CGs to FCA Including Peirce's Cuts

POLOVINA, Simon <http://orcid.org/0000-0003-2961-6207> and ANDREWS, Simon <http://orcid.org/0000-0003-2094-7456>

International Journal of Conceptual Structures and Smart Applications

January-June 2013, Vol. 1, No. 1

Table of Contents

The *International Journal of Conceptual Structures and Smart Applications* is currently listed or indexed in: Cabell's Directories; MediaFinder; The Standard Periodical Directory; Ulrich's Periodicals Directory

CGs to FCA Including Peirce's Cuts

Simon Polovina, Conceptual Structures Research Group, Sheffield Hallam University, Sheffield, UK

Simon Andrews, Conceptual Structures Research Group, Sheffield Hallam University, Sheffield, UK

ABSTRACT

Previous work has demonstrated a straightforward mapping from Conceptual Graphs (CGs) to Formal Concept Analysis (FCA), and the combined benefits these types of Conceptual Structures bring in capturing and reasoning about the semantics in system design. As in that work, a CGs Transaction Model (or `Transaction Graph') exemplar is used, but in the form of a richer Financial Trading (FT) case study that has its business rules visualised in Peirce's cuts. The FT case study highlights that cuts can meaningfully be included in the CGs to FCA mapping. Accordingly, the case study's CGs Transaction Graph with its cuts is translated into a form suitable for the CGtoFCA algorithm described in that previous work. The process is tested through the CG-FCA software that implements the CGtoFCA algorithm. The algorithm describes how a Conceptual Graph (CG), represented by triples of the form source-concept, relation, target-concept can be transformed into a set of binary relations of the form target-concept, source-concept⁀relation thus creating a formal context in FCA. Cuts though can now be included in the same formal, rigorous, reproducible and general way. The mapping develops the Transaction Graph into a Transaction Concept, capturing and unifying the features of Conceptual Structures that CGs and FCA collectively embody.

Keywords: *Conceptual Graphs (CGs), Conceptual Structures, Financial Trading (FT), Formal Concept Analysis (FCA), Transaction Graph*

INTRODUCTION

Previous work has demonstrated a straightforward mapping from Conceptual Graphs (CGs) to Formal Concept Analysis (FCA), and the combined benefits these types of Conceptual Structures bring in capturing and reasoning about the semantics in system design (Andrews & Polovina, 2011). However that mapping did not consider CGs' many more features, particularly its use of Peirce's Existential Graphs. Cited by Peirce as 'the logic of the future', this visualisation of logic and its visual approach to reasoning through novel techniques such as 'deiteration' and 'double negation' is claimed by Sowa as an enhancement of the traditional propositional and predicate logic of Peano, Russell, and Whitehead (Peirce & Sowa, 2010; Polovina, 2007). Sowa describes that Peirce indicated negation by drawing an oval enclosure, which he called a cut because it separated the

DOI: 10.4018/ijcssa.2013010105

sheet of assertion into a positive (outer) area and a negative (inner) area. The detail of this is described elsewhere (Peirce & Sowa, 2010; Polovina, 2007); pertinent to our interest is that cuts visualise contexts from which the nested negations enable inferencing to take place visually. Indeed Sowa refers to cuts as 'negative contexts'. The benefits of this visualisation have been demonstrated in capturing the semantics of business rules for enterprise system design (Launders, 2011a). Peirce's cuts thus provide a capability in CGs that the mapping could usefully be applied to, as we will now explore through a representative case study.

A FINANCIAL TRADING EXAMPLE

The case study is about a Financial Trading (FT) enterprise called TechRules Advisors (TRA Inc.), a fictitious asset management firm (© Said Tabet and Gerd Wagner). The firm buys and sells numbers of shares of securities and manages its clients' assets. Portfolio managers create and manage accounts. As in the previous work, a CGs Transaction Model (or 'Transaction Graph') illustration is used (Andrews & Polovina, 2011; Launders, 2011a; Polovina & Andrews, 2011). However unlike its simple case study scenario (namely a university's community objectives), the FT case study includes business rules visualised through Peirce's cuts. The detail of the case study is described as follows.

