style-type: none"> 1. Detecting identifiable information for historic persons 2. Developing name registries 3. Integrating vital records 4. Designing controlled vocabularies 5. Mapping events and people 6. Connecting events and people through networks 	<ol style="list-style-type: none"> 1. Detecting identifiable information for objects/collections 2. Developing object, treatment and/or material registries 3. Integrating vital records 4. Designing controlled vocabularies 5. Mapping events and objects 6. Connecting events and objects through networks

Digital skills are not limited to the knowledge and use of specific software or hardware products. At its core, digital skills utilise computational thinking as a problem-solving strategy. The four key methods employed in computational thinking are:

1. Decomposition (breaking down a problem),
2. Abstraction (removing extraneous information),
3. Pattern Recognition (identifying similarities with other problems or scenarios), and

4. Creating and Applying Algorithms (step-by-step instructions on how to solve a problem).

Marciano et al's work is based on Weintrop et al (2016) who proposed "to break computational thinking down into a set of well-defined and measurable skills, concepts, and/or practices" for high school mathematics and science education (ibid., 130). Weintrop et al identified 22 computational thinking practices and grouped them into four categories (Figure 2.8). While the authors acknowledge overlap between the practices¹⁴, there is a taxonomic usefulness to deconstructing computational thinking with the resulting framework providing a demonstrable progression towards the complex from the discrete.

Data Practices	Modeling & Simulation Practices	Computational Problem Solving Practices	Systems Thinking Practices
Collecting Data	Using Computational Models to Understand a Concept	Preparing Problems for Computational Solutions	Investigating a Complex System as a Whole
Creating Data	Using Computational Models to Find and Test Solutions	Programming	Understanding the Relationships within a System
Manipulating Data	Assessing Computational Models	Choosing Effective Computational Tools	Thinking in Levels
Analyzing Data	Designing Computational Models	Assessing Different Approaches/Solutions to a Problem	Communicating Information about a System
Visualizing Data	Constructing Computational Models	Developing Modular Computational Solutions	Defining Systems and Managing Complexity
		Creating Computational Abstractions	
		Troubleshooting and Debugging	

Figure 2.8. The Taxonomy of Computational Thinking Practices by Weintrop et al (2016) and adapted for archival sciences by Marciano et al (2019).

Weintrop et al's (2016) motivation for defining a taxonomy of computational thinking rested on:

- "the increasing use of computational methods to solve non-linear/non-deterministic (i.e. more complex) problems", and
- "to embed computational thinking within the subject context"

¹⁴ "Although we present our taxonomy as a set of distinct categories, the practices are highly interrelated and dependent on one another. In practice, they are often used in conjunction" (Weintrop et al 2016, p.134).

Data Practices	Modeling & Simulation Practices	Computational Problem Solving Practices	Systems Thinking Practices
Collecting Data	Using Computational Models to Understand a Concept	Preparing Problems for Computational Solutions	Investigating a Complex System as a Whole
Creating Data	Using Computational Models to Find and Test Solutions	Programming	Understanding the Relationships within a System
Manipulating Data	Assessing Computational Models	Choosing Effective Computational Tools	Thinking in Levels
Analyzing Data	Designing Computational Models	Assessing Different Approaches/Solutions to a Problem	Communicating Information about a System
Visualizing Data	Constructing Computational Models	Developing Modular Computational Solutions	Defining Systems and Managing Complexity
		Creating Computational Abstractions	
		Troubleshooting and Debugging	

Figure 2.9. The author uses the Taxonomy of Computational Thinking Practices (as adapted for archival sciences by Marciano et al 2019 after Weintrop et al 2016) to highlight (in blue) the extent of computational practices in the field of conservation at large.

By and large, the conservation profession actively employs the practices listed in blue in Figure 2.9. These practices are primarily localised to the first column, that of “Data Practices”, with a very limited example¹⁵ from the next adjacent column headed “Modeling and Simulation Practices”. Yet, as discussed in earlier sections, while there is an acknowledgement of the nonlinear nature and complexity inherent in conservation, Otero’s (2022) consternation at the lack of data scientific work in conservation is in direct relation to the lack of work in the remaining, unhighlighted practices depicted in Figure 2.9.

Applying a computational thinking framework identifies the trajectory for development in conservation that proceeds towards methods and practices for investigating and understanding complex systems. When viewed through this framework, the fragmented data landscape surveyed and described by Zorich (2016) evidently suffers from the lack of systems thinking practices and the modelling, simulation, and computational problem solving practices that contribute to countering digital landscape fragmentation and drives cohesivity of domain knowledge.

¹⁵ The only case of ‘Modelling and Simulation’ in conservation in the literature at the time of writing is The Linked Conservation Data (LCD) project (Velios & St John 2022; Campagnolo and Lieu 2022) where conservation data was mapped to the CIDOC CRM ontology resulting in machine-readable RDF graphs. The LCD project will be introduced in further detail in the later section 3.5.3 *Linked Conservation Data*.

Tal (2017) defines a “model” as:

an abstract and approximate representation of a local phenomenon, a representation that is used to predict (and sometimes also explain) aspects of that phenomenon.

While there is a recognition in conservation to investigate and understand complexity, the practice of constructing computational models specifically for conservation data has been limited with existing efforts (see section 3.5.3 below) situated largely within the scope of data integration while analysis and problem-solving using data is approached separately or viewed as subsequent to the integration process.

Such disconnected approaches have hampered advances in using wider data analysis techniques in the cultural heritage sphere, such as natural language processing (NLP). Sporleder (2010) has shown that NLP has been used to attenuate some of the challenges listed above, such as for error correction in databases and parsing free-text. However, similar hindrances such as non-standard terminology, free-text access issues, a variety of data formats, and a lack of domain-specific annotated training data creates a catch-22 situation where deployment of computational tools, such as NLP, lack the models (e.g. domain-specific training corpora) to drive adoption and maturation of the technology within cultural heritage. Without these machine-aided tools, creating domain-specific corpora becomes insurmountable due to the sheer volume of data necessary to manually interrogate and assemble such corpora. However, improved NLP technologies coupled with a graph-based analysis and a computational framework (this is demonstrated in Chapter 7 using data from The National Archives, UK) can support the creation of conservation-specific corpora, which in turn aids adoption, implementation, and integration of data science-derived insights.

In addition, data quality considerations are largely absent in existing data practices. Nevertheless, Kesper et al (2020) have been able to utilise graph “patterns to identify data quality problems independent of the underlying database technology and format”. Examples of *quality dimensions* (adapted from Laranjeiro et al 2015) are presented in Table 2.3. Data and data model verification, validation and calibration practices for conservation have also been absent from the literature, yet they are critical to reliability in data management.

Table 2.6.2 Examples of quality dimensions for research data as defined by Kesper et al 2020 (after Laranjeiro et al 2015)

Correctness	"the degree to which the data correctly represents the real-world values (semantic correctness) and is free of syntactical errors (syntactic correctness)"
Completeness	"the degree to which all required information is present in the data"
Consistency	"the absence of logical or representational contradictions within the data"
Precision	"describes how exactly the data represents real-world values"
Uniqueness	"the unambiguous interpretability of data and thus the absence of redundancies"
Understandability	"the ease with which humans can read and interpret the data"
Timeliness	"measures how up-to-date the data is"
Trustworthiness	"defined as the degree to which the data is accepted to be correct and credible"

Graham et al (2022) have identified the adoption of network analysis as the third wave of computational adoption by the digital humanities and, more specifically, in the field of history¹⁶. They argue that "the ability to retain information has been keeping up with the growing amount of generated data" (ibid., 52). Hence, "expectations have inverted. Everything may be recorded or preserved, at least potentially." (ibid., 52, quoting Gleick 2011). As a result, we must employ more computationally powerful tools to keep pace with the work. This comparison with how historical research methodologies have evolved is apt as historical research also deals primarily with text-based data. Strides have been made using network analysis for topic modelling¹⁷ (Graham et al 2022, 142-181) and visualisation (ibid., 100-141). The expectation for long term storage and use of digital records to communicate with future conservators, or even our future selves, presents documentation as history. Historical records analysis has itself evolved to leverage graph-based approaches. Documentation and computation have become

¹⁶ The first wave being the initial computational turn in the humanities, from punch cards to the world wide web, and the second wave being the adoption of text analysis (Graham et al 2022, 52).

¹⁷ See also p. 248 in the *Future Work* section 8.6.4 *Machine Learning and Deep Learning Models*.

intrinsically linked in practice elsewhere in the sciences and the humanities and it needs to be so in conservation.

In November 2020, the UK's ICON Heritage Science Group announced (Iconnect Special 2020) a new collaborative network dubbed *ConCode*, an initiative founded by four conservation professionals to establish a "community of coders that work in the cultural heritage field". The "initiative was created based on a growing need in the field to efficiently deal with large and complicated datasets. This has resulted in an increased interest in coding, data analysis, statistics, machine learning, and visualization" (ConCode 2021).

It is clear that the limited and slow progress of deriving and testing computational models is a hindrance and no longer an adequate position given the 'big data' demands of the conservation field today. Graph theoretic approaches not only include network analysis but provides the practical means to build models from conservation specialist knowledge and aid in the identification of patterns and anti-patterns specific to and representative of conservation practice. Chapter 3 focuses on graphs as the underpinning representational device and technology that supports both human cognition and machine-based computations and thus are highly conducive structures for encoding and computing conservation data and content.

3.0 Graphs

3.1 Mathematical Graphs and Graph Theory

A graph G is a mathematical model (Wilson 1996; Trudeau 1993; Diestel 2017) consisting of a set of *vertices* V , linked by a set of *edges* E , and is represented by the following notation:

$$G = (V, E)$$

An example of a simple graph would be a barbell structure: two nodes linked by one edge between them (Figure 3.1). The relationship between the sets of vertices and edges contributes to the definition of the particular graph. In figure 3.2, the set of vertices of the graph, that is $V(G)$, is $\{v, w, x, y, z\}$. The set of edges to the graph, $E(G)$, are $\{v,w\}$, $\{w,x\}$, $\{x,y\}$, $\{y,z\}$, and $\{z,w\}$. (Edges can also be represented using this alternative notation: vw, wx, xy, yz, zw). Thus, each edge is defined by a pair of vertices.

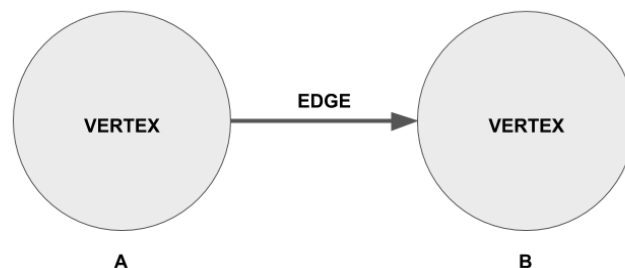


Figure 3.1. Example of a simple directed graph with one edge connecting two vertices, and the edge containing directional information.

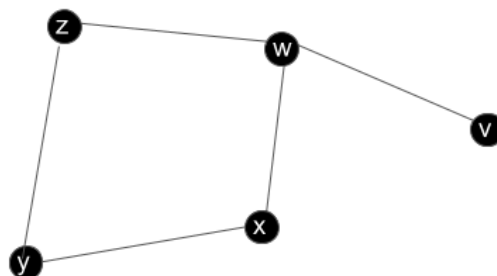


Figure 3.2. Example of a simple graph with five vertices and five edges.

Graph-based representations of connectivity are familiar constructs even to the general public at large. Many people will have been exposed to similar diagrammatic constructs such as flow charts, mind maps, and spider maps (for instance, the iconic London Tube Map). The diagram of folder structures by Barok, Thorez, et al (Figure 2.1) mentioned in the previous chapter and the *Map of Interactions* presented by Lawson et al (Figure 2.2) all fall well within this vein of diagrammatic information management and decision-making. They all have labelled nodes and connecting edges in the form of lines or arrows, and while the edge values are not necessarily made explicit in these diagrams, there are implied relationships. However, there is a latent opportunity to further leverage graphs in conservation, beyond diagrammatic purposes and towards computational analysis.

The *traversal* or *walk* as a problem-solving method stems from one of the earliest foundational elements of graph theory: *the Seven Bridges of Königsberg* problem. The problem refers to the seven bridges that span the Pregel River (now Pregolya River) in the 18th century which connected two river islands to each other and both banks of the city of Königsberg, Prussia (now Kaliningrad, Russia). The challenge was finding a route that involved crossing all of the bridges, but each bridge only once. In 1735, the mathematician Leonhard Euler presented a method for solving such a problem which involved abstracting the geographic locations by distilling the basic elements of the problem down to what later became known as the *nodes* or *vertices* (ie. the area of land where the traveller would arrive and disembark from between bridges) and the *edges* (i.e. the bridges themselves) (Powell and Hopkins 2015, 28). With this as the starting point, Euler demonstrated definitively that it was not possible in this particular case to find a route that crossed all seven bridges only once due to the configuration of the four areas of land (the nodes) in relation to how they connected with the bridges (the edges). In essence, each node (land) had several edges (bridges) connected to it. The number of connections a node has is known as the *degree*. In this case, each of the four nodes had an odd number of degrees which made it impossible to traverse without having to cross a bridge more than once. Euler's solution (Figures 3.3 and 3.4) demonstrated that how elements of the problem were connected had influenced the result and in doing so he provided a method to solve other similar problems by looking at the connectivity of the situation (Paoletti 2013).

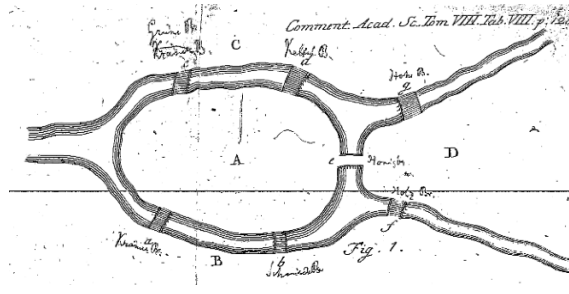


Figure 3.3. Euler's diagram of the 'Seven Bridges' problem. From Paoletti 2013. "Euler's Figure 1 from 'Solutio problematis ad geometriam situs pertinentis,' Eneström 53" [source: [MAA Euler Archive](#)]

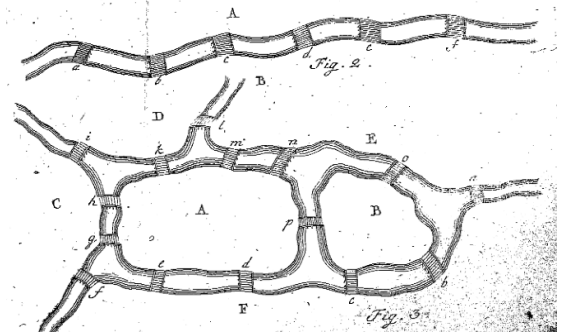


Figure 3.4. Euler's diagram representing how his methodology would still work even if there were more bridges and land masses. From Paoletti 2013. "Euler's Figures 2 and 3 from 'Solutio problematis ad geometriam situs pertinentis,' Eneström 53" [source: [MAA Euler Archive](#)].

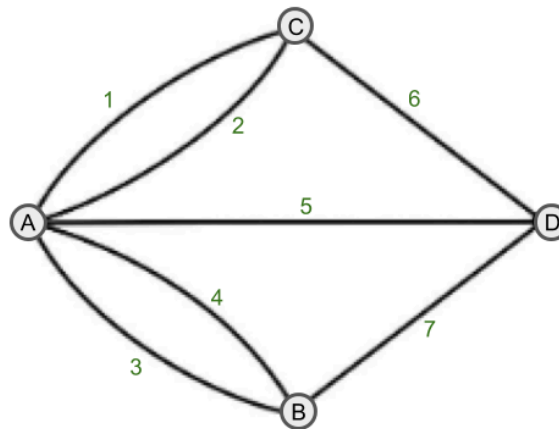


Figure 3.5. The Seven Bridges of Königsberg problem represented as a graph.

Since Euler's solution, graph theory has developed into a rich corpus of definitions, proofs, theorems, corollaries, and lemmas (Diestel 2016; Trudeau 1993; Wilson 1996) which in turn have contributed to the development of graph algorithms (Needham and Hodler 2019; Jungnickel 2013) with many practical applications in the study of *connectivity* and networks. Network analysis itself is considered to be synonymous with graph-based analysis. In fact, Pavlopoulos et al (2011) and Gros (2012) assert that graph theory underpins the study of complex networks. Figure 3.6 below is Castellani and Gerrits' (2021) depiction of the evolution of complexity studies. In a rather self-referential manner, the map is a graph through time, tracking various strands of research and development investigating various aspects of what the authors assert have contributed

to the complexity sciences.¹⁸ Although Castellani and Gerrits' *Map* representation is neither complete nor exhaustive and highlights some of the limitations in graph visualisations when restricted to two-dimensional planes. Nevertheless, graph theory is a highly-relevant method for investigating the nature of networks and complexity.

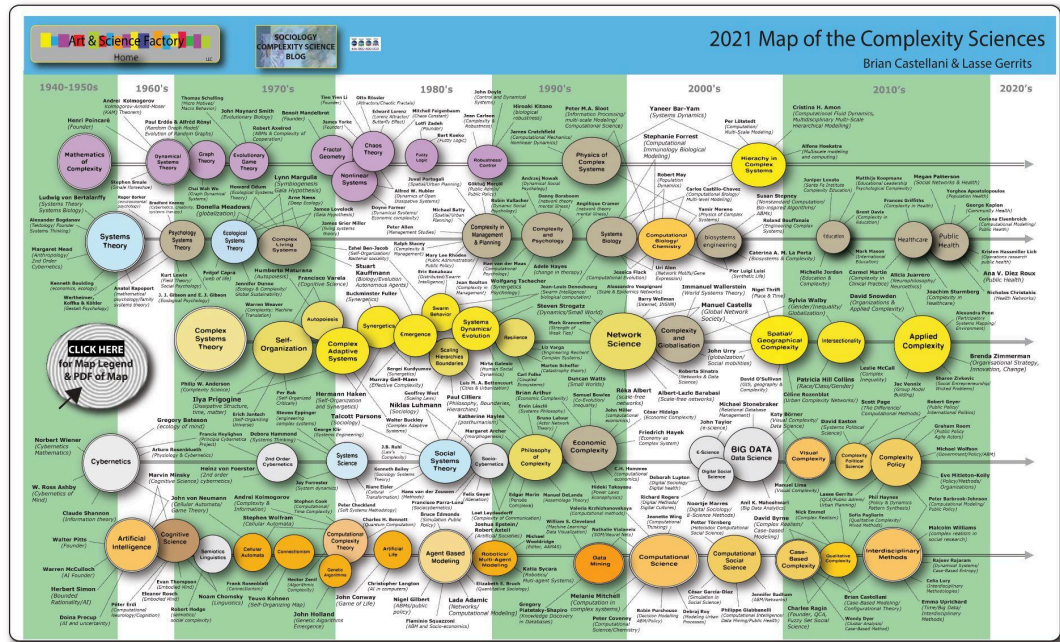


Figure 3.6. Map of the Complexity Sciences (Castellani and Gerrits 2021)

A *path*, as the name suggests, is a route through the graph along a sequence of nodes and edges. The *path length* is the number of edges in the path. The *shortest path* between two nodes is its graph *distance*. And the longest shortest path in a graph is known as the *diameter*. (Dale 2017, 65). A graph where nodes connect to more than one edge is known as a *multigraph*. Figure 3.5 above is a multigraph where node A has 5 edges connecting to 3 neighbours, or degree 3. A node with degree 1 is also known as a *leaf node*, and is indicative that it's at the end of a path, at the outer limit of the graph. *Branch nodes* have degrees greater than 1. Nodes that are not connected to other nodes, and therefore have degree 0, are *islands*. Being inherently unconnected, island nodes are not features within graph theory, however, their occurrence in real-world data is to be expected.

¹⁸ A note to the reader: Castellani and Gerrits provide explanatory notes (<https://www.art-sciencefactory.com/MapLegend.html>) and FAQs (<https://sacswebsite.blogspot.com/2021/09/q-for-2021-version-of-map-of-complexity.html>) that acknowledge limitations to their interactive and periodically updated diagram. The authors emphasise how *the Map* serves only as an introduction to certain topics and persons. They acknowledge key persons may be missing from the diagram and rely on hyperlinks on subject nodes to link to further information and favour inclusion of more recent researchers in subject areas over historically significant founders in some areas of *the Map*.

In *directed graphs*, or *digraphs*, the directionality of the edges also give rise to *in-degree* (number of incoming edges) and *out-degree* (number of outgoing edges) measures. A *triangle* is the smallest *cyclic graph*. Graph theory provides a rich language in which to describe networks and their features or patterns. If the conservation profession is to grapple with complexities, graph theory offers computational and linguistic assistance.

Large complex networks have been identified to have shared characteristics (Fagnani et al 2015; Chung and Lu 2006; Newman 2003; Newman 2000). One of these is the so-called *small world effect*, where despite a complex network's large size (i.e. a high number of nodes and connections), clustering occurs and some nodes will tend to be more highly-connected than others giving rise to what has come to be colloquially referred to as "six degrees of separation" thanks to Guare's eponymous play (1990). It is in fact a long-studied phenomenon (Korte and Milgram 1970) identified to be a feature of human communication and can be evidenced through graph representation (Newman 2000).

The shape or topology of a graph is another diagnostic feature in the study of complexity:

[Complexity] *arises because of the topology of the interconnection which enhance rapid diffusion of information and because of the non-linear and probabilistic nature of the interaction laws guiding the dynamics over the network. In these cases, the global behavior of the network shows complexity features which by no means can be seen as the addition of the many individual behaviors [i.e. is an emergent property].* (Fagnani, Fosson & Ravazzi 2015, 2)

Therefore, from the perspective of elucidating the complex nature of conservation, a graph-based approach is highly appropriate and strongly supported by the literature. Feature analysis, such as analysing for the small world effect, clustering, and revealing the topology of conservation documentation-derived graphs will contribute to clarifying the nature of complexity in conservation and the nature of its network of activities.

3.2 Diagrammatic Graphs in Cognition and Philosophy

Euler's use of abstraction as part of his methodology (e.g. labelling the land masses and bridges with numbers and letters) to solve the *Seven Bridges of Königsberg* problem allowed him to apply logic to the facts of the situation without distraction or unhelpful consideration of other details associated with the problem. This use of non-mathematical graphical representation to process information has a cognitive advantage according to Pinker (1990). His work on graph comprehension looked into the cognitive processes at play when we communicate and recognise visual representations. While Pinker's work focused on 'graphs' in terms of any visual representation (e.g. bar graphs, pie charts, etc.) it bears key points of relevance here and to node-and-edge-based graphs, which he called "visual arrays", as his means to deconstruct visual representations, a form of concept mapping (ibid, 78). When humans process maps and similar visual systems we tap into the highly-developed spatial awareness aspects of our neuro-cognitive functions (ibid; Lakoff and Johnson 1980). "[A] great many abstract concepts seem to be mentally represented by structures originally dedicated to the representation of space and the movement of objects within it" (Pinker 1990, 105). Indeed, Campagnolo's work (2015) on the visualization of historical bookbinding structures (an example is given in Figure 3.7) explicitly employs Pinker's "graph schemas" technique.

Graph representations using nodes to map perceptual input also enables cognitive understanding according to Gestalt laws of grouping (ibid, 83-85), that is:

Elements tend to be grouped perceptually if they are close together, similar to one another, form a closed contour or move in the same direction (Rock and Palmer 1990, 85).

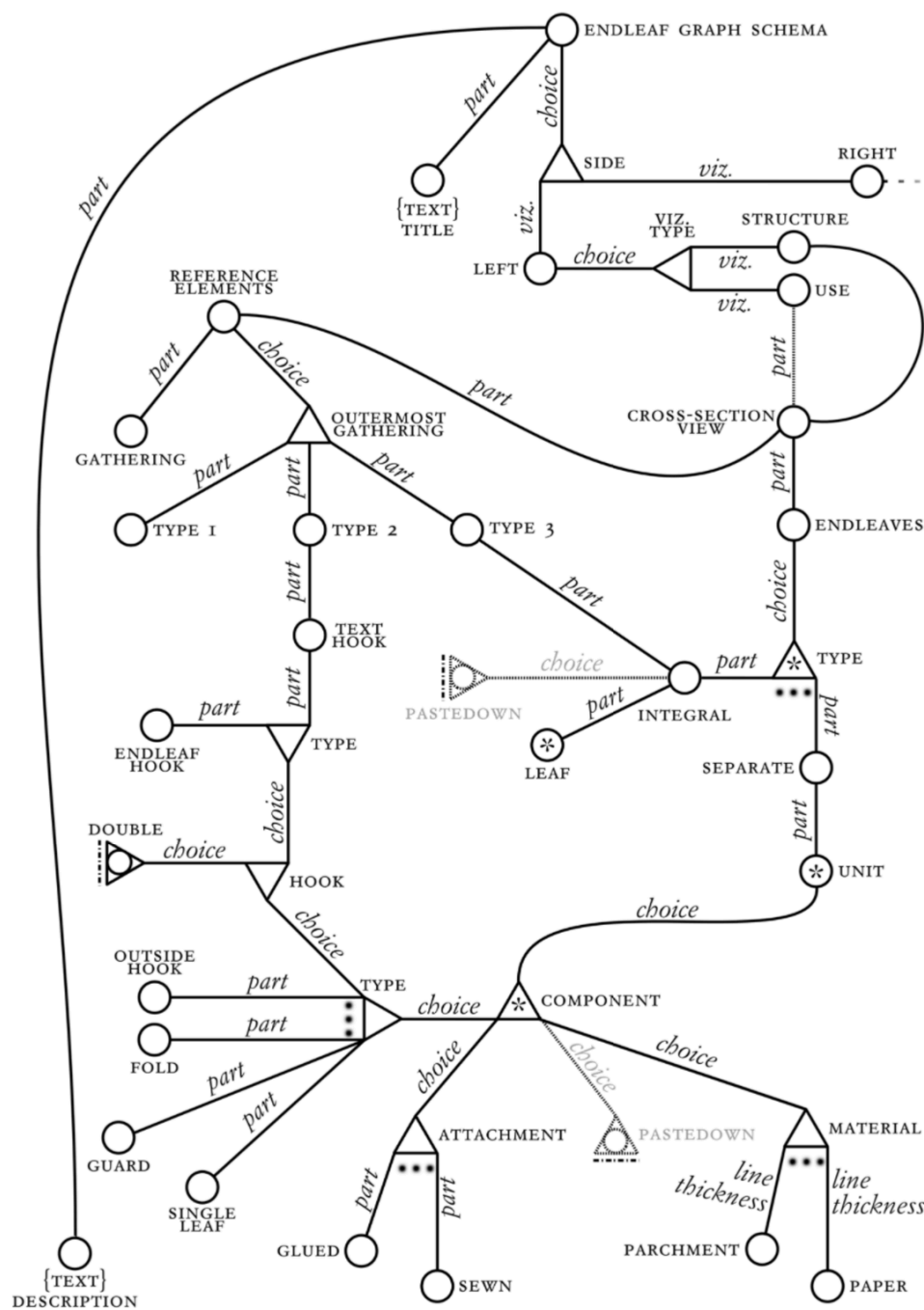


Figure 3.7 The graph schema for the construction of book endleaves including choice variations by Campagnolo (2015, vol. 2, p. 336).

As Newman (2003, 170-171) stated: “the human eye is an analytic tool of remarkable power, and eyeballing pictures of networks is an excellent way to gain an understanding of their structure”. Thus, mapping information contributes to our understanding of it. Albeit, errors in mapping, for example, in the abstraction process or in generating the schema (i.e. identifying the framework for how things are related) can lead to misrepresentations and misunderstandings. Gestalt-based decision-making is vulnerable to errors (Cook 2009), such as over-reliance on perceptual heuristics and confirmation

bias. Although derived errors from misrepresentations, such as those arising from unconscious bias, are challenges to conceptual representation, explicit mapping practices that are transparent and open for scrutiny aid the identification and correction of errors, and allow schema to be regularly assessed and reassessed, yielding a means to track the evolution of research and practices in abstraction to aid knowledge discovery and mutual understanding. Ultimately, graph representations are *models*. To paraphrase the statistician George Box: all models are wrong, but some are useful. As explained in Chapter 2, models are critical components to computational ways of working, particularly when striving to understand complex systems and to work collaboratively. Knowledge representation systems using graphs (e.g. RDF and property graphs, see sections 3.3) consist of explicitly declared nodes and relationships (see 4.3.2, tables 4.3 and 4.4) in their encoding, thereby minimising ambiguity.

In fact, having acknowledged the aforementioned (in Chapter 2) Barok, Thorez, et al's folder structure diagram (Figure 2.1) and Lawson et al's *Map of Interactions* (Figure 2.2) as graphs, in terms of visual representations, conservators are already familiar with graphs, even though the terminology is not common parlance. The benefits of graphs clearly go beyond their diagrammatic value because there are limitations to visual representation and visual analysis alone. At certain resolutions, important details may be left out of visualisations, for example, such as relationship labels¹⁹. Nevertheless, when used in knowledge representation systems (further details in sections 3.3 - 3.5), the relationships are explicitly encoded and queryable. However, as strictly visualisations, when a graph becomes too dense and too large (due to millions of data points, for example) this results in the "fuzzy ball" model that is difficult to discern or create without computer assistance. Here, the application of graph theoretic and probabilistic approaches can articulate hidden patterns within the "fuzzy ball" (Newman 2003). Both cognitive and computational models can exploit the same graph structure which has its advantages and efficiencies when crafting human-readable and machine-readable systems. "Graphs are conceptually simple, flexible, and intuitive for human users while being compatible with computational processing" (Bales and Johnson 2006, 459-460).

Hierarchical tree structures and networks are types of graphs. The branching tree is an acyclic graph, where paths along the branches have a terminus and do not connect back

¹⁹NB: Purely diagrammatic graphs may have implied relationships whereas encoded knowledge graphs will always have explicit relationships. Figures of visualised graphs of the case study datasets in this thesis may appear to be without labelled relationships for visual clarity on the page, however, the relationship labels are explicitly stated in the encoding (see Appendices). Relationships will be referred to with either the prefix "Pxx" (where "x" are numbers) in reference to CIDOC CRM relationships (RDF properties) or enclosed with square brackets as per Cypher syntax to denote relationship labels in a labelled property graph representation.

to any previous node or junction (Figure 3.8). On the other hand, a network is a cyclic graph (Figure 3.9) as it is possible to traverse around and back to a previously visited node without doubling backwards along the same route. The abstractive affordances of tree diagrams and network diagrams are not in contention with each other, but sit together within graph theory.

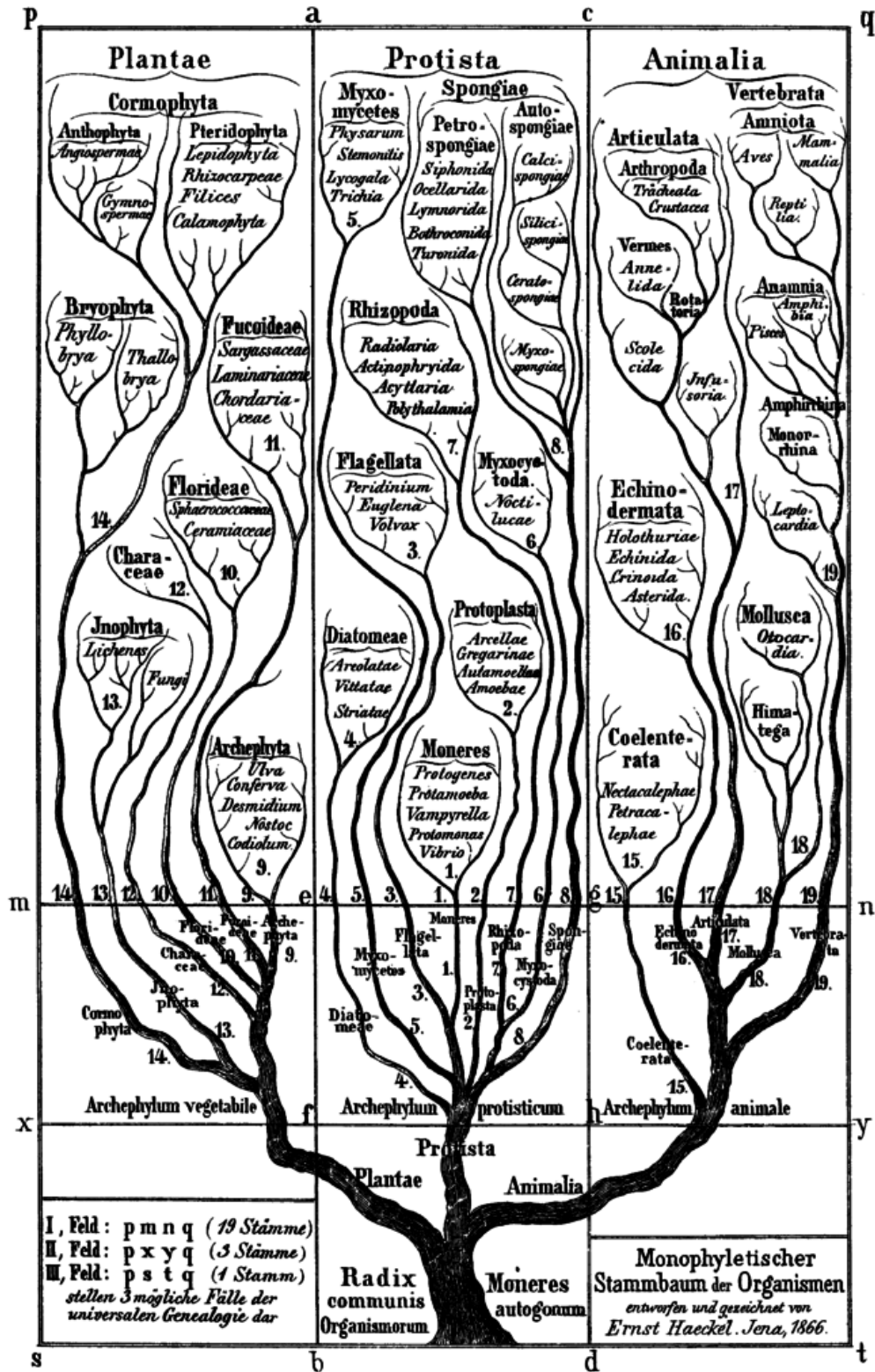


Figure 3.8 Haeckel's original (1866) conception of the three kingdoms of life as a tree (acyclic graph). Image source: https://en.wikipedia.org/wiki/File:Haeckel_arbol_bn.png

For example, the tree of life diagram by Haeckel (figure 3.8) shows taxa belonging to three compartmentalised branches of life (*plantae*, *protista*, and *animalia*) but not how they interact, whereas the marine food web diagram (figure 3.9) of predation depicts a system that is highly interactive amongst its components. Both diagrams can be represented as graphs. The tree of life diagram can be redrawn with taxa as nodes and hereditary or genetic links as relationships while the food web diagram can be redrawn with marine species as the nodes while the relationships can be more clearly encoded with various types of predation labels.

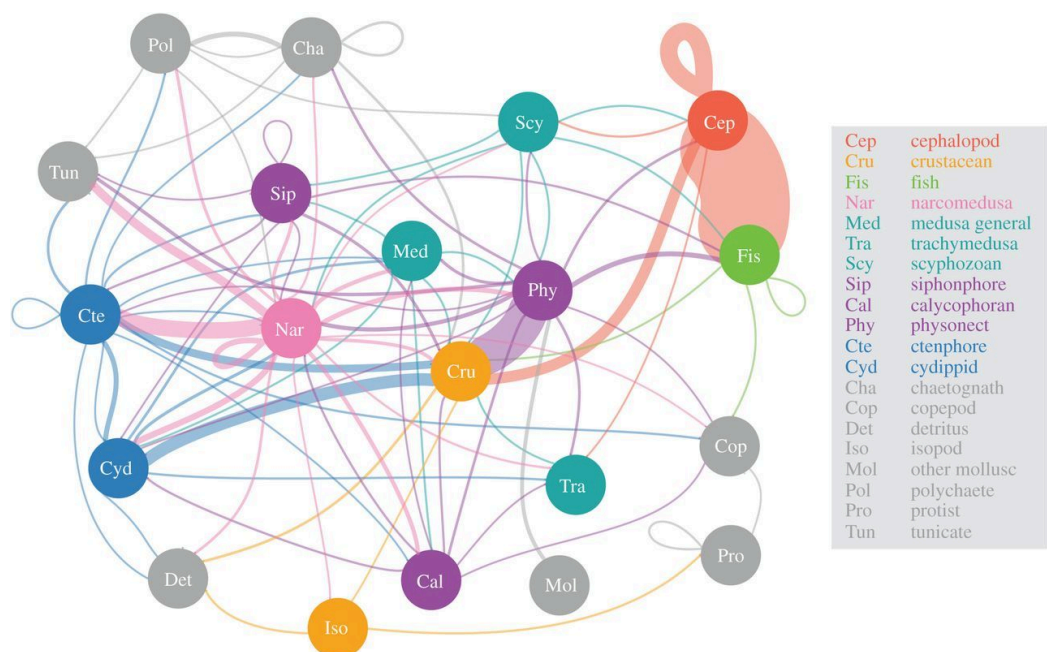


Figure 3.9 Marine food web network (cyclic graph) in the pelagic zone (Choy, Haddock and Robison 2017) Image source: https://en.wikipedia.org/wiki/Marine_food_web#/media/File:An_in_situ_perspective_of_a_deep_pelagic_food_web.jpg

The next two subsections will further connect the structural relevance of graphs to human cognition and understanding. Firstly, structure mapping theory is presented to explain a principally graph-based cognitive phenomenon in reasoning, particularly, analogical reasoning. Secondly, the philosophical concept of 'rhizomes', as originated by Deleuze and Guattari, will be briefly discussed in order to highlight the potential pitfalls of overfitting graph theoretic features to 'rhizomic' philosophy at this early stage.

3.2.1 Structure Mapping Theory

Structure Mapping Theory, as put forward by Gentner (1983), describes the implicit interpretation rules people apply when reasoning by analogy, that is, when using what one knows of one domain and applying it to make sense of another. In fact, Structure Mapping Theory [SMT] has been used to identify a cognitive continuum found across literal similarity, analogy, and abstraction (Gentner 1983, 161). Since Gentner's seminal work, there have been empirical findings to support structural mapping as a feature of cognition (Gentner & Asmuth 2019; Christie & Gentner 2010; Wolff & Gentner 2011). In addition to its influence in the fields of psychology and the cognitive sciences (Anderson & Lebiere 2014; Gentner & Maraville 2018; Gentner and Bowdle 2008), SMT has also been influential in computer science, particularly, in artificial intelligence (Torrey and Shavlik 2010; Crouse et al 2021) through the works of Gentner's early collaborators, Falkenhainer and Forbus, whom together they devised the Structure Mapping Engine (Falkenhainer, Forbus & Gentner 1986; 1989). While the Structure Mapping Engine used LISP²⁰ (i.e., linked lists as main data structures), SMT itself was originally framed in graph-based knowledge representational terms:

Knowledge is represented here as propositional networks of nodes and predicates. The nodes represent concepts treated as wholes; the predicates applied to the nodes express propositions about the concepts.

The means by which humans are able to harness analogical reasoning to learn, argue, and derive new knowledge, according to the conceptual frame of structure mapping theory is:

No matter what kind of knowledge (causal models, plans, stories, etc.), it is the structural properties (i.e. the interrelationships between the facts) that determine the content of an analogy. (Falkenhainer, Forbus, & Gentner 1989, §2)

Thus, through SMT, there is the acknowledgement of a core structural component to cognition including knowledge and reasoning, and that structure is a graph.

²⁰ The reliance on LISP or list-based data structures at the time, are in fact, compatible with today's graph-based systems in that lists are arrays and graphs are constructed via adjacency matrices, which are themselves two-dimensional arrays. Therefore, there can be consistency in the abstraction and direct potential for transformations in the implementations.

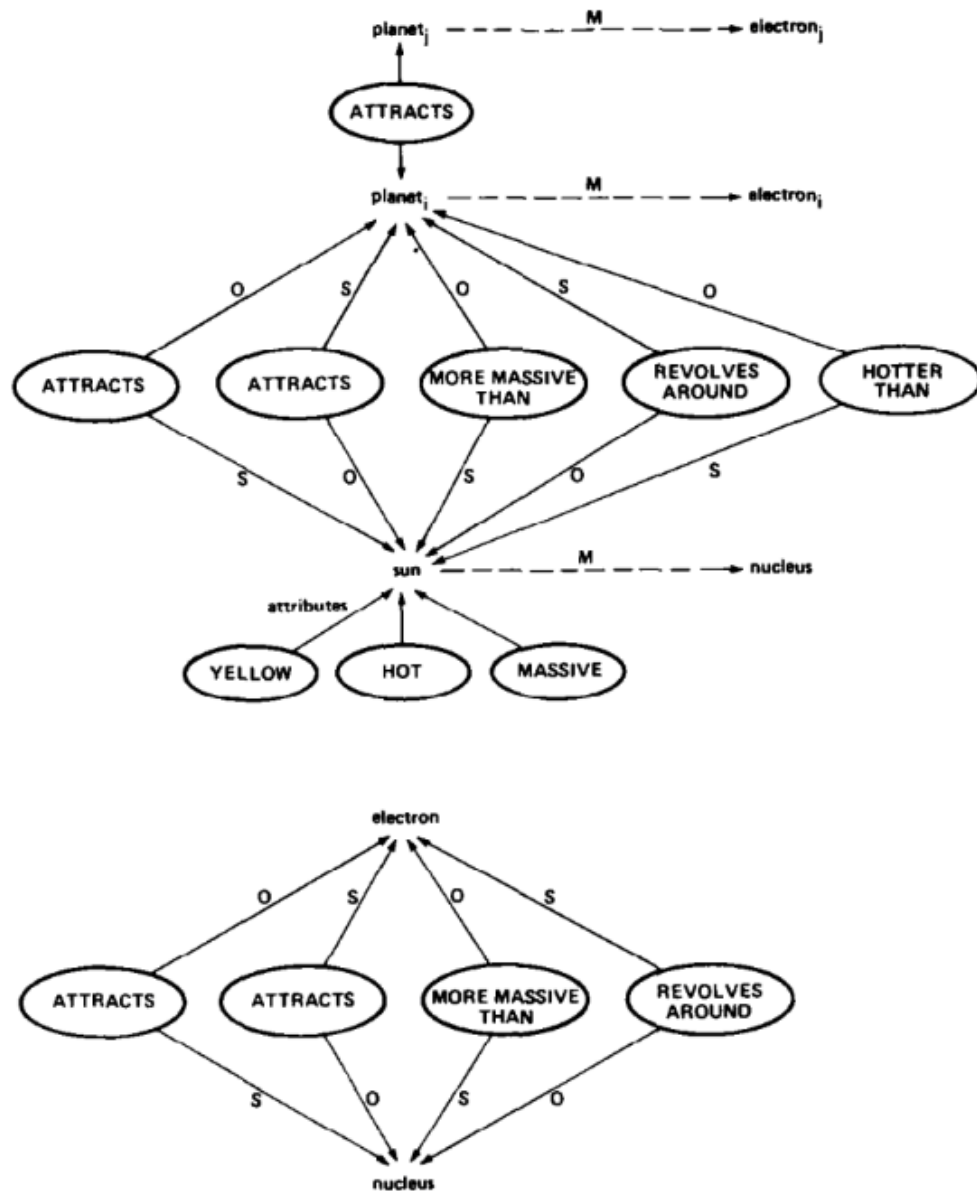


Figure 3.10 Structure mapping diagram from Gentner 1983 (160, Figure 1): Structure-mapping for the Rutherford analogy: "The atom is like the solar system."

3.2.2 Deleuze and Guattari's Rhizomes

A discussion of graphs (networks) within the arts, humanities, and social science contexts requires acknowledgement of the similarities between graphs and Deleuze and Guattari's conceptual *rhizome* (1980) and clarification for where the two differ. The *rhizome* is a metaphor derived from the botanical rhizome - a specialised stem structure that tends to run horizontally and can send out its own roots and shoots. It is in the botanical rhizome's lateral growth and non-hierarchical appearance from which Deleuze and Guattari construct their conceptual *rhizome* to explain language, text and politics

(Coyne 2008, 553) and characterize it as a tool for mapping connections, heterogeneity, “non-signifying rupture, ...and decalcomania” (Colombat 1991; Deleuze and Guattari 1980, 13- 20). Coyne (2008) offers a comparison and delineation between networks and *the rhizome* with their principal overlap located in their agnostic and diagrammatic conceptual functions. While there may be potential to investigate whether querying and graph algorithmic-derived results would offer avenues for the transposition of the latter characteristics (i.e non-signifying rupture, such as de-labelling, and decalcomania, a further metaphor for tracing or transference named after the process of transferring designs from prepared paper on to glass or porcelain), such studies are absent in the literature and is beyond the scope of the current work.

Consideration of *the rhizome* here acknowledges and reflects its continued influence in the arts and humanities including but not limited to philosophy (Colombat 1991), the social sciences and anthropology (Shaw 2015; Bell and O’Hare 2020), pedagogy (Gravett 2019; Honan 2007), and design (Coyne 2008; Purcell 2013; Hillier 2021). Where rhizomic thinking and action precipitates artistic and cultural works, this has direct consequences for the conservation and documentation practices appropriate for such works, for example, by the challenges to conservation theory posed by emerging media (Cull 2011). Thus, it further emphasizes the broad applicability and potentiality for diagrammatic graphs and the need for graph-awareness in conservation epistemology. As Coyne (2008) states:

Networks may not be the same as rhizomes, but talk of networks is rhizomic, subject to the vagaries of interpretive practice, contexts, historical conditions, contingencies, and disruptions. Networks are neither tangible referents nor immutable schemas of signification, but discursive devices to be adopted or discarded as needed, and in keeping with their shifting authority, a position that accords with the pragmatics of any representational schema (words) in language (Coyne 2008, 560).

The remaining sections in this chapter will focus on the representational schemata of computed and computable semantics.

3.3 Graphs in Knowledge Representation and Semantic Networks

The previous two sections have demonstrated how graphs have assisted thinkers in epistemic study. Therefore, it should be of no surprise that graphs feature in the seminal

development of artificial intelligence. The foundational concept of Minsky's *frames* continue to underscore development in artificial intelligence to which he defined as graph representations: "We can think of a frame as a network of nodes and relations" (1975).

Subsequently, the invention of *conceptual graphs* by John Sowa (1984) used diagrammatic graphs for knowledge representation and computer-aided knowledge organisation (Figure 3.11). In a 2018 interview, Sowa explains he designed conceptual graphs for "mapping language to logic" (Kyndi 2018) which contributed to the further development of artificial intelligence through the building of derivable semantics, e.g. 'semantic nets', and the mapping of data flow diagrams (Sowa 1987; Rassinoux et al 1998).

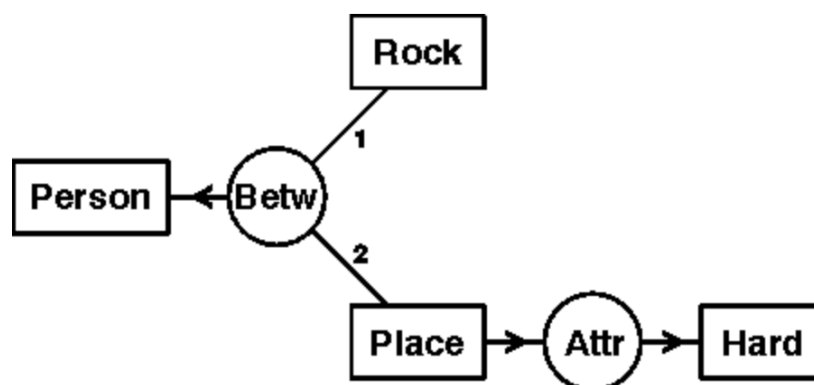


Figure 3.11 Example of a conceptual graph of the phrase "a person is between a rock and a hard place" by John Sowa (<http://www.jfsowa.com/cg/cgexampw.htm>). The instances of concepts [in rectangles] have relations (in circles).

Homan and Kovacs (2009) proposed a more associative data model where "data items [are] in nodes with links represented by vectors or lines with arrows connecting the items" (ibid., 211), or in other words, a graph model as an alternative to the relational data model.

Furthermore, cultural heritage professionals have been a target user group for developments in graph-based modelling techniques, particularly within the context of applicability and relative ease of adoption of conceptual graphs and UML (Unified Modeling Language) by archaeology doctoral candidates (Hug and Gonzalez-Perez 2012). However, there has not been any likewise interest or developments reflected in the conservation literature beyond the limited usage as visual representations as stated above.

The field of artificial intelligence (AI) has since expanded into a spectrum of technologies ranging from highly-human-mediated to highly-computer-mediated ones with the role of graphs featuring throughout from knowledge representation to large-scale pattern matching. The semantic knowledge representation discussed thus far is largely human-mediated. Nevertheless, the structure and form of the proposed knowledge graphs are conducive for use to support automated semantic knowledge representation and pattern matching via machine learning.

Semantics as a branch of study is concerned with meaning, particularly, the relationship between signifiers (e.g. words, symbols) and meaning (Allemang and Hendler 2011; Hollis and Westbury 2016). Davis et al (1993, 7) differentiated a semantic net from a graph with the former defined as a representation and the latter as a data structure. However, by the 21st century, while the distinction remains, as graphs continue to play a foundational role in semantic technologies and enterprise data solutions, the terms have come to be used interchangeably. In a 2007 blog post, Tim Berners-Lee revealed an alternative name for the Semantic Web, musing it “should have been Giant Global Graph!”. His comment refers to the underlying data model and interchange standard known as the Resource Description Framework (RDF) (see Figure 3.12). In 2004, the World Wide Web Consortium (W3C), who develops Open Web Platform standards, formally recognised the value of the graph model through the specification of the RDF graph, which is a “directed-arc” graph (Hayes and Patel-Schneider 2014; Cyganiak, Wood, and Lanthaler 2014), also known as a directed edge-labelled graph (Hogan et al 2021). SPARQL²¹, which is the RDF querying language, “is essentially a graph-matching query language” that looks for triple patterns (Perez et al. 2006, 98).

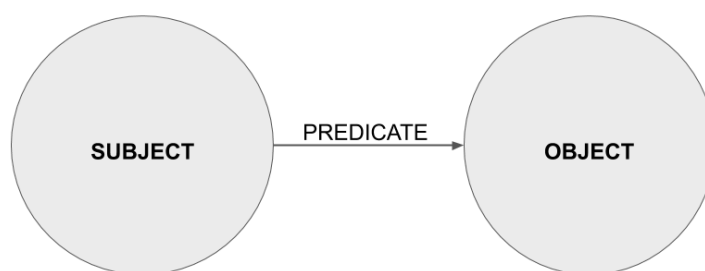


Figure 3.12 The Subject - Predicate - Object “triple” structure of RDF.

Linked Open Data (LOD) supports knowledge discovery via “a publicly available interlinked collection of datasets from various topical domains” (Ristoski and Paulheim

²¹ SPARQL is a recursive acronym for SPARQL Protocol and RDF Query Language and is pronounced like the word “sparkle”.

2016; who cites Bizer et al 2009; Schmachtenberg et al 2014). The Linked Open Data Cloud²² demonstrates how linked RDF data can be browsed as a graph. Figure 3.13 below shows the generated, browser-based interactive graph of interlinked publicly available LOD datasets with the various node colours representing the different domains from which the datasets originate. For example, red is from the life sciences while yellow is from publically available government data.

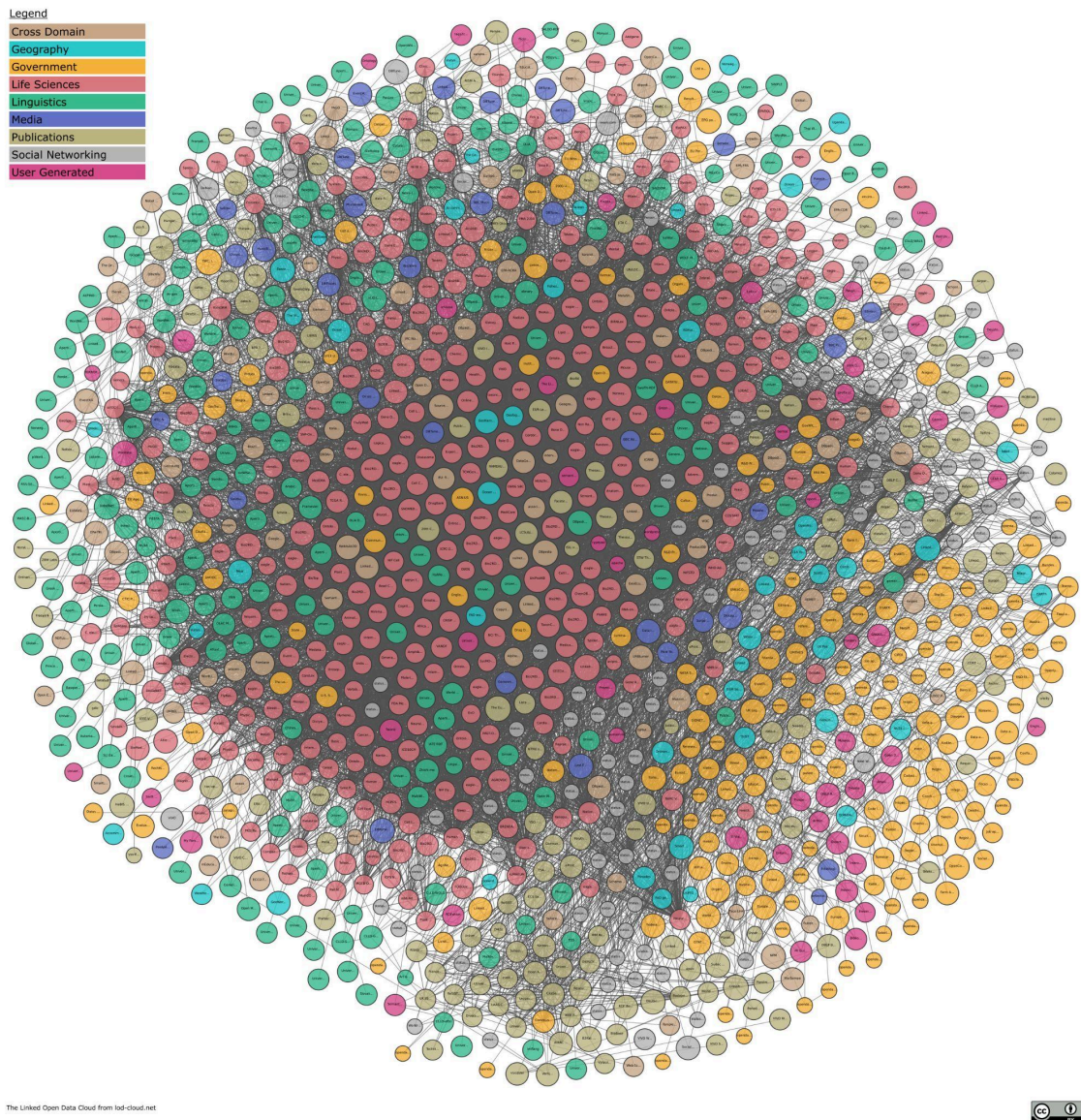


Figure 3.13 The LOD Cloud graph (generated 6 May 2023).

Another LOD visualiser is the LoDLive project (Camarda, Mazzini, and Antonuccio, 2012) (Figure 3.14). By choosing the DBpedia resource, which is the RDF version of Wikipedia, this author was able to use LoDLive to find incidental conservation-related topics, such as *Paraloid B-72* (teal node near center, in Figure 3.14 below), and their relationships to

²² <https://lod-cloud.net/>

the conservation-specific document space on Wikipedia. A layperson would be able to deduce from these connections based on the node labels alone, that:

- Paraloid B-72 is an adhesive,
- It is used in Conservation and Restoration,
- and falls under subjects of Art History, Cultural Heritage, and Museology.

OWL (Web Ontology Language) relationships provide further semantic richness to the DBpedia nodes in this case, for example, the “sameAs” relationship between the English language wikipedia page on Paraloid B-72 (teal node nearest centre) and the Italian language Wikipedia page for Paraloid B-72 (yellow node).

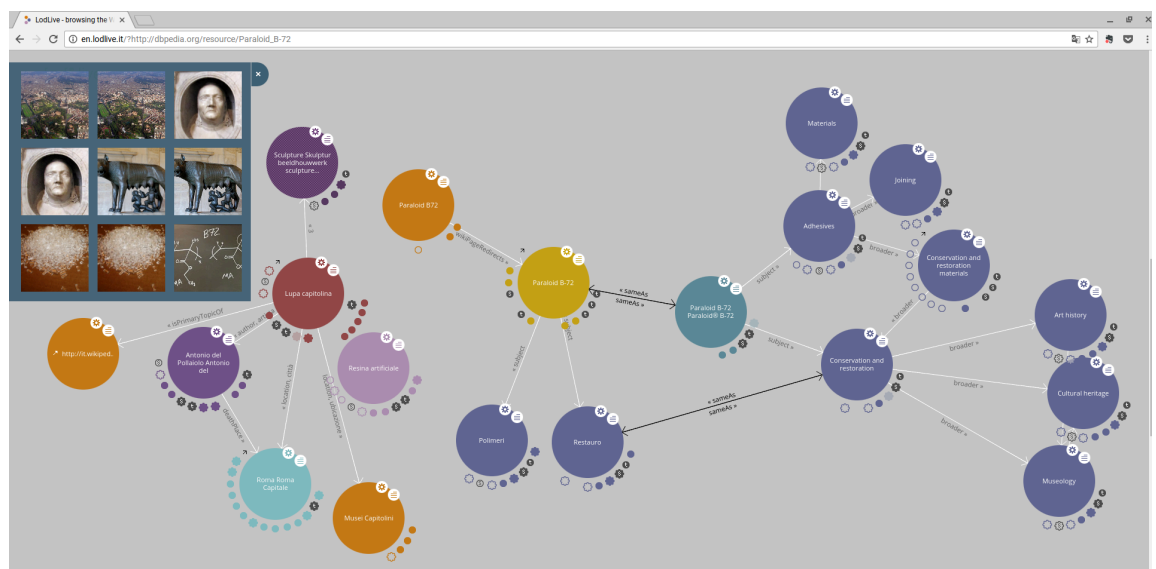


Figure 3.14 The DBpedia data graph produced by the LodLive project expanding from the English keyword node for “Paraloid B-72” (teal-coloured node to the right of centre), http://en.lodlive.it/?http://dbpedia.org/resource/Paraloid_B-72. (Accessed 2020.01.31)

Zeng (2008) provides a useful summary and representation of the many approaches to structuring or organising knowledge. As Zeng shows (Figure 3.15), taxonomies and ontologies reside on a spectrum of techniques for structuring data through metadata using a range of controlled vocabularies to formalise relationships between entities (Haynes and Vernau 2019). These manifestations of structured data can contribute to enriching a graph, while the use of ontologies to automatically infer new knowledge is considered a form of artificial intelligence (different to Machine Learning and Deep Learning).

A Taxonomy of KOS

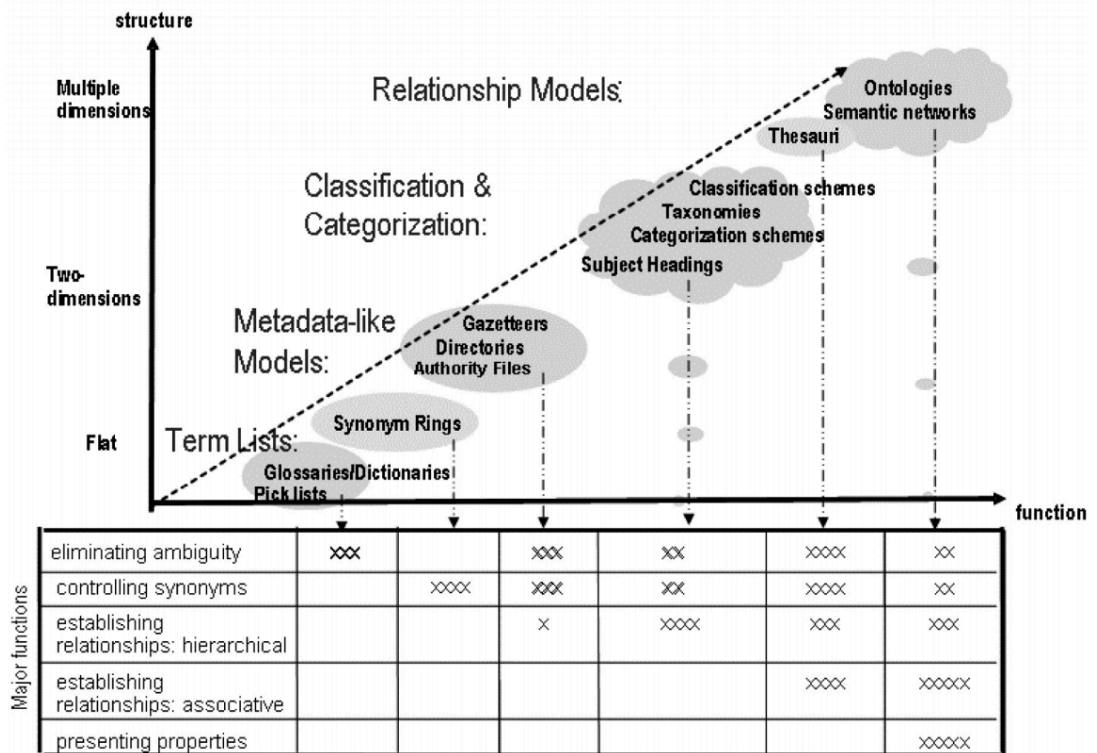


Figure 3.15 An overview of the structures and functions of KOS [knowledge organization systems] from (Zeng 2008, p. 161, Figure 1).

Sowa defined knowledge representation as “the application of logic and ontology to the task of constructing computable models for some domain” (2000). An ontology is a conceptualisation of a domain of knowledge that is explicitly specified (i.e has formal definitions and structures) “which is not limited to a taxonomy or a set of (conservative) definitions, but may also contain knowledge about the world” (Schneider and Simkus 2020, 329). In other words, as depicted in Zeng’s taxonomic hierarchy of knowledge organisation systems (Figure 3.15 above), ontologies are positioned at the upper-right corner of the complexity trend line, akin to semantic networks, or semantic graph representations. They are more complex than strictly hierarchical taxonomies of entities or classes with *properties* (relationships and attributes of classes) also represented (encoded) within the network structure.

Building on from Minsky’s (1975) frames, ontologies are situated in the subfield of ‘symbolic AI’ (Smolensky 1987; Angelov, et al 2021) where methods are human-readable. Ontologies serve three key purposes: 1) they are representations of knowledge (i.e. models) and 2) they provide a system for automated reasoning, which when combined 3) supports data management and data integration across

heterogeneous data sources, also known as Ontology-based Data Access (OBDA) (Poggi et al 2008).

As reasoning systems, ontologies enable automated reasoning and computable inference and must be able to carry out reasoning tasks such as:

consistency checking, satisfiability testing, or classification, but there are also tools that solve more advanced relevant reasoning tasks such as query answering, module extraction, forgetting, explanation generation, abduction, etc. (Schneider and Simkus 2020, 330).

The specificity, sophistication or complexity of a reasoning system can vary and therefore there can be multiple ontologies developed and used across a subject domain, for example, as of January 2023, Stanford University's National Center for Biomedical Ontology's BioPortal (Whetzel et al 2011)²³ holds 1,047 biomedical ontologies.

The W3C specification for the Web Ontology Language (OWL) sets the standard for how to encode ontologies so that the aforementioned reasoning tasks can be performed across datasets that are mapped to these ontologies (i.e. for interoperable inference).

A user of the system poses a query using the vocabulary of the ontology, i.e., over the conceptual view. The OBDA system is then tasked to answer the user query by incorporating the information from the various information sources, possibly employing the domain knowledge in the ontology to infer new information (Schneider and Simkus 2020, 332).

The combined use of graphs and the semantic web to manage and harness data have been expounded by Powell and Hopkins (2015) in their volume *A Librarian's Guide to Graphs, Data and the Semantic Web*. In it, Powell and Hopkins present librarians as experienced network navigators, employing graph theory approaches for information seeking in terms of proximity (distance) to identify relevance by using metadata and citation relationships, as opposed to content (Powell and Hopkins 2015, 103), although greater availability of computing power and more sophisticated search algorithms have also enabled searching via unstructured content. Nevertheless, despite the ubiquity of search and search engines, information management best practices are necessary and

²³ <https://bioportal.bioontology.org/> Accessed 20 January 2023.

relevant to efficient search functions, particularly in terms of structuring unstructured data. This has been the purview of library and information sciences where discoverability has always been at the fore in knowledge organisation (Haider and Sundin 2019). This demonstrates how graphs can be used to represent and support knowledge organisation systems.

Zeng’s overview of knowledge organisation systems can be correlated with Bellinger et al’s (2004) interpretation of Ackoff’s (1989) DIKW model (data → information → knowledge → wisdom) in systems thinking. That is, structure is to connectedness as function is to understanding. Malhotra and Nair (2015) provide a detailed summary of the evolution of knowledge representation systems from ca. 1948 to ca. 2010s that reflect this. Although Zins (2007) has highlighted criticisms of the DIKW model due to its restrictive hierarchical implications on human learning, nevertheless, from the perspective of machine-based inferencing, the two representations align, albeit with “wisdom” inferred as evermore connected nodes, and not as equivalent to human wisdom. Duan et al (2017) have even proposed identifying progressive graph development in terms of the DIKW model: data graph, information graph, knowledge graph, and wisdom graph. Progressing understanding from discrete data elements to increased connectivity within a repository helps to capture and represent context and aid in sense-making and comprehension.

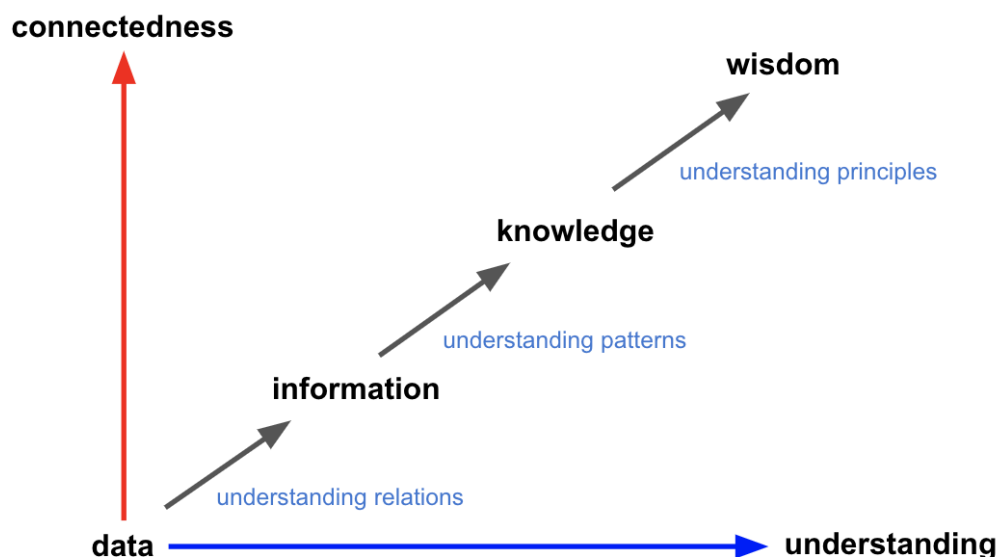


Figure 3.16 The DIKW model based on the diagram by Bellinger et al 2004.

A graph-based and semantically-enhanced approach to data management can facilitate ongoing and long-term data and research accessibility in line with FAIR principles (Wilkinson et al 2016). The importance of incorporating FAIR data principles into

conservation activity was recently discussed at an Icon Heritage Science Group seminar (Brown 2019). This undoubtedly has a direct bearing on the utility and content of conservation documentation systems to be FAIR compliant. Natsiavas et al (2018) have demonstrated that “graph-based articulated knowledge significantly enhances the capabilities of linking, sharing, and automatically processing” and therefore supports compliance with FAIR principles as part of the data model design.

3.4 Knowledge Graphs

3.4.1 Definition of a Knowledge Graph

Wilcke et al (2017) identifies the knowledge graph as “the default data model for learning on heterogeneous knowledge” in data science and identifies it as where knowledge is encoded on the Semantic Web (2017, 42). Furthermore, as they are task-independent, “the same knowledge graph can be used for many different tasks” (ibid, 55).

Most recently, the term *knowledge graph* has been associated with Google Knowledge Graph (Singhal 2012), presenting an impression that Google coined the term and is used in association with Big Data harvesting or in reference to openly available web-scale repositories such as DBpedia, YAGO and Freebase (Paulheim 2016). In actual fact, a key stage in the history of knowledge graph research began in 1982, with a joint project at the University of Twente and University of Groningen to devise systems for representing scientific theories to aid information retrieval, decision-making, and instruction in the medical and social sciences (Nurdiati and Hoede 2008; Stokman and de Vries 1988). Hoede (1994) based the development of knowledge graphs firmly in graph theory. However, nearly 40 years on, definitions put forth by the semantic community for “knowledge graph” are varied and inexact when compared.

Paulheim (2016) generalises it to include any graph-based knowledge representation. Whereas, Ehrlinger and Wöß (2016) compiled several of these definitions (reproduced in Table 3.1) in order to identify a common understanding of the term “Knowledge graph”, particularly if there is any distinction between knowledge graphs and the Semantic Web. A criticism they posed of two of the selected definitions (namely that presented in the Journal of Web Semantics and provided by “a Semantic Web Company”) is that the definition “could equally well describe an ontology or – even more generally – any kind of semantic knowledge representation and do not even enforce a graph structure” (ibid).

Ultimately, Ehrlinger and Wöß concluded: “the Semantic Web could be interpreted as the most comprehensive knowledge graph, or – conversely – a knowledge graph that crawls the entire web could be interpreted as self-contained Semantic Web”.

Table 3.4.1 Selected definitions of “knowledge graph” as collated by Ehrlinger and Wöß (2016).

Definition	Source
<i>“A knowledge graph (i) mainly describes real world entities and their interrelations, organized in a graph, (ii) defines possible classes and relations of entities in a schema, (iii) allows for potentially interrelating arbitrary entities with each other and (iv) covers various topical domains.”</i>	Paulheim 2016
<i>“Knowledge graphs are large networks of entities, their semantic types, properties, and relationships between entities.”</i>	Journal of Web Semantics 2016 [Kroetsch and Weikum 2016]
<i>“Knowledge graphs could be envisaged as a network of all kinds of things which are relevant to a specific domain or to an organization. They are not limited to abstract concepts and relations but can also contain instances of things like documents and datasets.”</i>	Semantic Web Company [Blumauer 2014]
<i>“We define a Knowledge Graph as an RDF graph. An RDF graph consists of a set of RDF triples where each RDF triple (s, p, o) is an ordered set of the following RDF terms: a subject $s \in U \cup B$, a predicate $p \in U$, and an object $U \cup B \cup L$. An RDF term is either a URI $u \in U$, a blank node $b \in B$, or a literal $l \in L$.”</i>	Farber et al. 2016
<i>“[...] systems exist, [...], which use a variety of techniques to extract new knowledge, in the form of facts, from the web. These facts are interrelated, and hence, recently this extracted knowledge has been referred to as a knowledge graph.”</i>	Pujara et al. 2013

Since Ehrlinger and Wöß’s collation efforts (2016), other authors, such as Zeng et al (2021), have also defined a knowledge graph as an RDF graph, akin to Farber et al’s (2016). In the aptly titled volume *Knowledge Graphs*, Hogan et al. (2021) reference all of the above as different implementations of a knowledge graph (KG), and which they broadly define as:

a graph of data intended to accumulate and convey knowledge of the real world, whose nodes represent entities of interest and whose edges represent relations between these entities.

Furthermore, they add:

Property graphs can be converted to/from directed edge-labelled graphs. In summary, directed edge-labelled graphs [e.g. RDF] offer a more minimal model, while property graphs offer a more flexible one (Hogan et al 2021, § 2.1.3).

But there are challenges in these transformations and it is not always straightforward, requiring indirect transformations (Voegeli 2018).

3.4.2 Building a Knowledge Graph for a Specific Domain

The theoretical and general procedures to construct a representational domain knowledge base encoded in a graph structure remains largely similar to the precepts followed in the early research at the University of Twente. The first phase of building a knowledge graph can be summarised in the following three steps:

[A1]. Text analysis : Mapping a text on a graph.

[A2]. Construct analysis : Determining subgraphs that form “natural” units.

[A3]. Link integration : Using a “path algebra” to derive new knowledge from the extracted knowledge. This required a multiplication rule for consecutive links and a summation rule for parallel links” (Nurdiati and Hoede 2008, 2)

This method was derived from the *Knowledge Integration and Structuring System (KISS)* devised by Bakker (1987) for modelling expert systems and subsequently adopted by the wider University of Twente/University of Groningen knowledge graph research team in their later works (Stokman and de Vries 1988; Hoede 1994; Nurdiati and Hoede 2008). Gardner’s work (2015) on reading and reasoning with knowledge graphs continues from the scientific reasoning models of Stokman and de Vries et al. Such continuity in the use of graphs for building reasoning models proves promising for applying them for automatic inferencing in conservation.

More recent methods, such as Fathalla et al’s (2017) methodology to create a semantic knowledge graph (i.e. ontology) from survey data, continue to employ a similar sequence of steps:

[B1] Article selection

[B2.a] Formalization

[B2.b] Ontology development with manual extraction of instances

[B3] Querying the ontology

To compare and infer from both these methods, it can be seen that the first step to devising a knowledge graph is identification and analysis of data sources and the deconstruction and extraction of meaningful components from them. This can be achieved in various ways, for example, by using statistical analysis methods (Silge and Robinson 2017) or sentiment analysis (Liu 2015). Neill and Kuczera (2019) have reflected on text itself as a database:

[text is a structured carrier of information] *constrained by sequential presentation*
[but] *makes capable the modelling of thought in its multidimensional complexity.*

The preference for conservators to use expository text as a “database” in this sense provides insights to Velios’ (2016) findings on the over-use of free-text fields in conservation repositories. Kuczera (2016) has also introduced a further text analysis technique, where the full text is modelled as “a chain of nodes” (a linear graph) to counter the more reductionist approaches of extraction. The proclivity for using textual sources instead of broader data types, is grounded in the breadth of human-derived expert information stored in this form in both traditional documents and born-digital records. This does not preclude connecting other data types (e.g. numeric, image content, pixel content, etc.) to create a knowledge graph.

The next step, *concept analysis* (in KISS by Bakker 1987) or *formalization* (in Fathalla et al 2017) investigates the relationship between the extracted units (e.g. terms and values) and/or interconnected subgraphs (e.g. small-scale graphs within a larger knowledge graph, including triples in RDF graphs) derived from the data in the form of preliminary, source-specific graphs. Based on these relationships, the concept analysis stage seeks to combine several preliminary graphs.

The final stage of *link integration* (in KISS by Bakker 1987)) or *querying* (in Fathalla et al 2017) utilises graph theoretic analysis to “derive new knowledge”, that is, to identify patterns or indicative features that can be used to profile existing data or predict missing data. For example, Stokman and de Vries utilised knowledge graphs to map causality (1988, 192). In their work, they identified the major advantages for achieving this using a graph-theoretic approach include:

[the availability of a] *large number of graph-theoretical procedures and concepts*” [for analysis such as] *features of the overall structure of the graph, the*

relative centrality of arcs and vertices [and] detection of subgraphs with certain characteristics (ibid., 196).

Continuing on the trajectory from Bakkar, Zhang's thesis (2002) focused on modelling knowledge in terms of the semantics and syntactics of natural language to parse *concepts*, that is, meaningful chunks of perceived reality identifiable in thought and with corresponding *tokens* (representations) in language (Zhang 2002, 3, 20). Further modelling rests on grouping similar tokens together thereby constituting *types*.

The building of a knowledge graph is an iterative process evolving through a series of static representations over time. These procedures provide a preliminary, and rudimentary knowledge graph upon which further refinement techniques (such as those identified by Paulheim 2016) are necessary to enrich the graph.

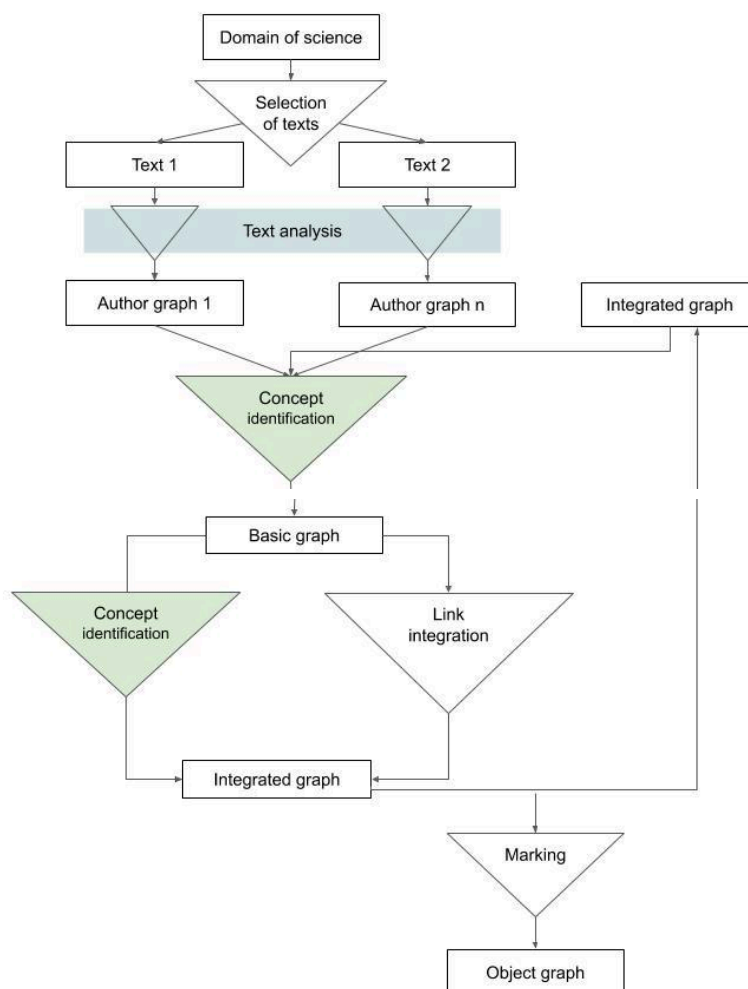


Figure 3.17 Stokman and de Vries' (1988, 196) workflow diagram for the Knowledge Integration and Structuring System (KISS) for knowledge graph construction.

3.5 Existing Knowledge Graphs Relevant to Conservation

At present, there are several existing conservation-related knowledge graphs (KGs) in the form of linked data RDF graphs derived by mapping data instances to the CIDOC CRM schema. However, despite this, limitations in SPARQL to execute graph traversals or recursive graph theoretic algorithms (Matsumoto et al 2018) has restricted any examination of the computational and analytical affordances of the graph structure. Nevertheless, it is pertinent to present an overview of these KGs here.

3.5.1 The CIDOC CRM

The Comité International pour la Documentation (CIDOC) Conceptual Reference Model (CRM), or CIDOC CRM, is an internationally recognized standard (ISO 21127:2014²⁴) for the interchange of cultural heritage information. As a formalised ontology, the CIDOC CRM is itself a domain-specific knowledge graph – a graph representation of broad cultural heritage concepts and relationships. The CIDOC CRM encoded in OWL is thus a description logic-based ontology for enabling automatic inference across the data mapped to it.

The CIDOC CRM is an event-centric conceptual model of historic cultural phenomena (Bruseker et al 2017, Doerr 2003, Doerr et al 2007). Its bottom-up development deriving from mostly museum catalogue records appear to have reinforced a modelling blindspot for 'future-looking' constructs such as planning. According to the *Definition of the CIDOC Conceptual Reference Model*²⁵, its "classes and properties (corresponding to predicates in a logical language) are usually considered to be universals (after Gangemi et al. 2002, pp. 166-181)."

Figures 3.18 and 3.19 demonstrate the event as the hub node and related data about that event is modelled as an immediate neighbour, thereby creating clusters for each event. Sanderson (2000) refers to this as the "conceptual model from 50,000 feet", that is, as viewed from an objective distance. 'E2 Temporal Entity' is the parent class for 'E5 Event' where '[E7] Activity' is a child class of E5 Event.

²⁴ <https://www.iso.org/standard/57832.html>

²⁵ Version 6.2.1, October 2015, p. xiii

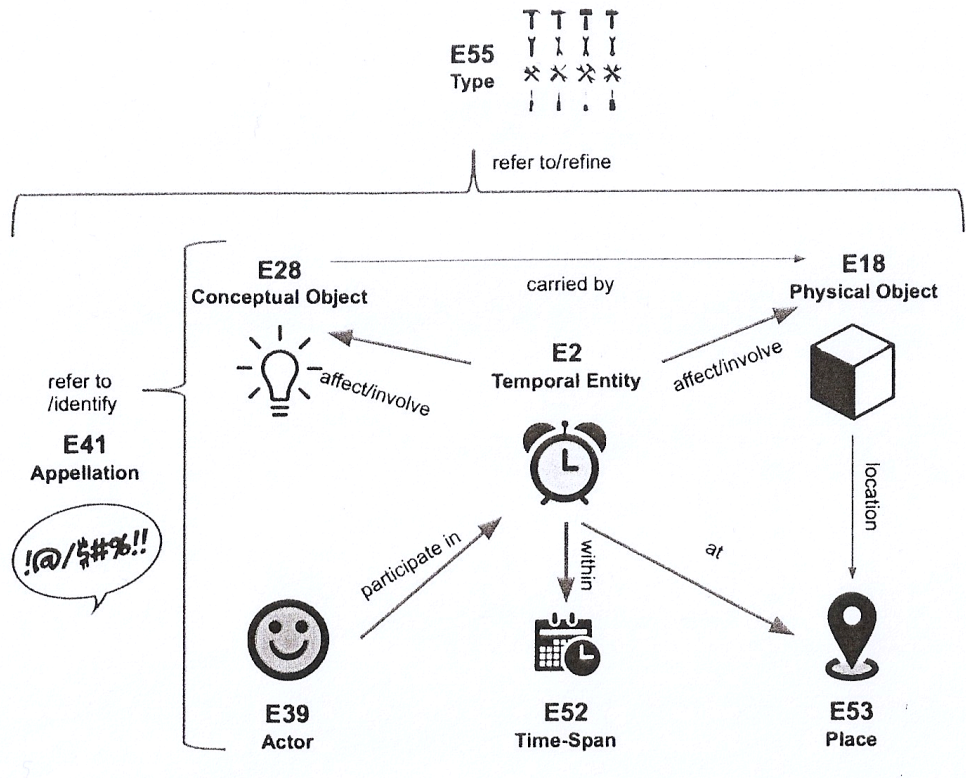


Figure 3.18 The CIDOC CRM top level categories. From Bruseker et al 2017, p.112, figure 1.

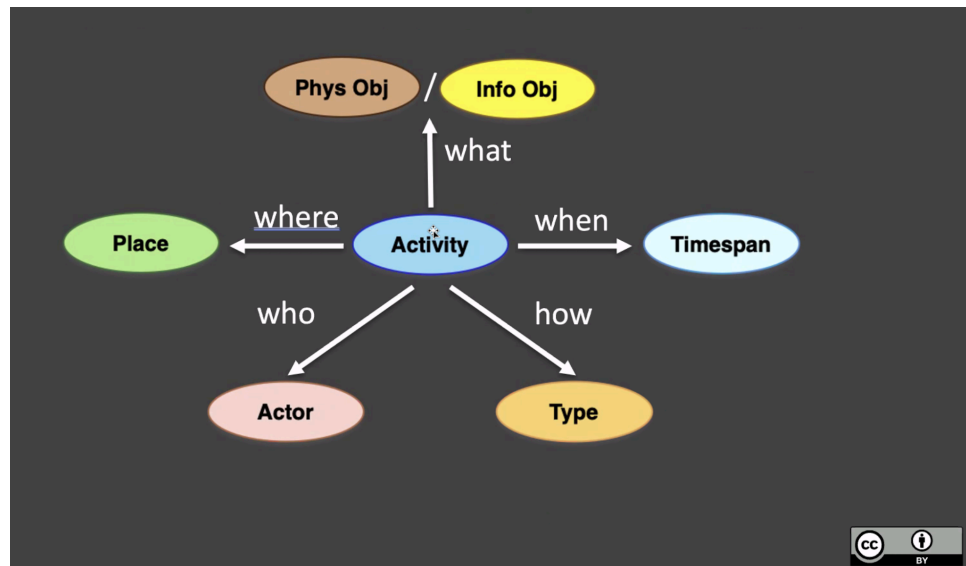


Figure 3.19 Conceptual modelling of event data using the CRM. From Sanderson, R. (2020)

According to Bruseker et al (2017):

[The CIDOC CRM] does not present a neutral view, but by making its commitments explicit, it neutralizes the ambiguity and overreach problems...The goal is not a perfect representation of knowledge, but one adequate to the aims of the domain users and consistent with reality. (ibid., , 105)

Aside from avoiding obvious errors of syntax and misunderstanding of terms, there is no "correct way" to map to CIDOC CRM or any standard...the data producer is best placed to produce the most representative translation of their data into the common expression. It is the domain specialist who has the knowledge of what their data means and what questions they want to be able to ask of it (ibid., 126).

The ontology is but one aspect of the representational system and framework for which the CIDOC CRM has been formally devised and maintained. The full system:

differentiates between [1] the ontology [i.e. CIDOC CRM], [2] the conceptualization that it is committed to [i.e. RDF], [3] the language [i.e. OWL] used for its implementation, and [4] the objective world that it refers to [i.e. cultural heritage data] (Bruseker et al 2017) [enumeration and clarification added].

While the CIDOC CRM can be viewed as a graph (as in Figure 3.20), conservators do not typically work in a graph-based computed medium despite such a medium being conducive for cognition and diagrammatic reasoning.

Classes tree

Use mouse wheel for dout

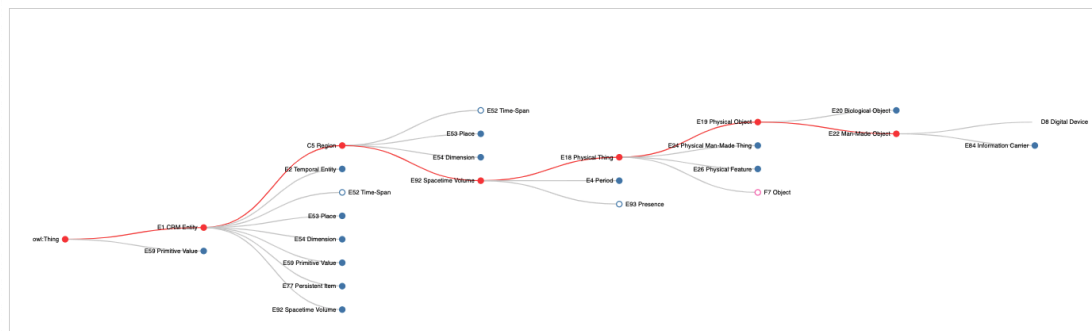


Figure 3.20 Graph representation of the classes tree for E22 Man-Made Object (CRM v. 7.1.1) via <https://ontome.net/classes-tree>.

Doerr (2009) anticipated a long phase of knowledge engineering necessary to overcome the gap between “a general underestimation of the complexity of cultural heritage conceptualization by IT experts” and the shortfall of articulated domain knowledge in objective computational terms. Thus, within the foundation of the CIDOC CRM is the anticipated role of the Knowledge Engineer, a bridging specialist in the domain knowledge and the computational tools and systems (Brachman and Levesque 2004).

As Velios and St. John (2022) have found, the adequacies of the CIDOC CRM to represent conservation-related information have proven challenging in the following areas:

- *negative information, e.g. that an object is not made of a material, which is often significant when considering provenance and material evidence;*
- *planned activities, such as proposed treatment;*
- *presence of multiple things, e.g. a book with many leaf markers, allowing a description of the types of markers, without referring to each individually;*
- *risk, such as the potential for flood damage from heavy rainfall (ibid.)*

Nevertheless, work on a conservation-specific extension (tentatively CRM_{cr}) is being undertaken by Moraitou et al (2022), Guillem et al 2017, Marinica (2019) and Bannour et al. (2018).

3.5.2 Cultural Heritage Thesauri

Examples of other conservation-related cultural heritage metadata utilising the RDF data structure include the Getty's (2017) *Art & Architecture Thesaurus* (AAT)²⁶, the EU's overarching humanities thesaurus, the *Backbone Thesaurus* (BBT) for use in top-level data integration (DARIAH EU et al 2019), and the Crisatel thesaurus, which was developed specifically for a multispectral imaging project launched in 2001 (Lahanier et al 2002; Cotte and Dupouy 2003; Ribés et al 2003; Berns et al 2005).

While the Crisatel thesaurus was developed specifically with terms for the conservation and restoration of paintings (Axaridou 2020) the other two are not specifically focused on conservation activities. The *Art & Architecture Thesaurus* (AAT) is continuously compiled from over 300 contributors with monthly updated releases. At the time of writing, the thesaurus consisted of 22,456,525 triples or over 3GB of data content. In contrast, the *Backbone Thesaurus* (BBT) contains very broad terms, such as "activities", "physical features", and "conceptual objects", and version 1.2.2 consists of 140KB of data content.

3.5.3 Linked Conservation Data

Most recently, the Linked Conservation Data (LCD) consortium project (St. John and Velios 2022; Lieu and Campagnolo 2022) investigated the application of the CIDOC CRM ontological model to a wider corpus of conservation documentation. One of the aims of the consortium is to "assess the suitability of the CRM and its extensions for conservation" (Linked Conservation Data)²⁷. It demonstrated how conservation data can be mapped to the CIDOC CRM by transforming existing conservation data into machine-readable RDF graphs. However, the computational benefits of these CIDOC CRM-mapped graphs have not been investigated beyond their interoperability via semantic web standards. Validation procedures have not been developed nor implemented, thus far. Furthermore, variations in modelling practices (Lieu and Campagnolo 2022) indicate a potential for uncertain downstream implications in terms of interoperability, retrieval, and automatic reasoning. Thus, there is significant scope for further investigation in terms of graph-based knowledge representation, analysis, and method standardisation for conservation.

²⁶ <http://vocab.getty.edu/>

²⁷ *Linked Conservation Data*. (n.d.). [Project website]. Linked Conservation Data. Retrieved June 5, 2023, from <https://www.ligatus.org.uk/lcd/>

3.6 Examples of Graph-Based Applications and Analysis

This section presents several cases of graph-based applications and methods to provide a glimpse of the broad areas of application afforded by graphs.

3.6.1 Graphs applied to reasoning, prediction, and discovery

Finlayson, LePendou and Shah (2014) achieved validated pattern recognition for prediction derived from electronic medical records and clinical narratives. Bean et al's (2007, 2) work in medical knowledge graphs found "representing facts as a graph allowed both highly efficient querying and automated reasoning." They were able to leverage the knowledge graph of existing electronic health records (EHRs) to infer and predict missing information related to adverse drug reactions with high confidence. They utilised a predictive algorithm to infer missing edges representing adverse drug reactions between drug nodes and adverse drug reaction (ADR) nodes in the cases where the ADR is a fact but missing from the source database and in cases where "the drug can cause ADR but is not yet known to [do so]" (Bean et al 2017, 2). Their final graph contained 70,382 edges for 524 drugs.

In the short history of conservation as a profession, it has already been shown how past treatments may have unintended consequences leading to potentially new risks. In the 1980s, the questioning by Koob (1982) regarding the instability of the commonly used and commercially available adhesive, cellulose nitrate, is a textbook example that resulted in a marked change in practice (Shashoua, Bradley and Daniels 1992; Selwitz 1988). Shallow searches to identify what materials and substances were used on objects remains a key part of conservation planning and activities. The ability to make more complicated searches and identify other risk patterns based on existing records increases the scope for how collections care can be planned and implemented.

3.6.2 Graph models for monitoring places and spaces

Buckley and Harary (1990, 5) and Foulds (1985) have described the use of graphs to depict spatial relationships in architecture such as accessible routes and centrality (e.g. a meeting hall). This has been applied to the design of traffic flow through rooms where a

path is required to always lead back to an exit, like in a museum, for example. A use for conservation can include the modelling of the museum environment for tracking environmental monitoring data, such as the movement of pests or the logging of temperature or humidity data. A spatial network analysis can be devised to model, track, alert, and inform on increasing risk factors and proximity to highly susceptible objects in the collection, for example. Another spatial modelling potentiality is in tracking travel of objects as part of loans and travelling exhibitions. What are the correlations between miles travelled and risk of damage? Do incident reports cluster around certain locations? Certain carriers? Or are there sequences (i.e. what combination of factors, and in what order?) which increase such risks?

3.6.3 Graphs for workflows and risk assessments

The Apgar score (American Academy of Pediatrics 2006) is a method for rapidly assessing newborn babies at 1 minute, 5 minutes, and 10 minutes after birth using 5 criteria (Appearance, Pulse, Grimace, Activity and Respiration), and a score of 0, 1 or 2 for each, at each time interval. The scores and their cumulative total provide an indication as to the level of care and immediacy of their needs.

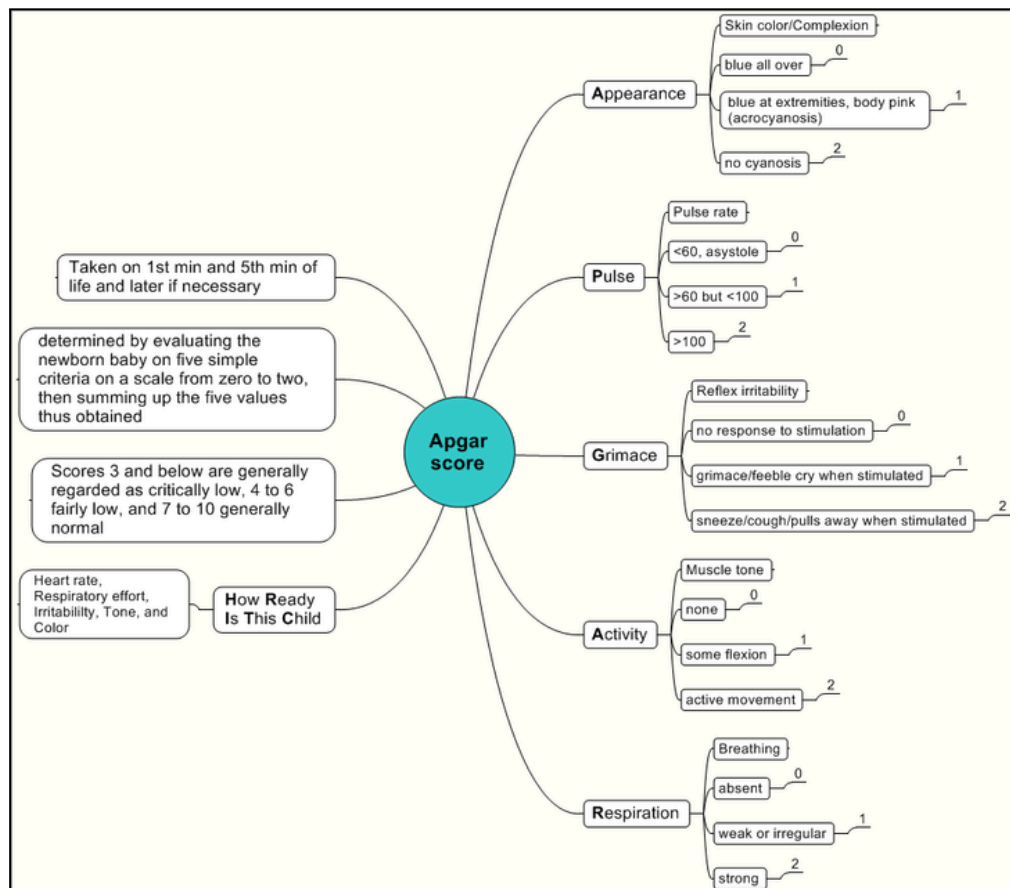


Figure 3.21 Apgar Score. (Madhero88, 2010) Wikimedia Commons.

The Apgar score is a widely employed heuristic that aids assessment and systematic recording. Research on Apgar score data have identified patterns that correlate with the health and development of the individual even in later life (ibid.). Figure 3.21 is a graph representation of the Apgar score method. The visualisation can serve as a reference for the scorer while the representation itself can inform computer models for storage or analysis. It is not a far leap to mirror such a system for managing conservation assessments, for example, by encoding condition categories (as above, see Table 2.1 in §2.4.1) as a graph in such a manner.

Beyond assessing risk, graph modelling and analysis has also been applied to assess data quality. Kesper et al (2020) have successfully applied graph theoretic approaches to identify patterns and anti-patterns in cultural heritage research data for the assessment of data quality. Their method, firstly, identifies a *generic graph pattern*, “defined as a first-order logic expression over graph structures”, which is then adapted to create an *abstract pattern* that is specific to a database technology (e.g. XML of TEI-encoded texts, in their case). Finally, a *concretised pattern* is formalised (consisting of a graph and a condition) derived from an understanding of the database format and with domain knowledge. These graph patterns and anti-patterns serve as metamodels for identifying local-level data quality.