Description of the Case Study

The company (TRA Inc.) buys and sells shares of securities and manages its clients' assets. Portfolio managers create and manage accounts. A portfolio is owned by a legal entity. The portfolio is managed by a portfolio manager who works for an investment firm. A portfolio is described by a creation date and a value. It has a number of positions. Each position holds an asset and is described by a quantity and an acquisition date. The value of a portfolio is the total value of all the securities held in the portfolio.

There are three different categories of assets: real estate, cash, and securities. Real estate and cash are described by a name. Securities are described by: a security ID, a name and a price. There are three categories of securities: options, bonds, and stocks. Securities are issued by a legal entity that is called an issuer. The issuer can be: a company, a municipality, an agency, or a government.

There are many reasons that motivate issuers to issue securities. For example, the issuer might need to repay debts or raise capital (get some money to invest). Issuers and the securities they have issued can be positively or negatively affected by market events. Market events could be upgrades or downgrades by credit rating agencies. Some issuers are classified as `restricted' by portfolio owners and investment firms. Orders (for buying or selling assets) are placed in the interest of a portfolio. An order is placed by a trader or by a portfolio manager.

FT's Business Rules

The following are FT's business rules, which are captured with the aid of Peirce's cuts:

1. Securities issued by a "restricted" issuer must NOT be bought;
2. An asset must NOT be sold if it has been in the portfolio for less than 30 days;
3. The total asset value (TAV) is the sum of the market value of all positions;
4. The value of cash assets must be less than or equal to 10% of total asset value;
5. A portfolio is rated platinum, if TAV is greater than 1 Mio dollars. It is rated gold, if TAV is less than 1 Mio dollars and greater than 100.000 dollars. It is rated regular, if TAV is less than 100.000 dollars;
6. If there is a downgrade for a security held in a portfolio, the portfolio owner must be sent a "dispose recommendation". This advises the owner that they should sell the security;
7. An order placed in the interest of a portfolio must not refer to more than one asset held in a position of that portfolio;

8. The trade date of an order placed in the interest of a portfolio must be after the date that portfolio was created;
9. An order must not be placed both by the trader and by the portfolio manager.

The Transaction Graph for the FT Example

Using the *CharGer* CGs software (http://charger.sourceforge.net/), the FT example's Transaction Graph is given by Figure 1.

From this Transaction Graph for the FT case study we can observe that those concepts and relations capture the concepts and relations of financial trading enterprise of TRA Inc. Figure 1 reveals that the transaction comprises of two economic events, namely the concepts Order and Cash_Movement. The transaction is complete when both economic events balance i.e. an agreed exchange of resources just like, for example, exchanging cash to buy a book. (The bookstore gives up the book to get the cash; the buyer gives up the cash to get the book.) Additionally there are two related economic resources, Asset and Dollar, each having independent source and destination agents. The parties to the transaction are the Outside Agent (i.e. Issuer) and Inside Agent (i.e. the enterprise itself, Investment_Firm: TRA_Inc., delegated as the owner of the Asset by the Client).

Peirce logic visualises the inferences in CGs. Essentially these are the business rules in the case study. "For example Securities issued by a 'restricted' issuer must NOT be bought." Peirce's cuts capture that:

- IF a security is issued by a 'restricted' issuer THEN it must not be bought.

It also captures that:

- IF a security is bought from an issuer THEN the issuer is NOT restricted.

This is visualised through the cut and coreferent link between the concepts Issuer and Restricted_Issuer at the bottom of Figure 1. The business rules referring to the types of portfolio are also visualised by Peirce's cuts and coreferents. The whole transaction is visualised as a rule, showing that if the conditions as described in the transaction are satisfied then it is an FT_Transaction. That distinguishes it from other types of transactions e.g. buying a book. It defines the very nature of FT; the business that TRA Inc. is in.

Not all aspects of FT's descriptions above are captured in this Conceptual Graph (CG). This is partly due to clarity for the purposes of this discussion and partly that certain details may not be deemed pertinent at this level. However it is also to reflect that in the Transaction Agent Modelling (TrAM) process that this case study illustrates the CG would undergo a number of iterations as it is reviewed by an enterprise systems architect (Launders, 2011a). We thus see it visualised at one such stage. An earlier stage of this Transaction Graph, as well as a more step-by-step explanation of its visualisation can be found elsewhere (Launders, 2011a).

Representing the Cuts as Positive Contexts

Other than some early work there are to date no known tools CG or otherwise known to us that can process Peirce's cuts in CGs (Heaton, 1994). Essentially it is claimed that Peirce logic presents too high a burden of computational complexity for it to be implemented (Chein & Mugnier, 2008). Accordingly, cuts are presently only used for model visualisation in TrAM (Launders, 2011b). Recognising this issue, Figure 1 is translated into a form where the cuts are turned from negative into 'positive' contexts (Sowa, 2008). The result is shown by Figure 2.

Whilst losing the richness of Peirce logic operations such as deiteration and double negation (Polovina, 2007), Figure 2 shows the inher-

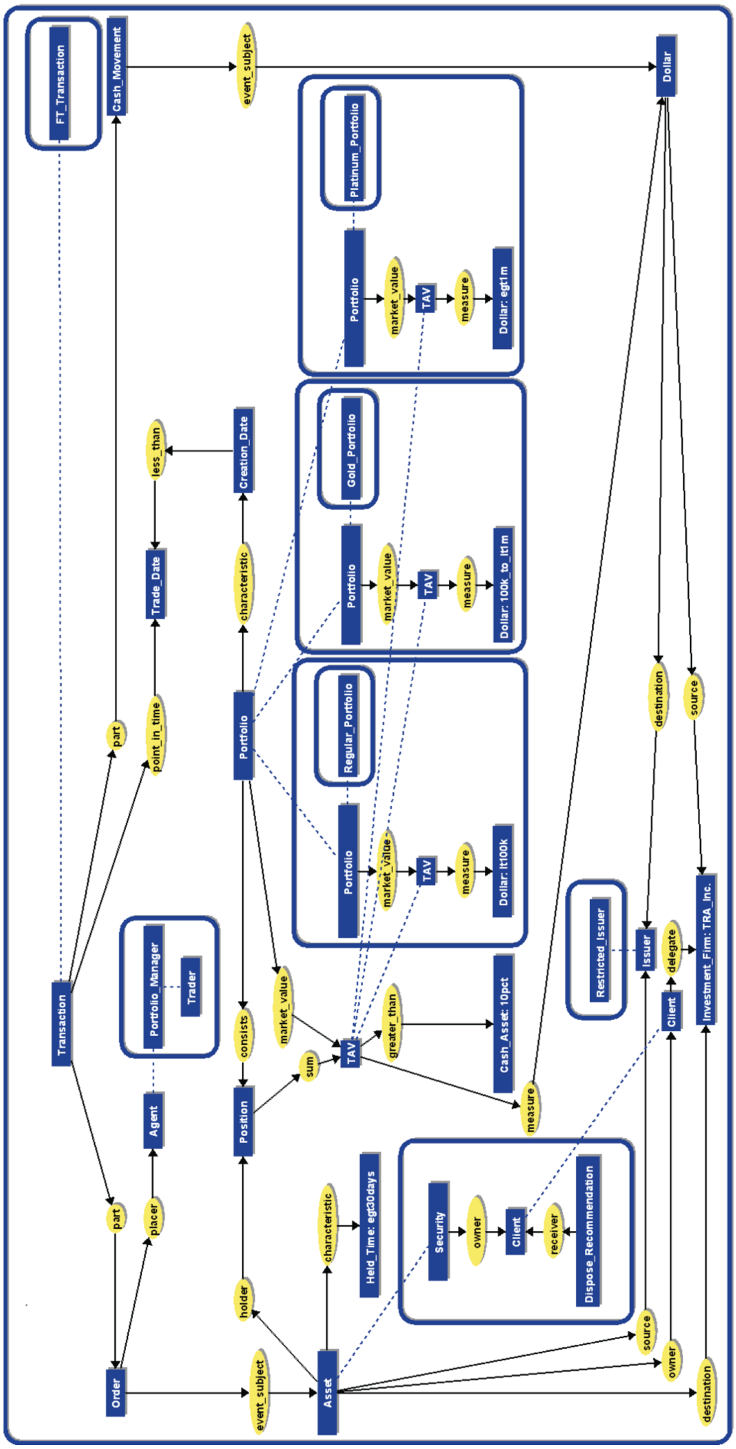


Figure 1. The transaction graph for TRA Inc.