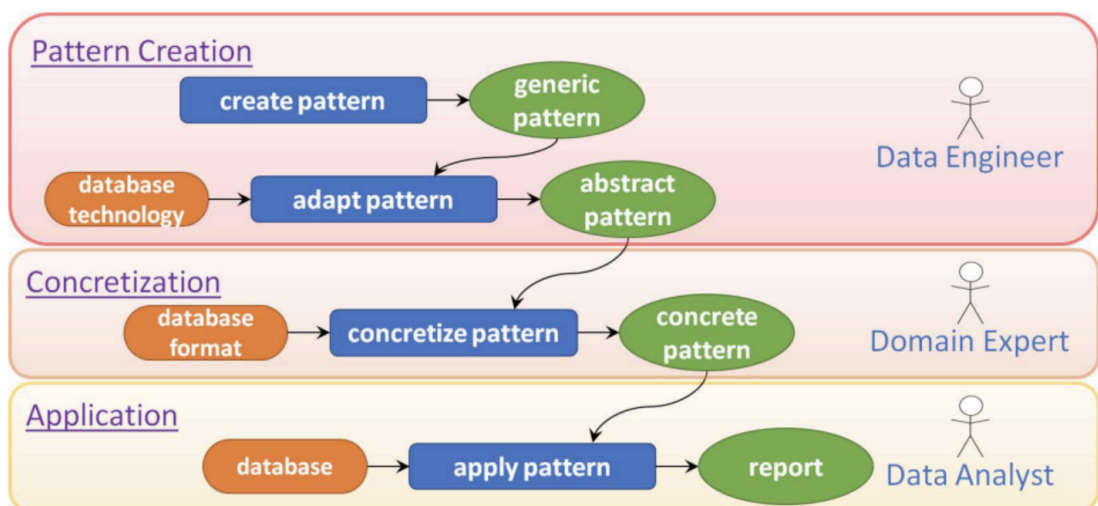


Figure 3.22. “Workflow of pattern creation and application” by Kesper et al (2020).

3.6.4 Graph models for flexibility at scale

The flexibility of graphs allows for modelling very large and very small things. On the large scale, for example, Dale (2017) demonstrates how graph theory can be applied to the study of ecosystems and used at varying scales in ecological sciences. On the smaller scale, for example, Pothaud et al (2000) used graph analysis to study the structure of bone porosity. The ability to model the very small and scale up to the very large, such as at population level, makes graphs very powerful and very useful as re-usable models can be incorporated as needed.

One method for investigating graph structures at varying scales is via motifs and related constructs known as graphlets (Espejo et al 2020) and orbits. Graphlets provide insights into the topology (shape) of larger network graphs by looking at local elements. For example, graphlets have been used to understand large bioinformatic networks through the analysis of smaller, local sub-graphs. Sarajlić et al (2016) leveraged graphlets to study directed (single-direction flow) networks of metabolic reactions in humans. This then enabled characterisation and comparison of similar metabolic reactions across other species and enabled prediction of enzymatic functions in other species.

Graphlets have also been used to investigate the dynamic nature of other real world networks. Yaveroğlu et al (2014) devised a graphlet correlation distance (GCD) measure to evaluate world trade networks over time and found a correlation between crude oil price changes with world trade network topology, particularly the rise of oil prices during periods of crisis in world trade networks. Positions of graphlets in relative distance to each other correlated with economic wealth and poverty. Wealthy “broker” nations were situated within clusters (where they can act as brokerage mediators between unconnected countries) and impoverished countries tended to be positioned in the peripheries of the network.

Dale presents another use of graphlet analysis as a means to algorithmically infer or “extract the rules of process from observed patterns”, that is, by “deducing function from form” (2017, 257). He cites the work of Minh et al (2013; 2015) where the rules of a game were deduced by an artificial system “observing” player choices and behaviours.

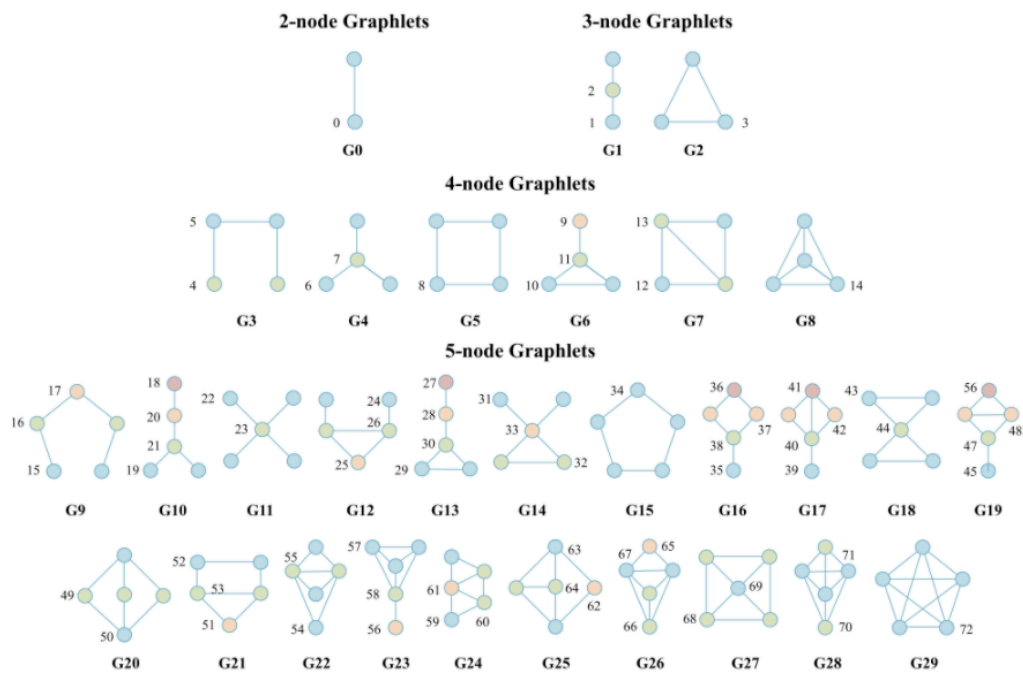


Figure 3.23. Graphlet Permutations. Image and caption from Espejo et al 2020, Figure 1. “2- to 5-node graphlets (from G_0 to G_{29}) and their automorphism orbits from (0 to 72). For each graphlet, nodes in the same automorphism orbits are identified with the same color (e.g. all blue nodes in G_1 are in O_1 , they are in a symmetric position in the graphlet, the green node is in a different topological position, it is in O_2).”

Potential applications of graphlet-based analysis in conservation can include characterising treatment efficacy patterns across objects or collections, predicting or suggesting timely interventions and for profiling larger documentation networks (see Chapter 6 for a demonstration of this latter application).

3.6.5 Graph-based research in cultural heritage and archival sciences

Outside of conservation but within adjacent fields, there has been a greater adoption of graph-based research methods in the digital humanities/digital history, cultural heritage, and archival sciences in recent years. Two principal strands of adoption of graphs for research have been the use of graph databases and the application of network analysis. While a graph database can support graph theoretic (network) analysis, not all projects utilising network analysis employ a graph database (see section 4.2.1 in the next chapter on Tools for examples of other pipelined approaches). *The Codex*²⁸ project on annotating and mapping medieval manuscripts (Kuczera 2017; Neill and Kuczera 2019) and the European Holocaust Research Infrastructure (EHRI) Project²⁹ (Bryant 2013, Blanke

²⁸ https://zfdg.de/sb004_008

²⁹ <https://www.ehri-project.eu/>

et al 2015) both utilised the Neo4j graph database platform. Graham et al (2022) have prepared a “Macroscopic” of big data tools for historians with two dedicated chapters on network analysis consisting of an introduction to several network analysis tools, including Gephi³⁰ and R³¹, and guidance for practical implementation.

The field of archaeology has utilised a specific sub-domain of network analysis known as social network analysis (SNA) (Wasserman and Faust 1994; Carrington, Scott, and Wasserman 2005) to investigate “patterns and processes of interactions in past societies” (Knappett 2013, 3) with an emphasis on investigating social structures and social theories, such as matching social processes to existing theories, informing the development of new theories, and for bridging method and theory in archaeology (Brughmans 2013). Criticisms of SNA (Knappett 2013; Brughmans 2013) have focused on the limited techniques and tools being adopted from the wider network science domain into archaeology and that research had been predominantly occupied with network reconstruction, also known as network synthesis (Knappett 2013b, 8). The three broad categories for such network synthesis and network interaction studies were spatial (geography), temporal (historic), and material (material culture) (ibid., after Mills 2017). However, more recent developments in the use of network analysis in archaeology (Brughmans and Peeples 2023) expand from these limits to focus on exploratory networks, uncertainty in network analysis, visualisation and comparing networks using R. Sindbaek (2013) and Gjesfjeld and Phillips (2013) have also applied network analysis towards data validation against generalised models “hence providing a level of validation to results obtained by other means” (Knappett 2013b, 9).

Other example projects that have featured network analysis include exploratory work into investigating methodologies at The National Archives (UK) to study the UK Government Web Archive (UKGWA) (Storrar and Talboom 2019)³², a study of European courtly political networks and intrigue through letter correspondences from the Tudor period (Ryan and Ahnert 2021), and a study of historic sources and modern social media archives to evaluate art historic research (Noble et al 2022). In this last example, the researchers examined digitised stock books in the archive of the art dealer Thos. Agnew & Sons and the social media archive derived from Tweets related to The National Gallery as two contrasting sources for network analysis and argued that it need not be reductive in method as the network model can be devised to represent conceptual discursivity and

³⁰ <https://gephi.org/>

³¹ <https://www.r-project.org/>

³² <https://blog.nationalarchives.gov.uk/network-analysis-of-the-uk-government-web-archive/>

not just object-centric models that replicate physical things and their attributes, thereby allowing for “new types of artistic discourse, and even criticism” (ibid.).

3.6.6 Graph-based measures for domain analysis

Domain	Max. # of Vertices	Max. # of Edges	Avg. # of Vertices	Avg. # of Edges
Cross Domain	614,448,283	2,656,226,986	57,827,358	218,930,066
Geography	47,541,174	340,880,391	9,763,721	61,049,429
Government	131,634,287	1,489,689,235	7,491,531	71,263,878
Life Sciences	356,837,444	722,889,087	25,550,646	85,262,882
Linguistics	120,683,397	291,314,466	1,260,455	3,347,268
Media	48,318,259	161,749,815	9,504,622	31,100,859
Publications	218,757,266	720,668,819	9,036,204	28,017,502
Social Networking	331,647	1,600,499	237,003	1,062,986
User Generated	2,961,628	4,932,352	967,798	1,992,069

Figure 3.24 Basic descriptive statistics of all analyzed datasets from Zloch et al 2021. Source: <https://data.gesis.org/lodcc/2017-08/>

Zloch et al (2021)³³ analysed 280 RDF datasets from nine knowledge domains³⁴ captured in ca. 2017 from the LOD Cloud (as mentioned above in section 3.3 and Figure 3.13). Using 56 measures, they found four measures worked well for differentiating these domains. These are:

- Mean_predicate_list_degree
- Pseudo_diameter
- Max_labelled_out_degree
- Mean_out_degree

While there is some incidental conservation data in the LOD Cloud graph (as shown in Figure 3.14 above), there is not currently a large enough subgraph of conservation-specific triples in the LOD Cloud to be able to extract, deduce or infer deeper characterisations of the conservation domain. The available conservation RDF data in the LOD Cloud is too sparse and shallow. Nevertheless, the work of Zloch et al (2021) are indicative of the possibilities with graph analysis for the characterisation of domain- and sub-domain-specific phenomena.

³³ <https://data.gesis.org/lodcc/2017-08/>

³⁴ The nine domains include: a cross domain category, geography, government, life sciences, linguistics, media, publications, social networking, and user generated.

3.7 Summary

The statistician George Box famously said:

"... all models are approximations. Essentially, all models are wrong, but some are useful." (Box and Draper 1987, p. 424) [emphasis added].

Graphs have proven to be very useful in many other domains of research. As traditional, tabular-only, strictly relational models are seen to be unfit for purpose in a conservation context, graph models offer a promising alternative. The practice of structuring domain knowledge as a graph has a developmental history spanning over 40 years and continues to be relevant and heavily utilised today. However, the application of graphs in conservation remains under-developed and under-utilised. Therefore, there is a strong case to support further research into the application of graph-based modelling and graph theoretic analysis of conservation documentation.

Graphs are now recognised as a standard data model in semantic technologies for implementing and achieving data management and interchange which speaks directly to the concerns raised by the FAIC report (Zorich 2016). Powell and Hopkins (2015) have shown its wider uses in information management from an information and library sciences perspective and it has also been shown that graphs support FAIR compliance. Both cognitive and computational models can exploit the same graph structures enabling advantages and efficiencies when crafting human-readable and machine-readable systems. Figure 3.24 proposes a revision of Zeng's (2008) diagram to better reflect this wider use and involvement of graphs.

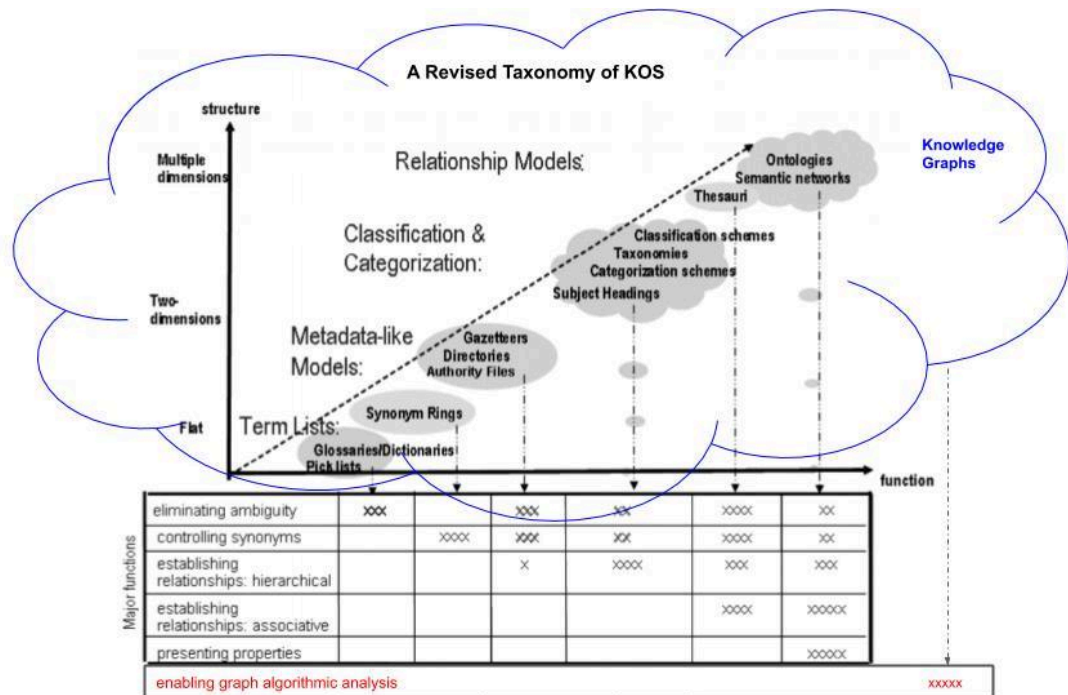


Figure 3.25 A Revised Taxonomy of Knowledge Organisation Systems. Based on "An overview of the structures and functions of KOS [knowledge organization systems] from Zeng 2008 (p. 161, Figure 1). Modifications in blue and red.

A graph-based approach would support conservation documentation and not hinder its purpose and use in decision-making. Knowledge representation utilising a graph model is flexible and can support highly-connected relationships between conservation activity, documentation activity and information management activities. This, in turn, supports automatic inferencing. It also supports investigations into the nature of complexity in conservation by providing a language and a means of discerning features that help describe the components and contributors to complexity.

The affordances of a graph-aided approach to conservation cannot be overstated. As the contents of this chapter have demonstrated, there is a strong case for the adoption of graphs into conservation theory and practice, not only for their broad usefulness but as a point of necessity at this moment of the computational turn in conservation. Given the myriad graph-based solutions already well-known and well-documented, and if we are to "cope with complexity", we must document in a manner that is conducive to complexity as it is defined: "a group or system of different things that are linked in a close or complicated way; a network" (Pocket OED 2023).

For conservation to begin to leverage such a powerful method, it is necessary to begin by understanding what are the existing structures in our data? Before we begin searching for hidden patterns, or optimising current patterns, what are the obvious or

typical patterns for conservation? How do we build conservation knowledge graphs that are not only a representational repository but also can be analysed directly using graph theoretic means in order to maximise our data management efforts?

Thus, the next chapters will address the first steps towards a more graph-aware approach to conservation, the aims of which are:

- to demonstrate an encoding-based documentary practice for conservation
- to articulate a method for conservation knowledge graph construction and validation
- to demonstrate how conservation documentary practice can be extended through a computational thinking framework, and
- to demonstrate modelling and simulation in conservation in the form of graphs, which in turn can support further problem-solving and articulation of complexity.

4.0 A Graph Representation Method for Conservation

4.1 Overview

The previous chapters have laid out the argument in support of adopting graphs into the field of conservation, particularly for documentation and data analysis. This chapter aims to demonstrate, at the most foundational level, the principal method for the modelling of conservation data using graph structures, which in turn allows these models, in both their content and structuring, to be analysed using graph theoretic means.

Allemang and Hendler (2011, 11) have framed modelling as “[a] craft” that makes “sensible, usable, and durable information resources from this medium [i.e. Semantic Web technologies]”. The modelling method presented here is a prototype for how to transform conservation data and encode conservation practice into the structure of a graph by using the property graph (PG) model. This can be done without the need for mapping first to the CIDOC CRM. However, it can still facilitate subsequent mapping to the CIDOC CRM and thus can be utilised as a preceding metamodel.

The remainder of this chapter is as follows: Section 4.2 provides an overview of the tools used and the rationale for choosing the Neo4j graph database platform for this study. Section 4.3 describes the modelling principles for preparing conservation graphs, drawing from foundational principles in mathematics and cognitive representation. Section 4.4 will describe a set of graph theoretic features and algorithms to profile and analyse graph models. This is not an exhaustive list of graph theoretic features and algorithms but a compiled list of baseline characteristics and tools shown to aid understanding and to inform further exploration of networked data. Section 4.5 will describe the role of querying and query design for content analysis and path-based inference. Section 4.6 defines and elaborates on the roles of verification, validation and calibration (VVC) in computational model development. Finally, section 4.7 provides a summary of the method for conservation data integration, management and analysis, where foundational graph concepts are placed at the fore.

4.2 Tools

Table 4.2.1 List of Tools

Tool	Version(s)	Source/Reference
Neo4j Cypher APOC GDSL Neosemantics	4.2.0 to 4.4.7	https://neo4j.com/ Francis et al (2018)
Python	3.7.4	https://www.python.org
beautifulSoup	4.12.0	https://www.crummy.com/software/BeautifulSoup/bs4/doc/
spaCy	2.2.4	https://spacy.io/ Srinivasa-Desikan (2018)

Graph-based modelling, analysis, and data integration was undertaken using the Neo4j graph database platform and its Cypher query language. Python scripting was used to extract data from public resources for inclusion in the graph models. BeautifulSoup is a Python library specifically for extracting data from HTML. The Python natural language processing library spaCy was used to parse and annotate text for inclusion in the graphs.

4.2.1 The Limitations of RDF and the Rationale for Using Neo4j/LPG

Graph databases fall within the NoSQL category of databases. There are many different types of NoSQL databases (others include document stores, triple stores for RDF, key-value stores, etc.) and there are many mature graph database platforms available (e.g. Apache TinkerPop/Gremlin, TigerGraph, etc.). The graph database platform used in this research is *Neo4j* which is based on a labelled property graph (LPG) model.

As mentioned in section 3.5 *Existing Knowledge Graphs Relevant to Conservation*, conservation data has been captured using the RDF graph or triple structure through mappings to the CIDOC CRM RDFS serialisation to facilitate the public accessibility of Linked Open Data. However, limitations with the RDF graph model have been expounded (Hayes and Gutierrez 2004; Birkholz and Meroño Peñuela 2019a; Reutter et al 2015; Libkin, Martens and Vrgoč 2016). This, coupled with anecdotal evidence from

conservation mapping workshops (see Chapter 6.0) regarding the challenges in adoption, implementation and deployment, demonstrate a demand for further study and trial of potential alternatives, such as the labelled property graph (LPG) structure. Nevertheless, key criteria for research using an alternative graph structure must also consider:

- What can be used that can be easily replicated by other heritage professionals? especially conservators?
- What supports a flexible, graph theoretic approach?
- What can, ideally, support both RDF and LPG models for analysis and comparisons between existing and case study-derived graph models?

Fiorelli and Stellato (2021) have surveyed the various methods of converting tabular data to RDF. However, challenges and pitfalls in using a RDF structure include encountering “multiple occurrences of the same resource in the data structure [which] leads to undesirable redundancies and...[obfuscates] the connectivity of resources” (Hayes and Gutierrez 2004, §1). Furthermore, in regards to graph analysis of RDF data, Birkholz and Meroño Peñuela (2019a) observed:

analysis of networks from RDF is largely done with a pipeline of tools (e.g. Groth & Gil 2011; Gil & Groth 2011) In these workflow approaches, researchers specify SPARQL queries, extract networks and export data as matrices, and implement network analysis tools to investigate graphs.

In their own work (2019b), Birkholz and Meroño Peñuela’s pipeline is Python-based and utilises the `rdflib`¹ library to capture the RDF graphs or subgraphs and then analysed using the `networkx`² library. Sanchez-Alonso et al (2020) used an alternative pipeline where they accessed SPARQL endpoints via the CKAN³ open source data management system using the Python SPARQLWrapper⁴. Willighagen (2014) employed a further alternative pipeline that utilised R⁵ (a data analysis environment run on the R programming language) to implement graph analysis of biological RDF data. In all three examples, any further data analysis requirements, such as visualisations of the results,

¹ <https://rdflib.readthedocs.io/en/stable/>

² <https://networkx.org/documentation/stable/index.html>

³ <https://ckan.org/>

⁴ <https://pypi.org/project/SPARQLWrapper/>

⁵ <https://cran.r-project.org/>

would have needed additional pipeline tools, such as the Python matplotlib⁶ library or the Javascript D3.js⁷ library for web-based visualisations.

Programming in networkx requires a mathematical grounding in graph theory that exceeded this researcher's knowledge and experience, particularly, at the start of this research project. Preparing a pipeline-specific environment and a wholly coding-based approach without any support of a graphical user interface for non-specialists makes such an approach, at present, prohibitive to many in the general conservation community. In contrast, Neo4j presented an advantage as graph building, querying, analysis, and visualisation is available within the single platform, all using the Cypher querying language.

Furthermore, although RDF is a graph-based model, performing analysis directly in RDF is hampered by the limitations of its query language, SPARQL (Reutter et al 2015). Libkin, Martens and Vrgočl 2016 have identified these and additional limitations, such as the lack of many graph constructions and the lack of "a more general transitive closure operator". Therefore, an alternative approach that nevertheless remains graph-based and can support more advanced graph theoretic analysis would deserve further study.

Coincidentally, Blazegraph, the RDF triple store that underpins WikiData and ResearchSpace, was purchased by Amazon ca. 2018 (Anadiotis 2018) placing a great deal of uncertainty around its open source status and raised questions in regard to the ongoing maintenance and support of the product for the user community. Thus, this further underscored a need to identify alternative graph-based platforms for consideration.

As mentioned in the last chapter (section 3.6 *Examples of Graph-Based Applications and Analyses*), there has been an increase in the use of network analysis in the digital humanities of which the Neo4j graph database platform has repeatedly been identified as the principal project platform. In the years immediately preceding the start of this research, other cultural heritage projects working with graph databases, most notably *The Codex*⁸ project on annotating and mapping medieval manuscripts (Kuczera 2017 and Neill and Kuczera 2019⁹) and the European Holocaust Research Infrastructure (EHRI)

⁶ <https://matplotlib.org/>

⁷ <https://d3js.org/>

⁸ https://zfdg.de/sb004_008

⁹ Neill, I., & Kuczera, A. (2019). *The Codex – an Atlas of Relations*.
http://dx.doi.org/10.17175/sb004_008

Project¹⁰ (Bryant 2013, Blanke et al 2015), were beginning to publish and promote their research, which included their use of the Neo4j graph database platform in each case. In 2019, the Neo4j developer community began supporting graph-based analytical algorithms as a plugin (Needham and Hodler 2019), subsequently renamed the *Graph Data Science Library*¹¹. This plugin capability provides direct, in-platform access to run algorithmic analyses as opposed to relying on outside-of-platform options that would require additional mathematics-specific programming in languages such as Python or *R*.

Finally, again coinciding with the start of this research, Neo4j was developing an RDF-compatible plugin, subsequently called *Neosemantics* (Barrasa 2016, Barrasa 2018), for importing and exporting RDF data. Both of the Neo4j *Graph Data Science Library* and *Neosemantics* plugins are open source and are presently fully integrated with the standard community and enterprise versions of the platform, enabling building, querying, and analysing across property graph and RDF-graph models directly using Neo4j's Cypher query language. The declarative Cypher query language itself replicates a graph pattern in the syntax of the query and the formal specification is described by Francis et al 2018 (for more on Cypher, see section 4.3.2 *Sets, Tuples and Subgraphs*).

For these reasons, the Neo4j platform was chosen for the purposes of this research. Nevertheless, the outcomes of this research, the procedures and analytical measures applied are achievable using other tools and pipelines as the overall methods are transferable (albeit any attempts to duplicate results must account for the specific parameters and conditions detailed here and found in the companion GitHub repository for this thesis¹²). The graph models described in this research can, in principle, be recreated using pipeline tools specifying *directed graph* (or *digraph*) and *multigraph* representations.

4.2.2 Notes to the reader regarding Neo4j configuration and Cypher syntax

Please note the specific Neo4j configurations in *Appendix A*, particularly, those referring to increased memory settings to ensure that Graph Data Science Library algorithms are able to run to completion. Further details on Cypher syntax can be found in the Cypher Manual¹³.

¹⁰ <https://www.ehri-project.eu/>

¹¹ <https://neo4j.com/docs/graph-data-science/current/>

¹² <https://github.com/ana-tam/conservation-graphs>

¹³ <https://neo4j.com/docs/cypher-manual/current/syntax/>

This thesis uses the Cypher syntax convention when referring to nodes, node properties, and relationships in the Neo4j labelled property graph models. Nodes are signified using enclosed parentheses, relationships are denoted using square brackets and node properties are denoted using a preceding variable followed by a period, for example, "n." or "a.". Node labels and relationship types are preceded by a colon, ":". For example, reference to a node with the label "Treatment Event" will appear as:

```
(:TreatmentEvent)
```

A node property key for "name" will be conveyed as:

```
n.name
```

where the 'n' preceding is a variable. Node properties are in key-value pairs and can be denoted in a query with curly brackets, "{}". For example, a (:Person) node, might have the node property "n.name" with the value "John Smith":

```
MATCH (n:Person{name: "John Smith"}) RETURN n
```

or

```
MATCH (n:Person) WHERE n.name = "John Smith" RETURN n
```

For relationships, for example, the following Cypher statement seeks to MATCH the path pattern for where any nodes (a) are connected to node (b) which has the node label "Material" by any relationship type, using the variable r. The direction of the edge is not specified in the first query so both incoming and outgoing results will be matched.

```
MATCH (a) - [ r ] - (b:Material)
```

A directed version of the same query uses the 'greater than' or 'less than' symbols to create an arrow to indicate the direction:

```
Match (a) - [ r ] -> (b:Material)
```

A full record of node labels, node properties, and relationship types can be found in the *Appendices* and GitHub repository.

Table 4.2.2. Examples of Cypher Syntax

Graph Component	Example Using Cypher Syntax where n, r and b are variables
Nodes, identified by their labels	(n:TreatmentEvent)
Relationships, identified by their types	[r:INVOLVED_USE_OF]
Node property	n.property (n:TreatmentEvent{reference:"abc/1/2"})
Path syntax	(n)-[r:ASSESSED]→(b)

As demonstrated in these examples, the Cypher query language is not only machine-readable but also easily human-readable. The benefits of familiarisation with Cypher is its ubiquity in the graph database sphere. An open source implementation is available (OpenCypher¹⁴) and it has been a significant influence on the development of the GQL¹⁵ specification and new ISO standard¹⁶ (see Figure 4.1). As of May 2024, the OpenCypher project has counted 15 implementations¹⁷ of Cypher across research and various commercial products. Hence, familiarity with Cypher aids familiarity with many other alternative databases and improves adoption and/or experimentation by those working in Neo4j.

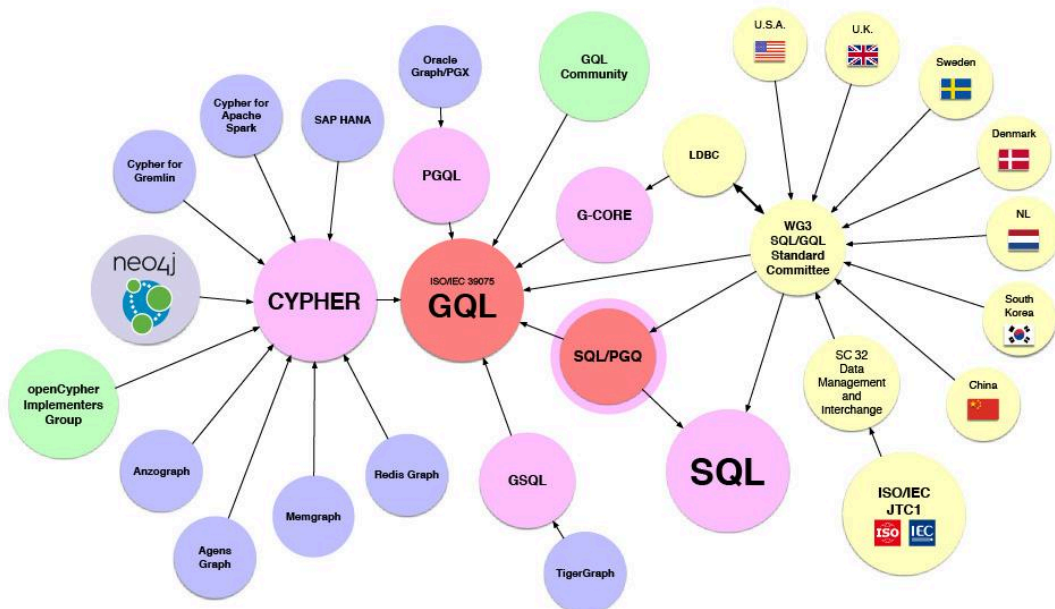


Figure 4.1 Contributions and influences on the GQL specification (Neo4j 2019; Green 2019)

¹⁴ <https://opencypher.org/>

¹⁵ <https://www.gqlstandards.org/>

¹⁶ <https://www.iso.org/standard/76120.html>

¹⁷ <https://opencypher.org/projects/>

4.3 Modelling Principles for a Graph Representation Method for Conservation

4.3.1 The Representational Basis for the Data Model(s)

Encoding is itself a form of documentary practice (Scifleet et al 2009), therefore, encoding is an extension of the conservation profession's imperative for documentation. The graph-based encoding method utilises a *declarative representation* for encoding knowledge in a computer-readable data model:

The key property of a declarative representation is the separation of knowledge and reasoning. The representation has its own clear semantics, separate from the algorithms that one can apply to it. (Koller and Friedman 2009, 1).

For a property graph-based representation, the structure of the encoding will also be a directed, multigraph representation.

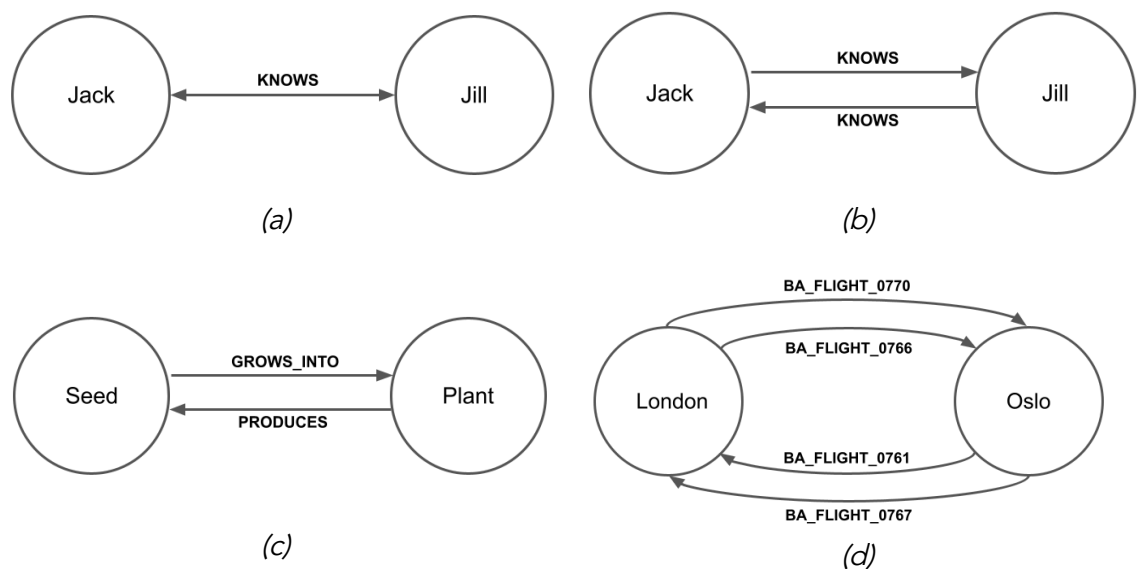


Figure 4.2. Directed graphs and multigraphs. (a) shows an undirected edge representing mutuality. (b) shows the same information but uses a directed representation (c) shows reciprocal directed edges where the edge types are different. (d) shows a directed multigraph representation where a pair of nodes can have multiple directed relationships with each other.

When encoding from text-based documentation, an awareness of the differences between natural language semantics and axiomatic semantics is necessary. Mathematics is grounded in rules, called axioms, and these rules are consistent and precise. Mathematics is intrinsically linked to representation. Variables, formulae, notations – are all ways of representing reality and concepts about reality in consistent and shorthanded ways. Attention must be drawn to the distinctions between things that are represented in natural language and how they are represented with mathematical precision. For example, when we say:

"A banana is a fruit."

what we mean is that:

"A banana is a type of fruit."

The first is an equivalence statement: $\text{banana} = \text{fruit}$, which is incorrect as it is factually and semantically inaccurate: "fruit" is not always "banana". The latter is a statement about sets: a "banana belongs to the set of all fruit", or "banana is a member or element of the set known as 'fruit'". In mathematical notation, this is written as:

$\text{banana} \in \text{fruit}$.

This distinction between what "is" and what "is an element of the set" has strong bearing on the modelling of reality. The terms adopted in this thesis for sets include referring to them as groups, types, and categories while elements of sets can also be referred to as instances or particulars. When modelling events and elements from reality, this work strives to refrain from SAME_AS relationships unless the two representations being compared are conceptually identical. For example:

("fire") - [SAME_AS] → ("la feu")

Both "fire" and "la feu" have the same semantic meaning with one being in English and the other in French.

However, in the example:

("Smith Family") - [SAME_AS] → ("Tom", "Stacy", "Dan")

equating “Smith Family” with a list of members of this family, “Tom, Stacy, and Dan” can lead to reasoning problems further down the line. It is always better practice to model individuals, individually. However, there are often situations when transforming or aggregating data that some of that data is not fully or consistently decomposed and will require pre- or post-process data cleaning. While it is humanly understandable to extract from this representation that the Smith Family consists of persons named Tom, Stacy and Dan, such a deduction uses additional premises drawn from one’s own understanding of context, experiences and memory, such as “families consist of individual members”, “individuals have names”, and “‘Tom’ is a name, etc.”, hence if there is a relationship between this list of names and a specific family (“Smith family”), it is likely that these are the names of members of that family. However, from a machine-readability standpoint, this representation only says a list with three values is explicitly the same as “Smith Family” which, semantically, is not what is intended. Therefore, another relationship label would be more appropriate and precise, such as [HAS_MEMBERS] or [HAS_ELEMENTS]. In practice, “SAME_AS” relationships should be used sparingly. The next section further clarifies conceptualisations of sets versus elements and how these are computed as RDF and property graphs.

4.3.2 Sets, Tuples, and Subgraphs

This section provides the conservation and heritage specialist reader with a brief overview of three foundational concepts in the modelling of graph-based information systems: sets, tuples, and subgraphs. Firstly, a key concept that underlies graphs (and, in fact, all of mathematics) is that of *sets* (Cunningham 2016). A set¹⁸ is a collection of *elements*, which can be finite or infinite in number. For example, the ‘set of all sonnets written by Shakespeare’ is a finite set while the ‘set of all real numbers’ is an infinite set. As mentioned earlier, a graph is itself defined in these terms: as a set of nodes, V , and a set of edges, E , that constitute the graph, G :

$$G = \{V, E\}$$

This simple definition of a graph stands true for both an RDF graph and a property graph. However, the RDF graph utilises the minimalist structure of the triple (subject-predicate-object) where each part of the triple carries an element and the

¹⁸ The curly brackets { } are the standard mathematical notation for sets.

predicate is the edge relationship. Hence, the number of edges in the RDF graph will always be the same as the number of triples.

On the other hand, a property graph (PG) structure is not limited to a triple structure and can carry more elements per node and edge. While a property graph consists of a set of nodes and a set of edges, these nodes and edges can each also carry sets of properties, which are themselves sets of names (or keys) and values (Angles et al 2019, §3.2). A labelled property graph (LPG), such as the graph structure used in the Neo4j graph database, also allows for the identification of subsets of nodes and subsets of edges through the use of labels. Thus, these two types of graphs –RDF vs. property graph– differ in how they carry their respective sets.

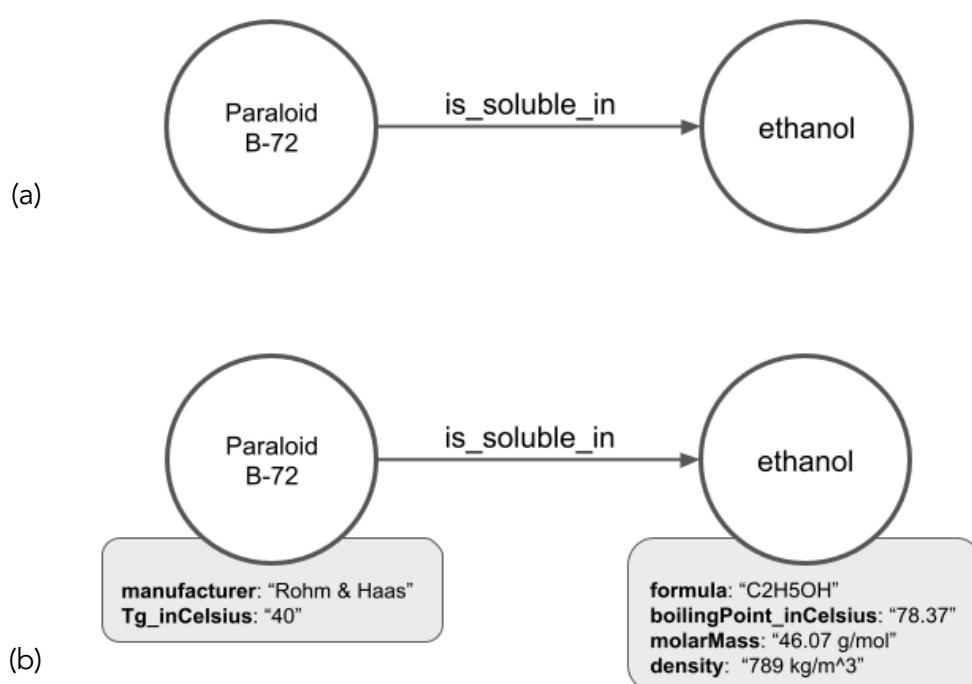


Figure 4.3 Comparisons between RDF and LPG structures. (a) Depicts a typical RDF triple structure while (b) depicts a LPG structure where each node can hold additional data content as node properties. To represent the same additional content from (b) in a RDF triple structure would utilise a triple statement for each property's key-value pair, and again another triple statement to connect to the principal node as the subject to the property key as the object. Further examples of transformations between RDF to LPG will be explored in Section 4.3.4 and Section 7.4.

What is also missing from the simple definition of a graph, as presented above, is an explanation or rule for how to map the two sets together, that is, which node has which edge? In mathematical terms, this mapping is conveyed with a *function* which encapsulates how you assign or transform elements from one set to another. Therefore, a better way of defining and contrasting these two types of graphs is in terms of tuples.

A *tuple* is a finite ordered list of *elements or sets*, that is, a sequence of elements or sets where the order is important. For example, (a, b, c, d) is not the same as (a, c, b, d) . Tuples are particularly important conceptualisations and representations used in databases. For example, a row of data in a spreadsheet or relational database table can be handled as a tuple. A tuple-based definition of a graph is like a very concise instruction manual. First it lists all the parts and next it tells you how to put the parts together. Very simply, with sets and tuples, you have ‘things’ and you have ‘how to relate those things to each other’.

Hartig (2014) defines the RDF tuple as:

$$(s, p, o) \in (I \cup B) \times I \times (I \cup B \cup L)$$

where (s, p, o) is the ‘subject, predicate, object’, I represents IRIs¹⁹, B represents blank nodes, and L represents literals. Plainly, the notation above can be interpreted as saying the s (subject) can be an IRI or a blank node, the p (predicate) is always an IRI, and the o (object) can be an IRI, blank node, or a literal. In practice, literals are strings that cannot be decomposed any further as a triple, and thus tend to be objects and terminal nodes (i.e. leaf nodes).

The property graph tuple (Angles 2018), instead, consists of two finite sets and a series of three further functions that handle the mappings between nodes and edges, labels, and properties.

Table 4.3.1 Angles’ (2018) definition of a property graph:

Assume that L is an infinite set of labels (for nodes and edges), P is an infinite set of property names, V is an infinite set of atomic values, and T is a finite set of datatypes (e.g., integer). Given a set X , we assume that $SET^+(X)$ is the set of all finite subsets of X , excluding the empty set. Given a value $v \in V$, the function $type(v)$ returns the data type of v . The values in V will be distinguished as quoted strings.

Definition: A property graph is a tuple $G = (N, E, \rho, \lambda, \sigma)$ where:

- (a1) N is a finite set of nodes (also called vertices);*
- (a2) E is a finite set of edges such that E has no elements in common with N ;*
- (a3) $\rho : E \rightarrow (N \times N)$ is a total function that associates each edge in E with a pair of nodes in N (i.e. ρ is the usual incidence function in graph theory);*
- (a4) $\lambda : (N \cup E) \rightarrow SET^+(L)$ is a partial function that associates a node/edge with a*

¹⁹ IRI stands for ‘Internationalised Resource Identifier’, a persistent web address encoding for a data element.

set of labels from L (i.e., λ is a labeling function for nodes and edges);
 (a5) $\sigma: (N \cup E) \times P \rightarrow SET^*(V)$ is a partial function that associates nodes/edges with properties, and for each property it assigns a set of values from V .

Angles' definition (Table 4.1) is akin to Francis et al's (2018) definition (Table 4.2) of a labelled property graph (LPG) for the formal specification of the Cypher query language and is the most accurate definition of the Neo4j graph model.

Table 4.3.2 Francis et al's (2018) definition of a labelled property graph:

Let L and T be countable sets of node labels and relationship types, respectively.

Definition: A property graph is a tuple $G = (N, R, src, tgt, t, \lambda, r)$ where:

- (f1) N is a finite subset of N , whose elements are referred to as the nodes of G .*
- (f2) R is a finite subset of R , whose elements are referred to as the relationships of G*
- (f3) $src: R \rightarrow N$ is a function that maps each relationship to its source node.*
- (f4) $tgt: R \rightarrow N$ is a function that maps each relationship to its target node.*
- (f5) $t: (N \cup R) \times K \rightarrow V$ is a finite partial function that maps a (node or relationship) identifier and a property key to a value.*
- (f6) $\lambda: N \rightarrow 2^L$ is a function that maps each node id to a finite (possibly empty) set of labels*
- (f7) $r: R \rightarrow T$ is a function that maps each relationship identifier to a relationship type.*

Francis et al's definition of the Neo4j labelled property graph model bears seven elements or sets in its tuple while Angle's generic property graph definition has only five elements or sets. Nevertheless, they are functionally similar as the extra tuple elements (f3) *src* and (f4) *tgt* in the Neo4j LPG model serve the same purpose as (a3) ρ in Angles' generic definition which does not differentiate between node and edge labels but they are further differentiated in Francis et al's Neo4j model. Likewise, Angles' (a4) λ is delineated as (f6) λ and (f7) r in the Neo4j model.

The differences in the tuple definitions between RDF graph and property graph can be addressed through direct transformations or indirect transformations. The latter is especially necessary when dealing with predicate-object property transformations (Voegeli 2018). Nevertheless, there are cases where it has been done successfully (Matsumoto et al 2018, Hernandez et al 2015; Hartig 2014). RDF entailment and reification further complicates such transformations. However, section 4.3.4 below will

provide further detail regarding the rationale used in this research to minimise transformation issues between the two tuple forms by adopting attribute and relationship representational rules from Structure Mapping Theory (SMT).

Firstly, to identify the elements or sets of elements of interest in conservation for representation, we consider the data directly. In Timothy Hannigan's essay on disambiguating social structure from text, he asserts:

It is not heaps of transactional data that make an inquiry scientific. Being scientific is an effect of work done to establish stable, quantifiable concepts...the concepts are a prerequisite for the existence of the data. [emphasis added].

The data we collect reflects the knowledge deemed significant for our purposes. To transpose data gathering from traditional tabular practices towards a more networked and systems-based approach demands consideration of the data in sets. Figure 4.4 provides an example of how systems of information in conservation are often siloed, providing the frames for day-to-day practice. We can imagine each silo as a set of elements. For the first silo in the top left of figure 4.4, a Health and Safety set, the elements can include the risk assessments (RAs), materials safety data sheets (MSDS), and COSHH²⁰ documents regarding hazardous materials. There is a high certainty of overlap between elements in this first set with elements in the next set, Materials.

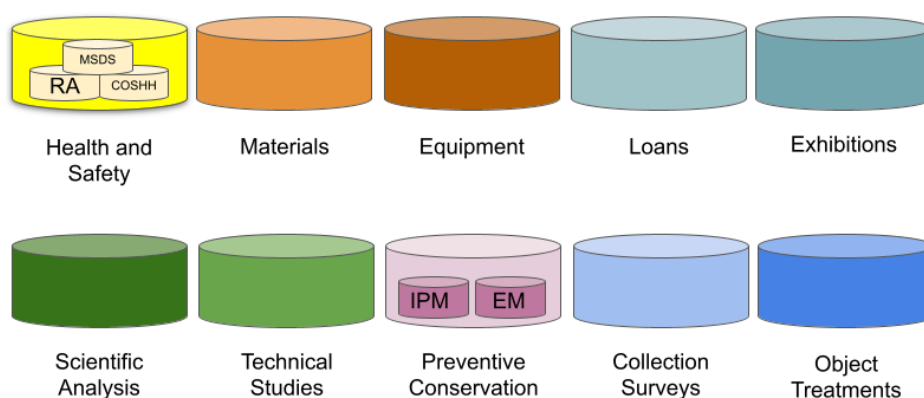


Figure 4.4. An example of systems of information in conservation (not exhaustive).

²⁰ The *Control of Substances Hazardous to Health (COSHH) Regulations 2002* requires employers in the UK to control substances that are hazardous to health by putting in place measures to inform, prevent and reduce exposure to such substances.

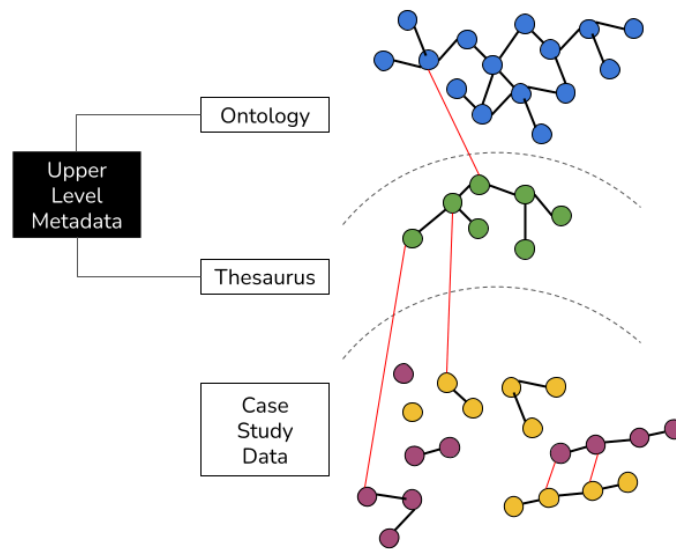


Figure 4.5. Hypothetical representation of datasets connected as subgraphs.

Figure 4.5 provides an hypothetical representation whereby elements in the first set (yellow, S_y) are linked with elements in the second set (brown, S_b) via a route along the thesaurus graph, (green, S_t). That which were separate datasets: S_y, S_b, S_t are now subgraphs in a larger graph, G . That is:

$$G \supseteq S \text{ (G includes S)}$$

Each dataset (including metadata sets) is transformed into a *data graph*. Linking these data graphs together to improve semantic representation is known as *enrichment*.

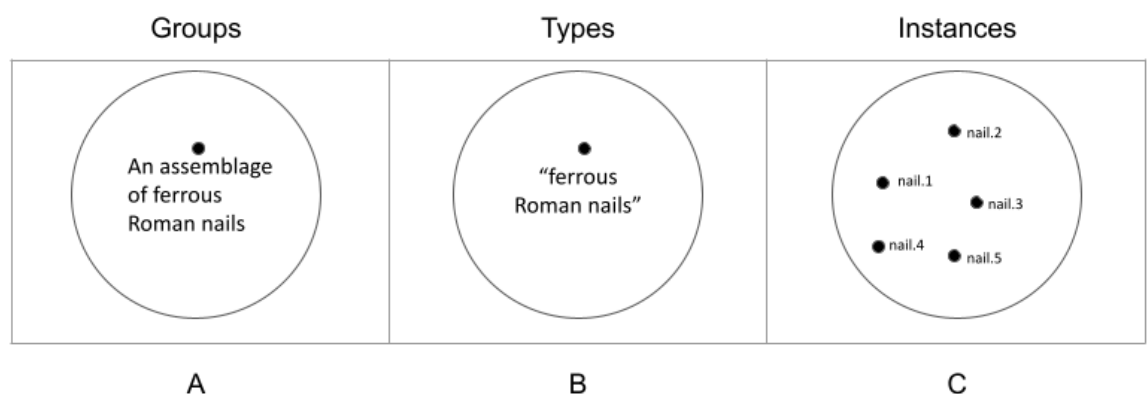


Figure 4.6 Conceptualisations of sets

Set conceptualisation is not limited to the datasets themselves, but also applies to the contents of each dataset, which has direct implications on the modelling of each as a subgraph and therefore on the overall modelling of the knowledge graph. Figure 4.6

gives three different conceptualisations using archaeological examples (A, B, C), that can form a set. 'A' is an example of a situation where the documentation only refers to a whole collection of things and not the parts of the whole. We are told by the name/label that the whole is 'an assemblage of ferrous Roman nails' so we can infer that the 'parts' are likely each a ferrous Roman nail, but representation at this level remains as a singular entity denoting an instance of the group itself: 'an assemblage...', and therefore, it is a set of one. To model 'A' in a property graph, we can leverage the *node label* and *properties* as set identifiers. For example:

```
(:Assemblage
  {findLocation: "quadrant 4",
   strata: "4b",
   findDate; 20180622,
   bagNumber: "1",
  content:"ferrous Roman nails"})
```

The label and property combinations can be used to disambiguate between sets and subsets, for example, if there is another similar entity:

```
(:Assemblage
  {findLocation: "quadrant 4",
   strata: "4b",
   findDate; 20180622,
   bagNumber: "2",
  content:"ferrous Roman nails"})
```

Node (1) and Node (2) are distinctly different as they represent different entities with different bagNumber. (They may even have entirely identical labels and properties but can be distinguished within the system with different UUIDs, universally unique identifiers). The :Assemblage label identifies the two nodes as belonging to the same *entity set* (i.e. a super-set), as both nodes refer to entities of the same type and they have the same attributes, that is, the list of property keys (e.g. findLocation, strata, findDate, bagNumber, content).

Example 'B' in Figure 4.6 provides an *entity type* or category of entities, representing all situations of "ferrous Roman nails". Depending on the needs and purposes of the graph model, further decomposition may be necessary as depicted in Figure 4.7. (This can be

achieved by importing an existing reference resource that has the decomposed values as entities that can be connected, i.e. *enrichment*). Modelling an entity type as a node improves querying via graph patterns, especially when pattern matching for paths.

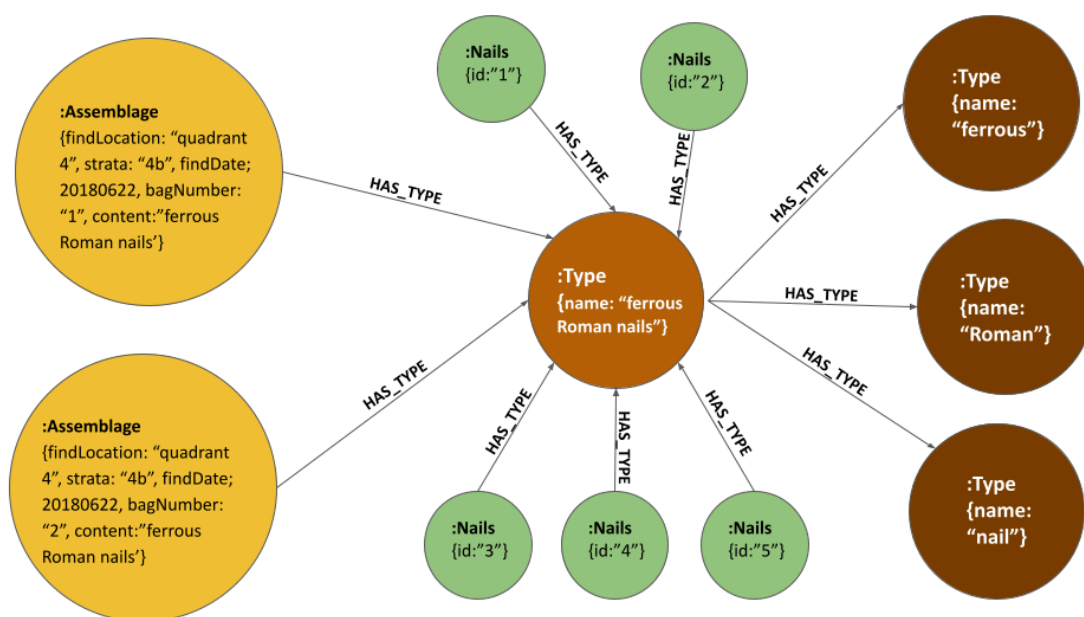


Figure 4.7 Example of further type decomposition using the example sets from Figure 4.6.

Example 'C' can represent an explicit set of instances (consisting of nails numbered 1 to 5), or it can represent a set that was 'defined on the fly' with a query, for example, a Cypher MATCH query²¹ to find all individual nails with the n.description property for 'ferrous Roman nails' and that has not already been linked to an :Assemblage, i.e. with degree 0 (assuming, in this case, individual nail nodes have no other relationships):

```
MATCH (n)
WITH n, size((n)-[r]->(b)) AS degree
WHERE n.description = "ferrous Roman nails" AND degree = 0
RETURN n
```

The results are a set of entities that match the graph patterns in the query.

In real world scenarios, records on artefact assemblages and each individual object within those assemblages can be scant and inconsistent due to constraints in time and resources. By using a graph-based system, it is flexible enough to represent local and ad hoc relational patterns that may not necessarily be universal to the whole system. In a

²¹ The letters "n", "r", and "b" are variables in the example query and are not fixed values.

labelled property graph model, semantic subsets are made explicit via node labels, relationship types and property keys. Set A and Set C might be referring to the same collection of nails in reality. Although each node has been modelled as different conceptualisations and representations, (i.e. 'A' is a set of 1 whereas 'C' is a set of 5), this is reflected in their labels and property attributes, which may serve different purposes, but nevertheless, both nodes can be linked²² via a relationship (see Figure 4.8), for example:

(A) - [IS_COMPOSED_OF] → (C)

Both sets are related and this can be made explicit, resulting in a triadic closure between a representation of a whole, each part, and the entity type they share.

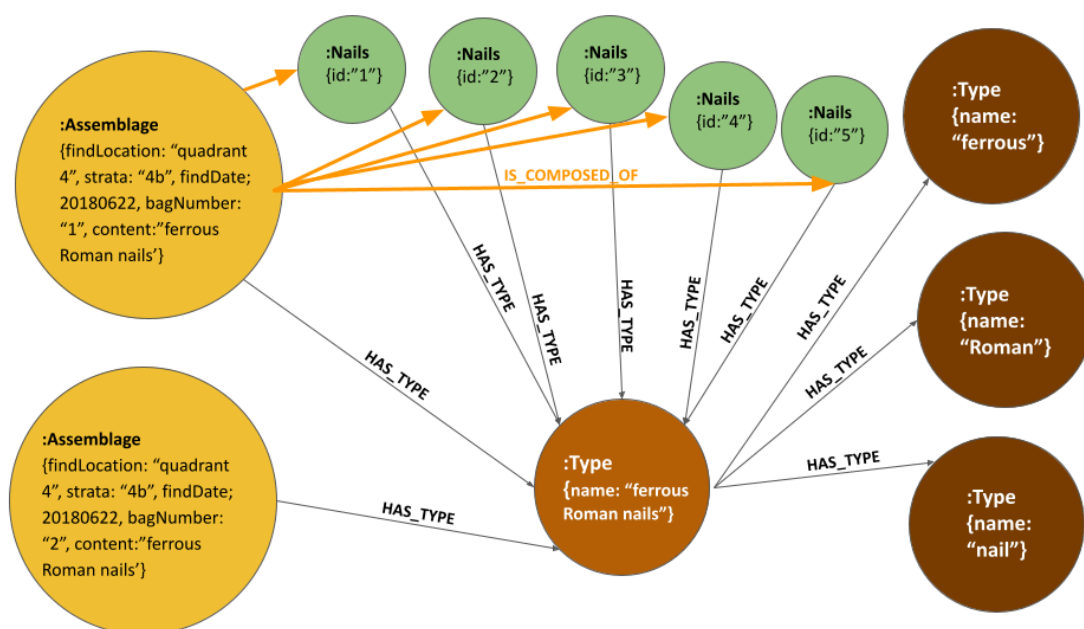


Figure 4.8 Modelling of the sets from Figure 4.6 and how they interrelate.

Filtering through nodes, edges, and properties (in a property graph) by specifying sets using labels and property keys is one way of modelling, querying, and analysing the contents of a graph. An alternative method is to investigate the graph's structure or topology. The advantages of a structural approach is that it is data agnostic. In cases where there is no prior knowledge of the data content, its labels, or property keys (and aside from running initial sampling queries), structural approaches can provide quantifiable insights.

²² For modelling edges, one can adopt predicates from existing KOS systems such as an ontology or derive an edge set from the dataset itself via verb lists.

Data agnostic subgraphs can be identified at the local scale (i.e. using specific nodes and edges) or meso-scale, while the global scale refers to the overall graph network (Tantardini et al 2019). Rombach et al (2014) identify meso-scale structures with *community structures* where “the role of suitable subnetworks is possibly evidenced”. Graphlets are local scale subgraphs comprised of 3-node, 4-node, and 5-node structures – the simplest network structures. When the presence of certain graphlet structures are statistically significant, the graphlet is considered a motif of the graph. Thus, zooming in on the local levels can reveal patterns in the building blocks of larger graph networks. For example, Figure 4.9 below shows three subgraphs of the graph model in Figure 4.8. Figure 4.8 can itself be the subgraph of a larger global graph. More on how subgraphs can be leveraged in the analysis of a graph will be covered in section 4.4.3. *Triangles, Graphlets and Motifs* as part of the section on *Graph Theoretic Analysis*.

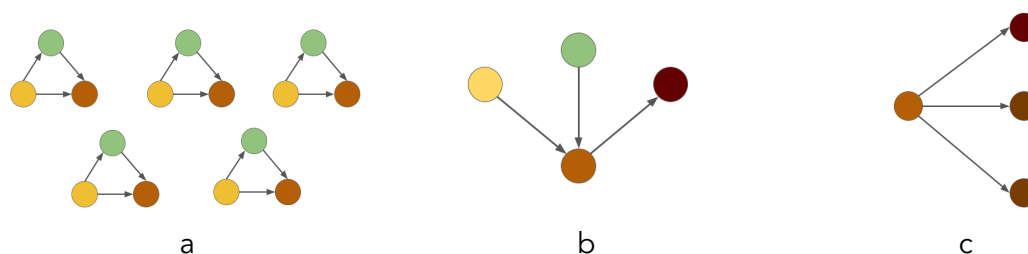


Figure 4.9. Examples of subgraphs of Figure 4.8.

The conceptualisation of sets serves a significant representational function in the structuring of a graph. In practice, when conservators conduct object and treatment assessments, it is necessary to consult multiple sources of information to build a better understanding of the object, such as its material, history, context and use (e.g. for travel or exhibition as a loaned item). For conservation, the representation of sets and building connections between sets offers a method to model the practice of cross-referencing through records and authority documents.

4.3.3 Categorical Representation and Graph Enrichment

In *Metaphors We Live By*, Lakoff and Johnson proclaim:

Once we can identify our experiences as entities or substances, we can refer to them, categorize them, group them, and quantify them—and, by this means, reason about them (Lakoff and Johnson 2003, 25).

When mapping data to an ontology (e.g. the CIDOC CRM) in RDF, the classes are themselves defined as *“a category of items that share one or more common traits serving as criteria to identify the items belonging to the class”* (Le Boeuf et al. 2016). A graph-based modelling approach allows for the representation of varying levels of semantic detail and specificity, from universal concepts to particular instances, all within the same graph. Shaban-Nejad (2012) describes categorical representation *“as the process of expressing things in different modes and layers of abstraction based on similarities and differences in their attributes and relations.”*

How particulars relate to universals are often inferred by the domain specialist but may not be explicit in the raw data. Figure 4.10 demonstrates the relationship between the specific (particulars) to the general (universals). Accordingly, the more generalised the representation, the larger the set of entities that can be encompassed by the abstraction while a more specific representation reduces the set of possible entities to match that abstraction. Categoricals reside on the spectrum in between the most general and the most specific and helps to further segment abstraction levels. Mapping datasets to categorical representations in a graph improves connectivity, makes explicit the level of specificity a concept is situated, and in turn enhances querying and visual graph explorations (Ristoski and Paulheim 2016, §9). Therefore, by conceptually grouping things in this way, we are defining further sets within our models. For example, ‘blue pigment’ can be an entity type or category for a set consisting of ‘azurite’, ‘lazurite’, and ‘cobalt blue’ (Reedy and Reedy 1988, 97). Representation of conservation materials at a general level allows for reasoning on materials in terms of its types and its attributes before pointing to specific cases (instances) of its use.

Graph enrichment procedures can be automated, supervised, or manual (Blanke et al 2015) and are those activities along the data pipeline that not only add further nodes and/or connections (e.g. additional datasets) to an existing graph model, but also improves the semantic depth of the content. Graph enrichment can be achieved by combining similarly related datasets with structural correspondences or by identifying additional relationships with metadata to improve semantics (Cheung and Shin 2000). For example, Hagaseth et al (2016) demonstrated enrichment by including a thesauri of 1500 terms in English, French, and Spanish that enhanced the matching of entities in their RDF-based ontology for digital maritime regulations.

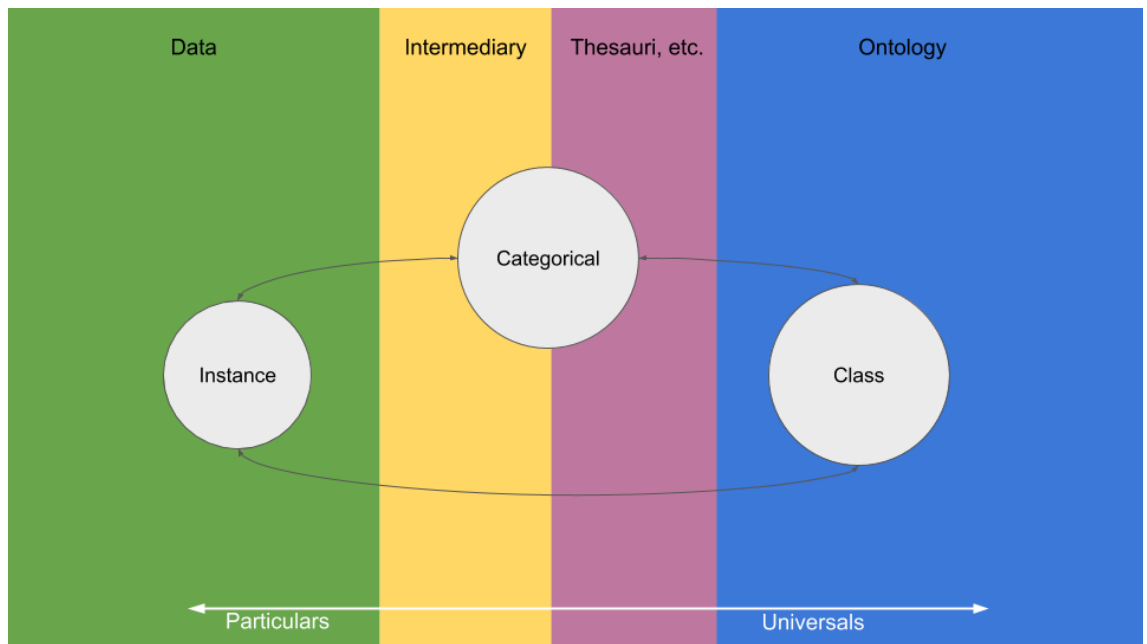


Figure 4.10. A demonstration of connectivity across representational levels.

Access and connectivity to various levels of detail and granularity are aided by “intermediaries”, such as controlled vocabularies and curated thesauri, local institutional-level categoricals or “folksonomies” (Plangprasopchok et al 2010; Price 2019) where it is more specific than an ontological class (a formalised generalisation) but less specific than a discrete data point (Doerr 2009; International Council on Archives 2000). The cyclic graph as represented in Figure 4.10 can be encoded as a property graph directly, allowing for traversals in both directions along a cyclic path. The modelling method needs to allow for mapping from the particular representational level of discrete data nodes directly to these categoricals, forming a continuum for inferential traversals (as denoted by the bi-directional arcs in Figure 4.10).

Figure 4.11 shows an annotated version of the ‘systems of information in conservation’ diagram (Figure 4.4) demonstrating how the content of each hypothetical silo can be identified against the spectrum of abstract representational levels (Doerr 2018, personal communication). Notice how conservation methods are inherently categorical as they are meant to be applied again and again.

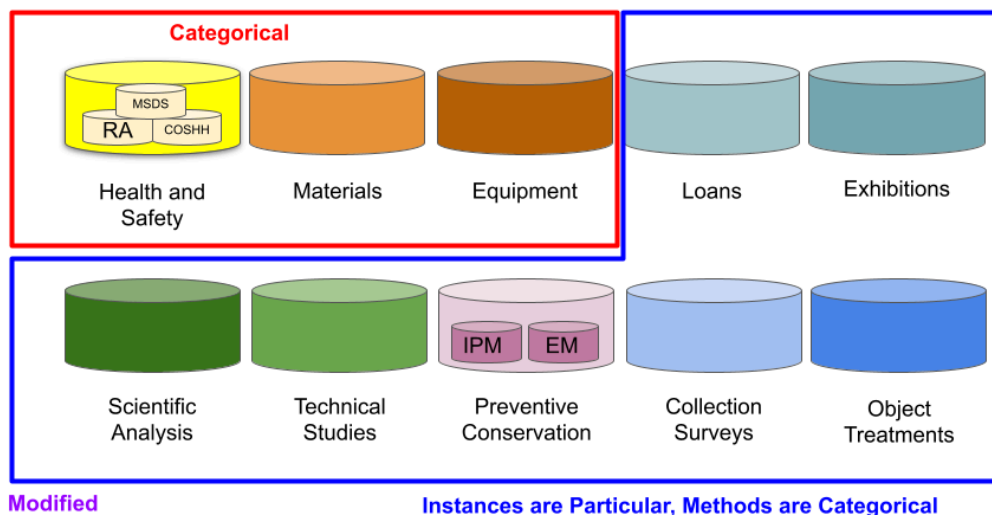


Figure 4.11 The systems of information in conservation from Figure 4.4 annotated by conceptual representational level.

Beyond compiled lists and thesauri from domain experts, categoricals can be derived from within datasets as high-frequency named entities or defined through statistical segmentation. Reedy and Reedy (1988) discussed statistical approaches to identifying significant and pertinent categorical variables such as “counting the number of objects falling into each category (a frequency distribution)” (*ibid.*, 40) or “dividing (analyzing) the observed variation of the continuous variable into components” (*ibid.*, 97).

Categoricals can also derive over time from relations; these are aptly referred to as relational categories (Gentner and Kurtz 2005). Examples in natural language include ‘anchor’ and ‘bridge’ where the entities, i.e. the things themselves, through the experience of an actual, physical anchor and an actual, physical bridge at some point in time, precede the subsequent abstracted meaning as metaphoric anchors and metaphoric bridges. Gentner and Asmuth (2019) assert that “there is a natural correlation between relationality and abstractness”. In fact, “100 highest frequency nouns in the British National Corpus reveal that about half of them refer to relational categories.” (Gentner and Asmuth 2019).

To connect heterogeneous datasets or subgraphs to create a larger graph, this study borrows the three standard SKOS RDF properties—“broader”, “narrower” and “related” relationships—to serve as edge types for linking nodes from one subgraph (or dataset) with nodes from another. Thus, when transformed into RDF, the bridged nodes serve to define and identify an inherent and extractable thesaurus from the larger graph.

4.3.4 Attributes and Relationships: Disambiguating “Property” with Analogical Reasoning

The term “property” can be a source of confusion as it can refer to two distinctly different representations, one in the context of RDF graphs and the other in the context of property graphs. For clarity, this thesis will explicitly state which “property” definition is being referred to using phrases such as “RDF property” versus “node property”²³ in the context of a property graph.

In RDF, a property is used to define a class. Each class has a set of properties and properties can have subsets of properties. The intention being:

Using the RDF approach, it is easy for others to subsequently define additional properties with a domain of eg:Document or a range of eg:Person. This can be done without the need to re-define the original description of these classes. One benefit of the RDF property-centric approach is that it allows anyone to extend the description of existing resources, one of the architectural principles of the Web [BERNERS-LEE98] (Brickley and Guha 2001).

Thus, in RDF, the property serves as the edge between the subject node and object node.

By contrast, in a property graph (PG) model, including labelled property graphs (LPG), the term “property” is an attribute stored on the node (a node property) or on the edge (an edge property).

Let’s take the following example statements:

The Titanic set sail for New York (USA) from Southampton (UK) in 1912.
While it had a capacity for 2,453 passengers and 874 crew, it only held 20 lifeboats, a maximum capacity for 1,178 people.

Figure 4.12 demonstrates how the semantic content of the above two statements can be modelled as RDF triples (subject - predicate/RDF property - object). In this example,

²³ Relationship (edge) properties are also possible in Neo4j labelled property graph (LPG), however, they are not used in this work.

there are two declared classes: 'Ship' and 'Location'. The RDF properties (i.e. the arrows in the diagram) are properties of the class 'Ship' and serve as the predicates in each triple statement, where the subject is the Domain of the RDF property, and the Range (or target) is the object. This would be explicitly expressed in RDFS (RDF Schema) to constrain and ensure the triple is semantically sound for inference. The subject in these triples, 'Ship: Titanic' refers to a single resource and would be defined with an IRI. The objects that the 'capacity_ ...' RDF properties point to are literal values, and are processed as integers or strings by the system. Processing literal values such as integers can include using arithmetic functions. For example, the hypothetical ship database can return an answer for the query 'what is the sum of all passenger capacities of all ships in the database?' by adding up all objects of property 'capacity_passenger' of datatype 'integer'.

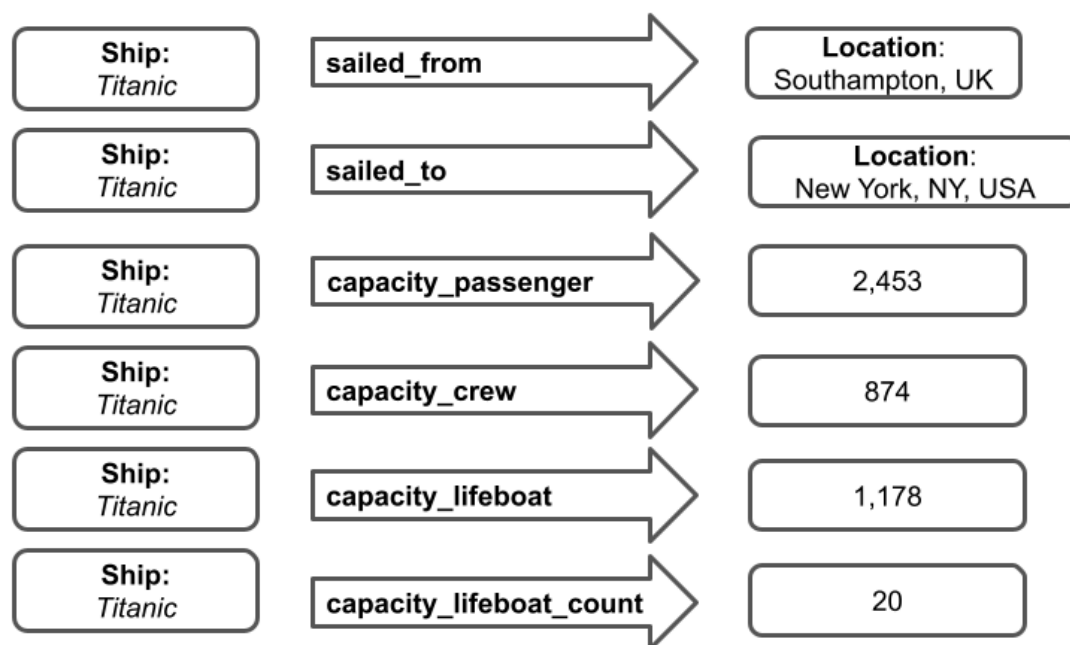


Figure 4.12. Illustration of the Titanic example statements as RDF triples (s-p-o).

RDF utilises a minimalist structure to encode on a single dimension of semantics. Herein lies its strength, but also it is a source of difficulty for beginners, especially those who are domain specialists in their areas. RDF can seem too abstract and diffused compared to the highly-connected perspective of one with domain knowledge.

Comparatively, Figure 4.13 demonstrates how the same semantic content in the example statements about the Titanic can be structured as a labelled property graph. Utilising the node property tuple, the node with label 'Ship' has several attributes stored

as key-value pairs²⁴: 'name: 'Titanic'', 'capacity_passenger: 2453', etc.. Here the attributes of the specific ship 'Titanic' form a set that more specifically defines and represents the ship as a single node. In fact, it is also possible to model the 'sailed_from' and 'sailed_to' information²⁵ as node properties, providing a fuller profile of the entity as a "snapshot" representation of the Titanic ship. However, modelling the locations as separate entities can enrich our understanding as we create a network of interactions. For example, there may be other ships in our database that have relationships to Southampton and New York. As a general rule for LPG, if the datum is likely to have relationships with many entities or attributes of its own, it is best to be modelled explicitly as a node. The result in Figure 4.13 demonstrates a more flexible and compartmentalised LPG model with multiple levels of connected semantics organised by sets.

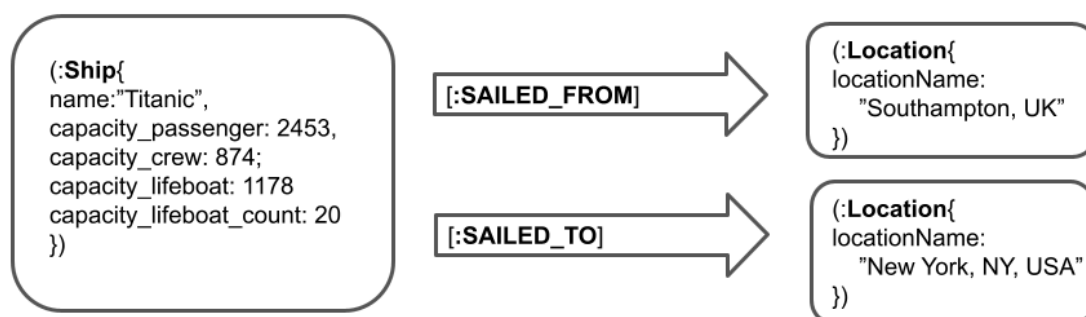


Figure 4.13 Illustration of the Titanic example statements in LPG.

The practice of abstractive decomposition required for modelling in RDF is a skill in itself. The challenges highlighted by conservation domain specialists to prepare RDF encoded data as part of Linked Data or Wikidata projects may be due to the misalignment between the diffuse RDF schema and their own highly-connected cognitive schemas in relation to their domain knowledge prior to engaging in enough semantic modelling work to have derived transitional schemas. Such discordances are recognised in cognitive load theory:

Although there appears to be a limit on the amount of units that can be loaded simultaneously in working memory, there seems to be no constraint on the size

²⁴The labelled property graph (LPG) structure using node labels, edge labels, node properties and edge properties. However, edge properties were not used in this study. An example of an edge property could be a disembarkation date in the :SAILED_FROM edge.

²⁵ Although only as an example, a key weakness in the above models is the labelling of the predicates as "sailed from" and "sailed to" which can lead to confusion by human readers or inference problems as in the case of the Titanic where the ship never reached its destination but had a planned destination of New York. A more accurate metamodel relationship would be 'had_designated_route_start' and 'had_designated_route_end'.

and complexity of these units of information (Sweller et al. 1998). More specifically, it is believed that one unit of information loaded in working memory (often referred to as 'information chunk') corresponds with one cognitive schema in long-term memory (Sweller et al. 1998). This can explain why a person seems to be able to store more information in working memory for tasks in which he is experienced, because for such tasks he was able to build up larger and stronger cognitive schemas in the past (Claes et al 2015, p.1410).

Modelling in LPG offers a metamodel paradigm that is more akin to the highly-connected cognitive schemas of the domain specialist while allowing for computational assistance for transforming LPG metamodels into RDF models for wider, FAIR usage. Broadly speaking, to transform data in an LPG structure into an RDF triple structure the node-to-node LPG structures are equivalent to a single RDF triple with the start node as the subject, the relationship as the predicate, and the target or end node as the object in the triple. Nodes with properties would be encoded as further triples—the node label serves as the subject, the property key would form the predicate, and the property value would be the object.

To ensure transformations from RDF to LPG, and vice versa, can proceed as smoothly and directly as possible, we must address the ambiguity in the term “property” and disambiguate it through the decomposition of predicates into attributes, which take one argument, and relations, which take two arguments, a distinction made in Structure Mapping Theory by Gentner (1983).

Gentner (1983) summarises her theory as follows:

structure mapping theory describes the implicit interpretation rules of analogy. The central claims of the theory are that analogy is characterized by the mapping of relations between objects, rather than attributes of objects, from base to target; and, further, that the particular relations mapped are those that are dominated by higher-order relations that belong to the mapping (the systematicity claim). These rules have the desirable property that they depend only on syntactic properties of the knowledge representation, and not on the specific content of the domain (Gentner 1983, 168).

In semantic modelling, mapping is identifying how a source schema can be transposed into another (target) schema (Bruseker et al 2017, 126-127). This is akin to the process we employ in cognition for deriving analogies as presented in Structure Mapping Theory.

Structure Mapping Theory (SMT) acknowledges there are sets of implicit constraints people apply to interpret analogy. Analogy is a form of modelling (McMarty 2005).

Firstly, there is a syntactic distinction between predicate types:

Attributes are predicates taking one argument, and relations are predicates taking two or more arguments (Gentner 1983, 157).

To illustrate this, Gentner offers these examples:

(a) LARGE is an attribute that takes one argument (x), as in:

x is LARGE

(b) while COLLIDE is an example of a relation taking two arguments (x, y), as in:

x COLLIDEs with y

The second rule is the *systematicity* principle which “conveys a system of connected knowledge, not a mere assortment of independent facts” (Falkenhainer, Forbus, & Gentner 1986; 1989). The systematicity principle is central to analogical thinking and has been supported by empirical studies in adults and children (ibid 1986). The third constraint is for one-to-one mappings of objects from source domain to target structure with “carryover of predicates” (Falkenhainer, Forbus, & Gentner 1986, p.2). Finally, SMT, like LPG, allows for multigraph representation (i.e. where the same pair of nodes can share more than one type of relationship).

In LPG, attributes are assigned to node and edge properties while relations are expressed as edges. This aligns with Voegeli (2018) who highlighted the need for indirect transformations on occasions with predicate-object from RDF to key-value node or edge property in PG (ibid. 38, 41), as demonstrated in Figures 4.12 and 4.13. There is also the fundamental difference between RDF and PG where RDF allows for properties of properties but PG does not (ibid. 42). A recognised method to rectify this as per Matsumoto et al (2018) is to have “[RDF] Resources mapped to nodes while literals [are] mapped to property values”. This transforms all RDF resources (entities/classes and predicates/properties) into nodes while using instead the Domain and Range RDFS constraints as edges to indicate which is the subject and which is the object. In fact, this is how Neo4j’s Neosemantics plugin tool transforms RDFS into LPG. The RDF classes and RDF properties are “nodified”, that is, modelled as nodes with the node labels :Class and :Relationship, while literals, which are object nodes that are never also subject

nodes (i.e. leaf nodes with only one neighbour) are transformed into node property values.

Modelling in LPG using characteristics and constraints of cognitive structure mapping which enables conspicuous modelling of domain knowledge while retaining the highly-connected coherence and systematicity for the domain specialist. Thus, modelling using an LPG structure works towards conspicuous structure mapping for inference.

As Gentner (1983) clarifies:

These representations, including the distinctions between different kinds of predicates, are intended to reflect the way people construe a situation, rather than what is logically possible (ibid, 156, 157).

Using property graph as a metamodel supports semantic conceptual modelling (Sequoiah-Grayson & Floridi 2022) closely along cognitive constraints and provides modelling as a method for conservation not only for encoding documentation but also for problem solving and scientific reasoning (Carbonell et al 1983) through analogical reasoning (Batha 2022) and similarity, predominantly through the identification of graph patterns. A representational graph theoretic approach is itself an analogical method:

Analogy may be used to guide reasoning, to generate conjectures about an unfamiliar domain, or to generalize several experiences into an abstract schema. Consequently, analogy is of great interest to both cognitive psychologists and artificial intelligence researchers (Falkenhainer, Forbus, & Gentner 1989).

Analogy is a kind of similarity in which the same system of relations holds across different sets of elements (Gentner and Maravilla 2018, 186).

In analogy, only the relational structure is shared, whereas in overall similarity the two representations share both relational structure and object properties [attributes] (ibid., 191).

Analogical reasoning occurs when there is mapping of knowledge from a source to a target domain where the "system of relations hold in both". The process itself consists of three stages: (1) memory and access, (2) mapping and inference, and (3) evaluation

and use (Falkenhainer, Forbus, & Gentner 1989, §2.1). The first informs upon that which is the base domain and that which is the target domain. Secondly, how the source and target correspond to each other are structurally mapped. Finally, there is an evaluation of the *structural alignment* and of *candidate inferences* to determine the validity and relevance of the analogy for the intended use (Wolff and Gentner 2011; Gentner, Bowdle, Wolff, and Boronat 2001, 200).

4.3.5 Star schema

The star schema is a network model structure²⁶ well-suited for heterogeneous information networks (HINs) and “the most widely used [for] conversion of relational databases [to graph] (Koukaras et al 2021, after Kong et al 2013). The transformation leverages an information object to act “as a hub, where other objects connect to it” (ibid.; Sun, Yu and Han 2009, 797), see Figure 4.14 below. The star schema is also a fundamental network model structure in the CIDOC CRM as shown in chapter 3.5.1 (see figures 3.18 and 3.19).

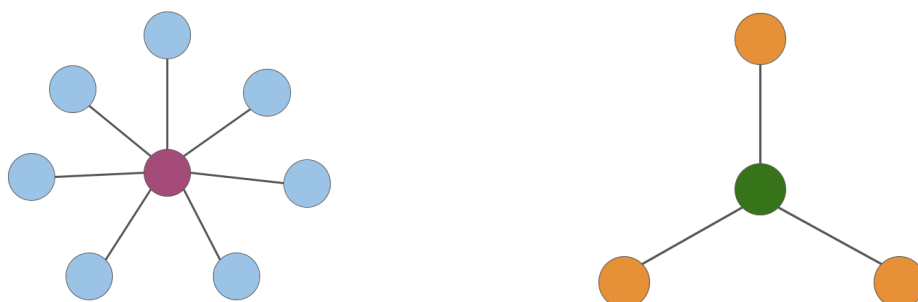


Figure 4.14. Generic star schema diagrams for illustrative purposes.

Bearing in mind the rules adopted from Structure Mapping Theory, these immediately adjacent, radial objects are best modelled from those entities with relational predicates that take two or more arguments, i.e. is likely to be connected further to other entities beyond its adjacent hub node. On the other hand, those attributes that are predicates taking one argument are best modelled as node properties. The star schema pattern can be expected to repeat for each event instance, for example, in a bibliographic graph, each central hub node can represent a written work and immediately adjacent neighbouring nodes can include the author, publisher, etc..

²⁶ For clarification, a graph-based star schema is not to be confused with a table-based star schema in a relational database where radial dimensional tables link to a hub fact table.

The star schema network model structure is employed in data management for data mining, or extracting meaningful or useful data from large datasets where clustering or similarity algorithms are used to predict or classify content. “Multi-hub networks” are a step up from star schema “in terms of information complexity” and are used in complex sciences such as bioinformatics and astrophysics (Koukaras et al 2021). Building from a foundational base of star schema structured data enables scaling up towards multi-hub networks as the graph becomes enriched with additional data.

4.4 Graph Theoretic Analysis

A graph theoretic approach enables the use of measures derived from graph features for identifying patterns within networks and for comparing networks. The structure of the graph can be leveraged to explore the conceptual and semantic relationships being modelled. In *Complex Graphs and Networks*, Chung and Lu (2006) assert:

Graph theory has emerged as a primary tool for detecting numerous hidden structures in various information networks, including Internet graphs, social networks, biological networks, or more generally, any graph representing relations in massive data sets (ibid, vii).

However, before we begin searching for hidden patterns, what are the obvious or typical graph patterns for conservation? What are the existing relational structures in our data? To address this, this study applies graph theoretic data profiling techniques to compare different datasets or different models of the same dataset by identifying reference structures and measures to enable comparisons.

Several types of graph measures were chosen to provide ‘fine- to medium-grained’ and ‘coarse-grained to overall’ views of the case study-derived graphs in order to aid characterisation of each graph’s topology (shape). These are:

- Order and Size
- Density/Sparsity, Degree and Clustering
- Subgraphs (including Triangles, Graphlets and Motifs)
- Paths and Distance (including Shortest Path and Diameter)
- K3,3 bipartite graph for detection of planarity
- Eigenvector Centrality

As such a study on conservation data has not been recorded in the literature, what these features can reveal about conservation remains to be seen. Nevertheless, the chosen measures are well-documented measures to begin these investigations.

4.4.1 Order and Size

The total number of nodes in a graph is known as the order and the total number of edges is the size of the graph. The models in this study are directed graphs with directed edges, as opposed to *undirected edges*²⁷. The distinction here is, when assuming all relationships between two nodes are reciprocal, this is modelled using a single edge in an undirected graph, or with two edges in a directed graph—a *natural* direction and a *reverse*. Hence, it is expected that edge counts in a directed graph will be twice the number of an undirected graph representation, assuming all edges are part of reciprocal pairs.

4.4.2 Density/Sparsity, Degree, and Clustering

The density or sparsity of a graph is an indicator of connectivity and can be determined using various measures, the simplest global profile is computed from the order and size. The definition for a sparse graph is one where “the vast majority of nodes are connected only to a small percentage of other nodes” (Kumar et al. 2015), such that the number of edges tends towards “being linear, i.e., within a small multiple of the number of vertices” (Chung and Lu 2006, 2) while dense graphs have more connections between nodes, where “the number of edges is closer to the number of nodes, than to the square of the number of nodes” (Kumar et al. 2015, *ibid.*) or “a quadratic number of edges in terms of vertices” (Chung and Lu 2006, *ibid.*).

The edge density is the number of actual edges divided by the number of possible connections for a network with V nodes and E directed edges (after Kaiser 2008). Edge density is a measure of sparsity. Graphs with more leaf nodes (i.e. have only one neighbour) and isolated nodes (i.e. unconnected nodes or “islands”) would tend towards being more sparse. Edge density for a directed graph, D , is:

$$D = \frac{E}{V(V-1)}$$

²⁷ NB: While the overall graph models are directed, certain algorithmic analyses in Neo4j can be parameterised to treat a graph projection as undirected.

Degree and clustering measures are other ways to determine graph density. The degree is the number of edges connected to a node (Newman 2003). Similar to the edge density measure is the average node degree, which is the average number of edges per node. It is calculated by dividing the number of edges by the number of nodes, and then multiplying by two for an undirected graph or when not differentiating between in-degree and out-degree for a directed graph (Bales and Johnson 2006, 454).

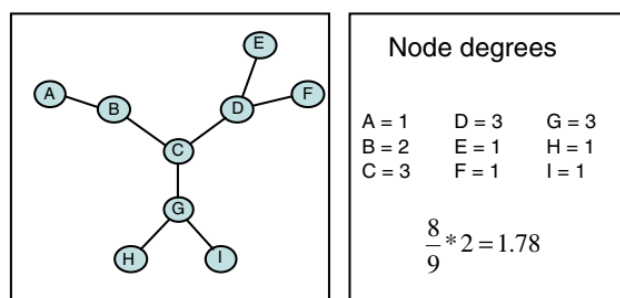


Figure 4.15 Diagram to illustrate node degrees. For example node A has a degree of 1 and node D has degree of 3. (Image Source: Bales and Johnson 2006, Fig. 1.)

Degree centrality is a measure of the number of incoming or outgoing (or both) relationships from a node. The Neo4j Graph Data Science Library's (GDSL²⁸) degree centrality algorithm supports parameterising the orientation of relationships as "NATURAL" for out-degree relationships, "REVERSE" for in-degree relationships, and "UNDIRECTED" for a sum of both out-degree and in-degree relationships of each node. Degree measures provide insight into the graph structure by identifying high-degree connections or hubs where a given node has many neighbours.

Patterns of higher connectivity amongst a small number of nodes in localised areas is known as clustering. Strong local clustering is a characteristic of networks where "adjacent nodes show significant correlations in their properties" (Park and Barabási 2007) with neighbours of a given node "more likely to be connected to one another than would be expected through chance alone" (Bales and Johnson 2006, 454). Cluster analysis can assist in, for example, identifying key terms for building controlled vocabularies. Such measures are often categorised as 'community detection' algorithms.

²⁸ Neo4j 2022. "Degree Centrality". *The Neo4j Graph Data Science Library Manual v2.1* <https://neo4j.com/docs/graph-data-science/current/algorithms/degree-centrality/>

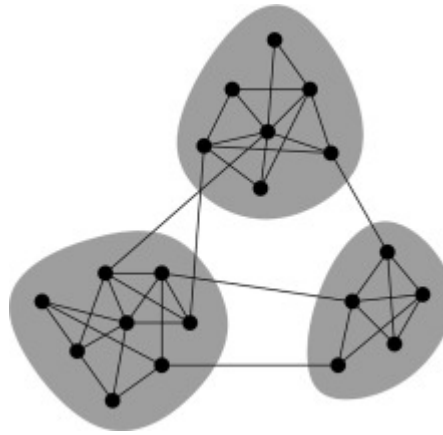


Figure 4.16. Illustration of the structure of graph clusters. Image source: (Roy and Chakrabarti 2017, Figure 1)

The clustering coefficient, C , “is the average fraction of pairs of neighbors of a node which are also neighbors of each other (Newman 2000). That is:

by counting the number of edges between the node’s neighbors, and then dividing by all their possible edges. This results in a value between 0 and 1, which is then averaged over all nodes in a graph (Bales and Johnson 2006, 454).

In a fully connected network [for example] in which everyone knows everyone else, $C = 1$ (Newman 2000).

The closer the local clustering coefficient is to 1, the more likely it is for the network to form clusters (Pavlopoulos et al 2011).

Wills and Meyer (2020) defines a clustering coefficient metric for graph comparison “as the ratio of the number of triangles to the number of connected triplets of vertices in the graph” in a Watts-Strogatz model, that is, the “simplest random graph that has high local clustering and small average shortest path distance between vertices” (Wills and Meyer 2020, §2.4.4). [The Watts-Strogatz model (1998) is a random graph with small-world properties.]

Given a node n , the Neo4j GDSL Local Clustering Coefficient algorithm “describes the likelihood that the neighbours of n are also connected”.²⁹ Akin to Wills and Meyer, the Neo4j version of the algorithm uses the number of triangles a given node is a part of, T_n ,

²⁹ <https://neo4j.com/docs/graph-data-science/current/algorithms/local-clustering-coefficient/>

and the degree of the node, d_n , to calculate the local clustering coefficient for each node, C_n :

$$C_n = \frac{2T_n}{d_n(d_n - 1)}$$

Additionally, the algorithm can compute the average clustering coefficient for the whole graph. This is the normalised sum over all the local clustering coefficients (ibid.)

Pavlopoulos et al (2011) clarifies that despite hub nodes appearing to be well-connected themselves, if their neighbours are not well-connected with each other, and are in fact, not connected to each other at all, then such a hub node would have a clustering coefficient of 0. This is relevant to conservation graphs as despite the event-centric star schema structure forming the basis to the modelling principles in this study, they do not, in graph theoretic terms, constitute clusters.

Kaiser (2008) has proposed a revised mean clustering coefficient measure as the previous measure defined by Watts and Strogatz tended to underestimate connectivity by not counting nodes with 1 or 0 neighbours (i.e. leaf nodes and isolated nodes, respectively). The results of Kaiser's investigation into the contribution to connectivity afforded by leaf nodes and isolated nodes found a difference up to 140% higher with the revised measure than with the traditional local clustering measures. Such underestimation can have a significant impact on small-world features, even tipping a network towards a non-small-world network definition³⁰.

$$\theta = \frac{\text{count of nodes with degree} \leq 1}{\text{total node count}}$$

In order to document indicators of underestimation, as per Kaiser's recommendations, this study recorded the edge density, both the number of leaf and isolated nodes, and the ratio of nodes with less than two neighbours (degree ≤ 1) divided by the total node count, θ , for each graph profile.

³⁰ Kaiser has instead proposed redefining the traditional Watts and Strogatz measure as an alternative measure, D , for disconnectedness "which is less influenced by leafs and isolated nodes" (2008).

4.4.3 Triangles, Graphlets and Motifs

Section 4.1.3 *Sets, Tuples, and Subgraphs* above introduced subgraphs and demonstrated how heterogeneous datasets brought together and modelled as a single network can themselves be subsequently identified as subgraphs of that network. Of course, other fine-to-coarse grained subgraphs can be identified and induced, providing another structure-led approach to interrogating a network. In addition to serving as indicators of density or sparsity, clusters can also be further examined as subgraphs. For example, a neighborhood subgraph is “a subset of nodes in a network consisting of a node and all of its neighbors” (Bales and Johnson 2006; after Steyvers and Tenenbaum 2005). Likewise, the concept of triangles have come up in the previous section along with a few ways to exploit their presence within a network. This section further expands upon the use of small, fine-grained subgraphs for characterising local-level structures.

Firstly, a triangle or triad is a type of graphlet and is a complete graph, that is, all nodes within a triangle are connected to each of the other nodes in the triangle. Graphlets are simple networks that can be derived from only 3, 4, or 5 nodes and are the smallest network structures (Espejo et al 2020). That is:

$$k \in \{3,4,5\}$$

Graphlets are “small induced subgraphs of a larger network that appear at any frequency” (Yaveroğlu et al 2014). Here, “induced” means a sub-graph that is identified by identifying specific nodes and including all the edges between these nodes. Pržulj, Corneil and Jurisica (2004) proposed *graphlet frequency* (a count of occurrences of specific graphlets in a network) “as a new network parameter”. These local structure subgraphs inform upon the overall network (Stone, Simberloff and Artzy-Randrup 2019) and can be used to compare networks (Tantardini et al 2019 after Sarajlic, et al 2016; Aparicio, Ribeiro, and Silva 2015). When certain graphlets are statistically prevalent within a network, these are then referred to as graph motifs. Nevertheless, both “graphlet” and “motif” are used somewhat interchangeably in the literature. Milo et al (2002) undertook early research into motifs. Subsequently, Milo et al (2002) utilised these substructures to identify *significance profiles*, which were then found to match across datasets from similar domains. Domain-specific interpretations of network topology using motif patterns continue to be used for network characterisation (Sarajlic

et al 2016), albeit motif counting can be computationally resource heavy (see Appendix A: Neo4j Configuration).

Figure 4.17 below by Abuoda, Morales, and Aboulnage (2020) shows the twenty-nine graphlet permutations. With three nodes, there are two possible ways of ensuring all of the nodes are connected (e.g. m3.1 and m3.2 in Figure 4.17). When there are four nodes, there are up to six possible edge combinations (e.g. m4.1 - m4.6). With five nodes, there are up to 21 edge combinations to connect all the nodes (e.g. m5.1 - m5.21).