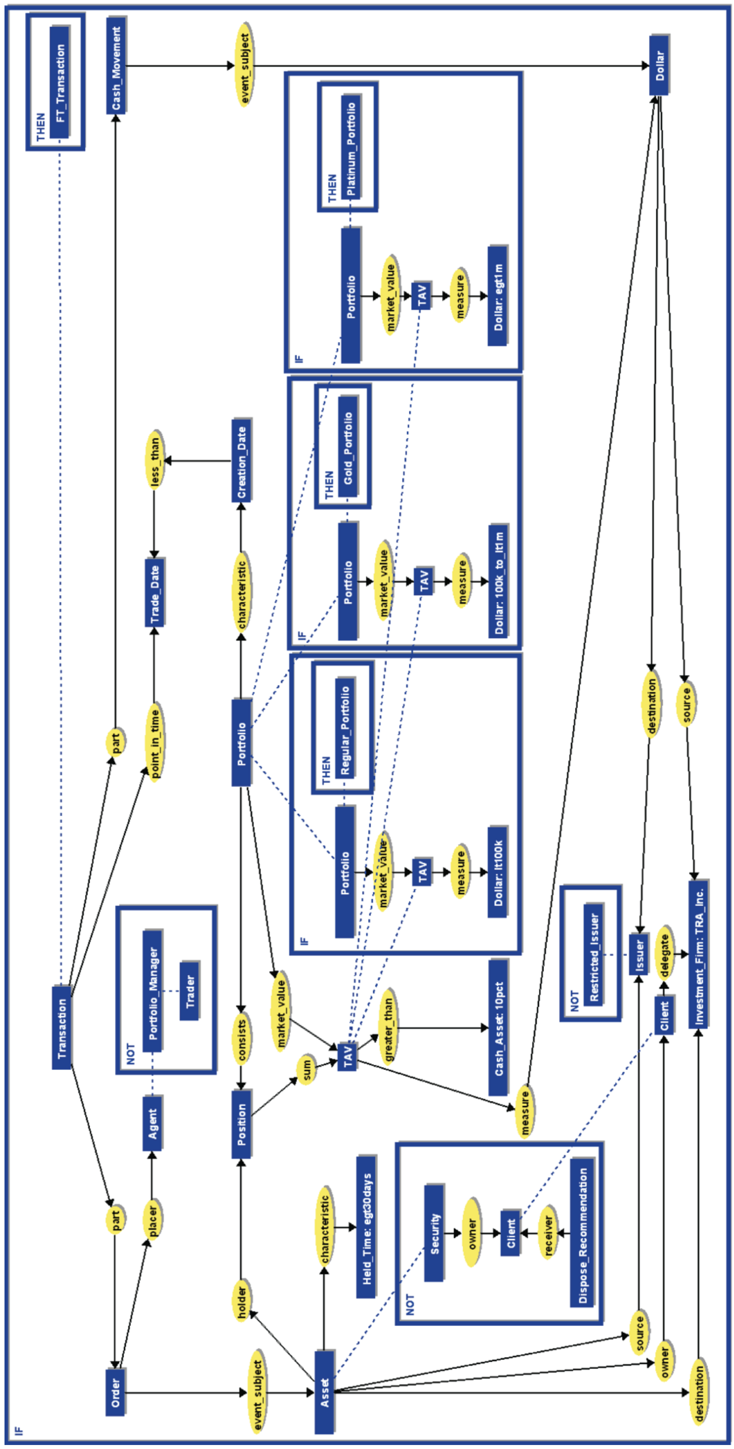


Figure 2. The transaction graph for TRA Inc., as positive contexts

ent content of the CG is preserved within these constraints. It thus preserves a useful mapping from cuts, and makes the CG implementable in CGs model automation tools such as CoGui (www.lirmm.fr/cogui).

Removing the Contexts

There is a further translation that needs to be made to the FT Transaction Graph for the CG to FCA mapping for to be applied to it as described in the previous work. That is to remove the contexts completely. The result would be simple CGs (Chein & Mugnier, 2008). The result is shown by Figure 3.

Whilst the translation from negative to positive contexts is straightforward, the translation to simple CGs requires an element of human judgement. There is thus no predictable translation, but the human intervention in the translation itself supports the iterative process mentioned earlier as far as TrAM is concerned. As such it can be seen as a benefit rather than a limitation. That stated, there still is a significant predictable element in the translation. In particular the IF_THEN for options, as illustrated in the type of portfolio (regular, gold or platinum) are predictably replaced by can_be relations as shown in Figure 3. Similarly for the IF_THEN for the Transaction Graph itself (FT_Transaction), as it effectively defines the transaction as an FT one that overall positive context can be replaced by a definition relation. Furthermore the division of duties between Trader and Portfolio Manager can be translated in a standard way by prefixing negation (i.e. ¬) to the coreferent concept. The same happens to the Restricted Issuer for its reasons. These can all be seen in Figure 3. We are now in a position to translate the CG to FCA.

CGtoFCA

The *CGtoFCA* algorithm (Andrews & Polovina, 2011) takes the CG source concept concatenated with its relation become formal attributes in FCA and the CG target concept becomes a formal object. Thus for example [Asset]$^\cap$(destination) becomes the formal attribute Asset destination and [Investment_Firm: TRA_Inc.] becomes the formal object Investment_Firm: TRA_Inc. A software implementation of *CGtoFCA*, *CGFCA* (http://sourceforge.net/projects/cgfca/) takes as its input the .cgif CG file format as produced by the CGs drawing tool *CharGer* that was referred to earlier for producing the CGs. The *CGFCA* program outputs a formal context in the well-known .cxt format.

Applying CGtoFCA to FT

Using *CGFCA* and displaying the result using the Concept Explorer software (http://sourceforge.net/projects/conexp/), Figure 4 shows the FCA lattice for TRA Inc.'s TM in CG as shown by Figure 3, thus ultimately Figure 1 with Peirce's cuts.

From the lattice for the FT case study we can observe the following:

1. The identification of FT_Transaction object as the overarching superconcept at the top of the lattice with the flow down through the lattice in line with the direction of relational arrows (the arcs that link the concepts by their relations) in the FT Transaction Graph. There is no object identified with the bottommost concept, which we will investigate further;
2. The flow down from the Transaction object aligns with the FT Transaction Graph. For example Transaction flows along the Transaction_part attribute to the objects Message, Cash_Movement and Order;
3. The TAV concept can be identified in the lattice as one of the central objects with its extents in line with the FT Transaction Graph, namely Platinum_Portfolio market_value, Gold_Portfolio market_value, Regular_Portfolio market value, Position sum, and Portfolio market_value;
4. There is no explicit relationship that points between 'inside' agent (Issuer or Restricted_Issuer) in this transaction to Investment_Firm: TRA_Inc., which again we will investigate further.