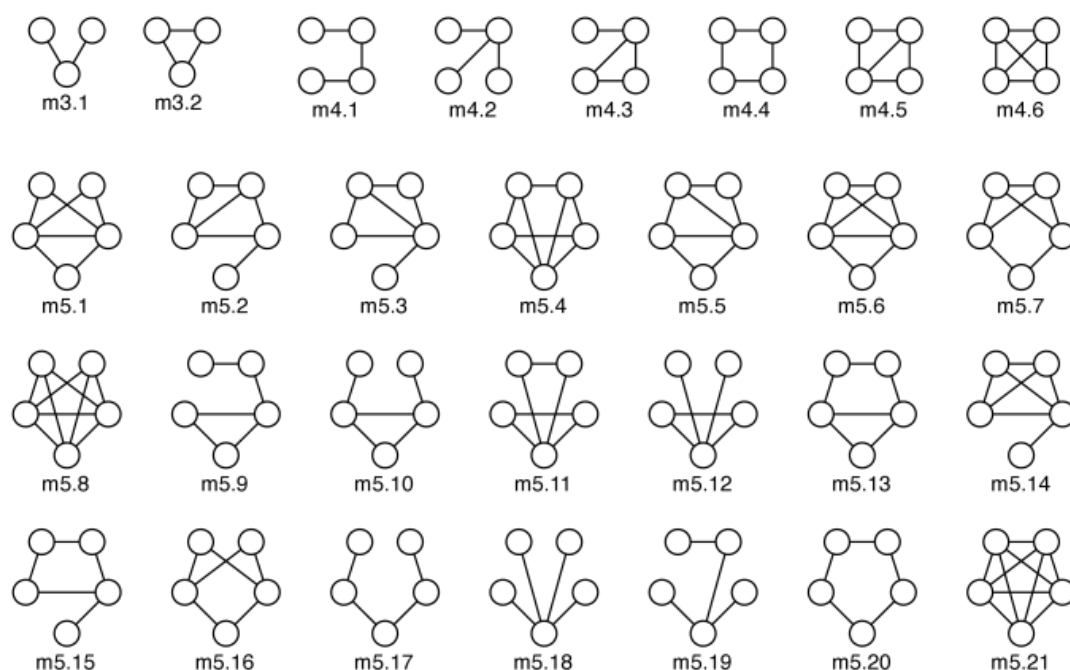


Figure 4.17 Motif patterns in sequence after Abuoda, Morales, and Aboulnage (2020).

This study uses the naming pattern for motifs devised by Abuoda, Morales, and Aboulnage (2020) with each motif name following the $m_{k.n}$ pattern (Figure 4.17) where “ k is the number of nodes in the motif and n is an ordinal number which identifies the specific edge pattern in the motif” (ibid. §2.1) as per the diagram. The undirected graphlet/motif patterns were executed using Cypher queries after de Marzi (2019) to aggregate frequencies of each motif. [Please see recommendations for using Przulj’s naming pattern (2006) for motifs in section 8.5.2 *Recommendations for Implementing Graph Theoretic Analysis*.]

The Global Triangle Count (GTC) is related to the m3.2 count but the GTC disregards counting the same set of 3 nodes that make up a triangle more than once, whereas the m3.2 count (as expressed in the Cypher query) does not discount such repetitive counts. The relationship between the two counts tends to be a factor of 6. For example, the motif count would consider A-B-C-A, B-C-A-B, and C-A-B-C, and their reverses to be separate triangle counts while the GTC algorithm counts it only once. Triangles in a graph are important features as triadic closures have been shown to correspond with better-connected networks.

4.4.4 Paths, Distance, Shortest Path and Diameter

A *path* in a graph G is a subgraph of G consisting of the sequence of edges that connects one node to another or whose vertices can be listed in some order (Benjamin et al 2017, 46). Moving along a path from one node in a graph to another node elsewhere in the graph is a *traversal*. The number of edges in the path is its length. The *shortest path* or *distance* between two nodes is the number of edges between the start or source node and the end or target node (Buckley and Harary 1990). Albert et al. (1999) calculated, at the time, that despite the World Wide Web containing 800 million documents, the average distance between documents was only 19 hyperlinks (edges), exemplifying a large but highly-connected graph.

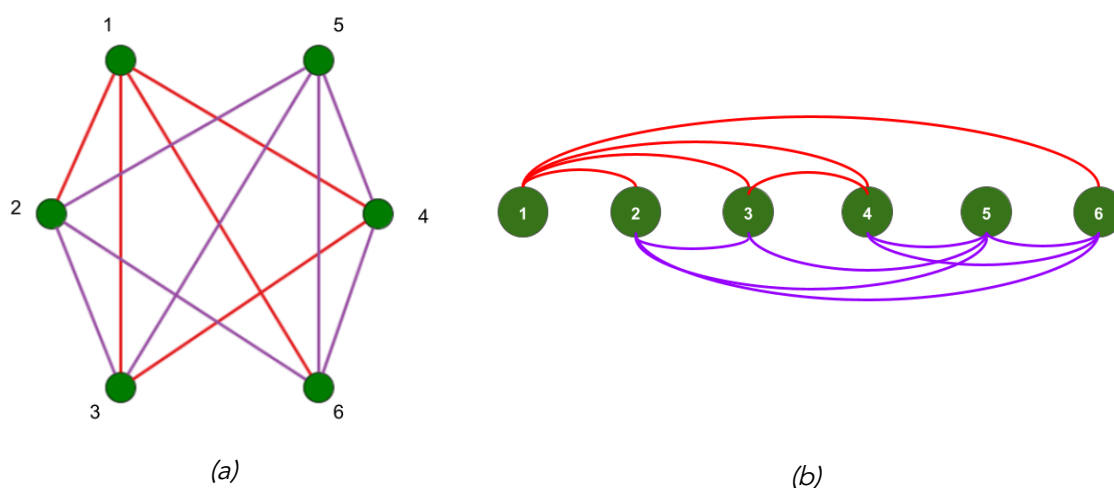


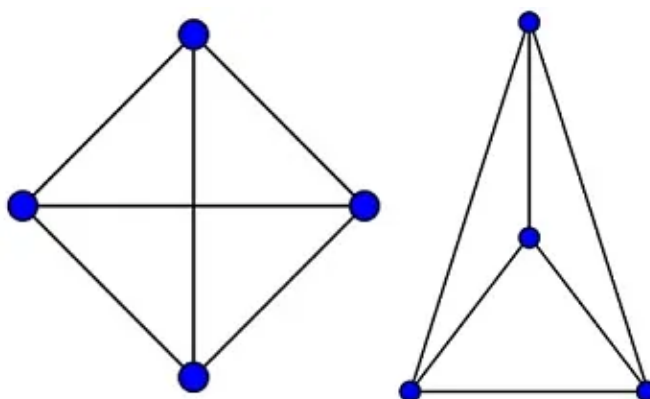
Figure 4.18 Isomorphic graphs. (a) Diagram of a cyclic graph (modified from image source: Pupyrev 2023) where a cyclic path begins at node 1 and traverses over a single red or purple edge, in order, to node 6 from whence it can traverse to node 1 again. (b) Diagram of a linear graph representation of the same cyclic graph in (a) [own work].

An open *trail* is a path where you pass each edge only once. It is open because the first and last nodes aren't the same, i.e. it doesn't loop (Benjamin et al 2017, 98-99). Vertices (nodes) can be repeated in a *circuit* but not the edges. A closed trail is also a circuit. *Cycles* are like circuits, but can only go through each node and edge only once, save for the start/end node. Cycles, circuits and trails are all denoted as sequences of vertices.

The diameter of a graph, G , is the maximum distance between two vertices. It is identified by ascertaining all of the shortest paths between all pairs of nodes in the graph and singling out the longest of these, hence it is the 'longest shortest path' (Newman 2003; Kumar, Wainright, and Zecchina 2015). It too can be used as a metric to gauge overall connectivity of a graph, for example Chung and Lu (2006) showed it as an indicator of the degrees of separation to consider in order to reach all people in a social network. In the aforementioned example by Albert et al (1999), when representing webpages as nodes and hyperlinks (URLs) as edges, the calculated average diameter of the World Wide Web back then was only 19 links. However, Albert et al (1999) also considered the growth of the World Wide Web over time and calculated that even if there was "the expected 1,000% increase in the size of the web over the next few years [that] will change $\langle d \rangle$ [the diameter] very little, from 19 to only 21."

4.4.5 Planarity and $K_{3,3}$ Bipartite Graphs

A planar graph is a graph that can be drawn so that no edges cross. Planarity offers a means to partition large non-planar graphs and identify isomorphic structures. For example, it is used in computer chip design for integrated circuits (Valiant 1981) and to study urban patterns across diverse global cities through a comparable and computable representational dimension (Cardillo et al 2006).



(a)

(b)

Figure 4.19. Isomorphs of K_4 where (a) is a nonplanar representation and (b) uses a planar representation. (Image Source: Hoang 2018). In graph theoretic terms, a planar graph is one that does not contain subdivisions of K_5 or $K_{3,3}$.

Figure 4.19 demonstrates how K_4 can be drawn as a nonplanar graph (a) and redrawn as a planar graph (b). Both versions (a) and (b) are isomorphs. Real-world graphs are typically large and non-planar (Kobourov, Pupyrev, and Saket 2014). Therefore, graph models of conservation activity are expected to be non-planar. However, this has not previously been confirmed. Confirmation in regards to planarity can be achieved by identifying the presence of $k_{3,3}$ bipartite or k_5 -complete subgraphs as the occurrence of either subgraph renders the graph non-planar based on Kuratowski's proven Theorem (1930) which states:

A graph is planar if and only if it does not contain any subdivision of K_5 or $K_{3,3}$.

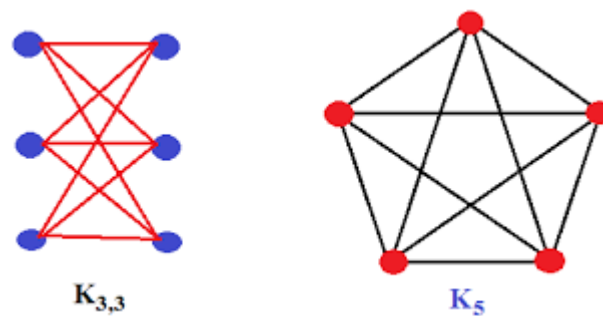


Figure 4.20 Representations of $K_{3,3}$ and K_5 graphs. (Image Source: Tienminh91, 2013).

Thus, in addition to the above motifs which include k_5 (numbered above as m5.21), the case study graph models were interrogated for the presence and frequency of the significant 6-node sub-structure of $k_{3,3}$ bipartite graph. This graph is bipartite as the constituent nodes form two groups and each node of each group connects to all nodes of the other group (making it a *complete* bipartite graph). Confirmation of non-planarity would be the first measured indication of complexity in conservation. It also opens up further downstream research potential for identifying and studying different facets of complexity in conservation through the study of cuts or "slices" which are portions of planar subgraphs within a non-planar graph.

4.4.6 Eigenvector Centrality

Pavlopoulos et al (2011) defines eigenvector centrality as a measure that ranks “higher the nodes that are connected to important neighbors”. Eigenvector centrality is used in this study as a measure of transitive influence, that is, to identify which nodes have greater influence in the graph. The measure provides a transitivity profile that can serve as a benchmark, akin to a snapshot of the graph at a particular point or a profile of a model for comparison with alternative build parameters. This helps to assess how a graph is connected and to track it as it evolves over time as nodes and edges are added or removed.

According to Dale (2017, 76), the typical calculation for eigenvector centrality is based on Katz (1953) and Bonacich (1972):

$$C_e(v) = \alpha \sum_{(u,v) \in E} \mathbf{D}(v,v) \mathbf{c}_e(u).$$

While working for the Department of Sociology at UCLA, Bonacich published his seminal work, *Power and Centrality: A Family of Measures* (1987). In it he diverges from Freeman (1979) who identified centrality as an interplay of three parameters: betweenness, c , nearness, α , and degree³¹, β :

$$c(\alpha, \beta)$$

However Bonacich demonstrated how β can be affected by direct and indirect influence through not just contacts, but contacts of contacts in a network. β can be positive (e.g. where “exchange in one relation is contingent on exchange in others” or negative (e.g. where “exchange in one relation precludes exchange in others”) (Bonacich 1987, 1171). This offered a new perspective on modelling power relationships. Previous centrality measures modelled proximity to high-status individuals, however, such a representation for betweenness, c , was:

hopelessly ambiguous; $c(\alpha, \beta)$ can give radically different rankings on centrality, depending on the value of β [degree] (ibid., 1181).

Instead, Bonacich used the example of power in bargaining:

In bargaining situations, it is advantageous to be connected to those who have few options; power comes from being connected to those who are powerless

³¹ “Degree” in this usage means a non-negative probability measure and not the graph theoretic term.

(ibid., 1171)...*one's status is a function of the status of those one is connected to*
(ibid., 1181).

Thus, unlike previous centrality measures, Bonacich proposed a centrality measure that is “the summed connection to others, weighted by their centralities” (ibid., 1172).

The use of the eigenvector centrality measure in this study is based on the work of Page et al (1999), Pavlopoulos et al (2011), Rodriguez (2009), and Dablander and Hinne (2019). The eigenvector centrality algorithm is similar to the PageRank algorithm that forms the basis of the Google search engine, its name deriving from the aspiration for assigning a ‘ranking for every page on the web’. Both algorithms rely on calculating eigenvectors. Page et al (1999) demonstrated how applying eigenvector centrality measures can aid in identifying web pages (i.e. nodes) with greater relevance or importance based on the premise that better-connected nodes with more incoming links ought to rank higher (Page et al 1999). Pavlopoulos et al (2011) identified several uses of eigenvector centrality to study genetic interactions, disease associations and network hubs in biological networks. Rodriguez (2018) applied eigenvector calculations towards lead community detection in the Linked Data Cloud and found a correlation between triple count and transitive ranking. Finally, while Dablander and Hinne (2019) found that centrality measures are not always indicative of causal influence, they found an exception with eigenvector centrality due to the measures accounting of the transitive importance of nodes. It has also been used as a substitute measure for complexity (Saha and Sarkar 2022). Thus, it serves as a strong candidate for determining the inherent influences within a modelled knowledge graph, whether the influences are derived from the CIDOC CRM ontology, the source heterogenous datasets used to construct the knowledge graph, or the modelling choices made along the way.

The Neo4j’s Graph Data Science Library implementation of the eigenvector centrality algorithm calculates “the centrality score for each node...[by deriving from]...the scores of its incoming neighbors.” (Neo4j, n.d.)

Relationships originating from high-scoring nodes contribute more to the score of a node than connections from low-scoring nodes. A high eigenvector score means that a node is connected to many nodes who themselves have high scores...Centrality scores for nodes with no incoming relationships will converge to 0. (ibid.)

4.5 Query-based Analysis and Inference

One of the recognised advantages of working in a labelled property graph environment, such as Neo4j, is the expressivity of the Cypher graph query language to modify, search and perform analyses on a graph (Francis et al 2018) to an extent where the “level of expressiveness...can blur the line between querying and data analysis.” (Gomes-Jr, Jensen and Santanche 2013; Gomes-Jr and Santanche 2015).

Query design itself extends the computational analysis method. The practice of query design translates research questions into machine-readable scripts. Queries enable specific searches and general exploration of database content. The nature of a graph-based model not only supports data analysis of static entities, but by leveraging directed relationships, it can also model dynamic processes (Polyvyanyy et al 2017). Querying also presents the means in which to encode validation (competency) questions to test the model design for its immediate purposes. This intimately linked relationship between model design and query design will be explored further in the next section.

4.6 Verification, Validation and Calibration

Verification, validation and calibration are significant and related activities in computational model development (Sankararaman and Mahadevan 2015). However, at this preliminary stage of developing modelling practice, it is pertinent to see the distinction between verification, validation, and calibration, and acknowledge the current limitations due to the small number of conservation graphs available for assessment.

4.6.1 Code Verification

Verification pertains to the accuracy of the encoding to represent the intended conceptual description and logic of the model (Rakha et al 1996; Thacker et al 2004). In simplest terms, a verification process determines if the computer code has been written correctly and therefore can be more clearly referred to as code verification (Stevens and Atamturktur 2017).

4.6.2 Model Validation

Validation pertains to the accuracy to which the model represents the real-world, for example, whether or not the application of the model yields results as expected (Rakha

et al 1996; Thacker et al 2004). Competency questions (as termed by Allemang and Hendler 2011, 308, 324) are the questions a model is meant to answer and presents one means of validation. Other validation metrics can be devised based on querying the model using high-quality data (Thacker et al 2004).

The data providers for this research, The National Archives (TNA), UK, have identified several areas for investigation that cannot be achieved or aggregated directly within their current database system. The Head of Conservation and Treatment Development, Sonja Schwoil, and Senior Conservation Manager, Sarah VanSnick, have framed their research interests as the principle data providers as follows (2018):

Our work aims at guarding our collection and providing access to it. Facing a vast collection dating over a thousand years, we need to prioritise our work efforts according to impact. For this we undertake research to continuously improve our preservation and conservation approaches. Furthermore our research aims to better understand our collection items, which again allows greater access to the information inherent to the collection item.

We are hoping to address the following subjects:

- 1. Impact of our work for prioritisation*
- 2. What does impact on our work? i.e. trends*
- 3. Quantification of certain treatment techniques/application of materials*
- 4. History/development of certain treatment techniques and application of materials*
- 5. Compare treatment efforts against preservation efforts (how often do items return for further/repeated treatments?)*
- 6. Relate item condition with places of deposit before their accessioning at TNA*

The research interests of the Collections Care Department at The National Archives have set the key purpose for the data model and database in this study (see Chapter 7.0). While not all potential user queries can be anticipated, it sets a baseline requirement for the data model to support queries aligned with these research interests. Similar interests from the wider conservation community regarding quantifying treatments, tracking trends, aiding prioritisation and the cross-referencing of conditions and treatments (Velios and St John 2022), means this current study, although limited in scope, has the

potential to support wider interests and applicability. Therefore, the validation (competency) questions derived from this work will be transferable in principle.

Materials in conservation are inextricably linked to object use, deterioration, significance, and aesthetics, and is therefore a principal consideration in conservation decision-making processes. What is it made of? What conservation materials were used previously? How are they used? How often are they used? Hence, a key element that the model must support is material-related queries. However, additional validation metrics will be necessary as the field of conservation knowledge graphs develop.

A fundamental assumption underpinning the Semantic Web is known as the triple-A slogan or AAA, "Anybody can say Anything about Any topic". Essentially, there is no "wrong way" to model. Nevertheless, in practice, there are pitfalls or "antipatterns" (Allemang and Hendler 2011, 313-324) to avoid. Modelling pitfalls also exist in the practice of creating labelled property graphs in Neo4j (Armbruster 2016). Modelling decisions, such as whether to model a property as an attribute or relationship, can have significant downstream consequences in terms of querying functionality and processing resources, for example, needing to write long, complicated or nested queries to access data in one paradigm versus more simple queries under a different modelling paradigm. Thus, model design and "queryable"-ness are inextricably linked.

4.6.3 Model Calibration

Calibration pertains to the process of adjusting modelling parameters towards alignment with a reference such as an established standard or observed or experimental data. (Rakha et al 1996; Thacker et al 2004). Calibration has been framed (Tal 2017) as:

the complex activity of constructing...and iteratively testing models of a measurement process.

Accordingly, it is calibration that:

clarifies the source of objectivity of measurement outcomes, the nature of measurement accuracy, and the close relationship between measurement and prediction. (ibid.)

Types of calibration approaches include trial-and-error, explicit methods, and implicit models (Sophocleous, et al 2016). Modelling is iterative, and primarily, the refinements are a form of trial-and-error calibration. An example of an explicit calibration method is

the use of a weight with known measurement and unit to calibrate a scale. In computational terms, this can be achieved using a high-quality dataset, for example, so that the end user is confident that a query which produces zero results means the objects specified in the query do not exist in the database and not because there is a flaw in the data model or query script. However, as graph-based modelling and analysis in conservation is in its infancy, it is premature to assert a standard at this stage as such an assertion would be arbitrary. Calibration against a reference is nevertheless necessary to identify tolerances, that is, thresholds of accuracy that are considered acceptable. As there are currently only a small number of conservation graph models, comparison of graphs against each other will at least demonstrate thresholds based on current practices and can inform calibration practices as graph modelling for conservation continues to develop.

4.6.4 Integrating Verification, Validation and Calibration (VVC)

Modelling practices for complex systems and simulations tends towards integrating verification, validation and calibration (Sankararaman and Mahadevan 2015). An example of the blurring of verification, validation and calibration (VVC) activities can be where a model fails to return a known result during a validation test. The issue was identified as having occurred during the ETL stage (export, transform and load) where data was exported from an existing system, transformed and loaded into the modelling system. Despite the original ETL code being correct and functional with previous successful imports, the failure in this instance with this particular dataset has highlighted additional pre-processing requirements or a revised ETL code was necessary. Subsequently, adjusting for different circumstances and choosing different prepared ETL codes is an example of calibration in model development (see chapter 5 re: CIDOC CRM RDFS).

Each phase of development has specific VVC questions to address, or at least specific encodings for the same or similar lines of questioning. Broadly speaking, however, VVC procedures entail the following core questions:

Verification - did it work? Did the ETL process build the intended graph?

Validation - does the graph match the real world?

Calibration - do I need to tweak it to improve how it works?

4.7 Summary of Method

In summary, the graph modelling and analysis method explored in this study is premised upon a declarative representation using labelled property graphs (LPG) where the resulting model is also the database. Connectivity between conceptual sets in the model provides enrichment through the inclusion of varying levels of micro-to-macro representations. The identification and analysis of graph patterns, including paths, will serve as analogs for studying conservation reasoning. Modelling is an iterative practice and the development of graph models include using graph theoretic measures to profile, compare, and identify benchmarks to inform modelling practice and development. The LPG-based model will also serve as a metamodel in preparation for transformation into RDF-graph representations of conservation data content.

The foundational graph concepts underpinning the Semantic Web approaches to data integration have largely been unacknowledged in recent implementations and knowledge transfer efforts regarding conservation documentation and Linked Data. This thesis, with the aforementioned method, seeks to reintroduce the foundational graph components which provide the necessary conceptual and technological frameworks to address the documentation issues in conservation.

The next three chapters (and Appendices A - H) will present the phased development of a graph model using the aforementioned method and based on data and the specific research interests of The National Archives (UK). Phase 1 consisted of preliminary trials with the creation of small-scale models to identify the most-suitable ETL (extract, transform, load) processes. Due to their limited and specific scope, details for Phase 1 can be found in Appendix H. Nevertheless, the results from Phase 1 identified additional questions and areas where clarification from existing conservation graphs (e.g. the CIDOC CRM and Linked Conservation Data (LCD) RDF graphs) were needed. This included identifying RDF modelling patterns that may be relevant to the LPG case, such as for later export to RDF, and to investigate potential preliminary modelling thresholds that may serve as calibration parameters downstream. The analysis and results from this work constitutes Phase 2 of the research and are presented in chapters 5 and 6, for the CIDOC CRM and Linked Conservation Data project, respectively. Finally, Phase 3 (chapter 7) applied the verification, validation and calibration insights gleaned from the first two phases to create a revised graph model implementation for analysis.

5.0 CIDOC Conceptual Reference Model (CRM)

5.1 Background

As mentioned in chapter 3 (section 3.5.1), the CIDOC CRM is an event-centric conceptual model of historic cultural phenomena (Bruseker et al 2017, Doerr 2003, Doerr et al 2007). It is the intention (as shown in Figure 3.18 and 3.19) that data mapped to the CIDOC CRM will take on a star schema structure, forming clusters, around an event node acting as a hub. When modelled in RDF, each instance of a hub-to-spoke-to-neighbour structure represents a single triple statement of subject-predicate-object, likewise, will other triple-matching statements elsewhere in the wider mapped graph.

Before reviewing and analysing CRM-mapped data (which will follow next in chapter 6), this chapter seeks to review and analyse the RDF Schema (RDFS) serialisation of three versions of the CIDOC CRM, taking into foremost account their inherent graph structure. The aim of this part of the investigation is to ascertain the graph theoretic characteristics of the CIDOC CRM in order to aid familiarisation with the CRM for subsequent application. Implementations, thus far, of the CIDOC CRM, require novice modellers to build fluency and practical knowledge of the CIDOC CRM, firstly, via the extensive documentation available¹, and particularly, via the formal declarations including scope notes of over 80 classes and ca. 150 RDF properties (see Table 5.1.1 for direct links to declarations for each version), and the directly embedded comments within the RDFS file itself. A key document to consult for any user is the *Definition of the CIDOC Conceptual Reference Model*. At the start of this research in 2018, the *Definition...(version 7.0*, Doerr et al 2020) was a 125-page document. However, the latest iteration of the *Definition...(version 7.2.2*, Bekiari et al 2022), due to reformatting, has expanded to a 240-page document. While the CRM model itself has reduced in size by the number of classes in version 7.2.2, nevertheless, as the initial means of familiarisation with the CIDOC CRM, reviewing documentation alone can prove daunting and slow. This can be further complicated by the presence of multiple versions of the CIDOC CRM due to its 30+ years of development. Many of the documentation and related tools, such as the Ontology Management Environment² visualiser, present the CIDOC CRM as a tree representation, however, as the following analysis will show, the classes and properties of the CIDOC CRM itself yields a cyclic graph³. Accessing the CIDOC CRM as a graph as

¹ <https://www.cidoc-crm.org/versions-of-the-cidoc-crm>

² <https://ontome.net/classes-tree>

³ "Cyclic" here refers to the graph theoretic definition and does not refer to cyclic reasoning.

part of initial training and during implementation contributes a more intuitive dimension that aids in the acquisition of the CIDOC CRM as a new schema paradigm for novice modellers.

5.1.1 The Versions

Table 5.1.1 below lists the three versions of the CIDOC CRM reviewed in this study. The official release version 5.0.4 of the CIDOC CRM is the basis for the standard ISO 21127:2014⁴ Information and documentation—A reference ontology for the interchange of cultural heritage. Version 6.2.1 was the latest stable version of the CIDOC CRM at the start of this study. Finally, version 7.1.1 was the latest stable version of the CIDOC CRM at the midpoint of this study (ca. August 2021) and has since been submitted to the International Organization for Standardization (ISO) for recertification. All three versions were analysed using graph theoretic means and the results appear below in section 5.3.

Table 5.1.1. The CIDOC CRM Versions

Version	No. of Classes	No. of Properties (Relationships)	Release Date, Source & Declarations
v.5.0.4	86	138	RDFS: December 2011 https://www.cidoc-crm.org/sites/default/files/cidoc_crm_v5.0.4_official_release.rdfs Declarations: https://cidoc-crm.org/html/cidoc_crm_v5.0.4.html
v.6.2.1	89	149	RDFS: April 2018 http://www.cidoc-crm.org/sites/default/files/cidoc_crm_v6.2.1-2018April.rdfs Declarations: https://cidoc-crm.org/html/cidoc_crm_v6.2.1.html
v.7.1.1	81	160	RDFS: August 13, 2021, https://cidoc-crm.org/rdfs/7.1.1/CIDOC_CRM_v7.1.1.rdfs Declarations: https://cidoc-crm.org/html/cidoc_crm_v7.1.1.html

5.2 ETL: Importing the CIDOC CRM RDFS Models into Labelled Property Graph

⁴ International Organization for Standardization. (2020). *ISO 21127:2014 Information and documentation—A reference ontology for the interchange of cultural heritage information*. ISO. Retrieved April 24, 2023, from <https://www.iso.org/standard/57832.html>

5.2.1 Standard Transformation Procedure (ETL1)

The CIDOC CRM is constructed on a multi-level metamodel basis with meta-classes (“classes of classes”) and meta-properties (“properties of properties”). Akin to Neumayr and Schuetz’ (2017) characterisation of a multi-level metamodel, the classes can be describing attributes rather than instances, which “can allow schema for multiple levels of instantiation (also known as deep characterisation)” and the “multilevel property specialisation allows for the specialisation of properties to more specific properties”. Thus, transformation of CIDOC CRM in RDFS into Neo4j labelled property graphs (LPG) must perpetuate the metamodel semantic constructs.

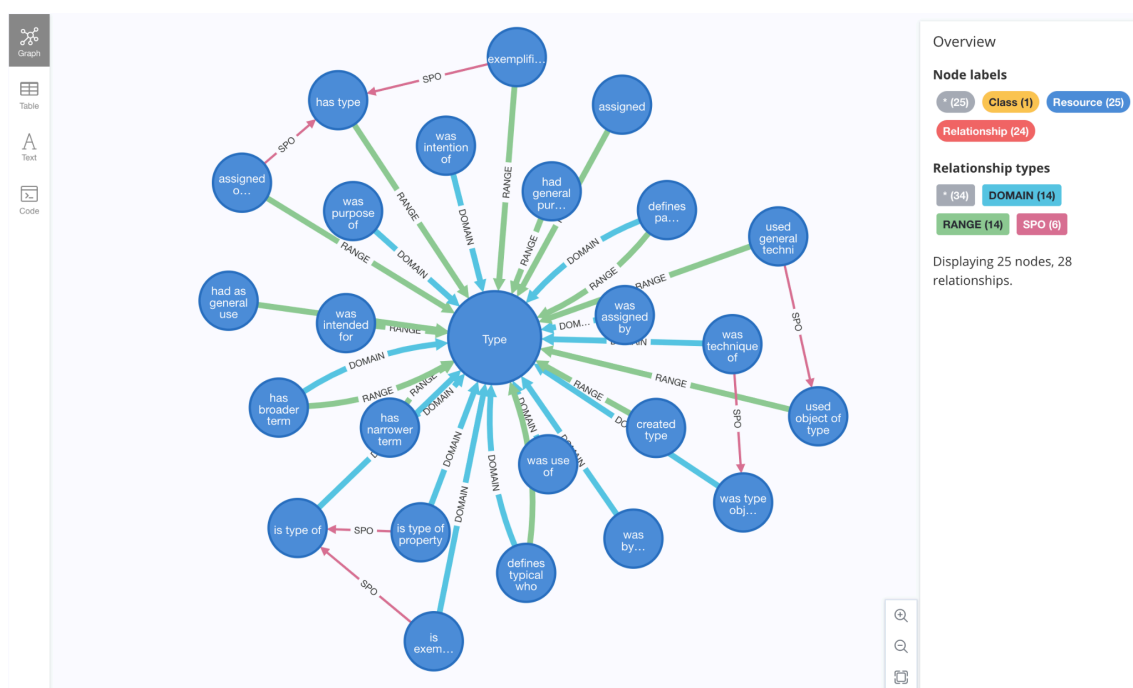


Figure 5.1 Image of CIDOC CRM in LPG after ETL1 transformation procedures demonstrating an example of “nodified” Relationships (RDF properties) for class E55_Type (the larger central node).

The CIDOC CRM entities encoded as `rdfs:Class` were transformed into `(:Class)` nodes while entities encoded as `rdf:Property` were transformed into `(:Relationship)` nodes. This “nodification” of CRM predicates is a significant change to the graph representation structure of the model. However, it is a common and known transformation for converting RDF to property graph (Voegeli 2016). The CIDOC CRM RDF Schema (RDFS) serialisation graph in particular makes transformations of RDF properties into `:Relationship` nodes a necessary step as PG models, like Neo4j’s LPG model, do not support “edges of edges” to represent sub-properties. Therefore, as `(:Relationship)`

nodes, this is reified while retaining the [:SPO], SubPropertyOf, relationship between (:Relationship) nodes (see Figure 5.1).

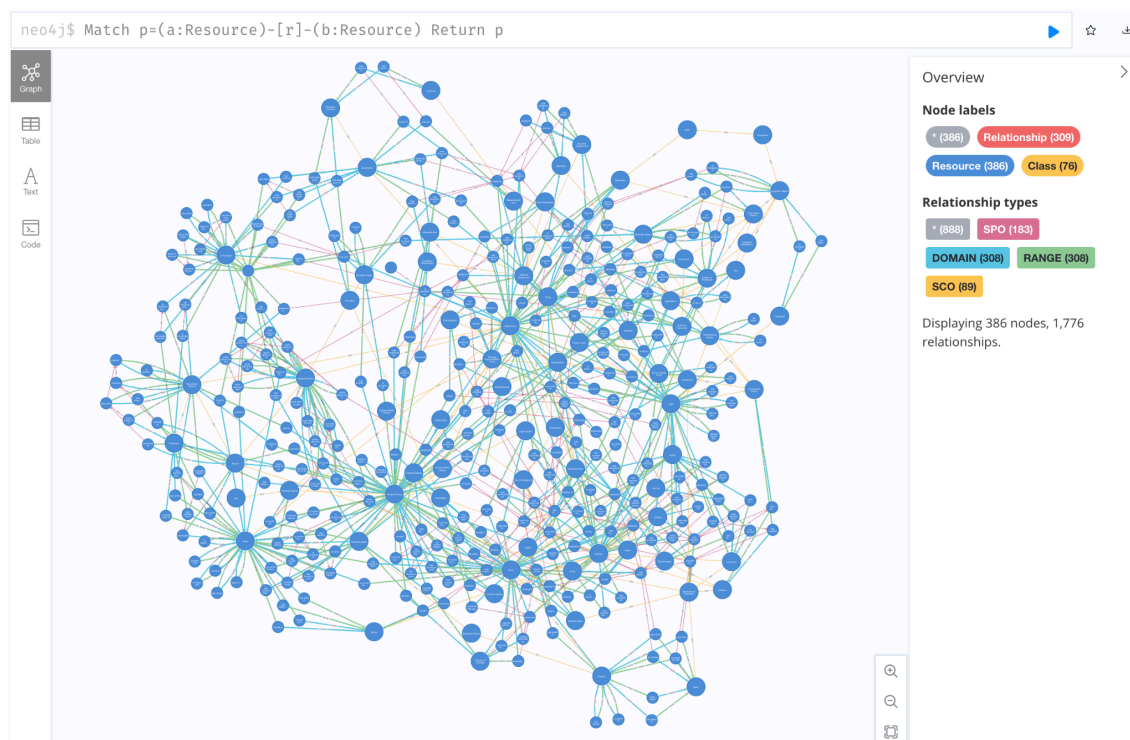


Figure 5.2 Graph visualisation of the CIDOC CRM RDF Schema using ETL1 shows the CIDOC CRM is a highly-connected cyclic graph.

5.2.2 Modified Transformation Procedure (ETL2)

Following the initial import (ETL1) of the RDF Schema (RDFS) serialisation of CRM v6.2.1, it was discovered (see Appendix H for Phase 1 Models, on directionality in modelling) that the Neosemantics plugin transformation of RDF schema relationships, i.e. Domain and Range, results in only outgoing relationships from CIDOC CRM properties, now (:Relationship) nodes (Figure 5.3). This is due to the interpretation of the RDFS encoding which is UNDIRECTED and encoded using an equivalence (=) (see RDF/XML code in Figure 5.4). However, such equivalence statements are transformed using an outgoing edge. In the directed graph environment of Neo4j, Neosemantics interprets the encoding by pointing towards the `rdf:resource` node, thereby resulting in the LPG modelling having only outgoing edges from the RDF properties that are now “modified” (:Relationship) nodes.

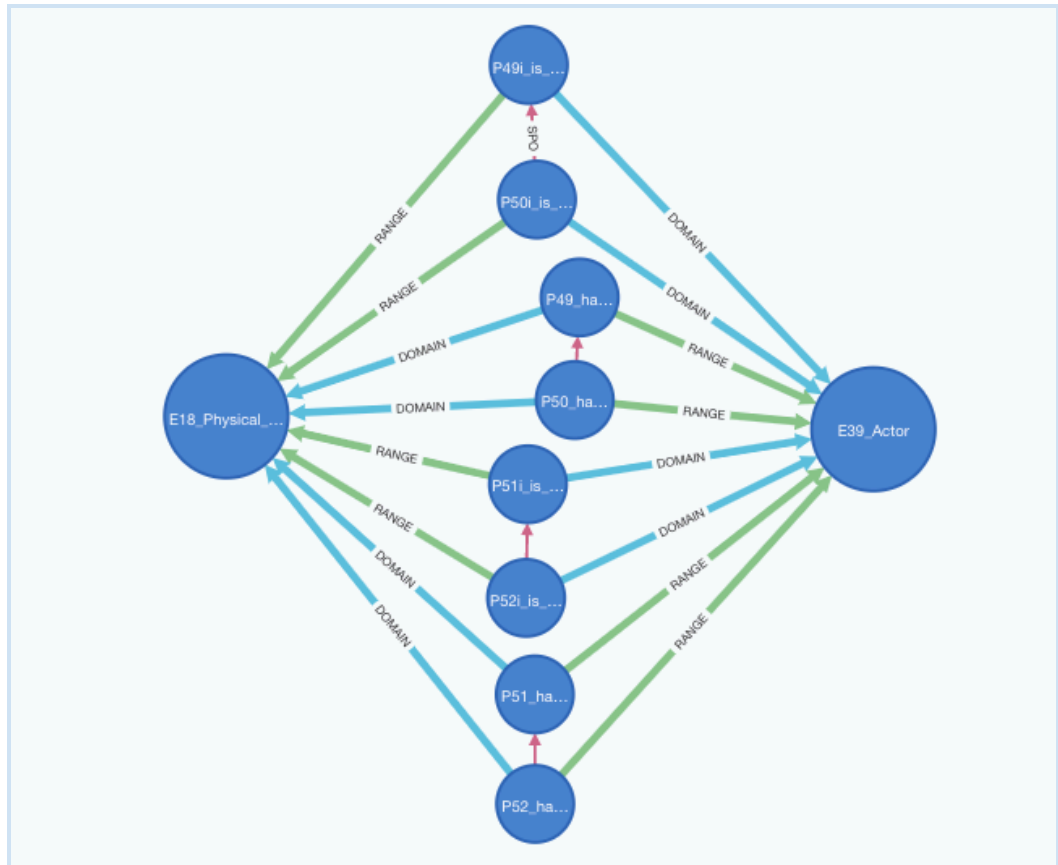


Figure 5.3 Two classes from ETL1 model of CIDOC CRM v.6.2.1 showing the results of transforming RDFS into Neo4j's LPG model using Neosemantics. Note that the RDF property, i.e. (:Relationship), nodes along the center have only outgoing [:DOMAIN] and [:RANGE] edges that point to their respective classes due to how RDFS declares domains and ranges. The ETL1 model therefore will not support traversal paths through or end on a (:Relationship) node, rendering this model incompatible for directed path-based queries.

To re-enable full traversal of the CIDOC CRM graph as demonstrated in Phase 1 trials (see Appendix H), a reverse relationship, [:xDOMAIN], was created to reestablish the directed path of the triple from subject to predicate to object, $s \rightarrow p \rightarrow o$. A reverse relationship, [:xSCO], has also been created for each [:SCO], subClassOf, relationship. Semantically, the [:xSCO] reverse relationship functions as a 'superClassOf' relationship. The 'x' prefix was used in order to distinguish between encoded relationships and manually added reverse relationships during the transformation process. This transformation step is only necessary when working with the CRM itself as RDFS (.rdfs extension) and is not necessary with CRM-mapped RDF data (.rdf, .nt, .trig., .ttl formats). To distinguish between the original encoding and when this manual transformation step has been added, the former is designated as ETL1 and the latter designated as ETL2. In order to use the CRM RDFS itself as a subgraph, to enable queryable directed paths and traversals that can reach all nodes, ETL2 must be used. This enables querying with the semantic direction and syntactic direction in alignment. Again, this is not the case with RDF data that has already been mapped to the CRM (see next chapter which reviews and analyses CRM-mapped data from the Linked Conservation Data project).

```

<rdf:Property rdf:about="P49_has_former_or_current_keeper">
  <rdfs:domain rdf:resource="E18_Physical_Thing"/>
  <rdfs:range rdf:resource="E39_Actor"/>
</rdf:Property>

<rdf:Property rdf:about="P49i_is_former_or_current_keeper_of">
  <rdfs:domain rdf:resource="E39_Actor"/>
  <rdfs:range rdf:resource="E18_Physical_Thing"/>
</rdf:Property>

```

Figure 5.4. Excerpt from the CIDOC CRM v.6.2.1 RDFS presenting how 'domain' and 'range' are encoded. Other associated encodings in nearby lines, such as `rdfs:label` and `rdfs:comment`, have been left out for clarity.

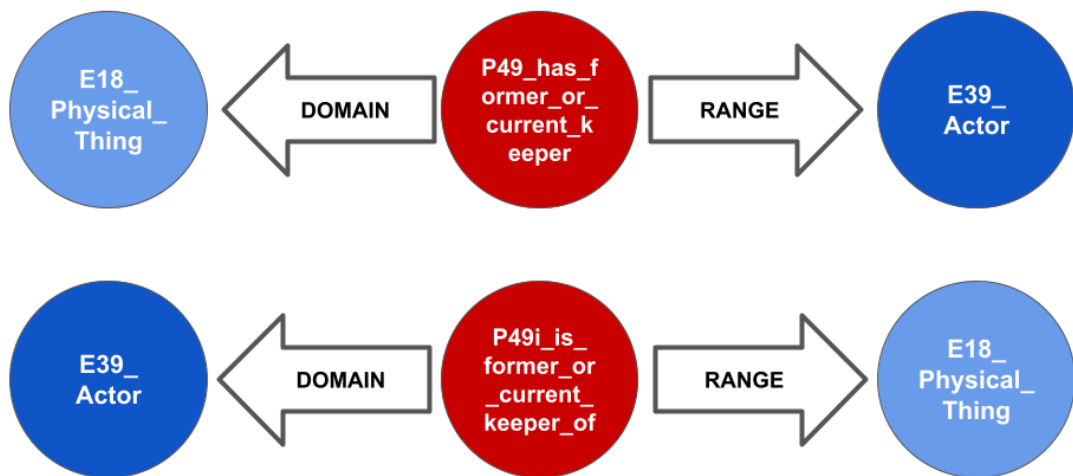


Figure 5.5 The standard transformation procedure (ETL1) of the RDFS for CIDOC CRM showing the Neosemantics interpretation of the RDFS encoding results in only outgoing edges for (:Relationship), i.e. RDF property, nodes.

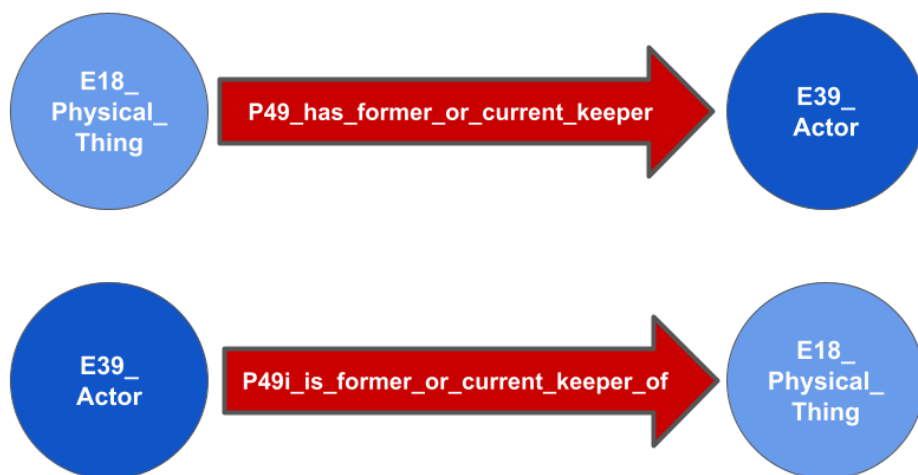


Figure 5.6 Example of the declared semantic triple $E18 \rightarrow P49 \rightarrow E39$ and its reciprocal statement $E39 \rightarrow P49i \rightarrow E18$.

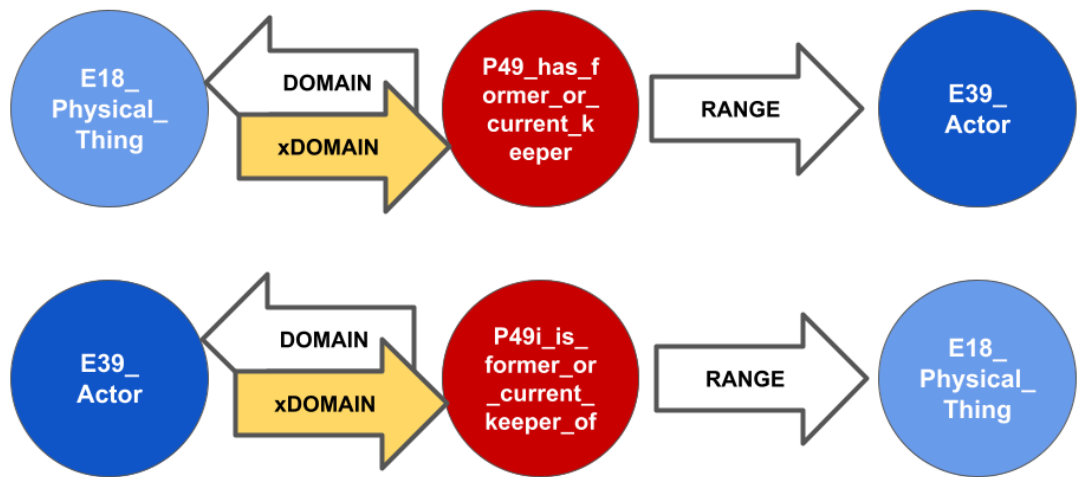


Figure 5.7 The ETL2 modified transformation procedure adds a reciprocal xDOMAIN and xSCO relationship to reassert the semantic intention as in Figure 5.6. Only the xDOMAIN addition is represented in this diagram.

The following graph theoretic analysis was applied to all three versions of the CIDOC CRM imported using ETL1. Only the most recent version (at the time of the study), v.7.1.1, had its ETL2 transformation model analysed. The results of which are also presented in the next section.

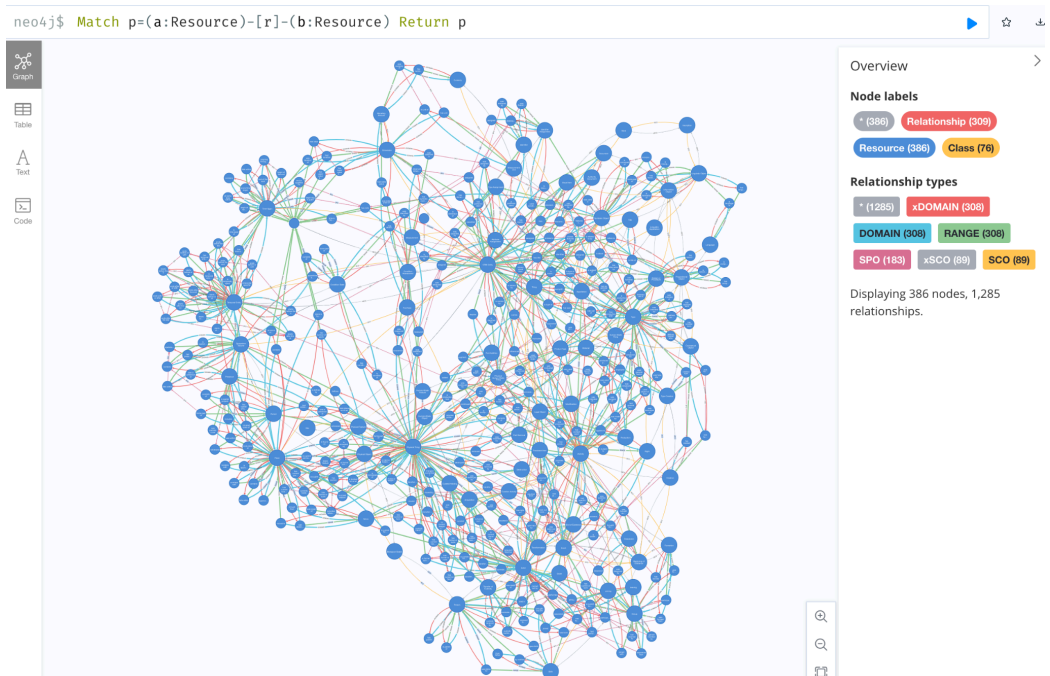


Figure 5.8 Graph visualisation of the CIDOC CRM RDF Schema using ETL2 shows the CIDOC CRM remains a highly-connected cyclic graph.

5.3 Results of Graph Theoretic Analysis

Please note, the “node id” property referred to in these results is not a persistent identifier and is unique to the local databases created for this analysis. This is why the same class or CRM property when reference across the different analysed CRM versions have different node id numbers.

Descriptions and uses of each graph theoretic measure can be found in section 4.4. The results here are arranged in the same order as how the measures are described in section 4.4. For the full results, please refer to <https://github.com/ana-tam/conservation-graphs>.

5.3.1 Order and Size

Table 5.3.1. Order and Size Results for CIDOC CRM Group

	ETL1			ETL2
Version	v.5.0.4	v.6.2.1	v.7.1.1 (ETL1)	v.7.1.1 (ETL2)
Order (node ct)*	346	374	387	387
Size (edge ct)	762	830	888	1285
Node:Edge Ratio	1:2.20	1:2.22	1:2.29	1:3.32
Node:Edge (as quotient)	0.45	0.45	0.44	0.30

The order and size of the three formal releases show a progression towards more nodes and edges with each new edition. This is as expected as the evolution of the CIDOC CRM, from version to version, has principally been to specify further classes or additional RDF properties, both of which are represented as nodes in these models of the RDFS serialisations. Therefore, any increase in classes or properties will result in an increase in overall node counts due to the “nodification”. The size, or edge count, does not increase from ETL1 to ETL2 as a perfect doubling as not all edges are reciprocal, e.g. [:RANGE] and [:SPO] relationships do not have manually created reciprocal edges.

5.3.2 Density/Sparsity

The single isolated node (i.e. where measured degree = 0) in each model (Table 5.3.2, row 4, including headers) refers to the (:_GraphConfig) node which stores the

Neosemantics configuration parameters. While it was included in the accounting against density/sparsity, its singular presence has negligible influence on the results.

The number of leaf nodes (where degree = 1) are reduced with v.7.1.1, however, there are also fewer declared classes in the later version with 81 classes, whereas v.6.2.1 had 89 classes. Follow-up investigations show that many of the leaf nodes in v.6.2.1 and v.5.0.4 were removed (deprecated) as classes in v.7.1.1.

Table 5.3.2. Density/Sparsity Results for CIDOC CRM Group

Version	v.5.0.4	v.6.2.1	v.7.1.1 (ETL1)	v.7.1.1 (ETL2)
Edge Density*	0.0064	0.0059	0.0059	0.0086
Leaf Nodes	7	7	2	1
Isolated Nodes*	1	1	1	1
Leaf + Isolated*	8	8	3	2
Theta Ratio*, θ	0.0231	0.0214	0.0078	0.0052
Average Clustering Coefficient	0.119	Infinity	infinity	infinity

**Metrics marked with an asterisk include the (:_GraphConfig) node in the accounting.*

In versions 5.0.4 and 6.2.1, these leaf nodes were as follows, most of which were deprecated as classes by v.7.1.1:

- E27_Site (continues to exist in v.7.1.1)
- E38_Image (no longer exists as a class in v.7.1.1)
- E40_Legal_Body (no longer exists as a class in v.7.1.1)
- E47_Spatial_Coordinates (no longer exists as a class in v.7.1.1)
- E48_Place_Name (no longer exists as a class in v.7.1.1)
- E50_Date (no longer exists as a class in v.7.1.1)
- E84_Information_Carrier (no longer exists as a class in v.7.1.1)

A search for these deprecated classes in the official Issues tracker⁵ reveals several were deprecated as it was deemed other existing classes were sufficient, albeit, sometimes properties of the deprecated class were migrated over to the other sufficient, existing class. For example, "E84_Information_Carrier" was deprecated and its property "P128_carries (is_carried_by)" was migrated over to "E18_Physical_Thing". However, it

⁵ https://www.cidoc-crm.org/issue_summary2

was not evident through review of the Issues whether leaf-node detection was a factor in identifying these derivative class declarations.

From a graph connectivity perspective, the new declarations in v.7.1.1 result in a more connected, cyclic graph and evolves away from a less-connected graph with leaf nodes, i.e. branched endpoints, as is evident in the two earlier versions of the CIDOC CRM reviewed here. In fact, relying only on hierarchical tree-like representations of a cyclic network can obscure leaf node elements of low connectivity that can or should be pruned.

Table 5.3.3. Local Clustering Coefficient Results for CIDOC CRM Group

Local Clustering Coefficient - CIDOC CRM Group (highest scores)				
	<i>node id</i>	<i>n.label</i>	<i>node Label/CRM Entity</i>	<i>degrees</i>
v.5.0.4	252	["contains"]	["Resource", "Relationship"]	Infinity
v.6.2.1	32	"had specific purpose"	["Resource", "Relationship"]	1
v.7.1.1 (ETL1)	119	["ends after or with the start of"]	["Resource", "Relationship"]	1
v.7.1.1 (ETL2)	235	<i>null</i> (rdf-schema#label)	["Resource", "Relationship"]	Infinity

The local clustering coefficient (LCC) measures connectivity on a node-by-node basis. Table 5.3.3 only presents the top local clustering coefficient result from each graph while Figure 5.9 shows a colour-coded image of the top twenty results from each CRM graph model analysed. In this figure, the two “infinity⁶” top results for v5.0.4 and v7.1.1 (ETL2) are marked in green while results with a score of “1” are marked in yellow. The remaining scores in the top 20 results have a score of “0.6667” (rounded to the nearest ten-thousandths).

It is important to bear in mind that these results provide insight into the CIDOC CRM graph itself as a closed system, largely in terms of the Domain and Range relationships between entities and without any influence from data and relationships between data entities.

⁶ Nodes with “infinity” scores have very low-connectivity. Further explanation for “infinity” results are given in chapter 6, section 6.3.2.

v.5.0.4				v.6.2.1				v.7.1.1 (ETL1)				v.7.1.1 (ETL2)			
id	name	CRMEntity	lcc	id	name	CRMEntity	lcc	id	name	CRMEntity	lcc	id	name	CRMEntity	lcc
252	["contains"]	["Resource", "Relationship"]	infinity	32	"had specific purpose"	["Resource", "Relationship"]	1	119	["ends after or with the start of"]	["Resource", "Relationship"]	1	235	null (rdf-schema#label)	["Resource", "Relationship"]	infinity
191	["continued"]	["Resource", "Relationship"]	1	75	"had presence"	["Resource", "Relationship"]	1	174	["forms part of"]	["Resource", "Relationship"]	1	174	["forms part of"]	["Resource", "Relationship"]	1
216	["is current or former member of"]	["Resource", "Relationship"]	1	89	"was continued by"	["Resource", "Relationship"]	1	206	"had specific purpose"	["Resource", "Relationship"]	1	206	"had specific purpose"	["Resource", "Relationship"]	1
224	["was purpose of"]	["Resource", "Relationship"]	1	116	"is composed of"	["Resource", "Relationship"]	1	226	["starts before or with the end of"]	["Resource", "Relationship"]	1	226	["starts before or with the end of"]	["Resource", "Relationship"]	1
78	"was continued by"	["Resource", "Relationship"]	1	208	"continued"	["Resource", "Relationship"]	1	245	["ends before the start of"]	["Resource", "Relationship"]	1	245	["ends before the start of"]	["Resource", "Relationship"]	1
329	["has current or former member of"]	["Resource", "Relationship"]	1	233	"is current or former member of"	["Resource", "Relationship"]	1	253	["is composed of"]	["Resource", "Relationship"]	1	253	["is composed of"]	["Resource", "Relationship"]	1
267	["was type of object"]	["Resource", "Relationship"]	0.6666666666666666	241	"was purpose of"	["Resource", "Relationship"]	1	285	["starts after the end of"]	["Resource", "Relationship"]	1	119	["ends after or with the start of"]	["Resource", "Relationship"]	1
284	["is exemplified by"]	["Resource", "Relationship"]	0.6666666666666666	248	"was a presence of"	["Resource", "Relationship"]	1	312	["is current or former member of"]	["Resource", "Relationship"]	1	285	["starts after the end of"]	["Resource", "Relationship"]	1
298	["is former or current member of"]	["Resource", "Relationship"]	0.6666666666666666	348	"forms part of"	["Resource", "Relationship"]	1	319	["was purpose of"]	["Resource", "Relationship"]	1	312	["is current or former member of"]	["Resource", "Relationship"]	1
327	["is subject of"]	["Resource", "Relationship"]	0.6666666666666666	355	"has current or former member of"	["Resource", "Relationship"]	1	379	["was purpose of"]	["Resource", "Relationship"]	1	319	["was purpose of"]	["Resource", "Relationship"]	1
335	["has former or current member of"]	["Resource", "Relationship"]	0.6666666666666666	287	"supported type criterion"	["Resource", "Relationship"]	0.6666666666666666	256	["ends before the end of"]	["Resource", "Relationship"]	0.6666666666666666	370	["has time-span"]	["Resource", "Relationship"]	1
143	["was motivated by"]	["Resource", "Relationship"]	0.6666666666666666	288	"was type of object"	["Resource", "Relationship"]	0.6666666666666666	269	["is occupied by"]	["Resource", "Relationship"]	0.6666666666666666	372	["from father"]	["Resource", "Relationship"]	1
155	["has type"]	["Resource", "Relationship"]	0.6666666666666666	305	"is exemplified by"	["Resource", "Relationship"]	0.6666666666666666	271	["was motivated by"]	["Resource", "Relationship"]	0.6666666666666666	379	["has current or former member of"]	["Resource", "Relationship"]	1
163	["used general technique"]	["Resource", "Relationship"]	0.6666666666666666	319	"is former or current member of"	["Resource", "Relationship"]	0.6666666666666666	276	["has value"]	["Resource", "Relationship"]	0.6666666666666666	256	["ends before the end of"]	["Resource", "Relationship"]	0.6666666666666666
173	["is current keeper of"]	["Resource", "Relationship"]	0.6666666666666666	353	"is subject of"	["Resource", "Relationship"]	0.6666666666666666	282	["used general technique"]	["Resource", "Relationship"]	0.6666666666666666	289	["is occupied by"]	["Resource", "Relationship"]	0.6666666666666666
205	["is about"]	["Resource", "Relationship"]	0.6666666666666666	362	"has former or current member of"	["Resource", "Relationship"]	0.6666666666666666	289	["is current keeper of"]	["Resource", "Relationship"]	0.6666666666666666	271	["was motivated by"]	["Resource", "Relationship"]	0.6666666666666666
247	["supported type criterion"]	["Resource", "Relationship"]	0.6666666666666666	1	"exemplifies"	["Resource", "Relationship"]	0.6666666666666666	291	["starts before or with the end of"]	["Resource", "Relationship"]	0.6666666666666666	276	["has value"]	["Resource", "Relationship"]	0.6666666666666666
1	["exemplifies"]	["Resource", "Relationship"]	0.6666666666666666	7	"has current keeper of"	["Resource", "Relationship"]	0.6666666666666666	295	["spatiotemporally of"]	["Resource", "Relationship"]	0.6666666666666666	282	["used general technique"]	["Resource", "Relationship"]	0.6666666666666666
6	"has current keeper of"	["Resource", "Relationship"]	0.6666666666666666	8	"used object of type"	["Resource", "Relationship"]	0.6666666666666666	321	["was a presence of"]	["Resource", "Relationship"]	0.6666666666666666	289	["is current keeper of"]	["Resource", "Relationship"]	0.6666666666666666

Figure 5.9. The top 20 results from each CRM version are presented side by side. Nodes with an LCC score of "1" are highlighted in yellow. Nodes with a score of "infinity" are highlighted in green.

Table 5.3.4. Degree Centrality Results for CIDOC CRM Group

Degree Centrality - CIDOC CRM Group (highest degrees)					
		<i>node id</i>	<i>n.label</i>	<i>node Label/CRM Entity</i>	<i>degrees</i>
v.5.0.4	Natural	142	["was used for"]	["Resource", "Relationship"]	4
	Reverse	340	["CRM Entity"]	["Resource", "Class"]	40
	Undirected	342	["Actor"]	["Resource", "Class"]	41
v.6.2.1	Natural	156	"was used for"	["Resource", "Relationship"]	4
	Reverse	253	"Physical Thing"	["Resource", "Class"]	45
	Undirected	253	"Physical Thing"	["Resource", "Class"]	47
v.7.1.1 (ETL1)	Natural	79	["was used for"]	["Resource", "Relationship"]	4
	Reverse	323	["Physical Thing"]	["Resource", "Class"]	59
	Undirected	323	["Physical Thing"]	["Resource", "Class"]	60
v.7.1.1 (ETL2)	Natural	323	["Physical Thing"]	["Resource", "Class"]	32
	Reverse	323	["Physical Thing"]	["Resource", "Class"]	60
	Undirected	323	["Physical Thing"]	["Resource", "Class"]	92

The results of the degree centrality analysis across the four models emphasise how the ETL2 procedure reestablishes traversability across the graph model. In all three versions where the ETL1 procedure was utilised, the nodes with the highest number of natural or outgoing edges was the "P16_was_used_for" property. However, when the analysis considered reverse or incoming directions, there were substantially more edges (in fact by a factor of 10!) for all ETL1 processed graphs. This imbalance strongly confirms what

has been presented in section 5.2 above, that the model is not semantically representative as a directed graph without reciprocating edges. The final series of analyses on v.7.1.1 which had been processed using ETL2 procedures shows a more representative distribution of outgoing versus incoming edges, roughly one outgoing edge for every two incoming edges, which is in better alignment with the CRM's documentation. For example, the fact that "E18_Physical_Thing" has the most incoming and outgoing connections, and therefore the greatest number of immediate neighbours aligns with expectations and is visualised in Figure x below.

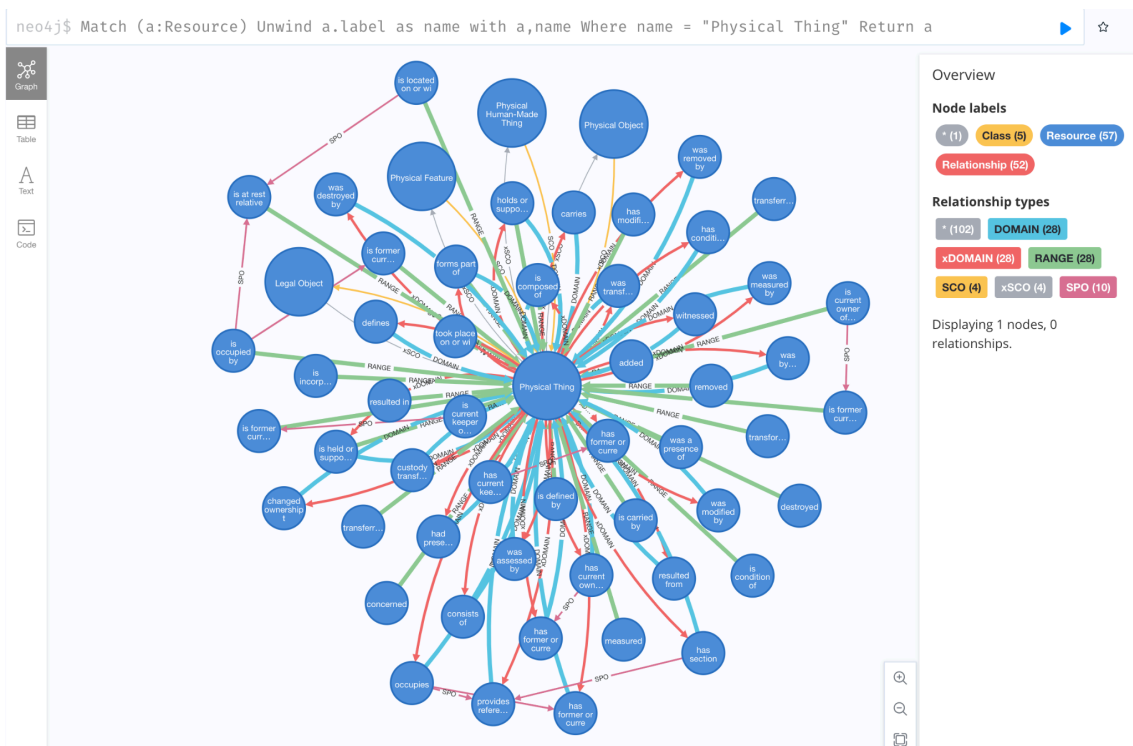


Figure 5.10 Visualisation of E18_Physical_Thing as a hub node surrounded by its properties (:Relationship), depicted here as the smaller nodes, and its directly related (:Class) nodes from v.7.1.1(ETL2).

5.3.3 Global Triangle Count

Table 5.3.5. Global Triangle Count Results for CIDOC CRM Group

Version	v.5.0.4	v.6.2.1	v.7.1.1	v.7.1.1 (ETL2)
Global Triangle Count	92	108	133	136

Iterations of the CIDOC CRM thus far, have shown a tendency towards increased connectivity. The results of the global triangle count further demonstrate a gradual

increase in triangle counts from the earliest version (5.0.4) to the later versions. Interestingly, despite adding two reciprocal edge types to v.7.1.1 (ETL2), this only increased the global triangle count slightly by 3.

5.3.4 Diameter

Despite the differences between the three versions of the CIDOC CRM, each version's graph has a diameter of 8 when measured as undirected edges (see Table 5.3.4). That is, one can start at any node on the graph and traverse to any other node in the graph within 8 hops. This traversal length is retained even when applying the modified transformation procedure (ETL2) to version 7.1.1 which adds two edge types, [:xDOMAIN] and [:xSCO] to serve as reciprocal directions for [:DOMAIN] and [:SCO] edges.

Table 5.3.6. Diameter Results for CIDOC CRM Group

Version	v.5.0.4	v.6.2.1	v.7.1.1 (ETL1)	v.7.1.1 (ETL2)
Diameter (undirected)	8	8	8	8
Diameter (directed, outgoing)	7	7	7	9
Diameter (directed, incoming)	6	6	7	9

However, a clear distinction can be made between the graphs which had the standard transformation (v. 5.0.4, v.6.2.1, and v.7.1.1 ETL1) and the graph that had the modified transformation (v.7.1.1 ETL2) when assessed using directed edges. Comparatively, both the diameters from outgoing and incoming orientations were shorter than the diameter resulting from an undirected assessment, save for the v.7.1.1 ETL2 graph where the diameter was lengthened in both directions.

There is also a distinction between the v.5.0.4 and v.6.2.1 graphs from the v.7.1.1 graph. The former two have the same diameter length given each edge permutation studied (7 for outgoing paths and 6 for incoming paths) while the latter, for both ETL1 and ETL2, have another pattern where both incoming and outgoing directed diameters have the same lengths (7 for ETL1 and 9 for ETL2). Further distinction patterns between the first two, v5.0.4 and v.6.2.1, from the latter v.7.1.1 can also be found in the next section on planarity.

5.3.5 Planarity and $K_{3,3}$ Bipartite Graph

Table 5.3.7 $K_{3,3}$ Bipartite Graph Results for CIDOC CRM Group

Version	v.5.0.4	v.6.2.1	v.7.1.1 (ETL1)	v.7.1.1 (ETL2)
$k_{3,3}$ Count	0	0	768	823,104

There is a significant change in the CIDOC CRM's graph from version 6.2.1 to version 7.1.1 where the graph topology goes from being planar to nonplanar, that is, with zero $k_{3,3}$ bipartite graphs in the former to having 768 (using the standard procedure, ETL1) $k_{3,3}$ bipartite graphs in the latter. The $k_{3,3}$ count jumps several orders of magnitude to 823,104 when assessing version 7.1.1 using the modified transformation procedure (ETL2). That is, the adding of the reciprocal [:xDOMAIN] and [:xSCO] edges increased the number of $k_{3,3}$ bipartite structures to the graph. As the semantic reasons for this has been established, there is a strong indication that $k_{3,3}$ bipartite structures are inherent in existing heritage data structures. For example, this can be reasoned by imagining a case with two datasets, one is a list of objects, while the other is a list of collection activities such as accession or registration, condition assessment, scientific analysis, treatment and display. We can easily imagine each object in group 1 (the list of objects) underwent each activity in group 2 (the list of collection activities) creating edges between the two groups. Meanwhile, the objects in group 1 are not recorded as having interacted with each other and likewise activities in group 2 are perceived as siloed and have been recorded as such and therefore there are no explicit edges connecting items within their own groups, resulting in a bipartite graph representation. Therefore, $k_{3,3}$ bipartite graph detection bears further consideration and exploration within further datasets (as will be the case in the next two chapters.)

Furthermore, this reinforces the pattern that v.5.0.4 and v.6.2.1 are more similar to each other than either are to v.7.1.1. This transition from v6.2.1 to v7.1.1 marks a significant moment in the CIDOC CRM's development as it is the domain representation's transition from a planar model to a non-planar model with higher dimensionality.




























5.3.6 Undirected Motifs Frequency

All motif permutations for $k = 3, 4,$ and 5 were assessed in this study for completeness. Table 5.3.8 presents the results in the motif order as devised by Abuoda et al 2020.

However, upon further review, taking into account the node-to-edge ratios of each motif, Table 5.3.9 presents the same results in node-to-edge ratio order. By using the ratio-based ordering, it is clear that the number of edges in the motif is about twice the number of nodes, the instances of these structures across all four versions of the CIDOC CRM graphs decline to zero.

Figures 5.10 and 5.11 present the results from each table using a logarithmic bar graph which shows that the occurrence of the motifs follow a logarithmic pattern. Figure 5.11 which depicts the ratio-based ordering of the motif identifiers also reveals distinct motif regions which coincide with the node-to-edge ratios of 0.80, 1, 1.20, between 1.25 - 1:40, and between 1.50 - 2. Also, within each bracketed region, the frequency of motif occurrence follows a gentle bell curve pattern.

Table 5.3.8. Undirected Motifs Frequency Results for the CIDOC CRM Group

Motif		v.5.0.4	v.6.2.1	v.7.1.1 (ETL1)	v.7.1.1 (ETL2)
	m3.1	14,126	16,322	18,942	44,208
	m3.2	576	672	1,146	2,784
	m4.1	66,646	78,904	89,782	306,164
	m4.2	320,094	408,150	552,336	2,067,360
	m4.3	6,804	7,824	13,530	49,820
	m4.4	3,448	4,040	5,136	20,080
	m4.5	348	384	1,164	4,436
	m4.6	0	0	0	0
	m5.1	432	432	1,188	7,284
	m5.2	2,128	2,632	4,912	27,468
	m5.3	5,406	5,626	16,962	98,740
	m5.4	88	96	860	4,800
	m5.5	88	96	860	4,800
	m5.6	0	0	0	0
	m5.7	204	256	516	3,968
	m5.8	0	0	0	0
	m5.9	31,488	37,404	55,132	306,206
	m5.10	32,096	35,272	49,318	218,468
	m5.11	4,912	5,160	18,448	98,728
	m5.12	191,568	225,256	404,652	2,377,980
	m5.13	1,256	1,496	2,586	14,434
	m5.14	0	0	0	0
	m5.15	33,122	42,962	56,532	353,216
	m5.16	2,664	3,588	4,980	37,260
	m5.17	368,372	459,154	537,624	2,651,202
	m5.18	9,657,192	13,572,336	22,223,784	127,708,896
	m5.19	1,118,206	1,441,620	1,743,384	9,791,278
	m5.20	7,740	9,230	10,720	59,420
	m5.21	0	0	0	0

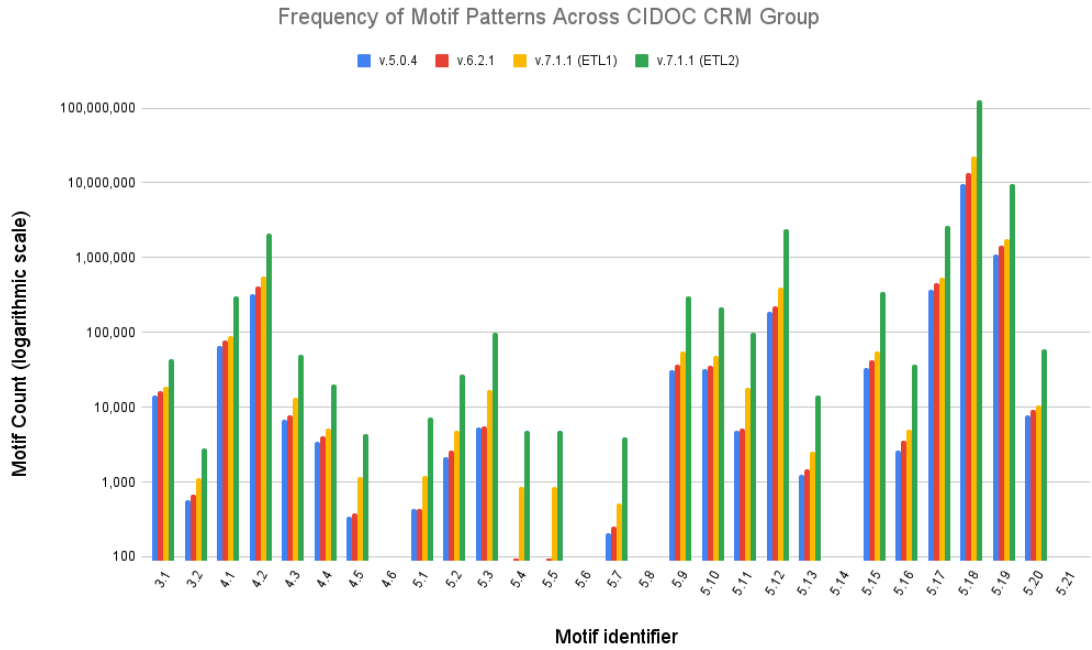


Figure 5.11 Bar graph of results from table 5.3.8 in motif identifier order after Abuoda et al 2020.

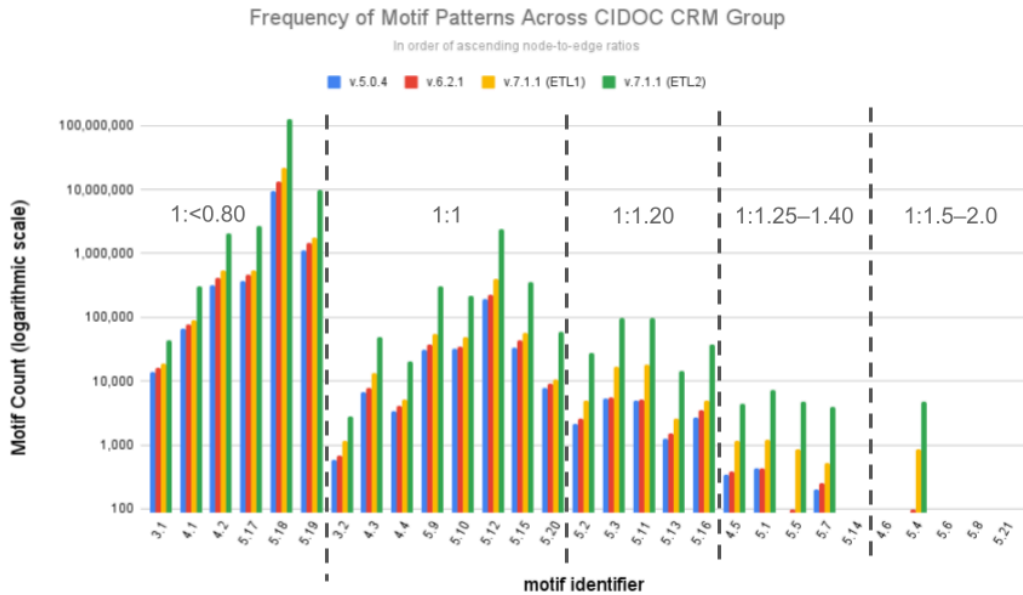


Figure 5.12. Bar graph of results from Table 5.3.9 which re-orders the results in line with ascending node-to-edge ratios of each $k=3,4,5$ motif.

Table 5.3.9 CIDOC CRM Group Motif Node:Edge Ratios*

node count, k=	edge count	Ratio	motif identifier	v.5.0.4	v.6.2.1	v.7.1.1 (ETL1)	v.7.1.1 (ETL2)
3	2	1:0.66	3.1	14,126	16,322	18,942	44,208
4	3	1:0.75	4.1	66,646	78,904	89,782	306,164
4	3	1:0.75	4.2	320,094	408,150	552,336	2,067,360
5	4	1:0.80	5.17	368,372	459,154	537,624	2,651,202
5	4	1:0.80	5.18	9,657,192	13,572,336	22,223,784	127,708,896
5	4	1:0.80	5.19	1,118,206	1,441,620	1,743,384	9,791,278
3	3	1:1	3.2	576	672	1,146	2,784
4	4	1:1	4.3	6,804	7,824	13,530	49,820
4	4	1:1	4.4	3,448	4,040	5,136	20,080
5	5	1:1	5.9	31,488	37,404	55,132	306,206
5	5	1:1	5.10	32,096	35,272	49,318	218,468
5	5	1:1	5.12	191,568	225,256	404,652	2,377,980
5	5	1:1	5.15	33,122	42,962	56,532	353,216
5	5	1:1	5.20	7,740	9,230	10,720	59,420
5	6	1:1.20	5.2	2,128	2,632	4,912	27,468
5	6	1:1.20	5.3	5,406	5,626	16,962	98,740
5	6	1:1.20	5.11	4,912	5,160	18,448	98,728
5	6	1:1.20	5.13	1,256	1,496	2,586	14,434
5	6	1:1.20	5.16	2,664	3,588	4,980	37,260
4	5	1:1.25	4.5	348	384	1,164	4,436
5	7	1:1.40	5.1	432	432	1,188	7,284
5	7	1:1.40	5.5	88	96	860	4,800
5	7	1:1.40	5.7	204	256	516	3,968
5	7	1:1.40	5.14	0	0	0	0
4	6	1:1.50	4.6	0	0	0	0
5	8	1:1.60	5.4	88	96	860	4,800
5	8	1:1.60	5.6	0	0	0	0
5	9	1:1.80	5.8	0	0	0	0
5	10	1:2	5.21	0	0	0	0

5.3.7 Eigenvector Centrality

Table 5.3.10. Eigenvector Centrality Results for CIDOC CRM Group

Eigenvector Centrality - CIDOC CRM Group (highest scores)					
	<i>projection</i>	<i>node id</i>	<i>n.label</i>	<i>node Label/CRM Entity</i>	<i>score</i>
v.5.0.4	default	340	["CRM Entity"]	["Resource", "Class"]	0.99387
	undirected	340	["CRM Entity"]	["Resource", "Class"]	0.32762
v.6.2.1	default	368	"CRM Entity"	["Resource", "Class"]	0.99487
	undirected	253	"Physical Thing"	["Resource", "Class"]	0.34582
v7.1.1 (ETL1)	default	384	["CRM Entity"]	["Resource", "Class"]	0.99501
	undirected	76	["Temporal Entity"]	["Resource", "Class"]	0.43248
v7.1.1 (ETL2)	default	76	["Temporal Entity"]	["Resource", "Class"]	0.42923
	undirected	323	["Physical Thing"]	["Resource", "Class"]	0.41475

The eigenvector centrality results proved useful during the Phase 1 trials (see Appendix H) in identifying the directionality problem mentioned above in section 5.2 which led to the development of the ETL2 procedures. Therefore, the eigenvector centrality measure has demonstrated an applicability for use in checking and validating a graph model against expected transitivity characteristics. Unlike the Phase 1 trials, the above results reflect an eigenvector centrality analysis of only the four versions of the CIDOC CRM graph models by themselves without any additional connections to a data graph.

The Neo4j Graph Data Science Library's eigenvector centrality algorithm calculates "the centrality score for each node...[by deriving from]...the scores of its incoming neighbors."⁷ The top scoring node, and therefore the most transitively influential node in all three ETL1 produced graphs for v.5.0.4, v6.2.1, and v7.1.1 (ETL1) is "E1_CRM_Entity". This is as expected as the CRM has been designed with E1_CRM_Entity as a top super class and all other classes are subclasses of it. However, once ETL2 is applied and we have more reverse directions and directed access to (:Relationship) nodes, this position is assumed by "E2_Temporal_Entity" and "E18_Physical_Thing", both of which have many more incoming edges from RDF properties, i.e. (:Relationship) nodes.

⁷ Neo4j. (n.d.). *Eigenvector Centrality*. Neo4j Graph Data Science Library Manual v2.3. Retrieved June 7, 2023, from <https://neo4j.com/docs/graph-data-science/2.3/algorithms/eigenvector-centrality/>

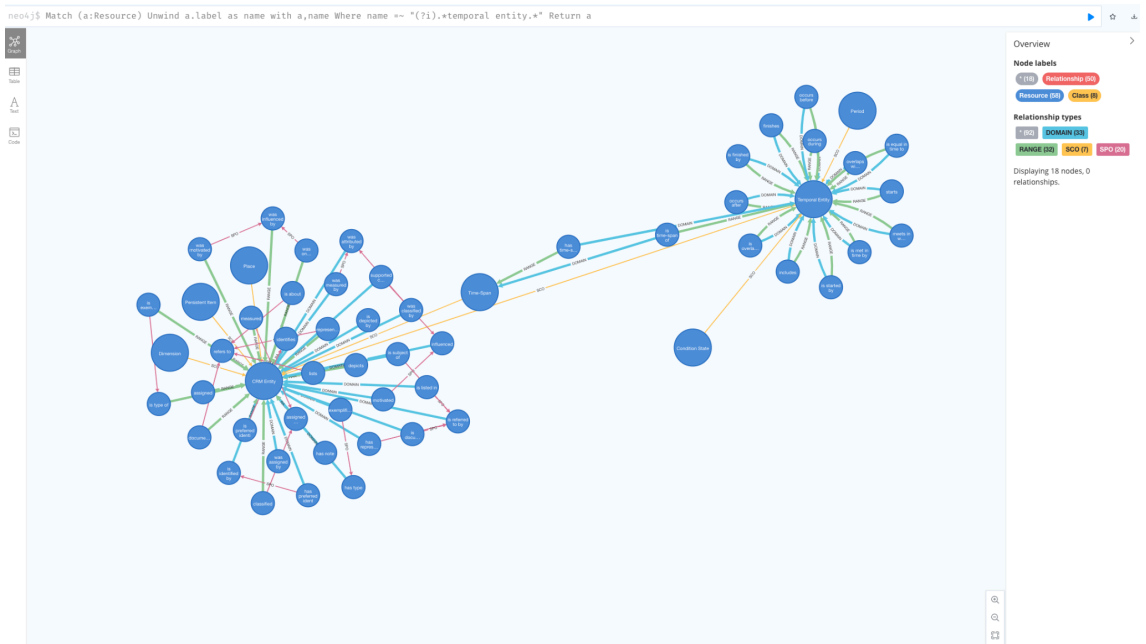


Figure 5.13 Visualisation of the “E1 CRM Entity” cluster of immediate neighbours (cluster on the left) and its connection with the “E2 Temporal Entity” cluster (on the right) from CRM v5.0.4

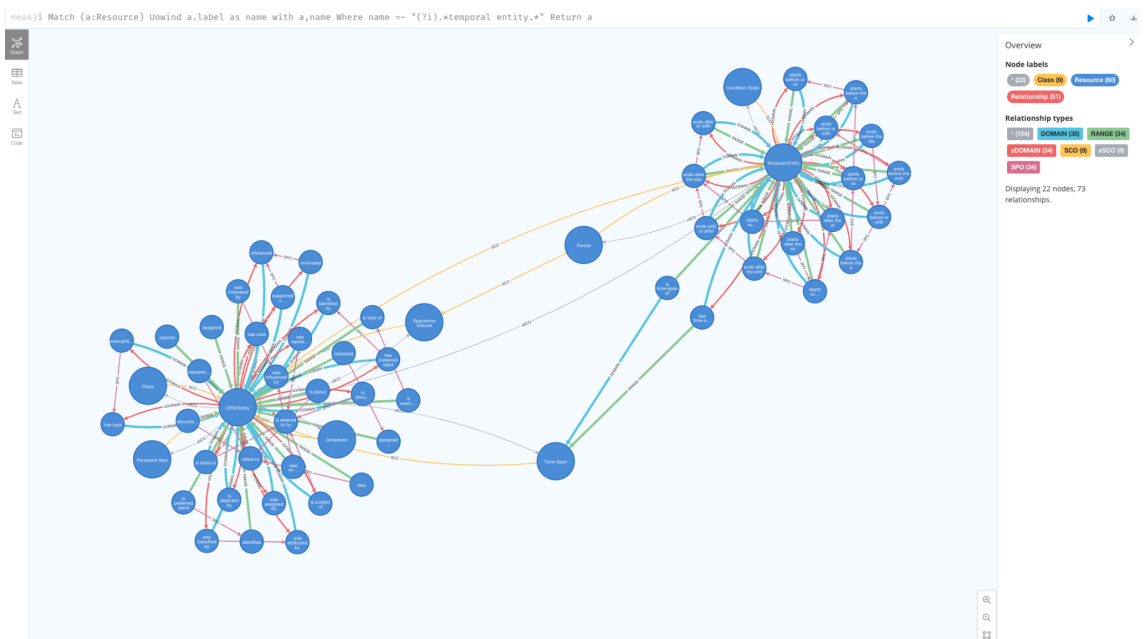


Figure 5.14 Visualisation of the “E1 CRM Entity” cluster of immediate neighbours (on the left) and its connection with the “E2 Temporal Entity” cluster (on the right) from CRM v7.1.1 (ETL2). Here the reciprocal $[:xSCO]$ edge (in gray) transfers influence to E2 Temporal Entity,

5.4 Query-based Analysis and Inference

As mentioned in the last section, follow-up investigations to understand and interpret the analyses results were undertaken through queries and visualisations that target the result-specific graph elements or structures. While the classes, CRM properties, i.e. $(:Relationship)$ nodes, and the overall RDFS imposed edges (i.e. domain, range,

subClassOf, and subPropertyOf) are preserved in the representation, its inheritance and inference rules can only be accessed and simulated as explicit paths and do not conform directly to the automatic inference capabilities of OWL when viewed and queried in Neo4j. Nevertheless, query-based analysis and inference through structured discovery can also be undertaken as a means for familiarisation with the CIDOC CRM.

A common challenge for novice modellers is adapting to the formalism of the CIDOC CRM and its documentation where misinterpretations can arise when reading the specifications in natural language. For example, a novice modeller who misapplies or mistakenly misinterprets the scope notes and instead applies semantic meaning to the labels, looks to apply a “contains” property, as in:

“The alabaster jar contains kohl residue.”

finds “P10_contains” on the numerically sequential property list in the documentation. However, as Figure 5.14 shows, in actuality, there are four different “contains” properties in v7.1.1 (and three different properties in v6.2.1 and v5.0.4 which don’t include “P172 contains”). Figure 5.15 shows screenshots of the full specifications of the first three CRM properties listed, (a) for P10, (b) for P86, and (c) for P89.

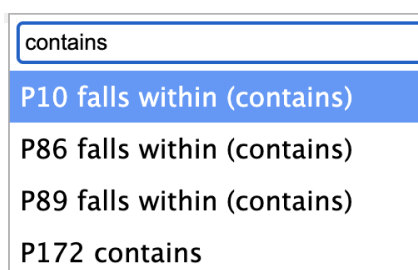





Figure 5.15. Searchable dropdown navigation menu from the Classes & Properties Declaration for v.7.1.1 (links available via Table 5.1.1). A search for “contains” shows three results.

Figure 5.16 shows the resulting visualised graph when a Cypher query is executed to find the last CRM property on the list, P172, using the v7.1.1 (ETL2) model, which has also been stored as its own standalone database. Likewise to the documentation, the scope notes, label and uri, as node properties, are available to the user via the side panel in the graphical user interface. Furthermore, Figure 5.16 shows that by clicking for an expanded view on the “P172_contains” (:Relationship) node reveals its domain and range nodes, that is “E53_Place” (as the subject) on the left and “RDF-schema#Literal” (as the object) on the right. (NB: The “RDF-schema#Literal” node appears blank as it does not have a label property.)

(a) **P10 falls within (contains)**   

Domain: [E92 Spacetime Volume](#)

Range: [E92 Spacetime Volume](#)

SubProperty Of: [E92 Spacetime Volume](#), [P132](#) spatiotemporally overlaps with: [E92 Spacetime Volume](#)

SuperProperty Of: [E93 Presence](#), [P166](#) was a presence of (had presence): [E92 Spacetime Volume](#), [E4 Period](#), [P91](#) forms part of (consists of): [E4 Period](#)

Quantification: many to many, necessary, dependent (1,n:0,n)

Scope Note: This property associates an instance of E92 Spacetime Volume with another instance of E92 Spacetime Volume that falls within the latter. In other words, all points in the former are also points in the latter. This property is transitive and reflexive.

Examples:




- The Great Plague (E4) *falls within* The Gothic period (E4). (Porter, 2009)

In First Order Logic:

- $P10(x,y) \Rightarrow E92(x)$
- $P10(x,y) \Rightarrow E92(y)$
- $P10(x,y) \Rightarrow P132(x,y)$
- $P10(x,y) \wedge P10(y,z) \Rightarrow P10(x,z)$
- $P10(x,x)$

Properties: -

P10 falls within (contains)

(b) **P86 falls within (contains)**   

Domain: [E52 Time-Span](#)

Range: [E52 Time-Span](#)

SubProperty Of: -

SuperProperty Of: -

Quantification: many to many (0,n:0,n)

Scope Note: This property describes the inclusion relationship between two instances of E52 Time-Span. This property supports the notion that the temporal extent of an instance of E52 Time-Span falls within the temporal extent of another instance of E52 Time-Span. It addresses temporal containment only, and no contextual link between the two instances of E52 Time-Span is implied. This property is transitive.

Examples:




- The time-span of the Apollo 11 moon mission (E52) *falls within* the time-span of the reign of Queen Elizabeth II (E52). (Riley, 2009) (Robinson, 2000)

In First Order Logic:

- $P86(x,y) \Rightarrow E52(x)$
- $P86(x,y) \Rightarrow E52(y)$
- $[P86(x,y) \wedge P86(y,z)] \Rightarrow P86(x,z)$

Properties: -

P86 falls within (contains)

(c) **P89 falls within (contains)**   

Domain: [E53 Place](#)

Range: [E53 Place](#)

SubProperty Of: -

SuperProperty Of: -

Quantification: many to many, necessary, dependent (1,n:0,n)

Scope Note: This property identifies an instance of E53 Place that falls wholly within the extent of another instance of E53 Place. It addresses spatial containment only and does not imply any relationship between things or phenomena occupying these places. This property is transitive and reflexive.

Examples:

- The area covered by the World Heritage Site of Stonehenge (E53) *falls within* the area of Salisbury Plain (E53). (Pryor, 2016)

In First Order Logic:

- $P89(x,y) \Rightarrow E53(x)$
- $P89(x,y) \Rightarrow E53(y)$
- $[P89(x,y) \wedge P89(y,z)] \Rightarrow P89(x,z)$
- $P89(x,x)$

Properties: -

P89 falls within (contains)

Figure 5.16. Screenshots of the documentation for P10, P86, and P89.

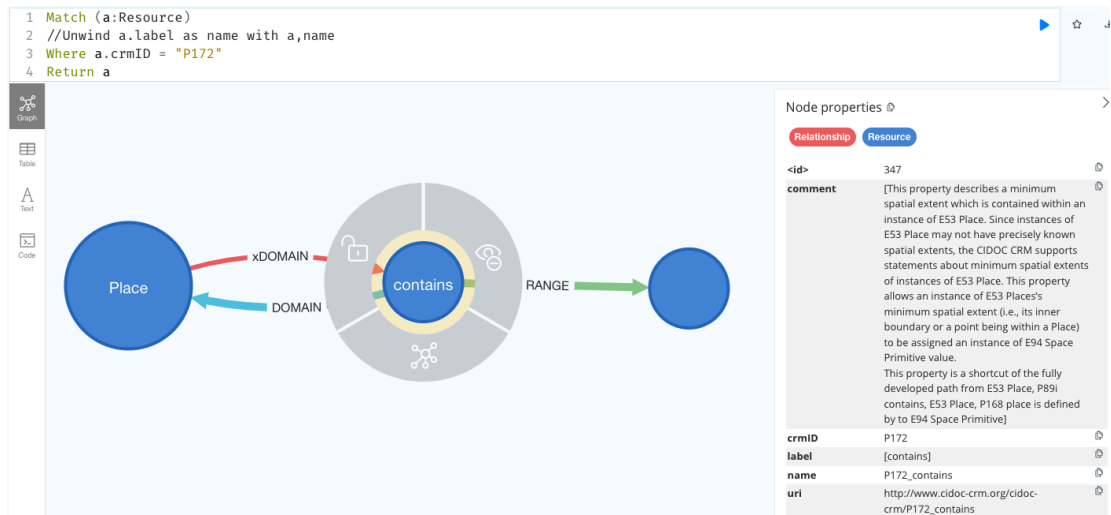


Figure 5.17 Visualisation of results to the Cypher query for finding CRM property “P172”.

The advantage of using a graph-based documentation resource is further demonstrated in Figure 5.17 where the same manual search-and-review process captured in Figure 5.15 is simulated through a Cypher query against the v.7.1.1 CIDOC CRM RDFS graph. All four properties with “contains” is returned (note the row of four nodes near the middle of image) with the three related classes (note the row of three larger nodes near the top of the image) are immediately evident, allowing the user to review each node’s scope notes, explore neighbouring nodes, and to conduct a side-by-side review with the other results.

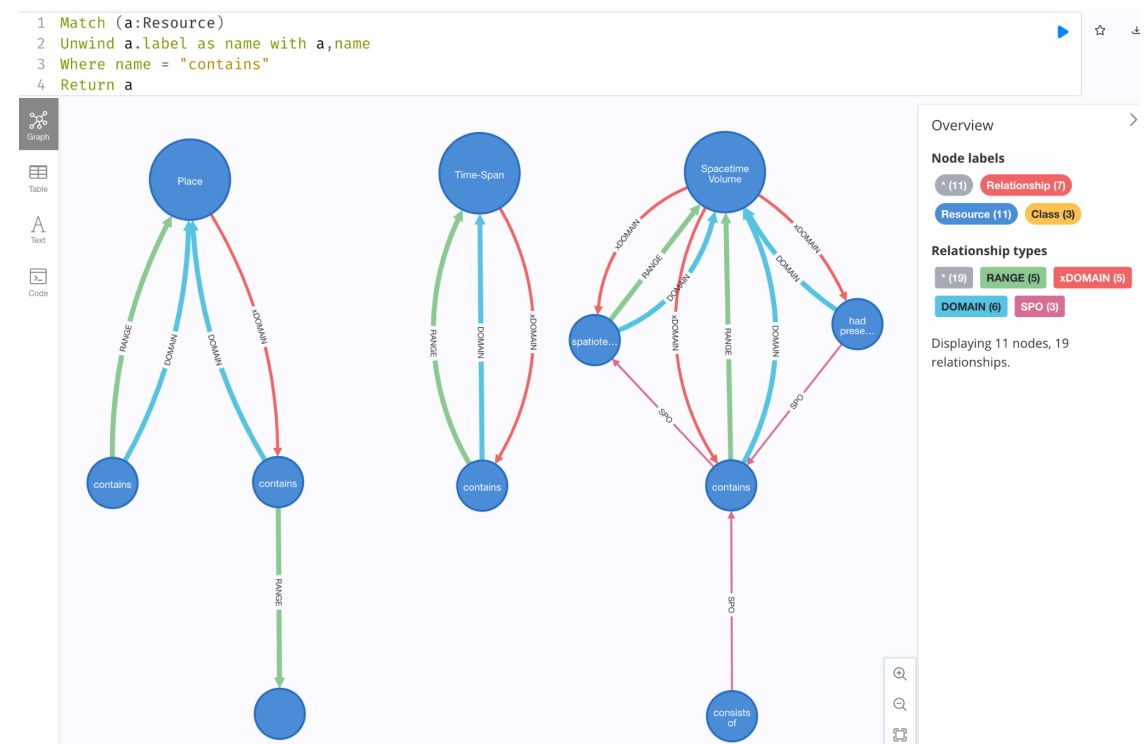


Figure 5.18. The results of searching for [Relationship] nodes with “contains” as the label, simulating review of the documentation.

5.5 Verification, Validation, Calibration

While investigating the CIDOC CRM as a graph, it has been mentioned above that metrics such as eigenvector centrality presents itself as a strong candidate for use as a model validation metric. However, while its application appears promising, this single test case across four variations of one conceptual model is far too limited to assert its role as a validation metric. At best, for now, like leaf node detection, it can serve as a diagnostic tool.

Nevertheless, in this case, a verification and validation procedure did not exist to ensure the imported RDFS model conforms to the formally defined CIDOC CRM. To address this, the first-order logic (FOL) definitions of each CRM resource (classes and properties) were translated into Cypher queries which should successfully return a matching path when executed. This was achieved by using a newly created property, `n.crmID` (see Appendix F on how it was created) to shorten the labels used in the query, that is, instead of using the long `rdf:label`, one can replicate the FOL statements directly in Cypher. For example, the documentation for the following property⁸:

`P68_foresees_use_of (P68i_use_foreseen_by)`

in CIDOC CRM v.7.1.1 (2021)⁹ is defined by FOL statements as follows:

- (a) $P68(x,y) \Rightarrow E29(x)$,
- (b) $P68(x,y) \Rightarrow E57(y)$,
- (c) $P68(x,y) \Rightarrow P67(x,y)$

where the `[:Domain]` is `E29_Design_or_Procedure` and the `[:Range]` is `E57_Material`. The corresponding Cypher query equivalent are:

- (a) `Match p= (a:Resource{crmID:"P68"})-[]->(b:Resource{crmID:"E29"})`
`Return p, relationships(p)`
- (b) `Match p= (a:Resource{crmID:"P68"})-[]->(b:Resource{crmID:"E57"})`
`Return p, relationships(p)`
- (c) `Match p= (a:Resource{crmID:"P68"})-[]->(b:Resource{crmID:"P67"})`
`Return p, relationships(p)`

Thereby a correct match and successful return indicates a correct model structure in the LPG instance of the RDFS encoding. Figure 5.18 below shows the visualised and matched results.

⁸ http://cidoc-crm.org/cidoc-crm/6.2.1/P68_foresees_use_of

⁹ *Classes & Properties Declarations of CIDOC-CRM version: 7.1.1*. (2021). Retrieved May 31, 2023, from https://cidoc-crm.org/html/cidoc_crm_v7.1.1.html

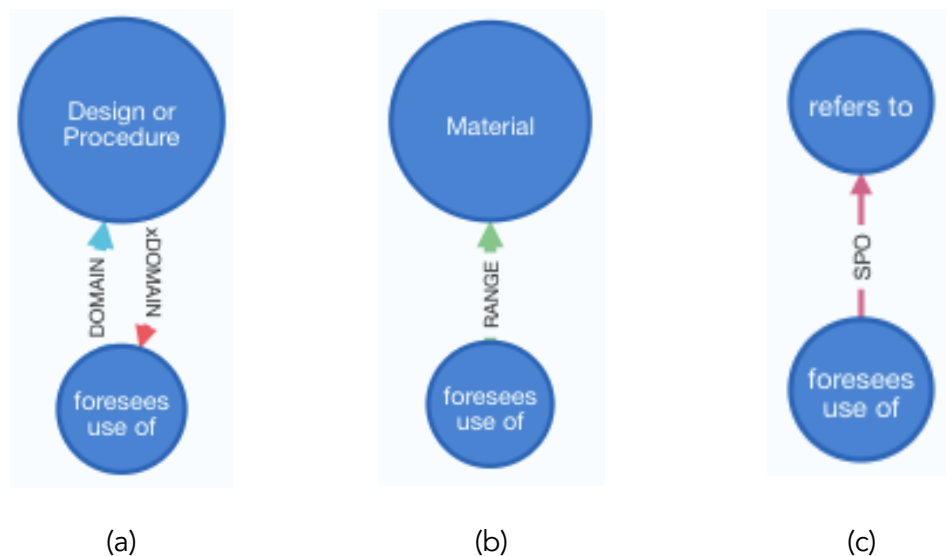


Figure 5.19 The visualised FOL statements as graphs.

5.6 Summary Findings

The key results following LPG-based graph theoretic analysis of the CIDOC CRM RDFS structure have demonstrated that the CIDOC CRM is itself a queryable graph structure where the first-order logic (FOL) statements can be encoded and applied as a code verification and import validation procedure. A full cyclic graph representation of the CIDOC CRM provides a means to explore the RDFS graph beyond limited hierarchical and acyclic tree representations which can mask true leaf nodes that may be diagnostic of areas of sparsity that can be improved or pruned. However, the local and average clustering coefficient analyses has, thus far, been less informative.

The results of the diameter measure and the $k=3,4,5$ motif frequency measures demonstrate clear patterns, respectively. An 8-length undirected diameter was found across all four versions and transformation procedures. The frequency of motif structures are found to be in a logarithmic scale with bell-curve-like bumps across each bracketed ratio region. This may serve as comparative benchmarks or provide some diagnostic insight, that is not yet clear, for subsequent dataset analyses. (In fact, this will be the case in the next chapter.)

A clear outcome from the graph theoretic analysis is the nonplanarity of the CIDOC CRM in its recent iteration (v.7.1.1), demonstrating a domain ontology that is multidimensional with evidence of a development trajectory since v.5.0.4 that tends towards greater connectivity as exemplified by the increasing global triangle count results.

The eigenvector centrality measure has also shown to be a diagnostic measure, particularly for highlighting directionality issues in a model. This coupled with the FOL-based Cypher queries can serve as checks and provide a means to identify and correct problem areas in the model.

Finally, the combined benefits of graph visualisation and queryability has the potential to improve understanding of the CIDOC CRM, particularly for novice modellers, and to support mapping endeavours by allowing users/researchers to explore the ontology's structure and diversifies modelling practice to include making explicit queries instead of browsing across extensive lists or tree diagrams.

6.0 Linked Conservation Data

This chapter will review and analyse CRM-mapped conservation RDF data produced by the Linked Conservation Data (LCD) project. Following the analysis of the core CIDOC CRM ontology in the last chapter, this chapter will explore the choices made by the LCD project modellers when mapping to the CIDOC CRM as evidenced by the resulting conservation RDF graphs. These results and insights will be used to inform the revised LPG model (see chapter 7) and set a benchmark for what is expected from the RDF version of that model.

This chapter is structured as follows: firstly, section 6.1 provides general background information on the Linked Conservation Data project along with key challenges encountered when mapping conservation data to the CIDOC CRM to produce four RDF datasets(?). Section 6.2 highlights the ETL process for the LCD datasets with a focus on key differences between how RDFS is transformed compared to how RDF is transformed into LPG. Section 6.3 presents the results of graph theoretic analysis of the four LCD RDF graphs. The graph theoretic analysis applies agnostic graph measures to each LCD dataset based on the structure and other calculable features of the graph. It does not leverage the content beyond using labels and types, etc. to identify sets. Section 6.4 presents the results of graph-based path queries used to explore the data content. Section 6.5 interprets the results from both graph theoretic and query-based analysis in terms of potential verification, validation and calibration concerns. Finally, section 6.6 provides a summary of the findings.

6.1 Background

The Linked Conservation Data (LCD) project¹ (Velios and St John 2022), funded by the UK Arts and Humanities Research Council, supported a network of over twenty consortium organisations to investigate and devise resources “to improve the dissemination of conservation records”. One of the outputs of this project was a pilot implementation to transform existing conservation datasets into Linked Data that conforms to W3C standards and the CIDOC CRM with the resulting implementation hosted by ResearchSpace².

¹ Linked Conservation Data Project <https://www.ligatus.org.uk/lcd/>

² <https://lcd.researchspace.org/resource/rsp:Start>

The four processed datasets³ in the pilot originate from:

- The Bodleian Library, Oxford, UK (LCD-BOD)
- The Library of Congress, USA (LCD-LOC)
- The National Archives, UK (LCD-TNA)
- Stanford University Libraries, USA (LCD-SUL)

Each participating institution “provided 30-50 conservation treatment reports spanning 40-50 years, all focused on a common book conservation treatment: reattaching detached boards” (Velios and St John 2022). The principal aims for their case study was to determine:

- the history of board reattachment techniques over the last 50 years,
- the time periods across institutions for when certain materials/techniques were used,
- and the relationship between board reattachments to other book conservation treatments and conditions. (ibid.)

The LCD project queried for 15 board reattachment techniques⁴ (see Table 6.1.1) to determine usage trends over time (Velios and St. John 2022, Figure 3).

Table 6.1.1 List of the 15 board reattachment techniques

• Board edge consolidation	• Humidification
• Board reattachment	• Lacing in
• Board slotting/slotting	• Oversewing
• Boards split	• Pasting
• Building up	• Reattaching
• Consolidation	• Rehitching
• Drying	• Repair
• Fraying	

The results of the pilot implementation were achieved with the work of Lieu and Campagnolo (2022) who undertook the modelling and transformation of each contributed dataset from their original formats into RDF/XML format.

³ <https://github.com/linked-conservation-data/board-pilot-data>

⁴ There are 17 named techniques in Velios and St. John (2022), however, one of these is due to tense variations in spelling, e.g. “repairs” and “repaired” and another is due to duplication of terms, e.g. “slotting” and “board slotting”. Hence these duplicates have been removed from Table 6.1.1.

The modelling process (Campagnolo and Lieu [2022]) involved an iterative process of testing queries corresponding to research questions based on modelled data, refining the model and repeating the exercise until all data were reflected in the resulting queries (Velios and St John 2022).

Table 6.1.2 summarises the various original formats in which the contributed datasets were presented to the modellers. However, the process highlighted several challenges that stemmed from modelling legacy data including the variability in formats (e.g. text-based documents, handwritten reports, and checkboxes with limited searchability), the tendency for electronic systems to mirror legacy formats in word-processing file formats, the prevalence of free-text embedded in spreadsheet cells, and the added challenges of encoding from handwritten records. Therefore, plural transformation pipelines were necessary. Figures 6.1 and 6.2 show the transformation pipelines Lieu and Campagnolo devised. An extract from the resulting RDF/XML file in .trig format for the TNA dataset can be seen in figure 6.3 in the next section on ETL.

Table 6.1.2 Summary of LCD datasets for secondary analysis

Dataset	Original format	RDF Modeller	Resulting LCD File, i.e. source for analysis
LCD-BOD	.csv, .docx	Campagnolo	bod-data-2020-12-31.trig
LCD-LOC	.csv	Campagnolo	loc-data-2020-12-31.trig
LCD-TNA	.csv	Campagnolo	tna-data-2020-12-31.trig
LCD-SUL	.docx, .pdf	Lieu	sul-data-2021-01-22.trig

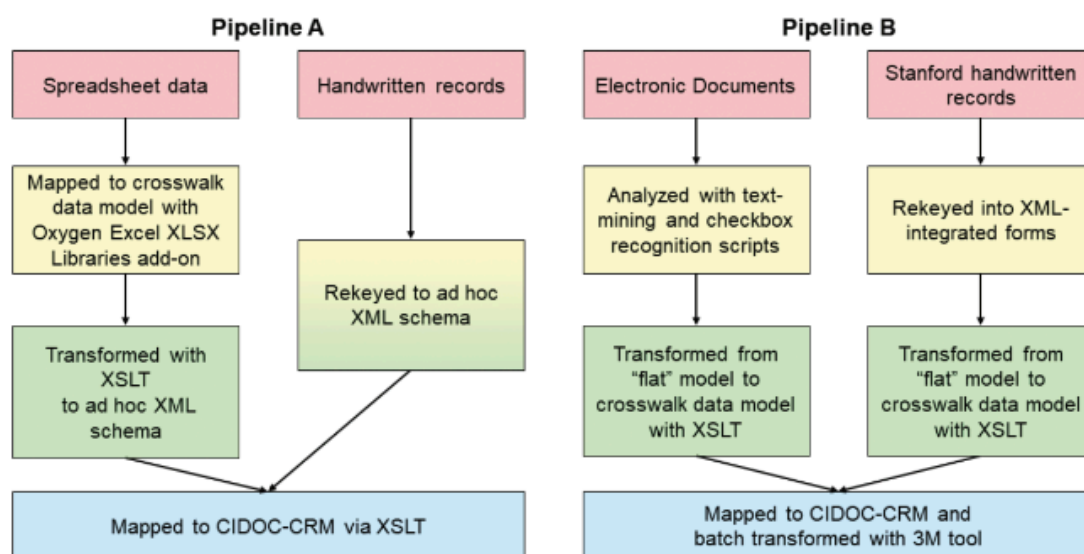


Figure 6.1. LCD project transformation pipelines (Image source: Lieu and Campagnolo 2022, Figure 5).

The 3M tool at the end of pipeline B in Figure 6.1 stands for ‘Memory Mapping Manager’, a free and open source tool for mapping XML data to Linked Data and was designed specifically for the CIDOC CRM (FORTH 2019). Finally, the resulting RDF output was reviewed using the RDF graph visualisation tool, CRMVIZ⁵, a Python-based tool and library developed by Velios (2020) specifically for CRM-mapped triples (Lieu and Campagnolo 2022). While the transformations conform to the CIDOC CRM, a simplified summary data model for the LCD project is shown in Figure 6.2.

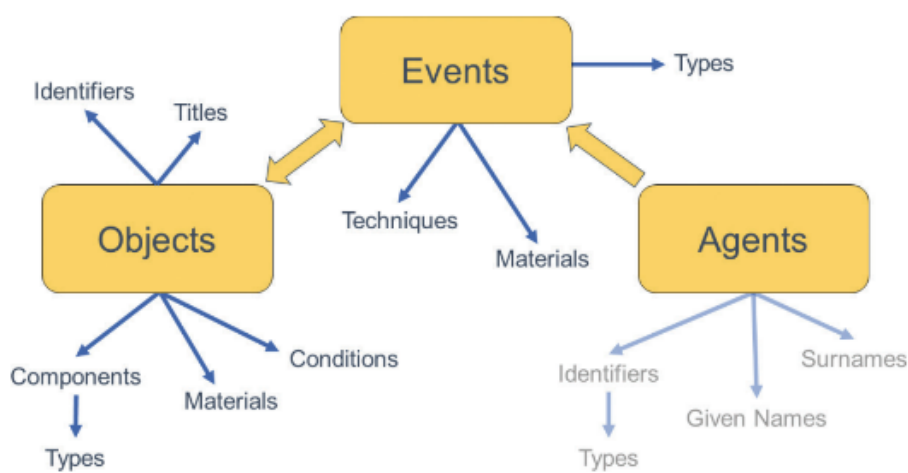


Figure 6.2. LCD project data model. (Image source: Lieu and Campagnolo 2022, Figure 3).

The process highlighted several challenges in modelling and encoding of the conservation records:

Linked Data can be produced in a variety of ways depending on the resources and expertise available in each institution. Integrating the resulting models to allow cross-searching required considerable effort (Velios and St John 2022).

Each modeller approached their primary sources as conceptual objects. Lieu (2022, personal communication) conceived of the treatment record as the main conceptual object in the mapping process, whereas Campagnolo’s approach (2022, personal communication) centred on describing the collection item (i.e. the book) as a conceptual object.

The transformation pipeline for the BOD, LOC, and TNA datasets were influenced by the most complex dataset of the group, BOD, which included data from different time

⁵ <https://github.com/natuk/crmviz>.

periods, had handwritten content, tickbox content, and both digital content in word and scanned long-hand. Campagnolo (2022) devised a schema for manually mapping data in XML. Three sections of the schema described the object, the condition, and the treatment. However, a drawback of the XML schema approach was the difficulty in reconciling which part of the book was treated with its corresponding description. Another challenge was mapping encodings for description and conditions across multiple parts of an object. The LOC and TNA data were more straightforward in their transformations as they were contributed in spreadsheet format (.xls) and were directly exported as XML. This intermediate XML schema was then further aligned with the XML schema devised for the BOD dataset (ibid.).

By contrast, the SUL dataset (Lieu 2022) was derived from born-digital records that began in 2014. The highly-structured data had been captured by a process which utilised over 400 checkboxes per record (i.e. TRUE/FALSE values) and a system that already utilised triple-like statements, for example:

- leather consolidated with specific material A,
- leather consolidated with specific material B.

However, the existing data model was object-centric which led to a modelling focus on aggregating related parts before modelling the treatment event. In hindsight, Lieu recommends modelling by event first and object second in line with the CIDOC CRM (ibid.).

The Linked Conservation Data project hosted several modelling workshops to model conservation data to the CIDOC CRM. This author attended two of these workshops, one in person (September 12, 2019) and the other online (January 25, 2021). It was observed during these workshops, which were primarily attended by conservation professionals, that a common challenge for novice modellers was to familiarise themselves with the CIDOC CRM and attempt mapping conservation data to it. Which class is the correct class to map to? Which property? Such early-stage modelling decisions have a significant impact upon the outcome.

For clarity, this author was not involved in the transformations carried out by Lieu and Campagnolo and only had a limited role in the LCD project as an attendee to the two LCD-sponsored workshops. However, one of the principal investigators (PI) of the LCD project, Dr. Athanasios Velios, is also supervisor for this thesis. Access to the original data sources were provided by both principal investigators, Dr. Athanasios Velios and Kristen

St. John, Head of Conservation Services at Stanford Libraries. Interviews were also conducted with the data modellers (Lieu and Campagnolo).

6.2 ETL: Importing the LCD RDF models into Labelled Property Graph

The ETL procedure for importing the four LCD RDF graph models into Neo4j's LPG platform was straightforward and undertaken using the Neosemantics plugin (full ETL details can be found in Appendix G). The ETL1 import procedure was sufficient and ETL2 is not necessary. Unlike the CIDOC CRM's RDF Schema (RDFS) graphs in the last chapter, the LCD data graphs are mapped to the CIDOC CRM and directly encoded as RDF/XML using the TriG format. TriG is an extension of the Turtle format (Bizer and Cyganiak 2014) to accommodate for named graphs⁶ (Carroll et al 2005). Figure 6.4 below shows an excerpt from the TNA dataset encoding in TriG format.

```
1361 <http://www.ligatus.org.uk/lcd/02cf2016-ec38-4f9d-90d8-a34529fb0026/> a crm:E22_Man-Made_Object ;
1362 rdfs:label "Bookblock (The National Archives, CRIM 10/28)"@en ;
1363 crm:P2_has_type <http://w3id.org/tna-vocab/2af7d0e3-474f-4069-9cd0-2186c2016670> ;
1364 crm:P46_is_composed_of <http://www.ligatus.org.uk/lcd/29893e7a-1b62-4c97-a649-6e33024232fa/>,
1365 <http://www.ligatus.org.uk/lcd/4d2d687b-8364-4349-af76-6a9f99e38872/>,
1366 <http://www.ligatus.org.uk/lcd/576fa711-4ef9-4e1e-92b6-f6e73ca0e354/>,
1367 <http://www.ligatus.org.uk/lcd/8e363fe1-fc63-419f-8070-335dc00e58a8/>,
1368 <http://www.ligatus.org.uk/lcd/c2f5cefa-65d1-4877-bc3c-f06be9514fc1/>,
1369 <http://www.ligatus.org.uk/lcd/ea4d74c0-c1f6-4c35-b48d-efe77cb68680/>,
1370 <http://www.ligatus.org.uk/lcd/eff046fe-5526-4814-9c42-3f460a03ee55/> ;
1371 crm:P56_bears_feature <http://www.ligatus.org.uk/lcd/2045d2b3-e1b8-4602-b519-8698f9b0cd4f/> ;
1372 crm:P59_has_section <http://www.ligatus.org.uk/lcd/02cf2016-ec38-4f9d-90d8-a34529fb0026#headPlace>,
1373 <http://www.ligatus.org.uk/lcd/02cf2016-ec38-4f9d-90d8-a34529fb0026#leftPlace>,
1374 <http://www.ligatus.org.uk/lcd/02cf2016-ec38-4f9d-90d8-a34529fb0026#rightPlace>,
1375 <http://www.ligatus.org.uk/lcd/02cf2016-ec38-4f9d-90d8-a34529fb0026#spinePlace>,
1376 <http://www.ligatus.org.uk/lcd/02cf2016-ec38-4f9d-90d8-a34529fb0026#tailPlace> .
1377
1378 <http://www.ligatus.org.uk/lcd/02f3b135-5831-41c5-be92-513df555a444/> a crm:E3_Condition_State ;
1379 rdfs:label "Broken (The National Archives, SP 104/143)"@en ;
1380 crm:P2_has_type <http://w3id.org/tna-vocab/240e44f6-6191-483a-8d6e-bea3c1bef719> .
1381
1382 <http://www.ligatus.org.uk/lcd/0315e06a-bd0f-4e55-9160-8f97f193ad28/> a crm:E11_Modification ;
1383 rdfs:label "Modification of The National Archives, B 3/5243"@en ;
1384 crm:P31_has_modified <http://www.ligatus.org.uk/lcd/34700b1c-a452-4dc3-a32f-00840d3fdcf/> ;
1385 crm:P32_used_general_technique <http://w3id.org/tna-vocab/aad6a0c7-50d2-46c2-b5cd-a13bbfb3219> .
1386
1387 <http://www.ligatus.org.uk/lcd/033b4036-3fad-4a53-8bcc-82245bd28a83-1/> a crm:E22_Man-Made_Object ;
1388 rdfs:label "Sewing supports (The National Archives, E 405/550)"@en ;
1389 crm:P2_has_type <http://w3id.org/tna-vocab/e808420b-db71-47a1-92cc-a88de1b6746a> .
1390
1391 <http://www.ligatus.org.uk/lcd/03b4228d-5f73-4cea-af79-c5818c9c6118/> a crm:E22_Man-Made_Object ;
1392 rdfs:label "Right endleaves (The National Archives, SP 44/335)"@en ;
1393 crm:P2_has_type <http://w3id.org/tna-vocab/33258881-c175-40f9-be1f-2ccc3aceb41> ;
1394 crm:P55_has_current_location <http://www.ligatus.org.uk/lcd/0e34a853-793b-47bc-b37c-4c9b8654e4a6#rightPlace> .
1395
1396
```

Figure 6.3 Excerpt in TriG from *tna-data-2020-12-31.trig*

In the last chapter, it was shown how the CIDOC CRM RDFS import into Neo4j's LPG structure used the broad terms for (:Class) and (:Relationship) as the node labels while the CRM specific class and property names were transformed as node properties. In contrast, the LCD datasets being RDF files and not RDF Schema were imported via Neosemantics so that the specific class names were mapped as the node labels, for example, (:E11_Modification) and (:E57_Material), with the data instance *rdfs:labels* transformed into node properties. Neosemantics also converts all object nodes that are

⁶ Named Graphs are RDF triples that are identifiable by a URI.

literals, and therefore are leaf nodes, into node properties (Barrasa 2016). For example, the datetime values “2010-01-01T00:00:00” and “2020-12-31T23:59:59” for the CRM properties P82_begin_of_the_begin and P82b_end_of_the_end, respectively, are converted to node property key-value pairs on a (:E52_Time-Span) node as shown in Figure 6.4. Another CRM property with a literal as object is P3_has_note, which are therefore also converted into node properties with “P3_has_note” as the property key and the literal object as the property value. Otherwise, CRM properties remain as edges and are not transformed into nodes as in the CIDOC CRM RDFS examples in the last chapter. Neosemantics attends to the repeated mention of a resource by only creating it once so to avoid the repeated resource visualisation problem noted by Hayes and Gutierrez (2004) (and as demonstrated in Figure 4.11 above regarding the Titanic example).

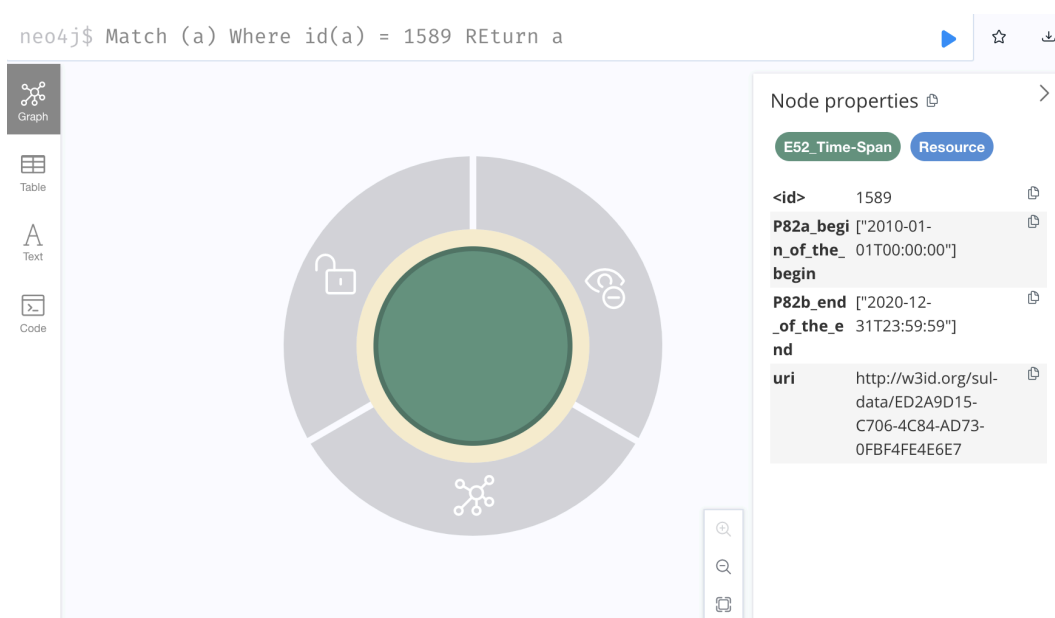


Figure 6.4 Visualisation of a E52_Time-Span node with CRM properties P82a and P82b transformed into node properties.

6.3 Results of Graph Theoretic Analysis

The results of the graph theoretic analyses are presented as follows with the BOD, LOC, and TNA datasets grouped together in alphabetic order followed by the SUL dataset. This presentation has been chosen to emphasise detected patterns found across the results of several graph theoretic methods. It reveals a shared pattern in the modelling of BOD, LOC, and TNA (particularly when comparing the latter two) while consistently, the SUL dataset stands apart. The reason for these distinct patterns align with how the original contributed datasets were compiled and subsequently modelled as noted above

in section 6.1 *Background*. Nevertheless, as the following results will also demonstrate, despite these differences, contextual commonalities were also identified.

Unlike the CIDOC CRM RDFS graphs in Chapter 5 where graph theoretic analysis provided insights into the Classes and Properties of the ontology and therefore retained similar results across each version, the LCD graphs are graphs of data instances and, therefore, not every CRM Class or Property will feature, only those classes matched and mapped to data instances by the modeller.

6.3.1 Order and Size

Table 6.3.1. Order and Size Results for Linked Conservation Data Group

	LCD-BOD	LCD-LOC	LCD-TNA	LCD-SUL
Order (node ct)*	2,451	1,707	2,119	2,219
Size (edge ct)	5,481	3,611	4,611	5,753
Node:Edge Ratio	1:2.24	1:2.12	1:2.18	1:2.59
Node:Edge (as quotient)	0.45	0.47	0.46	0.39

Firstly, all four datasets in this group have a similar node-to-edge ratio that is roughly one-to-two. This falls within the trend as seen from the CIDOC CRM results, bearing in mind the CIDOC CRM nodes are conceptual classes while the LCD group are instances matched to such classes.

6.3.2 Density/Sparsity

Table 6.3.2. Density/Sparsity Results for Linked Conservation Data Group

	LCD-BOD	LCD-LOC	LCD-TNA	LCD-SUL
Edge Density*	0.0009	0.0012	0.0010	0.0012
Leaf Nodes	214	205	115	220
Isolated Nodes*	1	1	1	1
Leaf + Isolated*	215	206	116	221
Theta Ratio*, θ	0.0878	0.1207	0.0547	0.0996
Average Clustering Coefficient	0.0282	0.0250	infinity	infinity

Following on, the Neosemantics-derived RDF triple structure used across all four LCD datasets yields an edge density that is comparable across the board and can be rounded to 0.001. Due to the influence of the RDF triple structure on the node to edge counts, as the edge density is also calculated from these counts, the current conjecture is that an edge density of 0.001 may be indicative of CRM-mapped RDF triples. However, this will need to be revisited with further research.

Each LCD graph contained a small ratio (θ) of leaf nodes compared to total nodes, between 5-12%. This is due to the transformation from RDF to LPG where all objects that are literals (the literal node having been a leaf node in the RDF graph), have now been transformed into node properties on the only neighbouring node to the literal. Thus, the leaf nodes count in Table 6.2.2 is smaller than the leaf node count of the original RDF graph. For example, using the above E52_Time-Span example from figure 6.5, instead of have two leaf nodes in RDF for "2010-01-01T00:00:00" and "2020-12-31T23:59:59", in LPG, the E52_Time-Span node itself, with both those datetimes now as node values, becomes the leaf node.

The single isolated node in Table 6.3.2 refers to the (:_GraphConfig) node which is an artefact of the Neosemantics import process and stores the import configuration. Its presence as the only isolated node is to be expected and its contributions to each graph measure is consistent and negligible.

The average clustering coefficients for BOD and LOC are fairly low, out of a range of 0 to 1 (where 1 is fully connected forming triadic closures with neighboring nodes and neighboring nodes also being connected). Therefore, these results indicate these graphs are not well-connected within themselves and there is potential for enrichment to increase triadic closures. However, the "infinity" results for TNA and SUL tell a different story. When following up on the "infinity" scoring nodes under a local clustering coefficient analysis (see Table 6.2.6), these nodes are found to be not highly connected and are actually leaf nodes (i.e. with degree 1). *Then why are they scoring "infinity"?* Further review of the literature suggests a possible explanation from the work of Estrada (2016), who found:

that the average [Watts-Strogatz] clustering coefficient and the network transitivity can diverge for certain classes of graphs. The windmill graphs are examples of graphs in which this phenomenon occurs due to the fact that there

are many cliques⁷ connected to a single node in which no pair of nodes from different cliques are connected.

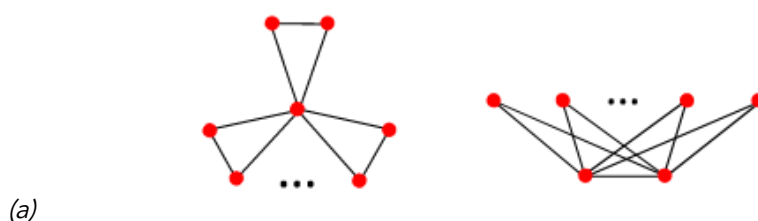
Figure 6.5 (a) shows the two classes Estrada (2016) has identified that exhibit this divergence in clustering behaviour. While Estrada refers to the first of this type as a windmill graph, he does not make any reference to alternative names for the second type (which this author suggests resembles the side-view of a lotus flower). Figure 6.5 (b), also by Estrada, shows the smallest windmill graphs.

To clarify, the “infinity” scoring leaf nodes are not the single, central nodes of a windmill graph as Estrada describes, nevertheless, they are connected to neighbouring nodes which are highly connected and there is visual evidence that windmill-like patterns exist (see Figure 6.6). However, more follow-up work will be needed in this matter as this falls beyond the scope of the current study (See Chapter 8, section 8.6 regarding recommendations for further work).

Table 6.3.3. Local Clustering Coefficient Results for Linked Conservation Data Group

Local Clustering Coefficient - Linked Conservation Data (LCD) Group (top scores)				
<i>dataset</i>	<i>node id</i>	<i>n.label</i>	<i>node Label/CRM Entity</i>	<i>score</i>
LCD-BOD	857	["New endbandstail (Bodleian, Inc.d.f2.1494.2)"]	["Resource", "E22_Man-Made_Object"]	4.6667
LCD-LOC	487	["Broken (Library of Congress, 3968)"]	["Resource", "E3_Condition_State"]	5
LCD-TNA	1281	["Modification of The National Archives, ADM 37/5039"]	["Resource", "E11_Modification"]	3
LCD-SUL	1589	null*	["Resource", "E52_Time-Span"]	Infinity

*This node refers to an E52_Time-Span node where "P82a_begin_of_the_begin: "2010-01-01T00:00:00" and P82b_end_of_the_end: "2020-12-31T23:59:59" and encompasses the time from 1 Jan 2010 - 31 Dec 2020, inclusive.



⁷ A clique is an induced subgraph that is complete, that is, each node is connected to all the other nodes in the subgraph.

BOD's "New endbandstail (Bodleian, Inc.d.f2.1494.2)" and TNA's "Modification of The National Archives, ADM 37/5039", respectively, which do not feature any cliques. Reflections on the choice of using Local Clustering Coefficient will be discussed further in chapter 8. Likewise, the Label Propagation method will be discussed further in section 8.6. as recommended for future work.

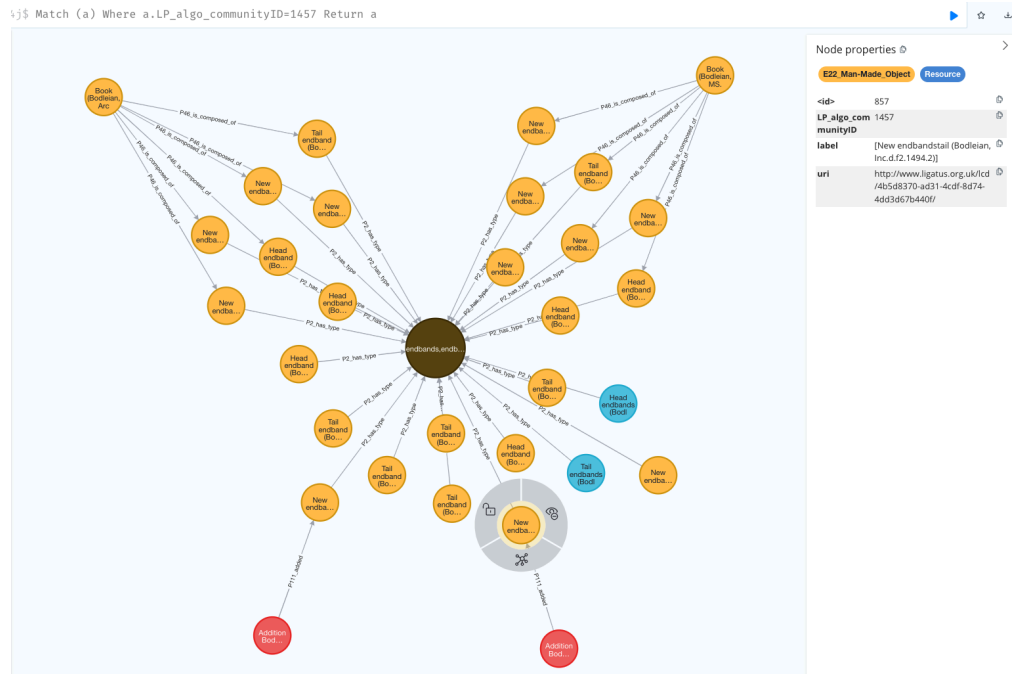


Figure 6.7. The LCD-BOD results of the Label Propagation analysis that identifies the community that includes "New endbandstail (Bodleian, Inc.d.f2.1494.2)" (highlighted with a gray ring, nodeID 857).

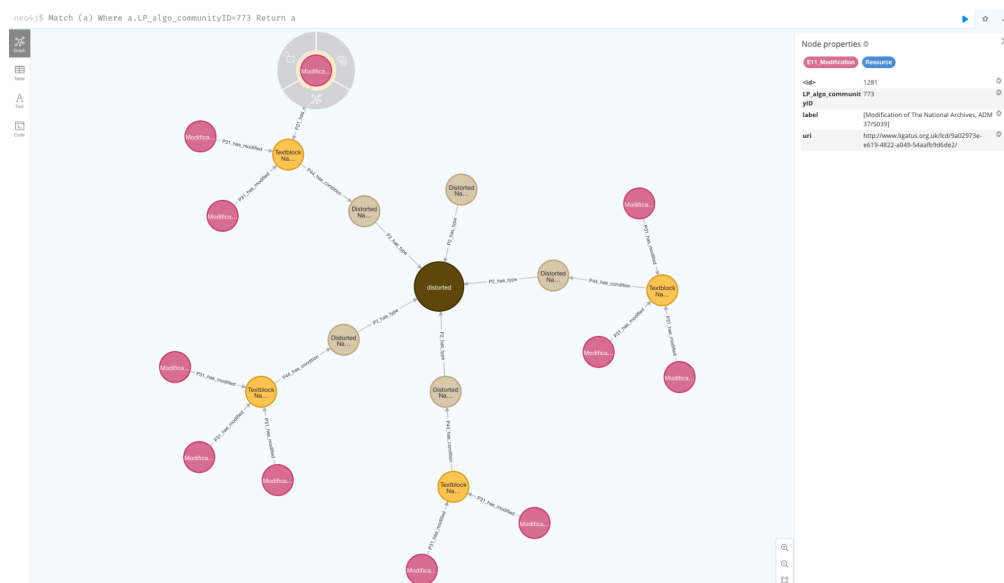


Figure 6.8. The LCD-TNA results of the Label Propagation analysis that identifies the community that includes "Modification of The National Archives, ADM 37/5039" (highlighted with a gray ring, nodeID 1281).

Table 6.3.4. Degree Centrality Results for Linked Conservation Data Group

Degree Centrality - Linked Conservation Data (LCD) Group (highest degrees)					
		<i>node id</i>	<i>n.label</i>	<i>node Label/CRM Entity</i>	<i>degrees</i>
LCD-BOD	Natural	628	["thread"]	["Resource", "E55_Type", "E57_Material"]	49
	Reverse	1117	["corners"]	["Resource", "E55_Type"]	102
	Undirected	1117	["corners"]	["Resource", "E55_Type"]	102
LCD-LOC	Natural	1368	["Main conservation event (Library of Congress, 3995)"]	["Resource", "E11_Modification"]	23
	Reverse	1045	["right"]	["Resource", "E55_Type"]	74
	Undirected	1045	["right"]	["Resource", "E55_Type"]	74
LCD-TNA	Natural	1456	["Main conservation event (The National Archives, DL 30/603/2)"]	["Resource", "E11_Modification"]	28
	Reverse	2043	["repaired"]	["Resource", "E55_Type"]	122
	Undirected	2043	["repaired"]	["Resource", "E55_Type"]	122
LCD-SUL	Natural	700	["dataset"]	["Resource", "E89_Propositional_Object"]	52
	Reverse	405	["spine linings", "spine lining"]	["Resource", "E55_Type"]	106
	Undirected	405	["spine linings", "spine lining"]	["Resource", "E55_Type"]	106

When analysing for nodes in order of greatest degree (i.e. Degree Centrality), Table 6.3.4 shows that for undirected results across all four LCD datasets it is an E55_Type node that has the most incoming and overall most undirected edges. The greatest degree for natural or outgoing direction for BOD remains a type node for "thread". This aligns with expectations as such categorical type nodes serve to pull together and reflect proportionately with data instances. The E11_Modification nodes for LOC and TNA with high outgoing edges suggests these two events involved many components, and therefore, it is not unusual to find they are "Main" conservation events. Finally, the

“dataset” node for SUL is representative of the dataset itself and therefore also aligns with expectations for which nodes tend towards being high-degree nodes.

6.3.3 Global Triangle Count

Table 6.3.5. Global Triangle Count Results for Linked Conservation Data Group

	LCD-BOD	LCD-LOC	LCD-TNA	LCD-SUL
Global Triangle Count	152	97	141	1047

The LCD graphs appear very similar in terms of their results to density and sparsity measures. However, a noticeable variation between the datasets emerge when measuring for global triangle frequency. LOC, the smallest dataset of the group, also has the fewest triangles at 97. However, the order (node count) is not indicative of how those nodes are connected. While BOD, TNA, and SUL each have over 2000 nodes, the SUL dataset has nearly 7 times more triangles than the other two, even though BOD technically has the most nodes of the three. This rises to nearly 11 times more triangles than the smallest dataset, LOC. Thus, although all four datasets are represented in RDF and have similar edge densities, SUL is a much more highly-connected network of data at the small, local level than the other three. The variability and similarities in the data capturing practices (presented in section 6.1) across the institutions is reflected in the network connectivity of each dataset.

6.3.4 Diameter

Table 6.3.6. Diameter Results for Linked Conservation Data Group

	LCD-BOD	LCD-LOC	LCD-TNA	LCD-SUL
Diameter (undirected)	12	10	9	10
Diameter (directed, outgoing)	8	6	6	6
Diameter (directed, incoming)	7	6	6	7

All four LCD datasets have diameters between lengths 9 to 12 (depending on the direction parameter) which are greater than the range of diameters of the CIDOC CRM (not including the v7.1.1 ETL2 diameters as ETL2 does not apply here). However, using

directed path assessments, LOC and TNA have shorter diameters with both incoming and outgoing diameters at 6. In fact, the LOC dataset, containing the smallest set of nodes and edges of the four datasets, has the same measured diameter as that of SUL with a path length of 10. The standout dataset here is BOD which shows a lengthier traversal through the graph under all path assessments, despite sharing in the RDF edge density as the others and having a fairly high global triangle count, if we exclude SUL as an outlier. This indicates that while a substantial part of the BOD data network is well-connected through triads, there are also areas of the data network that are less well-connected and require traversing “the long way around” specific routes to reach certain nodes. Like the diameters of the tested versions of the CIDOC CRM, the shortest diameter possible from the LCD group is length 6. Further study is needed to determine if this is a specific characteristic of a CRM-mapped dataset. The longest diameter result is length 12 for the BOD dataset (undirected).

As shortest path queries on well-connected graphs can take a long time to run through the exhaustive path permutations, they can be costly in terms of processing, memory usage, and system performance. Therefore, identifying diameter thresholds or ranges can be applied to bind diameter queries with an upper limit. Including upper limit binding parameters will stop the query once the limit is reached and return the last calculated diameter. To put this in perspective, as mentioned above in section 4.4.4 *Paths, Distance, Shortest Path and Diameter*, the calculated diameter of the World Wide Web in the 1990s was 19 and if it were even “1,000%” larger, its diameter would still only be 21 (Albert et al 1999). By comparison, the BOD dataset only has 5,481 edges but a diameter of length 12 (undirected). Despite the original LCD modelling procedures having tailored a mapping schema specifically to accommodate the complexities of this dataset, the diameter result strongly indicates that there is complementary data missing, likely related to the sampling across various legacy records and legacy record types to compose his dataset.

6.3.5 Planarity and $K_{3,3}$ Bipartite Graph

Table 6.3.7. $K_{3,3}$ Bipartite Graph Results for Linked Conservation Data Group

	LCD-BOD	LCD-LOC	LCD-TNA	LCD-SUL
$k_{3,3}$ Count	293,400	1,753,632	1,096,704	368,424

Successful detection of $k_{3,3}$ bipartite graphs in all four LCD datasets demonstrates that each institutions’ contributed dataset exhibits a non-planar topology. The implications

of this are that the data captured do reflect the expected pattern of real-world graphs. There also appears to be an inverse correlation between global triangle counts and $k_{3,3}$ bipartite graph counts. While BOD and SUL have higher global triangle counts, LOC and TNA have higher $k_{3,3}$ counts, albeit the relationships are not linear. However, it is premature to assert such correlation as diagnostic.

6.3.6 Undirected Motif Frequency

The motif patterns reveal (in Table 6.3.8) a shared pattern of motif sub-structures that are likewise present or absent between the LOC and TNA datasets. The BOD dataset appears to largely adhere to this pattern, with a few exceptions (m4.5, m5.2, m5.3, and m5.7 are present in BOD, whereas they are absent in LOC and TNA). The SUL dataset appears distinctly different in this regard, with all motif sub-structures present except m5.21 (also known as k_5 or k-5 complete). None of the datasets had the m5.21 (k-5 complete) motif sub-structure.

However, a closer look at the node-to-edge ratios of each motif reveals a similar pattern to that found with the CIDOC CRM graphs as presented in the previous chapter, that is, those motifs with node-to-edge ratios greater than 1:1 tend to be absent from the graphs (i.e. have zero motif counts). Table 6.3.9 breaks down each motif by node and edge counts and provides their ratios. The green-highlighted ratios are less than 1:1 whereas red-highlighted ratios are greater than 1:1. It is clear when reviewing these ratios in order on a bar graph (Figure 6.10) that the frequency of motifs tend to drop off as the bar graph is read from left to right. The stand-out exception is the m5.16 motif which can be found in high frequency across all four LCD datasets.

Follow-up investigation to identify what data content do m5.16 motif structures represent in each of the LCD datasets (see Figure 6.9) reveal that the high frequency of the motif is due to its structure being isomorphic to a tripartite graph structure where a central set contains more than three nodes while the two sets represented at either side consist only of a single node and where relationships connect across the three sets, but the central nodes do not connect with each other (see Figure 6.10). The m5.16 results shown in Figure 6.9 were queried with a "Limit 1" specification in the RETURN clause in order to review only one visualised example. Figure 6.10 shows the results to the same query but the limit was increased to "Limit 5" which reveals that the m5.16 results were only partial subgraphs to these more significant patterns, i.e. the tripartite graphs in BOD, LOC, and TNA and connected m4.6 and m5.7 in SUL.

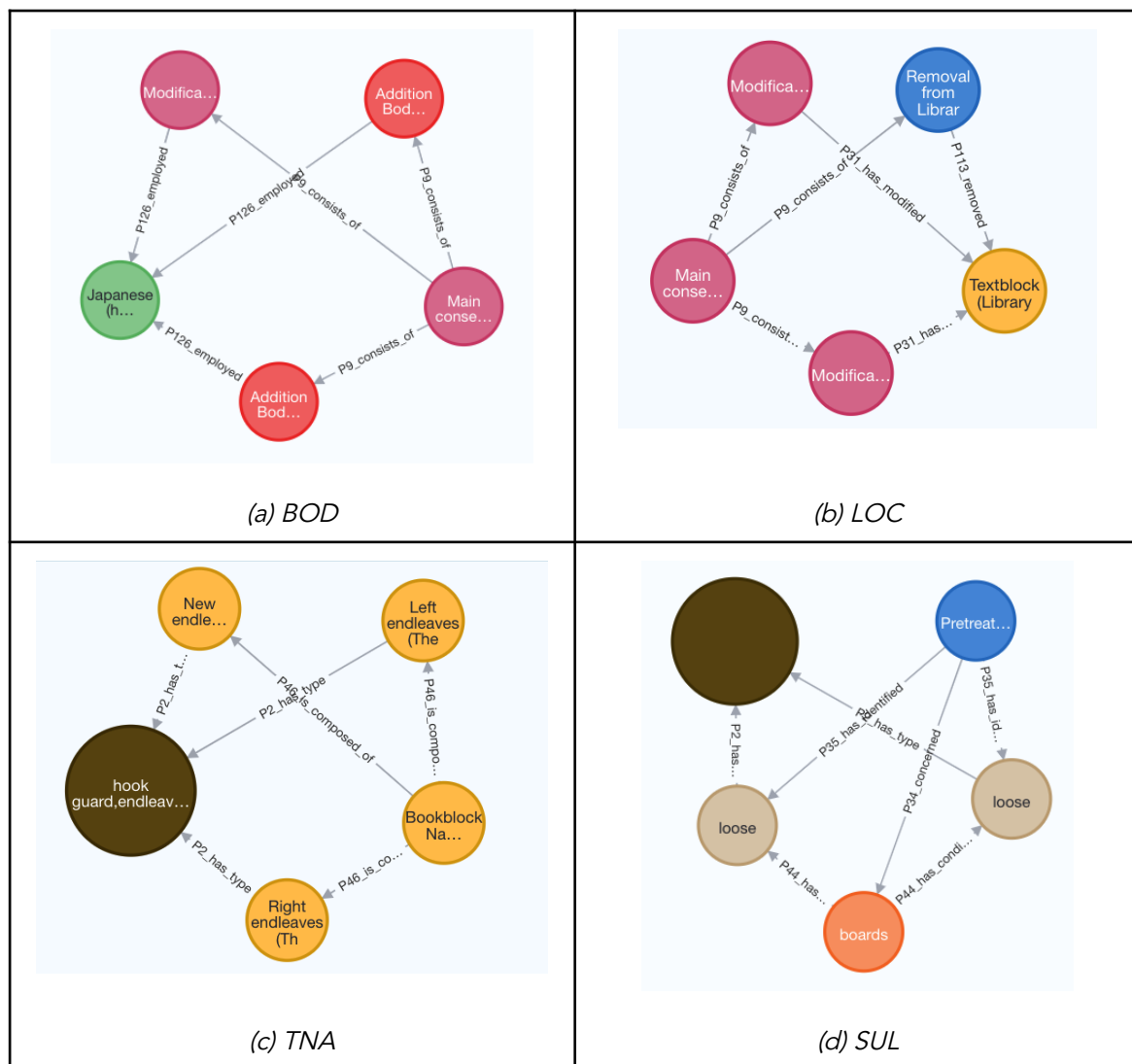


Figure 6.9 Visualisation of m5.16 motif (Limit 1) from each LCD dataset. Note that image (d) for SUL contains an m5.16 structure within another subgraph structure.

For example, in the BOD results in Figure 6.9 (a), the green node is “Japanese tissue” whereas the single pink node at the bottom right is a “main conservation event”. The middle set of red and pink nodes are E79_Part_Addition and E11_Modification nodes that are sub-treatments to the “main conservation event”. Each path across represents the single “main conservation event” consisting of a sub-modification (the central nodes) which employed “Japanese tissue”. Likewise, Figure 6.9 (b) for LOC shows a series where a “main conservation event” consists of several sub-modifications that all modified a specific “textblock”. Finally, Figure 6.9 (c) for TNA shows a series where a “Bookblock” is composed of various parts, such as old and new endleaves, which are all of type “hook guard, endleave”.

This closer look also reveals that while m5.16 substructures technically exist in SUL, they are only a partial representation of where m4.6 (also known as k-4 complete) shares nodes with an m5.7 structure (see Figure 6.10 (d)). As Figure 6.9 (d) shows, there is an

extra edge from the top right blue node to bottom center orange node in the visualisation.

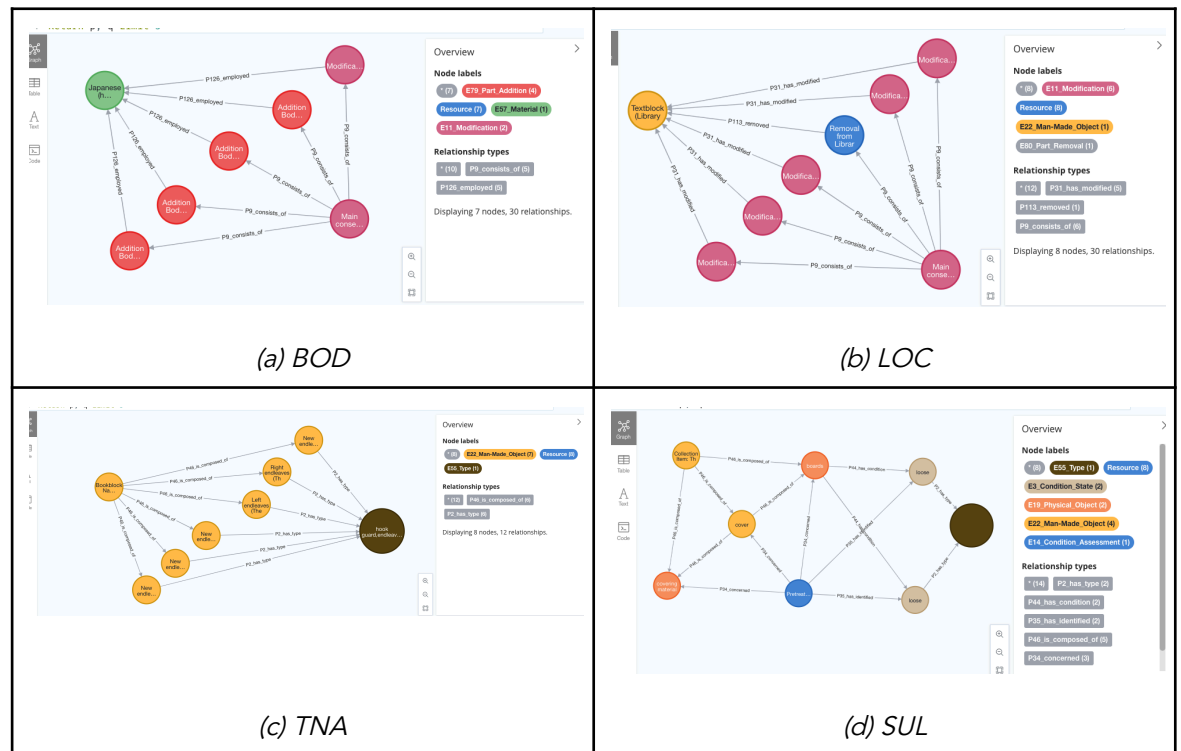































Figure 6.10 Visualisation of *m5.16* motifs (limit 5) from each LCD dataset.

The motif queries in this study were undirected with no limits placed on which node labels were relevant nor on which positions node labels can be found in the motif structure. Future work can specify further node labels and positions to constrain or differentiate structures.

Table 6.3.8 Undirected Motifs Frequency Results for the Linked Conservation Data Group

Motif	LCD-BOD	LCD-LOC	LCD-TNA	LCD-SUL
 m3.1	152,144	90,396	158,304	156,452
 m3.2	840	414	882	6,600
 m4.1	870,456	618,124	882,298	1,070,632
 m4.2	7,166,172	3,021,888	9,546,456	6,627,762
 m4.3	6,816	2,432	5,186	45,358
 m4.4	78,936	60,136	43,920	83,672
 m4.5	40	0	0	7,476
 m4.6	0	0	0	2,496
 m5.1	0	0	0	7,872
 m5.2	782	0	0	39,114
 m5.3	200	0	0	47,540
 m5.4	0	0	0	5,144
 m5.5	0	0	0	5,144
 m5.6	0	0	0	1,464
 m5.7	260	0	0	9,732
 m5.8	0	0	0	624
 m5.9	69,178	40,520	89,568	415,020
 m5.10	32,594	8,966	16,968	259,832
 m5.11	976	288	920	38,256
 m5.12	168,832	22,960	62,244	510,204
 m5.13	1,782	312	896	27,328
 m5.14	0	0	0	8,784
 m5.15	2,046,642	1,227,318	1,055,444	1,485,738
 m5.16	856,284	675,168	607,596	732,216
 m5.17	9,736,142	5,973,822	8,031,906	10,053,802
 m5.18	497,568,360	144,846,528	817,006,008	439,049,520
 m5.19	22,357,694	13,259,724	28,022,138	29,767,890
 m5.20	23,400	3,440	7,670	88,260
 m5.21	0	0	0	0

Frequency of Motif Patterns Across LCD Group

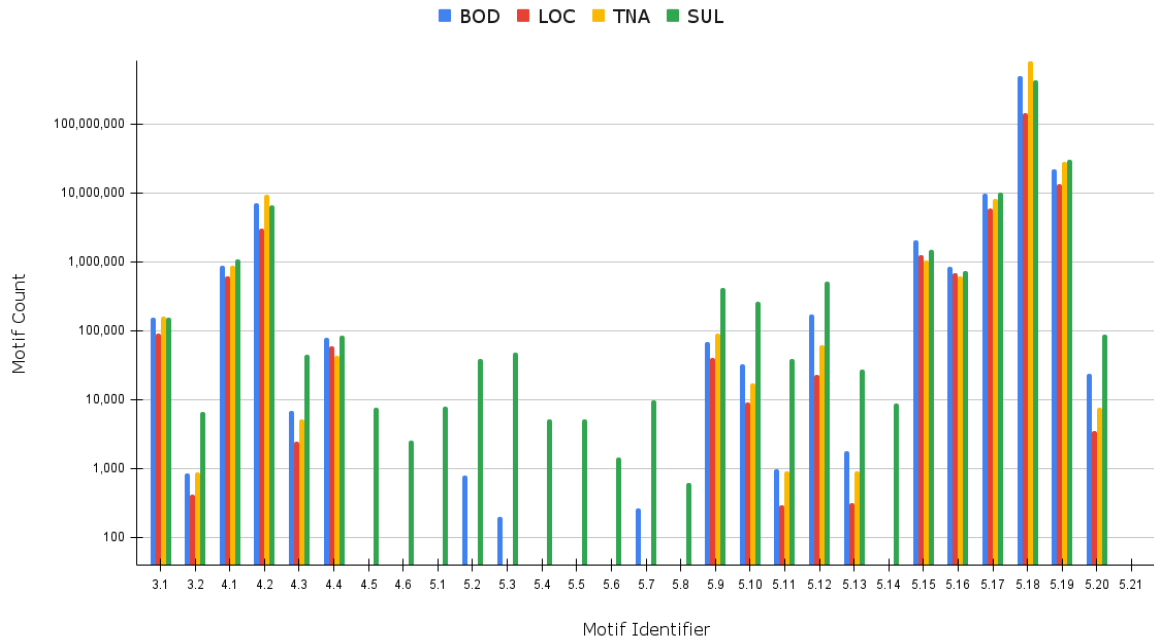


Figure 6.11. Bar graph of results from table 6.3.8 in motif identifier order after Abuoda et al 2020.

Frequency of Motif Patterns Across LCD Group

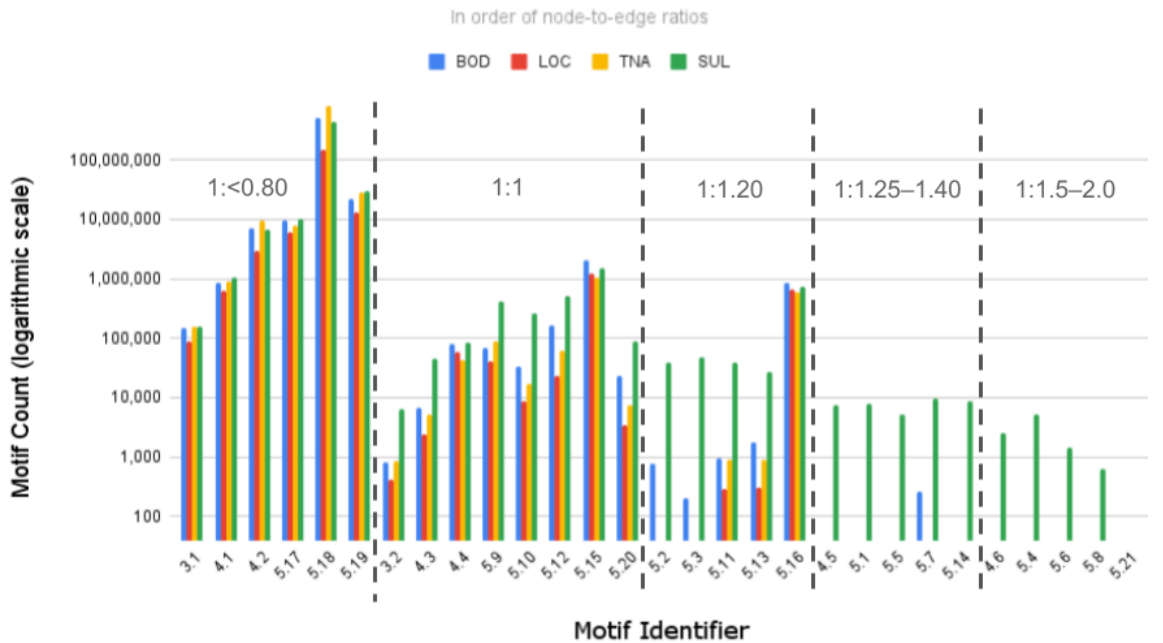


Figure 6.12. Bar graph of results from Table 6.3.9 which re-orders the results in line with ascending node-to-edge ratios of each $k=3,4,5$ motif.

Table 6.3.9 LCD Group Motif Node:Edge Ratios*

node count, k=	edge count	ratio	motif identifier	BOD	LOC	TNA	SUL
3	2	1:0.66	3.1	152,144	90,396	158,304	156,452
4	3	1:0.75	4.1	870,456	618,124	882,298	1,070,632
4	3	1:0.75	4.2	7,166,172	3,021,888	9,546,456	6,627,762
5	4	1:0.80	5.17	9,736,142	5,973,822	8,031,906	10,053,802
5	4	1:0.80	5.18	497,568,360	144,846,528	817,006,008	439,049,520
5	4	1:0.80	5.19	22,357,694	13,259,724	28,022,138	29,767,890
3	3	1:1	3.2	840	414	882	6,600
4	4	1:1	4.3	6,816	2,432	5,186	45,358
4	4	1:1	4.4	78,936	60,136	43,920	83,672
5	5	1:1	5.9	69,178	40,520	89,568	415,020
5	5	1:1	5.10	32,594	8,966	16,968	259,832
5	5	1:1	5.12	168,832	22,960	62,244	510,204
5	5	1:1	5.15	2,046,642	1,227,318	1,055,444	1,485,738
5	5	1:1	5.20	23,400	3,440	7,670	88,260
5	6	1:1.20	5.2	782	0	0	39,114
5	6	1:1.20	5.3	200	0	0	47,540
5	6	1:1.20	5.11	976	288	920	38,256
5	6	1:1.20	5.13	1,782	312	896	27,328
5	6	1:1.20	5.16	856,284	675,168	607,596	732,216
4	5	1:1.25	4.5	40	0	0	7,476
5	7	1:1.40	5.1	0	0	0	7,872
5	7	1:1.40	5.5	0	0	0	5,144
5	7	1:1.40	5.7	260	0	0	9,732
5	7	1:1.40	5.14	0	0	0	8,784
4	6	1:1.50	4.6	0	0	0	2,496
5	8	1:1.60	5.4	0	0	0	5,144
5	8	1:1.60	5.6	0	0	0	1,464
5	9	1:1.80	5.8	0	0	0	624
5	10	1:2	5.21	0	0	0	0

6.3.7 Eigenvector Centrality

Table 6.3.10. Eigenvector Centrality Results for Linked Conservation Data Group

Eigenvector Centrality - Linked Conservation Data (LCD) Group (highest scores)					
	<i>projection</i>	<i>node id</i>	<i>n.label</i>	<i>node Label/CRM Entity</i>	<i>score</i>
LCD-BOD	default	1437	["ply"]	["Resource", "E58_Measurement_Unit"]	0.88725
	undirected	1117	["corners"]	["Resource", "E55_Type"]	0.36571
LCD-LOC	default	520	["damaged"]	["Resource", "E55_Type"]	0.80108
	undirected	1077	["conservation (process)"]	["Resource", "E55_Type"]	0.30754
LCD-TNA	default	1564	["damaged"]	["Resource", "E55_Type"]	0.53724
	undirected	651	["Kew (place)"]	["Resource", "E53_Place"]	0.29221
LCD-SUL	default	299	null*	["Resource", "E55_Type"]	0.68190
	undirected	1694	["board reattachment"]	["Resource", "E55_Type"]	0.34836

*This E55_Type node does not have a label property, only a uri property, however it can be inferred by 53 incoming [:P2_has_type] relationships from (:E3_Condition_State{label:"deterioration"}) nodes that its label should have been "deterioration".

The eigenvector centrality results show that the most influential node in these datasets tends to be an E55_Type node, specifically, a type of deterioration. This aligns with expectations as being datasets of conservation treatments, it stands to reason that much of the recorded activities and observations in the LCD datasets are a result of detrimental adverse condition states related to collection materials. The default projection measures have higher scores than undirected projection measures and therefore are likely to represent the influence of deterioration upon the network more accurately while the results of the undirected projections appear to be general nodes that are meant to contextualise the dataset itself: LOC's "conservation process", TNA's "Kew (place)", and SUL's "board reattachment".

However, BOD's results were unusual and did not follow this pattern. Follow-up investigations into the E58_Measurement_Unit "ply" node in the BOD dataset revealed a modelling error. The "ply" node is highlighted in Figure 6.13 as the blue hub node to the star schema component on the left. It is surrounded by E54_Dimension nodes with numeric values, that is, these identify how many "ply" per instance. However, these instances stem from the green E57_Material node for "cord", the hub node for the component on the right of the image. The "cord" node is surrounded in star schema by

E22_Man-Made_Object nodes and treatment/sub-treatment nodes (i.e. E11_Modification, E79_Part_Addition, and E12_Production). The error here is the modelling of instance-related attributes of each cord object to the cord type node. While it is reasonable that types of things can have specific dimensional attributes, for example, Olympic-sized swimming pools are by definition 50 meters in length, the 10 (:E54_Dimension) nodes with varying values and their [:P43_has_dimension] relationships from (E57_Material{label:"cord"}) do not confer the same type-specific semantic usage when there are more than one such attribute and can be misconstrued as values related to the number of "ply" in instances of "cords".

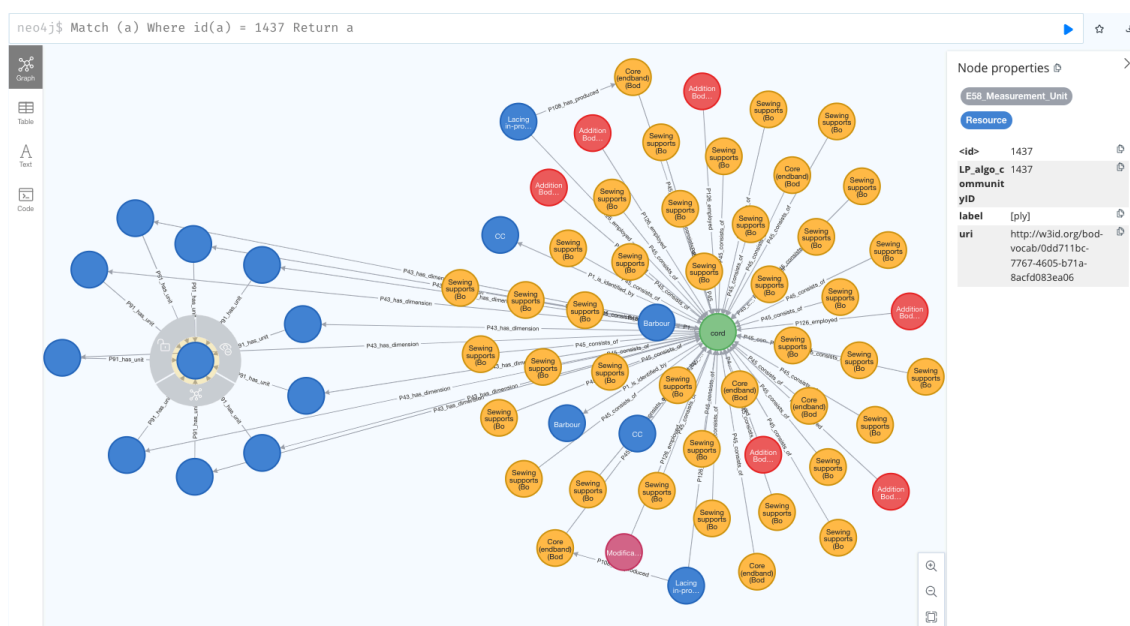


Figure 6.13 The blue hub node on the left, haloed in gray, is the node for "ply" in the BOD dataset. It is surrounded by E54_Dimension nodes which have incoming P43_has_dimension relationships from the green E57_Material "cord" node.

The scope notes for E57_Material state:

This type is used categorically in the model without reference to instances of it, i.e., the Model does not foresee the description of instances of instances of E57 Material, e.g.: "instances of gold".

This poses a challenge for modelling conservation contexts where there are specific instances of materials and these instances can be defined by physical dimension and there are also categories of materials that can also be defined by their physical dimension. The thing itself and the category of the thing are conceptually different. The representation of the physical material can have variable instance-dependent

dimension(s) whereas the representation of the categorical material type may only have dimension that contributes to the type's categorical definition.

Therefore, were the various quantities of "ply" related to instances, then an alternative way to model this would be to have the P43_has_dimension relationships connect to instances of "cord" classed as E19_Physical_Object (e.g. yellow nodes) and the E19 objects have "cord" as their type (e.g. a single green node). Another alternative model would be to model several E57_Material "cord" nodes and they each point to their "ply" dimension, thereby preserving the categorical definition of "3-ply cord", "10-ply cord", etc. The model as it currently stands obscures which modification of an object employed which cord type while distorting and over-emphasising the transitive influence of "ply" and "ply" dimensionality on the rest of the network.

However, the error in the BOD model and dataset were not corrected for the purposes of this study. Instead, the eigenvector centrality algorithm was re-ran on a graph projection that explicitly excluded P43_has_dimension relationships. This modified examination found the resulting top scorer to be an E55_Type node like the other results that describe an E3_Condition_State, specifically the "repaired" type with a score of 0.46166. The algorithm was also re-run excluding P43_has_dimension using an undirected projection. However, this attempt was unaffected by the exclusion and still produced the same result with "corners" (E55_Type), a type node for categorising E53_Place.

This section on the results of graph theoretic analyses has demonstrated how measures can highlight patterns within the datasets in a data agnostic manner, that is, while the inherent data content contributed to these structural and graph theoretic patterns, it was not necessary to be familiar with the data or its semantics to begin to leverage these measurable relationships and patterns to examine the networks captured. By contrast, the next section will present a more in-depth look into the data content of each dataset using search and filter queries as a means of analysis.

6.4 Query-based Analysis and Inference

This section will demonstrate how data content can be explored and analysed using search and filter queries. Only one RETURN clause can follow a query, however, alternative RETURN clauses are provided in the appendix and in some of the examples to follow to demonstrate the different ways to extract, display, filter, and sort through results. Cypher queries for this chapter can be found in Appendix G and at <https://github.com/ana-tam/conservation-graphs/>.

6.4.1 Graph Exploration

In order to extract useful content from a graph of unknown data, exploration is a key first step to gain a sense of the general elements constituting the graph. As Cypher uses a graph-based syntax, knowledge of the general elements, such as node labels, node property keys, and relationship types, are needed to compose the Cypher queries. A list of these exploration queries can be found in Appendix A and the Github repository.

Table 6.4.1 and 6.4.2 below show the node labels and relationship types for each LCD dataset and how many times (in count and as percentage) they appear. The node label and relationship type profiles provide four immediate insights into the datasets in this case. Firstly, the relationship type percentage profiles reveal that approximately one-third of each LCD dataset consists of “P2_has_type” relationships where instances are explicitly matched with their corresponding categorical representations (i.e. E55_Type or E57_Material). Secondly, it also shows the presence of “E22_Man-Made_Object” across all four LCD datasets which indicate that the modellers applied a version of the CIDOC CRM earlier than v.7.1.1 as by v.7.1.1 the name of the class was updated to “E22_Human-Made_Object”. Thirdly, E11_Modification is the class for treatment events and other conservation activity, as the E11_Modification scope note states:

This class includes the production of an item from raw materials, and other so far undocumented objects, and the preventive treatment or restoration of an object for conservation.

Table 6.4.1 Count and Percentage of Nodes by Label per LCD Dataset

no.	BOD	ct	%	LOC	ct	%	TNA	ct	%	SUL	ct	%
1	["E22_Man-Made_Object"]	561	22.91%	["E53_Place"]	339	19.87%	["E22_Man-Made_Object"]	547	25.83%	["E22_Man-Made_Object"]	416	18.76%
2	["E53_Place"]	518	21.15%	["E22_Man-Made_Object"]	339	19.87%	["E53_Place"]	413	19.50%	["E53_Place"]	318	14.34%
3	["E11_Modification"]	231	9.43%	["E11_Modification"]	273	16.00%	["E11_Modification"]	376	17.75%	["E3_Condition_State"]	255	11.50%
4	["E55_Type"]	212	8.66%	["E52_Time-Span"]	178	10.43%	["E3_Condition_State"]	231	10.91%	["E79_Part_Addition"]	176	7.94%
5	["E3_Condition_State"]	206	8.41%	["E3_Condition_State"]	155	9.09%	["E79_Part_Addition"]	103	4.86%	["E12_Production"]	161	7.26%
6	["E79_Part_Addition"]	102	4.16%	["E25_Man-Made_Feature"]	69	4.04%	["E52_Time-Span"]	90	4.25%	["E52_Time-Span"]	158	7.12%
7	["E52_Time-Span"]	87	3.55%	["E55_Type"]	65	3.81%	["E57_Material"]	68	3.21%	["E42_Identifier"]	126	5.68%
8	["E12_Production"]	79	3.23%	["E57_Material"]	51	2.99%	["E55_Type"]	60	2.83%	["E22_Man-Made_Object", "E19_Physical_Object"]	122	5.50%
9	["E54_Dimension"]	79	3.23%	["E39_Actor"]	49	2.87%	["E31_Document"]	45	2.12%	["E35_Title"]	104	4.69%
10	["E80_Part_Removal"]	56	2.29%	["E79_Part_Addition"]	40	2.34%	["E42_Identifier"]	45	2.12%	["E11_Modification"]	97	4.37%
11	["E41_Appellation"]	52	2.12%	["E31_Document"]	37	2.17%	["E25_Man-Made_Feature"]	45	2.12%	["E55_Type"]	63	2.84%
12	["E29_Design_or_Procedure"]	46	1.88%	["E13_Attribute_Assignment"]	37	2.17%	["E13_Attribute_Assignment"]	45	2.12%	["E14_Condition_Assessment"]	52	2.34%
13	["E57_Material"]	44	1.80%	["E42_Identifier"]	37	2.17%	["E39_Actor"]	32	1.51%	["E13_Attribute_Assignment"]	52	2.34%
14	["E25_Man-Made_Feature"]	36	1.47%	["E12_Production"]	30	1.76%	["E80_Part_Removal"]	16	0.76%	["E31_Document"]	52	2.34%
15	["E39_Actor"]	27	1.10%	["E80_Part_Removal"]	5	0.29%	["E40_Legal_Body"]	1	0.05%	["E7_Activity"]	52	2.34%
16	["E42_Identifier"]	24	0.98%	["E40_Legal_Body"]	1	0.06%	["E41_Appellation"]	1	0.05%	["E57_Material"]	11	0.50%
17	["E31_Document"]	24	0.98%	["E41_Appellation"]	1	0.06%				["E29_Design_or_Procedure"]	2	0.09%
18	["E13_Attribute_Assignment"]	24	0.98%							["E89_Propositional_Object"]	1	0.05%
19	["S10_Material_Substantial"]	13	0.53%									
20	["E26_Physical_Feature"]	10	0.41%									
21	["E58_Measurement_Unit"]	5	0.20%									
22	["E60_Number"]	5	0.20%									
23	["S11_Amount_of_Matter"]	3	0.12%									
24	["E55_Type", "E57_Material"]	2	0.08%									
25	["E73_Information_Object"]	2	0.08%									
26	["E40_Legal_Body"]	1	0.04%									
	Total Node Counts	2449		Total Node Counts	1706		Total Node Counts	2118		Total Node Counts	2218	

Table 6.4.2 Relationships in Order of Frequency Count and Percentage of LCD Dataset

no.	BOD	ct	%	LOC	ct	%	TNA	ct	%	SUL	ct	%
1	"P2_has_type"	1787	32.60%	"P2_has_type"	988	27.36%	"P2_has_type"	1365	29.60%	"P2_has_type"	1578	27.43%
2	"P46_is_composed_of"	524	9.56%	"P46_is_composed_of"	302	8.36%	"P46_is_composed_of"	496	10.76%	"P7_took_place_at"	522	9.07%
3	"P59_has_section"	382	6.97%	"P59_has_section"	286	7.92%	"P9_consists_of"	449	9.74%	"P46_is_composed_of"	436	7.58%
4	"P126_employed"	367	6.70%	"P32_used_general_technique"	280	7.75%	"P59_has_section"	382	8.28%	"P59_has_section"	342	5.94%
5	"P9_consists_of"	363	6.62%	"P9_consists_of"	280	7.75%	"P31_has_modified"	376	8.15%	"P35_has_identified"	255	4.43%
6	"P32_used_general_technique"	238	4.34%	"P31_has_modified"	273	7.56%	"P32_used_general_technique"	349	7.57%	"P9_consists_of"	251	4.36%
7	"P31_has_modified"	231	4.21%	"P126_employed"	239	6.62%	"P44_has_condition"	231	5.01%	"P44_has_condition"	242	4.21%
8	"P44_has_condition"	216	3.94%	"P44_has_condition"	155	4.29%	"P126_employed"	231	5.01%	"P111_added"	234	4.07%
9	"P55_has_current_location"	201	3.67%	"P55_has_current_location"	105	2.91%	"P55_has_current_location"	191	4.14%	"P126_employed"	230	4.00%
10	"P45_consists_of"	165	3.01%	"P4_has_time-span"	104	2.88%	"P111_added"	103	2.23%	"P4_has_time-span"	208	3.62%
11	"P8_took_place_on_or_within"	132	2.41%	"P45_consists_of"	87	2.41%	"P70_documents"	51	1.11%	"P34_concerned"	201	3.49%
12	"P111_added"	102	1.86%	"P86_falls_within"	74	2.05%	"P1_is_identified_by"	46	1.00%	"P1_is_identified_by"	198	3.44%
13	"P108_has_produced"	78	1.42%	"P56_bears_feature"	69	1.91%	"P50_has_current_keeper"	45	0.98%	"P108_has_produced"	161	2.80%
14	"P1_is_identified_by"	76	1.39%	"P8_took_place_on_or_within"	52	1.44%	"P4_has_time-span"	45	0.98%	"P31_has_modified"	146	2.54%
15	"P91_has_unit"	65	1.19%	"P14_carried_out_by"	49	1.36%	"P56_bears_feature"	45	0.98%	"P134_continued"	104	1.81%
16	"P43_has_dimension"	65	1.19%	"P14.1_in_the_role_of"	49	1.36%	"P86_falls_within"	45	0.98%	"P55_has_current_location"	100	1.74%
17	"P4_has_time-span"	62	1.13%	"P111_added"	40	1.11%	"P140_assigned_attribute_to"	45	0.98%	"P32_used_general_technique"	93	1.62%
18	"P113_removed"	60	1.09%	"P1_is_identified_by"	38	1.05%	"P14.1_in_the_role_of"	32	0.69%	"P110_augmented"	68	1.18%
19	"P56_bears_feature"	46	0.84%	"P70_documents"	37	1.02%	"P14_carried_out_by"	32	0.69%	"P160_has_temporal_projection"	54	0.94%
20	"P33_used_specific_technique"	46	0.84%	"P140_assigned_attribute_to"	37	1.02%	"P8_took_place_on_or_within"	24	0.52%	"P140_assigned_attribute_to"	54	0.94%
21	"P70_documents"	33	0.60%	"P50_has_current_keeper"	37	1.02%	"P113_removed"	16	0.35%	"P70_documents"	52	0.90%
22	"P16_used_specific_object"	30	0.55%	"P108_has_produced"	30	0.83%	"P156_occupies"	6	0.13%	"P86_falls_within"	52	0.90%
23	"P14_carried_out_by"	27	0.49%	"P113_removed"	5	0.14%	"P45_consists_of"	5	0.11%	"P148_has_component"	52	0.90%
24	"P14.1_in_the_role_of"	27	0.49%	"P183_ends_before_the_start_of"	1	0.03%	"P183_ends_before_the_start_of"	1	0.02%	"P33_used_specific_technique"	48	0.83%

no.	BOD	ct	%	LOC	ct	%	TNA	ct	%	SUL	ct	%
25	"P86_falls_within"	25	0.46%							"P45_consists_of"	38	0.66%
26	"P50_has_current_keeper"	24	0.44%							"P48_has_preferred_identifier"	34	0.59%
27	"P89_falls_within"	24	0.44%									
28	"P140_assigned_attribute_to"	24	0.44%									
29	"O25_contains"	21	0.38%									
30	"O12_has_dimension"	14	0.26%									
31	"P156_occupies"	9	0.16%									
32	"P183_ends_before_the_start_of"	8	0.15%									
33	"P57_has_number_of_parts"	5	0.09%									
34	"P7_took_place_at"	2	0.04%									
35	"P129_is_about"	2	0.04%									
36												
37	Total Relationship Counts	5481		Total Relationship Counts	3611		Total Relationship Counts	4611		Total Relationship Counts	5753	

Table 6.4.3. Colour-code key for Table 6.4.1. and Table 6.4.2

Node Table Key	Relationship Table Key
Converted leaf nodes with literals as property values	Property of a property used as edge
Types classes	Properties with type classes as objects
Class from a CRM extension	Property from a CRM extension

The use of E11_Modification as the primary conservation event representation was further confirmed by consulting LCD project output literature (Moraitou and Christodoulou 2021). E11_Modification is also the superclass for E12_Production, E79_Part_Addition, and E80_Part_Removal which can also be found applied across the LCD datasets and feature in Table 6.4.1

Finally, of the four datasets, the BOD graph has more node labels and relationship types than the other three datasets including classes and properties outside of the core CIDOC CRM. Two classes, "S10_Material_Substantial" and "S11_Amount_of_Matter", and their related CRM properties, "O25_contains" and "O12_has_dimension" belong to the CRM_{sci} extension. The BOD graph also has the greatest number of node labels representing less than 2% each of the graph and greatest number of relationship types representing less than 1% each of the graph. As general node and edge counts across the LCD datasets are broadly similar, this distribution of the elements of BOD across many more labels and types aligns with the higher diameter for BOD. That is, imagine traversing the graph as a map and the BOD map has many roads with only one or two houses and these roads themselves are not directly connected to the roads found in the denser "high street" areas of the map. Therefore, it is not surprising that there is only a limited number of routes to travel from one of these remote properties to another remote property with no shortcuts, hence the longer diameter.

By knowing which nodes these more remote ones are, it is possible to discount them or exclude them when running future analyses. For example, the single E40_Legal_Body node represents the Bodleian Library itself within the graph. Therefore, when conducting further analyses of treatment-specific patterns, this node can be excluded from the parameters (while the data remains in the database). Of course, if the BOD dataset was expanded and further data was added so that more E40_Legal_Body nodes were present in the graph, for example, if loan data or other accession data were added, this would allow for different types of analytical queries to be run, such as to visualise a graph representation of legal entities from a specific time in the past until now. Hence, an initial graph theoretic inspection of the graphs coupled with follow-up queries can reveal comprehensive insights into not only the nature of the data content but also on how it has been modelled and organised. Such insights can be used to fine-tune further analyses.

Not only do queries handle data retrieval, they are also used to perform filtering functions. Table 6.4.4 shows the list of distinct node labels (i.e. CRM classes) in each LCD dataset that are also leaf nodes. To clarify, this does not mean all nodes with these

node labels are leaf nodes. For example, the E55_Type nodes that happen to be leaf nodes may be a rare or unique type only referred to once in one treatment found in the database. It does not mean all E55_Type nodes are positioned at the periphery of the graph. E31_Document and E41_Appellation also make sense as potential leaf nodes as they can each represent a document whose content is very specific, for example, a report that only pertains to a unique event or single object. However, likewise, there can be documents and appellations with connections to many instances within the graph. Those nodes that have literal-based node properties are those most expected to be leaf nodes, e.g. E52_Time-Span, E54_Dimension.

Table 6.4.4. List of Distinct Node Labels for Leaf Nodes for each LCD Dataset

no.	BOD	LOC	TNA	SUL
1	["E52_Time-Span"]	["E31_Document"]	["E31_Document"]	["E12_Production"]
2	["E57_Material"]	["E52_Time-Span"]	["E57_Material"]	["E52_Time-Span"]
3	["E55_Type"]	["E55_Type"]	["E52_Time-Span"]	["E35_Title"]
4	["E41_Appellation"]	["E42_Identifier"]	["E55_Type"]	["E55_Type"]
5	["E31_Document"]	["E57_Material"]	["E41_Appellation"]	["E57_Material"]
6	["E58_Measurement_Unit"]	["E41_Appellation"]		
7	["E54_Dimension"]			
8	["E60_Number"]			

It was also observed across the BOD, LOC, and TNA datasets that the modellers used strings of composite values for rdfs:labels, often consisting of a short descriptive phrase, followed by the name of the institution, and then by the collection or object number identifier. It is likely the modellers wanted to ensure the content remained human-readable when working and debugging directly in RDF/XML format and were concerned about disassociating the semantic content if they were to decompose the data further. However, while this aids human-readability, it complicates querying as a direct match requires the full exact string. The composite labeling approach has consequences such as requiring additional lines of code to parse the label before pattern matching in a search or filtering query which has knock-on effects on computational efficiency such as increasing the time it takes to process the longer query.

LCD-LOC example:

```
<http://www.ligatus.org.uk/lcd/39fab89e-cbc7-4397-ae33-3c2202eb9323> a crm:E12_Production ;  
  rdfs:label "Book production (Library of Congress, 4240)"@en ;
```

LCD-TNA example:

```
<http://www.ligatus.org.uk/lcd/010e55a2-6e66-4dd3-92fc-c1081830aef0/> a crm:E11_Modification  
 ;  
  rdfs:label "Main conservation event (The National Archives, HCA 13/78)"@en ;
```

LCD-SUL example:

```
<http://w3id.org/sul-data/2699e737-a2c7-4517-857e-d37df14175b0>  
  a <http://www.cidoc-crm.org/cidoc-crm/E11_Modification> ;  
  <http://www.w3.org/2000/01/rdf-schema#label>  
    "treatment steps"@en , "Treatment Steps"@en ;
```

Figure 6.14. Examples from LOC and TNA original .trig files exhibiting encodings of rdfs:label with composite value labels although SUL does not exhibit this practice.

When presented with such composite labels, using regular expressions (in Cypher and SPARQL) for pattern matching text is the most straightforward workaround. Other options in Cypher include using nested queries with `split()` or other functions to parse text. For example, Table 6.4.5 lists all nodes with the partial label of "4240" in the LOC database which has been returned using the following Cypher query:

```
MATCH (a)  
UNWIND a.label as item  
WITH item, a  
WHERE item =~ "(?i).*4240.*"  
RETURN item, labels(a) as crm
```

This has been the most direct way to query on collection identifier numbers due to the composite labels. The 40 results returned by the query in Table 6.4.5 match the 40 instances that "Library of Congress, 4240" occurs in the original LOC TriG file.

The next section will apply path queries to find typical items of conservation interest, namely collection objects, materials and familiar terms (i.e. as `E55_Type`).

Table 6.4.5 Example of composite string values for *rdfs:labels* and their classes

	item	crm
1	"Main conservation event (Library of Congress, 4240)"	["Resource", "E11_Modification"]
2	"Modification of Library of Congress, 4240"	["Resource", "E11_Modification"]
3	"Modification of Library of Congress, 4240"	["Resource", "E11_Modification"]
4	"Modification of Library of Congress, 4240"	["Resource", "E11_Modification"]
5	"Modification of Library of Congress, 4240"	["Resource", "E11_Modification"]
6	"Modification of Library of Congress, 4240"	["Resource", "E11_Modification"]
7	"Book production (Library of Congress, 4240)"	["Resource", "E12_Production"]
8	"Conservation assessment (Library of Congress, 4240)"	["Resource", "E13_Attribute_Assignment"]
9	"Textblock (Library of Congress, 4240)"	["Resource", "E22_Man-Made_Object"]
10	"Book (Library of Congress, 4240)"	["Resource", "E22_Man-Made_Object"]
11	"Bookblock (Library of Congress, 4240)"	["Resource", "E22_Man-Made_Object"]
12	"Left board (Library of Congress, 4240)"	["Resource", "E22_Man-Made_Object"]
13	"Right board (Library of Congress, 4240)"	["Resource", "E22_Man-Made_Object"]
14	"Sewing supports (Library of Congress, 4240)"	["Resource", "E22_Man-Made_Object"]
15	"Sewing structure (Library of Congress, 4240)"	["Resource", "E22_Man-Made_Object"]
16	"Cover (Library of Congress, 4240)"	["Resource", "E22_Man-Made_Object"]
17	"Bookblock edges (Library of Congress, 4240)"	["Resource", "E25_Man-Made_Feature"]
18	"Bookblock spine (Library of Congress, 4240)"	["Resource", "E25_Man-Made_Feature"]
19	"Gold tooling (Library of Congress, 4240)"	["Resource", "E25_Man-Made_Feature"]
20	"Conservation Report (Library of Congress, 4240)"	["Resource", "E31_Document"]
21	"Detached (Library of Congress, 4240)"	["Resource", "E3_Condition_State"]
22	"Broken (Library of Congress, 4240)"	["Resource", "E3_Condition_State"]
23	"Detached (Library of Congress, 4240)"	["Resource", "E3_Condition_State"]
24	"Broken (Library of Congress, 4240)"	["Resource", "E3_Condition_State"]
25	"Damaged (Library of Congress, 4240)"	["Resource", "E3_Condition_State"]
26	"Project Number (Library of Congress, 4240)"	["Resource", "E42_Identifier"]
27	"Printing or Production date (Library of Congress, 4240)"	["Resource", "E52_Time-Span"]
28	"Attribute assignment decade (Library of Congress, 4240)"	["Resource", "E52_Time-Span"]
29	"Attribute assignment date (Library of Congress, 4240)"	["Resource", "E52_Time-Span"]
30	"Conservation treatment date (Library of Congress, 4240)"	["Resource", "E52_Time-Span"]
31	"Conservation treatment decade (Library of Congress, 4240)"	["Resource", "E52_Time-Span"]
32	"Outer joints (Library of Congress, 4240)"	["Resource", "E53_Place"]
33	"Tail (Library of Congress, 4240)"	["Resource", "E53_Place"]
34	"Right (Library of Congress, 4240)"	["Resource", "E53_Place"]
35	"Left (Library of Congress, 4240)"	["Resource", "E53_Place"]
36	"Outer joints (Library of Congress, 4240)"	["Resource", "E53_Place"]
37	"Head (Library of Congress, 4240)"	["Resource", "E53_Place"]
38	"Spine (place) (Library of Congress, 4240)"	["Resource", "E53_Place"]
39	"Right (Library of Congress, 4240)"	["Resource", "E53_Place"]
40	"Left (Library of Congress, 4240)"	["Resource", "E53_Place"]

6.4.2 Analysing for Objects, Materials and Types

As the last section demonstrated, queries are the means to collect, collate and retrieve data content. Querying with visualised results provides another means for pattern recognition by a human user and can prove valuable when communicating about data.

The principal class applied across all four LCD datasets for representing objects within each institution's collection is `E22_Man-Made_Object`. Although the modelling paradigm for the CIDOC CRM is event-centric, the practical realities for conservation professionals remain strongly object-centric and therefore accessing object-related data remains a necessity. Therefore, let's begin with a collection overview for each dataset.

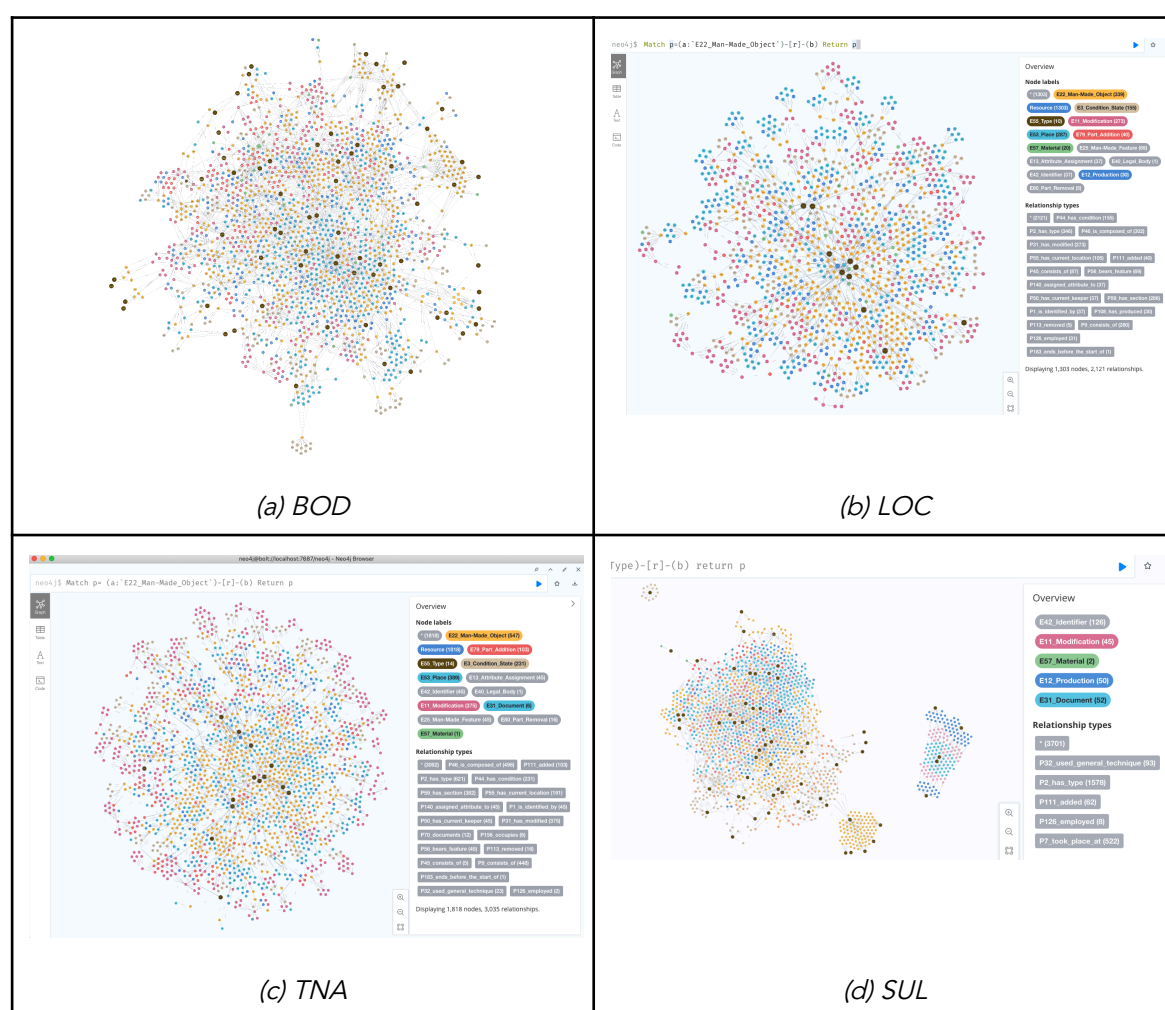


Figure 6.15. Visualisations of `E22_Man-Made_Object` nodes in each LCD dataset and their immediate neighbours.

The four images in Figure 6.12 show `E22_Man-Made_Object` (yellow) nodes in each dataset and their immediately adjacent neighbours.⁸ A colour-coded key can be found

⁸ Each visualisation shows a sample of the query results up to 2,000 nodes. This maximum view setting can be changed.

in Appendix A, however, visual representation of the datasets have already begun to take on a “fuzzy ball” appearance where it becomes too visually dense to discern meaningful patterns. The next several figures demonstrate visualisations that “peel back” on the quantity of data represented by inspecting three specific classes:

- E22_Man-Made_Object for collection objects,
- E55_Type for the categories within the dataset, and
- E57_Material, a subclass of E55_Type, for categories of materials used or encountered

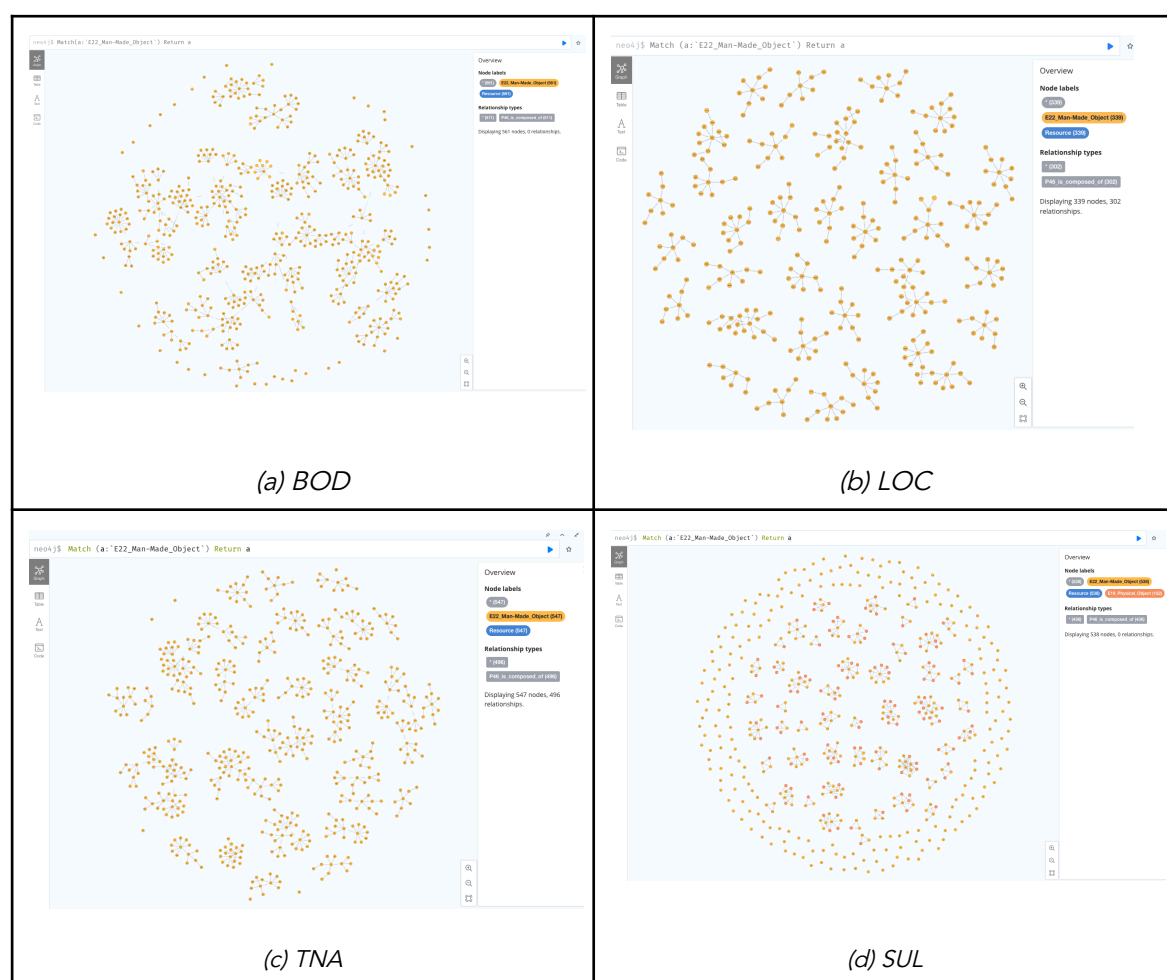


Figure 6.16. The objects graph (E22_Man-Made_Object) for each LCD dataset.

The clusters of dendritic patterns in Figure 6.16 (a) to (b) are representations of E22_Man-Made_Object nodes connected to each other via P46_is_composed_of relationships. These represent “main” objects and their many constituent parts. Figure 6.17 shows a close-up view of two star schema clusters from the LOC dataset. The “Book” nodes are the central “main object” nodes from which nodes representing parts of each book radiate out from. At this scale, each “book graph” can be explored and the conceptual decomposition of the object is made explicit. However, the

representation may not include all parts, only those that have been observed or treated, and therefore, recorded. It may not always be necessary to represent objects in the form of a graph with decomposed parts. Depending on the aims and requirements of the model, using a single node to represent each object can suffice. This would still have the potential to develop into an object graph over time if part nodes were added later, for example, if a future treatment only treated a specific area and not the whole object.

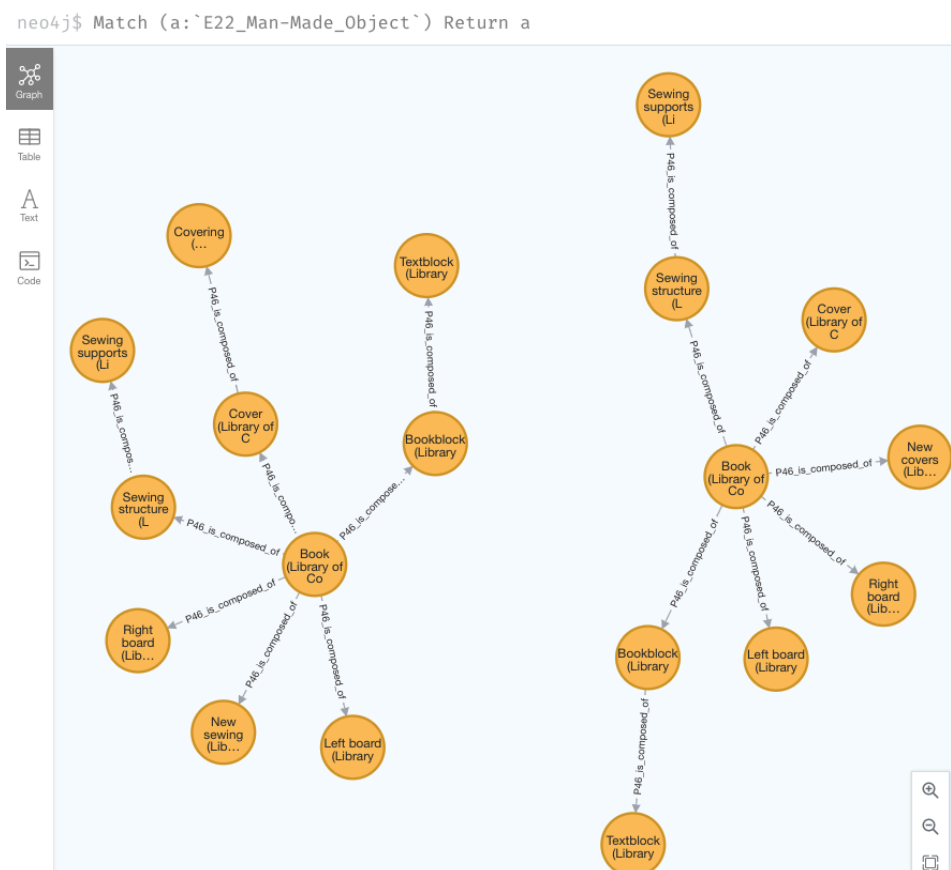


Figure 6.17. Two object graphs, each representing a book from the LOC dataset.

Similarly, Figure 6.18. shows a book graph from the TNA dataset. However, as the TNA example demonstrates, the “central” node is not always the “main” node as the “Bookblock” node in this case has more adjoining parts. The “Book” node that represents the main object is to the right of the “BookBlock” star schema representation.

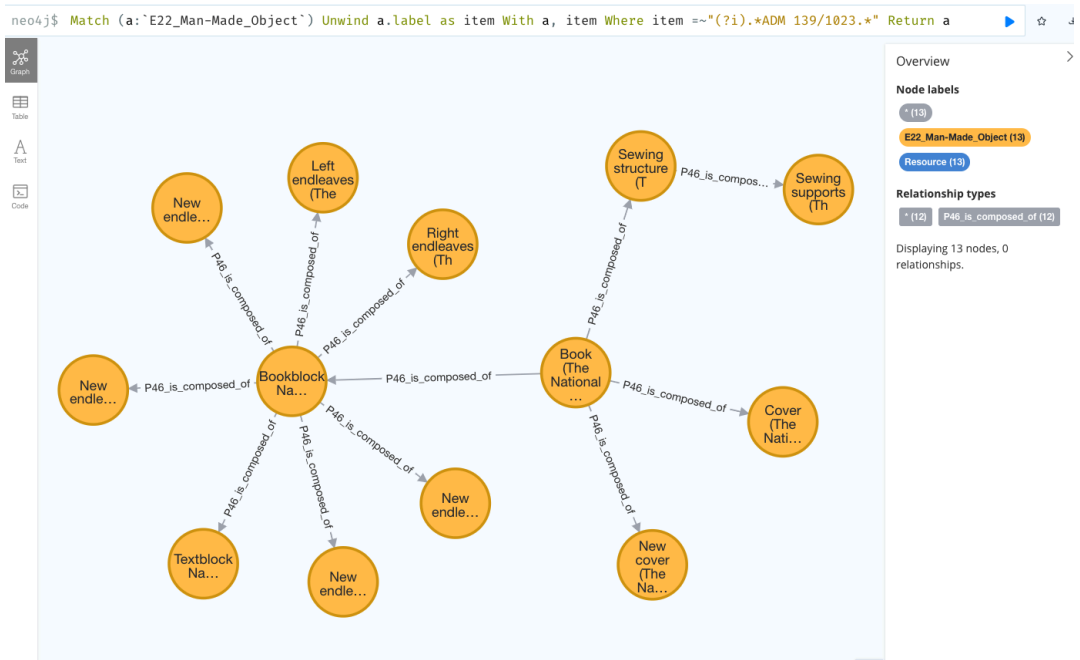


Figure 6.18. The “Book Graph” for “Book (The National Archives, ADM 139/1023)” and its parts from the TNA dataset.

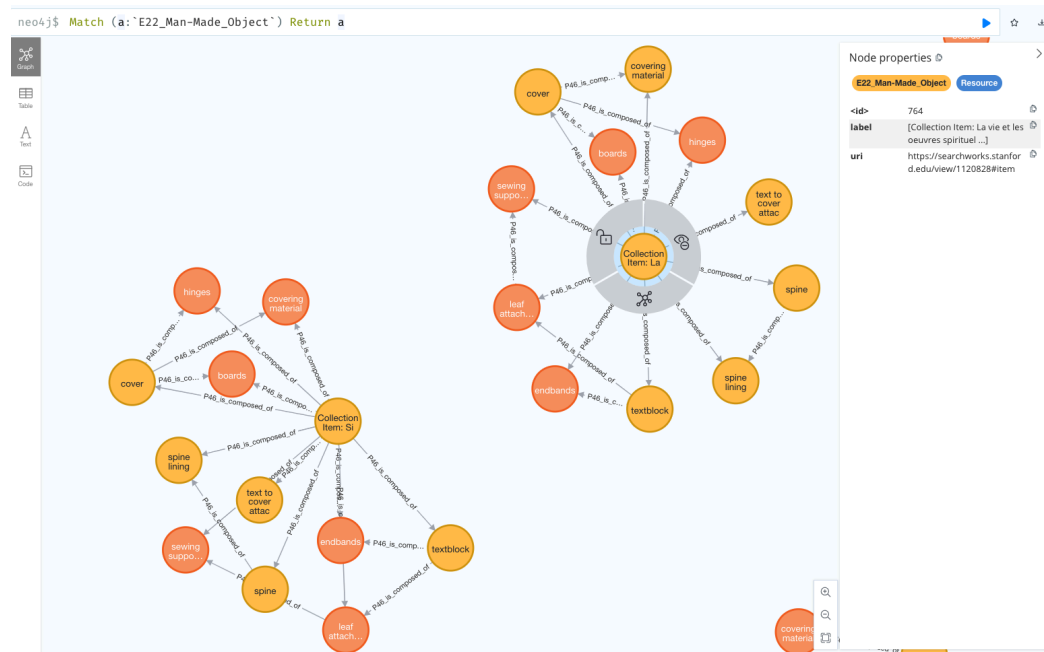


Figure 6.19. View of two SUL object graphs showing E22 and E19 nodes, the latter have been used to designate condition assessed parts. This modelling practice was not found in the other LCD datasets.

Figure 6.19 shows two object graphs from the SUL dataset. It can be observed in the SUL dataset that instead of using the word "Book" to prefix the label as can be found in the BOD, LOC and TNA datasets, it uses the phrase "Collection Item" to distinguish between the main object from its parts. It can also be observed that the SUL object graphs consist of two kinds of nodes, E22_Man-Made_Object (yellow) and E19_Physical_Object (orange). Closer inspection reveals that all E19_Physical_Object nodes have been assigned two node labels, E19 and E22. The difference between their uses in the model appears to be that those also assigned E19 are parts of the main object which have been condition assessed, i.e. have a relationship with an E3_Condition_Assessment node that specifies the assessment finding, usually a type of deterioration. This pattern of modelling was not found in the other datasets.

Unlike the E22_Man-Made_Object graphs which show discrete objects with their connected constituent parts as dendritic star schema clusters, the E57_Material nodes (in green) do not appear to have any direct relationships with other E57_Material nodes. Therefore, for more visual context, the materials graph for each dataset have been queried to return each E57_Material node and its immediate neighbor. This shows significant distinctions between each dataset and where materials fit. For example, in Figure 6.20, the BOD dataset shows E57_Material nodes distributed throughout having relationships with many different kinds of nodes, which themselves are also connected. On the other hand, LOC's material graph shows that the modelling of material types in this dataset only speaks to treatment materials (i.e. where they are connected to the pink E11_Modification nodes) as well as object materials (i.e. where green material nodes form star schema clusters with yellow E22_Man-Made_Object nodes). The TNA dataset shows an even more exclusive relationship between the material type nodes to pink E11_Modification (i.e. treatment) nodes. The SUL materials graph, Figure 6.20 (d), shows that materials are clustered mostly with the E12_Production (blue) nodes, which is a subclass of E11_Modification. The small cluster of E79_Part_Addition (red) nodes on the bottom right of image (d) is also a subclass of E11_Modification. The only type of material associated directly with objects, as depicted by the orange (E19_Physical_Object) and yellow (E22_Man-Made_Objects) nodes at the top of the graph, is "leather".

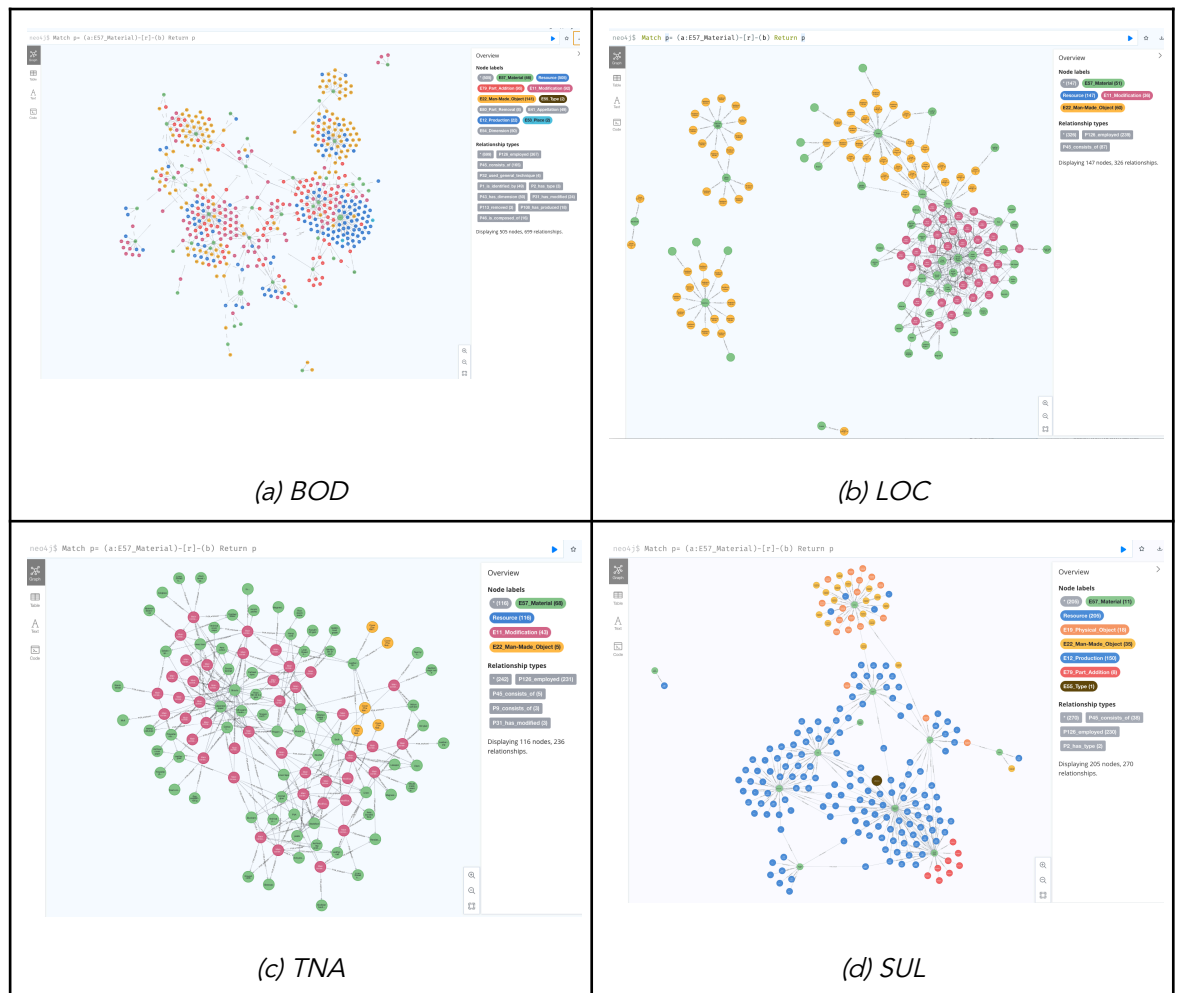


Figure 6.20 . The material graphs for each LCD dataset.

The CIDOC CRM's E55_Type class allows for the adoption of existing thesauri to provide categorical representation to any entity, whether it is conceptual, physical or temporal. Therefore, most instances can have a type. Visualising the distribution of E55_Type nodes (larger, dark brown nodes) throughout each dataset graph provides an overview of how categorical representations interact with each other. The BOD, LOC and TNA datasets all exhibit a general but irregular heterogeneous distribution. However, the SUL types appear to be more compartmentalised with four components visible and only two of these being connected. The wholly separate components include a small star schema of condition states around their shared type (upper left corner) and a much larger component (to the right) that consists of other recorded E7_Activity (light pink) and related documents and identifiers. The largest component shows type nodes distributed around in a manner similar to the other datasets while a smaller component is only connected to E22_Man-Made_Objects (bottom of image). More discerning visualisations can also be achieved if the queries were slightly modified, such as specifying a particular E55_Type or a specific neighbour node, for example.

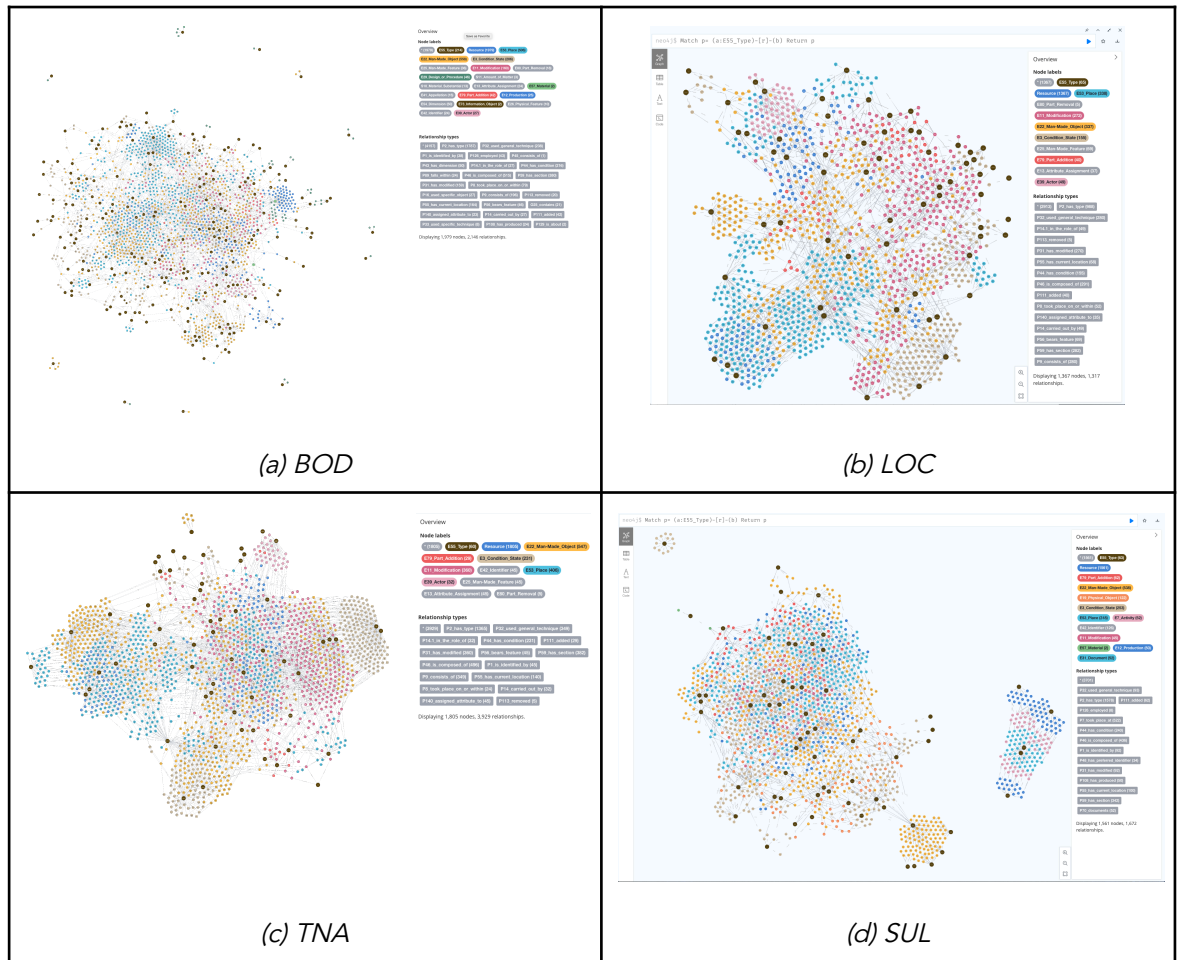


Figure 6.21. The Type graphs demonstrate the distribution of categorical type nodes in each dataset.

6.4.3 Analysing for Treatment Events

Similar to the object graphs in the last section, retrieving conservation events in overview can be achieved by querying for only E11_Modification nodes (see Figure 6.22) or E11 and their immediate neighbours (as in Figure 6.23). Once again, the BOD, LOC, and TNA datasets are very similar in their general structure. Figures 6.24 and 6.25 show close-up views of the star schema structures of LOC and TNA. Although not pictured, the BOD dataset shared in this star schema structure around a “main” conservation event node that is surrounded by sub-events such as other E11_Modification nodes, E79_Part_Addition, E80_Part_Removal or E12_Production nodes with cross-schema connections made via E55_Type and E57_Material type nodes.

The SUL dataset, once again, exhibits very different characteristics. Figure 6.23 (d) shows a uniform radial structure with edges pointing inward towards the E55_Type hub node for “reattachment”. Unlike the other datasets, there are no direct relationships (where length = 1) between E57_Material and E11_Modification in the SUL model.

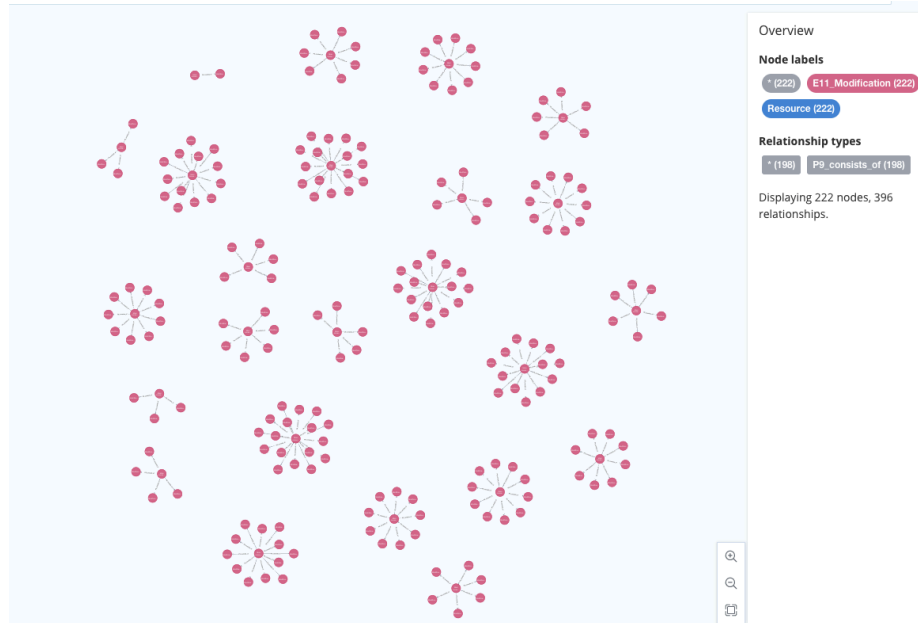


Figure 6.22. Visualisation of the treatment events in the BOD dataset where a central “main” conservation event” hub node is surrounded by other E11_Modification “sub”-events. This does not include sub-class events such as E79_Part_Addition or E12_Production.

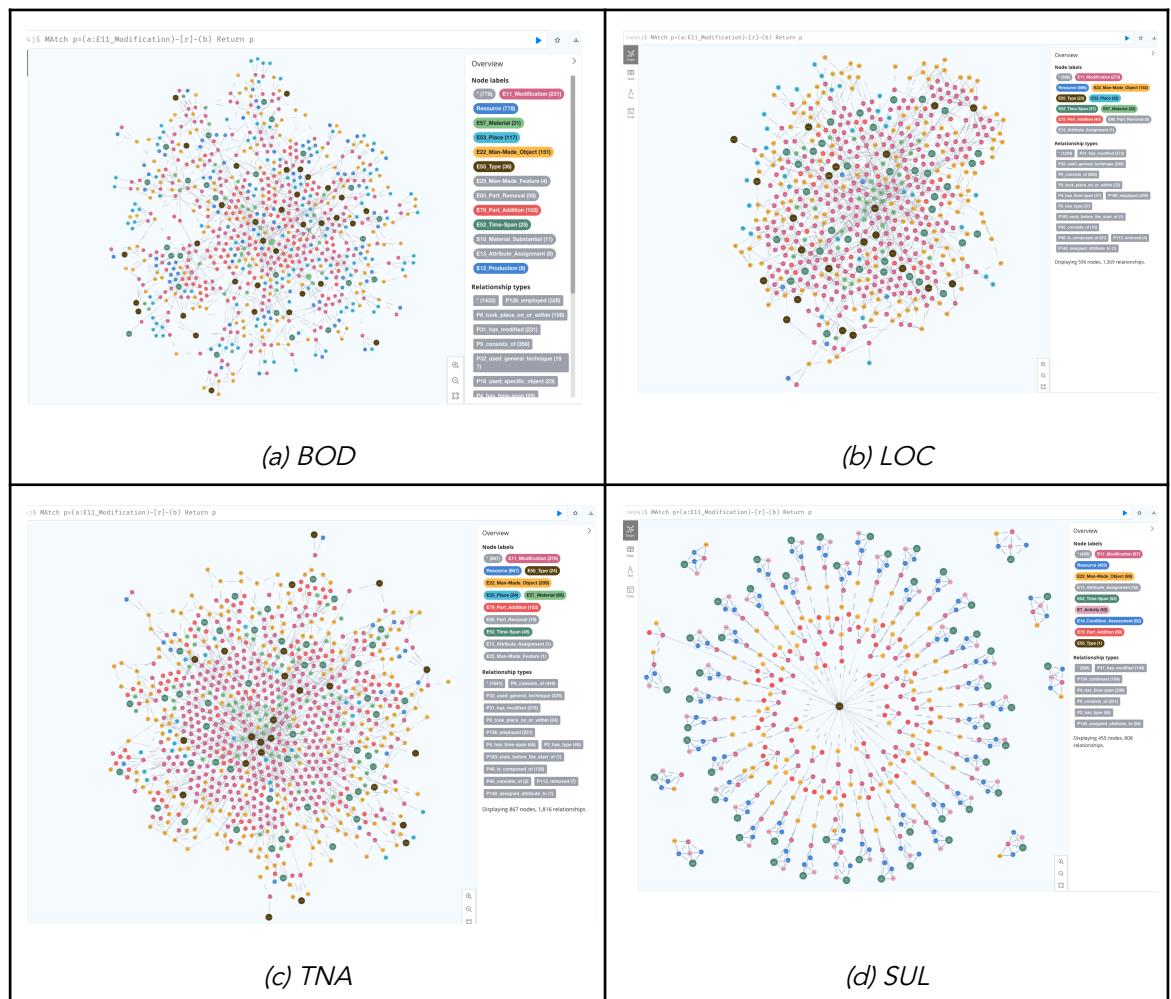


Figure 6.23. Visualisations of E11_Modification and neighbouring nodes (of length 1) for each LCD dataset.

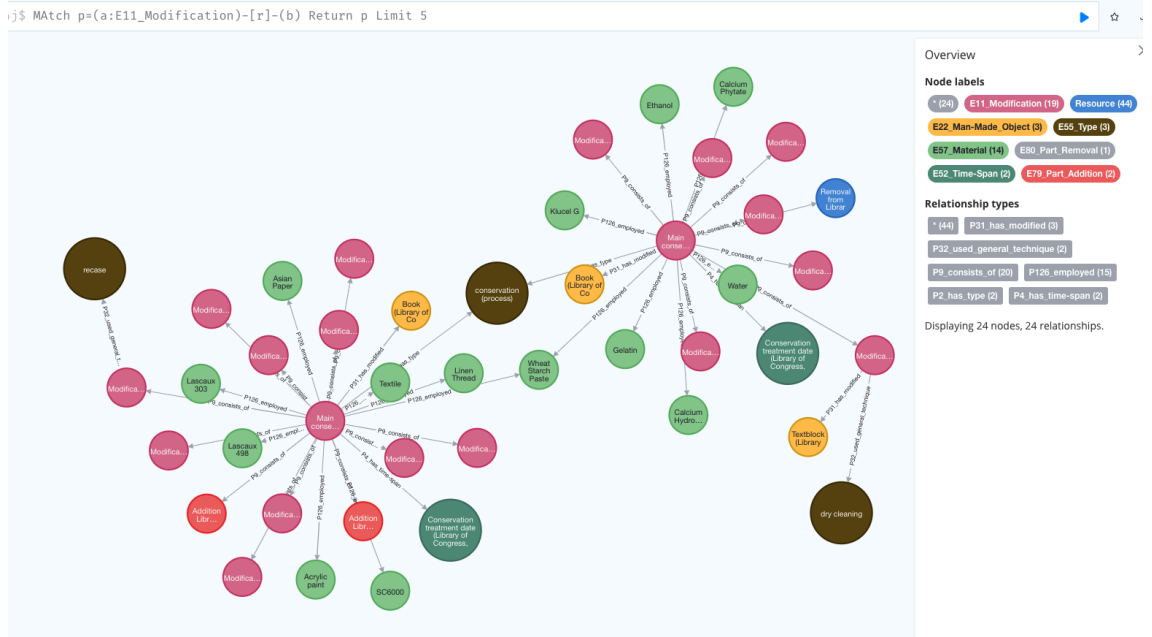


Figure 6.24. Visualisation of the LOC E11 graph, limited to viewing 2 main conservation events. Note the star schema structures around “main” event nodes and cross-star schema connections via material types and deterioration or process types.

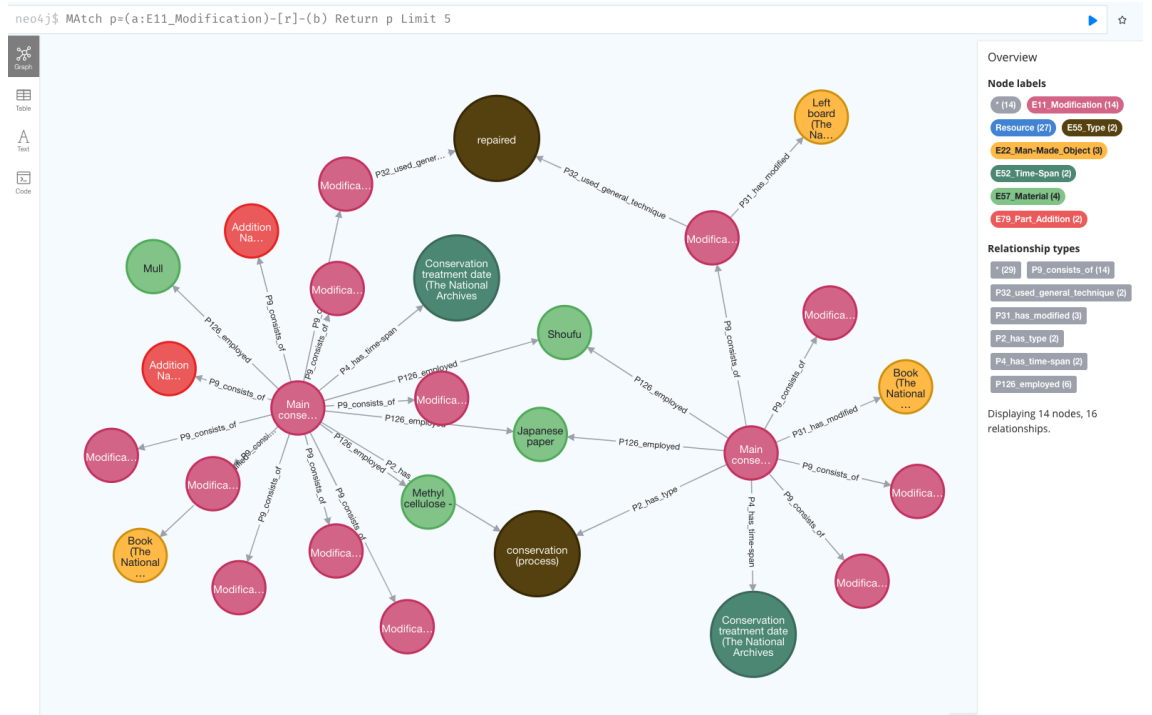


Figure 6.25. Visualisation of the TNA E11 graph, limited to viewing 2 main conservation events. As above, note the star schema structures around “main” event nodes with cross-star schema connections also present via material types and deterioration or process types.

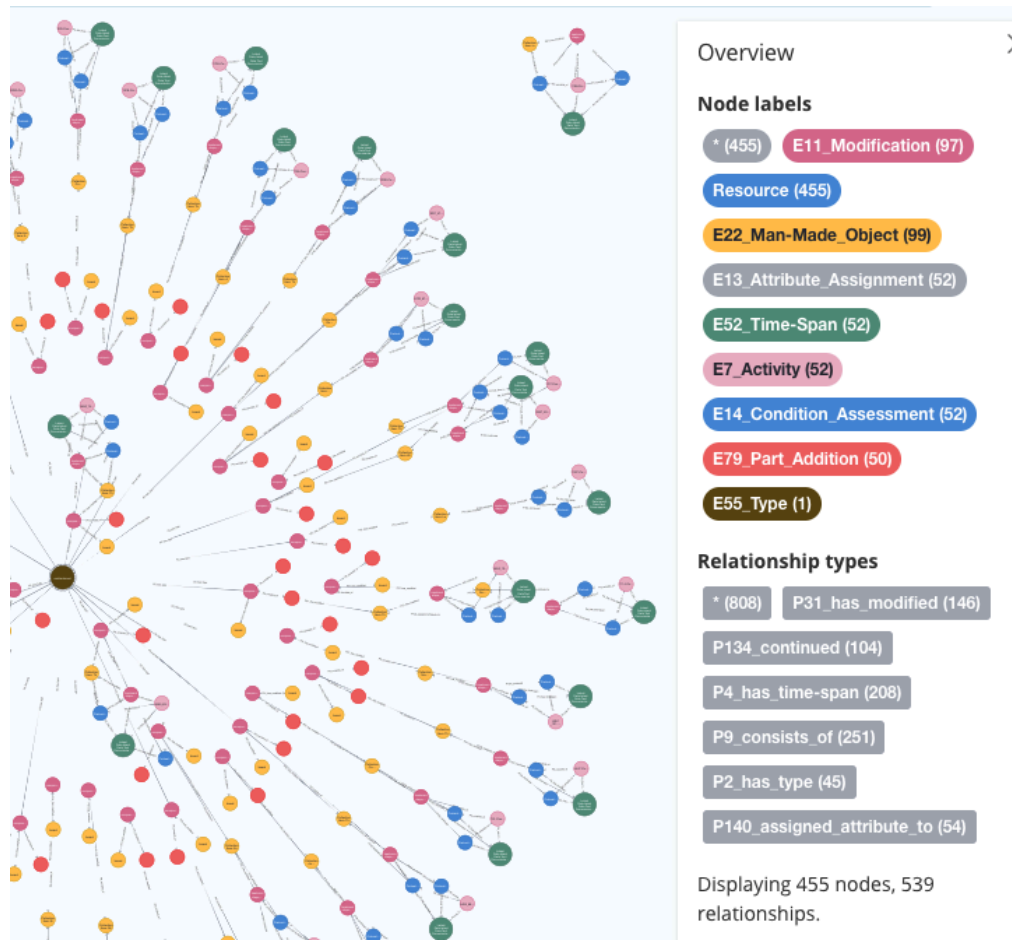


Figure 6.26 Partial detailed view of the SUL E11 and neighbours graph. The E55_Type hub node is for "reattachment".

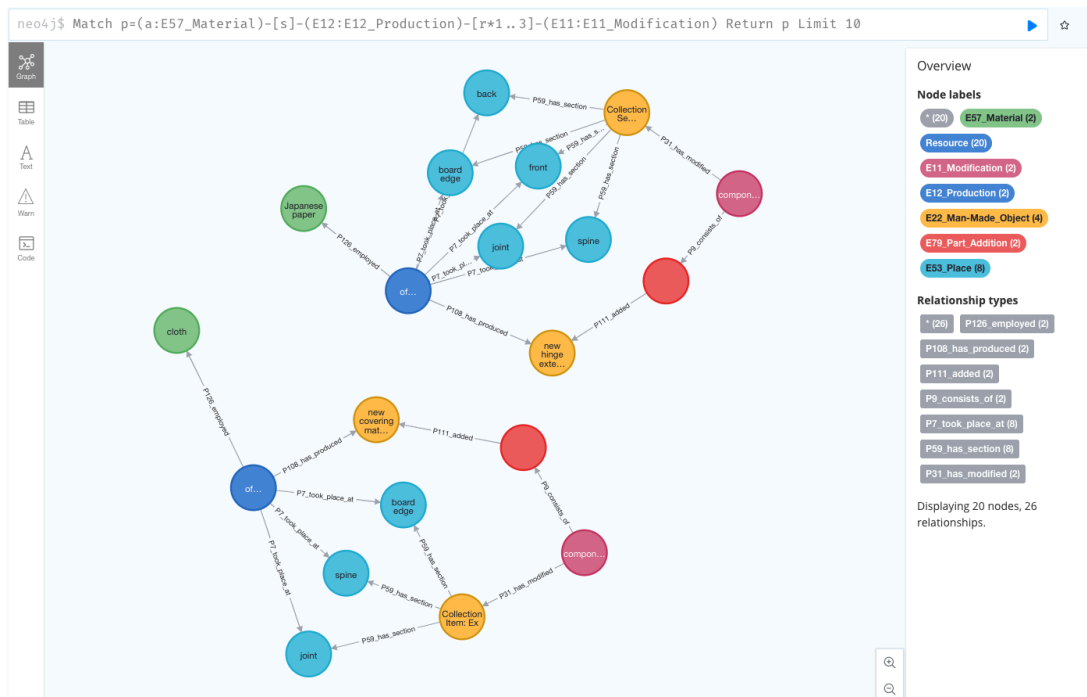


Figure 6.27 Visualisation of where E11_Modification nodes are situated relative to E57_Material nodes.

In order to identify how materials relate to modification, production and additions (i.e. specific subclasses of the E11 conservation treatment class) a variable path length query is needed. The results can be found in Figure 6.27 which confirm that there are no direct relationships between E57_Material and E11_Modification in this model. Instead, the bottom component in Figure 6.27 can be decoded as a reattachment event (E11, pink), which consists (P9) of a part addition event (E79, red) that added (P111) new covering material (E22, yellow). The reattachment event (E11, pink) modified (P31) the collection item (E22, yellow) which has 3 places (all P59)—a board edge (E53, light blue), the spine (E53, light blue), and a joint (E53, light blue)—where took place (P7) production (E12) of the aforementioned new covering material (E22, yellow). Thus, adjacency and distance characteristics in the SUL model are very different to BOD, LOC and TNA.

6.4.4 Analysing for Techniques

As stated above, the LCD project queried for 15 board reattachment techniques (see Table 6.1.1) to determine usage trends over time (Velios and St. John 2022, Figure 3). However, unlike examining for objects, materials, and types in the last few sections, which all fall within specific mapped classes, analysing for the techniques used by conservators required several strategies. In this study, the LCD datasets, were queried for as many techniques as identifiable (except for strategy 7) using the following strategies:

- Strategy1. Search for P32_used_general_technique relationships and their E11 (or related subclass) subject and E55_Type object (range).
- Strategy2. Search for P33_used_specific_technique relationships and their E11 (or related subclass) subject and E29_Design_or_Procedure object (range).
- Strategy3. Search for E29_Design_or_Procedure nodes directly.
- Strategy4. Use a variable path length query from an E55_Type node to E57_Material node via a P32_used_general_technique relationship.
- Strategy5. Use a variable path length query from an E55_Type node to E57_Material node via a P33_used_specific_technique relationship.
- Strategy6. Use strategy 4 or 5 with an explicit E55_Type node property.
- Strategy7. Using a prepared csv list of the 15 techniques identified by Velios and St. John (Table 6.1.1 above) to compare against, retrieve, and visualise matches in each LCD dataset.

The reason for requiring multiple strategies in finding conservation techniques is due to the various means by which such techniques can be mapped to the CIDOC CRM via specific classes and properties. Strategies 1 and 2 explicitly search for the two CRM properties (P32 and P33) that identify a technique type (encoded as E55_Type). That is, in the set of all E55_Types in a database, a subset of those will be technique-related types which can be identified, and therefore filtered out, by specifying one of two properties (P32 or P33) of which have been designed to identify (i.e. point to) this subset of E55_Type. There were duplications in some instances, for example, in the SUL dataset, "component application", "production of hinge", and "production of covering material" were objects to both P32_used_general_technique and P33_used_specific_technique.

Strategy 3 is to find all E29_Design_or_Procedure nodes which are plans or sequences of action and can themselves constitute a technique. However, further filtering may be necessary by specifying the expected neighbouring classes that would differentiate, for example, a documentation procedure from a treatment procedure. The relationship between E29_Design_or_Procedure and E57_Material is explicitly expressed via a P68_foresees_use_of (use_foreseen_by) property. However, as this is a forward-looking statement, the user must apply domain specialist knowledge to determine whether or not techniques identified by E55_Type and those identified by E29_Design_or_Procedure are semantically comparable for the purposes of their search or data mining tasks.

Techniques identified via strategies 1, 2 and 3 can be found in Table 6.4.6, however, any duplicates have been removed for clarity. Nevertheless, while Velios and St John (2022) focused on 15 board reattachment techniques, we can see from Table 6.4.6 that BOD, LOC and TNA datasets captured other techniques related to board reattachment as well. The BOD dataset was modelled with additional triples where E29_Design_or_Procedure nodes (with composite rdfs:labels that named the technique and the object identifier) had P2_has_type relationships with E55_Type nodes with unblended technique labels. The LOC and TNA datasets did not have any P33_used_specific_technique relationships nor any E29_Design_or_Procedure nodes. In the case of SUL, the same rdfs:labels were applied to E55_Type and E29_Design_or_Procedure for 'reback' and "paper hinge".

Strategy 4, 5, and 6 searches for the connection between techniques and materials using variable path length queries. These strategies recognise that, in order to identify trends such as which technique uses what materials, requires looking beyond immediate

neighbours and leveraging other classes such as E3_Condition_State and/or E53_Place which can identify where a treatment has been applied.

Strategy 7 uses a prepared csv list of the 15 techniques identified by Velios and St. John (Table 6.1.1 above) to compare against, retrieve, and visualise matches in each LCD dataset. The query can be further refined by specifying the start or end nodes as specific E55_Type or E29_Design_or_Procedure. However, this was not implemented for the results seen in Figure 6.28 which includes matches with E11_Modification or other nodes where the technique name string matches any rdfs:label. Table 6.4.6 provides the statistical overview using strategy 7 on each dataset in terms of how many techniques match the defined list.

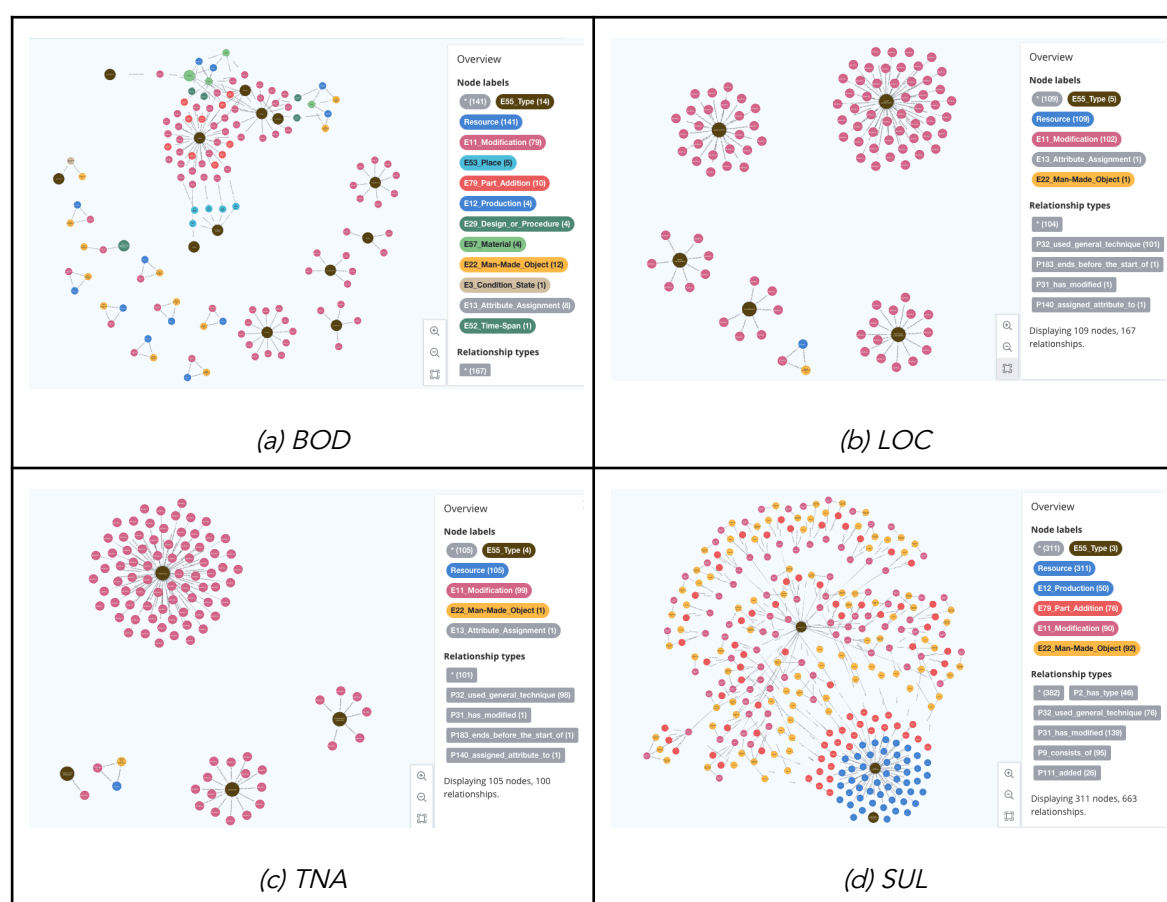


Figure 6.28 Visualisation from each LCD dataset showing instances matching the list of 15 board reattachment techniques identified and used for queries by Velios and St. John (2022).

Table 6.4.6 Matches to the LCD techlist (see Table 6.1.1)

LCD dataset	No. of Instances Matching LCDtechlist items	No. of Matched E55_Types
BOD	156	14
LOC	167	5
TNA	100	4
SUL	663	3

Table 6.4.7 Techniques identified via strategies 1-3 from each LCD dataset

	BOD	LOC	TNA	SUL
1	["sewing", "concertina guard"]	["inner joint repair"]	["sewing supports"]	["board reattachment"]
2	["pinning"]	["board reattachment"]	["paste wash"]	["Component Application"]
3	["two-on"]	["recase"]	["infilled/pulp/leafcast"]	["production of hinge"]
4	["primary sewing (endband techniques)"]	["disbinding"]	["laminated"]	["production of covering material"]
5	["pasting"]	["resewing"]	["original spine reattached"]	["adhered"]
6	["secondary sewing (endband techniques)"]	["flattening"]	["consolidated"]	["production of hinge extensions"]
7	["Front bead (Bodleian, Ms. Auct. D.4.17)", "Front bead (Bodleian, Ms.BOM.187)"]	["mending"]	["loose material attached"]	["guarded"]
8	["rolling"]	["dry cleaning"]	["reattaching"]	
9	["folding in half"]	["rebinding"]	["leather dressed"]	
10	["herring bone"]	["humidification"]	["paper repair"]	
11	["dyeing"]	["board edge consolidation"]	["outside joint strengthened"]	
12	["lacing in"]	["rebacking"]	["lifting"]	
13	["Vat (dyeing) (Bodleian, Arch.B.c.4)"]	["guarding"]	["original cover reattached"]	
14	["lining (technique)"]	["leather consolidation"]	["guarding"]	
15	["sanding"]	["sizing"]	["rounding"]	
16	["stitching"]	["washing"]	["new board(s) attached"]	
17	["one on, one off"]	["tooling"]	["volume pulled"]	
18	["all-along"]	["readhere lifting covering material"]	["lined"]	
19	["supporting"]	["outer joint repair"]	["mould cleaned"]	
20	["link-stitch"]	["previous mend removal"]	["old repairs removed"]	
21	["rebacking"]	["alkalization"]	["washing"]	
22	["consolidation"]	["fills"]	["sewing"]	
23	["repairing"]	["tape removal"]	["sewing reinforced"]	
24	["detaching"]	["board slotting"]	["flattened"]	
25	["slotting"]	["hinging"]	["mechanical surface cleaning"]	
	BOD	LOC	TNA	SUL

26	["fraying"]	["removing guards"]	["re-backed"]	
27	["building up"]	["hinge removal"]	["boards split"]	
28	["humidification"]	["mold remediation"]	["repaired"]	
29	["retanning"]			
30	["drying"]			
31	["guarding"]			
32	["washing"]			
33	["deacidifying"]			
34	["bookbinding (process)"]			
35	["paring"]			
36	["rehitching"]			
37	["lifting"]			
38	["tipping"]			
39	["poulticing"]			
40	["dry cleaning"]			
41	["folding"]			
42	["cleaning"]			
43	["oversewing"]			
44	["brushing"]			
45	["in the round"]			
46	["cutting"]			
47	["sizing"]			
48	["rubbing"]			
49	["dry (technique)"]			
50	["infilling"]			
51	["reinforcing"]			
52	["scraping"]			

6.4.5 Analysing for Trends Over Time

Trend analysis was an identified aim for the LCD project. Identifying trends can help in planning and in reviewing work practices. Two specific areas for trend analysis from the LCD datasets include looking at the use of materials over time and techniques over time. Time-related graph results can be referred to as temporal graphs. In the CIDOC CRM, E52_Time-Span is the class for “abstract temporal extents” and will be the principal classes necessary for time-based queries. However, in the LCD datasets, E52_Time-Span nodes make up only a very small portion of each dataset, therefore, any insights or patterns that can be extracted from these cases may be limited and not necessarily representative of each institutional collection.

Table 6.4.8 Number and Percentage of E52_Time-Span nodes per LCD dataset

	Number of E52_Time-Span nodes	Percentage of total network
BOD	87	3.55%
LOC	178	10.43%
TNA	90	4.25%
SUL	158	7.12%

Table 6.4.8 summarises the number of E52_Time-Span nodes from each LCD dataset and the percentage of the network this represents. The datetime entries for “P82a_begin_of_the_begin” and “P82b_end_of_the_end” (now node properties to E52_Time-Span are partially artificial, for example, “2018-01-01T23:59:59”. The years may be accurate, however, the month, day, hour, minute and second values have been artificially all set to 1 January at 23:59:59, likely as more precise data was not available to the modelers.

The LOC and SUL datasets were investigated for time-based trends as they have the highest proportions of E52 nodes. Firstly, temporal correlations regarding materials used as captured in the LOC dataset was investigated using an undirected, variable path length query to determine the proximity of E52_Time-Span nodes to E57_Material:

```
MATCH p= (a:E52_Time-Span)`-[r*1..3]- (b:E57_Material) RETURN COUNT(p)
```

The results for LOC showed 478 paths found between E52 and E57 within 3 hops. There were 239 paths found within 2 hops but 0 paths within 1 hop, confirming that E57 and

E52 did not have direct relationships in the LOC network. Figure 6.29 shows that the length 2 distance is due to the E57_Material type nodes and E52_Time-Span nodes are situated with a “main conservation event” (E11_Modification) node in between. Figure 6.30 shows that the extra hop (length 3) are the distance from material nodes to “treatment decade” (E52_Time-Span) nodes.

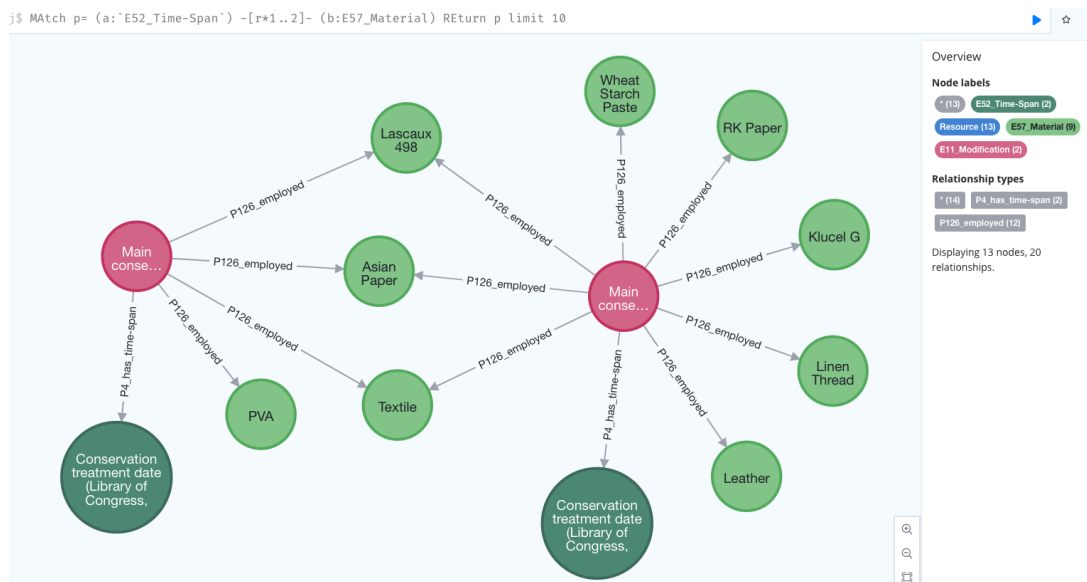


Figure 6.29 Visualisation of distance between E52_Time-Span (large, dark green) nodes to E57_Material (light green) nodes are of length 2 in LOC.

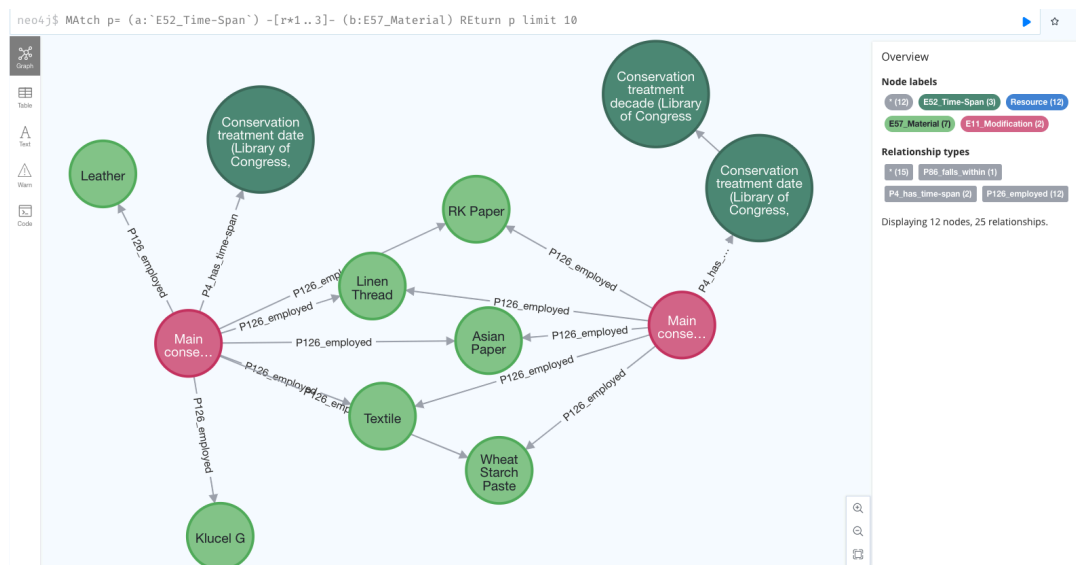


Figure 6.30 Visualisation of distance between E52_Time-Span “decade” node to E57_Material are of length 3 in LOC.

Table 6.4.9 Material Type Usage Over Time

MaterialType	EndDateMin	EndDateMax
--------------	------------	------------

["Acrylic paint"]	01/01/2013	01/01/2019
["Agarose Gel"]	01/01/2015	01/01/2018
["Aquazol"]	01/01/2018	01/01/2018
["Asian Paper"]	01/01/2010	01/01/2019
["BASF leather dye"]	01/01/2013	01/01/2018
["Binder's board"]	01/01/2018	01/01/2018
["Calcium Hydroxide"]	01/01/2010	01/01/2016
["Calcium Phytate"]	01/01/2016	01/01/2016
["Cast Acrylic "]	01/01/2016	01/01/2017
["Chalk Pencils"]	01/01/2017	01/01/2017
["Colored Pencils"]	01/01/2014	01/01/2019
["Ethanol"]	01/01/2014	01/01/2019
["Gelatin"]	01/01/2010	01/01/2016
["Gellan gum"]	01/01/2016	01/01/2017
["Handmade Western Paper"]	01/01/2010	01/01/2019
["Heat set tissue"]	01/01/2014	01/01/2014
["Klucel G"]	01/01/2013	01/01/2019
["Lascaux 303"]	01/01/2018	01/01/2018
["Lascaux 498"]	01/01/2013	01/01/2019
["Leather"]	01/01/2013	01/01/2019
["Linen Thread"]	01/01/2013	01/01/2019
["Machine Made Western Paper"]	01/01/2018	01/01/2019
["Magnesium Bicarbonate"]	01/01/2014	01/01/2014
["Methylcellulose"]	01/01/2014	01/01/2019
["Pastel"]	01/01/2018	01/01/2018
["PVA"]	01/01/2014	01/01/2019
["Remoistenable Tissue"]	01/01/2018	01/01/2018
["RK Paper"]	01/01/2015	01/01/2019
["SC6000"]	01/01/2017	01/01/2018
["Textile"]	01/01/2013	01/01/2019
["Water"]	01/01/2010	01/01/2016
["Watercolor"]	01/01/2018	01/01/2018
["Wheat Starch Paste"]	01/01/2010	01/01/2019

Once the paths between E52_Time-Span and E57_Material were identified, it was possible to filter and collate the results. Table 6.4.9 presents the results for all E57_Material in the LOC dataset, their earliest mention (EndDateMin) and their latest mention (EndDateMax). In the LOC dataset, only P82b_end_of_the_end were assigned

values. Figure 6.31 is a timeline/Gantt chart⁹ of the data from Table 6.4.9. The material types highlighted in red in Table 6.4.9 correspond to the short, red bands (appearing more like small red rectangles) in Figure 6.31. These are material types where the EndDateMin and EndDateMax occurred within the same year. In fact, due to the artificial month and date values, the data appears to occur on the same day. However, we can infer from the artificial values that the intention may include the full duration of those years. Either way, they highlight material types that were only used very briefly, possibly only for a single treatment. It is possible there are other instances of these materials being used that were not included in the sample that formed the LCD dataset but exist in the wider institutional databases. The material types highlighted in green in Table 6.4.9 and visualised as green bars in Figure 6.31 show the min and max end dates (mention dates). While it is not clear if there were periods of hiatus in the application of some of these materials using this visualisation method, it provides a means to compare overview ranges for usage within the 2010-2019 decade. This demonstrates one example of trend analysis afforded by path-based proximity between temporal data and other data content.

Table 6.4.9 and Figure 6.31 show that 3 materials were declared to be used across the full decade, these being "Asian Paper", "Handmade Western Paper", and "Wheat Starch Paste". It is highly likely that this range is representative of wider practices based on domain knowledge and the ubiquity of these materials in paper-based conservation. However, to draw wider correlations or make high-confidence inferences typically would require confidence in how representative the data is. For example, the visualisation in Figure 6.31 shows the use of many more material types from 2013 onward. However, this can be due to the sampling choices for this dataset. Nevertheless, for demonstration purposes, let's take the data from Table 6.4.9 and Figure 6.31 at face value. It can be observed in Figure 6.31 that there are significantly more green bars beginning in 2013, the material types being "BASF leather dye", "Klucel G", "Leather", "Linen Thread", and "Textile". This may have coincided with an exhibition or loan consisting of more leather-based objects than was required for treatment before and of which required making facsimile parts, i.e. the use of BASF leather dye, which would have been highly unorthodox to have been applied directly on a collection item itself, but may have been used to colour match a modern infill piece.

⁹ The timeline/Gantt chart visualisation was prepared outside of Neo4j using Google Sheets.

Next, temporal correlations regarding techniques as captured in the SUL dataset were investigated using an undirected variable path length query to determine the proximity of E52_Time-Span nodes to P32_used_general_technique and P33_used_specific_technique relationships. (Earlier investigations found only Strategy 1 and 2, as mentioned above, produced technique-related results for SUL.)

```
MATCH p=(a:`E52_Time-Span`)-[r*3]-(b)-[s:P32_used_general_technique]-(c)
RETURN COUNT(p)
```

Prior to running the technique-related queries, a query was run to retrieve all of the E52_Time-Span nodes for review and to determine if the datetime values would be appropriate for technique-related queries. It was found that the datetime node properties were specified as publication years for the collection items (e.g. books) and were not datetimes associated with conservation events. Thus, the only potential inferences that can be made are possible correlations between subsequent treatment techniques required and the age or original production of the objects (books). However, due to the limited sample size of the SUL dataset, any correlation patterns suggested here are only demonstrative of the process.

When matching on an undirected, variable path length of 3 from any E52_Time-Span to any P32_used_general_technique relationship found 494 paths. A variable path length where the first sequence of relationships has length 2 produced 0 paths that then led to a node followed by a P32 relationship. A path length of 1 for the first variable length sequence (variable "r") was also tested with 0 results. A direct relationship between E52 and P32 was also tested, ie. (:E52)-[:P32]-(c), which also produced 0 results, thereby confirming the [r*3] element to the path.

Table 6.4.10 Results of E52_Time-Span datetimes and general technique types (via P32).

techniqueType	minStartDate	maxStartDate	minEndDate	maxEndDate
["board reattachment"]	["1476-01-01T00:00:00"]	["1904-01-01T00:00:00"]	["1476-12-31T23:59:59"]	["1904-12-31T23:59:59"]
["adhered"]	["1610-01-01T00:00:00"]	["1854-01-01T00:00:00"]	["1610-12-31T23:59:59"]	["1854-12-31T23:59:59"]
["guarded"]	["1578-01-01T00:00:00"]	["1870-01-01T00:00:00"]	["1578-12-31T23:59:59"]	["1870-12-31T23:59:59"]

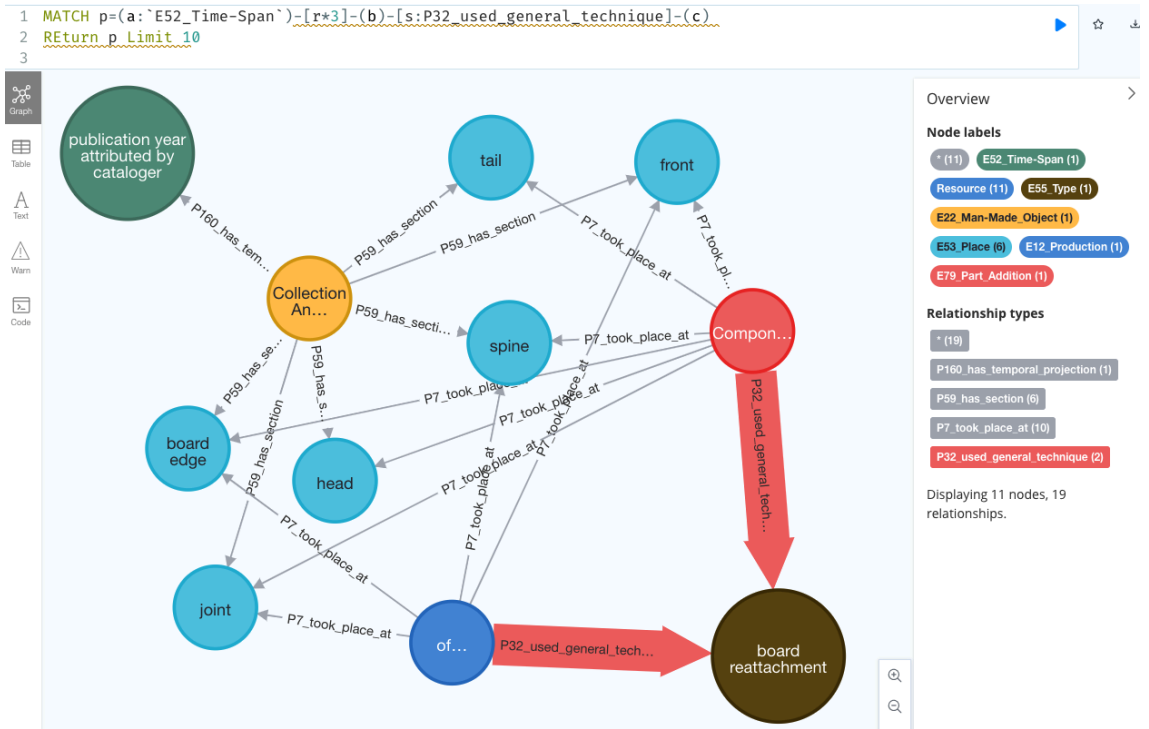


Figure 6.32. Relative position of *E52_Time-Span* to *P32_used_general_technique* relationships (shown here as red arrows) demonstrates how proximity correlates with relevance in the *SUL* graph.

Likewise, when applying an undirected, variable path length of 3 from any *E52_Time-Span* to any *P33_used_specific_technique* relationship, that is:

```
MATCH p=(a:`E52_Time-Span`)-[r*3]-(b)-[s:P33_used_specific_technique]-(c)
RETURN COUNT(p)
```

found 350 paths. As above, length 2 and length 1 were also tested but both had 0 results, thereby confirming the length 3 distance in the *[r*3]* element to the path. Table 6.4.11 presents the datetime ranges for each technique type and Figure 6.33 provides a visualisation of how proximity correlates with relevance in these queries.

Table 6.4.11. Results of *E52_Time-Span* datetimes and specific technique types (via *P33*).

techniqueType	minStartDate	maxStartDate	minEndDate	maxEndDate
["paper hinge"]	["1476-01-01T00:00:00"]	["1841-01-01T00:00:00"]	["1476-12-31T23:59:59"]	["1841-12-31T23:59:59"]
["reback"]	["1640-01-01T00:00:00"]	["1870-01-01T00:00:00"]	["1640-12-31T23:59:59"]	["1870-12-31T23:59:59"]

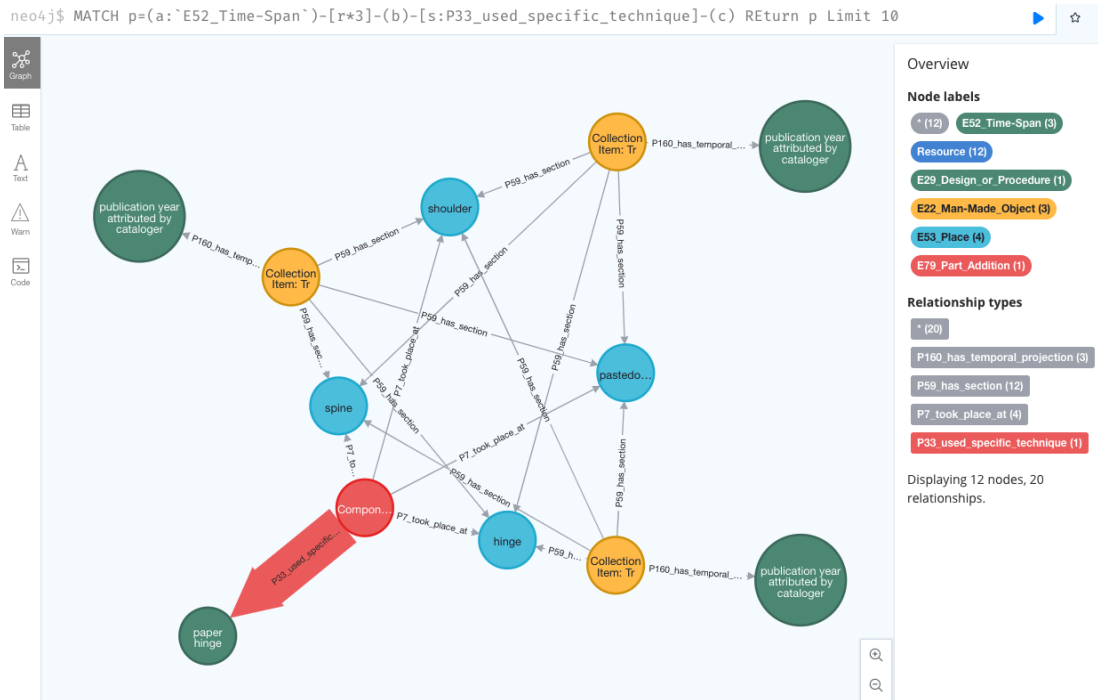


Figure 6.33. Relative position of E52_Time-Span to P33_used_specific_technique relationships (shown here as red arrows) demonstrates how proximity correlates with relevance in the SUL graph.

An additional source for technique context is in the domain nodes of the P32 and P33 relationships, these being E79_Part_Addition, E12_Production which are both subclasses of E11_Modification. By reviewing these domain nodes for P32 and P33, it presents a fuller context to the five identified techniques: "board reattachment", "adhered" "guarded", "paper hinge" and "reback" which are "production of hinge extensions", "Component Application", "production of covering material", and "production of hinge".

Although the SUL dataset had the most E52_Time-Span nodes, as the datetime data content held by these nodes were specifically limited to publication dates, it is not possible to determine when, for example, a paper hinge was made. Instead it informs us on when the book was published that needed a paper hinge added to it. The limited data in the contributed SUL dataset is not enough to identify correlations between date, time, manner of production, and subsequent treatment requirements.

6.5 Verification, Validation and Calibration

The results, separately and in aggregate, reveal previously unknown commonalities between the four LCD datasets as well as accentuate distinguishing patterns in the original source material. Therefore, there is scope for the application of graph theoretic

analysis for use within a verification, validation, and calibration (VVC) framework. Based on the study of the LCD group, the most promising measures for characterising and assessing each model were motif frequencies for profiling and eigenvector centrality for model assessment. Motif profiling demonstrated a means to characterise graph networks using quantifiable shared characteristics and quantifiable divergent characteristics. And as has been shown in the previous chapter, using eigenvector centrality has once again assisted in identifying (in section 6.3.7) potential errors in the modelling process. Therefore, for iterative model development, the use of eigenvector centrality should be included as part of a model assessment framework that informs downstream iterations. While Velios and St. John (2022) highlighted iterative development through querying until all expected results were produced as a means of model validation, this study shows how eigenvector centrality can also be useful in identifying errors and aid in refining a model to better align with its intended uses.

Finally, the diameter results for the LCD group are similar to the diameter results for the CIDOC CRM group. It is premature to assert these ranges of diameter for diagnostic means, however, there is the potential with further study that these diameter results indicate a threshold for conservation data, or certain types of conservation data as the LCD datasets were all related to board reattachment. Nevertheless, the diameter range offers a potential threshold for shallow analysis purposes, such as for including bounds in the diameter queries to avoid lengthy, exhaustive query runs.

6.6 Summary Findings

Chapters 5 and 6 have shown how organising and publishing data in a graph structure makes it conducive for analysis by graph theoretic means, the results of which can be applied to iteratively assess the knowledge graph and to inform ongoing data integration and study.

Graph theoretic analysis was able to identify commonalities in modeling patterns and reflect where those modelling patterns and choices diverged. The similarity across the graph structures for BOD, LOC, and TNA indicate shared commonalities in practice, either in the data capture process (e.g. the types of data captured) and/or the modelling process (e.g. how the data types have been related or mapped to the CIDOC CRM). The modelling variations observed in the LCD group have highlighted different conceptual approaches to using the CIDOC CRM which were confirmed through

interviews with the modellers. These variations in modelling decision-making resulted in clear structural differences, particularly between the BOD, LOC, and TNA datasets from the SUL dataset.

This study has also shown how RDF graphs can be transformed into LPG models in order to conduct such analysis using an LPG platform (Neo4j). Each dataset was treated as a separate database. Due to the slight differences between each LCD model, queries required slight modifications in order to access comparable data across the LCD group. However, the affordances of a labelled property graph approach to reviewing RDF graphs included being able to use Cypher as an alternative to SPARQL which allowed for more flexible path-based querying with built-in string parsing capabilities to accommodate this need for slightly modified queries that took into account path-based proximity, depending on the dataset.

Review of the LCD datasets revealed the challenge of modelling material instances versus material type, which is the scope of E57_Material, particularly where occasions of instance and type have dimensional attributes as was highlighted by the eigenvector centrality results for “ply” in the BOD dataset (section 6.3.7).

Instances of misunderstanding the scope notes to CIDOC CRM classes and properties or overfitting instances to certain classes can reduce the accuracy of the resulting model. However, the results of the graph theoretic analyses demonstrates that the measures included in this study's methods can be used to identify key elements within the graphs for closer inspection that leads to iterative improvements.

Based on these four datasets, conservation data has a strong tendency towards being non-planar. Planarity refers to whether or not a graph can be represented on a two-dimensional plane without any of its edges crossing. The presence of a $k_{3,3}$ or k_5 subgraph renders such a flat and non-crossing representation impossible as has been proven by Kuratowski's Theorem. In other words, the analyses thus far show conservation data and its relationships are inherently not flat structures. This higher dimensionality is an intrinsic characteristic and explains why tabular and traditional relational data models, while able to capture facets of conservation, have been so difficult to use to capture and model across conservation's more complex nature.

Based on the analysis of the LCD group, this author recommends the use of unique identifiers for treatment events that avoid compounding other data key-values. At present, to find the exact conservation treatment in a query would require knowing the

naming convention or object number but even more semantically confusing is when two events that were actioned upon the same object would, using the existing convention, have the same event identifier.

Finally, the insights gained from reviewing the LCD models have informed upon:

1. the key classes and properties used in a CRM-mapped model, and
2. how to best model in LPG so that it can be transformed straightforwardly into a CRM-mapped RDF structure.

The next chapter will apply these insights towards creating a direct LPG model of contributed data from The National Archives in order to test the affordances of modelling directly in LPG using it as a metamodel to aid transformations into RDF.

7.0 The Revised LPG Model

7.1 Introduction

This chapter demonstrates the implementation and results of integrating heterogeneous datasets to simulate the practice of cross-referencing across data sources in conservation using a labelled property graph structure. This chapter also aims to demonstrate how a labelled property graph approach as a metamodel can assist in the creation of new RDF knowledge graphs (Linked Data) by firstly allowing the domain expert to model data in a highly-associative way more akin to the cognitive structural schemas of domain specialists while still making it transformable to RDF for semantic validation using W3C validators to ensure machine-readability and semantic reasoning.

The labelled property graph (LPG) model described in this chapter and built from the TNA CCD dataset was the only dataset in this study built with all of the graph modelling principles as outlined in the method (chapter 4, section 4.3). While the mathematically consistent and computationally sound CIDOC CRM and LCD graphs satisfied the sets, tuples and categorical representation requirements listed, they were not necessarily modelled with structure mapping theory principles in place to clearly differentiate attributes from relationships, as is the case with RDF. However, this was a crucial consideration to minimise transformation errors and any subsequent need for reification to support transformation from LPG to RDF.

The use of knowledge graphs as a means to aggregate and integrate data was presented in chapter 3 on *Graphs*, specifically sections 3.3-3.5, with further explanations regarding Linked Data (RDF) as a type of knowledge graph presented in chapters 4, section 4.3. For the reasons stated in section 4.2.1 (*Limitations of RDF..*), investigations on three versions of the CIDOC CRM serialised in RDFS (in chapter 5) and four RDF datasets resulting from the Linked Conservation Data (LCD) project (chapter 6) were undertaken within a labelled property graph (LPG) environment, primarily to enable graph theoretic analyses, visualisation, and the encoding of queries within a single coding paradigm (i.e. the Cypher language). Nevertheless, modelling of conservation data using a labelled property graph model has yet to be investigated and recorded in the literature. Given the affordances of LPG as demonstrated by the findings in chapters 5 and 6, the transformational capabilities from LPG to RDF and vice versa, and the closer resemblance of the LPG structure to the highly-connected cognitive schemas of domain

experts unlike the diffuse structure of RDF (both points covered in section 4.3.4), it is critical to prepare a LPG model of conservation data to address and test these points.

For the remainder of the chapter, the specific LPG model created for this study will be referred to as the P3 (or Phase 3) model. (Or more specifically, P3-LPG for the LPG model and P3-RDF for the transformed version.) The principal aim of the P3 model is to address the research interests of the Collections Care Department at The National Archives in London who have provided the original data (i.e. the TNA CCD dataset) that enables this study.

7.2 Applying the Method to Case Study Components

To begin tackling *“how to build a conservation knowledge graph using a labelled property graph (LPG) model”* requires further decomposition of this research question itself. Before a truly large and comprehensive domain knowledge graph can be created, modelling begins at the local scale of the graph. These smaller-scale graphs are then aggregated and in doing so contribute towards the topology of the global scale knowledge graph. Hence, the first research question was reframed as the following series of questions to elucidate local scale modelling choices:

- What are the local-level rules for modelling conservation treatment data as a graph?
 - What elements are to be represented? (e.g. node labels, node properties, relationship types, edge properties)
 - What attributes (node or edge¹ properties) and relationships do entities have? (i.e. as this dictates the structure)
 - What structure would return meaningful results? i.e. how should these entities be connected given the benefits of a path-based query system and graph theoretic analysis?
 - How would these results correlate with reality? (e.g. does distance matter?)

A series of preliminary trials were conducted to identify the local-level rules for modelling. These trials (see Appendix H) constituted Phase 1 of this research which began with a review of the case study TNA CCD dataset and progressed to test the addition of other authoritative or high-quality datasets and metadata (Phase 1, models A

¹ Although relationship properties are defined tuples in LPG, no data was assigned to this set in this study.

- C in Appendix H) as a means of graph enrichment. The results of these preliminary trials showed there was scope for improvements and fine tuning and a need for a better understanding of the CIDOC CRM structure and the scope of its classes and properties. This led to the Phase 2 investigations which focused on the versions of the CIDOC CRM (chapter 5) and on existing RDF knowledge graphs from the LCD project (chapter 6) for comparative reference. Together, Phases 1 and 2 have informed a revised ETL process which was used for this phase (Phase 3) of the study.

The core premise for building these models and conducting the analyses has been to demonstrate how combining a fine-grained dataset of *specific* instances to more *general* conceptual representational levels and upper-level ontology, can aid more complex queries across multiple representational levels.

To reiterate the modelling principles presented in section 4.3, the model consists of representations (or elements, i.e. entities and relationships) which can be grouped into sets. Some representations fit into many sets, likewise, the sets in which an element belongs to helps to define that element. Hence, instances (particulars) and types (categoricals) are represented within the same structure and system.

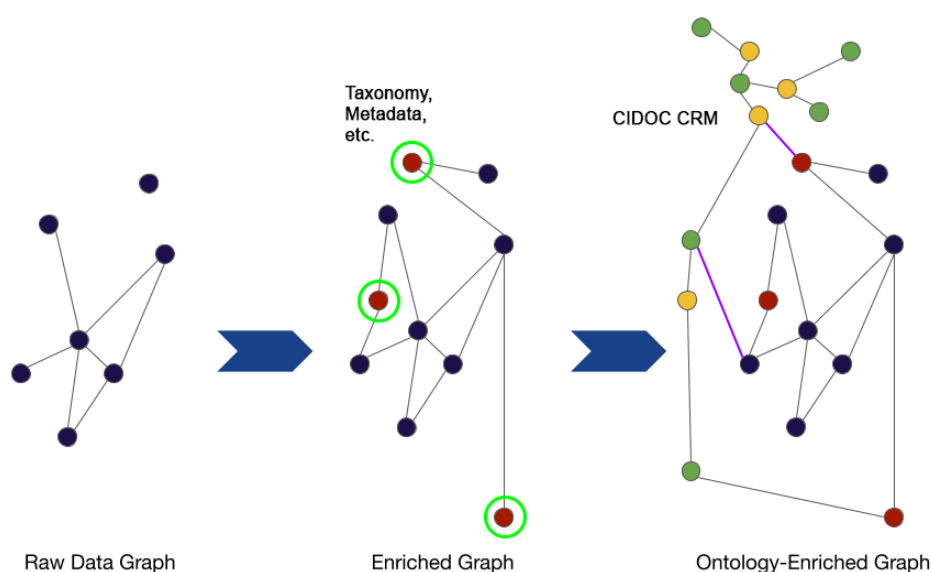


Figure 7.1. Building a prototype conservation knowledge graph, using a hypothetical representation.

The design of the model simulates the practice of cross-referencing across data sources in conservation practice. By using a graph representation and incorporating these sources into the graph model, the connections are made explicit. In addition to the

principle TNA CCD dataset of conservation treatment records and the CIDOC CRM ontology, the knowledge graph building process also modelled data from the following:

1. a dataset derived from natural language processing (NLP) procedures containing parsed terms from the “Comments” and “ConditionComments columns” of the TNA CCD dataset,
2. the National Archives’ online catalogue content via their Discovery² API (application programming interface) matching reference numbers in the TNA CCD dataset, and
3. the Conservation and Art Materials Encyclopedia Online (CAMEO).

These data components pertain to the building of graph models in Phase 1 (see Appendix H) and Phase 3 (this chapter). The Phase 2 investigations did not utilise these additional datasets. The following subsections provide further background to each dataset and general data handling procedures for the TNA CCD dataset and the three listed components mentioned above. More detailed preparation and ETL procedures can be found in the appendices.

7.2.1 The TNA CCD Dataset

As mentioned above, the principle dataset used in this study was provided by The National Archives, UK (TNA) and contains approximately 6,000 rows of conservation treatment-related data spanning the years 2015 - 2018. The principal dataset is a subset of The National Archives’ Collections Care Database and will be referred to throughout this thesis as the “TNA CCD” or simply the “CCD” dataset. Each row from the dataset corresponds to a single treatment event as demonstrated in Figure 7.2. RowIDs were assigned to enable provenance checking between the model and the original dataset³. There is some overlap between the TNA CCD dataset to be covered in this chapter and that of LCD-TNA in chapter 6. This overlap consists of 45 object reference numbers.

The modelling and analysis was undertaken on the author’s local computer without direct network connection to The National Archive’s systems beyond normal public internet access to their Discovery API (see section 7.2.3 below). The TNA CCD database itself is not linked via the API. Access to the approximately 6,000 rows of data was shared through a series of CSV files. Meetings and interviews were conducted with the Head of

² <https://discovery.nationalarchives.gov.uk/>

³ It is recommended to use a persistent unique identifier such as a UUID (universal unique identifier) for each data row when undertaking similar work beyond experimental implementation.

Conservation and Treatment Development, Sonja Schwill, Senior Conservation Manager, Sarah VanSnick, and Senior Digital Archivist, David Underdown.

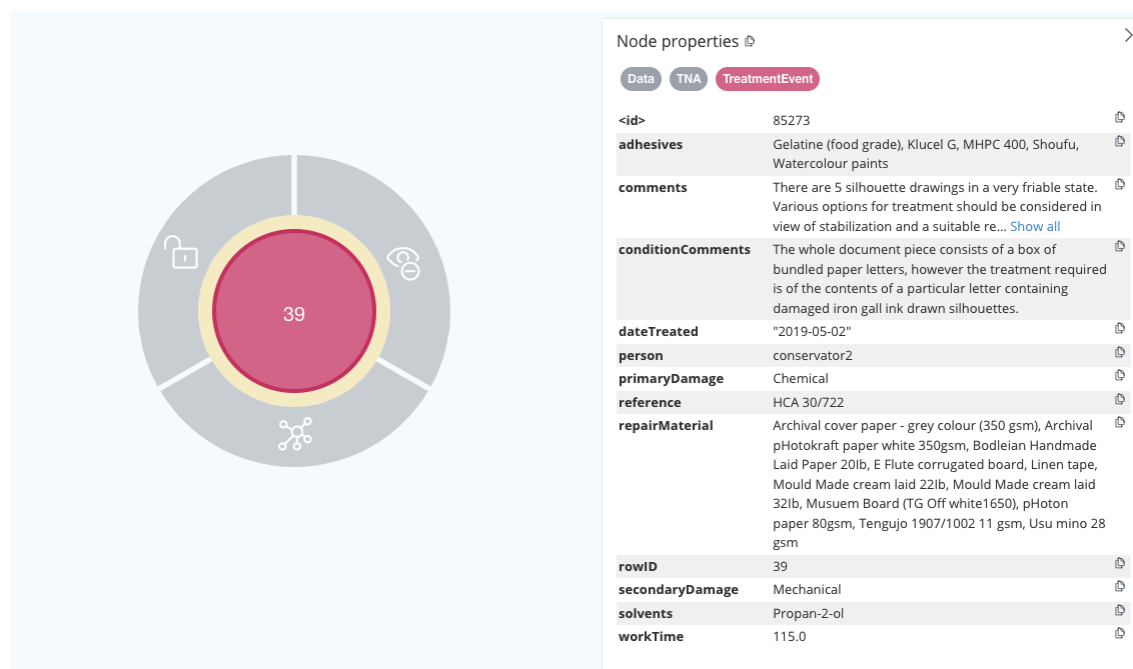


Figure 7.2. TreatmentEvent node with full row of data content mapped as properties to the node.

As the TNA CCD dataset is a series of conservation treatments recorded over time, it contains multiple instances of the same persons, materials, and object identifiers (reference strings). For example, an object may have needed treatment several times over the three-year span of the dataset. Each conservator would have been responsible for various treatments across various objects in that period. Likewise, the same types of materials would have been used during similar treatments of various objects by several of the staff. Therefore, these entity instances were modelled as their own nodes. To derive these nodes from the TNA CCD dataset, three sub-datasets were derived which, firstly, consisted of a list of unique persons (pseudo-anonymised as Conservator1, etc.) derived from the Person column, secondly, a list of unique materials derived from the Adhesives, Solvents, and RepairMaterial columns, and finally, a list of unique Reference strings derived from the Reference column.

For persons, a person-specific tabular dataset was created and duplicate mentions⁴ were removed and transformed into pseudo-anonymised labels. This resulted in 20 unique values representing 20 unique individuals who undertook the treatments. For materials,

⁴ There were no instances in this limited dataset where multiple persons possessed the same name.

entries under the column headings "RepairMaterial", "Solvent", and "Adhesive" were, first, separated if a part of a list⁵ (array), and then collated. In all, 105 unique values were identified for all materials based on direct string matching alone. These unique string values were retained at this level of transformation and not decomposed any further. For example, "Polyester-100 micron" and "Polyester-50 micron" were both retained as unique values and not further decomposed to "Polyester" as the additional sizing detail contributed to the specificity of the material type in their instances.

It is implicit in the original TNA CCD dataset that the identification of the same name refers to treatments carried out by the same person and not to be inferred that there are multiple individuals with identical names each carrying out only one treatment in the dataset. Likewise, it is implicit in the dataset that the mention of a material signals the use of that type of material in general and were not conveying direct instances of specific materials each time. As catalogue reference numbers are assigned to identify individual objects, the same inference can be made regarding unique Reference numbers. Thus, derivation of the unique Person list, unique Material list, and unique Reference list allowed for the identification and modelling of distinct entities (persons and objects) and entity types (materials) within the dataset that persist regardless of the :TreatmentEvent.

Unlike the persons, materials, and references, which are perceived as unique entities and should be modelled as such, dates, durations and comments, in the context of this dataset, are not entities but attributes. For example, while a specific day, let's say New Year's Day in 2022 (20220101) can be modelled as a unique entity, the current model does not model specified days as nodes. This does not preclude modelling as such at a later iteration.

Another example for this reasoning in deciding what is a node and what is a node property, is the time duration for each treatment. One row indicates the treatment event took 20 hours and another treatment (i.e. another row in the dataset) also took 20 hours to complete. However, while the duration lasted the same amount of time, it does not necessarily refer to the exact same window of time (e.g 5th March 2010 from 3PM BST until 11AM the following day), therefore having both events point towards the same "20 hours" node would be misleading, inaccurate and bad modelling practice. Furthermore, while a hypothetical model may have discrete durations as nodes in order to privilege the significance of "conceptual blocks of time" on par with other entities in a

⁵ This was only split in this manner to aggregate the list of unique materials used by TNA. The upload procedure for the main :TreatmentEvent and :Materials nodes (see ETL codes) were achieved using Cypher code to split the lists and create individual nodes for each adhesive, solvent and repair material named.

network, modeling it as such would require a clear conceptual need which in the current case it does not satisfy. From a data interrogation and querying perspective, while a user is likely to want to find out which treatments took longer than some duration or occurred between two dates, this is entirely achievable with these key-value pairs as node properties, and can still derive a temporal graph from time-related properties across labelled nodes.

“Pulling out” node properties and transforming them into their own nodes that now have the potential to be linked can be argued as an enrichment technique as the result increases the number of nodes and edges in the model and therefore confers a greater potential for network connections. In practice, it straddles both data preparation and enrichment and results in a star schema for each treatment event on record (see Figure 7.3). This star schema structure echoes the star schema structures seen in the LCD group where the hub node is the main conservation event. Unlike the LCD graphs, the hub (:TreatmentEvent) nodes carry data mapped from the full original data row. By querying on the “pulled out” nodes enables path-based queries on the network (see Figure 7.4).

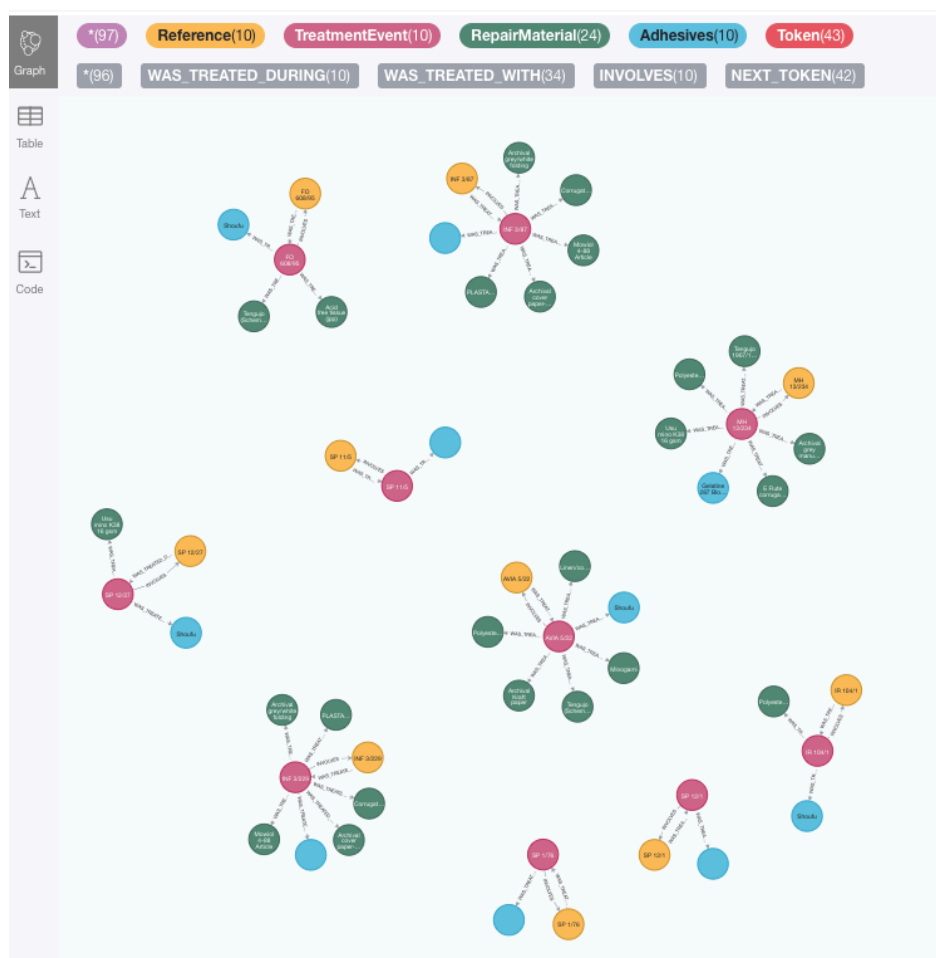


Figure 7.3. Visualisation of 10 treatment events represented as star schema.

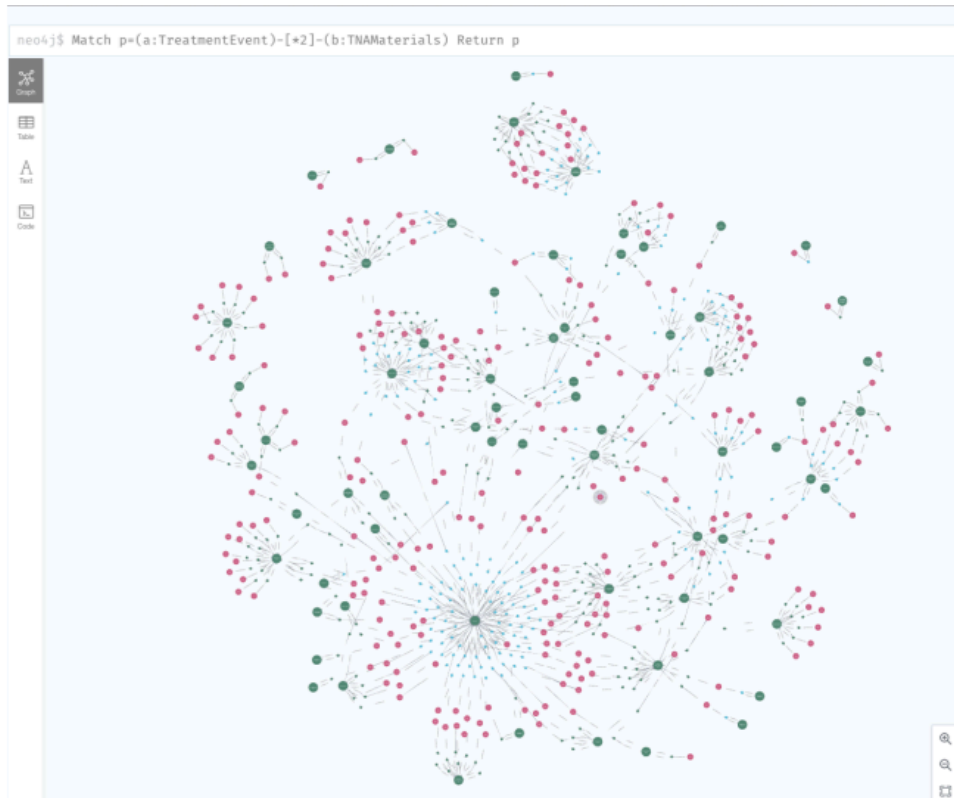


Figure 7.4. Materials Graph. Visualisation of the network of treatment event (pink) nodes and material type (green) nodes.

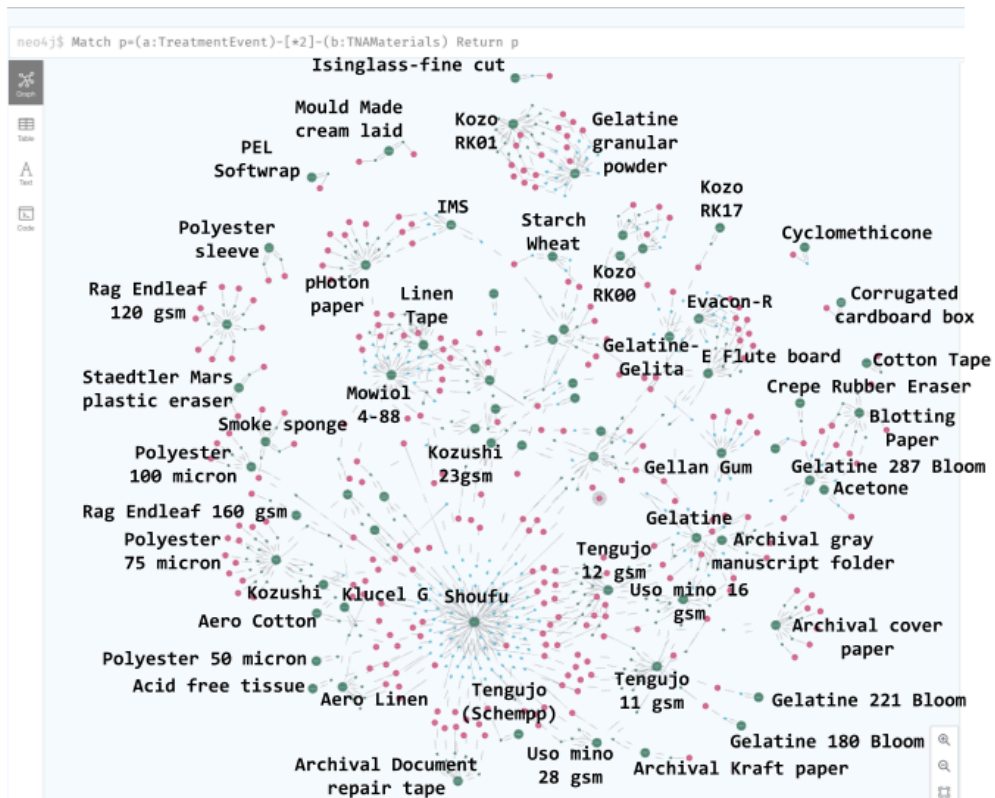


Figure 7.5 Annotated Materials Graph (manually annotated by the author).

7.2.2 The NLP-derived Dataset

The NLP-derived dataset is a prototype demonstration for how existing text-based data from conservation records can be processed and used to enrich a graph-based knowledge model. In this case, data from the “Comments” and “ConditionComments” columns⁶ of the principal TNA CCD dataset have been processed using the open-source natural language library spaCy⁷ to parse text, annotate parts of speech, extract tokens, and more specifically, to extract noun chunks and verbs (see Appendix C). This was undertaken to capture domain knowledge entities for expanding connections in the graph model. This simulates a conservator reading the text and identifying certain materials or procedures noted in the comment. This was especially informative for records where values for the structured fields “Adhesives”, “Solvents” or “RepairMaterial” are missing. To this end, the noun phrases (or chunks) and verbs conservators choose to use in the comments are of particular interest. The parsing of noun phrases provides access to potential named entities such as material names, for example “acid-free tissue” and “Japanese paper” without being restricted to using a predefined named entities list, which may be out of date or inexhaustive. (It also enables dynamic updating of such named entities based on the contents of the database.) In terms of verbs, if something was “added”, “removed”, “cut”, “desalinated”, “surface cleaned”, etc. these are all semantically significant, particularly in regards to techniques, and contribute to the assessment and decision making processes a conservator undertakes when reviewing treatment records.

The NLP-derived dataset in this study serves as a demonstration for how derived datasets can be used to augment highly-textual data and enrich the data network with previously inaccessible free-text. As such, it utilises a very basic natural language processing (NLP) approach. The NLP-derived dataset has enriched the P3 database by providing approximately 1,721 unique verbs and 11,493 unique noun chunks extracted from the Comments and Condition Comments to support further pattern matching and path-building. The addition of NLP-derived nodes, which include both (:NounChunk) and (:Verb) nodes, increased connections from :(TreatmentEvent) nodes to (:Cameo) nodes by 726 directed edges (Figure 7.4).

The modelling approach assumes that over time, nodes that do not have significant

⁶The data content also corresponds to the values in each n.comments and n.conditioncomments property key on :TreatmentEvent nodes.

⁷ <https://spacy.io/>

connections with incoming or outgoing edges will be those edges that are less relevant and can be trimmed from the database. The model development strategy will continue to connect relevant nodes while irrelevant nodes will tend to remain with low connectivity and therefore can be programmatically flagged for user review and subsequent pruning (i.e. deletion) at intervals.

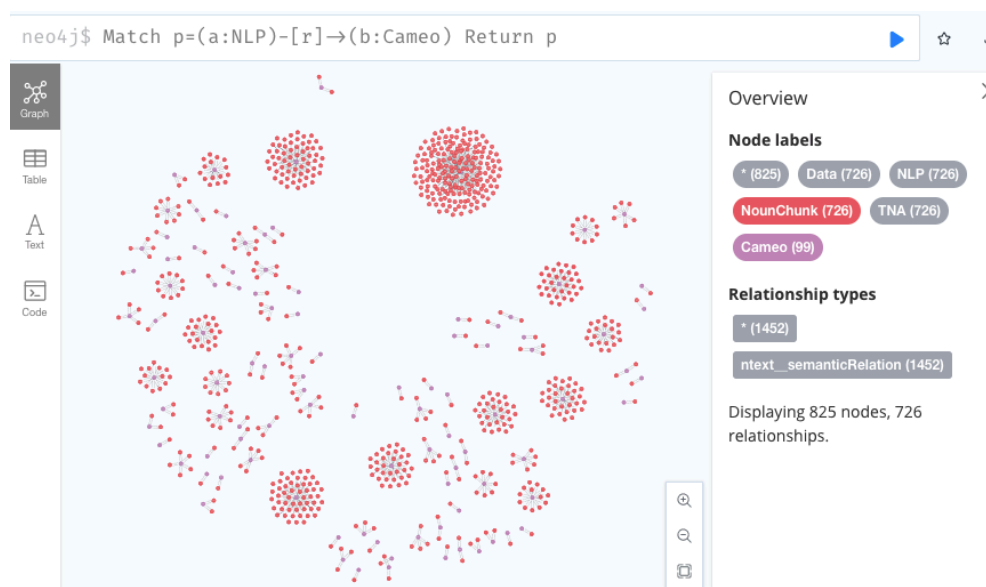


Figure 7.6. Visualisation of NLP-derived nodes as instances matching to Cameo (type) hub nodes.

Figure 7.7 shows an example where the large dark pink (:TreatmentEvent) node is only connected to a :Cameo node via a path through a red NLP-derived node where the token and noun chunk "Tyvek" was extracted, thus, enabling the treatment record to be accessible were a user to search from general terms using (:Cameo) nodes.

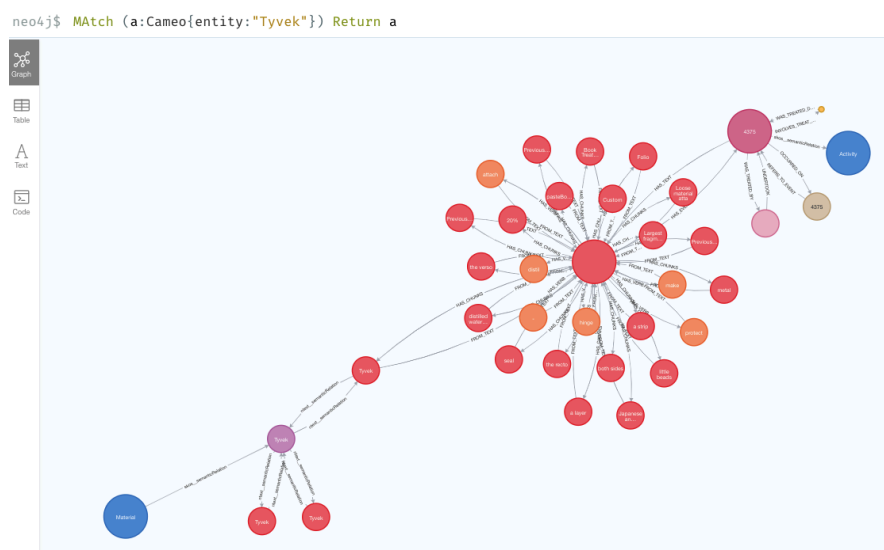


Figure 7.7 Connecting an NLP-derived mention of "Tyvek" to the categorical (:Cameo) node for "Tyvek".

neo4j\$ `Match (a:Verb) Return Distinct a.tokenText, Count(a) as Count Order by Count Desc`

	a.tokenText	Count
29	"sewn"	95
30	"required"	94
31	"lined"	86
32	"reattached"	82
33	"damaged"	77
34	"Treated"	76
35	"Repaired"	76
36	"left"	75
37	"approx"	74
38	"torn"	73
39	"inserted"	72
40	"seen"	71
41		

Started streaming 1721 records after 2 ms and completed after 3 ms, displaying first 1000 rows.

Figure 7.8 An excerpt of the results of a filter query to return extracted verbs and the frequency in which each verb appears as a (:Verb) node. The results show many of the actions captured in the verbs can be associated with conservation techniques.

Figure 7.8 shows a sample of (:Verb) nodes, the tokenized text it represents and a count of how many times the word/node exists in the database. These verbs are then linked to (:Tech) nodes created as categorical techniques or actions. A more sophisticated NLP approach, such as word vectors, is beyond the scope of this current thesis on

graph-based data analysis and integration. (Recommendations for further work can be found in chapter 9.)

7.2.3 The TNA Discovery Catalogue Data

This dataset was extracted from The National Archives' (TNA) online catalogue Discovery via its public API (application programming interface) (Underdown 2018). It is a subset of the Discovery catalogue that includes records matching the reference identifiers found in the principal TNA CCD dataset with both mid-level (catalogue level 3) and upper-level (catalogue level 1) related records.

The hierarchical records structure of the Discovery system consist of the following levels (The National Archive, n.d.) :

“There are seven levels in the catalogue, ranging from ‘department’ at the top of the tree to pieces and, occasionally items at the bottom:

- 1. Department – a government department, agency or body that creates the records*
- 2. Division – administrative section of a department, when appropriate*
- 3. Series – the main grouping of records with a common function or subject*
- 4. Sub-series – smaller groupings of records with a common function or subject*
- 5. Sub sub-series – smaller groupings of records with a common function or subject*
- 6. Piece – not a single piece of paper: a piece can be a box, volume or file*
- 7. Item – part of a piece: can be a bundle, a single document, a letter, and so on”*

The TNA Discovery catalogue levels are aggregated to form object identifiers (known as Reference numbers) and employ the slash (‘/’) as a separator (figure 4.17). Figure 4.18 demonstrates how the levels relate in the catalogue’s tree hierarchy.

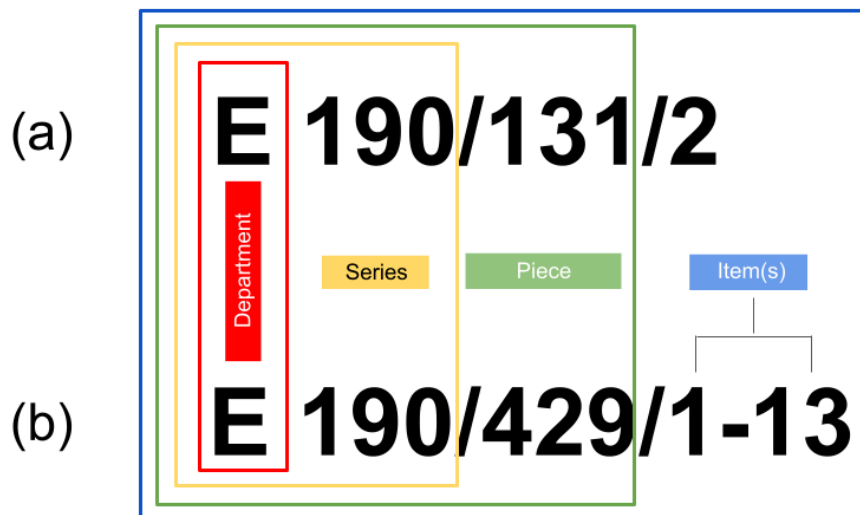


Figure 7.9. The TNA Discovery catalogue identifiers (known as Reference numbers) employ the slash as a delimiter. (a) An example reference string in the TNA CCD dataset. The full string refers to one object. (b) This second example reference string from the same dataset shows how several objects have been recorded in a batched range (not decomposed).

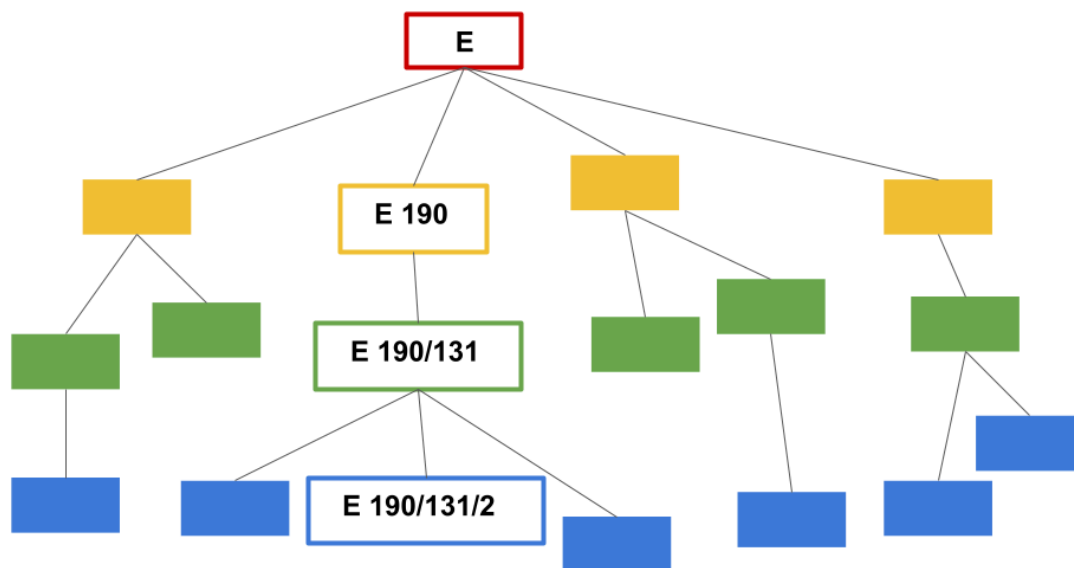


Figure 7.10 The TNA Discovery catalogue tree hierarchy and how Reference numbers are aggregated at different levels.

The uppermost-level reference identifiers consist of information about the originating departments while each sub-level narrows the specificity and tends towards information about a particular object or its constituent part. Mid-level identifiers refer to a collection, for example, letters of correspondence to and from a specific office. However, for historic and legacy reasons, not all catalogue levels are activated for each object. It was

found during preliminary calls to the API using the Discovery API sandbox⁸, that most reference identifiers do not have Level 4 or Level 5 related records. For example: "ASSI 94/1452" is a Level 6 identifier. If searching via the sandbox for this string using a 'Level 4' specified parameter, it returns 0 results. However, it returns a relevant record using 'Level 6'.

Furthermore, separators in the reference identifier string are not necessarily indicative of the actual catalogue level with multiple string and delimiter patterns found within the same departments. For example, "ASSI 25/2/2", despite two slash separators, is also a 'Level 6' record, as per the sandbox result. Therefore, simply splitting on a separator does not accurately represent the catalogue level hierarchies and there may not be a record for the previous one-level-up position (especially in the cases of Levels 4 and 5 which tend to only exist for certain collections). Another catalogue-specific idiosyncrasy is that "sometimes Level 6 entries refer to Items instead of Pieces" (Underdown 2021, personal correspondence). As a result of this variability in the reference string patterns and legacy catalogue level assignments, GET requests were batched by specified catalogue levels 1, 3, 6, and 7. For consistency, the existing catalogue levels and their associated labels (Department, Series, Piece, and Item) have been carried over.

Identifier strings that returned zero results using the established parameters can be found in Appendix D. These outlier strings were further subdivided into two groups. The first are batches where the entry within the principal TNA CCD dataset was entered as a hyphenated range of reference identifiers, for example "E 190/334/1-25" is to be understood as "E 190/334/1", "E 190/334/2"...and so on, consecutively, up to and including "E 190/334/25". The other outlier group are those with 'Folio' in the identifier string or other words like 'Part' or 'Tray', for example, "HCA 13/141 Folio 1-50". A separate Python script (see Appendix D) was written to iterate through the "batched" outliers as part of the API call. For the latter outlier group, where folio or part-specific queries returned zero entries, only the mid-level (level 3) and upper-level (level 1) records were returned. In the remaining cases, using the HCA example, "HCA 13" corresponds to catalogue level 3 and "HCA 13/141" corresponds to catalogue level 6. Search queries were revised to only search and return level 3 and level 6 catalogue data. In total this derived dataset consisted of 11844 nodes. A list of the second outlier group can be found in table D8.1.1 in Appendix D.

Content related to the treated objects were extracted from the Discovery catalogue database from multiple record levels including (:Department), (:Series), (:Piece), and

⁸ <https://discovery.nationalarchives.gov.uk/API/sandbox/index>

(:Item), representing record levels 1, 3, 6 and 7, respectively. Each catalogue record became a node and each of up to 34 field parameters for every record were transformed into node properties, although not every record has values for all fields. A list of the imported fields for each record from the Discovery API can be found in Appendix D. The key fields in the graph enrichment process focused on accessing the following specific parameters/node properties to provide content for searching and matching:

- content
- context
- department
- description
- heldBy
- physicalCondition
- place
- taxonomies
- title

Of the 11,759 (:Discovery) Nodes, 11,637 do not have a physicalCondition value or have a blank value (i.e. =""). The remainder, 122 records, have a physical condition of which can be further distilled down to 46 unique values.

7.2.4 Conservation and Art Materials Encyclopedia Online (CAMEO)

This section describes inclusion of content from CAMEO, the *Conservation and Art Materials Encyclopedia Online*⁹, as an authoritative reference on conservation materials. As a curated encyclopedia, the CAMEO content represents a specificity level that falls between the fine-grained instances of an institution's dataset and the more general ontology, at a level that Doerr (2009)¹⁰ refers to as 'categorical'. Therefore, it is a strong candidate dataset for use in graph enrichment as it contributes mid-level semantic entities for connecting the instances and the ontology.

CAMEO was originally developed at the Museum of Fine Arts, Boston in 1999 and currently contains ca. 10,000 records with technical, historical, chemical and visual information on materials used in artistic and cultural production (Derrick 2016)¹¹. Entries

⁹ <https://cameo.mfa.org>

¹⁰ Doerr, M.. (2009). Ontologies for Cultural Heritage. In S. Staab & R. Studer (Eds.), *Handbook on Ontologies*. Springer-Verlag Berlin Heidelberg.

¹¹ Derrick, M. (2016). CAMEO: The Science in Art. *Chemistry International*, 38(5), 8–13.

are wide ranging including materials by name, chemical and physical classes, composition groups and tools. Entries can include:

- tabular physical data, such as boiling points, freezing points, and glass transition temperatures,
- images, such as photographs, drawings, prepared cross-section micrographs (microscopic views),
- analysis patterns, such as XRD spectra of a material,
- and a bibliography.

The content is selected and curated based on mentions in conservation literature and is compiled to aid decision making in preventive and interventive conservation. It is a free and publicly available resource hosted on the MediaWiki platform. In ca. 2016, there were over 60 editors of the wiki (ibid).

"[CAMEO] provides a time saving resource for the conservation field, where knowledge regarding material properties, reactivity, and history can be crucial to success and safety" (ibid, 12).

The content of an encyclopedic resource such as CAMEO not only informs preventive and interventive treatments, but it also informs other conservation-related activities such as display case construction (ibid., 11) and managing health and safety (ibid., 8).

In the context of creating a domain knowledge graph for conservation, CAMEO provides high-quality data and metadata for persistent things (some with physical properties and others wholly conceptual) that are utilised or invoked within the conservation domain.

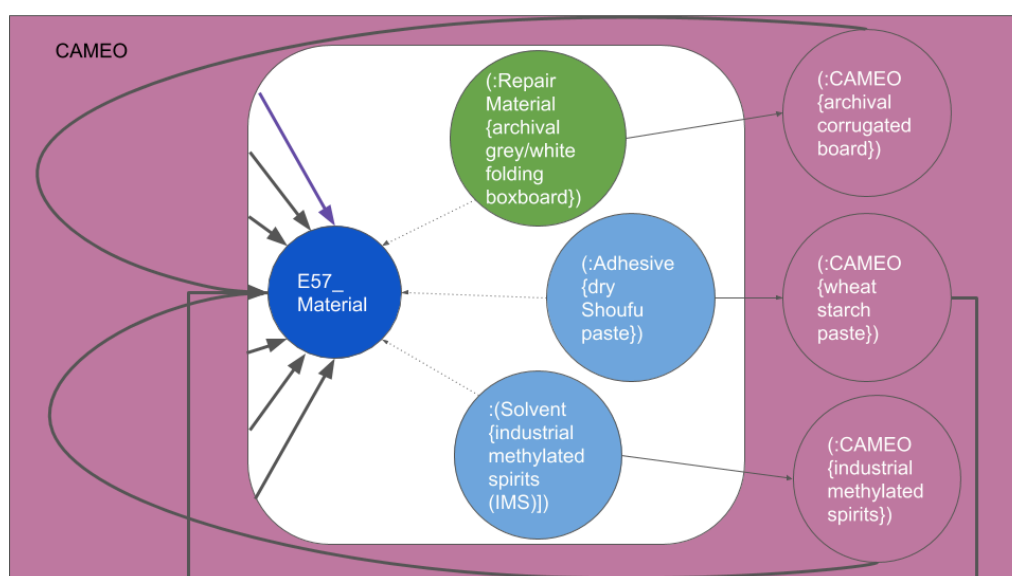


Figure 7.11. Representation of the specific to more general semantic relationships between data nodes, CAMEO nodes and CRM node.

Figure 7.11 demonstrates the situating of CAMEO nodes between instance nodes (the green and light blue nodes in the figure) and an ontology node (the dark blue (:Class{rdfLabel:"E57_Material"}) node) as an example of the cyclic modelling framework mentioned previously in section 4.3.3 (*Categorical Representation and Graph Enrichment*) and specifically in Figure 4.10. The dataset-derived instance nodes for (:RepairMaterial), (:Adhesive), and (:Solvent) are modelled as a path to E57_Material using a relationship edge that simulates the activity of mapping to the CIDOC CRM (represented here using dotted arrows). In this cyclic modelling framework, the instance nodes can also be mapped directly to their categorical CAMEO nodes:

(:CAMEO{entity:"archival corrugated board"}),
(:CAMEO{entity:"wheat starch paste"}), and
(:CAMEO{entity:"industrial methylated spirits"}), respectively.

In theory, traversing in one direction, using a shortest path to a CAMEO node (in this example, path length of 1), provides specificity as to the type of material. Traversing in the other direction, using the shortest path to the nearest ontology node (in this example, path length of 1), provides a much broader class identification.

Although the labelled property graph model does not have in-built automated inference capability in the same way RDF does for the Semantic Web, it can simulate inferential activity through the structure of the graph and the query. Therefore, the addition of the CAMEO dataset as a subgraph improves enrichment potential and enhances further query-based inference capabilities.

7.2.5 Overview of the Resulting P3-LPG Schema

The lowest level depicted in Figure 7.12, labelled “TNA .csv”, is the direct, instance-level data from the original TNA CCD dataset transformed into LPG using a star schema basic structure. This nascent level consists of all instance-related data specific to each treatment event.

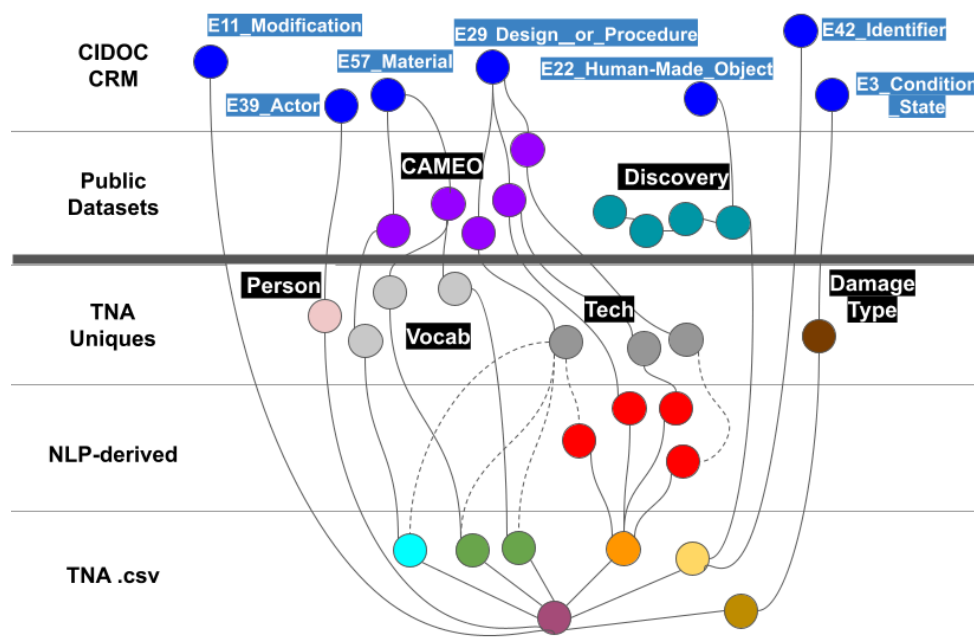


Figure 7.12 The P3- LPG Schema.

Table 7.2.1. Simulated Mappings to CIDOC CRM via paths.

TNA Dataset Node Label	CIDOC CRM Class Node
(:TreatmentEvent)	E11_Modification
(:Reference)	E42_Identifier
(:RepairMaterial)	Intermediate categorical node(s), then E57_Material
(:Adhesive)	Intermediate categorical node(s), then E57_Material
(:Solvent)	Intermediate categorical node(s), then E57_Material
(:Person)	E39_Actor
(:PrimaryDamage)	Intermediate categorical node(s), then E3_Condition_State
(:SecondaryDamage)	Intermediate categorical node(s), then E3_Condition_State
(:Vocab)	Intermediate categorical node(s), then E57_Material
(:Tech)	Intermediate categorical node(s), then E55_Type

The next level up is the NLP-derived level which has nodes with potential to aid enrichment but are neither semantically the original data itself, nor the higher-quality and higher confidence data derived and compiled in the next categorical levels above. The NLP-derived level is also a conceptual level where derived content is stored and provenance remains traceable (via direct paths to :TreatmentEvent node) but not contextualized for wider consumption and use (e.g. not for direct linking to other instance-level datasets). By making (:Comment) and (:ConditionComment) nodes separate from the original (:TreatmentEvent), this enabled attaching all related NLP-derived content to these 'instance' nodes. The benefit was threefold. It meant that it didn't mix the intended length 1 neighbours of (:TreatmentEvent) which were instances related to the event, with data about the instances that contributed to the event, which sets it as a second order relationship. From a visualisation perspective, the NLP-derived tokens and chunks would often if not always create larger clusters than the particular instance nodes of the treatment. This would have altered the treatment hub node's performance measures under centrality and community detection (such as Triangle Counts and Clustering Coefficient) algorithms. For example, a treatment node that had many textual comments would have higher degree centrality than a node with no text-related relationships and a few materials, and other instances. However, some of this may be alleviated by running on projections that leave out :NounChunk and :Verb nodes. Although, in doing so, it would mean accepting default native projections may be skewed and therefore less reliable.

The middle tier as shown in Figure 7.12 is the TNA Uniques level, which is the conceptual level of local or institution-specific named entities. These were compiled from unique values that repeatedly occur in the dataset. This is the level of abstraction and generality that most queries are expected to come in on before drilling down to more specific instances and/or not including those queries that are specific to begin with, i.e. that start from known reference or treatment event numbers. This level allows for the aggregation of instance-based data to quickly derive collection-specific categorical terms, such as for materials (:Vocab) or techniques (:Tech), for use as repositories of local-specific data (such as manufacturers details per material type) or as local thesauri which can be reviewed, assessed and linked to more broad or formal thesauri (see Figure 7.13). For example, while the (:Adhesive), (:Solvent), (:RepairMaterial) nodes as mapped from the original TNA CCD data into instances of materials, the (:Vocab) nodes are a distilled extraction of these (:Material) sets and represents the types of materials used by the institution.

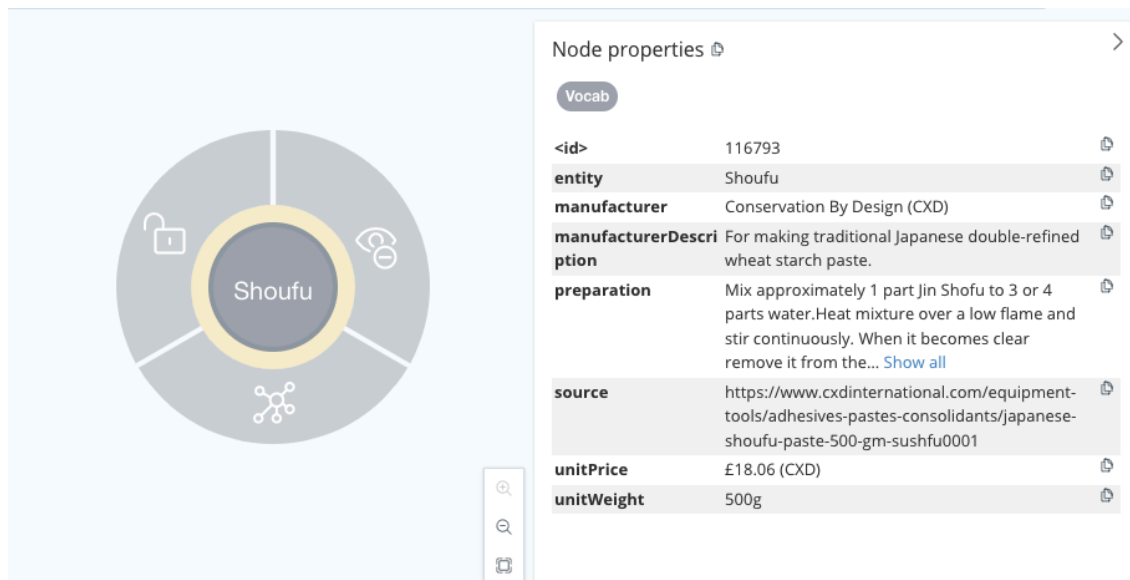


Figure 7.13. Example of a (:Vocab) node derived from distinct materials in the database and subsequently enriched with manufacturer's details as node properties.

The next level, above the thick gray line, is the public or other dataset level which marks the transition from nodes belonging to the original TNA CCD representations and those additional representations that have been derived from the original TNA CCD dataset now connected with other public datasets. Although the TNA Discovery catalogue is the in-house collection catalogue and obviously refers to the same objects and collection materials in the TNA CCD dataset, it is a separately held and managed resource. The diagram here shows four discovery nodes connected in a row, representing a path connection with a level 7 node (:Piece) as the direct connection with a :Reference node, then a level 6 (:Item), a level 3 (:Series) and ,finally, a level 1 (:Department) node.

The final uppermost level in this schema diagram is the CIDOC CRM resources relevant to the case study dataset. This uppermost level of transformed CIDOC CRM RDFS nodes is not necessary for transformation into RDF as the mapping process is declared explicitly as detailed in Appendix I. However, the beneficial nature of including this layer is as a gauge when applying the eigenvector centrality measure. Similar to results in the previous chapters, the expected highest scoring nodes should be in line with the classes most often modelled and therefore are linked upwards through the schema to the CIDOC CRM level. It is also possible to specify in the eigenvector centrality algorithm parameters to ignore (:Resource) nodes so to more accurately assess the data graph itself.

7.3 Results of Graph Theoretic Analysis

7.3.1 Order and Size

Table 7.3.1. Order and Size Results for the Revised LPG Model

	Revised NLP-Model
Order (node ct)*	116793
Size (edge ct)	278870
Node:Edge Ratio	1:2.39
Node:Edge (as quotient)	0.42

7.3.2 Density/Sparsity

Table 7.3.2. Density/Sparsity Results for the Revised LPG Model

	Revised NLP-Model
Edge Density*	0.00002
Leaf Nodes	10292
Isolated Nodes*	2
Leaf + Isolated*	10294
Theta Ratio*, θ	0.0881

7.3.3 Undirected Motif Frequency

As the P3 schema has many more data nodes and edges than the CIDOC CRM RDFS schema and the LCD Datasets, it was always expected to take longer to run each motif query and that the results would be larger counts. The queries were terminated after 72 hours of run time. Therefore at high orders, motif counts become a less practical measure to ascertain.

7.3.4 Eigenvector Centrality

Table 7.3.4 Eigenvector Centrality Results for the Revised LPG Model

	<i>projection</i>	<i>node id</i>	<i>n.label</i>	<i>node Label/CRM Entity</i>	<i>score</i>
P3 - LPG	default	30735	["Human-Made Object"]	["Resource", "Class"]	0.26015 20041

The highest scoring nodes under the eigenvector centrality measure, in order, were E22_Human-Made_Object, E42_Identifier, and E11_Modification. Beyond the top three scorers, the next largest group of high scorers are :Reference nodes. This corresponds to expectations given the composition of the original dataset and the enriching datasets that besides the three CIDOC CRM resource nodes, the next group of highly transitive influencers would be the Reference identifier nodes themselves as they link the conservation data with the Discovery data, and multiple treatments can pertain to the same object/Reference.

7.4 Transforming the Phase 3 LPG Graph to CRM-mapped RDF graph

The practice of mapping is assigning concepts from one schema onto another schema representation whereas modeling is constructing a representation that is as representative of the known or real as possible. Transformation of LPG to RDF, and vice versa, is a mapping exercise from a concise model structure to a more diffuse structure. There is an elongation of paths and expansion in data volume under RDF. For illustration purposes, take this sentence:

My daughter, Maya, initially refused to have a bath last night, but quickly acquiesced when she realised our dog could join her.

If this were re-written using only a subject-predicate-object structure to convey the same semantics, the text elongates to:

My daughter is Maya. Maya refused her bath. The refusal was initial. This was last night. She acquiesced later. She acquiesced quickly. Acquiescence was due to realisation. We have a dog. The dog could join her. The realisation pertained to the dog joining her.

The encoding of this anecdotal situation in words elongates from 22 words and 4 punctuation marks (as these also connote meaning) to 44 words and 10 punctuation marks if constrained to a subject-predicate-object syntax.

Using the Neosemantics plugin, LPG content can be exported as RDF in one of four ways¹²:

1. By the node identifier (ID or URI)
2. By the node label and property value
3. By using Cypher
4. By exporting the graph ontology

Instead of creating a fully CRM-mapped database, we can specify which elements in our database we want to be CRM-mapped by using the 'export by using Cypher' option, as depicted in Figure 7.14.. (See Appendix I for details).

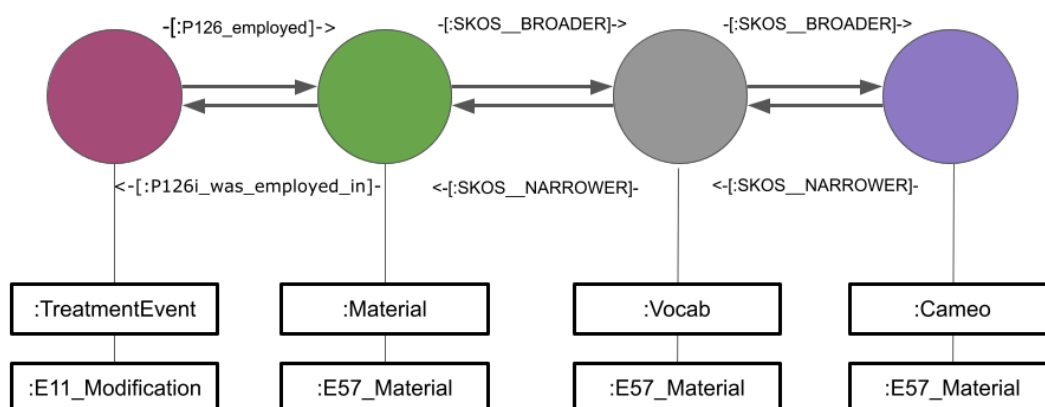


Figure 7.14. Example mapping for transformation to RDF.

E57_Material remains a challenge to map as it is technically a categorical or type class and not a class of instances, such as E18_Physical_Thing. However, to assign the instance of material use (represented by the green (:Material) node to any other class would forego the CRM property/relationship P126_employed(P126i_was_employed_in) which has been defined to have E11_Modification as its domain and E57_Material as its range, and remains the most appropriate relationship. A further alternative would be to "double declare" as the SUL dataset showed where a E57_Material was at times also an

¹² *Exporting RDF data—Neosemantics*. (n.d.). Neo4j Graph Data Platform. Retrieved May 27, 2023, from <https://neo4j.com/labs/neosemantics/4.3/export/>

E19_Physical_Object. This conundrum remains in the CIDOC CRM model. However, in the LPG model, it can be clearly distinguished between instances related to an event (the lowest tier in Figure 7.12), and a collection of those instances into categorical types (the middle tier in Figure 7.12 labelled “TNA uniques”).

The RDF output can also have defined namespaces beyond the CIDOC CRM. For example, in Figure 7.14, the diagram uses the SKOS namespace for “broader” and “narrower” relationships. However, to output a wholly CRM-consistent RDF encoding, these can also be mapped to P127_has_broader_term(has_narrower_term).

7.5 Verification, Validation and Calibration

As this chapter and its related Appendices (B, C, D, E, H, and I) attest, many decisions have been taken in the process of building these models. Even if we frame these models in George Box’s terms and consider them all to be wrong but still want to determine if some are useful, it is paramount to also acknowledge that:

“[Data cleaning] is conceptual modelling in another sense” (Guizzardi 2021).

In order to validate these models and determine their usefulness, it is imperative to build in means for data provenancing. This not only aids the modellers in assessing and evolving the graphs, but it also assists local end users. The data provenancing techniques used in the LPG models include the use of IDs to trace back to the original raw data sources, for example, the use of RowIDs to trace back and check the original CSV entry, if necessary. Furthermore, the inclusion of all treatment content on the (:TreatmentEvent) node enables links to be verified, which improves confidence in the model while allowing for ad hoc error discovery and correction. As data repositories get bigger, and datasets combine and grow, it will not be possible to check each connection. To be able to check on the fly ensures models are built with transparency and corrections can be made that don’t interfere with or disrupt the whole system.

The data cleaning decisions were informed by small-scale models created as part of Phase 1 (see Appendix H). The validation questions are derived from the TNA research interests (see section 4.6.2), specifically regarding quantification of materials, techniques, and individual objects and trends interpreted as quantification over time, i.e. frequency, and historical trends, and to identify any trends specific to individual objects and

collections or departments. Validation questions from Phase 1 (see Appendix H) that can be applied to the revised final LPG model include:

- VQ1. Which treatment materials were most often used?
- VQ2. Which techniques can be identified?
- VQ3. Are there patterns/frequencies of material or technique used over time?
(including clustering of materials/techniques within a specific temporal range?)
- VQ4. Are there patterns/frequencies in objects returning for treatment?
- VQ5. Are there patterns/frequencies by departments or collections that require conservation?

The focus of these questions are in line with TNA's interests as outlined in section 4.6.2 *Model Validation*, particularly regarding materials-based queries and patterns.

The resulting RDF version was run through the W3 RDF Validator tool (<https://www.w3.org/RDF/Validator/>). However, despite the tool being named a "validator", this tool serves to ensure the code is RDF compliant and whether or not it will run properly. Therefore, this step is actually a means of code verification and not for content validation. Thus far, validation is still best carried out using queries derived from validation questions.

7.6 Summary Findings

The P3 LPG model demonstrated how conservation data can be modelled as a labelled property graph and how easily data can be integrated and enriched. It has also demonstrated how representational levels from specific to general can be more clearly defined by a combination of the node contents and by the relationships the node has, which as the reader may recall provides more direct access to the tuples defining the sets, instead of only defining a node by its domain or range.

8.0 Discussion

8.1 Thesis Summary

This research was premised upon the findings and key arguments from two seminal publications addressing the conservation profession. The first is the *Charting the Digital Landscape of the Conservation Profession* report by Zorich (2016) which surveyed the wide variety of digital content and data repositories in conservation and highlighted key problems due to the sheer breadth and diversity of these resources. In essence, the problem is that resources are not joined up. The other publication is the reflection piece by Otero (2022) in honor of the retirement of Prof. Charola, Emeritus Research Scientist at the Museum Conservation Institute of the Smithsonian Institution. Otero's is a forward-looking piece purposefully provocative as a call to action. The "urgent need to develop new strategies to organize, summarize, and disseminate existing knowledge... [via] a sharing information network" (ibid.), in other words, is a Knowledge Graph (KG).

This thesis demonstrates how graph models are well-suited for managing, analysing, and integrating conservation data and that graph-based encodings are themselves a form of documentation that are human-readable and machine-readable [Aim 1]¹. The proposed graph representation method (chapter 4) for conservation consists of the following modelling principles:

1. Use a representational basis for the data model(s)
2. Use a "set"-aware basis for modelling (Sets, Tuples, and Subgraphs) - to identify which sets are nodes, which are relationships, and what sets will be properties, etc.
3. Use a categorical representation to enable graph enrichment and achieve multi-level representation
4. Accept "property"-awareness, where attributes and relationships are distinct, as defined by the rules of Structure Mapping Theory (Analogical Reasoning)
5. Use a star schema subgraph structure (expected and planned)
6. Use graph theoretical analysis to assess and document models at various stages of iteration
7. Leverage the graph structure for query-based analysis
8. Work towards integrated model verification, validation and calibration (WC) practices

¹ §1.3.1, p.4-6

An assessment of the CIDOC CRM RDFS serialisation graph was undertaken which informed 'extract, transform, and load' (ETL) procedures and conservation knowledge graph construction using a labelled property graph (LPG) model. This enabled knowledge graph construction without the need to first map to the CIDOC CRM, although it also supported subsequent onward transformation into CRM-mapped RDF triples, if necessary. The LPG-based knowledge graph served as a metamodel for the creation of the more abstract RDF triples [Aim 2].

The following measures were found to have diagnostic [Aim 3]² applicability:

- Leaf Node Count,
- Eigenvector Centrality,
- Triangle Count,
- Motif Frequency Count,
- Planarity via the presence or absence of bipartite $k_{3,3}$ and K_5 subgraphs,
- Diameter.

In particular, leaf node detection and eigenvector centrality showed the most promise as validation tools for identifying errors in modelling when results deviated from what was expected. Comparing motif frequencies across the four LCD datasets revealed commonalities and distinguishing attributes for each institutional dataset including 'signatures' of different modellers and different documentation practices, for example, the Stanford University Libraries (SUL) dataset was identified as an outlier. Finally, the presence of $k_{3,3}$ bipartite graphs as evidence of non-planarity in both the CIDOC CRM structure itself and in the LCD datasets strongly confirms the multidimensional nature of conservation while also offering downstream research opportunities in studying different facets of complexity (i.e. such as sampling by taking 2D slices of a 3D or higher graph).

Applying more than one graph theoretic measure provided complementary analyses with insights into different aspects of a graph. For example, the Bodleian dataset of the LCD Group had a comparatively high global triangle count which indicated areas of the graph with high-connectivity, while a long diameter showed there were areas of the same graph that were isolated and less well-connected.

The NLP-derived dataset (chapter 7, section 7.2.2) demonstrated how natural language processing (NLP) can be used to access free-text and add further layers of qualitative

² §1.3.3, p.6-7

(e.g. parts of speech) and quantitative information (e.g. word frequencies). The resulting enrichment nodes can aid in creating named entities lists to inform or revise vocabularies and help build high-quality conservation-specific training corpora.

The analyses undertaken and presented in chapters 5, 6 and 7 demonstrate not only how graphs address the integration challenges identified by Zorich (2016) through its compatibility with Semantic Web standards but also compatibility with wider computational thinking methods such as data modelling and querying for problem solving. The graph representation method as described in chapter 4 provides a means to model, analyse, and verify/validate data via graph theoretic means which not only addresses the low rate of data science usage in conservation as highlighted by Otero (2020) but it also demonstrates how to correct and improve from this position. The crux of what's needed in conservation – when we combine the state of the profession as presented by Zorich and the call to action by Otero – is to better leverage what we collect and what we already have. We need a network-based approach to achieve joined-up understanding. Such a Knowledge Graph or networked approach has both epistemological and methodological implications. Thus, the Research Questions (RQs) were framed to address this confluence of epistemological and methodological factors. The next section will reflect on and interpret the answers to these research questions.

8.2 Reflections and Interpretations

To reiterate, the research questions were:

RQ1. How do we build a conservation knowledge graph (KG)?

RQ2. How can knowledge graph construction clarify the nature of complexity in conservation?

RQ3. What are the affordances of graph-based analysis for conservation?

RQ1 centered on the “how?”. Not only how to technically achieve knowledge graphs, but are there good or best practices in building KGs? Addressing RQ2 involved working along the Computational Thinking Framework (see Figure 8.1). A full and detailed roadmap of this was not the aim and would be beyond the scope of this first effort, but nevertheless, RQ2 sought to identify a clearer trajectory in practice from Modelling Practices, through to Computational Problem Solving, and on towards Systems Thinking Practices. Finally, RQ3, in essence, is interested in the gains that can be achieved within the scope of this study in terms of knowledge and methods of practice.

Data Practices	Modeling & Simulation Practices	Computational Problem Solving Practices	Systems Thinking Practices
Collecting Data	Using Computational Models to Understand a Concept	Preparing Problems for Computational Solutions	Investigating a Complex System as a Whole
Creating Data	Using Computational Models to Find and Test Solutions	Programming	Understanding the Relationships within a System
Manipulating Data	Assessing Computational Models	Choosing Effective Computational Tools	Thinking in Levels
Analyzing Data	Designing Computational Models	Assessing Different Approaches/Solutions to a Problem	Communicating Information about a System
Visualizing Data	Constructing Computational Models	Developing Modular Computational Solutions	Defining Systems and Managing Complexity
		Creating Computational Abstractions	
		Troubleshooting and Debugging	

Figure 8.1. Computational thinking practices employed in this research (highlighted in light green, with the predominant contribution highlighted by the yellow outline). Framework after Weintrop et al (2016) and Marciano et al (2019).

8.2.1 (RQ1) How to build a conservation knowledge graph?

A conservation knowledge graph can be built by:

- using RDF/Semantic Web technologies,
- employ a labelled property graph (LPG) database,
- or a combination of the two where the more cognitively intuitive LPG model is used as a metamodel or 'in-progress' graph to integrate data, analyse data and to test modelling decisions. Results can then be transformed into the more diffuse RDF structure for wider publication using Linked Data/Semantic Web standards.

As an ontology, the CIDOC CRM is itself a knowledge graph, albeit one that represents a highly abstracted network of cultural heritage relationships and entities. While the CIDOC CRM is the recognised interchange standard for cultural heritage, implementation of the CIDOC CRM has found recurring challenges from adoption to model validation. The use of an LPG metamodel provides an alternative approach to building such models that nevertheless aids downstream transformation to CIDOC CRM RDF.

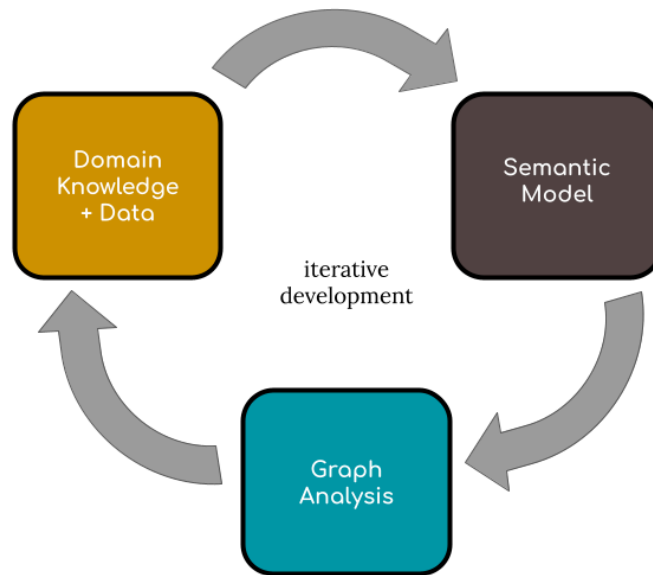


Figure 8.2. The iterative nature of knowledge graph development.

Thus far, organising data of instances (particulars) onto the CIDOC CRM schema framework (i.e. mapping) has been the primary method for structuring heritage data into knowledge graphs encoded for the Semantic Web (i.e. as RDF), as exemplified by the LCD dataset (chapter 6). However, as this research has shown, there are several issues with this method:

- While code verification exists via semantic web tools but there are no data validation techniques for catching mapping or modelling errors.
- The E57_Material problem of using a categorical or type class when the intention is to model instances of materials used.
- It is difficult to adopt and implement the CIDOC CRM as modelling and mapping remains largely a time consuming and manual process (e.g. LCD was mapped manually) even with tools such as 3M (Mapping Memory Manager).

Using an LPG structure as a metamodel enabled ways to address these issues. Chapter 5, section 5.5 demonstrated how the first-order logic (FOL) statements underlying the CIDOC CRM can be encoded and applied as a code verification and import validation procedure. The eigenvector centrality measure also proved highly diagnostic for detecting modelling errors which led to calibrated ETL procedures (chapter 5, section 5.2 and Appendix H)

In regards to the E57_Material problem, the LPG metamodel allows the user to model instances as they are known before needing to assign a class or “double declare” (see Chapter 7, section 7.4, p.215 and as depicted in Figure 7.14).

Although RDF and the CIDOC CRM have been promulgated as strong candidate solutions for data integration, it had yet to be acknowledged in a conservation context that the graph is the basis for both. The Linked Conservation Data (LCD) RDF datasets provided an example of how CIDOC CRM-mapping influences an RDF dataset. Nevertheless, manual mapping is susceptible to discrepancies and errors as revealed by the graph theoretic analyses.

Could we create a knowledge graph without the CIDOC CRM? Chapter 7 explored how this could be possible by enriching a graph of treatment records with multiple authority resources (in this case, Cameo and Discovery API) to simulate cross referencing. By adding more direct access to key terms in free-text extracted from the records using NLP resulted in a more connected graph with improved queryability. Access to content included retrieving themed subgraphs such as materials graph and objects graph. These themed subgraphs offered a solution to the validation circumstances set by The National Archives to aid understanding materials uses and treatment patterns.

8.2.2 (RQ2) How can knowledge graph construction clarify the nature of complexity in conservation?

The practice of knowledge graph construction and problem solving using graphs contribute towards moving the profession from a generalised and abstracted acknowledgement of complexity to an improved and more refined articulation of it. For example, as the $k_{3,3}$ results have shown, conservation data tends towards being non-planar, a clear confirmation of multidimensionality indicative of complexity. Therefore, an affordance of a graph theoretic approach is a richer language for discussing complexity as well as a means to quantify (e.g. triangles and motif counts, diameter) and qualify (e.g. positions of nodes in the context of the wider graph, such as leaf nodes) different aspects of complexity.

Adopting a computational thinking framework (after Weintrop et al 2016, and Marciano et al 2019) acknowledges the depth and breadth of the conservation endeavour while flexibly supports different computational and reflexive practices that work towards elucidating those complex systems (physical-chemical, temporal, social, etc.) that

Table 8.2.1 Examples of tasks undertaken during the research that matches each computational thinking practice category and sub-category.

Data Practices	Modelling & Simulation Practices	Computational Problem Solving Practices	Systems Thinking Practices
Collecting Data	Using Computational Models to Understand a Concept	Preparing Problems for Computational Solutions	Investigating a Complex System as a Whole
<ul style="list-style-type: none"> Accessing encoded authoritative resources (e.g. Cameo, Discovery) 	<ul style="list-style-type: none"> Encoding KG to simulate knowledge aggregation and aid data search 	<ul style="list-style-type: none"> Prepare extracted data for direct query unhindered by in-house RDBMS user interface and the system's fixed queries. 	
Creating Data	Using Computational Models to Find and Test Solutions	Programming	Understanding the Relationships within a System
<ul style="list-style-type: none"> NLP-derived dataset of TNA terms 	<ul style="list-style-type: none"> Use of LPG and RDF versions to check models using graph analysis and graph query 	<ul style="list-style-type: none"> Programming in Cypher (for graph building, querying and graph analysis) and Python (for data extraction from HTML and GET requests, NLP) 	<ul style="list-style-type: none"> Applied structure mapping theory (SMT) to model relationships and attributes
Manipulating Data	Assessing Computational Models	Choosing Effective Computational Tools	Thinking in Levels
<ul style="list-style-type: none"> via Cypher graph queries 	<ul style="list-style-type: none"> Graph analysis across case studies, esp. the use of Eigenvector analysis 	<ul style="list-style-type: none"> Semantic Web tools (RDF) Graph database (LPG, Cypher) Python (spaCy, BeautifulSoup) 	<ul style="list-style-type: none"> Modelling LPG to include particular, categorical and universal levels
Analyzing Data	Designing Computational Models	Assessing Different Approaches/Solutions to a Problem	Communicating Information about a System
<ul style="list-style-type: none"> via Cypher graph queries 	<ul style="list-style-type: none"> Phase 1 Trials (Appendix H) Proposed Graph Representation Method 	<ul style="list-style-type: none"> CIDOC CRM-mapped vs unmapped graph modelling Linear graphs vs star schema for modelling NLP-derived tokens/chunks 	<ul style="list-style-type: none"> K3,3 bipartite graph analysis to confirm complexity (nonplanar)
Visualizing Data	Constructing Computational Models	Developing Modular Computational Solutions	Defining Systems and Managing Complexity
<ul style="list-style-type: none"> via Cypher graph queries 	<ul style="list-style-type: none"> Transform CIDOC CRM RDFS and LCD RDF datasets to review in LPG 	<ul style="list-style-type: none"> Reusable transformation scripts/coding Star schema Thematic Cypher queries 	
		Creating Computational Abstractions	
		<ul style="list-style-type: none"> Graph analysis to find patterns and anti-patterns (e.g. Triangle Count, Diameter, Leaf Nodes) 	
		Troubleshooting and Debugging	
		<ul style="list-style-type: none"> Used throughout modelling and programming; iterative revisions added to VVC procedures 	

pervade conservation work and decision-making. As Marciano et al (2019) demonstrated how archival science can adopt a computational thinking framework, this research has demonstrated how conservation can likewise adopt such an approach as part of the profession's computational turn. A graph-aware approach to computation immediately engages 'Modeling & Simulation' and 'Computational Problem Solving' techniques into conservation research and practice which, in turn, contributes towards more complexity-elucidating research. Table 8.2.1 is an annotated version of the computational thinking framework with examples from this research.

8.2.3 (RQ3) What are the affordances of graph-based analysis for conservation?

The data mining opportunities afforded by a graph-based approach include the improved queryability of a path-based query language like Cypher and the seamless use of natural language processing in tandem to capture named entities and derive further relationships with existing sources. Additional benefits of the proposed graph representation method include:

- facilitating data exploration and enabling views of the data content from different themed perspectives (e.g. Materials Graph. Objects Graph, etc.),
- providing a flexible structure that supports varying degrees of semantic representation from the specific to the general,
- combining resources together to simulate cross-referencing within a single system at the bench.
- Prototypical modelling at the bench-level can be used to inform future DBMS upgrades.
- Graph theoretic analysis can be applied to any dataset and is not hindered by format, content or structure of the original dataset, albeit data preparation will be necessary.

The graph-based review of existing conservation RDF datasets and the graph theoretic analyses of these datasets provided qualitative and quantitative insights on existing data collecting practices and data modelling practices as well as where adjustments or improvements can be made for both. For example, the distinct differences between the SUL dataset from the BOD, LOC, and TNA datasets in the LCD Group would have been impossible to identify from manually reviewing the encoded TriG files alone.

Graph-based analysis is available for immediate deployment at the bench or studio level. As this study has demonstrated, it is not necessary to have full deployment at the CMS or institutional level to use and benefit from graph analysis and to build semantic

computational models and metamodels. The advantages of working at the bench-level scale at this point, is the low cost of using this technology and the opportunities for upskilling. Starting at bench-level allows for a bottom-up development where conservators are directly informing the analysis and model building. Trial and error can occur locally without institution-wide investment nor disruption to existing processes. Yet, the aggregation of knowledge, practice and experience has the potential to better inform institution-level CMS changes in the future³. It is sensible and advantageous to create and identify the models that are of use and to allow time to iterate, re-use and feedback on these models before larger changes are implemented. When changes are to be made, there will be better evidence to identify the most appropriate direction.

Furthermore, the value added by using graphs is its compatibility with existing data and flexibility for future data. It is foundational for more advanced analytical approaches that are at their core, graph-based, such as machine learning or other artificial intelligence approaches (see section 8.6 on Future Work).

8.3 Implications

8.3.1 The Craft of Modelling

Hendler, Gandon and Allemang (2020, 11) frame modelling as a craft:

“[The working ontologist’s] craft is to make sensible, usable, and durable information resources from this medium [i.e. Semantic Web technologies]. We call that craft modelling”

The use of the term “craft” is telling of the work involved. It is a process that requires a series of ongoing microdecisions, applied from a gradual building up of tacit knowledge through doing. “[Data cleaning] is conceptual modelling in another sense (Guizzardi 2021). Therefore, data cleaning⁴ and data wrangling⁵ both influence the final model. It is important that conservators play a role in building and testing these models for data management and data integration so that domain knowledge is not lost. Using

³ For example. multi-modal data management systems (which combine SQL and NoSQL technologies) exist and are commonplace in enterprise scenarios.

⁴ Data cleaning is defined as the process of removing inaccuracies or inconsistencies in data (McKinney 2018).

⁵ Data wrangling is “process of manipulating unstructured and/or messy data into a structured or usable form” (McKinney 2018, 14).

metamodels as “in-progress” graphs and building towards formal knowledge graphs allows conservation practitioners to hone our understanding while building tools that are necessary for the profession at large.

The labelled property graph (LPG) was used as a general purpose modelling tool where data was easily accessed and analysed outside of rigid in-house database systems or RDF/SPARQL’s limited tools and application functions. This made it possible to trial data science and other computational methods for interrogating and visualising the data.

Data provenance was built into the model for transparency, to increase end user confidence in the model and to aid future troubleshooting. For example, an additional column and heading, ‘RowID’, was created as a unique identifier for each specific instance of a conservation treatment as structured in the original TNA CCD spreadsheet. This allows a user to verify the model’s components when encountering unexpected query results in real time. As documentation derives from the modeller, it is pertinent to build in data provenance⁶ meta-content to inform the user or provide a means for the user to question the model, verify the data and scrutinise whether its position in the graph is sound.

For example, for the Phase 3 graph using the TNA CCD dataset, all (:TreatmentEvent) nodes were modelled with the full original record as node property key-values. The “pulled out” nodes (e.g. those deriving from the spreadsheet’s headings) enabled modelling of these key-values as nodes themselves along paths. By having both, a user can check to see if the relationships were modelled correctly, via the central (:TreatmentEvent) node, which itself bears a RowID value that corresponds to the original data. Another TNA CCD example is being able to check NLP-derived token and chunk nodes in the context of the original free-text and other fields of the record along a direct path back to the central (:TreatmentEvent) node and to any categorical nodes. Proximity to the (:TreatmentEvent) node increases relevance to that event. Likewise, proximity to a categorical node increases the relevance of specific instances with that category.

Figure 8.3 shows a random sample of the TNA CCD dataset generated during the Phase I trials (see Appendix H) where each row of data (i.e. a treatment) was depicted in star

⁶Data provenance is defined as a “documented trail that accounts for the origin of a piece of data and where it has moved from to where it is presently...The concept of provenance guarantees that data creators are transparent about their work and where it came from and provides a chain of information where data can be tracked as researchers use other researchers’ data and adapt it for their own purposes.” (National Library of Medicine, n.d.)

schema. The competency questions suggested by TNA were already answerable at this stage. In fact, many of the competency questions were already answerable immediately after import when treatment rows were transformed solely into unconnected nodes with column data set as node properties (see Appendix H, Section H2, Model A). These earliest stages for developing a knowledge graph revealed that the TNA CCD dataset was not highly-connected enough by itself with each treatment represented as isolated stars. While all treatment aspects were queryable in principle, the knowledge graph lacked more generalised categories to aid search and fully take advantage of a path-based querying paradigm.

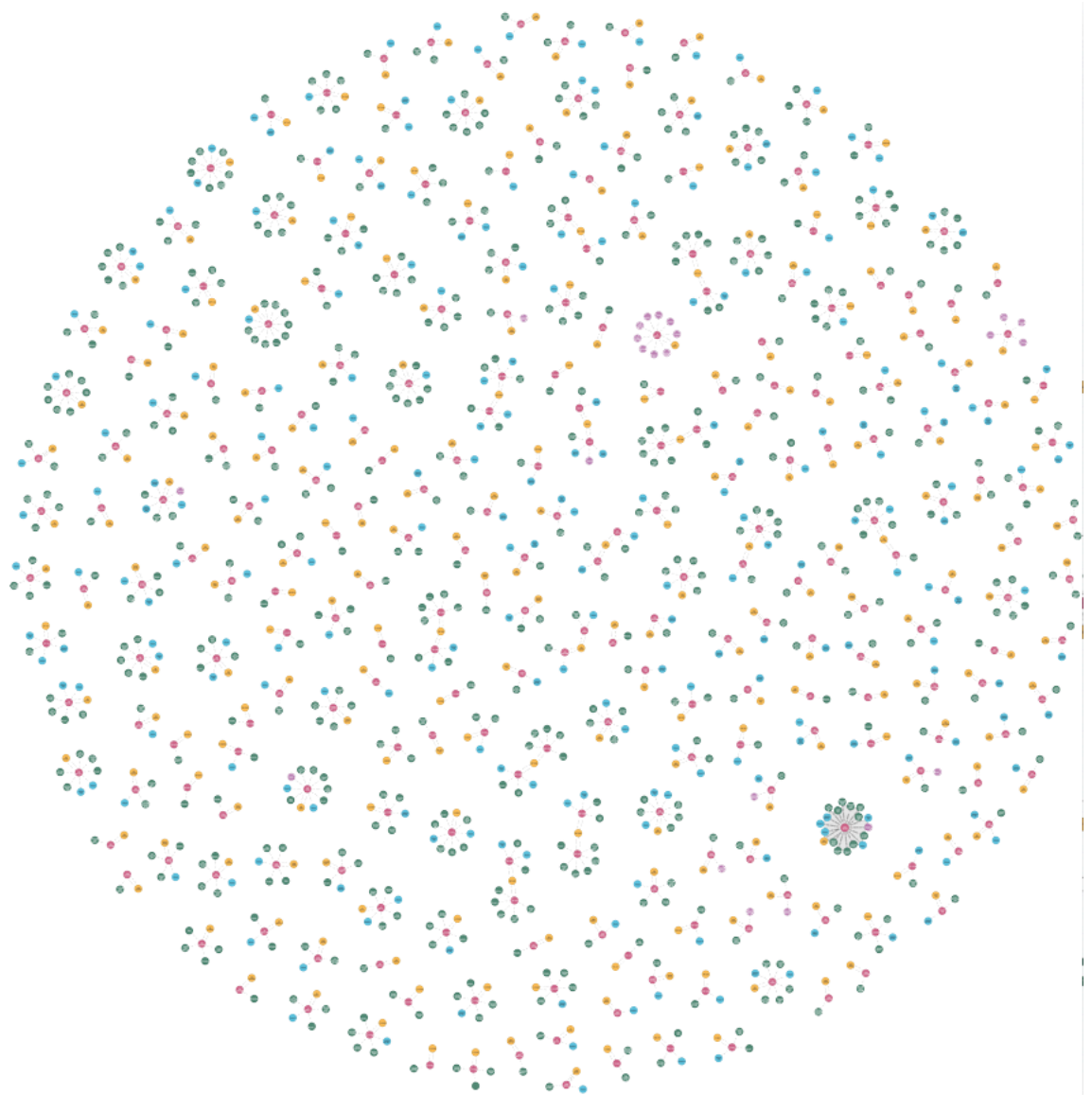


Figure 8.3 View of a random sample of the TNA CCD dataset⁷

⁷ The edges are hard to see at this resolution. Figure H2.4 in Appendix H shows a small sample of 10 treatment star schema graphs with more visible edges. For node labels, see Appendix B. For edge labels, see section B4 in Appendix B.

Searching via node properties in this way without mediating categorical nodes, such as Vocab nodes or Cameo nodes, increased the complexity and length of queries in terms of the number of clauses and functions used per query while it lowered the overall search results. Without mediating categorical nodes, misspellings, for example, would not be included in the results. Therefore, relationships to categorical nodes not only serve to represent meso-levels in semantics, they also provide additional paths for searching and serve as a means to correct or compensate for discrepancies and differences in natural language inputs, which in turn improves overall knowledge graph connectivity and provides a more accurate representation of how a domain specialist perceives, aggregates, cross-references and otherwise uses data.

For the TNA CCD dataset in chapter 7, there was a need to access content from the “Comments” and “Condition Comments” fields. This content contained data dumped from previous system migrations where the new system lacked specific fields to match and contain previous content. Such content became aggregated unstructured data over time which included expository text and was not searchable via the existing Collections Care Database (CCD) system. Nevertheless, the full record, including these fields, remain exportable via general purpose .CSV (comma-separated value) files. This enabled experimentation and exploration of the data for more informed and iterative data science pipeline development that included using NLP and matching with more general categorical terms. This procedure can be used to computationally assemble high-quality named entity lists (in JSON) for training conservation or heritage-specific machine learning tools. By having new and processed datasets in FAIR-compliant formats means any such dataset can be the source dataset for further downstream use.

8.3.2 Verification, Validation and Calibration (VVC)

The iterative development process for creating a LPG-based knowledge graph consisted of discoveries and some deadends. These were marked junctures in the modelling process that precipitated decisions resulting in a suite of verification, validation, and calibration (VVC) practices that were pivotal to the development and implementation of the graph representation method. The VVC practices were those decisions and actions that helped to clarify and direct how to clean and wrangle the data and to achieve the aims of the models in both its representational structure and content and its queryability. This also included decisions and actions to ensure the models were robust in their accuracy.

Figure 8.4 has been annotated to highlight visually diagnostic features that led to further ETL refinement. The red boxes show examples of where multiple treatments had been undertaken on the same object, denoted by the presence of 2 star schema groups with (:TreatmentEvent) [in dark pink] hubs sharing a (:Reference) node [in yellow]. The yellow box highlights an eye-catching treatment that uses many materials [nodes in green and blue] more so than the average. While both red and yellow highlighted features can also be found through querying, these emphasise the added exploratory affordances of a visualised representation. Finally, the blue boxes show treatments with unusual degree 1 neighbours, in these cases, multiple (:Person) nodes [in pink] per (:TreatmentEvent). A review of the original TNA.csv showed no cases where multiple persons were associated with a treatment, therefore these features were errors in the model. This led to troubleshooting and refinement of the ETL scripts to avoid such errors, i.e. an example of model validation leading to code verification and ETL calibration.

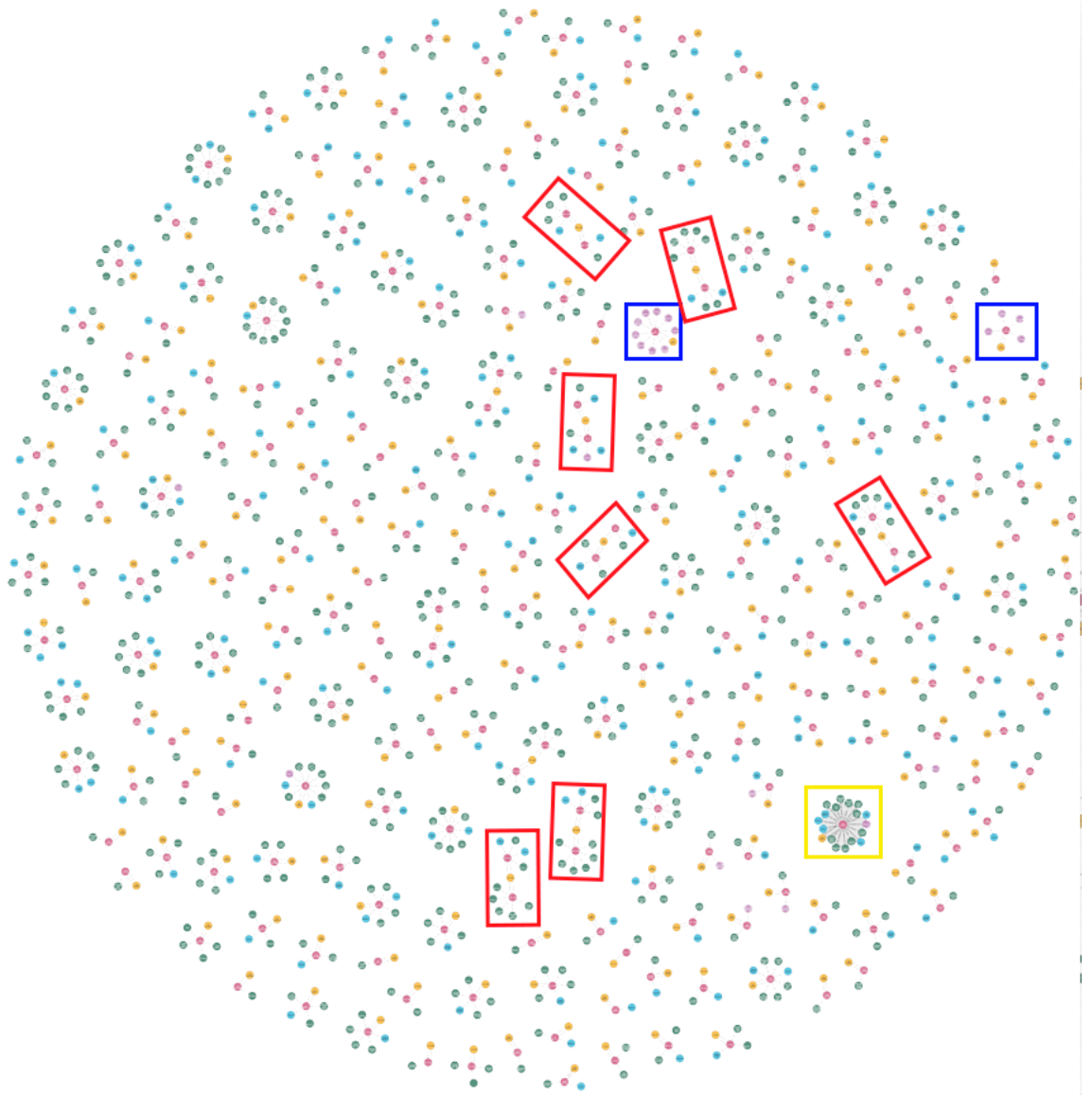


Figure 8.4. Annotated version of Figure 8.3 to highlight visually diagnostic features.

Figures 8.3 and 8.4 demonstrate an example of a general VVC procedure where a resulting graph or sample is reviewed and expected outcomes are compared with actual outcomes. For this research, the VVC techniques used included:

- reviewing node and relationship counts with expected or calculated counts,
- using visual inspection of random samples,
- encoding competency (validation) questions into queries and checking results,
- revising ETL procedures through trial and error

The eigenvector centrality measure, as noted above, was also identified as useful for VVC.

8.3.3 Understanding the CIDOC CRM

While the CIDOC CRM is a standard for interoperability, transforming data into CIDOC CRM-mapped RDF triples has been laborious and without specified tools for verification, validation and calibration of the resulting data models. Since the work was premised on creating a knowledge graph, it was not constrained to transforming into CIDOC CRM in the first instance. Nevertheless, for interoperability purposes, any assistance with transformations would be highly beneficial. Hence, the CIDOC CRM was analysed using graph measures to identify patterns or antipatterns that may indicate a potential validation metric. Analysis across three versions of the CIDOC CRM has shown that evolution of the model involved the removal of leaf node classes from v.6.2.1 to v.7.1.1. Although these changes were arrived at over time through the stewardship of the CRM Special Interest Group without the aid of leaf node detection, by applying such characterisation measures, future modifications of the CRM can be assessed more quickly prior to deployment.

8.3.4 Implications on Conservation Practice

This research has demonstrated how graph analysis can inform our understanding of our collection materials and how it is also a powerful tool for managing our data, assessing our documentations systems and practices and for scrutinising our epistemological models. However, at present, the functionality of graph-based algorithms and graph-based querying sits outside of current in-house museum SQL-based data systems. While multimodal database systems that combine SQL and noSQL databases into a hybrid system exist, these are bespoke enterprise systems developed with greater resources and with dedicated in-house staff to manage the backend technologies and the frontend content.

Nevertheless, as this thesis demonstrates, graph technology is presently available and accessible for free as an at-the-bench productivity tool requiring only a single laptop computer. Experimental data modelling, data exploration and analysis can be undertaken at a manageable resolution of detail to address the interests of conservation teams in day-to-day work.

The implications of this to conservation practice include its influences on the evolving role and skills of the conservator, particularly in the area of coding. To better understand the coding competencies across conservation roles, further research to survey these competencies and roles, akin to the work by Royal and Kosterich (2024) for the field of journalism, is necessary.

Kesper et al (2020) identified three actors involved in creating and applying the patterns in a research data system: the domain expert, the data analyst, and the data engineer. Previously, we accepted that the conservator was the domain expert and, at times, also the data analyst in this multidisciplinary group while the data engineer was positioned outside conservation in the information science sphere. However, such clear distinctions are no longer adequate as more data/knowledge engineering is required within the conservation field. An overlap of skills becomes increasingly relevant. The complex nature of the problems conservators face necessitates scaling up skills. While not every conservator will need to upskill to the same level of competency, nevertheless, a subset of conservators must specialise in knowledge engineering for the domain. Two-dimensional spreadsheets alone no longer suffice, we must work in tandem with multidimensional graphs.

8.3.5 Challenges to Implementation

As mentioned above, one of the key challenges to implementing this method is skills-related. The work requires a wide (but not necessarily deep) skill set in programming, conceptual modelling, graph theory and its algorithms. An openness to upskilling in programming is necessary. Time and practice will assist in gaining conceptual modelling skills and knowledge of graph theory terms, concepts, and algorithms. Like any craft, modelling is a skill that requires time and engagement with the craft to personally develop. A drawback is that data engineering and conceptual modelling are skills not currently taught in conservation training programs. Cultural

assumptions separating computational craft from more traditional crafts risks the othering of technological methodologies and may hinder adoption and upskilling.

Time also presents a challenge for iterative development, which is needed for exploring and preparing data⁸. The end result of data cleansing need not be perfect, faultless data content, which is unrealistic when dealing with real world data. Instead, threshold uses of the prepared dataset should be indicated in the model building documentation to identify the extent of data cleaning so that subsequent users do not risk overfitting their models. For example, in the NLP-derived dataset (see Appendix C, section C6), where “amp” (residual markup artefact in the original free text for *ampersand*) had been incorrectly identified as a noun chunk or verb, these were not removed. Additional cleaning of the dataset was not prioritised in this case as “amp” nodes were unlikely to match any graph enrichment queries for connecting to the other datasets. In this case, these can be identified and pruned at a later stage.

The research also demonstrated how existing conservation data may be very isolated and have too few connections which means graph theoretic pattern identification using existing data capture practices could be limited. Conversely, relationships that are tacitly known by the domain specialist but not explicitly expressed in the data can be reintroduced. This will require further graph enrichment with additional content or revised decomposition strategies for the data into ever-more discrete but still meaningful content to increase relational clarity and explicitness, thereby increasing connectivity.

While this study highlights graphs and new approaches towards computational practices in conservation, this study did not include research into how training and dissemination of these methods to the wider conservation community will have on implementation. Further engagement with the conservation community is needed.

8.3.6 Cultural Assumptions

Cultural assumptions may influence graph modelling and/or, as briefly remarked upon above (in section 8.3.5), present a challenge to implementation. While logic is a culturally agnostic capacity, premises used in logic and the conclusions drawn from those premises can be culturally specific. Categorical distinctions have historically been associated with Western philosophical contexts (e.g. Aristotle, Peter Abelard, etc.). However, Eastern

⁸ Details of data preparation procedures undertaken for each dataset in this study can be found in Appendices B-G.

philosophies, such as the Chinese Mohists and later Confucianists, framed such distinctions similarly as the meaning behind “names” (Willman 2023).

Reference to culture here is not limited to geographical or ethnic culture but also to cultures of work and institutional cultures as these also encompass culture-specific knowledge with practices, habits, behaviours, values and preferences specific to them. Therefore, the distillation of culture as an aspect of knowledge graph modelling will likely result in culturally-informed graphs with variations over time and geographical space. This is not to say there won’t be some overlap in such diverse models. The knowledge graphs demonstrated in this study are snapshots – static representations. However, dynamic knowledge graphs, that are programmatically updated as information changes, for conservation purposes, will still require a mechanism to document networks at retrievable intervals for comparison.

Further discussions of cultural assumptions of the method must also pay heed to the mechanics of technology adoption. For example, the proposition, affordances, and adoption of graph-based modes of working can be discussed from perspectives of or similar to technological determinism (Wyatt 2008) or social construction of technology (Bijker, Hughes & Pinch 2012), that is, does technology make conservation or do people (conservators and other heritage professionals) make the technology? While this research advocates for using graph modelling in conservation as a research and productivity tool, the epistemological directions this takes the profession must be regularly reflected upon. With George Box’s quote⁹ in mind, we must ensure our models are useful, and not *too* wrong.

8.4 Limitations of the Study

This research formally started in the autumn of 2018 and was significantly impacted by Covid-19-related delays. The greatest impact was constraining further follow-up work with case study participants. For example, more in-depth follow-up with the Linked Conservation Data project (chapter 6) to review modelling errors identified through algorithmic analysis was not possible within the timeframe. Further work would include reviewing the original documentation (in .docx or .pdf) and the transformed CRM-mapped XML versions to pinpoint manual transformation issues. For the TNA CCD

⁹ Paraphrased as: All models are wrong, but some are useful. For wider context, see also p. 41 and p. 71 of this thesis.

dataset (chapter 7), direct end-user engagement through workshops with The National Archives staff to trial the LPG-based database would have been another important iterative step in the development of the graph models. Such workshops would present opportunities to gather and gauge user experience feedback and to survey for further validation questions.

While this study focused on introducing graph theory and graph applications to conservation, it does not present a completed software with a user interface. This would require further refinement and software development considerations which are beyond the current scope of this study. Although this study has produced many results using several graph theoretic methods, not all possible insights have been extracted at this time. However, by committing to the FAIR principles raised in Chapter 3 and making the queries and results accessible, it is hoped that further scholarship continues to be gained from this initial work.

8.5 Recommendations

To replicate this work, an awareness of the following would prove beneficial.

- Computational thinking skills in practice
- Familiarity with regular expressions (regex)

The Neo4j system can search via node property values even when the full value is a very long string. By simply using string operators for regex matching (=~), wildcards (.*) and case insensitivity (?i), it is possible to search and retrieve from within free text.

To facilitate ongoing work that builds on this thesis and to assist wider adoption and implementation of graphs in conservation, issues can be raised and addressed via this publicly accessible link: <https://github.com/ana-tam/conservation-graphs/issues>.

8.5.1 Recommendations for Implementation

The extract, transform and load (ETL) procedures from RDF to LPG are straightforward (see Appendix G, section G3). However, the ETL procedures for RDFS requires additional checks to ensure directionality for domain and range attributions are structurally aligned to the semantic scope of the schema. In short:

- Use ETL1 for RDF/XML files
- Use ETL2 for RDF Schema files.

8.5.2 Recommendations for Implementing Graph Theoretic Analysis

It became evident upon reviewing the graphlet/motif analyses results in chapter 6 that the graphlet pattern order used by Przulj (2006) and Espejo et al (2020, see p. 66) where all 3-node, 4-node, and 5-node graphlets were sequenced from lowest edge count to highest, provides a more systematic and sequential overview for profiling purposes than the ordering used by Abouda, Morales, and Aboulnage (2020). This would avoid the need to re-order the motif frequency results as was necessary for analysis and as demonstrated in Figures 6.11 to 6.12.

Due to the calculation resources required, fully comprehensive motif frequency analyses across all 29 graphlet permutations works best on small datasets similar in size to the LCD datasets. For larger datasets, analysis of a smaller sample subset against only a few motif subgraphs is recommended.

8.6 Further Work

This research has demonstrated how graph theory and graph applications can be applied to conservation by focusing on the basic building blocks for creating conservation graphs. Ongoing development in the conservation graph space is necessary to take advantage of the wider tools and techniques afforded by graphs for data analysis and prediction.

The use of knowledge graphs in conservation is a nascent area of study. As the previous section on limitations show, there are many areas to further expand. The value of the research undertaken herein is in providing tools and rationale that will enable this ongoing work to happen, using both LPG and RDF approaches, while ensuring transformations can be revised and re-integrated with semantic coherence. This work has been premised upon the need to integrate different data sources and data models. Therefore, of paramount importance is the need for an ingestion pipeline that can be iterative in its ingestion and validation phases. Further development of conservation knowledge graph(s) would be best served by interoperability between LPG and RDF-style graphs as it enables ingestion and analysis of a wide variety of data and corpora while supporting publication and querying in both formats.

We know from the literature that graphs give us a language for describing patterns of connections which can help quantify interactions and predict what is missing. The work undertaken thus far and presented in this thesis has provided some understanding of these connection patterns and measures for investigating these interactions. However, this has been only a small step on a longer journey and already we can see the onward journey will need to include:

- further integration to enhance cross-searching across datasets,
- testing other graph theoretic measures,
- preparing bibliographic or topic networks,
- machine learning and deep learning models,
- considerations for implementation and the future of conservation practice.

Examples of each area will follow.

8.6.1 Further integration to enhance cross-searching across datasets

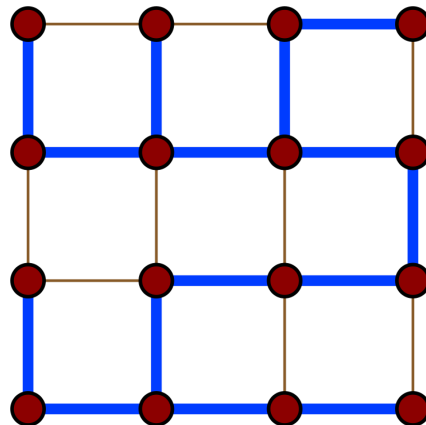
While this study provides an alternative approach to modelling conservation records with heterogeneous resource datasets using an LPG metamodel, it has been a limited case study encompassing a single institution's dataset and reviewing a limited number of existing data graphs. Velios and St. John (2022) have commented on the "considerable effort" required to integrate resulting models from different institutional sources to enable cross-searching. Nevertheless, the aggregation of models from additional institutions is necessary to continue to test the LPG metamodel approach in an iterative manner using graph algorithmic analyses (e.g. eigenvector centrality) for model validation and identifying modelling errors. A data quality assessment method such as that by Kesper et al (2020) should also be applied.

8.6.2 Other Graph Theoretic Measures

Although the motif measures were challenging to employ once the graph reached a certain size, the motif profiles did prove useful in identifying the commonalities and differences amongst small datasets such as the LCD Group and, therefore, motif analysis can still be applied to small samples. However, due to the significant relationship found between the ratios of nodes-to-edges per motif, instead of using the motif ordering of Abuoda et al (2020), it is recommended that the motif ordering after Pržulj, Corneil and Jurisica (2004) be used as this ordering places each motif permutation in order of node count and edge count and thus the progressive ratios are organised more sequentially.

Other graph theoretic measures are available to try and apply. For example there are spanning tree algorithms that can be incorporated into search queries. A spanning tree is a subgraph that is a tree (i.e. acyclic) that includes all the vertices of the graph, G . It can be used to identify paths between any two nodes, such as solving mazes. A spanning tree algorithm can be used to trace out the many routes from a start node to an end node. Parameters for this algorithm can include specifying which edge types and node labels are permitted or avoided on the traversal. In principle, the traversal starts at the start node and proceeds down every route available until it reaches the specified target node or a node it must avoid. If there are no other routes to take, that traversing branch stops and that path is discounted. This algorithm is useful for identifying the distance and the possible paths between two nodes or for ensuring paths can be traversed through specific nodes or edges. This not only allows the user to discover and retrieve

data, it can also aid the modeller as a verification or validation procedure, albeit the procedure is resource heavy.



*Figure 8.5 Visualisation of a spanning tree (in blue) connecting all nodes of the grid graph, G .
Image Source: Eppstein, D. (2007) A spanning tree (blue heavy edges) of a grid graph.
Accessed 23 May 2023 via*

https://en.wikipedia.org/wiki/Spanning_tree#/media/File:4x4_grid_spanning_tree.svg

As mentioned in section 6.3.2, windmill graphs and related graph classes have been espoused by Estrada (2016) to explain the divergence of Watts-Strogatz clustering coefficient and network transitivity when the number of nodes tends to infinity. These classes are defined by having many connected cliques, which are themselves only connected to a single node and are not otherwise connected to the other cliques. Therefore, another small-scale, subgraph structure to explore is the clique (an induced subgraph that is complete).

8.6.3 Bibliographic or Topic Networks

A typical application for graphs is the construction of domain bibliography graphs such as the GraphDBLP (Mezzanzanica et al 2018) of computer science bibliographic entries. These tend to be akin to collaboration networks (Newman 2001) but can also be used to build keyword, topic or co-occurrence graphs to assess relationships between existing areas within a domain. Graph-based analysis of conservation bibliographic networks can provide immediate insights into the lack of aggregated knowledge as lamented by Otero (2022) but also identify adjacencies that may improve connectivity.

This is also an area that may be beneficial to documenting Intangible Heritage, as a way of gathering the immediately related entities adjacent to the intangible, such as oral traditions, performing arts, knowledge and practices concerning nature and the universe,

traditional crafts, ethical and human rights considerations (Convention on the Safeguarding of Intangible Cultural Heritage 2003).

8.6.4 Machine Learning and Deep Learning Models

Working with graphs allows a composite and iterative workflow that utilises more machine learning processes in supervised ways. The use of the spaCy small English model in this study demonstrates at a very elementary level how natural language processing (NLP) can contribute to data integration. Further work to build more conservation graphs from data and textual documents would benefit from using other NLP techniques such as word embedding approaches to build high-quality conservation-specific corpora or sentiment analysis and topic modelling for supervised learning to extract and apply named entities/taxonomies (Dawar et al 2019).

Graph embeddings (Cai, Zheng and Chang 2018) are used to study high-dimensional graphs in a low dimensional environment, which in turn, can inform an evolving underlying graph representation and are used to build deep learning models such as graph neural networks. Applications for graph embedding include link prediction, triple classification (i.e. RDF subject-predicate-object), and graph classification (ibid.). Graphs can also be used for semantic similarity metrics (Zhu and Iglesias 2017).

8.6.5 The Future of Conservation Practice

The use of graphs in conservation can influence and even change existing practices as different ways of working will inevitably alter the language we use to discuss our work. It can fundamentally change how we think about conservation. This speculation, based on the Sapir–Whorf hypothesis (Kay and Kempton 1984), is nevertheless an interesting premise to consider during the ongoing development of graph-based decision-making models and tools.

9.0 Conclusions

This study establishes an introduction to how graph-based data science and data management can contribute to the ongoing needs and demands of the conservation profession by revealing patterns in our efforts. At the core of this approach is the development of graph models (knowledge graphs) with the potential to help us be more reflective in our practice. There are also the practical benefits of processing and analysing our heterogeneous data in a more integrated manner. It allows us to elaborate on records we have already collected and will collect. The work presented here provides modelling principles that are flexible enough for adoption in all areas of conservation.

Approaching conservation documentation as connected graphs addresses the data integration and access problems while providing robust analysis methods and compatible data standards needed by the conservation profession in today's digital and data-heavy world. This thesis demonstrates how graph models are well-suited for such multifaceted roles and how graph-based encodings are themselves a form of documentation that are both human-readable and machine-readable. A graph-based approach contributes towards a much-needed computational thinking framework for the conservation profession that also supports deeper investigations into the nature of complex systems within heritage sciences.

All three research questions were addressed. In terms of "How do we build a conservation knowledge graph?" (*RQ1*), this thesis has shown that a conservation knowledge graph can be built using the Semantic Web/Linked Data standard of RDF or by using the labelled property graph (LPG) model. The LPG alternative graph model is less diffuse in structure and more akin to the cognitive schemas of specialists, yet can still be transformed into RDF, and vice versa, for wider dissemination as Linked Data and for visualisation and analysis. To use an LPG structure as a metamodel in this way serves as both a means to integrate and analyse data while allowing close examination of the results for errors and provides a means to review a model using graph theoretic measures.

Knowledge graph construction enables further research into the nature of complexity in conservation (*RQ2*) as it offers a means to build and test models and simulations while allowing for more flexible and multidimensional data capture. Knowledge graphs are compatible with a computational thinking framework and therefore opens up many avenues for ongoing research and problem-solving. Graphs introduce a language for

talking about highly-connected things. In regards to clarifying complexity in conservation, the ability to describe and provide measures to structural phenomenon reduces the obscuring influence of perceived ineffability that is associated with complexity. This research has demonstrated how the language of graphs can be applied to speak more precisely about the relationships within conservation and developing computational models for conservation will aid in elucidating complexity. The value added in using a graph-based approach for knowledge representation and data management for the conservation domain is that the richness of graph theory helps the profession to articulate different aspects of complexity.

The affordances of graph-based analysis for conservation (*RQ3*) allows for multi-parameter queries, and include combining data science with documentation, and data curation with the building of further research corpora or training sets via the use of graph-compatible systems. The research has demonstrated how algorithmic analysis using graph-based data modelling can yield useful insights from conservation data. These insights can be applied reflectively to inform data collection and management, and query design (data retrieval). The same conservation data model can be employed to answer many different questions (e.g. materials graph, temporal graph, etc.).

This thesis has demonstrated how to embrace encoding as a form of documentation, which can transform data from passive repositories into dynamic analysis engines with the use of graph-based technologies such as the Semantic Web and property graph databases. By representing conservation graphs as directed multigraphs and using graph theoretic algorithms to profile this metamodel, this work has identified a method for conservation knowledge graph construction with foundational discoveries for the development of benchmarking and validation procedures for computational graph models.

Leaf node detection highlighted areas where an earlier version of the CIDOC CRM ontology model required further data connectivity or, conversely, targeted pruning (node removal). Motif frequency counts revealed signature patterns in the Linked Conservation Data graphs that correlated with modelling and data capture practices. Of the profiling measures investigated, the following measures provided the widest applicability across all case studies:

- 9.1. The eigenvector centrality measure can be used to verify the accuracy of directed graph models. It is useful in identifying errors and can aid in refining a model to better align with its intended representational uses. This measure can serve as a

diagnostic measure during the graph building process, therefore, it is recommended for inclusion in the VVC (verification, validation, calibration) stage of data model development.

- 9.2. Triangle Count, as an indicator of connectivity, was found to be indicative of comparable or contrasting data collection practices across institutions and sample datasets. Therefore, triangle counts can be used to identify and characterise datasets by type for processing and for identifying low connectivity where improvements can be made to data gathering practices.
- 9.3. Common across all case study datasets was the recurrent bipartite ($k_{3,3}$) subgraph which suggests non-planarity as a feature of conservation data. This higher dimensionality is an intrinsic characteristic and explains why tabular and traditional relational data models, while able to capture facets of conservation, have been so difficult to use to capture and model across conservation's more complex nature.
- 9.4. Diameter measures were also found to be similar across the datasets regardless of the order (total nodes) or size (total edges) of the particular conservation graph suggesting a traversal threshold given current data gathering and representation practices.

Graph-based data science and data management is presently available for use as a productivity tool for data modeling, exploration and analysis at substantially low material costs (i.e. free software and minimum hardware requirement of a single laptop) without disrupting the institutional collections management system/software (CMS). Furthermore, the experimentation and exploration afforded by graph-based modes of working has the potential to inform future CMS upgrade trajectories. However, challenges to the implementation of graph-based data science and data management for conservation include:

- a need for broad, but not necessarily deep, skills in programming, conceptual modelling, graph theory and graph algorithms,
- a lack of introductory training in such knowledge engineering skills in existing conservation training programs,
- requirement for data preparation, data exploration and iterative development in data analysis and graph modelling,
- limited connectivity within siloed data due to disconnected data practices, which will require identifying relationships and resources for graph enrichment,
- a need for further dissemination of graph-enhanced modes of working.

Although there are currently challenges to increasing computational skills in conservation, the role of conservators will evolve, particularly as more tools necessitate coding. It will be necessary for the field to support the upskilling of a subset of conservators to specialise in knowledge engineering. This must include the practice of conceptual modelling from natural language with cognisance of the cultural assumptions that may be captured in natural language.

In answering the call to action by Otero, this thesis contributes a first step to sharing the language and methods of graph theory and providing several case examples that demonstrate how graph theory plays a role in data science and an underlying role in information technology, particularly in information management technology (i.e. tuples, property graphs, RDF). Further dissemination of graph fundamentals and its affordances will equip conservation professionals with a wider set of tools and understanding to apply to new and existing problems without the platform-specific constraints of existing database solutions.

This research has demonstrated how the challenges raised by Zorich (2016) in their influential report on *Charting the Digital Landscape of the Conservation Profession* can all be addressed via a graph-based approach. It is fitting that graph theory, which traces its beginnings to Euler's solution to the *Seven Bridges* problem, can provide a means for the conservation profession to engage, expand, and explore our own digital landscape.

10.0 Bibliography

- Abuoda, G., Morales, G. D. F., & Aboulnaga, A. (2020). Link Prediction via Higher-Order Motif Features. *ArXiv:1902.06679 [Cs, Stat]*.
<http://arxiv.org/abs/1902.06679>
- Ackoff, R. L. (1989). From Data to Wisdom. *Journal of Applied Systems Analysis*, 16, 3–9.
- AIC/FAIC (American Institute for Conservation/ Foundation for Advancement in Conservation) (1994). *Code of Ethics*.
<https://www.culturalheritage.org/about-conservation/code-of-ethics>
- Aitchison, K. (2013). *Conservation Labour Market Intelligence 2012–13*. Institute of Conservation (Icon).
- Albert, R., Jeong, H., & Barabasi, A.-L. (1999). The diameter of the world wide web. *Nature*, 401(6749), 130–131. <https://doi.org/10.1038/43601>
- Allemang, D., & Hendler, J. (2011). *Semantic Web for the Working Ontologist: Effective Modeling in RDFS and OWL* (2 edition). Morgan Kaufmann.
- American Academy of Pediatrics, Committee on Fetus and Newborn & American College of Obstetricians and Gynecologists and Committee on Obstetric Practice. (2006). The Apgar score. *Pediatrics*, 117(4), 1444–1447.
<https://doi.org/10.1542/peds.2006-0325>
- Anderson, J. R., & Lebiere, C. J. (2014). *The Atomic Components of Thought*. Psychology Press. <https://doi.org/10.4324/9781315805696>
- Angelov, P. P., Soares, E. A., Jiang, R., Arnold, N. I., & Atkinson, P. M. (2021). Explainable artificial intelligence: An analytical review. *WIREs Data Mining and Knowledge Discovery*, 11(5), e1424. <https://doi.org/10.1002/widm.1424>
- Angles, R. (2018). The Property Graph Database Model. In D. Olteanu & B. Pobleto (Eds.), *Proceedings of the 12th Alberto Mendelzon International Workshop on Foundations of Data Management, Cali, Colombia, May 21–25, 2018* (Vol. 2100). <http://ceur-ws.org/Vol-2100/paper26.pdf>
- Angles, R., Thakkar, H., & Tomaszuk, D. (2019). RDF and Property Graphs Interoperability: Status and Issues. *Proceedings of the 13th Alberto Mendelzon International Workshop on Foundations of Data Management, Asunción, Paraguay, June 3-7, 2019*, 2369, 11.
- Aparício, D., Ribeiro, P., & Silva, F. (2015). *Network comparison using directed graphlets* (arXiv:1511.01964). arXiv. <http://arxiv.org/abs/1511.01964>

- Appelbaum, B. (2007). *Conservation Treatment Methodology*. Routledge.
<https://doi.org/10.4324/9780080561042>
- Armbruster, S. (2016, February 24). *Welcome to the Dark Side: Neo4j Worst Practices (& How to Avoid Them)*. Neo4j Graph Data Platform.
<https://neo4j.com/blog/dark-side-neo4j-worst-practices/>
- Arns, J. W. (Ed.). (2016). *Annual Review of Cultural Heritage Informatics: 2015* (2015 ed. edition). Rowman & Littlefield Publishers.
- Ashley-Smith, J. (2000). Developing Professional Uncertainty. *Studies in Conservation*, 45(sup1), 14–17.
<https://doi.org/10.1179/sic.2000.45.Supplement-1.14>
- Ashley-Smith, J. (2016). Losing the edge: The risk of a decline in practical conservation skills. *Journal of the Institute of Conservation*, 39(2), 119–132.
<https://doi.org/10.1080/19455224.2016.1210015>
- Ashley-Smith, J. (2018). The ethics of doing nothing. *Journal of the Institute of Conservation*, 41(1), 6–15. <https://doi.org/10.1080/19455224.2017.1416650>
- Axaridou, A (2020) personal communication, 9 October 2020.
- Bakker, R. R. (1987). *Knowledge Graphs: Representation and structuring of scientific knowledge* [University of Twente].
https://www.researchgate.net/profile/Rene_Bakker4/publication/244467540_Knowledge_Graphs_representation_and_structuring_of_scientific_knowledge/links/582edf5a08ae138f1c0315b1/Knowledge-Graphs-representation-and-structuring-of-scientific-knowledge.pdf
- Bales, M. E., & Johnson, S. B. (2006). Graph theoretic modeling of large-scale semantic networks. *Journal of Biomedical Informatics*, 39(4), 451–464.
<https://doi.org/10.1016/j.jbi.2005.10.007>
- Banks, S. B. (2015). Managing risks from hazardous substances in the Economic Botany Collection at the Royal Botanic Gardens, Kew: A pragmatic approach. *Journal of the Institute of Conservation*, 38(2), 130–145.
<https://doi.org/10.1080/19455224.2015.1068200>
- Bannour, I., Marinica, C., Bouiller, L., Pillay, R., Darrieumerlou, C., Malavergne, O., Kotzinos, D., & Niang, C. (2018). CRMCR - a CIDOC-CRM extension for supporting semantic interoperability in the conservation and restoration domain. *2018 3rd Digital Heritage International Congress (DigitalHERITAGE) Held Jointly with 2018 24th International Conference on Virtual Systems & Multimedia (VSMM 2018)*, 1–8.
- Barok, D., Noordegraaf, J., & Vries, A. P. de. (2019). From Collection Management to Content Management in Art Documentation: The Conservator as an Editor.

- Studies in Conservation*, 0(0), 1–18.
<https://doi.org/10.1080/00393630.2019.1603921>
- Barok, D., Thorez, J. B., Dekker, A., Gauthier, D., & Roeck, C. (2019). Archiving complex digital artworks. *Journal of the Institute of Conservation*, 42(2), 94–113. <https://doi.org/10.1080/19455224.2019.1604398>
- Barrasa, J. (2016, June 7). Importing RDF data into Neo4j. *Jesús Barrasa*.
<https://jbarrasa.com/2016/06/07/importing-rdf-data-into-neo4j/>
- Barrasa, J. (2018). *Connecting connected data: Importing RDF into Neo4j & exposing the LPG in Neo4j as RDF : jbarrasa/neosemantics* [Java].
<https://github.com/jbarrasa/neosemantics> (Original work published 2016)
- Bartha, Paul, (2022) "Analogy and Analogical Reasoning" in Edward N. Zalta (ed.), *The Stanford Encyclopedia of Philosophy*, (Summer 2022 Edition). Accessed via <https://plato.stanford.edu/archives/sum2022/entries/reasoning-analogy/>
- Bean, D. M., Wu, H., Iqbal, E., Dzahini, O., Ibrahim, Z. M., Broadbent, M., Stewart, R., & Dobson, R. J. B. (2017). Knowledge graph prediction of unknown adverse drug reactions and validation in electronic health records. *Scientific Reports*, 7(1), 16416. <https://doi.org/10.1038/s41598-017-16674-x>
- Bedjaoui, M. (2004). The Convention for the Safeguarding of the Intangible Cultural Heritage: The legal framework and universally recognized principles. *Museum International*, 56(1–2), 150–155.
<https://doi.org/10.1111/j.1350-0775.2004.00469.x>
- Bekiari, C., Bruseker, G., Canning, E., Doerr, M., Michon, P., Ore, C.-E., Stead, S., & Velios, A. (Eds.). (2022). *Volume A: Definition of the CIDOC Conceptual Reference Model, Version 7.2.2*. ICOM/CRM Special Interest Group.
- Bell, L., & O'Hare, P. (2020). Latin American politics underground: Networks, rhizomes and resistance in cartonera publishing. *International Journal of Cultural Studies*, 23(1), 20–41. <https://doi.org/10.1177/1367877919880331>
- Bellinger, G., Castro, D., & Mills, A. (2004). *Data, Information, Knowledge, & Wisdom*. Systems Thinking. <https://www.systems-thinking.org/dikw/dikw.htm>
- Benjamin, A., Chartrand, G., & Zhang, P. (2017). *The Fascinating World of Graph Theory*. Princeton University Press.
- Berners-Lee, T. (2007). Giant Global Graph. *Timbl's Blog*.
<http://dig.csail.mit.edu/breadcrumbs/node/215>
- Berners-Lee, T., Hendler, J., & Lassila, O. (2001). The semantic web. *Scientific American Digital*, 284, 34–43.
- Berns, R. S., Taplin, L. A., Imai, F. H., Day, E. A., & Day, D. C. (2005). A Comparison of Small-Aperture and Image-Based Spectrophotometry of Paintings. *Studies in Conservation*, 253–266.

- Birkholz, J.M. & Meroño Peñuela, A.. (2019a). *Decomplexifying networks: A tool for RDF/Wikidata to network analysis—CORE*.
https://core.ac.uk/display/237012067?utm_source=pdf&utm_medium=banner&utm_campaign=pdf-decoration-v1
- Birkholz, J.M. & Meroño Peñuela, A. (2019b). Network Analysis of RDF Graphs - Jupyter Notebook
<https://colab.research.google.com/github/descepolo/rdf-network-analysis/blob/master/rdf-network-analysis.ipynb>
- Blagoderov, V., Penn, M., Sadka, M., Hine, A., Brooks, S., Siebert, D. J., Sleep, C., Cafferty, S., Cane, E., Martin, G., Toloni, F., Wing, P., Chainey, J., Duffell, L., Huxley, R., Ledger, S., McLaughlin, C., Mazzetta, G., Perera, J., ... Kitching, I. J. (2017). iCollections methodology: Workflow, results and lessons learned. *Biodiversity Data Journal*, 5, e19893. <https://doi.org/10.3897/BDJ.5.e19893>
- Blake, E. (2013). Social networks, path dependence, and the rise of ethnic groups in pre-Roman Italy. In C. Knappett (Ed.), *Network analysis in archaeology: New approaches to regional interaction* (1. ed, pp. 203–222). Oxford University Press.
- Bijker, W. E., Hughes, T. P., & Pinch, T. (2012). *The Social Construction of Technological Systems, anniversary edition: New Directions in the Sociology and History of Technology*. MIT Press.
- Bizer, C., & Cyganiak, R. (2014). *RDF 1.1 TriG* (G. Carothers & A. Seaborne, Eds.).
<https://www.w3.org/TR/trig/>
- Bizer, C., T. T. Heath, & Berners-Lee, T. (2009). Linked data—The story so far,. 5 (3) (2009) 1–22. *International Journal on Semantic Web and Information Systems*, 5(3), 1–22.
- Blanke, T., Bryant, M., & Speck, R. (2015). Developing the collection graph. *Library Hi Tech*, 33(4), 610–623. <https://doi.org/10.1108/LHT-07-2015-0070>
- Blumauer, A. (2014). *From Taxonomies over Ontologies to Knowledge Graphs*. July 2014. <https://blog.semantic-web.at/2014/07/15/from-taxonomies-over-ontologies-to-knowledge-graphs>
- Bogdanova, G., Todorov, T., & Noev, N. (2016). Using Graph Databases to Represent Knowledge Base in the Field of Cultural Heritage. *Digital Presentation and Preservation of Cultural and Scientific Heritage, At Bulgaria*, VI.

- Bonacich, P. (2010). Factoring and weighting approaches to status scores and clique identification. *Journal of Mathematical Sociology*.
<https://www.tandfonline.com/doi/abs/10.1080/0022250X.1972.9989806>
- Bonacich, P. (1987). Power and Centrality: A Family of Measures. *American Journal of Sociology*, 92(5), 1170–1182.
- Brachman, R., & Levesque, H. (2004). *Knowledge Representation and Reasoning* (1st ed.). Morgan Kaufmann.
- Brickley, D., & Guha, R. V. (Eds.). (2001). *RDF Schema 1.1*. W3C.
https://www.w3.org/TR/rdf-schema/#ch_literal
- British Museum. (2018). *Oddy Test Results Database 2014-2018*. British Museum.
https://research.britishmuseum.org/research/publications/research_publications_series/2004/selection_of_materials.aspx
- Brown, N. (2019). Making Heritage Science Data Fair and Impactful. *Icon News*, June 2019, 25.
- Brughmans, T., & Peeples, M. A. (Eds.). (2023). *Network Science in Archaeology*. In *Network Science in Archaeology*. Cambridge University Press.
<https://www.cambridge.org/core/books/network-science-in-archaeology/network-science-in-archaeology/F175314A04D4E6539A9FA6DE188C9F5A>
- Brughmans, T. (2013). Thinking Through Networks: A Review of Formal Network Methods in Archaeology. *Journal of Archaeological Method and Theory*, 20(4), 623–662.
- Bruseker, G., Carboni, N., & Guillem, A. (2017). Cultural Heritage Data Management: The Role of Formal Ontology and CIDOC CRM. In M. L. Vincent, V. M. López-Menchero Bendicho, M. Ioannides, & T. E. Levy (Eds.), *Heritage and Archaeology in the Digital Age: Acquisition, Curation, and Dissemination of Spatial Cultural Heritage Data* (pp. 93–131). Springer International Publishing. https://doi.org/10.1007/978-3-319-65370-9_6
- Brushke, J., & Wacker, M. (2014). Application of a graph database and graphical user interface for CIDOC CRM. *Access and Understanding – Networking in the Digital Era*. The Annual Conference of CIDOC, the International Committee for Documentation of ICOM, 6 - 11 September 2014, Dresden, Germany.
- Bryant, M. (2013). *Archival Integration using Neo4j*. Linked Data Benchmark Council, Technical User Community Meeting, Nov 2013, London, UK.
https://github.com/ldbc/tuc_presentations/blob/master/20140303-ehri.pptx

- Buckley, F., & Harary, F. (1990). *Distance in graphs*. Addison-Wesley.
- Cai, H., Zheng, V. W., & Chang, K. C.-C. (2018). A Comprehensive Survey of Graph Embedding: Problems, Techniques, and Applications. *IEEE Transactions on Knowledge and Data Engineering*, 30(9), 1616–1637.
<https://doi.org/10.1109/TKDE.2018.2807452>
- Camarda, D. V., Mazzini, S., & Antonuccio, A. (2012). *LodLive—Browsing the Web of Data*. <http://en.lodlive.it>
- Campagnolo, A. (2015). *Transforming structured descriptions to visual representations. An automated visualization of historical bookbinding structures* [PhD, University of the Arts London].
<http://ualresearchonline.arts.ac.uk/8749/>
- Caple, C. (2012). *Conservation Skills: Judgement, Method and Decision Making*. Routledge.
- Carbonell, J.G, Larkin, J.H., & Reif, F. (1983). *Towards a general scientific reasoning engine* (Joint Computer Science and Psychology Technical Report CMU-CS-93-120). Office of Naval Research (US).
<https://www.semanticscholar.org/paper/Towards-a-general-scientific-reasoning-engine-Carbonell-Reif/c4893ce1cc62b2950da97535b04114979b32ad8a>
- Cardillo, A., Scellato, S., Latora, V., & Porta, S. (2006). Structural properties of planar graphs of urban street patterns. *Physical Review E*, 73(6), 066107.
<https://doi.org/10.1103/PhysRevE.73.066107>
- Carpinone, E. C. (2010). *Museum Collections Management Systems: One Size Does Not Fit All* [Master of Arts, Seton Hall University]
<https://scholarship.shu.edu/dissertations/2366>
- Carrington, P. J., Scott, J., & Wasserman, S. (2005). *Models and Methods in Social Network Analysis*. Cambridge University Press.
- Carroll, J. J., Bizer, C., Hayes, P., & Stickler, P. (2005). Named graphs, provenance and trust. *Proceedings of the 14th International Conference on World Wide Web*, 613–622. <https://doi.org/10.1145/1060745.1060835>
- Castellani, B. and Gerrits, L. (2021). Map of the Complexity Sciences. *Art & Science Factory, LLC*. https://www.art-sciencefactory.com/complexity-map_feb09.html
- Cheung, K.-H., & Shin, D.-G. (2000). A graph-based meta-data framework for interoperation between genome databases. *Proceedings IEEE International Symposium on Bio-Informatics and Biomedical Engineering*, 109–117.
<https://doi.org/10.1109/BIBE.2000.889597>
- Choy, C. A., Haddock, S. H. D., & Robison, B. H. (2017). Deep pelagic food web structure as revealed by in situ feeding observations. *Proceedings of the Royal*

- Society B: Biological Sciences*, 284(1868), 20172116.
<https://doi.org/10.1098/rspb.2017.2116>
- Christie, S., & Gentner, D. (2010). Where Hypotheses Come From: Learning New Relations by Structural Alignment. *Journal of Cognition and Development*, 11(3), 356–373. <https://doi.org/10.1080/15248371003700015>
- Chung, F. R. K., & Lu, L. (2006). *Complex graphs and networks*. American mathematical Society.
- Cilliers, P. (2013). Understanding Complex Systems. In J. P. Sturmborg & C. M. Martin (Eds.), *Handbook of Systems and Complexity in Health* (pp. 27–38). Springer Science & Business Media.
- Cioffi, J. (1997). Heuristics, servants to intuition, in clinical decision-making. *Journal of Advanced Nursing*, 26(1), 203–208.
<https://doi.org/10.1046/j.1365-2648.1997.1997026203.x>
- Cook, C. (2009). Is Clinical Gestalt Good Enough? *The Journal of Manual & Manipulative Therapy*, 17(1), 6–7.
- Claes, J., Vanderfeesten, I., Gailly, F., Grefen, P., & Poels, G. (2015). The Structured Process Modeling Theory (SPMT) a cognitive view on why and how modelers benefit from structuring the process of process modeling. *Information Systems Frontiers*, 17(6), 1401–1425.
<https://doi.org/10.1007/s10796-015-9585-y>
- Classes & Properties Declarations of CIDOC-CRM version: 7.1.1*. (n.d.). Retrieved May 31, 2023, from https://cidoc-crm.org/html/cidoc_crm_v7.1.1.html
- Codd, E. F. (1970). *A relational model for large shared data banks*. Association of Computing Machinery.
- Codd, E. F. (2007). *Relational database: A practical foundation for productivity*. Association of Computing Machinery.
- Collections Trust. (n.d.) *Spectrum: Primary procedures*. Retrieved April 3, 2024, from <https://collectionstrust.org.uk/spectrum/primary-procedures/>
- Colombat, A. P. (1991). A Thousand Trails to Work with Deleuze. *SubStance*, 20(3), 10–23. <https://doi.org/10.2307/3685176>
- Convention on the Means of Prohibiting and Preventing the Illicit Import, Export and Transfer of Ownership of Cultural Property, (1970).
http://portal.unesco.org/en/ev.php-URL_ID=13039&URL_DO=DO_TOPIC&URL_SECTION=201.html
- Convention concerning the Protection of the World Cultural and Natural Heritage, (1972).
http://portal.unesco.org/en/ev.php-URL_ID=13055&URL_DO=DO_TOPIC&URL_SECTION=201.html

- Convention on the Protection of the Underwater Cultural Heritage, (2001).
http://portal.unesco.org/en/ev.php-URL_ID=13520&URL_DO=DO_TOPIC&URL_SECTION=201.html
- Convention on the Safeguarding of Intangible Cultural Heritage, (2003).
<https://ich.unesco.org/en/convention>
- Cotte, P., & Dupouy, M. (2003). CRISATEL high resolution multi-spectral system. *Proceedings PICS Conference, Society of Imaging Science and Technology*, 161–165.
- Coyne, R. (2008). The net effect: Design, the rhizome, and complex philosophy. *Futures*, 40(6), 552–561. <https://doi.org/10.1016/j.futures.2007.11.003>
- Cronyn, J. M. (2003). *Elements of Archaeological Conservation*. Routledge.
- Crouse, M., Nakos, C., Abdelaziz, I., & Forbus, K. (2021). Neural Analogical Matching. *Proceedings of the AAAI Conference on Artificial Intelligence*, 35(1), Article 1.
- Cull, D. H. (2011). Rhizomatic Restoration: Conservation Ethics in the Age of Wikipedia. In B. E. Drushel & K. German, *The Ethics of Emerging Media: Information, Social Norms, and New Media Technology*. A&C Black.
- Cunningham, D. (2016). *Set Theory: A First Course*.
<https://www.cambridge.org/gb/academic/subjects/mathematics/logic-categories-and-sets/set-theory-first-course>
- Cyganiak, R., Wood, D., & Lanthaler, M. (2014). *RDF 1.1 Concepts and Abstract Syntax*. <https://www.w3.org/TR/rdf11-concepts/>
- Dablander, F., & Hinne, M. (2019). Node centrality measures are a poor substitute for causal inference. *Scientific Reports*, 9(1), 6846.
<https://doi.org/10.1038/s41598-019-43033-9>
- Dale, M. R. T. (2017). *Applying Graph Theory in Ecological Research*. Cambridge University Press.
- DARIAH EU, Thesaurus Maintenance Working Group, & VCC3. (2019). *Backbone Thesaurus*. <https://www.backbonethesaurus.eu/>
- Davis, R., Shrobe, H., & Szolovits, P. (1993). What is a Knowledge Representation? *AI Magazine*, 14(1), 17–33.
- Dawar, K., Samuel, A. J., & Alvarado, R. (2019). Comparing Topic Modeling and Named Entity Recognition Techniques for the Semantic Indexing of a Landscape Architecture Textbook. *2019 Systems and Information Engineering Design Symposium (SIEDS)*, 1–6.
<https://doi.org/10.1109/SIEDS.2019.8735642>

- de Marzi, M. (2019, June 22). *FindMotifs.java*.
<https://github.com/maxdemarzi/motifs/blob/master/src/main/java/com/maxdemarzi/results/FindMotifs.java>
- Deleuze, G., & Guattari, F. (2013). *A Thousand Plateaus: Capitalism and Schizophrenia*. Bloomsbury Revelations.
<https://www.amazon.co.uk/Thousand-Plateaus-Bloomsbury-Revelations/dp/1780935374>
- Diestel, R. (2017). *Graph Theory* (5th ed.). Springer-Verlag.
<https://www.springer.com/gb/book/9783662536216>
- Dietrich, C. (2010). Decision Making: Factors that Influence Decision Making, Heuristics Used, and Decision Outcomes. *Inquiries Journal*, 2(02).
<http://www.inquiriesjournal.com/articles/180/decision-making-factors-that-influence-decision-making-heuristics-used-and-decision-outcomes>
- Doerr, M. (2003). The CIDOC Conceptual Reference Module: An Ontological Approach to Semantic Interoperability of Metadata. *AI Mag.*, 24(3), 75–92.
- Doerr, M., Bekiari, C., Bruseker, G., Ore, C.-E., Stead, S., & Velios, T. (Eds.). (2020). *Volume A: Definition of the CIDOC Conceptual Reference Model, Version 7.0*. ICOM/CRM Special Interest Group.
- Doerr, M., Ore, C. E., & Stead, S. (2007, November 5). The CIDOC Conceptual Reference Model—A New Standard for Knowledge Sharing. *Challenges in Conceptual Modelling*. The 26th International Conference on Conceptual Modeling, Auckland, New Zealand.
https://www.researchgate.net/publication/221269820_The_CIDOC_Conceptual_Reference_Model_-_A_New_Standard_for_Knowledge_Sharing
- Duan, Y., Shao, L., Hu, G., Zhou, Z., Zou, Q., & Lin, Z. (2017). Specifying architecture of knowledge graph with data graph, information graph, knowledge graph and wisdom graph. *2017 IEEE 15th International Conference on Software Engineering Research, Management and Applications (SERA)*, 327–332. <https://doi.org/10.1109/SERA.2017.7965747>
- Egon L. Willighagen. (2014). *Accessing biological data in R with semantic web technologies—ProQuest*. <https://doi.org/DOI:10.7287/peerj.preprints.185v3>
- Ehrlinger, L., & Wöß, W. (2016). Towards a Definition of Knowledge Graphs. *SEMANTICS 2016: Posters and Demos Track, September 13-14, 2016*.
- English Heritage/Historic England. (2008). *Conservation Principles, Policies and Guidance | Historic England* (p. 78). Historic England.
<https://historicengland.org.uk/images-books/publications/conservation-principles-sustainable-management-historic-environment/conservationprinciplespoliciesandguidanceapril08web/>

- Espejo, R., Mestre, G., Postigo, F., Lumbreras, S., Ramos, A., Huang, T., & Bompard, E. (2020). Exploiting graphlet decomposition to explain the structure of complex networks: The GHuST framework. *Scientific Reports*, 10(1), Article 1. <https://doi.org/10.1038/s41598-020-69795-1>
- Estrada, E. (2016). When local and global clustering of networks diverge. *Linear Algebra and Its Applications*, 488, 249–263. <https://doi.org/10.1016/j.laa.2015.09.048>
- Europa Nostra. (2014, May 21). *EU Council adopts first-ever conclusions on cultural heritage*. Europa Nostra. <http://www.europanostra.org/eu-council-adopts-first-ever-conclusions-cultural-heritage/>
- Fagnani, F., Fosson, S. M., & Ravazzi, C. (2015). Some Introductory Notes on Random Graphs. In P. R. Kumar, M. J. Wainwright, & R. Zecchina (Eds.), *Mathematical Foundations of Complex Networked Information Systems: Politecnico di Torino, Verrès, Italy 2009*. Springer International Publishing. <https://www.springer.com/gb/book/9783319169668>
- Falkenhainer, B., Forbus, K. D., & Gentner, D. (1986). *The Structure-Mapping Engine* (Report No UIUCDCS-R-86-1275). Department of Computer Science, University of Illinois at Urbana-Champaign. https://www.qrg.northwestern.edu/papers/Files/QRG_Dist_Files/QRG_1986/2_99_Falkenhainer_1986_SME.pdf
- Falkenhainer, B., Forbus, K. D., & Gentner, D. (1989). The structure-mapping engine: Algorithm and examples. *Artificial Intelligence*, 41(1), 1–63. [https://doi.org/10.1016/0004-3702\(89\)90077-5](https://doi.org/10.1016/0004-3702(89)90077-5)
- Farber, M., Ell, B., Menne, C., Rettinger, A., & Bartscherer, F. (2016). Linked Data Quality of DBpedia, Freebase, OpenCyc, Wikidata, and YAGO. *Semantic Web Journal*. <http://www.semantic-web-journal.net/content/linked-data-quality-dbpedia-freebase-opencyc-wikidata-and-yago>
- Fathalla, S., Vahdati, S., Auer, S., & Lange, C. (2017). Towards a Knowledge Graph Representing Research Findings by Semantifying Survey Articles. In J. Kamps, G. Tsakonas, Y. Manolopoulos, L. Iliadis, & I. Karydis (Eds.), *Research and Advanced Technology for Digital Libraries* (pp. 315–327). Springer International Publishing. https://doi.org/10.1007/978-3-319-67008-9_25
- Finlayson, S. G., LePendu, P., & Shah, N. H. (2014). Building the graph of medicine from millions of clinical narratives. *Scientific Data*, 1, 140032. <https://doi.org/10.1038/sdata.2014.32>

- Fiorelli, M., & Stellato, A. (2021). Lifting Tabular Data to RDF: A Survey. In E. Garoufallou & M.-A. Ovalle-Perandones (Eds.), *Metadata and Semantic Research* (pp. 85–96). Springer International Publishing.
https://doi.org/10.1007/978-3-030-71903-6_9
- Fischer, A., & Funke, J. (2016). Entscheiden und Entscheidungen: Die Sicht der Psychologie. In S. Kirste (Ed.), *Interdisziplinarität in den Rechtswissenschaften. Ein interdisziplinärer und internationaler Dialog*. (pp. 217–229). Duncker & Humblot.
- Foulds, L. R. (1985). Enumeration of graph theoretic solutions for facilities layout. *Congressus Numerantium*, 48, 87–99.
- Francis, N., Green, A., Guagliardo, P., Libkin, L., Lindaaker, T., Marsault, V., Plantikow, S., Rydberg, M., Selmer, P., & Taylor, A. (2018). Cypher: An Evolving Query Language for Property Graphs. *Proceedings of the 2018 International Conference on Management of Data*, 1433–1445.
<https://doi.org/10.1145/3183713.3190657>
- Freeman, L. C. (1979). Centrality in Social Networks: Conceptual Clarification. *Social Networks*, 1, 215–239.
- Frink, O., & Smith, P. A. (1930). Irreducible non-planar graphs. *Bulletin of the AMS*, 36, 214.
- Gangemi, A., Guarino, N., Masolo, C., Oltramari, A., & Schneider, L. (2002). Sweetening Ontologies with DOLCE. In A. Gómez-Pérez & V. R. Benjamins (Eds.), *Knowledge Engineering and Knowledge Management: Ontologies and the Semantic Web* (pp. 166–181). Springer.
https://doi.org/10.1007/3-540-45810-7_18
- Gardner, M. (2015). *Reading and Reasoning with Knowledge Graphs*. Carnegie Mellon University.
- Gentner, D. (1983a). Structure-Mapping: A Theoretical Framework for Analogy*. *Cognitive Science*, 7(2), 155–170.
https://doi.org/10.1207/s15516709cog0702_3
- Gentner, D. (1983b). Structure-Mapping: A Theoretical Framework for Analogy. *Cognitive Science*, 7, 155–170.
- Gentner, D., & Asmuth, J. (2019). Metaphoric extension, relational categories, and abstraction. *Language, Cognition and Neuroscience*, 34(10), 1298–1307.
<https://doi.org/10.1080/23273798.2017.1410560>
- Gentner & Bowdle. (2008). Metaphor as Structure-Mapping. In Raymond W. Gibbs, Jr. (Ed.), *The Cambridge Handbook of Metaphor and Thought* (pp. 109–128). Cambridge University Press.

- Gentner, D., Holyoak, K. J., & Kokinov, B. N. (2001). *The Analogical Mind: Perspectives from Cognitive Science*. MIT Press.
- Gentner, D. & Maravilla, F. (2018). Analogical reasoning. L. J. Ball & V. A. Thompson (eds.) *International Handbook of Thinking & Reasoning* (pp. 186-203). NY, NY: Psychology Press
- George Anadiotis. (2018, March 5). *Back to the future: Does graph database success hang on query language?* ZDNET.
<https://www.zdnet.com/article/back-to-the-future-does-graph-database-success-hang-on-query-language/>
- Getty Research Institute. (2017). *Art & Architecture Thesaurus*.
<https://www.getty.edu/research/tools/vocabularies/aat/index.html>
- Giebeler, J., Sartorius, A., Heydenreich, G., & Fischer, A. (2021). A Revised Model for Decision-Making in Contemporary Art Conservation and Presentation. *Journal of the American Institute for Conservation*, 60(2–3), 225–235.
<https://doi.org/10.1080/01971360.2020.1858619>
- Gil, Y., & Groth, P. (2011). LinkedDataLens: linked data as a network of networks. In *Proceedings of the sixth international conference on Knowledge capture* (pp. 191-192). ACM. Groth, P., & Gil, Y. (2011). Linked data for network science. In *Proceedings of the First International Conference on Linked Science-Volume 783* (pp. 1-12). CEUR-WS. Org.
- Gjesfield, E., & Phillips, S. C. (2013). Evaluating adaptive network strategies with geochemical sourcing data: A case study from the Kuril Islands. In C. Knappett (Ed.), *Network analysis in archaeology: New approaches to regional interaction* (1. ed, pp. 281–306). Oxford University Press.
- Gleick, J. (2011). *The information: A history, a theory, a flood* (1st ed). Pantheon Books.
- Gomes, L., & Santanchè, A. (2015). The Web Within: Leveraging Web Standards and Graph Analysis to Enable Application-Level Integration of Institutional Data. In A. Hameurlain, J. Küng, R. Wagner, D. Bianchini, V. De Antonellis, & R. De Virgilio (Eds.), *Transactions on Large-Scale Data- and Knowledge-Centered Systems XIX* (Vol. 8990, pp. 26–54). Springer Berlin Heidelberg. https://doi.org/10.1007/978-3-662-46562-2_2
- Gomes-Jr, L., Jensen, R., & Santanche, A. (2013). Query-based inferences in the Complex Data Management System. *Structured Learning: Inferring Graphs from Structured and Unstructured Inputs (SLG-ICML)*.
- Graham, S., Milligan, I., Weingart, S. B., & Martin, K. (2022). *Exploring Big Historical Data: The Historian's Macroscope* (Second Edition). World Scientific.

- Grandjean, M. (2014). La connaissance est un réseau [Knowledge is a network]. *Les Cahiers Du Numérique*, 10(3), 37–54.
<https://doi.org/DOI:10.3166/LCN.10.3.37-54>.
- Gravett, K. (2019). Troubling transitions and celebrating becomings: From pathway to rhizome. *Studies in Higher Education*, 46(8), 1506–1517.
<https://doi.org/10.1080/03075079.2019.1691162>
- Green, D., & Mustalish, R. (2009). *Digital Technologies and the Management of Conservation Documentation: A survey commissioned by the Andrew W. Mellon Foundation* (pp. 1–92). Andrew W. Mellon Foundation.
<http://mac.mellon.org/mac-files/Mellon%20Conservation%20Survey.pdf>
- Green, A. (2019). GQL Is Now a Global Standards Project alongside SQL, September 16, 2019. *Neo4j Blog*.
<https://neo4j.com/blog/gql-standard-query-language-property-graphs/>
- Gros, C. (2012). Complex and Adaptive Dynamical Systems: A Primer. *ArXiv:0807.4838 [Cond-Mat, Physics:Nlin]*.
<https://doi.org/10.1007/978-3-319-16265-2>
- Guare, J. (1990). *Six Degrees of Separation: A Play*. Vintage.
- Guillem, A., Bruseker, G., & Ronzino, P. (2017). Process, concept or thing? Some initial considerations in the ontological modelling of architecture. *International Journal on Digital Libraries*, 18. <https://doi.org/10.1007/s00799-016-0188-0>
- Guizzardi, G.. (2021). Philosophical Ontology and Domain Modeling—An Introduction to the OntoUML Approach (part 1). Presentation at the IEEE International Requirements Engineering Conference (IEEE RE), September 22, 2021. <https://www.youtube.com/watch?v=ENGEIhbnAx4>, at time 00:01:45-00:01:50
- Gupta, N., Dutta, G., & Fourer, R. (2014). An Expanded Database Structure for a Class of Multi-period, Stochastic Mathematical Programming Models for Process industries. *Decision Support Systems, Decision Support Systems* <https://doi.org/10.1016/j.dss.2014.04.003>
- Haider, J., & Sundin, O. (2019). *Invisible Search and Online Search Engines: The Ubiquity of Search in Everyday Life*. Routledge.
- Hannigan, T. (2015). Close encounters of the conceptual kind: Disambiguating social structure from text. *Big Data & Society*, 2(2), 2053951715608655.
<https://doi.org/10.1177/2053951715608655>
- Hartig, O. (2014). Reconciliation of RDF* and Property Graphs. *ArXiv:1409.3288 [Cs]*. <http://arxiv.org/abs/1409.3288>

- Hayes, J., & Gutierrez, C. (2004). Bipartite graphs as intermediate model for RDF. *Proceedings of the 3rd International Conference on Semantic Web Conference*, 47–61. https://doi.org/10.1007/978-3-540-30475-3_5
- Hayes, P. J., Patel-Schneider, P. F., & World Wide Web Consortium. (2014). *RDF 1.1 Semantics*. <http://www.w3.org/TR/2014/REC-rdf11-mt-20140225/>
- Haynes, D., & Vernau, J. (2019). *The Human Position in an Artificial World: Creativity, Ethics and AI in Knowledge Organization*. Ergon-Verlag.
- Henderson, J. (2018). Managing Uncertainty for Preventive Conservation. *Studies in Conservation*, 63(sup1), 108–112. <https://doi.org/10.1080/00393630.2018.1479936>
- Henderson, J. (2019, June 11). *Who do we exclude when we keep things for the future?* Icon Triennial Conference 2019, New Perspectives: Contemporary Conservation Thinking and Practice, Belfast.
- Hernandez, D., Hogan, A., & Kroetzsch, M. (2015). Reifying RDF: What Works Well With Wikidata? *Proceedings of the 11th International Workshop on Scalable Semantic Web Knowledge Base Systems Co-Located with 14th International Semantic Web Conference (ISWC 2015), Bethlehem, PA, USA, October 11, 2015*, 1457, 16.
- Hillier, J. (2021). The “Flatness” of Deleuze and Guattari: Planning the City as a Tree or as a Rhizome? *DisP - The Planning Review*, 57(2), 16–29. <https://doi.org/10.1080/02513625.2021.1981008>
- Historic England. (2017). *BIM for Heritage: Developing a Historic Building Information Model*. Swindon. Historic England.
- Historic England. (2015). *Managing Significance in Decision-Taking in the Historic Environment* (Historic Environment Good Practice Advice in Planning: 2, p. 20). Historic England. <https://historicengland.org.uk/images-books/publications/gpa2-managing-significance-in-decision-taking/gpa2/>
- Hoang, T. (2018, November 20). *Graph planarity and path addition method of Hopcroft-Tarjan for planarity testing*. Medium. <https://towardsdatascience.com/graph-planarity-and-path-addition-method-of-hopcroft-tarjan-for-planarity-testing-c56d2df2b0b3>
- Hoede, C. (1994). Modelling knowledge in electronic study books. *Journal of computer assisted learning*, 10(2), 104–112. <https://doi.org/10.1111/j.1365-2729.1994.tb00287.x>
- Hogan, A., Blomqvist, E., Cochez, M., d’Amato, C., Melo, G. de, Gutierrez, C., Kirrane, S., Gayo, J. E. L., Navigli, R., Neumaier, S., Ngomo, A.-C. N., Polleres, A., Rashid, S. M., Rula, A., Schmelzeisen, L., Sequeda, J., Staab, S., &

- Zimmermann, A. (2021). Knowledge Graphs. *Synthesis Lectures on Data, Semantics, and Knowledge*, 12(2), 1–257.
<https://doi.org/10.2200/S01125ED1V01Y202109DSK022>
- Hollis, G., & Westbury, C. (2016). The principals of meaning: Extracting semantic dimensions from co-occurrence models of semantics. *Psychonomic Bulletin & Review*, 23(6), 1744–1756. <https://doi.org/10.3758/s13423-016-1053-2>
- Holtorf, C. (2001). Is the Past a Non-Renewable Resource. In P. G. Stone, R. Layton, & J. Thomas (Eds.), *Destruction and conservation of cultural property: Vol. One world archaeology* (pp. 286–297). Routledge.
- Homan, J. V., & Kovacs, P. J. (2009). A Comparison of the Relational Database Model and the Associative Database Model. *Issues In Information Systems*.
https://doi.org/10.48009/1_iis_2009_208-213
- Honan, E. (2007). Writing a rhizome: An (im)plausible methodology. *International Journal of Qualitative Studies in Education*, 20(5), 531–546.
<https://doi.org/10.1080/09518390600923735>
- Hug, C., & Gonzalez-Perez, C. (2012). Qualitative evaluation of cultural heritage information modeling techniques. *ACM Journal on Computing and Cultural Heritage*, 5(2, Article 8). <https://doi.org/10.1145/2307723.2307727>
- Hunger, M., & Lyon, W. (2016). *Analyzing the Panama Papers with Neo4j: Data Models, Queries & More*, April 8, 2016.
<https://neo4j.com/blog/analyzing-panama-papers-neo4j/>
- ICOM/CIDOC CRM Special Interest Group. (2018). *CIDOC Conceptual Reference Model* (M. Doerr & C. E. Ore, Eds.).
<http://www.cidoc-crm.org/Version/version-6.2.3>
- ICON. (2020). *The Institute of Conservation's Professional Standards*.
<https://www.icon.org.uk/resources/resources-for-conservation-professionals/standards-and-ethics/icon-professional-standards.html>
- ICON. (2016). *Icon Professional Accreditation of Conservator-Restorers (PACR): Accreditation Handbook*. Institute of Conservation (Icon).
https://icon.org.uk/system/files/pacr_handbook_2016.pdf
- International Council on Archives. (2000). *ISAD(G): General International Standard Archival Description, adopted by the Committee on Descriptive Standards, Stockholm, Sweden, 19-22 September 1999* (Second Edition).
https://www.ica.org/sites/default/files/CBPS_2000_Guidelines_ISAD%28G%29_Second-edition_EN.pdf
- International Organization for Standardization. (n.d.). *ISO 21127:2014 Information and documentation—A reference ontology for the interchange of cultural*

- heritage information*. ISO. Retrieved April 24, 2023, from <https://www.iso.org/standard/57832.html>
- Jungnickel, D. (2013). *Graphs, networks, and algorithms* (Fourth edition). Springer.
- Kaiser, M. (2008). Mean clustering coefficients: The role of isolated nodes and leafs on clustering measures for small-world networks. *New Journal of Physics*, *10*(8), 083042. <https://doi.org/10.1088/1367-2630/10/8/083042>
- Katz, L. (1953). A new status index derived from sociometric analysis. *Psychometrika*, *18*(1), 39–43. <https://doi.org/10.1007/BF02289026>
- Kay, P., & Kempton, W. (1984). What Is the Sapir-Whorf Hypothesis? *American Anthropologist*, *86*(1), 65–79. <https://doi.org/10.1525/aa.1984.86.1.02a00050>
- Kesper, A., Wenz, V., & Taentzer, G. (2020). Detecting quality problems in research data: A model-driven approach. *Proceedings of the 23rd ACM/IEEE International Conference on Model Driven Engineering Languages and Systems*, 354–364. <https://doi.org/10.1145/3365438.3410987>
- Knappett, C. (Ed.). (2013a). *Network analysis in archaeology: New approaches to regional interaction* (1. ed). Oxford University Press.
- Knappett, C. (2013b). Introduction: Why Networks? In C. Knappett (Ed.), *Network analysis in archaeology: New approaches to regional interaction* (1. ed). Oxford University Press.
- Kneebone, R. (2019a, June 26). *Embodied Knowing in Medicine and Science*. Encounters on the Shop Floor, V&A, London. <https://www.vam.ac.uk/research/projects/vari-encounters-on-the-shop-floor>
- Kneebone, R. (2019b, November 7). *Unnamed Territories of the Body*. Picturing the Invisible, Chelsea College of Art, UAL, London. <https://www.arts.ac.uk/research/groups-networks-and-collaborations/picturing-the-invisible>
- Kobourov, S. G., Pupyrev, S., & Saket, B. (2014). Are Crossings Important for Drawing Large Graphs? In C. Duncan & A. Symvonis (Eds.), *Graph Drawing* (pp. 234–245). Springer. https://doi.org/10.1007/978-3-662-45803-7_20
- Koller, D., & Friedman, N. (2009). *Probabilistic Graphical Models: Principles and Techniques*. MIT Press.
- Kong, X., Cao, B., & Yu, P. S. (2013). Multi-label classification by mining label and instance correlations from heterogeneous information networks. *Proceedings of the 19th ACM SIGKDD International Conference on Knowledge Discovery and Data Mining*, 614–622. <https://doi.org/10.1145/2487575.2487577>
- Koob, S. P. (1982). The instability of cellulose nitrate adhesives. *The Conservator*, *6*, 31–34.

- Korte, C., & Milgram, S. (1970). Acquaintance linking between white and negro populations: Application of the small world problem. *Journal of Personality and Social Psychology*, *15*, 101–118.
- Koukaras, P., Rousidis, D., & Tjortjis, C. (2021). An Introduction to Information Network Modeling Capabilities, Utilizing Graphs. In E. Garoufallou & M.-A. Ovalle-Perandones (Eds.), *Metadata and Semantic Research* (pp. 134–140). Springer International Publishing.
https://doi.org/10.1007/978-3-030-71903-6_14
- Kroetsch, M., & Weikum, G. (2016). *Journal of Web Semantics: Special Issue on Knowledge Graphs. August 2016*.
<http://www.websemanticsjournal.org/index.php/ps/announcement/view/19>
- Kuczera, A. (2016). Digital Editions beyond XML – Graph-based Digital Editions. In M. Düring, A. Jatowt, J. Preiser-Kapeller, & A. van den Bosch (Eds.), *Proceedings of the 3rd HistInformatics Workshop, 11 July 2016* (p. 10).
<http://ceur-ws.org>.
- Kuczera, A. (2017). Graphentechnologien in den Digitalen Geisteswissenschaften. *ABI Technik*, *37*(3). <https://doi.org/10.1515/abitech-2017-0042>
- Kumar, P. R., Wainwright, M. J., & Zecchina, R. (2015). *Mathematical foundations of complex networked information systems: Politecnico di Torino, Verrès, Italy 2009* (F. Fagnani, S. M. Fosson, & C. Ravazzi, Eds.). Springer International Publishing. <https://doi.org/10.1007/978-3-319-16967-5>
- Kuratowski, K. (1930). Sur le problème des courbes gauches en topologie. *Fund. Math. (in French)*, *15*, 271–283.
- Lahanier, C., Alquié, G., Cotte, P., Christofides, C., de Deyne, C., Pillay, R., Saunders, D., & Schmitt, F. (2002). CRISATEL: a high definition and spectral digitization of paintings with simulation of varnish removal. In R. Vontobel (Ed.), *ICOM Committee for Conservation 13th Triennial Meeting, Rio de Janeiro, 22–27 September 2002: Preprints* (pp. 295–300). James & James, London.
- Lakoff, G., & Johnson, M. (2003). *Metaphors We Live By* (New edition). University of Chicago Press.
- Laranjeiro, N., Soydemir, S. N., & Bernardino, J. (2015). A Survey on Data Quality: Classifying Poor Data. *2015 IEEE 21st Pacific Rim International Symposium on Dependable Computing (PRDC)*, 179–188.
<https://doi.org/10.1109/PRDC.2015.41>
- Lawson, L., Finbow, A., & Marçal, H. (2019). Developing a strategy for the conservation of performance-based artworks at Tate. *Journal of the Institute*

of Conservation, 42(2), 114–134.

<https://doi.org/10.1080/19455224.2019.1604396>

- Le Boeuf, P., Doerr, M., Ore, C. E., & Stead, S. (2016). *Definition of the CIDOC Conceptual Reference Model. Technical Report 6.2*.
- Lei Zhang. (2002). *Knowledge Graph Theory and Structural Parsing*. University of Twente.
- Letellier, R., Schmid, W., LeBlanc, F., Getty Conservation Institute, International Council on Monuments and Sites, & International Committee of Architectural Photogrammetry. (2007). *Recording, Documentation, and Information Management for the Conservation of Heritage Places: Guiding Principles*.
- Libkin, L., Martens, W., & Vrgoč, D. (2016). Querying Graphs with Data. *Journal of the ACM*, 63(2), 14:1-14:53. <https://doi.org/10.1145/2850413>
- Lieu, R., & Campagnolo, A. (2022). Modelling Linked Data for Conservation: A Call for New Standards. *KULA: Knowledge Creation, Dissemination, and Preservation Studies*, 6(3), Article 3. <https://doi.org/10.18357/kula.232>
- Light, R. B., Roberts, D. A., & Stewart, J. D. (2014). *Museum Documentation Systems: Developments and Applications*. Butterworth-Heinemann.
- Lindsay, H. (2018). Evidencing the Case for Preventive Conservation: The Role of Collections Care Documentation. *Studies in Conservation*, 63(sup1), 175–180. <https://doi.org/10.1080/00393630.2018.1504516>
- Linked Conservation Data. (n.d.). [Project website]. *Linked Conservation Data*. Retrieved June 5, 2023, from <https://www.ligatus.org.uk/lcd/>
- Liu, B. (2015). *Sentiment Analysis*. Cambridge University Press.
- Madhero88. (2010). *Apgar score*. Own work. https://commons.wikimedia.org/wiki/File:Apgar_score.png
- Malhotra, M., & Nair, T. G. (2015). Evolution of Knowledge Representation and Retrieval Techniques. *International Journal of Intelligent Systems and Applications*, 7(7), 18–28. <https://doi.org/10.5815/ijisa.2015.07.03>
- Mantegari, G., Palmonari, M., & Vizzari, G. (2010). Rapid Prototyping a Semantic Web Application for Cultural Heritage: The Case of MANTIC. In L. Aroyo, G. Antoniou, E. Hyvönen, A. ten Teije, H. Stuckenschmidt, L. Cabral, & T. Tudorache (Eds.), *The Semantic Web: Research and Applications* (pp. 406–410). Springer Berlin Heidelberg.
- Marchese, F. T. (2011). Conserving Digital Art for Deep Time. *Leonardo*, 44(4), 302–308. https://doi.org/10.1162/LEON_a_00206
- Marciano, R., Agarrat, S., Frisch, H., Hunt, M. R., Jain, K., Kocienda, G., Krauss, H., Liu, C., McKinley, M., Mir, D., Mullane, C., Patterson, E., Pradhan, D., Santos, J., Schams, B., Shiue, H. S. Y., Silva, A. J., Suri, M., Turabi, T., ... Xu, J. (2019).

- Reframing Digital Curation Practices through a Computational Thinking Framework. *2019 IEEE International Conference on Big Data (Big Data)*, 3126–3135. <https://doi.org/10.1109/BigData47090.2019.9006485>
- Marinica, C. (2019, September 12). *PARCOURS project: CRMcr—A CIDOC-CRM extension for supporting semantic interoperability in the conservation and restoration domain*. Linked Conservation Data Network: Modelling Workshop.
- Martin Doerr. (2009). Ontologies for Cultural Heritage. In S. Staab & R. Studer (Eds.), *Handbook on Ontologies*. Springer-Verlag Berlin Heidelberg.
- Martins, L. (2021). Plant artefacts then and now: Reconnecting biocultural collections in Amazonia. In F. Driver, M. Nesbitt, & C. Cornish (Eds.), *Mobile Museums* (pp. 21–43). UCL Press. <https://doi.org/10.2307/j.ctv18kc0px.8>
- Martins, L., Nesbitt, M., Milliken, W., Fonseca-Kruel, V., Cabalzar, A., Azevedo, D. L., Scholz, A., Sunnucks, L. O., Sekulowicz, L., Cabalzar, F., Katerinchuck, V., & Moyes, B. (n.d.). *Digital Repatriation of Biocultural Collections: Rio Negro, Amazonia* [Research project website]. Retrieved February 20, 2023, from <http://en.biocultural.wpengine.com/>
- Marty, P. F. (1999). Museum informatics and collaborative technologies: The emerging socio-technological dimension of information science in museum environments. *Journal of the American Society for Information Science*, *50*(12), 1083-1091.
- Matsumoto, S., Yamanaka, R., & Chiba, H. (2018). Mapping RDF Graphs to Property Graphs. *ArXiv:1812.01801 [Cs]*. <http://arxiv.org/abs/1812.01801>
- Max de Marzi. (2019, July 11). Finding Motifs in Cypher for Fun and Profit. *Max De Marzi*. <https://maxdemarzi.com/2019/07/11/finding-motifs-in-cypher-for-fun-and-profit/>
- McKinney, W. (2018). *Python for Data Analysis: Data Wrangling with Pandas, NumPy, and IPython*. O'Reilly Media, Inc.
- McLeod, R., & Schell, G. (2001). *Management information systems* (8th Edition). Prentice Hall.
- Merleau-Ponty, M. (1962). *The phenomenology of perception* (C. Smith, Trans.). Routledge & Kegan Paul.
- Mezzanzanica, M., Mercorio, F., Cesarini, M., Moscato, V., & Picariello, A. (2018). GraphDBLP: A system for analysing networks of computer scientists through graph databases. *Multimedia Tools and Applications*, *77*(14), 18657–18688. <https://doi.org/10.1007/s11042-017-5503-2>

- Mills, B. J. (2017). Social Network Analysis in Archaeology. *Annual Review of Anthropology*, 46(1), 379–397.
<https://doi.org/10.1146/annurev-anthro-102116-041423>
- Milo, R., Shen-Orr, S., Itzkovitz, S., Kashtan, N., Chklovskii, D., & Alon, U. (2002). Network Motifs: Simple Building Blocks of Complex Networks. *Science*, 298(5594), 824–827. <https://doi.org/10.1126/science.298.5594.824>
- Ministry of House, Communities and Local Government. (2019). *National Planning Policy Framework*.
<https://www.gov.uk/government/publications/national-planning-policy-framework--2>
- Minsky, M. (1975). A framework for representing knowledge. In *The Psychology of Computer Vision* (pp. 211–277). McGraw Hill.
<https://web.media.mit.edu/~minsky/papers/Frames/frames.html>
- Mnih, V., Kavukcuoglu, K., Silver, D., Graves, A., Antonoglou, I., Wierstra, D., & Riedmiller, M. (2013). *Playing Atari with Deep Reinforcement Learning* (arXiv:1312.5602). arXiv. <https://doi.org/10.48550/arXiv.1312.5602>
- Mnih, V., Kavukcuoglu, K., Silver, D., Rusu, A. A., Veness, J., Bellemare, M. G., Graves, A., Riedmiller, M., Fidjeland, A. K., Ostrovski, G., Petersen, S., Beattie, C., Sadik, A., Antonoglou, I., King, H., Kumaran, D., Wierstra, D., Legg, S., & Hassabis, D. (2015). Human-level control through deep reinforcement learning. *Nature*, 518 (7540), 529–533. <https://doi.org/10.1038/nature14236>
- Moore, M. (2001). Conservation Documentation and the Implications of Digitisation. *Journal of Conservation and Museum Studies*, 7, 6–10.
<https://doi.org/10.5334/jcms.7012>
- Moore, P. G., & Thomas, H. (1976). *The anatomy of decisions*. Penguin Books.
- Moraitou, E., Aliprantis, J., Christodoulou, Y., Teneketzis, A., & Caridakis, G. (2019). Semantic Bridging of Cultural Heritage Disciplines and Tasks. *Heritage*, 2(1), 611–630. <https://doi.org/10.3390/heritage2010040>
- Moraitou, E. & Christodoulou, Y. (2021). *Overview of current conservation and restoration models*. Linked Conservation Data (LCD) Modelling Working Group.
<https://www.ligatus.org.uk/lcd/sites/ligatus.org.uk.lcd/files/attachments/248/%CE%BFverview-of-current-conservation-and-restoration-models-20210312.pdf>

- Muñoz Viñas, S. (2022). Conservation science, conservation practice and the conservator's knowledge: A naïve exploration. *Journal of the Institute of Conservation*, 45(3), 173–189.
<https://doi.org/10.1080/19455224.2022.2112407>
- Munoz-Vinas, S. (2012). *Contemporary Theory of Conservation*. Routledge. Museum and Galleries of New South Wales (M&G NSW). (n.d.). *How to: Collection Management Systems*.
https://mgnsww.org.au/wp-content/uploads/2019/01/how-to_collection-management-systems.pdf
- National Library of Medicine. (n.d.). *Data Provenance*. Data Glossary. Retrieved June 29, 2024, from
<https://www.nlm.gov/guides/data-glossary/data-provenance>
- Natsiavas, P., Boyce, R. D., Jaulent, M.-C., & Koutkias, V. (2018). OpenPVSIGNAL: Advancing Information Search, Sharing and Reuse on Pharmacovigilance Signals via FAIR Principles and Semantic Web Technologies. *Frontiers in Pharmacology*, 9. <https://doi.org/10.3389/fphar.2018.00609>
- Needham, M., & Hodler, A. E. (2019). *Graph algorithms: Practical examples in Apache Spark and Neo4j*.
- Neill, I., & Kuczera, A. (2019). *The Codex – an Atlas of Relations*.
http://dx.doi.org/10.17175/sb004_008
- Neo4j. (2019). *New Query Language for Graph Databases to Become International Standard, September 17, 2019*. Neo4j Press Releases. Retrieved May 28, 2024 via
<https://neo4j.com/press-releases/query-language-graph-databases-international-standard/>
- Neo4j. (2024). *Neo4j Welcomes New GQL International Standard in Major Milestone for Database Industry, April 17, 2024*. Neo4j Press Releases. Retrieved May 28, 2024 via <https://neo4j.com/press-releases/gql-standard/>
- Neo4j. (n.d.). *Eigenvector Centrality*. Neo4j Graph Data Science Library Manual v2.3. Retrieved June 7, 2023, from
<https://neo4j.com/docs/graph-data-science/2.3/algorithms/eigenvector-centrality/>
- Neumayr, B., & Schuetz, C. G. (2017). Multilevel Modeling. In L. Liu & M. T. Ozsu (Eds.), *Encyclopedia of Database Systems*. Springer Science & Business Media. DOI 10.1007/978-1-4899-7993-3_80807-1
- Newman, M. E. J. (2000). Models of the Small World: A Review. *ArXiv:Cond-Mat/0001118*. <http://arxiv.org/abs/cond-mat/0001118>

- Newman, M. E. J. (2001). The structure of scientific collaboration networks. *Proceedings of the National Academy of Sciences of the United States of America*, 98(2), 404–409.
- Newman, M. E. J. (2003). The Structure and Function of Complex Networks. *SIAM Review*, 45(2), 167–256.
- Noble, L., Vavassori, V., Crookham, A., & Dunn, S. (2022a). Networking the Archive: The Stories and Structures of Thos. Agnew's Stock Books. *Journal on Computing and Cultural Heritage*, 15(1), 12:1-12:14.
<https://doi.org/10.1145/3479009>
- Noble, L., Vavassori, V., Crookham, A., & Dunn, S. (2022b). Networking the Archive: The Stories and Structures of Thos. Agnew's Stock Books. *Journal on Computing and Cultural Heritage*, 15(1), 12:1-12:14.
<https://doi.org/10.1145/3479009>
- Nurdiati, S. N. S., & Hoede, C. (2008). *25 years development of knowledge graph theory: The results and the challenge*.
<https://research.utwente.nl/en/publications/25-years-development-of-knowledge-graph-theory-the-results-and-the>
- Otero, J. (2022). Heritage Conservation Future: Where We Stand, Challenges Ahead, and a Paradigm Shift. *Global Challenges*, 6(1), 2100084.
<https://doi.org/10.1002/gch2.202100084>
- Page, L., Brin, S., Motwani, R., & Winograd, T. (1999, November 11). *The PageRank Citation Ranking: Bringing Order to the Web*. [Techreport]. Stanford InfoLab.
<http://ilpubs.stanford.edu:8090/422/>
- Paoletti, T. (2006). *Leonard Euler's Solution to the Konigsberg Bridge Problem*. *Convergence*, 3. *Mathematical Association of America*.
<https://www.maa.org/press/periodicals/convergence/leonard-eulers-solution-to-the-konigsberg-bridge-problem>
- Paul Scifleet, Susan Williams, & Creagh Cole. (2009). The Human Art of Encoding: Markup as a Documentary Practice. In M.-A. Sicilia & M. D. Lytras (Eds.), *Metadata and Semantics*. Springer US.
<https://www.springer.com/gp/book/9780387777443>
- Paulheim, H. (2016). Knowledge graph refinement: A survey of approaches and evaluation methods. *Semantic Web*, 8(3), 489–508.
<https://doi.org/10.3233/SW-160218>
- Pavlopoulos, G. A., Secier, M., Moschopoulos, C. N., Soldatos, T. G., Kossida, S., Aerts, J., Schneider, R., & Bagos, P. G. (2011). Using graph theory to analyze biological networks. *BioData Mining*, 4(1), 10.
<https://doi.org/10.1186/1756-0381-4-10>

- Perez, J., Arenas, M., & Gutierrez, C. (2006). Semantics and Complexity of SPARQL. *ArXiv:Cs/0605124*. <http://arxiv.org/abs/cs/0605124>
- Pinker, S. (1990). A Theory of Graph Comprehension. In R. Freedle (Ed.), *Artificial intelligence and the future of testing* (pp. 73-126.). Lawrence Erlbaum Associates.
- Plangprasopchok, A., Lerman, K., & Getoor, L. (2010). Growing a tree in the forest: Constructing folksonomies by integrating structured metadata. *Proceedings of the 16th ACM SIGKDD International Conference on Knowledge Discovery and Data Mining*, 949–958. <https://doi.org/10.1145/1835804.1835924>
- Pocket Oxford English Dictionary (2023). "Complex, adj." and "Complex, n.", Oxford University Press.
- Poggi, A., Lembo, D., Calvanese, D., De Giacomo, G., Lenzerini, M., & Rosati, R. (2008). Linking Data to Ontologies. In S. Spaccapietra (Ed.), *Journal on Data Semantics X* (pp. 133–173). Springer. https://doi.org/10.1007/978-3-540-77688-8_5
- Policy for the Care of Culturally Restricted Objects* (v1.1). (n.d.). Great North Museum: Hancock. <https://greatnorthmuseum.org.uk/files/407637-great-north-museum-hancock-policy-for-the-care-of-culturally-restricted-objects.pdf>
- Polyvyanyy, A., Ouyang, C., Barros, A., & van der Aalst, W. M. P. (2017). Process querying: Enabling business intelligence through query-based process analytics. *Decision Support Systems*, 100, 41–56. <https://doi.org/10.1016/j.dss.2017.04.011>
- Pothuau, L., Porion, P., Lespessailles, E., Benhamou, C. L., & Levitz, P. (2000). A new method for three-dimensional skeleton graph analysis of porous media: Application to trabecular bone microarchitecture. *Journal of Microscopy*, 199(2), 149–161. <https://doi.org/10.1046/j.1365-2818.2000.00725.x>
- Powell, J., & Hopkins, M. (2015). *A Librarian's Guide to Graphs, Data and the Semantic Web*. Elsevier. <https://doi.org/10.1016/C2013-0-16976-0>
- Price, L. (2019). Fandom, Folksonomies and Creativity: The case of the Archive of Our Own. In D. Haynes & J. Vernau (Eds.), *The Human Position in an Artificial World: Creativity, Ethics and AI in Knowledge Organization* (ISKO UK Sixth Biennial Conference, 15-16th July 2019, pp. 11–37). Ergon-Verlag.
- Pringle, E., Mavin, H., Greenhalgh, T., Dalal-Clayton, A., Rutherford, A., Bramwell, J., Blackford, K., & Balukiewicz, K. (2022). *Provisional Semantics: Addressing the challenges of representing multiple perspectives within an evolving digitised national collection*. Zenodo. <https://doi.org/10.5281/zenodo.7081347>

- Przulj, N. (2006). Graph Theory Analysis of Protein-Protein Interactions. In I. Jurisica & D. Wigle (Eds.), *Knowledge Discovery in Proteomics* (pp. 73–128). CRC Press, Taylor & Francis Group, LLC.
<https://www.routledge.com/Knowledge-Discovery-in-Proteomics/Jurisica-Wigle/p/book/9780367392178>
- Przulj, N., Corneil, D. G., & Jurisica, I. (2004). *Modeling Interactome: Scale-Free or Geometric?* (arXiv:q-bio/0404017). arXiv. <http://arxiv.org/abs/q-bio/0404017>
- Pujara, J., Miao, H., Getoor, L., & Cohen, W. (2013). Knowledge Graph Identification. In , pages. *Proceedings of the 12th International Semantic Web Conference - Part I, ISWC '13*, 542–557.
- Purcell, M. (2013). A new land: Deleuze and Guattari and planning. *Planning Theory & Practice*, 14(1), 20–38. <https://doi.org/10.1080/14649357.2012.761279>
- Rakha, H., Hellinga, B., Aerde, M. V., & Perez, W. (1996). *Systematic Verification, Validation and Calibration of Traffic Simulation Models*. 75th Annual Meeting of the Transportation Research Board, Washington, D.C.
- Rassinoux, A.-M., Baud, R. H., Lovis, C., Wagner, J. C., & Scherrer, J.-R. (1998). Tuning up conceptual graph representation for multilingual natural language processing in medicine. In M.-L. Mugnier & M. Chein (Eds.), *Conceptual Structures: Theory, Tools and Applications* (pp. 390–397). Springer Berlin Heidelberg.
- Reedy, T. J., & Reedy, C. L. (1988). *Statistical Analysis in Art Conservation Research*. <http://www.getty.edu/publications/virtuallibrary/0892360976.html>
- Ribés, A., Brettel, H., Schmitt, F., Liang, H., Cupitt, J., & Saunders, D. (2003). Color and multispectral imaging with the CRISATEL multispectral system. *Proceedings PICS Confer- Ence, Society of Imaging Science and Technology, Springfield, VA*, 215–219.
- Ristoski, P., & Paulheim, H. (2016). Semantic Web in data mining and knowledge discovery: A comprehensive survey. *Journal of Web Semantics*, 36, 1–22. <https://doi.org/10.1016/j.websem.2016.01.001>
- Roberts, M. (2019). *The Application of Machine Learning to At-Risk Cultural Heritage Image Data* [Master]. Durham University.
- Robinson, I., Webber, J., & Eifrem, E. (2015). *Graph Databases*. O'Reilly Media. <http://shop.oreilly.com/product/0636920041832.do>
- Rock, I., & Palmer, S. (1990). The Legacy of Gestalt Psychology. *Scientific American*, 263(6), 84–90. <https://doi.org/10.1038/scientificamerican1290-84>
- Rodriguez, M. A. (2009). A Graph Analysis of the Linked Data Cloud. *ArXiv:0903.0194 [Cs]*. <http://arxiv.org/abs/0903.0194>

- Rogerson, C., & Garside, P. (2017). Increasing the profile and influence of conservation—An unexpected benefit of risk assessments. *Journal of the Institute of Conservation*, *40*(1), 34–48.
<https://doi.org/10.1080/19455224.2016.1214848>
- Roy, A., Foister, S., & Rudenstine, A. (2007). Conservation Documentation in Digital Form: A Continuing Dialogue about the Issues. *Studies in Conservation*, *52*(4), 315–317. JSTOR.
- Roy, S. G., & Chakrabarti, A. (2017). Chapter 11—A novel graph clustering algorithm based on discrete-time quantum random walk. In S. Bhattacharyya, U. Maulik, & P. Dutta (Eds.), *Quantum Inspired Computational Intelligence* (pp. 361–389). Morgan Kaufmann.
<https://doi.org/10.1016/B978-0-12-804409-4.00011-5>
- Royal, C., & Kosterich, A. (2024). Coding Competencies across Roles: Computer Programming Practices in News Media Organizations. *Journalism Studies*, *25*(7), 738–758. <https://doi.org/10.1080/1461670X.2024.2340570>
- Russell, R., & Winkworth, K. (2009). *Significance 2.0: A guide to assessing the significance of collections*. Collections Council of Australia Ltd.
<https://www.arts.gov.au/sites/g/files/net1761/f/significance-2.0.pdf>
- Ryan, Y. C., & Ahnert, S. E. (2021). The Measure of the Archive: The Robustness of Network Analysis in Early Modern Correspondence. *Journal of Cultural Analytics*, *6*(3), 25943. <https://doi.org/10.22148/001c.25943>
- Saha, P., & Sarkar, D. (2022). Characterization and Classification of ADHD Subtypes: An Approach Based on the Nodal Distribution of Eigenvector Centrality and Classification Tree Model. *Child Psychiatry & Human Development*.
<https://doi.org/10.1007/s10578-022-01432-6>
- Salvatore, C. L. (2018). The Tools and Technology in Cultural Heritage Management. In C. L. Salvatore (Ed.), *Cultural Heritage Care and Management: Theory and Practice* (pp. 109–121). Rowman & Littlefield.
- Sanderson, R. (2020, December 3). *Introduction to Linked Art*. Linked Pasts 6, University of London and British Library (and online).
<https://ics.sas.ac.uk/ics-digital/linked-pasts-6>
- Sankaraman, S., & Mahadevan, S. (2015). Integration of model verification, validation, and calibration for uncertainty quantification in engineering systems. *Reliability Engineering & System Safety*, *138*, 194–209.
<https://doi.org/10.1016/j.ress.2015.01.023>
- Sarajlić, A., Malod-Dognin, N., Yaveroğlu, Ö. N., & Pržulj, N. (2016). Graphlet-based Characterization of Directed Networks. *Scientific Reports*, *6*(1), Article 1. <https://doi.org/10.1038/srep35098>

- Schneider, T., & Šimkus, M. (2020). Ontologies and Data Management: A Brief Survey. *Kunstliche Intelligenz*, 34(3), 329–353.
<https://doi.org/10.1007/s13218-020-00686-3>
- Schmachtenberg, M., Bizer, C. & Paulheim, H. (n.d.). Adoption of the Linked Data Best Practices in Different Topical Domains. *International Semantic Web Conference, 2014*.
- Scott, B., Baker, E., Woodburn, M., Vincent, S., Hardy, H., & Smith, V. S. (2019). The Natural History Museum Data Portal. *Database*, 2019.
<https://doi.org/10.1093/database/baz038>
- Sekulowicz, L. (2022). *Recording the Intangible: Approaches to the ethnographic archive* [presentation and workshop]. Techne Student-led Event at Royal Botanic Gardens, Kew, 15 June 2022.
- Selwitz, C. (1988). *Cellulose nitrate in conservation*. Getty Conservation Institute.
- Sequoiah-Grayson, S., & Floridi, L. (2022). Semantic Conceptions of Information. In E. N. Zalta (Ed.), *The Stanford Encyclopedia of Philosophy* (Spring 2022). Metaphysics Research Lab, Stanford University.
<https://plato.stanford.edu/archives/spr2022/entries/information-semantic/>
- Shaban-Nejad, A. (2012). Categorical Representation. In N. M. Seel (Ed.), *Encyclopedia of the Sciences of Learning* (pp. 515–517). Springer US.
https://doi.org/10.1007/978-1-4419-1428-6_529
- Shashoua, Y., Bradley, S. M., & Daniels, V. D. (1992). Degradation of Cellulose Nitrate Adhesive. *Studies in Conservation*, 37(2), 113–119.
<https://doi.org/10.2307/1506403>
- Shaw, R. (2015). Bringing Deleuze and Guattari down to Earth through Gregory Bateson: Plateaus, Rhizomes and Ecosophical Subjectivity. *Theory, Culture & Society*, 32(7–8), 151–171. <https://doi.org/10.1177/0263276414524451>
- Silge, J., & Robinson, D. (2017). *Text Mining with R*. O'Reilly Media.
<https://www.tidytextmining.com/>
- Sindbaek, E. (2013). Broken links and black boxes: Material affiliations and contextual network synthesis in the Viking world. In C. Knappett (Ed.), *Network analysis in archaeology: New approaches to regional interaction* (1. ed, pp. 71–94). Oxford University Press.
- Singhal, A. (2012, May). Introducing the Knowledge Graph: Things, not strings. *Official Google Blog*.
<https://googleblog.blogspot.com/2012/05/introducing-knowledge-graph-things-not.html>

- Sledge, J. (1999). Spectrum – a Review. *Archives and Museum Informatics*, 13, 55–61. <https://doi.org/10.1023/A:1009094008720>
- Sloggett, R. (2009). Expanding the Conservation Canon: Assessing Cross-Cultural and Interdisciplinary Collaborations in Conservation. *Studies in Conservation*, 54(3), 170–183. <https://doi.org/10.1179/sic.2009.54.3.170>
- Smith, B. J., & Prikryl, R. (2007). Diagnosing decay: The value of medical analogy in understanding the weathering of building stones. *Geological Society, London, Special Publications*, 271(1), 1–8. <https://doi.org/10.1144/GSL.SP.2007.271.01.01>
- Smolensky, P. (1987). Connectionist AI, symbolic AI, and the brain. *Artificial Intelligence Review*, 1(2), 95–109. <https://doi.org/10.1007/BF00130011>
- Sophocleous, S., Savic, D., Kapelan, Z., Shen, Y., & Sage, P. (2016). A Graph-based Analytical Technique for the Improvement of Water Network Model Calibration. *Procedia Engineering*, 154, 27–35. <https://doi.org/10.1016/j.proeng.2016.07.415>
- Sowa, J.F. (2000). *Knowledge representation: Logical, philosophical, and computational foundations*. Brooks/Cole.
- Sporleder, C. (2010). Natural Language Processing for Cultural Heritage Domains. *Language and Linguistics Compass*, 4(9), 750–768. <https://doi.org/10.1111/j.1749-818X.2010.00230.x>
- Srinivasa-Desikan, B. (2018). *Natural Language Processing and Computational Linguistics: A practical guide to text analysis with Python, Gensim, spaCy, and Keras*. Packt Publishing Ltd.
- Stevens, G., & Atamturktur, S. (2017). Mitigating Error and Uncertainty in Partitioned Analysis: A Review of Verification, Calibration and Validation Methods for Coupled Simulations. *Archives of Computational Methods in Engineering*, 24(3), 557–571. <https://doi.org/10.1007/s11831-016-9177-0>
- Steyvers, M., & Tenenbaum, J. B. (2005). The large-scale structure of semantic networks: Statistical analyses and a model of semantic growth. *Cognitive Science*, 29(1), 41–78. https://doi.org/10.1207/s15516709cog2901_3
- Stokman, F. N., & de Vries, P. H. (1988). Structuring Knowledge in a Graph. In G. C. van der Veer & G. Mulder (Eds.), *Human-Computer Interaction: Psychonomic Aspects* (pp. 186–206). Springer Berlin Heidelberg.
- Stone, L., Simberloff, D., & Artzy-Randrup, Y. (2019). Network motifs and their origins. *PLOS Computational Biology*, 15(4), e1006749. <https://doi.org/10.1371/journal.pcbi.1006749>
- Storrar, T., & Talboom, L. (2019, July 30). *The National Archives—Network analysis of the UK Government Web Archive* [Text]. The National Archives Blog; The

- National Archives.
<https://blog.nationalarchives.gov.uk/network-analysis-of-the-uk-government-web-archive/>
- Sturmberg, J. P., & Martin, C. (Eds.). (2013). *Handbook of Systems and Complexity in Health*. Springer Science & Business Media.
<https://doi.org/10.1007/978-1-4614-4998-0>
- Suenson-Taylor, K., Sully, D., & Orton, C. (1999). Data in conservation: The missing link in the process. *Studies in Conservation*, 44(3), 184–194.
<https://doi.org/10.1179/sic.1999.44.3.184>
- Summers-Stay, D. (2017). Deductive and Analogical Reasoning on a Semantically Embedded Knowledge Graph. *ArXiv:1707.03232 [Cs]*.
<http://arxiv.org/abs/1707.03232>
- Sweller, J., Van Merriënboer, J. J. G., & Paas, F. G. W. C. (1998). Cognitive architecture and instructional design. *Educational Psychology Review*, 10(3), 251–296.
- Tal, E. (2017). Calibration: Modelling the measurement process. *Studies in History and Philosophy of Science Part A*, 65–66, 33–45.
<https://doi.org/10.1016/j.shpsa.2017.09.001>
- Tantardini, M., Ieva, F., Tajoli, L., & Piccardi, C. (2019). Comparing methods for comparing networks. *Scientific Reports*, 9(1), Article 1.
<https://doi.org/10.1038/s41598-019-53708-y>
- Taylor, J. (2018a). In the quest for certainty: Tensions from cause-and-effect deductions in preventive conservation. *Journal of the Institute of Conservation*, 41(1), 16–31. <https://doi.org/10.1080/19455224.2017.1416649>
- Taylor, J. (2018b). In the quest for certainty: Tensions from cause-and-effect deductions in preventive conservation. *Journal of the Institute of Conservation*, 41(1), 16–31. <https://doi.org/10.1080/19455224.2017.1416649>
- The National Archives. (2019). *The National Archives—Network analysis of the UK Government Web Archive* [Text]. The National Archives Blog; The National Archives.
<https://blog.nationalarchives.gov.uk/network-analysis-of-the-uk-government-web-archive/>
- The National Archives. (n.d.). *The National Archives - Discovery for developers: About the application programming interface (API)*; The National Archives. Retrieved October 5, 2022, from
<https://www.nationalarchives.gov.uk/help/discovery-for-developers-about-the-application-programming-interface-api/>

- Thacker, B. H., Doebbling, S. W., Hemez, F. M., Anderson, M. C., Pepin, J. E., & Rodriguez, E. A. (2004). *Concepts of Model Verification and Validation* (p. 41).
- Thickett, D., Lee, L. R., & Lee, L. R. (2004). *Selection of materials for the storage or display of museum objects* (New and completely rev. ed). British Museum.
- Tienminh91. (2013). *Tiếng Việt: K3,3 và K5*. Own work.
https://commons.wikimedia.org/wiki/File:K3,3_v%C3%A0_K5.png
- Torrey, L., & Shavlik, J. (2010). *Transfer Learning* [Chapter]. Handbook of Research on Machine Learning Applications and Trends: Algorithms, Methods, and Techniques; IGI Global. <https://doi.org/10.4018/978-1-60566-766-9.ch011>
- Trudeau, R. J. (1993). *Introduction to Graph Theory* (1st Revised edition). Dover Publications Inc.
- UNESCO, ICCROM, ICOMOS, & IUCN. (2013). *Managing Cultural World Heritage* (World Heritage Resource Manual).
- Underdown, D. (2018). *The National Archives—Using the Discovery API to analyse catalogue data*, 27 July 2018. The National Archives Blog; The National Archives. <https://blog.nationalarchives.gov.uk/using-the-discovery-api/>
- Uyar, A., & Aliyu, F. M. (2015). Evaluating search features of Google Knowledge Graph and Bing Satori: Entity types, list searches and query interfaces. *Online Information Review*, 39(2), 197–213.
- Valiant, L. G. (1981). Universality considerations in VLSI circuits. *IEEE Transactions on Computers*, C-30(2), 135–140. <https://doi.org/10.1109/TC.1981.6312176>
- VanSnick, S., & Ntanos, K. (2018). On Digitisation as a Preservation Measure. *Studies in Conservation*, 63(sup1), 282–287.
<https://doi.org/10.1080/00393630.2018.1504451>
- Velios, A. (2016a). Beyond databases: Linked open data for bookbinding descriptions. In G. Boudalis, P. Engel, R. Ion, M. Ciechanska, I. Kecskeméti, E. Moussakova, F. Pinzari, J. Schirò, & J. Vodopivec (Eds.), *Historical Book Binding Techniques in Conservation* (pp. 173–194). Verlag Berger.
<https://www.verlag-berger.at/alle-produkte/fachliteratur/detail/v/isbn-978-3-85028-785-2.html>
- Velios, A. (2016b). Online event-based conservation documentation: A case study from the IIC website. *Studies in Conservation*, 61, 13–25.
- Velios, A. (2022). *CRMVIZ* [Python]. <https://github.com/natuk/crmviz> (Original work published 2020)
- Velios, A., & St John, K. (2022a). Linked Conservation Data: Driving Change in Documentation Practice. *Studies in Conservation*, 67(sup1), 293–300.
<https://doi.org/10.1080/00393630.2022.2065957>

- Velios, A., & St John, K. (2022b). Linked Conservation Data: Driving Change in Documentation Practice. *Studies in Conservation*, 0(0), 1–8.
<https://doi.org/10.1080/00393630.2022.2065957>
- Voegeli, D. (2018). *A Field Guide-ETL from RDF to Property Graph*. The MITRE Corporation.
https://www.mitre.org/sites/default/files/publications/pr-15-2949-ETL-from%20-RDF-to-property-graph_0.pdf
- Wasserman, S and Faust, K. (1994). *Social Network Analysis: Methods and Applications*. Cambridge: Cambridge University Press
- Watts, D. J., & Strogatz, S. H. (1998). Collective dynamics of ‘small-world’ networks. *Nature*, 393(6684), Article 6684. <https://doi.org/10.1038/30918>
- Weintrop, D., Beheshti, E., Horn, M., Orton, K., Jona, K., Trouille, L., & Wilensky, U. (2016). Defining Computational Thinking for Mathematics and Science Classrooms. *Journal of Science Education and Technology*, 25(1), 127–147.
<https://doi.org/10.1007/s10956-015-9581-5>
- Welcome to the NCBO BioPortal | NCBO BioPortal. (n.d.). Retrieved January 20, 2023, from <https://bioportal.bioontology.org/>
- Whetzel, P., Noy, N., Shah, N., Alexander, P., Nyulas, C., Tudorache, T., & Musen, M. (2011). BioPortal: Enhanced functionality via new Web services from the National Center for Biomedical Ontology to access and use ontologies in software applications. 2011 Jul; *Nucleic Acids Research*, 39(Web Server issue):W541-5).
- Wilcke, X., Bloem, P., & de Boer, V. (2017). The knowledge graph as the default data model for learning on heterogeneous knowledge. *Data Science*, 1(1–2), 39–57. <https://doi.org/10.3233/DS-170007>
- Wilkinson, M. D., Dumontier, M., Aalbersberg, Ij. J., Appleton, G., Axton, M., Baak, A., Blomberg, N., Boiten, J.-W., da Silva Santos, L. B., Bourne, P. E., Bouwman, J., Brookes, A. J., Clark, T., Crosas, M., Dillo, I., Dumon, O., Edmunds, S., Evelo, C. T., Finkers, R., ... Mons, B. (2016). The FAIR Guiding Principles for scientific data management and stewardship. *Scientific Data*, 3(1), 1–9. <https://doi.org/10.1038/sdata.2016.18>
- Willman, M. D. (2023). Logic and Language in Early Chinese Philosophy. In E. N. Zalta & U. Nodelman (Eds.), *The Stanford Encyclopedia of Philosophy* (Fall 2023). Metaphysics Research Lab, Stanford University. Accessed 30 June 2024.
<https://plato.stanford.edu/archives/fall2023/entries/chinese-logic-language/>

- Wills, P., & Meyer, F. G. (2020). Metrics for graph comparison: A practitioner's guide. *PLOS ONE*, *15*(2), e0228728.
<https://doi.org/10.1371/journal.pone.0228728>
- Wilson, R. J. (1996). *Introduction to Graph Theory* (4th edition). Addison Wesley Longman Ltd.
- Wolff, P., & Gentner, D. (2011). Structure-mapping in metaphor comprehension. *Cognitive Science*, *35*(8), 1456–1488.
<https://doi.org/10.1111/j.1551-6709.2011.01194.x>
- Wyatt, S. (2008). Technological Determinism Is Dead; Long Live Technological Determinism. In E. J. Hackett, O. Amsterdamska, M. Lynch, & J. Wajcman (Eds.), *The Handbook of Science and Technology Studies* (3rd ed., pp. 165–180). MIT Press.
- Yaveroğlu, Ö. N., Malod-Dognin, N., Davis, D., Levnajic, Z., Janjic, V., Karapandza, R., Stojmirovic, A., & Pržulj, N. (2014). Revealing the Hidden Language of Complex Networks. *Scientific Reports*, *4*(1), Article 1.
<https://doi.org/10.1038/srep04547>
- Zeng, K., Li, C., Hou, L., Li, J., & Feng, L. (2021). A comprehensive survey of entity alignment for knowledge graphs. *AI Open*, *2*, 1–13.
<https://doi.org/10.1016/j.aiopen.2021.02.002>
- Zeng, M. L. (2008). Knowledge Organization Systems (KOS). *Knowledge Organization*, *35*(No.2/No.3), 160–182.
- Zhu, G., & Iglesias, C. A. (2017). Computing Semantic Similarity of Concepts in Knowledge Graphs. *IEEE Transactions on Knowledge and Data Engineering*, *29*(1), 72–85. <https://doi.org/10.1109/TKDE.2016.2610428>
- Zins, C. (2007). Conceptual approaches for defining data, information, and knowledge. *Journal of the American Society for Information Science and Technology*, *58*(4), 479–493. <https://doi.org/10.1002/asi.20508>
- Zloch, M., Acosta, M., Hienert, D., Conrad, S., & Dietze, S. (2021). Characterizing RDF Graphs through Graph-based Measures—Framework and Assessment | www.semantic-web-journal.net. *Semantic Web – Interoperability, Usability, Applicability*, *12*(5).
<http://semantic-web-journal.net/content/characterizing-rdf-graphs-through-graph-based-measures-framework-and-assessment-0#>
- Zorich, D., & Fuentes, A. (2014). *What's Out There? A Baseline Review of Online Resources for the Conservation Community*. The Foundation of the American Institute for Conservation of Historic and Artistic Works (FAIC).
<https://www.culturalheritage.org/docs/default-source/reports/faic-baseline-report-final.pdf?sfvrsn=2>

Zorich, D. M. (2016). *Charting the Digital Landscape of the Conservation Profession: A Report to the Profession* (p. 35). The Foundation for the American Institute for the Conservation of Historic and Artistic Works.

Applying Graph Theory to Conservation Documentation

Ana Tam

A Thesis submitted in partial fulfillment of the
requirements for the degree of Doctor of Philosophy
at the University of the Arts London

July 2024

Volume 2 - Appendices

Table of Contents - Volume 2

<i>Table of Contents</i>	<i>i</i>
Appendix A: Neo4j Configuration, Node Colour Key & Exploration Queries	1
A1. Introduction	1
A2. Specifications	1
A3. Configurations	2
A4. System Memory Configurations	2
A5. Key to Node Colours Applied in this Study	4
A6. General Cypher Exploration and Profiling Queries for Inspecting an Unknown Graph DBMS	4
Appendix B: The TNA CCD Dataset: Preparation and ETL	5
B1. Data Preparation for Import into Neo4j	5
B2. Creating (:TreatmentEvent) Nodes and their Properties	7
B3. Creating Adjacent Star Schema Nodes and their Property Keys	8
B4. Creating Relationships for Treatment Star Schema	12
B5. Other Data Cleaning Guidelines	12
Appendix C: The NLP-Derived Dataset: Preparation, ETL and Threshold Uses	13
C1. Introduction	13
C2. Example of Natural Language Processing (NLP) using spaCy	13
C3. Condition Comments Sample	15
C4. Comments Sample	16
C5. The Handling of Incorrect or Imperfect Attributions	17
C6. Threshold Uses for this NLP-Derived Dataset	17
C7. The Transformation and Import Scripts	19
C8. Cypher Queries	21

Appendix D: The Discovery API: Preparation and ETL	22
D1. Introduction	22
D2. Python Script to Retrieve Catalogue Data from Discovery API	22
D3. List of Discovery API Fields	23
D4. Level 1 - Department	24
D5. Level 3 - Series	25
D6. Levels 6 and 7 - Pieces and Items	27
D7. Python Script for Iterating Through Batched Ranges in the API GET Request	27
D8. List of References with Outlier Reference String Patterns	28
D9. Cypher Scripts to Create Relationships between :Discovery nodes	29
Appendix E: The CAMEO Dataset: Preparation and ETL	31
E1. Introduction	31
E2. Extracting CAMEO data using a URLs list	31
E3. Load and Transform (using Cypher to map into a Neo4j dbms)	33
E4. Sample Cypher Queries	33
Appendix F: Phase 2: The CIDOC CRM Group: Preparation and ETL	34
F1. Introduction	34
F2. ETL of CIDOC CRM RDFS Serialisations	34
F3. Create a crmID to ease Cypher query	36
Appendix G: The LCD Group: Preparation and ETL	38
G1. Introduction	38
G2. Neo4j Specifications	38
G3. ETL: Importing the LCD Group into Labelled Property Graph	39
G4. Parsing Composite Labels for Identifier Strings	39
G5. Query-Based Analysis	40
Appendix H: The Phase 1 Models	44
H1. Introduction	44
H2. Model A - TNA CCD only	45
H3. Model B - Applying NLP	51
H4. Model C: TNA CCD + CRM v6.2.1 (ETL2) + NLP star schema + CAMEO	53

Appendix I: Phase 3 - Transformation of LPG to RDF	59
11. Introduction	59
12. Cypher	59
13. RDF Validation	60
Appendix J: Phase 2 Results	61

Appendix A

Neo4j Configuration, Node Colour Key & Exploration Queries

[A1. Introduction](#)

[A2. Specifications](#)

[A3. Configurations](#)

[A3.1 Neosemantics Configuration](#)

[A4. System Memory Configurations](#)

[A5. Key to Node Colours Applied in this Study](#)

[A6. General Cypher Exploration and Profiling Queries for Inspecting an Unknown Graph DBMS](#)

A1. Introduction

Some script samples will be included in the appendix text with explanations of parameter mappings. A Zenodo doi will be created for this thesis and related digital objects, such as sample or full scripts and other code will be accessible through this permanent identifier, including placement of digital objects on GitHub (which is supported by Zenodo). This appendix will include all ETL scripts (or samples thereof) for Phases I, II and III. The main repository will be via Zenodo to GitHub: <https://github.com/ana-tam/conservation-graphs>.

A2. Specifications

Software versions were regularly updated via releases by Neo4j dating ca. 2018-2023.

- Neo4j Desktop Enterprise
- Neo4j Browser
- APOC
- Neosemantics
- Graph Data Science Library (GDSDL)

The following configurations (i.e. settings) were standard recommended configurations at the time. For the latest installation and configuration details, please refer to the respective Neo4j documentation.

A3. Configurations

The following lines are added to the .conf file or via "Settings" on the Neo4j Desktop:

```
#General configuration
# `LOAD CSV` section of the manual for details.
dbms.directories.import=import

#Installation of built-in plugins will automatically update the following line:
dbms.security.procedures.unrestricted=jwt.security.*,apoc.*,gds.*,n10s.*

#Apoc config as of [add date]
apoc.import.file.enabled=true
apoc.export.file.enabled=true

#Neosemantics config [as of add date]
dbms.unmanaged_extension_classes=n10s.endpoint=/rdf
```

A3.1 Neosemantics Configuration

Details can be found at: <https://neo4j.com/labs/neosemantics/4.3/config/>

Ensure the following constraint is declared:

```
CREATE CONSTRAINT n10s_unique_uri ON (r:Resource)
ASSERT r.uri IS UNIQUE;
```

A4. System Memory Configurations

Running algorithms can require a lot of memory. It is recommended to increase the memory allocation from its default settings. Otherwise, the system will return a 'not enough memory' message like the following:

There is not enough memory to perform the current task. Please try increasing 'dbms.memory.heap.max_size' in the neo4j configuration (normally in 'conf/neo4j.conf' or, if you are using Neo4j Desktop, found through the user interface) or if you are running an embedded installation increase the heap by using '-Xmx' command line flag, and then restart the database.

The Neo4j Operations Manual¹ recommends:

The heap memory size is determined by the parameters
dbms.memory.heap.initial_size and dbms.memory.heap.max_size. It is

¹ <https://neo4j.com/docs/operations-manual/current/performance/memory-configuration/>

recommended to set these two parameters to the same value to avoid unwanted full garbage collection pauses.

For example, the default settings are set to:

```
# Java Heap Size: by default the Java heap size is dynamically calculated based
# on available system resources. Uncomment these lines to set specific initial
# and maximum heap size.
dbms.memory.heap.initial_size=512m
dbms.memory.heap.max_size=1G

# The amount of memory to use for mapping the store files.
# The default page cache memory assumes the machine is dedicated to running
# Neo4j, and is heuristically set to 50% of RAM minus the Java heap size.
dbms.memory.pagecache.size=512m
```

These settings were changed to double the default max settings and match the initial_size and max_size as per the Operations Manual recommendations:

```
# Java Heap Size: by default the Java heap size is dynamically calculated based
# on available system resources. Uncomment these lines to set specific initial
# and maximum heap size.
dbms.memory.heap.initial_size=2G
dbms.memory.heap.max_size=2G

# The amount of memory to use for mapping the store files.
# The default page cache memory assumes the machine is dedicated to running
# Neo4j, and is heuristically set to 50% of RAM minus the Java heap size.
dbms.memory.pagecache.size=1G
```

For example, using a MacBook Pro with 2.7 GHz processor and 16 GB RAM, running the Motif 4.4 query by deMarzi on the default settings ran out of memory. However, with the increased memory allocation, the query ran properly and quickly, completing in 930ms.

A5. Key to Node Colours Applied in this Study

<u>LCD RDF Graphs</u>	<u>Phase 3 LPG and RDF Graphs</u>
(:E11_Modification)	(:TreatmentEvent)
(:E22_Man-Made_Object)	(:Reference)
(:E79_Part_Addition)	(:Adhesive)
(:E57_Material)	(:RepairMaterial)
(:E3_Condition_State)	(:Solvent)
(:E55_Type)	(:Person)
(:E52_Time-Span)	(:Cameo)
(:E53_Place)	(:Vocab), (:Tech)
(:E14_Condition_Assessment)	(:Discovery)
(:E29_Design_or_Procedure)	(:DamageType)
(:E12_Production)	(:Comments), (:ConditionComments)
All other node labels in gray.	(:NounChunk), (:Verb)

A6. General Cypher Exploration and Profiling Queries for Inspecting an Unknown Graph DBMS

//Find All Node Labels and Return a Node Count for Each

```
MATCH (n)
RETURN
DISTINCT labels(n),
count(*);
```

//Find All Relationship Types and Return a Count of Each

```
MATCH ()-[relationship]->()
RETURN TYPE(relationship) AS type, COUNT(relationship) AS amount
ORDER BY amount DESC;
```

//Show All the Property Keys for a Specific Node Label

```
MATCH (a:TreatmentEvent) RETURN keys(a)
```

//To find statistical profile of the graph

//including by Node label and Relationship types.

//This procedure requires installing APOC

```
CALL apoc.meta.stats
```

Appendix B

The TNA CCD Dataset: Preparation and ETL

[B1. Data Preparation for Import into Neo4j](#)

[B2. Creating \(:TreatmentEvent\) Nodes and their Properties](#)

- [:TreatmentEvent :TNA :Data](#)

[B3. Creating Adjacent Star Schema Nodes and their Property Keys](#)

- [:Reference](#)
- [:Adhesives](#)
- [:RepairMaterial](#)
- [:Solvents](#)
- [:Person](#)
- [:PrimaryDamage](#)
- [:SecondaryDamage](#)

[B4. Creating Relationships for Treatment Star Schema](#)

[B5. Other Data Cleaning Guidelines](#)

B1. Data Preparation for Import into Neo4j

The original tabular dataset (in .csv) provided by TNA CCD contained multi-word headings with spaces. To prepare the data for loading into Neo4j, a “cleaned” version of the dataset was created (to avoid *loading errors*) where spaces in the headings were deleted (as shown in Table B.1.1 below). Furthermore, the heading “User” was changed to “Person” and “Time” was changed to “WorkTime” as both “User” and “Time” are reserved keywords in the Neo4j Cypher query language (< v.2.0). Date entries for “Date requested” and “Date treated” were reformatted from a dd/mm/yyyy format to a yyyyymmdd format prior to loading into Neo4j as this is the preferred syntax of the platform to enable date, time, and duration calculations. While it is possible to undertake date and time conversions within Neo4j, in this case, it was considered simpler to achieve this change via spreadsheet functions prior to loading rather than executing Cypher queries to achieve the same result after loading. Finally, an additional column and heading, “RowID”, was created as a unique identifier for each specific conservation treatment instance as captured in a data row. This also allowed for human-supervised referencing back to the tabular data wherever necessary to confirm modelling and any transformations remained true to the original. For non-experimental implementations, a more persistent identifier such as using a universal unique identifier (UUID) is recommended.

Table B.1.1 Modified headings to the .csv prior to loading into Neo4j

Original TNA CCD .csv Headings List	Modified Headings List
Reference	Reference
Primary Damage	Primarydamage
Secondary Damage	Secondarydamage
Condition Comments	Conditioncomments
Repair Material	RepairMaterial
Adhesives	Adhesives
Solvents	Solvents
Date requested	Daterequested
Date treated	Datetreated
User	Person
Time	WorkTime
Comments	Comments

Table B.1.2 Overview of TNA CCD case study data as (:TreatmentEvent) Nodes

Prepared TNA dataset headings (property keys)	Description of values (column content)
Reference	The unique identifier string assigned by TNA for items, collections, and departments. In this dataset's context, the Reference refers to the collection item(s) assessed or treated. The string can consist of alphanumeric characters, spaces, slashes (/) and hyphens (-).
Primarydamage	A value assigned by the conservator as the principle cause of or contributor to damage. The values are either: "Mechanical", "Chemical", "Deposits", "Biological", or "Other".
Secondarydamage	A value assigned by the conservator as the secondary cause of or contributor to damage. The values are either: "Mechanical", "Chemical", "Deposits", "Biological", or "Other".
Conditioncomments	Written in natural language. This is text entered by the conservator. Distinction between "condition comments" and "comments" is blurred and dependent upon the conservator's interpretation. There exists legacy markup tags (e.g. " ") as the result of a prior data transfer from a previous database system.
RepairMaterial	Details of repair material(s) used in the treatment, recorded as a single value or as an unordered list separated by a comma.
Adhesives	Details of adhesive(s) used in the treatment, recorded as a single value or as an unordered list separated by a comma.
Solvents	Details of solvent(s) used in the treatment, recorded as a single value or as an unordered list separated by a comma.
Daterequested	The date the item was requested by a researcher or member of the public.
Datetreated	The date of treatment by staff in the conservation department.

Person	The member of staff responsible.
Time	Duration of time, in hours, required to undertake the conservation activity (e.g. treatment).
Comments	Written in natural language. This is text entered by the conservator. Distinction between "condition comments" and "comments" is blurred and dependent upon the conservator's interpretation. There exists legacy markup tags (e.g. " ") as the result of a prior data transfer from a previous database system.
RowID	Unique identifier specific to this project, corresponding to the original order of the dataset, as received from TNA.

B2. Creating (:TreatmentEvent) Nodes and their Properties

This section uses the following data files:

TNA.csv

- `:TreatmentEvent :TNA :Data`
 - `nodeProperties`
 - `.reference`
 - `.rowID`
 - `.primaryDamage`
 - `.secondaryDamage`
 - `.repairMaterial`
 - `.adhesives`
 - `.solvents`
 - `.comments`
 - `.person`
 - `.conditionComments`
 - `.time` (i.e duration of work in hourly increments)
 - `.dateTreated`
 - `.dateRequested`
 - //if working with more than one dataset, or expect to aggregate multiple sets overtime, recommend adding a `.dataset` property so a value can be specified.
 - Description
 - Each `:TreatmentEvent` includes all data in one row of the TNA dataset. The node properties include all columns in the original dataset (.csv). Not all nodes will have each property, e.g. not all have `.secondaryDamage`. If it was a blank cell in the original dataset, the property will not be mapped.
 - Cypher for transformation

```
LOAD CSV WITH HEADERS FROM 'file:///TNA.csv' AS row
CREATE (:TreatmentEvent
{reference:row.Reference,
primaryDamage:row.Primarydamage,
```

```

secondaryDamage:row.Secondarydamage,
conditionComments:row.Conditioncomments,
repairMaterial:row.RepairMaterial,
adhesives:row.Adhesives,
solvents:row.Solvents,
dateRequested:row.Daterequested,
dateTreated:row.Datetreated,
person:row.Person,
workTime:row.Time,
comments:row.Comments,
rowID:row.RowID})

```

```

//If adding additional labels, for example :Data, :TNA
//Each node then ends up having 3 labels
Match (b) Set b:Data:TNA
Return b, labels(b) AS labels

```

```

//Setting workTime to float
MATCH (a:TreatmentEvent)
SET a.workTime = toFloat(a.workTime);
//output = set 3718 properties

```

```

//Setting date properties
Match (a:TreatmentEvent)
SET a.dateTreated = date(a.dateTreated)
//output = Set 5761 properties

```

```

Match (a:TreatmentEvent)
SET a.dateRequested = date(a.dateRequested)
//output = set 112 properties

```

B3. Creating Adjacent Star Schema Nodes and their Property Keys

- :Reference
 - nodeProperties
 - .reference
 - Description
 - The rationale for just having the .reference property is that the Reference nodes are representative of museum catalogue reference numbers and the object or group of objects this singular number represents. No .rowID is specified as a referenced object or group can have multiple treatment events (which the rowIDs refer to from the original dataset).
 - Cypher for transformation

```

LOAD CSV WITH HEADERS FROM 'file:///TNA.csv' AS row
CREATE (n:Reference {reference:row.Reference})
Return n

```

- Cypher for finding duplicate Reference nodes and Merging them

//*****Option 1 to check known reference string for duplicates, i.e. from multiple treatments of same object

```
Match (n:Reference{reference:"INF 1/292"}) WITH n.reference AS reference,
      COLLECT(n) AS nodelist, COUNT(*) AS count
WHERE count > 1
CALL apoc.refactor.mergeNodes(nodelist) YIELD node
RETURN node
```

//*****Option 2 for Finding multiple Reference nodes

Source:

<https://community.neo4j.com/t/delete-duplicate-node-checking-if-specific-keys-have-same-values/11652/4>

//Use this to check first:

```
MATCH (c:Reference)
WITH c.reference AS name, COLLECT(c) AS refs,
      SIZE(COLLECT(c)) AS nbr_nodes
WHERE SIZE(refs) > 1 // only want names that have more than one node
RETURN name, refs, SIZE(refs)
```

//Use this to perform the actual merging:

```
MATCH (c:Reference)
WITH c.reference AS name, COLLECT(c) AS refs,
      SIZE(COLLECT(c)) AS nbr_nodes
WHERE SIZE(refs) > 1 // only want names that have more than one node
// uncomment the RETURN and delete lines below it to see the grouping
//RETURN name, refs, SIZE(refs)
WITH name as name, refs as ref, nbr_nodes AS nbr_nodes
UNWIND RANGE(1, nbr_nodes - 1) as idx
      CALL apoc.refactor.mergeNodes([ref[0], ref[idx]], {properties:
      {name:'combine'}}) YIELD node //if merging things with more properties,
      specify which properties get combined or overwritten
RETURN name, max(idx) + 1 AS `Nbr Nodes Merged`
```

- :Adhesives
 - nodeProperties:
 - .reference
 - .adhesive
 - .rowID
 - Description
 - Each :Adhesives node is one named solvent from the Adhesives column in that specific TreatmentEvent row. Where there was a list

of adhesives named in the cell, these have been split and made into their own individual nodes during the transformation process.

- Cypher for transformation

```
LOAD CSV WITH HEADERS FROM 'file:///TNA.csv' AS row
WITH row, split(row.Adhesives, ",") AS adhesives
UNWIND adhesives AS adhesive
MERGE (n:Adhesives {reference: row.Reference,
adhesive: adhesive,
rowID:row.RowID})
Return n
```

- :RepairMaterial

- nodeProperties
 - .reference
 - .repairMaterial
 - .rowID
- Description
 - Each :RepairMaterial node is one named solvent from the RepairMaterial column from a specific TreatmentEvent row. Where there was a list of repair materials named in the cell, these have been split and made into their own individual nodes during the transformation process.
- Cypher for transformation

```
LOAD CSV WITH HEADERS FROM 'file:///TNA.csv' AS row
WITH row, split(row.RepairMaterial, ",") AS repairMaterials
UNWIND repairMaterials AS repairMaterial
MERGE (n:RepairMaterial {reference: row.Reference,
repairMaterial: repairMaterial,
rowID:row.RowID})
Return n
```

- :Solvents

- nodeProperties:
 - .reference
 - .solvent
 - .rowID
- Description
 - Each :Solvents node is one named solvent from the Solvent column from a specific TreatmentEvent row. Where there was a list of solvents named in the cell, these have been split and made into their own individual nodes during the transformation process.
- Cypher for transformation

```
LOAD CSV WITH HEADERS FROM 'file:///TNA.csv' AS row
WITH row, split(row.Solvents, ",") AS solvents
UNWIND solvents AS solvent
MERGE (n:Solvents {reference: row.Reference,
```

```
solvent: solvent,  
rowID:row.RowID})  
Return n
```

- :Person

- nodeProperties
 - .person
- Description
 - Unique nodes for specific people so not tied to .rowID or .references
- Cypher for Transformation

```
LOAD CSV WITH HEADERS FROM 'file:///TNA.csv' AS row  
WITH DISTINCT row.Person  
CREATE (:Person{person:row.Person});
```

- :PrimaryDamage

- nodeProperties
 - .primaryDamage
 - .reference
 - .rowID
- Description
 - Assigned damage type.
- Cypher for Transformation

```
LOAD CSV WITH HEADERS FROM "file:///TNA.csv" AS row  
WITH row WHERE NOT row.Primarydamage IS null  
MERGE(a:PrimaryDamage {primaryDamage: row.Primarydamage,  
reference: row.Reference,  
rowID:row.RowID})
```

- :SecondaryDamage

- nodeProperties
 - .secondaryDamage
 - .reference
 - .rowID
- Description
 - Assigned damage type.
- Cypher for Transformation

```
LOAD CSV WITH HEADERS FROM "file:///TNA.csv" AS row  
WITH row WHERE NOT row.Secondarydamage IS null  
MERGE(a:SecondaryDamage {secondaryDamage: row.Secondarydamage,  
reference: row.Reference,  
rowID:row.RowID})
```

B4. Creating Relationships for Treatment Star Schema

```
Match (a:TreatmentEvent), (b:Reference)
WHERE a.reference = b.reference
MERGE (a)-[r:INVOLVES]->(b);
```

```
Match (a:Reference), (b:TreatmentEvent)
WHERE a.reference = b.reference
MERGE (a)-[r:WAS_TREATED_DURING]->(b);
```

```
Match (a:TreatmentEvent), (b:Adhesives)
WHERE a.reference = b.reference AND a.rowID = b.rowID
MERGE (a)-[r:WAS_TREATED_WITH]->(b);
```

```
Match (a:TreatmentEvent), (b:RepairMaterial)
WHERE a.reference = b.reference AND a.rowID = b.rowID
MERGE (a)-[r:WAS_TREATED_WITH]->(b);
```

```
Match (a:TreatmentEvent), (b:Solvents)
WHERE a.reference = b.reference AND a.rowID = b.rowID
MERGE (a)-[r:WAS_TREATED_WITH]->(b);
```

```
Match (a:TreatmentEvent), (b:PrimaryDamage)
WHERE a.reference = b.reference AND a.rowID = b.rowID
MERGE (a)-[r:HAS_DAMAGE]->(b);
```

```
Match (a:TreatmentEvent), (b:SecondaryDamage)
WHERE a.reference = b.reference AND a.rowID = b.rowID
MERGE (a)-[r:HAS_DAMAGE]->(b);
```

```
Match (a:TreatmentEvent), (b:Person)
WHERE a.person = b.person
MERGE (a)-[r:WAS_TREATED_BY]->(b);
```

B5. Other Data Cleaning Guidelines

Tip: It's worth applying a `trim()` function to resulting `n.name` to remove any excess whitespaces from before or after the name. For example, if an original list was "apples, oranges, bananas", the extraction would have parsed " oranges" and " bananas" from immediately after the , [comma] delimiter. Note the gap with a leading whitespace between the first quotation mark and first letter of the word.

Appendix C

The NLP-Derived Dataset: Preparation, ETL and Threshold Uses

[C1. Introduction](#)

[C2. Example of Natural Language Processing \(NLP\) using spaCy](#)

[C3. Condition Comments Sample](#)

[C4. Comments Sample](#)

[C5. The Handling of Incorrect or Imperfect Attributions](#)

[C6. Threshold Uses for this NLP-Derived Dataset](#)

[C7. The Transformation and Import Scripts](#)

[C7.1 The NLP Code \(Python\)](#)

[C7.2 Cypher: Load and Map of JSON - NounChunks](#)

[C7.3 Cypher: Load and Map of JSON - Verbs](#)

[C8. Cypher Queries](#)

C1. Introduction

The NLP-derived dataset was employed in Phase 1 and Phase 3 of the case study. See also Appendix H (Phase 1) for how this dataset was modelled in both a linear-based graph structure and as a star schema cluster graph structure. Phase 3 utilises only the cluster-based modelling approach. As an illustrative example of how data was derived, the first entry ("row 1") in the principal TNA CCD dataset, relating to collection item reference "OS 3/26", is presented here to demonstrate how natural language processing was performed to extract the resulting NLP-derived dataset.

C2. Example of Natural Language Processing (NLP) using spaCy

The treatment record captured in "row 1" had missing values for the structured fields "Repair Material", "Adhesives", and "Solvents". However, valuable and semantically relevant content that speaks to these key categories (i.e. to the identification and use of materials) exist in the "Comments" and "Condition Comments" fields. To capture these entities, Natural Language Processing was undertaken using spaCy's small English language model ("en_core_web_sm") and applied to the content of the "Condition Comments" and "Comments" columns to parse and tag linguistic tokens (i.e. words and punctuation) and assign attributes such as parts of speech and syntactic dependencies. Once parsed and tagged, noun chunks and verbs were specifically extracted along with data lineage (i.e. provenance) identifiers corresponding to the TNA CCD dataset row ID and catalogue item reference string. These results were saved as a JSON file. The

returned values are presented in sections C3 and C4 below. They include the row ID from the CCD dataset, the reference number from CCD dataset, the token text as it was passed in, the lemma for that token (i.e. the canonical form for a set of words), the assigned part of speech, the part of speech tag, and the syntactic dependency¹.

Table C01. The ordered list of returned values and assigned attributes

noun chunks	(rowID, ref, chunk.text, chunk.root.text, chunk.root.dep_, chunk.root.head.text)
verbs	(rowID, ref, token.text, token.lemma_, token.pos_, token.tag_, token.dep_)

¹ spaCy uses the universal grammatical relations taxonomy based on the Universal Stanford Dependencies by de Marneffe et al 2014. A key can be found at <https://universaldependencies.org/u/dep/all.html>

C3. Condition Comments Sample

C3.1 Annotated Sample of original text from "Condition Comments" column, row 1, reference OS 3/26:

Original housing: 2 piece box made from millboard with wood side panels held together with nails; box broken with exposed nails & rough edges. Zinc plate wrapped in non-archival corrugated cardboard.

C3.2 Extracted verbs from "Condition Comments" for row 1, reference OS 3/26:

[1, "OS 3/26", "made", "make", "VERB", "VBN", "acl"]
[1, "OS 3/26", "held", "hold", "VERB", "VBN", "acl"]
[1, "OS 3/26", "broken", "break", "VERB", "VBN", "acl"]
[1, "OS 3/26", "exposed", "expose", "VERB", "VBN", "amod"]
[1, "OS 3/26", "wrapped", "wrap", "VERB", "VBN", "ROOT"]
[1, "OS 3/26", "corrugated", "corrugate", "VERB", "VBN", "amod"]

C3.3 Extracted noun chunks from "Condition Comments" for row 1, reference OS 3/26:

[1, "OS 3/26", "Original housing", "housing", "ROOT", "housing"]
[1, "OS 3/26", "2 piece box", "box", "appos", "housing"]
[1, "OS 3/26", "millboard", "millboard", "pobj", "from"]
[1, "OS 3/26", "wood side panels", "panels", "pobj", "with"]
[1, "OS 3/26", "nails", "nails", "pobj", "with"]
[1, "OS 3/26", "box", "box", "appos", "box"]
[1, "OS 3/26", "exposed nails", "nails", "pobj", "with"]
[1, "OS 3/26", "amp", "amp", "conj", "nails"]
[1, "OS 3/26", "rough edges", "edges", "conj", "box"]
[1, "OS 3/26", "Zinc plate", "plate", "nsubj", "wrapped"]
[1, "OS 3/26", "non-archival corrugated cardboard", "cardboard", "pobj", "in"]

C4. Comments Sample

C4.1 Annotated Sample of original text from "Comments" column, row 1, reference OS 3/26:

New housing: 4-flap enclosure with 10mm thick Gatorfoam back board with 30mm wide x 15mm deep black Plastazote foam border & lined with 2mm thick black Plastazote foam adhered with Mowiol 4-88. 4 x E flute board flaps adhered to verso of Gatorfoam board with EVA. Enclosure secured with 2 x 6mm thick cotton tapes & rivets with plastic washers. Plate fully secured within portfolio with piece of 10mm thick black Plastazote foam sitting within frame.

C4.2 Extracted verbs from "Comments" for row 1, reference OS 3/26:

[1, "OS 3/26", "lined", "line", "VERB", "VBD", "ROOT"]
[1, "OS 3/26", "adhered", "adhere", "VERB", "VBN", "acl"]
[1, "OS 3/26", "adhered", "adhere", "VERB", "VBN", "ROOT"]
[1, "OS 3/26", "verso", "verso", "VERB", "VB", "xcomp"]
[1, "OS 3/26", "secured", "secure", "VERB", "VBN", "ROOT"]
[1, "OS 3/26", "rivets", "rivet", "VERB", "VBZ", "conj"]
[1, "OS 3/26", "secured", "secure", "VERB", "VBN", "ROOT"]
[1, "OS 3/26", "sitting", "sit", "VERB", "VBG", "acl"]

C4.3 Extracted noun chunks from "Condition Comments" for row 1, reference OS 3/26:

[1, "OS 3/26", "New housing", "housing", "ROOT", "housing"]
[1, "OS 3/26", "4-flap enclosure", "enclosure", "ROOT", "enclosure"]
[1, "OS 3/26", "10mm thick Gatorfoam", "Gatorfoam", "pobj", "with"]
[1, "OS 3/26", "board", "board", "pobj", "with"]
[1, "OS 3/26", "15mm", "mm", "pobj", "with"]
[1, "OS 3/26", "deep black Plastazote foam border", "border", "ROOT", "border"]
[1, "OS 3/26", "amp", "amp", "conj", "border"]
[1, "OS 3/26", "2mm thick black Plastazote foam", "foam", "pobj", "with"]
[1, "OS 3/26", "Mowiol", "Mowiol", "pobj", "with"]
[1, "OS 3/26", "x E flute board flaps", "flaps", "nsubj", "adhered"]
[1, "OS 3/26", "Gatorfoam board", "board", "pobj", "of"]
[1, "OS 3/26", "EVA", "EVA", "pobj", "with"]
[1, "OS 3/26", "Enclosure", "Enclosure", "nsubj", "secured"]
[1, "OS 3/26", "2 x 6mm thick cotton tapes", "tapes", "pobj", "with"]
[1, "OS 3/26", "amp", "amp", "conj", "tapes"]
[1, "OS 3/26", "plastic washers", "washers", "pobj", "with"]
[1, "OS 3/26", "Plate", "Plate", "nsubj", "secured"]
[1, "OS 3/26", "portfolio", "portfolio", "pobj", "within"]
[1, "OS 3/26", "piece", "piece", "pobj", "with"]
[1, "OS 3/26", "10mm thick black Plastazote foam", "foam", "pobj", "of"]
[1, "OS 3/26", "frame", "frame", "pobj", "within"]

C5. The Handling of Incorrect or Imperfect Attributions

The English natural language model used here has been compiled from and based on English language usage found on the World Wide Web. The specific English natural language model used was the small model (ie. `spacy.load("en_core_web_sm")`), chosen for fast processing time. Although the results contain incorrect and imperfect attributions, this is largely due to the general purpose nature of the language model used and can be improved with retraining of the model using conservation-specific corpora. This is outside the scope of the current study. Nevertheless, the parsing and tagging results provided a substantially workable standard for the current demonstrative purposes. For example, noun chunk parsing was able to capture multi-word entities such as “acid-free tissue” or “non-archival corrugated cardboard”. The derived dataset was not processed any further to remove or improve the content of the dataset. However, threshold uses were identified so as to not risk overfitting the models.

C6. Threshold Uses for this NLP-Derived Dataset

These thresholds also bear in mind the text matching capabilities afforded by the Neo4j system such as regular expressions and fuzzy matching techniques.

Within the scope of this demonstration, semantically irrelevant parsed tokens, such as “amp” were not removed. Additional cleaning of the dataset was not prioritised in this case as these were unlikely to be connected to the other datasets within the composite graph and can be pruned and removed at a later stage. The dependency attributes (see list in Table C02) provide further confidence checks for a human-supervised approach. For example, tokens or chunks with *acl*, *amod*, *appos*, *nsubj*, and *ROOT* were found to be more reliably recognisable and useful whereas *conj* and *xcomp* attributions for parts of speech were found to be less accurate, especially when they pertained to entities as objects.

Table C02. Syntactic relational dependencies (Universal Stanford Dependency Taxonomy) attributed in the samples below.

<i>acl</i>	an adnominal clause or clausal modifier of a noun
<i>amod</i>	an adjectival modifier
<i>appos</i>	an appositional modifier, used between two nominals
<i>conj</i>	a conjunction, that is, the relation between two elements
<i>nsubj</i>	the nominal subject
<i>pobj</i>	prepositional phrase [in Stanford Dependency]; renamed as ‘ <i>nmod</i> ’, nominal identifier, in Universal Dependency.

ROOT	the grammatical relation at the root of the sentence; only one root dependency in every tree
xcomp	an open clausal complement of a verb or adjective

Certain tokens had multiple parts-of-speech attributions, for example, “exposed” and “corrugated” were assigned as both a verb and part of a noun chunk, which either are semantically plausible. When compared to the original text, in cases where tokens were attributed as both part of a noun chunk and a verb, the noun chunk attributions were found correct in the context of the statements. Thus, this served as a general rule, in this case, to favour the noun chunk attributions as correct attributions when there was a dual attribution. However, incorrect attributions were not corrected or removed for this demonstration. Grammatical dependencies deemed “acl” (adnominal clause) were more accurately attributed compared to other attributions (e.g. amod).

C7. The Transformation and Import Scripts

C7.1 The NLP Code (Python)

```
#this function is to output both Verbs and Nounchunks from the #'Conditioncomments' and
'Comments' columns by passing in the source CCD.csv #once and iterating through the
relevant cells using the pandas dataframe method

import pandas as pd
import spacy
import json

nlp = spacy.load("en_core_web_sm")
COMTNounChResults = open("1COMTNounChResults.json", 'a', newline='')
COMTverbResults = open("2COMTverbResults.json", 'a', newline='')
CONDNounChResults = open("3CONDnounChResults.json", 'a', newline='')
CONDverbResults = open("4CONDverbResults.json", 'a', newline='')
data = pd.read_csv("CCD.csv")

# this snippet calls to a specific cell via row index and column name

for index, row in data.iterrows():
    rowID = row['RowID']
    ref = row['Reference']
    comt = str(row['Comments'])
    cond = str(row['Conditioncomments'])
    nlpcomt = nlp(comt)
    nlpcond = nlp(cond)

    for token in nlpcomt:
        spaced = (rowID, ref, token.text, token.lemma_, token.pos_, token.tag_, token.dep_)
        if token.pos_ == "VERB" in spaced:
            json.dump(spaced, 2COMTverbResults)
        for chunk in nlpcomt.noun_chunks:
            nounChunks = (rowID, ref, chunk.text, chunk.root.text, chunk.root.dep_,
chunk.root.head.text)
            json.dump(nounChunks, 1COMTNounChResults)

    for token in nlpcond:
        spaced = (rowID, ref, token.text, token.lemma_, token.pos_, token.tag_, token.dep_)
        if token.pos_ == "VERB" in spaced:
            json.dump(spaced, 4CONDverbResults)
        for chunk in nlpcond.noun_chunks:
            nounChunks = (rowID, ref, chunk.text, chunk.root.text, chunk.root.dep_,
chunk.root.head.text)
            json.dump(nounChunks, 3CONDNounChResults)

1COMTNounChResults.close()
2COMTverbResults.close()
3CONDNounChResults.close()
4CONDverbResults.close()
```

The resulting .json files from the above script were imported into Neo4j using *apoc.load.json* procedures. Tokens, noun chunks and verbs were matched to relevant (:TreatmentEvent) star schema clusters using rowID and reference string ID.

C7.2 Cypher: Load and Map of JSON - NounChunks

```
CALL apoc.load.json("file:///1COMTnounChResults.json")
Yield value as result
MERGE (a:NounChunk{
dataset: 'TNA',
sourceTextType: 'Comments',
rowID:result.result[0],
reference:result.result[1],
chunkText:result.result[2],
chunkRootText:result.result[3],
chunkRootDep:result.result[4],
chunkRootHeadText:result.result[5]
});
```

```
CALL apoc.load.json("file:///3CONDnounChResults.json")
Yield value as result
MERGE (a:NounChunk{
dataset: 'TNA',
sourceTextType: 'Condition Comments',
rowID:result.result[0],
reference:result.result[1],
chunkText:result.result[2],
chunkRootText:result.result[3],
chunkRootDep:result.result[4],
chunkRootHeadText:result.result[5]
});
```

```
MATCH (n:NounChunk)
SET n.reference = toString(n.reference)
SET n.rowID = toString(n.rowID);
```

C7.3 Cypher: Load and Map of JSON - Verbs

```
CALL apoc.load.json("file:///2COMTverbResults.json")
Yield value as result
Merge (a:Verb{
dataset:'TNA',
sourceTextType: 'Comments',
rowID:result.result[0],
reference:result.result[1],
tokenText:result.result[2],
tokenLemma:result.result[3],
tokenPOS:result.result[4],
tokenTag:result.result[5],
tokenDep:result.result[6]
});
```

```
CALL apoc.load.json("file:///4CONDverbResults.json")
Yield value as result
Merge (a:Verb{
dataset:'TNA',
sourceTextType: 'Condition Comments',
rowID:result.result[0],
reference:result.result[1],
tokenText:result.result[2],
tokenLemma:result.result[3],
tokenPOS:result.result[4],
tokenTag:result.result[5],
tokenDep:result.result[6]
});
```

```
MATCH (n:Verb)
SET n.reference = toString(n.reference)
SET n.rowID = toString(n.rowID);
```

C8. Cypher Queries

//Find and Return unique NounChunks in Descending Order of their Frequency Counts

```
Match (a:NounChunk)
Return Distinct a.chunkText, Count(a) as Count Order by Count Desc
```

//Find and Return unique Verbs in Descending Order of their Frequency Counts

```
Match (a:Verb)
Return Distinct a.tokenText, Count(a) as Count Order by Count Desc
```

Appendix D

The Discovery API: Preparation and ETL

[D1. Introduction](#)

[D2. Python Script to Retrieve Catalogue Data from Discovery API](#)

[D3. List of Discovery API Fields](#)

[D4. Level 1 - Department](#)

[D5. Level 3 - Series](#)

[D6. Levels 6 and 7 - Pieces and Items](#)

[D7. Python Script for Iterating Through Batched Ranges in the API GET Request](#)

[D8. List of References with Outlier Reference String Patterns](#)

[D9. Cypher Scripts to Create Relationships between :Discovery nodes](#)

D1. Introduction

Catalogue information for each object or item Reference number from the TNA CCD dataset was retrieved via the public Discovery API as part of the Phase 3 (P3) LPG model.

D2. Python Script to Retrieve Catalogue Data from Discovery API

```
#This script Gets catalogue data from The National Archives Discovery API
#It passes in a known object/item reference number and receives the associated json
#The multiparameter search query in lines 15-16 enable a more accurate result and
lowest possible result count.

import urllib.parse
import requests
import json

url = 'https://discovery.nationalarchives.gov.uk/API/search/v1/records'
s = requests.Session()

#Remember to change input/output file names below to suit

with open('FILE_PATH/LIST_OF_REFERENCE_NUMBERS.csv') as refflist,
open('RESULTS.json', 'a') as results:
    for x in refflist:
        ref_param = {'sps.references': r'"'+ x + r'"', 'sps.departments': x.split()[0],
```



```

        'sps.referenceQuery': r'' + x + r'', 'sps.searchQuery': r'' + x +
r'', 'sps.batchStartMark': r'*'}
        enc_ref = urllib.parse.urlencode(ref_param)
        r = s.get(url, params=enc_ref)
        json.dump(r.json(), results)

results.close()
print(s) #this print command served as a prompt to indicate script completed

```

D3. List of Discovery API Fields

All content retrieved from the Discovery API was mapped to nodes with the fields in Table D2.1 stored as node properties. NB: Not all records have values for every field.

Table D2.1 Discovery API Fields

1	adminHistory
2	altName
3	arrangement
4	catalogueLevel
5	closureCode
6	closureStatus
7	closureType
8	content
9	context
10	corpBodies
11	coveringDates
12	department
13	description
14	documentType]
15	endDate
16	formerReferenceDep
17	formerReferencePro
18	heldBy
19	id
20	mapDesignation
21	mapScale
22	note
23	numEndDate
24	numStartDate
25	openingDate

26	physicalCondition
27	place
28	reference
29	score
30	source
31	startDate
32	taxonomies
33	title
34	urlParameters

D4. Level 1 - Department

D4.1. Preparing the source file:

The list of all Reference strings were processed to split out the first segment of letter(s) of the string. Duplicates were removed with the resulting list saved as "DepartmentsLevel1.csv".

D4.2. Assembling the GET queries:

The API call was made similar to the above code in section D2 save for the added parameter in bold:

```
ref_param = {'sps.references': r'"' + x + r'", 'sps.departments': x.split()[0],
'sps.catalogueLevels': 'Level1','sps.referenceQuery': r'"' + x + r'",
'sps.searchQuery': r'"' + x + r'",'sps.batchStartMark': r'*'}
```

The resulting JSON file was saved as "ResultsLevel1.json".

D4.3. Processing the JSON file before using in Neo4j

Next, I reviewed the json in TextEdit to check if it looks alright. One way to check is to FindAll "records" and see if the text between appearances of "records" is short or long. Long would be having to scroll down to find the next appearance of "records". Note the count of "records". Don't forget to check for "null" in the JSON. Replace these with "[]" if found. Otherwise it will throw an error when trying to import and map in Neo4j.

```
CALL apoc.load.json("file:///ResultsLevel1.json")
Yield value
Unwind value.records as record
MERGE (a:Discovery :Department {
reference:record.reference,
title:record.title,
context:record.context,
content:record.content,
description:record.description,
physicalCondition:record.physicalCondition,
closureStatus:record.closureStatus,
closureType:record.closureType,
```

```

closureCode:record.closureCode,
endDate:record.endDate,
numEndDate:record.numEndDate,
numStartDate:record.numStartDate,
startDate:record.startDate,
urlParameters:record.urlParameters,
department:record.department,
note:record.note,
adminHistory:record.adminHistory,
arrangement:record.arrangement,
mapDesignation:record.mapDesignation,
mapScale:record.mapScale,
catalogueLevel:record.catalogueLevel,
documentType:record.documentType,
coveringDates:record.coveringDates,
openingDate:record.openingDate,
id:record.id,
score:record.score,
source:record.source,
altName:record.altName,
place:record.places,
corpBodies:record.corpBodies,
taxonomies:record.taxonomies,
formerReferenceDep:record.formerReferenceDep,
formerReferencePro:record.formerReferencePro,
heldBy:record.heldBy});

```

D5. Level 3 - Series

D6.1. Preparing the source file:

The list of all Reference strings were processed to split out the first and second segments (before the separator) of the string as shown in Section 7.2.3 of the thesis. Duplicates were removed with the resulting list saved as "SeriesLevel3.csv".

D6.2. Assembling the GET queries:

The API call was made similar to the above code in section D2 save for the added parameter in bold:

```

ref_param = {'sps.references': r'"' + x + r'", 'sps.departments': x.split()[0],
'sps.catalogueLevels': 'Level3', 'sps.referenceQuery': r'"' + x + r'",
'sps.searchQuery': r'"' + x + r'", 'sps.batchStartMark': r'*'}

```

The resulting JSON file was saved as "ResultsLevel3.json".

D6.3. Processing the JSON file before using in Neo4j

Prior to importing the resulting JSON file into Neo4j, it's recommended to review the content (using a text editor, for example). Firstly, using FindAll "records" to gauge the length of content between appearances of "records" as short or long. Long would be having to scroll down to find the next appearance of "records". Also, note the count of "records" compared to the number of reference numbers passed in, this may be higher than expected, however if lower, than not all references were successfully matched. Finally, check for "null" values in the JSON file and replace any "null" values with empty arrays "[]" if found. Otherwise the "null" values will trigger an error when importing and mapping to Neo4j. The following import Cypher requires installing the APOC plugin.

```
CALL apoc.load.json("file:///ResultsLevel3.json")
Yield value
Unwind value.records as record
MERGE (a:Discovery :Series { //change catalogue level name as necessary
reference:record.reference,
title:record.title,
context:record.context,
content:record.content,
description:record.description,
physicalCondition:record.physicalCondition,
closureStatus:record.closureStatus,
closureType:record.closureType,
closureCode:record.closureCode,
endDate:record.endDate,
numEndDate:record.numEndDate,
numStartDate:record.numStartDate,
startDate:record.startDate,
urlParameters:record.urlParameters,
department:record.department,
note:record.note,
adminHistory:record.adminHistory,
arrangement:record.arrangement,
mapDesignation:record.mapDesignation,
mapScale:record.mapScale,
catalogueLevel:record.catalogueLevel,
documentType:record.documentType,
coveringDates:record.coveringDates,
openingDate:record.openingDate,
id:record.id,
score:record.score,
source:record.source,
altName:record.altName,
place:record.places,
corpBodies:record.corpBodies,
taxonomies:record.taxonomies,
formerReferenceDep:record.formerReferenceDep,
```

```
formerReferencePro:record.formerReferencePro,  
heldBy:record.heldBy}} ;
```

D6. Levels 6 and 7 - Pieces and Items

D6.1. Preparing the source file:

As is explained in Section 7.2.3 in the thesis, separators are not indicative of level. However, through a few initial cursory searches, it appears that the object Reference numbers in the TNA CCD dataset are predominantly Level 6 records with a smaller amount of Level 7 records. Therefore the full list of Reference numbers (not including the hyphenated batched set or the outlier group, see table below) was run twice, once specifying 'Level 6' and then 'Level 7' as part of the "ref_param".

The Cypher loading script is the same as above.

D7. Python Script for Iterating Through Batched Ranges in the API GET Request

Those Reference strings with hyphenated components were intended by the recording conservator as a range of References. The following script passes these hyphenated strings but retrieves each Reference consecutively. These batched references do not include the outlier group as listed below.

```
import urllib.parse  
import requests  
import json  
  
# this is the for loop to iterate through hyphenated batches:  
  
sample = open('/FILE PATH/Batches_Refs.csv')  
data = []  
results = open('ResultsBatches.json', 'a')  
url = 'https://discovery.nationalarchives.gov.uk/API/search/v1/records'  
s = requests.Session()  
  
for batch in sample:  
    splitRef = batch.rsplit('/', 1)  
    refRoot = str(splitRef[0])  
    refRange = splitRef[-1]  
    start, end = [int(item) for item in refRange.split('-')]  
    li = list(range(start, end + 1))  
    for i in li:  
        j = str(i)  
        newRef = refRoot + r'/' + j  
        data.append(newRef)
```

```
#this GETs the records from the Discovery API

for x in data:
    ref_param = {'sps.references': r'" + x + r'"', 'sps.departments': x.split()[0],
                'sps.referenceQuery': r'" + x + r'"', 'sps.searchQuery': r'" + x + r'"',
                'sps.batchStartMark': r'*'}
    enc_ref = urllib.parse.urlencode(ref_param)
    r = s.get(url, params=enc_ref)
    json.dump(r.json(), results)

results.close()
print(s)
```

D8. List of References with Outlier Reference String Patterns

These reference strings in the conservation dataset have been split to leave off text strings after the initial reference string, e.g. all things after and including 'Folio' or 'Part'.

Table. D8.1.1

Outlier Reference Strings
HCA 13/141 Folio 1-50
HCA 13/141 Folio 51-368
SC 2/175 Folio 60-90
CO 5/39 Folio 290-291
E 179/364/16 Part 1-16
E 179/364/12 Part 1-11
E 179/364/58 Part 1-5
E 407/38 Folio 187-204
SP 35/1 Part 1 Folio 70-71
DO 195/391 Folio between ff. 21-22
SP 9/37/4-14
HO 45/24514 Folio 55-58
SP 12/23 Folio 19-20
PROB 1/9/1 Folio 1-13
E 101/47/13 Folio 1-2
REQ 4/1/4/1 Folio 1B-2
FO 93/14/4 Part VI-IX

LC 2/4/5 - WRONG
E 402/1 From Tray 2 To 1-8
CO 1047/1091 Part 1-5

D9. Cypher Scripts to Create Relationships between :Discovery nodes

D9.1. Cypher for Linking Level1 with Level3 nodes

```
Match (a:Department),(b:Series)
Where a.department = b.department
Merge (b)-[:L3_HAS_L1_DEPARTMENT]->(a)
Merge (a)-[:L1_HAS_L3_SERIES]->(b);
```

D9.2. Cypher for Linking Level3 with Level6 nodes

```
Match(a:Discovery{catalogueLevel:3}),(b:Discovery{catalogueLevel:6})
WITH *, a.reference AS series, b.reference AS piece
With *, split(piece, '/') AS plist
Where series = plist[0]
//Return series, plist[0] //use this to test before Merge
Merge (a)-[:L3_HAS_L6_PIECE]->(b)
Merge (b)-[:L6_HAS_L3_SERIES]->(a);
```

D9.3. Cypher for Linking Level6 with Level7 nodes

```
Match(a:Discovery{catalogueLevel:6}),(b:Discovery{catalogueLevel:7})
WITH *, a.reference AS piece, b.reference AS item
//Return piece, item //use this to test before Merge
Merge (a)-[:L6_HAS_L7_PIECE]->(b)
Merge (b)-[:L7_HAS_L6_ITEM]->(a);
```

D9.4. Cypher for Linking Remaining Level6 and Level7 nodes to Level1

Not all Reference numbers had Level3 counterparts. In these cases, they were linked directly to the Level1 :Department nodes.

//Island L6 Pieces to L1

```
Match (a:Department),(b:Piece)
With *
Where apoc.node.degree.in(b) = 0 AND apoc.node.degree.out(b) = 0 AND
a.department = b.department
Merge (b)-[:L6_HAS_L1_DEPARTMENT]->(a)
Merge (a)-[:L1_HAS_L6_PIECE]->(b);
```

//Island L7 Items to L3

```
MATCH (a:Series), (b:Item)
With *, apoc.node.degree.in(b) = 0 AND apoc.node.degree.out(b) = 0 As islands
WITH *, a.reference AS series, b.reference AS piece
```

```
With *, split(piece, '/') AS plist  
Where series = plist[0]  
Return series, plist[0] //use this to test before Merge  
//Merge (a)-[:L3_HAS_L7_ITEM]->(b)  
//Merge (b)-[:L7_HAS_L3_SERIES]->(a);
```


Appendix E

The CAMEO Dataset: Preparation and ETL

[E1. Introduction](#)

[E2. Extracting CAMEO data using a URLs list](#)

[E3. Load and Transform \(using Cypher to map into a Neo4j dbms\)](#)

[E4. Sample Cypher Queries](#)

E1. Introduction

This appendix corresponds with Section 7.2.4 of the thesis regarding inclusion of reference data from CAMEO, the Conservation and Art Materials Encyclopedia Online, a publicly accessible wiki (https://cameo.mfa.org/wiki/Main_Page). The content extracted from the html website and used in Phase 1 and Phase 3 models included content up to 25 August 2021.

E2. Extracting CAMEO data using a URLs list

```
import json
import csv
import requests
from bs4 import BeautifulSoup

results = open("CameoContent_20210825.json", 'a', newline='')
datafile = open("Cameo_URLs_20210825.csv", 'r', newline='')

urlReader = datafile.read().splitlines()

for url in urlReader:
    req = requests.get(url)
    soup = BeautifulSoup(req.text, "html.parser")
    head1 = soup.h1.get_text()

#all h1's are added to the listHeads1 list for use later

    listHeads1 = []
    listHeads1.append(head1)

    listHeads2 = []
    listSubText = []
```

*#'head2' finds all h2's skipping the first one at [0] as it's the page 'contents' subheading which we don't need. The index returns the 2nd up to the last h2's found.
#'sections' find all p's immediately after an h2.*

```
head2 = soup.find_all("h2")[1:-1]
sections = soup.select('h2+p')
```

#this for-loop unwinds the head2 list and makes a new list of just the human-readable text

```
for x in head2:
    subheads = x.get_text()
    listHeads2.append(subheads)
```

#this for-loop unwinds the sections list and makes a new list of just the human readable text, ie. the paragraph contents

```
for y in sections:
    subtext = y.get_text()
    listSubText.append(subtext)
```

#using the zip() method to turn the two lists into a dictionary of key:value pairs where the first in each list becomes one pair, and the second and so on. This re-matches the subheadings back with its p content

```
zippy = (zip(listHeads2, listSubText))
c = [dict(zippy)]
```

#using the zip() method again, the dictionary just made is paired with the page title, h1

```
zappy = (zip(listHeads1, c))
b = [dict(zappy)]
```

#everything is dumped into the results json file.

```
json.dump(b, results)
results.close()
```

#the following print commands help in troubleshooting and debugging

```
#print(listHeads2)
#print(listSubText)
#print(c)
#print(b)
```

E3. Load and Transform (using Cypher to map into a Neo4j dbms)

```
CALL apoc.periodic.iterate(  
  "call apoc.load.json('file:///CameoContent_20210825.json')  
  YIELD value as value",  
  "UNWIND [k IN KEYS(value) | {entity: k, props: value[k]}] AS obj  
  Merge (n:CAMEO{entity:obj.entity})  
  Set n += obj.props",  
  {batchSize:500});
```

E4. Sample Cypher Queries

//Find (:Cameo) node by entity name using Regular Expressions

```
MATCH (n:Cameo) WHERE n.entity =~ "(?i).*wheat starch.*"  
RETURN n
```

//Find (:Cameo) node by synonymAndRelatedTerms using Regular Expressions

//The original wiki content has synonyms bound in a list so will need to UNWIND to access each entry

```
MATCH (n:Cameo) WHERE n.synonymsAndRelatedTerms IS NOT NULL  
UNWIND n.synonymsAndRelatedTerms as term  
WITH term, n  
WHERE term =~ "(?i).*boxboard.*"  
RETURN term, n
```

Appendix F

Phase 2: The CIDOC CRM Group: Preparation and ETL

[F1. Introduction](#)

[F2. ETL of CIDOC CRM RDFS serialisations](#)

[F2.1 Configuring Neosemantics](#)

[F3.2 Import via Linked Resource](#)

[F3. Create a crmID to ease Cypher query](#)

F1. Introduction

This appendix contains the Cypher codes to import CIDOC CRM RDFS graphs using the Neosemantics plugin in Neo4j.

F2. ETL of CIDOC CRM RDFS serialisations

F2.1 Configuring Neosemantics

Source: <https://neo4j.com/labs/neosemantics/4.0/config/>

```
CREATE CONSTRAINT n10s_unique_uri ON (r:Resource)
  ASSERT r.uri IS UNIQUE;
```

```
call n10s.graphconfig.init( { handleMultival: "ARRAY" , handleVocabUris:
  "SHORTEN_STRICT" , baseSchemaNamespace:
  "http://www.cidoc-crm.org/cidoc-crm/" , baseSchemaPrefix: "crm" });
```

The "ARRAY" parameter for handleMultival transforms multiple values into an array, otherwise only the first value would be retained. The "IGNORE" parameter for the handleVocabUris means the resulting node labels and relationship types are easier to read. However, if you plan to export as RDF later, it's best to keep handleVocabUris set to "SHORTEN" or "SHORTEN_STRICT" to use a predefined namespace, and specify the baseSchemaNamespace and baseSchemaPrefix, otherwise the default Neo4j base "n4sch" will be used.

Full list of configuration parameters and valid values can be found at <https://neo4j.com/labs/neosemantics/4.3/reference/>

param	value
"handleVocabUris"	"SHORTEN_STRICT"
"handleMultival"	"ARRAY"
"handleRDFTypes"	"LABELS"
"keepLangTag"	false
"keepCustomDataTypes"	false
"applyNeo4jNaming"	false
"baseSchemaNamespace"	"http://www.cidoc-crm.org/cidoc-crm/"
"baseSchemaPrefix"	"crm"
"classLabel"	"Class"
"subClassOfRel"	"SCO"
"dataTypePropertyLabel"	"Property"
"objectPropertyLabel"	"Relationship"
"subPropertyOfRel"	"SPO"
"domainRel"	"DOMAIN"
"rangeRel"	"RANGE"

Figure F1. The graph config profile set for Neosemantics used in this study.

F2.2 Import via Linked Resource

```
CALL n10s.onto.import.fetch(
  "https://cidoc-crm.org/rdfs/7.1.1/CIDOC_CRM_v7.1.1.rdfs", "RDF/XML", {
    languageFilter: 'en'});
```

Expected Results of Import procedure:

terminationStatus	triplesLoaded	triplesParsed	namespaces	extraInfo	callParams
"OK"	3601	3886	null	""	{languageFilter: "en"}

Excluding the languageFilter parameter would load all languages.

Table F.2.1 The CIDOC CRM Versions

Version	No. of Classes	No. of Properties (Relationships)	Release Date, Source & Declarations
v.5.0.4	86	138	RDFS: December 2011 https://www.cidoc-crm.org/sites/default/files/cidoc_crm_v5.0.4_official_release.rdfs Declarations: https://cidoc-crm.org/html/cidoc_crm_v5.0.4.html
v.6.2.1	89	149	RDFS: April 2018 http://www.cidoc-crm.org/sites/default/files/cidoc_crm_v6.2.1-2018April.rdfs Declarations: https://cidoc-crm.org/html/cidoc_crm_v6.2.1.html
v.7.1.1	81	160	RDFS: August 13, 2021, https://cidoc-crm.org/rdfs/7.1.1/CIDOC_CRM_v7.1.1.rdfs Declarations: https://cidoc-crm.org/html/cidoc_crm_v7.1.1.html

F3. Create a crmID to ease Cypher query

The 'crmID' property was created as an additional property per :Resource node to enable easier querying by specifying the E or P prefix to each class or property. This was particularly useful when running FOL (first order logic) as queries to check if the v.7.1.1 import into the LPG model conformed to the definition.

```
Match (a: Class) with a, a.name as splitz
Call{
with a, splitz
Return split(splitz, "_")[0] as crmID
}
with a, crmID
Set a.crmID = crmID;
//Return count(crmID)
```

```
Match (a: Relationship) with a, a.name as splitz
Call{
```

```
with a, splitz
Return split(splitz, "_")[0] as crmID
}
with a, crmID
Set a.crmID = crmID;
//Return count(crmID)
```

Appendix G

The LCD Group: Preparation and ETL

[G1. Introduction](#)

[G2. Neo4j Specifications](#)

[G3. ETL: Importing the LCD Group into Labelled Property Graph](#)

[G4. Parsing Composite Labels for Identifier Strings](#)

[G5. Query-Based Analysis](#)

[G5.1 Objects Graphs](#)

[G5.2 Materials Graphs](#)

[G5.3 Types Graph](#)

[G5.4 Treatment Event Graphs](#)

[G5.5 Technique Searches](#)

[G5.6 Analysing for Trends Over Time](#)

G1. Introduction

Each LCD dataset was imported into a standalone DBMS instance, essentially, each LCD TriG file was used to create a separate database in Neo4j.

G2. Neo4j Specifications

Table G3.1.1 Neo4j Specifications

Datasets	Neo4j version	Plugins Installed
LCD Group	Enterprise 4.3.6 Neo4j Desktop 1.4.9	APOC 4.3.0.3 GDSDL 1.7.2 Neosemantics 4.3.0.1

Source: <https://neo4j.com/labs/neosemantics/4.0/config/>

```
CREATE CONSTRAINT n10s_unique_uri ON (r:Resource)
ASSERT r.uri IS UNIQUE;
```

```
CALL n10s.graphconfig.init({ handleMultival: "ARRAY" , handleVocabUris:
"IGNORE"});
```


G3. ETL: Importing the LCD Group into Labelled Property Graph

Table G3.1.2 Cypher Import Calls For LCD Datasets

Dataset	Cypher Import Call
LCD-BOD	<code>CALL n10s.rdf.import.fetch("https://raw.githubusercontent.com/linked-conservation-data/board-pilot-data/main/bod/individual-records/23197d1.rdf", "RDF/XML", { languageFilter: 'en'});</code>
LCD-LOC	<code>CALL n10s.rdf.import.fetch("https://raw.githubusercontent.com/linked-conservation-data/board-pilot-data/main/loc/individual-records/1657_001.rdf", "RDF/XML", { languageFilter: 'en'});</code>
LCD-TNA	<code>CALL n10s.rdf.import.fetch("https://raw.githubusercontent.com/linked-conservation-data/board-pilot-data/main/tna/individual-records/ADM1-2628.rdf", "RDF/XML", { languageFilter: 'en'});</code>
LCD-SUL	<code>CALL n10s.rdf.import.fetch("https://raw.githubusercontent.com/linked-conservation-data/board-pilot-data/main/sul/individual-records/1724_illustrationof_birds1865.rdf", "RDF/XML", { languageFilter: 'en'});</code>

Table 3.1.3 Import Validation

Dataset	Triples Loaded/Parsed	Distinct Node Count, not including :GraphConfig node	Distinct Relationship Count
LCD-BOD	749	2449	5481
LCD-LOC	335	1706	3611
LCD-TNA	250	2118	4611
LCD-SUL	247	2218	5753

G4. Parsing Composite Labels for Identifier Strings

`Match (a:`E22_Man-Made_Object`) Return a.label`

Find and Return all E22 records pertaining to the same object

`Match (a:`E22_Man-Made_Object`)`

`Unwind a.label as labs`

`With labs`

Where labs = ~ "(?i).*Arch.B.c.4.*"

Return labs as Item

//The central hub node pattern is "Book (Stanford, referenceNumber)"

G5. Query-Based Analysis

G5.1 Objects Graphs

Graph with Neighbours

Match p= (a:E22_Man-Made_Object)-[r]-(b) Return p

Graph of only E22 nodes

Match (a:E22_Man-Made_Object) Return a

G5.2 Materials Graphs

Graph with Neighbours

Match p= (a:E57_Material)-[r]-(b) Return p

As a list (no duplicates)

Match (a:E57_Material) Return Distinct a.label

G5.3 Types Graph

Graph with Neighbours

Match p= (a:E55_Type)-[r]-(b) Return p

As a list (no duplicates)

Match (a:E55_Type) Return Distinct a.label

G5.4 Treatment Event Graphs

Graph of only E11 nodes

Match (a:E11_Modification) Return a

Graph with Neighbours

Match p= (a:E11_Modification)-[r]-(b) Return p

As a list (no duplicates)

Match (a:E11_Modification) Return Distinct a.label

G5.5 Technique Searches

Only one RETURN clause can follow a query, however, alternative RETURN clauses to return the results displayed in different ways are provided here as commented out ("//") text.

Strategy 1: Find by relationship P32_used_general_technique

Match p= (a)-[r:P32_used_general_technique]->(b)

```
Return Distinct b.label
//Return Count (p)
//Return p
//Return a.label, b.label
```

Strategy 2: Find by relationship P33_used_specific_technique

```
Match p= (a)-[r:P33_used_specific_technique]->(b)
```

```
//Return Count (p)
//Return p
//Return a.label, b.label
```

Strategy 3: Find class E29_Design_or_Procedure

```
Match p= (a:E29_Design_or_Procedure)-[r]->(b)
```

```
Return p
//Return a.label, b.label
```

Strategy 4: Find by variable length path from E55 via P32 to E57_Material

```
Match p= (a:E55_Type)-[r:P32_used_general_technique]->(b)-[s*1..3]->(c:E57_Material)
```

```
Return p
//Return a.label, c.label
```

Strategy 5: Find by variable length path from E55 via P33 to E57

```
Match p= (a:E55_Type)-[r:P33_used_specific_technique]->(b)-[s*1..3]->(c:E57_Material)
```

```
Return p
//Return a.label, c.label
```

Strategy 6: Find by variable length path from specific E55 via P32 to E57

```
Match p= (a:E55_Type)-[r:P32_used_general_technique]->(b)-[s*1..3]->(c:E57_Material)
```

```
Where a.label = ["board reattachment"]
```

```
Return p
//Return Distinct c.label, Count(c.label) Order by Count(c.label) Desc
```

Strategy 7: To Find and Match using .csv list of 15 techniques identified by Velios and St. John (2022)

//Before running this query, ensure that the .csv list to compare to has been placed into the Neo4j DBMS import folder.

//Comparing nodes to a CSV list with Regex

```
LOAD CSV FROM 'file:///LCDtechniquesList.csv' AS techlist
```

```
Unwind techlist as tech
```

```
With tech
```

```
Match p=(a)-[r]->(b)
```

Where a.label is NOT NULL

Unwind a.label as listItem

With tech, listItem, a, b, p

Where listItem = ~"(?i).*" + tech + ".*"

Return p // Returns found techlist nodes and their immediate neighbours.

//Return a.label //Returns only the node label for a match; results as a table and not visualised as a graph.

//Return Count(listItem) //Instances of matches with techlist string to the rdfs:label of any node.

G5.6 Analysing for Trends Over Time

Count of E52_Time-Span nodes

Match (a:`E52_Time-Span`) RETURN Count(a)

Variable path length query to identify DateTime with Materials

MATCH p= (a:`E52_Time-Span`) -[r*3]- (b:E57_Material)

Different Return parameters using above path length query:

RETURN COUNT(p)

RETURN a.P82a_begin_of_the_begin as StartDate, a.P82b_end_of_the_end as EndDate, a.label as TimeLabel, b.label as Material ORDER BY StartDate

RETURN DISTINCT b.label as Material

Match p= (a:`E52_Time-Span`) -[r*1..2]- (b:E57_Material)

RETURN Distinct b.label as MaterialType, min(a.P82b_end_of_the_end) as EndDateMin, max(a.P82b_end_of_the_end) as EndDateMax Order by EndDateMin

MATCH p= (a:`E52_Time-Span`) -[r*1..2]- (b:E57_Material) RETURN

a.P82a_begin_of_the_begin as StartDate, a.P82b_end_of_the_end as EndDate, b.label as MaterialType order by EndDate

Variable path length query to identify DateTime with Techniques

MATCH p= (a:`E52_Time-Span`) -[r*3]- (b)-[s:P32_used_general_technique]- (c)

RETURN Distinct b.label as MaterialType, min(a.P82b_end_of_the_end) as EndDateMin, max(a.P82b_end_of_the_end) as EndDateMax Order by EndDateMin

MATCH p=(a:`E52_Time-Span`)-[r*3]- (b)-[s:P32_used_general_technique]- (c)

RETURN a.P82a_begin_of_the_begin as StartDate, a.P82b_end_of_the_end as EndDate,

c.label as techniqueType

RETURN DISTINCT labels(b) //To determine treatment event class

RETURN DISTINCT b.label //Treatment event label

RETURN DISTINCT labels(c) //To determine technique type (E55_Type)

RETURN DISTINCT c.label //E55_Type label

MATCH p=(a:`E52_Time-Span`)-[r*3]-(b)-[s:P33_used_specific_technique]-(c)

RETURN COUNT(p)

//REturn p Limit 10

//REturn c.label as techniqueType, min(a.P82a_begin_of_the_begin) as
minStartDate, max(a.P82a_begin_of_the_begin) as maxStartDate,
min(a.P82b_end_of_the_end) as minEndDate, max(a.P82b_end_of_the_end) as
maxEndDate

Appendix H

The Phase 1 Models

[H1. Introduction](#)

[H2. Model A - TNA CCD only](#)

[H3. Model B - Applying NLP](#)

[H4. Model C: TNA CCD + CRM v6.2.1 \(ETL2\) + NLP star schema + CAMEO](#)

[H4.1. Eigenvector Centrality and Directionality](#)

[H4.2. Conditional Formatting to Visualise Eigenvector Centrality Results](#)

H1. Introduction

Phase 1 consisted of small-scale tests to identify the requirements for data, data cleaning, and data wrangling to adequately prepare a composite property graph for conservation purposes. While there is limited usefulness to running algorithmic queries on these graphs due to their small size, as any results would not be statistically representative, such trial builds are informative and allow for the sampling and reviewing of heterogeneous datasets to determine a graph and database design strategy. This phase is essential for highlighting pre-processing requirements and ETL (extract, transform, load) data integration pipeline sequences.

Each section to follow will describe a small-scale model including what data or metadata was used to build it and the rationale or hypothesis the resulting graph database can be used to test or inform, such as query design. The results have been gathered from a variety of shallow validation methods including visual, calculable and/or query-based analysis (i.e. using validation questions). Finally, the resulting build parameters are interpreted in terms of the core premise and specific initial hypothesis behind each model. The scope of this preliminary phase was in trialing how to bring the case study data of specific instances together with the other categorical and ontological data components into a composite labelled property graph and to identify any further clarifications necessary for such a process.

Each trial graph model was built as a separate DBMS (database management system) instance. The validation questions are derived from the TNA research interests, specifically regarding quantification of materials, techniques, and individual objects and trends, interpreted as quantification over time, i.e. frequency, and historical trends, and to identify any trends specific to individual objects and collections or departments.

Initial validation questions were:

- VQ1. Which treatment materials were most often used?
- VQ2. Which techniques can be identified?
- VQ3. Are there patterns/frequencies of material or technique use over time?
(including clustering of materials/techniques within a specific temporal range?)
- VQ4. Are there patterns/frequencies in objects returning for treatment?
- VQ5. Are there patterns/frequencies by departments or collections that require conservation?

Table H.1.1 Content Datasets





Key	Dataset
	TNA CCD
	NLP-derived
	CIDOC CRM
	CAMEO

Table H.1.2. Overview of Model Content

Model	Contents
Model A	
Model B	 
Model C	   

H2. Model A - TNA CCD only

This first model, Model A, was created to review the TNA CCD dataset and its contents as data mapped from the original .csv onto nodes in the Neo4j platform. The initial ETL procedures consisted of importing directly from the .csv file using the existing heading to map node properties. An additional node property was created to record the row number from this file for use as a unique identifier specific to the source data file. A description of the .csv file with headers and pre-processing procedures can be found in Appendix B. The initial import resulted in a database with 5,860 (:TreatmentEvent) nodes and no relationships (Figure H2.1).

However, while it is possible to run aggregation and filtering queries on the data nodes, for example, returning all events that used "Gelatine" as an adhesive (see Figures H2.2, H2.3):

Match (a:TreatmentEvent)

Where a.adhesives IS NOT NULL and a.adhesives =~ "(?i).*gelatine.*"

Return a

```
//Or Return a.adhesives, a.reference
```

keeping all data content on isolated nodes does not take full advantage of the graph paradigm.



Figure H2.1 Visualisation of a sample of (:TreatmentEvent) nodes containing imported TNA CCD content.

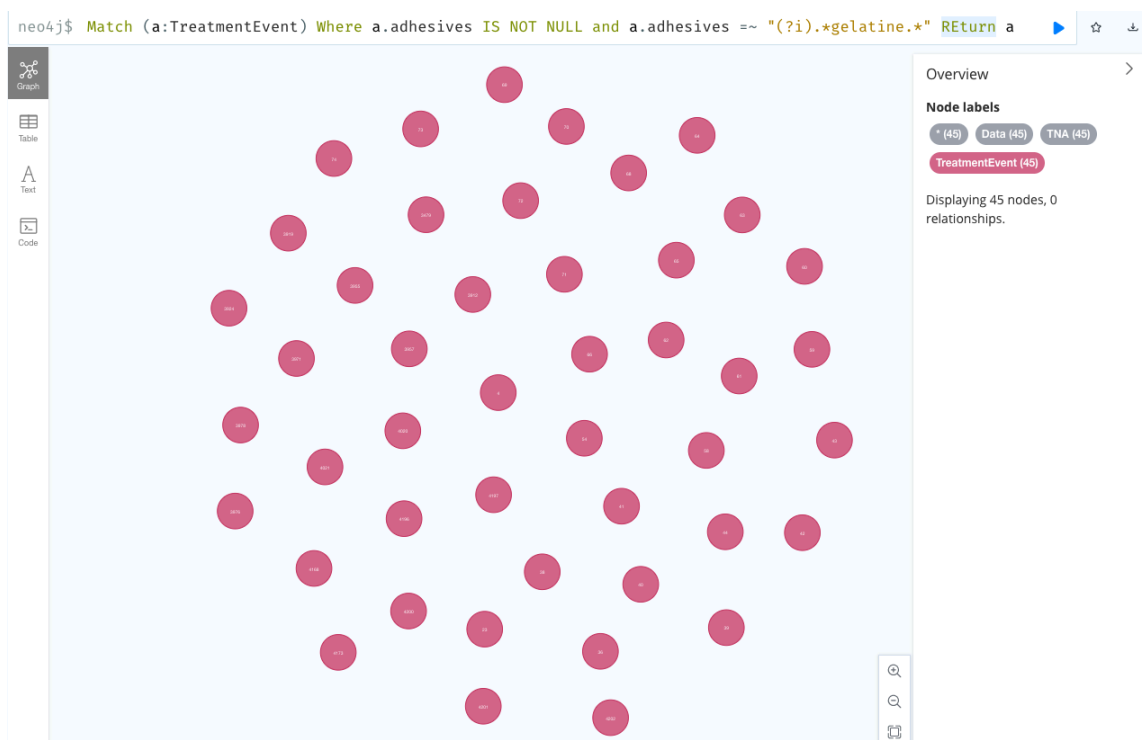


Figure H2.2 Results of query to find records where "gelatine" was used as an adhesive.

```
neo4j$ Match (a:TreatmentEvent) Where a.adhesives IS NOT NULL and a.adhesives =~ "(?i).*gelatine.*" RETURN a.adhesives, a.reference
```

	a.adhesives	a.reference
1	"Gelatine-Gelita gelatine (food)"	"E 101/547/5"
2	"Gelatine (food grade), Shoufu"	"CO 318/76"
3	"Gelatine 180 Bloom"	"C 66/3437"
4	"Gelatine 287 Bloom"	"HO 161/4"
5	"Gelatine (food grade), Klucel G, MHPC 400, Shoufu, Watercolour paints"	"HCA 30/722"
6	"Gelatine 225 Bloom"	"COPY 1/16"
7	"Gelatine 225 Bloom"	"COPY 1/18"
8	"Gelatine 225 Bloom"	"COPY 1/21"
9	"Gelatine 225 Bloom"	"COPY 1/22"
10	"Gelatine 225 Bloom"	"COPY 1/23"
11		

Started streaming 45 records after 1 ms and completed after 6 ms.

Figure H2.3 Results as Figure H2.2, with the added result specification for the adhesives property and the reference property.

As the datasets and derived-distinct-value subdatasets for Person, Material, and Reference were relatively small, counts of each were used to confirm ETL was successful.

Despite the lack of relationships, it was still possible to query the data and return useful quantifiable metrics and some trends, for example, quantification of distinct adhesives.

This also revealed some values, as extracted directly from the original .csv, were in list (array) format. Therefore, it was decided that ETL will expand to include modelling of some of the spreadsheet columns and their values. Further added value in doing this was improvements from a query standpoint. That is, otherwise, every MATCH clause would require an UNWIND clause in case there were embedded lists in the node property values, thereby increasing the use of computational resources such as memory and processing time. The result followed a star schema structure, as previously detailed in section 4.3.5, where more decomposed data content was represented by a more connected data structure.

The star schema representation leverages the graph-based paradigm and provides the added advantage to visually explore results. While retaining the full treatment record on the (:TreatmentEvent) nodes, this allows for node-specific aggregation and filtering queries to continue to work. Together, the star schema around the (:TreatmentEvent) hub node provides a means to refer back to the full record, while also being able to conduct graph-based analysis on aspects of the data (see Figure H2.6 and H2.7).

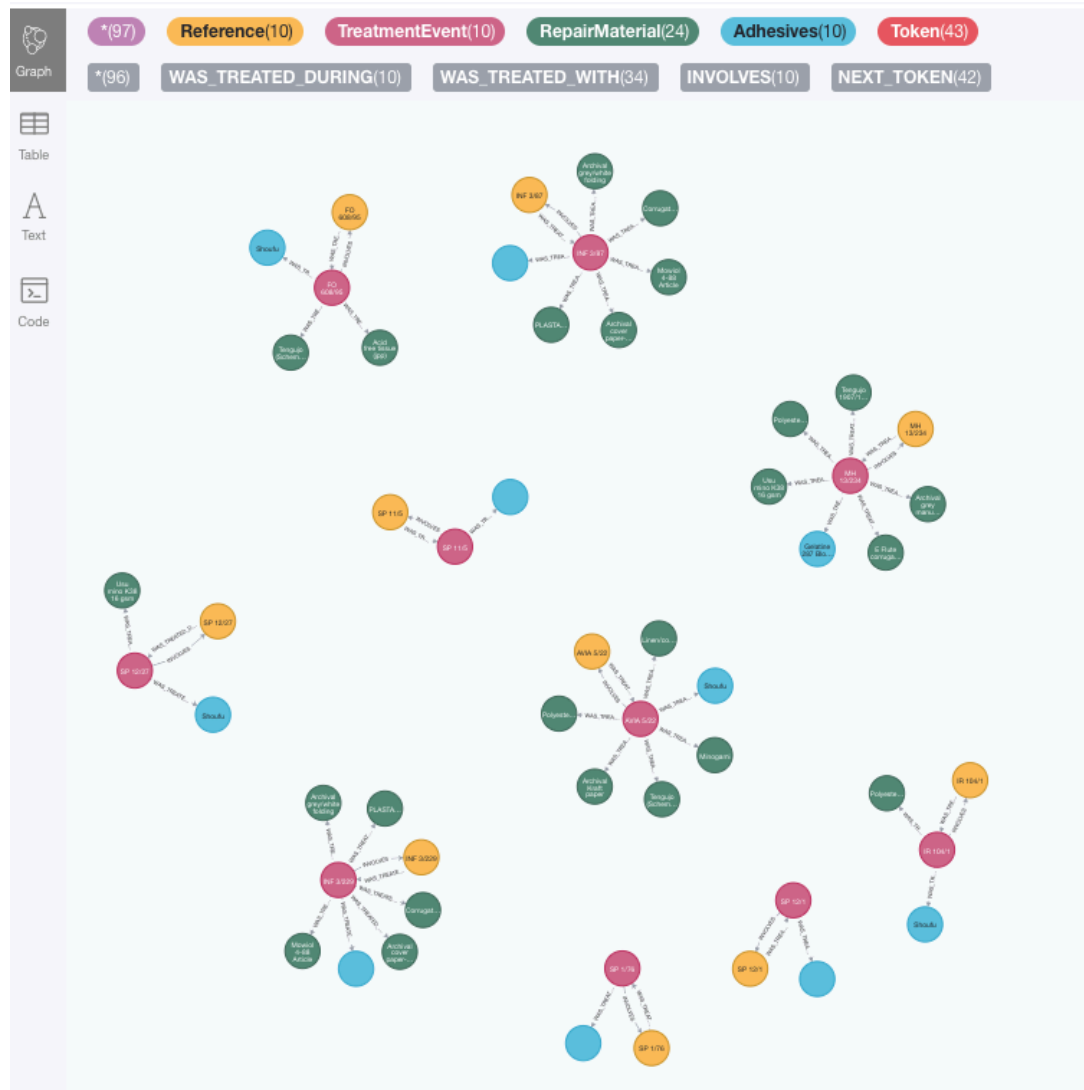
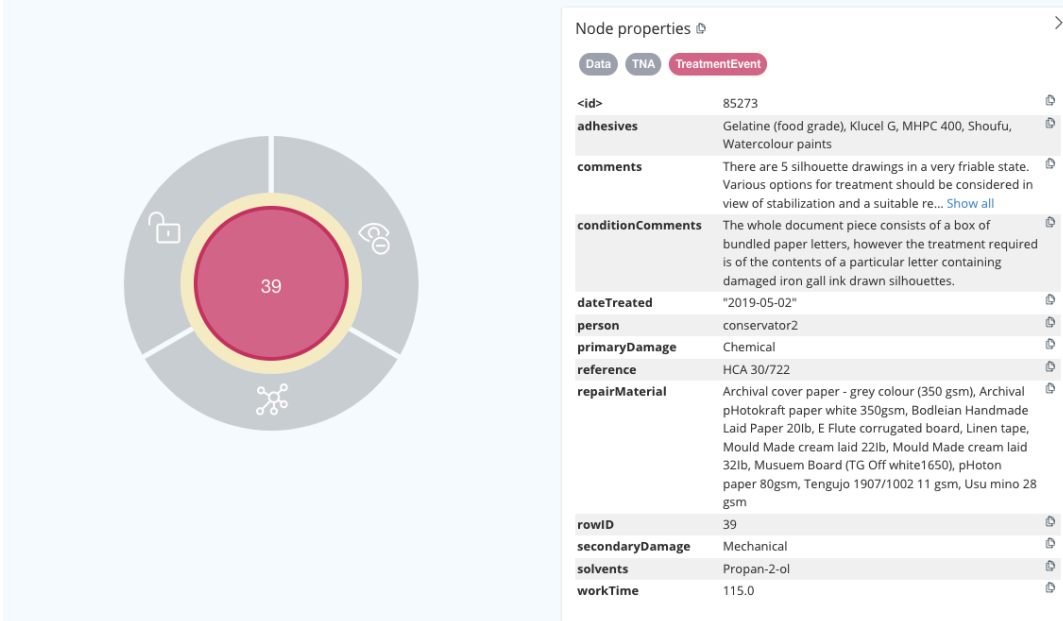


Figure H2.4. Treatment Event Star Schema

After representing the rows in star schema, queries were much simpler and better leveraged Cypher, which is syntactically path-oriented. The full data row content remained on the TreatmentEvent node (see Figure H2.5). This redundancy proved useful as confirmation that extracted instances represented by adjacent nodes for Person, Material, Reference, etc. were linked correctly with the relevant treatment event. From a cognitive and computational perspective, the combined use of a star schema representation where the hub node retains the same keys and values as node properties

helps to differentiate the treatment event and its attributes and the attribute as a conceptually unique thing with the potential for having its own attributes. For example, the treatment event has material x as a material attribute and stored as a node property. Looking more specifically at material x, it too can have attributes such as the name of the manufacturer, preparation, or amount used. The immediacy of the visual results improved the ability to visually explore patterns and identify visual landmarks and potential points of interest.



Node properties

Data TNA TreatmentEvent

<id>	85273
adhesives	Gelatine (food grade), Klucel G, MHPC 400, Shoufu, Watercolour paints
comments	There are 5 silhouette drawings in a very friable state. Various options for treatment should be considered in view of stabilization and a suitable re... Show all
conditionComments	The whole document piece consists of a box of bundled paper letters, however the treatment required is of the contents of a particular letter containing damaged iron gall ink drawn silhouettes.
dateTreated	"2019-05-02"
person	conservator2
primaryDamage	Chemical
reference	HCA 30/722
repairMaterial	Archival cover paper - grey colour (350 gsm), Archival pHotokraft paper white 350gsm, Bodleian Handmade Laid Paper 20lb, E Flute corrugated board, Linen tape, Mould Made cream laid 22lb, Mould Made cream laid 32lb, Musuem Board (TG Off white1650), pHoton paper 80gsm, Tengujo 1907/1002 11 gsm, Usu mino 28 gsm
rowID	39
secondaryDamage	Mechanical
solvents	Propan-2-ol
workTime	115.0

Figure H2.5. TreatmentEvent node with full row of data content mapped as properties to the node.

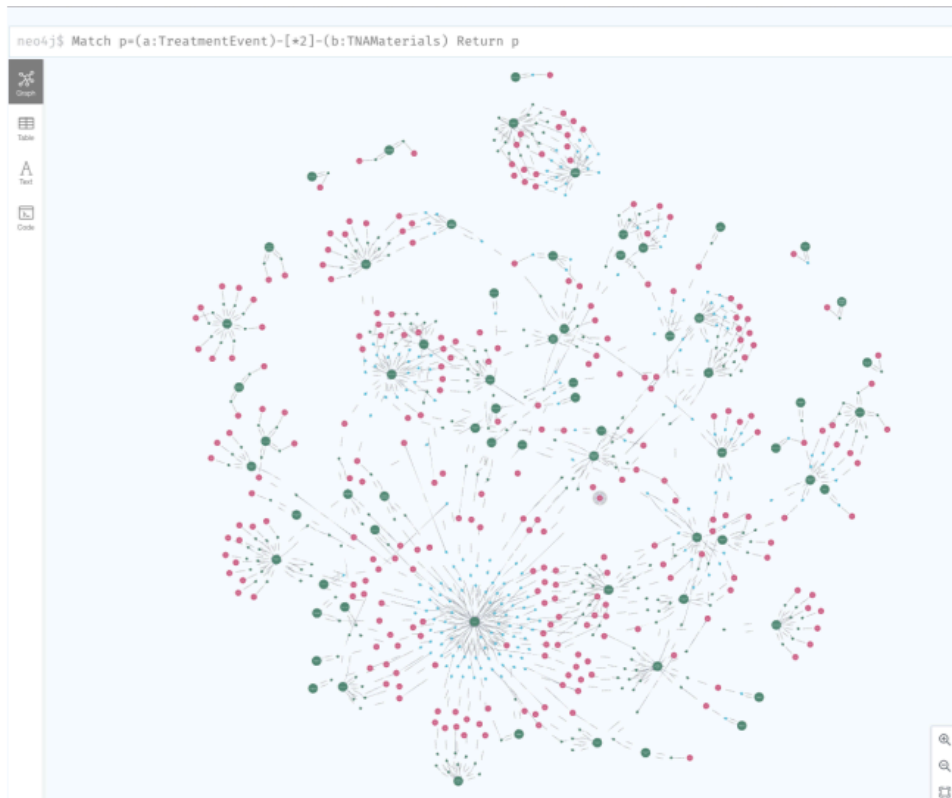


Figure H2.6. Materials Graph. Visualisation of the network of treatment event (pink) nodes and material type (green) nodes.

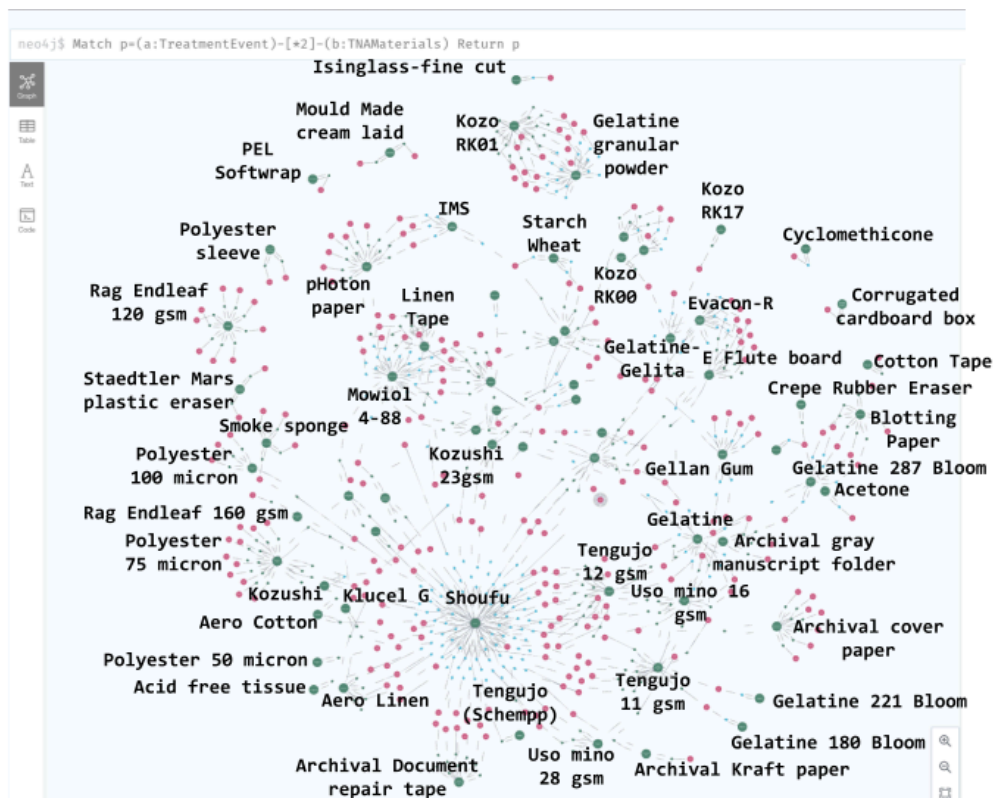


Figure H2.7. Annotated Materials Graph

H3. Model B - Applying NLP

Model B introduces derived data that has undergone natural language processing using a sample of ten data rows from the TNA CCD dataset. Content from the “Comments” and “Condition Comments” columns were processed using spaCy, a Python package for Natural Language Processing (NLP). See Appendix C for details on the preparation and ETL procedures, the spaCy code and the Cypher code for importing the results into the Neo4j database. The impetus for using NLP-derived data was to address the “issue of free-text” and demonstrate a means to use existing textual content to enrich the Model A star schema graphs. This section provides preliminary investigations that ultimately led to the final LPG prototype model described in chapter 7. This model adopted Neill and Kuczera’s (2019) linear token graph approach, using their `:NEXT_TOKEN` relationship, for encoding sequential textual data in a graph to improve the connectivity potential by deriving additional relevant nodes (see Figures H3.1 and H3.2). This allows mentions of conservation materials and techniques in the ‘Comments’ or ‘Condition Comments’ to be represented as nodes connected to each (`:TreatmentEvent`) star schema. This is particularly enriching for where there were no explicitly recorded materials in the ‘Repair Materials’, ‘Adhesives’, and ‘Solvents’ columns in the original data.

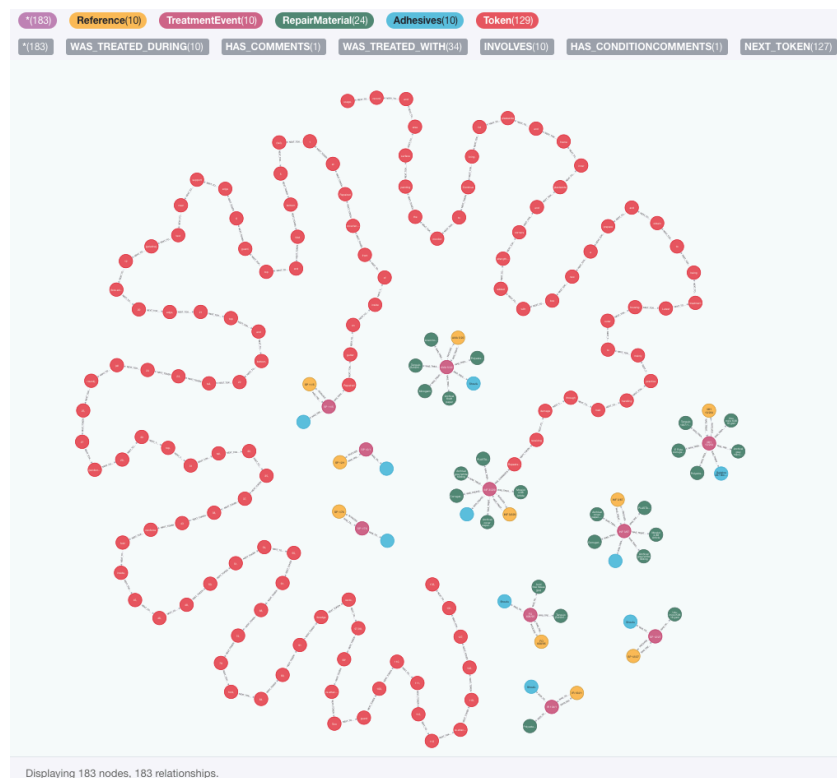


Figure H3.1. Visualisation of linear tokens graphs appended to two (`:TreatmentEvent`) clusters.

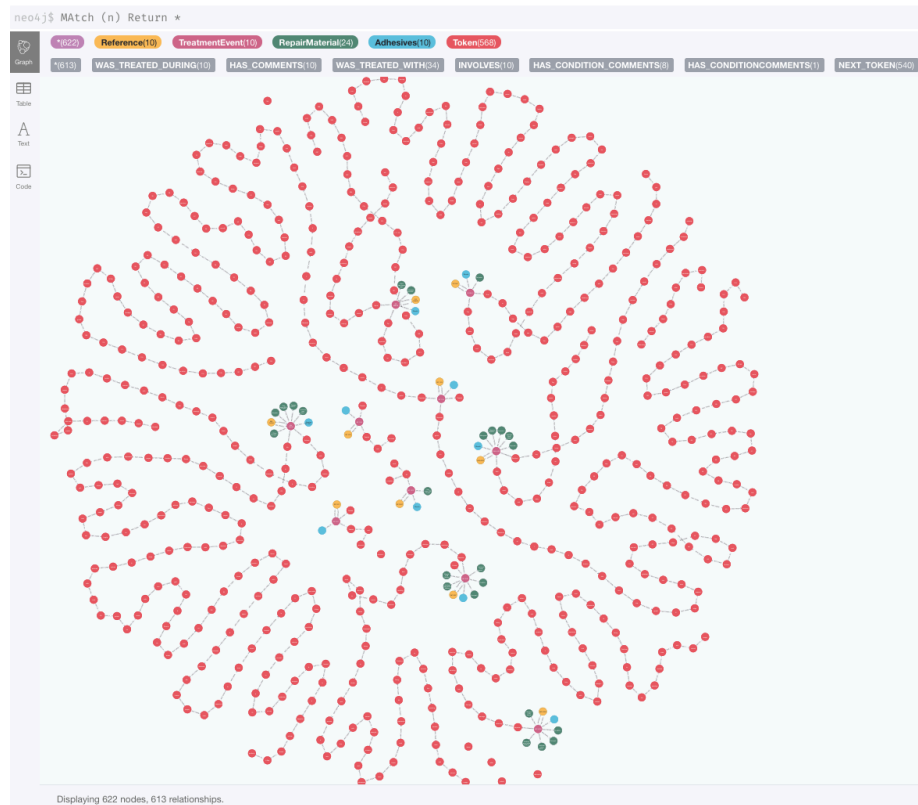


Figure H3.2 Visualisation of all ten (:TreatmentEvent) star schema clusters with both 'Comments' and 'Condition comments' content tokenised and modelled as trailing linear graphs.

The transformation of full text into sequential tokens increased the node count for the database as expected, providing new node content for network development. However, for the purposes of this study, a strictly linear approach that preserves the sequential structure of text was not necessary. Firstly, decomposing all text to tokens meant that the semantic content in multi-token terms, for example "acid-free tissue", were disaggregated. Querying for these specific multi-token terms would, under this model, always require pattern matching on paths instead of a single node. While this is feasible, the derived content also included many (:Token) nodes that do not contribute the same quality of semantic content, for example, where punctuation has been tokenised, and therefore while the node count increased, the overall enrichment potential afforded by these new nodes to the graph was unbalanced which has consequences in terms of the computational resources as the database size increases leading to querying and processing through a larger database that contains unnecessary content. Nevertheless, the creation of nodes from textual mentions of conservation materials and techniques remained a valid source for graph enrichment. Therefore, an alternative approach using star schema clusters to achieve this was implemented in Model C.

H4. Model C: TNA CCD + CRM v6.2.1 (ETL2) + NLP star schema + CAMEO

Model C sought to evolve the modelling of the NLP-derived content into star schema clusters (see Figure H4.1). Model C was also used to experimentally apply the CIDOC CRM (v.6.2.1) to simulate the mapping process where discrete data content is mapped to CIDOC CRM classes. The simulation was implemented by building direct relationships between CIDOC CRM classes and the TNA CCD dataset. (ETL and import details can be found in Appendix F.) The NLP-derived data content was not included in the simulated mappings (i.e. relationship creations).

The results of the NLP-derived star schema clusters provided a more conducive structure to creating further connections between the (:NounChunk) and (:Verb) nodes with categorical nodes such as (:Cameo), and therefore, improved visual results for star schema clusters around general terms.

On the other hand, the experiment in simulated “mapping” of the CIDOC CRM by explicitly creating a relationship between a data node and a CIDOC CRM node led to unusual results when eigenvector centrality was used to assess the resulting structure (see Figure H4.2).

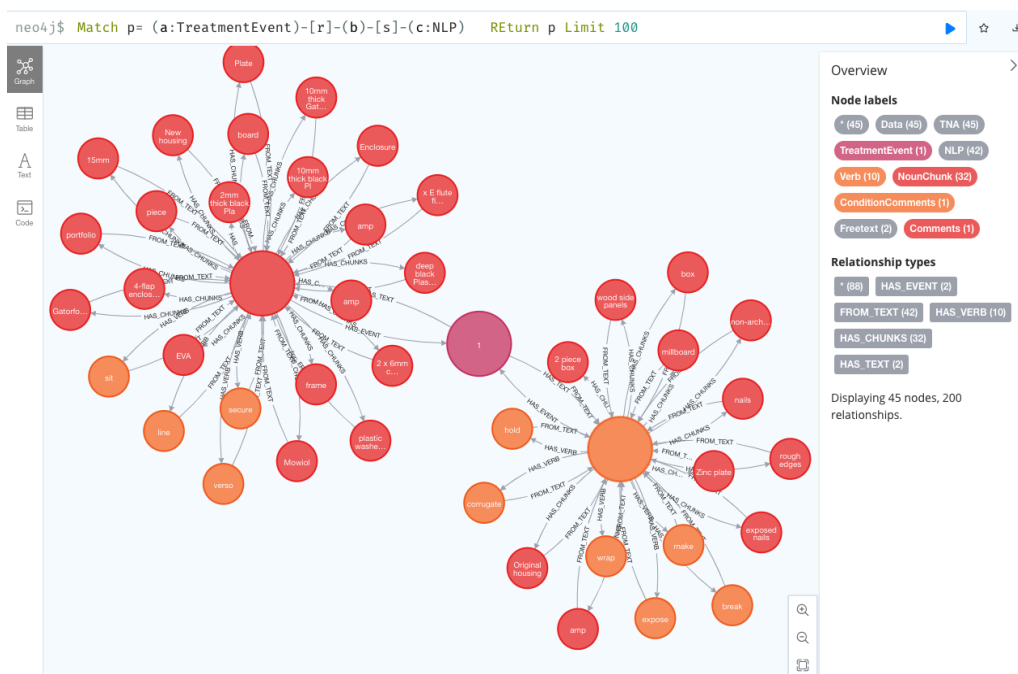


Figure H4.1 Free-text noun chunks and verbs extracted from “Comments” and “ConditionComments” and mapped into a star schema structure where the hub nodes are the (:Comments) and (:ConditionComments) nodes (large orange and red nodes). These (:NLP) hub nodes are themselves connected to the original (:TreatmentEvent) node.

Table H4.1.1 shows which TNA dataset nodes were explicitly linked with which corresponding CIDOC CRM class nodes using a [skos_semanticRelation] type relationship. However, this mapping correspondence was updated in Phase 3 (Chapter 7) based on the findings from Phase 2 (chapters 6 and 7).

Table H4.1.1. Preliminary Mappings to CIDOC CRM. [This has now been superseded by Phase 3 (P3) Mapping Schema as some are incorrect pairings.]

TNA Dataset Node Label	CIDOC CRM Class Node with Specific Name Property, (:Class{name:"[as below]"})
(:TreatmentEvent)	E7_Activity
(:Reference)	E42_Identifier
(:RepairMaterial)	E57_Material
(:Adhesive)	E57_Material
(:Solvent)	E57_Material
(:Person)	E39_Actor
(:PrimaryDamage)	E14_Condition_Assessment
(:SecondaryDamage)	E14_Condition_Assessment

H4.1. Eigenvector Centrality and Directionality

Section H4.3 below shows how the eigenvector centrality results were assessed to reveal a directionality problem that resulted from the import of the CIDOC CRM RDFS graph. This was detected using validation queries that tested the traversal of the graph from :Class node to :Relationship nodes. This highlighted how Neosemantics interpreted the RDFS encoding. An unexpected result was that :Relationship nodes resulted with nearly no incoming edges. To correct for this problem, that is, to calibrate a graph model consisting of any RDFS-derived subgraphs, two reciprocal edges, :xDOMAIN and :xSCO, were created. Firstly, Figure H4.3(a) below shows the original results (ETL 1) where no reciprocal edges were created to compensate for the lack of incoming edges to (:Relationship) nodes. Secondly, Figure H4.3(b) shows the difference to the eigenvector centrality results after :xDOMAIN edge was added (ETL1.5). Finally, Figure H4.3(c) shows the difference in the results once both :DOMAIN and :xSCO reciprocal edges (ETL2) were added to return the CIDOC CRM model to its intended semantic construct.

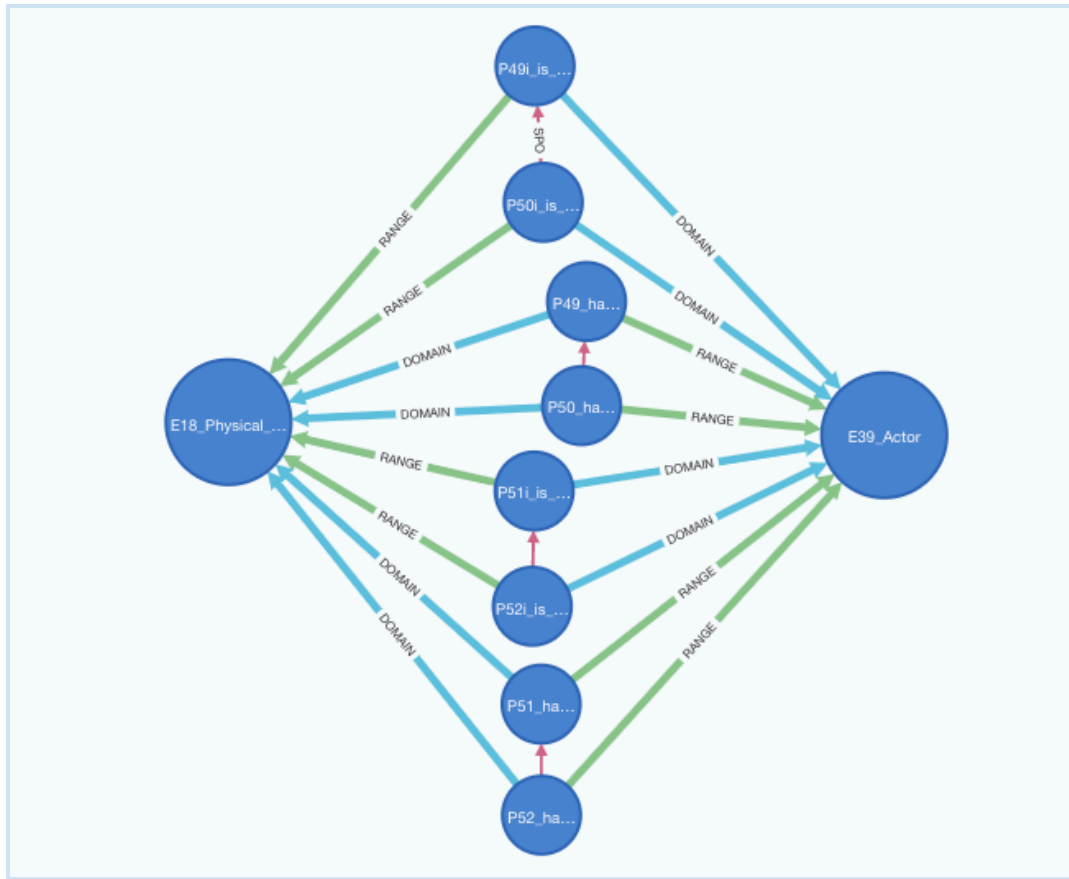


Figure H4.2 Visualisation of the results of importing RDFS into Neo4j's LPG model. Note that the RDF property, i.e. (:Relationship), nodes along the center have outgoing [:DOMAIN] and [:RANGE] edges that point to their respective classes due how RDFS declares domains and ranges. However, the semantic model should have an incoming edge from the class with the Domain edge.

These preliminary findings in Phase 1, contributed to the additional investigations into the CIDOC CRM RDFS graph and into CIDOC CRM-mapped data which will be presented in chapters 5 and 6 (Phase 2). These trials have highlighted the need to review the CIDOC CRM graph(s) in greater detail and to compare versions to determine if the CIDOC CRM graph changes from version to version. It also highlights a significant ETL transformation step necessary if considering to use the CIDOC CRM as a subgraph in an LPG model (see Chapter 5, Section 5.2).

H4.2. Conditional Formatting to Visualise Eigenvector Centrality Results

Table 4.1.2. Results Key

Color	Highlighted Node Type
red	CRM Relationship nodes
blue	CRM Class nodes
green	Person nodes
purple	Treatment Event nodes
yellow	Associated nodes, ie. Reference, Materials, DateTreated, etc that isn't listed elsewhere.
white	RowID nodes

Table 4.1.3. Conditional Formatting Rules to Visually Review Eigenvector Centrality Results

Color	Highlighted Node Type	Format Style	Format Rules	Explanation of rules	Applied to range
red	CRM Relationship nodes	Light red background with black font	Custom formula is =REGEXMATCH(D1,"P[\d]")	Find on "P" as all properties have a P followed by digits.	D1:D33446
blue	CRM Class nodes	Light blue background with black font	Custom formula is =REGEXMATCH(D1,"E[\d]")	Find on "E" followed by digits as that's all CRM entity prefixes	D1:D33446 Aka "name" column
green	Person nodes	Light green background with black font	Text contains person	Specifically looking for "person" node label	ibid
purple	Treatment Event nodes	Purplish-pink background with black font	Custom formula is =AND(REGEXMATCH(D1,"reference"), H1<>"")	This finds TreatmentEvents based on the dataset, with the rule it's anything that gives a reference property and where H column is not empty. This is a fudge. I split to columns. All other nodes would have less properties. So it doesn't matter what's in column H, if there was something, it was a TE node as only TE nodes held that	ibid

				many props at the time.	
yellow	Associated nodes, ie. Reference, Materials, DateTreated, etc that isn't listed elsewhere.	Light yellow background with black font	Custom formula is =REGEXMATCH(D1,"reference")	Finds :Reference nodes based on property	ibid
white	RowID nodes				

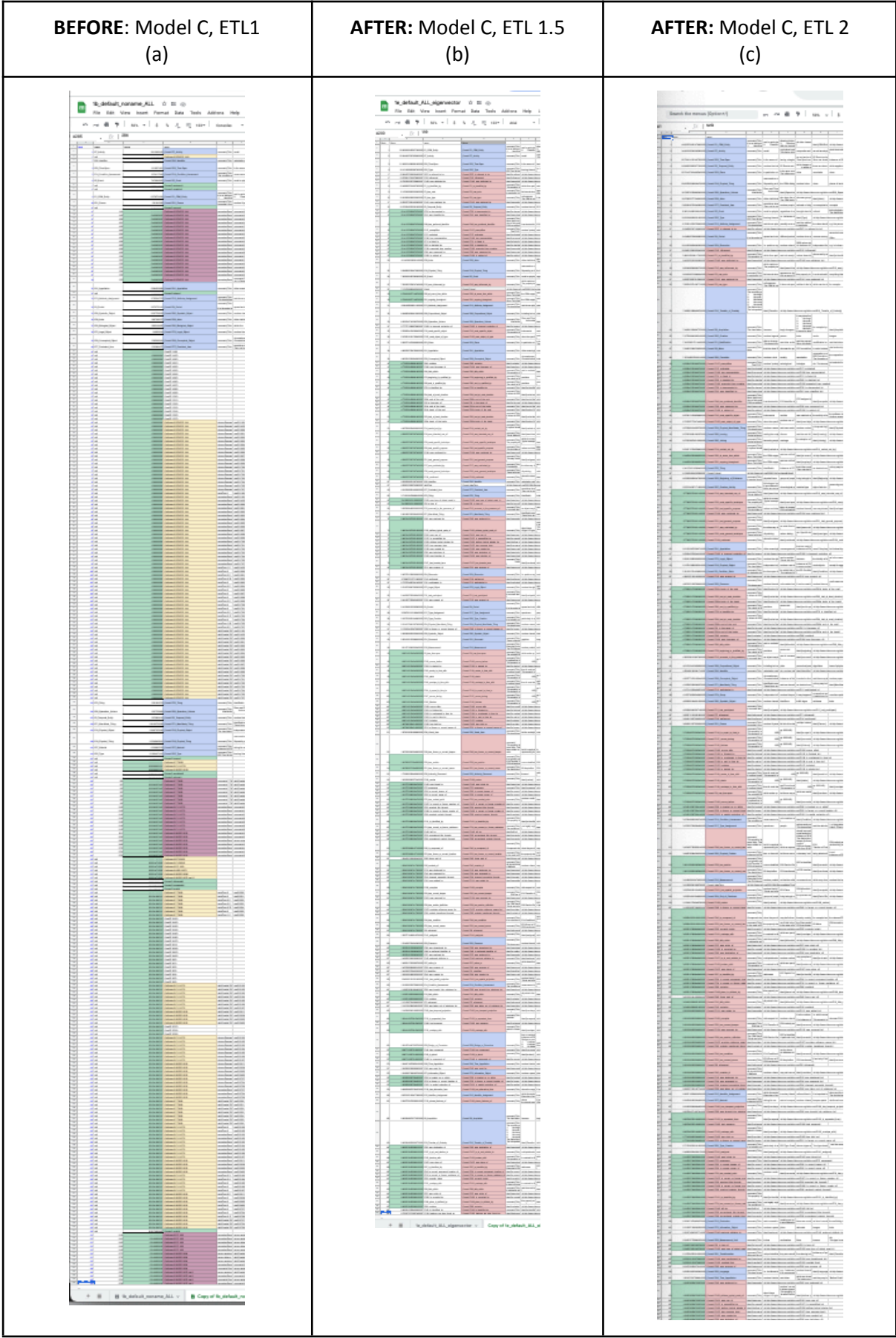


Figure H4.3 The colours highlight different node types (See Table 4.1.1 for key). Progress from (a) to (c) shows higher rankings for CRM classes (light blue) followed by TreatmentEvents (purple).

Appendix I

Phase 3 - Transformation of LPG to RDF

[I1. Introduction](#)

[I2. Cypher](#)

[I2.1 Defining Prefix Namespaces](#)

[I2.2 Defining Mappings](#)

[I2.3. Export Using Cypher](#)

[I3. RDF Validation](#)

I1. Introduction

The following Cypher configurations and scripts pertain to the transformation of labelled property graph (LPG) content in Neo4j to RDF presented in chapter 7 of the thesis.

Samples of transformed code available via the project repository: <https://github.com/ana-tam/conservation-graphs> .

I2. Cypher

Source: <https://neo4j.com/labs/neosemantics/4.0/export/>

I2.1 Defining Prefix Namespaces

Source:

<https://neo4j.com/labs/neosemantics/4.3/import/#custom-prefixes-for-namespaces>

```
CALL n10s.nsprefixes.add("crm", "http://www.cidoc-crm.org/cidoc-crm/7.1.1");
```

I2.2 Defining Mappings

Sources: <https://neo4j.com/labs/neosemantics/4.3/mapping/>

https://neo4j.com/labs/neosemantics/4.3/mapping/#_mappings_for_export

To check what mappings have been defined:

```
call n10s.mapping.list();
```

Example Cypher for adding to Mapping list:

The following CALL defines a mapping from the "entity" node property from (:Vocab) nodes and maps it to E57_Material from the CIDOC CRM:

```
CALL n10s.mapping.add("http://cidoc-crm.org/cidoc-crm/7.1.1/E57_Material",  
"entity");
```

Result:

schemaNs	schemaPrefix	schemaElement	elemName
1 "http://cidoc-crm.org/cidoc-crm/7.1.1/"	"crm711"	"E57_Material"	"entity"

12.3. Export Using Cypher

Example of using the 'Export Using Cypher'

```
:POST http://localhost:7474/rdf/neo4j/cypher { "cypher":"MATCH (n:Vocab)  
RETURN n", "mappedElemsOnly":true, "format":"RDF/XML"}
```

13. RDF Validation

Use: <https://www.w3.org/RDF/Validator/>















Appendix J: Phase 2 Results

Table J.1.1 Counts and Measures for the CIDOC CRM versions and Linked Conservation Data Project Datasets

	CIDOC CRM Versions				Linked Conservation Data Project			
	v.5.0.4	v.6.2.1	v.7.1.1 (ETL1)	v.7.1.1 (ETL2)	BOD	LOC	TNA	SUL
Order (node ct)*	346	374	387	387	2,451	1,707	2,119	2,219
Size (edge ct)	762	830	888	1285	5,481	3,611	4,611	5,753
Node:Edge ratio	1:2.20	1:2.22	1:2.29	1:3.32	1:2.24	1:2.12	1:2.18	1:2.59
Node:Edge Ratio (as quotient)	0.45	0.45	0.44	0.30	0.45	0.47	0.46	0.39
Edge Density*	0.0064	0.0059	0.0059	0.0086	0.0009	0.0012	0.0010	0.0012
Leaf Nodes	7	7	2	1	214	205	115	220
Isolated Nodes*	1	1	1	1	1	1	1	1
Leaf + Isolated*	8	8	3	2	215	206	116	221
Θ Ratio*	0.0231	0.0214	0.0078	0.0052	0.0878	0.1207	0.0547	0.0996
Average Clustering Coefficient	0.119	Infinity	infinity	infinity	0.0282	0.0250	infinity	infinity
Global Triangle Count	92	108	133	136	152	97	141	1047
Diameter - Undirected	8	8	8	8	12	10	9	10
Diameter - Directed Outgoing	7	7	7	9	8	6	6	6
Diameter - Directed Incoming	6	6	7	9	7	6	6	7
$k_{3,3}$ Count	0	0	768	823,104	293,400	1,753,632	1,096,704	368,424

*These measures include counting the single, isolated _GraphConfig node that the Neo4j Neosemantics system creates when importing RDF. Differences to calculated measures such as Edge Density and Theta Ratio are miniscule and negligible with differences detectable only by the fifth decimal place or after.

Table J.1.2 Motif Results for the CIDOC CRM versions and Linked Conservation Data Project Datasets

		CIDOC CRM Versions				Linked Conservation Data Project Datasets			
		v.5.0.4	v.6.2.1	v.7.1.1 (ETL1)	v.7.1.1 (ETL2)	BOD	LOC	TNA	SUL
	m3.1	14,126	16,322	18,942	44,208	152,144	90,396	158,304	156,452
	m3.2	576	672	1,146	2,784	840	414	882	6,600
	m4.1	66,646	78,904	89,782	306,164	870,456	618,124	882,298	1,070,632
	m4.2	320,094	408,150	552,336	2,067,360	7,166,172	3,021,888	9,546,456	6,627,762
	m4.3	6,804	7,824	13,530	49,820	6,816	2,432	5,186	45,358
	m4.4	3,448	4,040	5,136	20,080	78,936	60,136	43,920	83,672
	m4.5	348	384	1,164	4,436	40	0	0	7,476
	m4.6	0	0	0	0	0	0	0	2,496
	m5.1	432	432	1,188	7,284	0	0	0	7,872
	m5.2	2,128	2,632	4,912	27,468	782	0	0	39,114
	m5.3	5,406	5,626	16,962	98,740	200	0	0	47,540
	m5.4	88	96	860	4,800	0	0	0	5,144
	m5.5	88	96	860	4,800	0	0	0	5,144
	m5.6	0	0	0	0	0	0	0	1,464
















		CIDOC CRM Versions				Linked Conservation Data Project			
		v.5.0.4	v.6.2.1	v.7.1.1 (ETL1)	v.7.1.1 (ETL2)	BOD	LOC	TNA	SUL
	m5.7	204	256	516	3,968	260	0	0	9,732
	m5.8	0	0	0	0	0	0	0	624
	m5.9	31,488	37,404	55,132	306,206	69,178	40,520	89,568	415,020
	m5.10	32,096	35,272	49,318	218,468	32,594	8,966	16,968	259,832
	m5.11	4,912	5,160	18,448	98,728	976	288	920	38,256
	m5.12	191,568	225,256	404,652	2,377,980	168,832	22,960	62,244	510,204
	m5.13	1,256	1,496	2,586	14,434	1,782	312	896	27,328
	m5.14	0	0	0	0	0	0	0	8,784
	m5.15	33,122	42,962	56,532	353,216	2,046,642	1,227,318	1,055,444	1,485,738
	m5.16	2,664	3,588	4,980	37,260	856,284	675,168	607,596	732,216
	m5.17	368,372	459,154	537,624	2,651,202	9,736,142	5,973,822	8,031,906	10,053,802
	m5.18	9,657,192	13,572,336	22,223,784	127,708,896	497,568,360	144,846,528	817,006,008	439,049,520
	m5.19	1,118,206	1,441,620	1,743,384	9,791,278	22,357,694	13,259,724	28,022,138	29,767,890
	m5.20	7,740	9,230	10,720	59,420	23,400	3,440	7,670	88,260
	m5.21	0	0	0	0	0	0	0	0

Table J.1.3 Degree Centrality - CIDOC CRM Group (highest degrees)

		<i>node id</i>	<i>n.label</i>	<i>node Label/CRM Entity</i>	<i>degrees</i>
v.5.0.4	Natural	142	["was used for"]	["Resource", "Relationship"]	4
	Reverse	340	["CRM Entity"]	["Resource", "Class"]	40
	Undirected	342	["Actor"]	["Resource", "Class"]	41
v.6.2.1	Natural	156	"was used for"	["Resource", "Relationship"]	4
	Reverse	253	"Physical Thing"	["Resource", "Class"]	45
	Undirected	253	"Physical Thing"	["Resource", "Class"]	47
v.7.1.1 (ETL1)	Natural	79	["was used for"]	["Resource", "Relationship"]	4
	Reverse	323	["Physical Thing"]	["Resource", "Class"]	59
	Undirected	323	["Physical Thing"]	["Resource", "Class"]	60
v.7.1.1 (ETL2)	Natural	323	["Physical Thing"]	["Resource", "Class"]	32
	Reverse	323	["Physical Thing"]	["Resource", "Class"]	60
	Undirected	323	["Physical Thing"]	["Resource", "Class"]	92

Table J.1.4 Degree Centrality - Linked Conservation Data (LCD) Group (highest degrees)

		<i>node id</i>	<i>n.label</i>	<i>node Label/CRM Entity</i>	<i>degrees</i>
LCD- BOD	Natural	628	["thread"]	["Resource", "E55_Type", "E57_Material"]	49
	Reverse	1117	["corners"]	["Resource", "E55_Type"]	102
	Undirected	1117	["corners"]	["Resource", "E55_Type"]	102
LCD- LOC	Natural	1368	["Main conservation event (Library of Congress, 3995)"]	["Resource", "E11_Modification"]	23
	Reverse	1045	["right"]	["Resource", "E55_Type"]	74
	Undirected	1045	["right"]	["Resource", "E55_Type"]	74
LCD- TNA	Natural	1456	["Main conservation event (The National Archives, DL 30/603/2)"]	["Resource", "E11_Modification"]	28
	Reverse	2043	["repaired"]	["Resource", "E55_Type"]	122
	Undirected	2043	["repaired"]	["Resource", "E55_Type"]	122
LCD- SUL	Natural	700	["dataset"]	["Resource", "E89_Propositional_Object"]	52
	Reverse	405	["spine linings", "spine lining"]	["Resource", "E55_Type"]	106
	Undirected	405	["spine linings", "spine lining"]	["Resource", "E55_Type"]	106

Table J.1.5 Local Clustering Coefficient - CIDOC CRM Group (highest scores)

<i>dataset</i>	<i>node id</i>	<i>n.label</i>	<i>node Label/CRM Entity</i>	<i>score</i>
v.5.0.4	252	["contains"]	["Resource", "Relationship"]	Infinity
v.6.2.1	32	"had specific purpose"	["Resource", "Relationship"]	1
v.7.1.1 (ETL1)	119	["ends after or with the start of"]	["Resource", "Relationship"]	1
v.7.1.1 (ETL2)	235	Null (rdf-schema#label)	["Resource", "Relationship"]	Infinity

Table J.1.6 Local Clustering Coefficient - Linked Conservation Data (LCD) Group (highest scores)

<i>dataset</i>	<i>node id</i>	<i>n.label</i>	<i>node Label/CRM Entity</i>	<i>score</i>
LCD-BOD	857	["New endbandstail (Bodleian, Inc.d.f2.1494.2)"]	["Resource", "E22_Man-Made_Object"]	4.6667
LCD-LOC	487	["Broken (Library of Congress, 3968)"]	["Resource", "E3_Condition_State"]	5
LCD-TNA	1281	["Modification of The National Archives, ADM 37/5039"]	["Resource", "E11_Modification"]	3
LCD-SUL	1589	null*	["Resource", "E52_Time-Span"]	Infinity

*This node refers to an E52_Time-Span node where "P82a_begin_of_the_begin: "2010-01-01T00:00:00" and P82b_end_of_the_end: "2020-12-31T23:59:59" and encompasses the time from 1 Jan 2010 - 31 Dec 2020, inclusive.

	<i>projection</i>	<i>node id</i>	<i>n.label</i>	<i>node Label/CRM Entity</i>	<i>score</i>
v.5.0.4	default	340	["CRM Entity"]	["Resource", "Class"]	0.99387
	undirected	340	["CRM Entity"]	["Resource", "Class"]	0.32762
v.6.2.1	default	368	"CRM Entity"	["Resource", "Class"]	0.99487
	undirected	253	"Physical Thing"	["Resource", "Class"]	0.34582
v7.1.1 (ETL1)	default	384	["CRM Entity"]	["Resource", "Class"]	0.99501
	undirected	76	["Temporal Entity"]	["Resource", "Class"]	0.43248
v7.1.1 (ETL2)	default	76	["Temporal Entity"]	["Resource", "Class"]	0.42923
	undirected	323	["Physical Thing"]	["Resource", "Class"]	0.41475

Rounded scores to 5 decimal places.

	<i>projection</i>	<i>node id</i>	<i>n.label</i>	<i>node Label/CRM Entity</i>	<i>score</i>
LCD-BOD	default	1437	["ply"]	["Resource", "E58_Measurement_Unit"]	0.88725
	undirected	1117	["corners"]	["Resource", "E55_Type"]	0.36571
LCD-LOC	default	520	["damaged"]	["Resource", "E55_Type"]	0.80108
	undirected	1077	["conservation (process)"]	["Resource", "E55_Type"]	0.30754
LCD-TNA	default	1564	["damaged"]	["Resource", "E55_Type"]	0.53724
	undirected	651	["Kew (place)"]	["Resource", "E53_Place"]	0.29221
LCD-SUL	default	299	null*	["Resource", "E55_Type"]	0.68190
	undirected	1694	["board reattachment"]	["Resource", "E55_Type"]	0.34836

*This E55_Type node does not have a label property, only a uri property, however it can be inferred by 53 incoming [:P2_has_type] relationships from (:E3_Condition_State[label:"deterioration"]) nodes that its label should have been "deterioration".