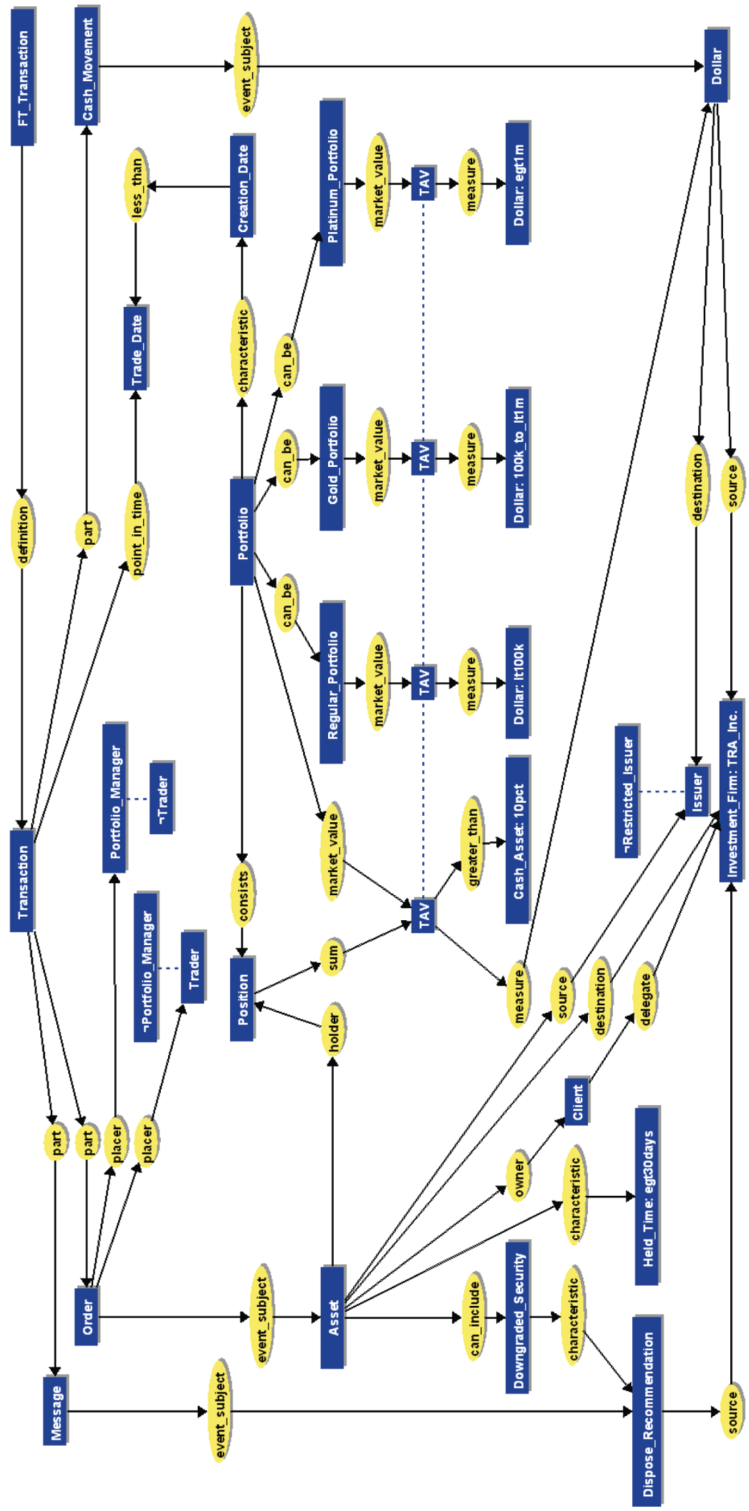
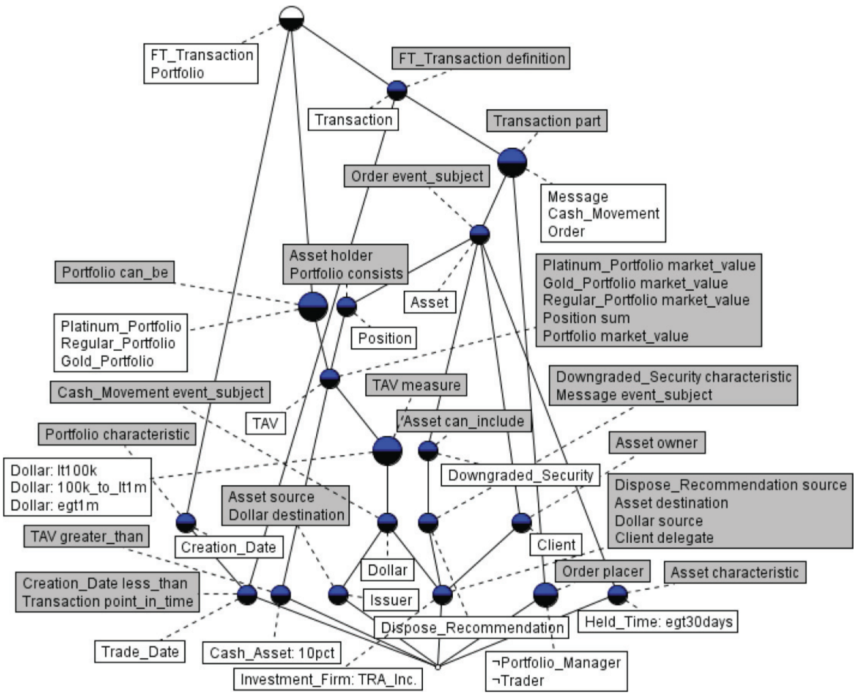


Figure 3. The transaction graph for TRA Inc., replacing the positive contexts with simple CGs

Figure 4. Concept Lattice for TRA Inc. from simple CGs



CONNECTING FT_TRANSACTION TO TRA_INC.

From the lattice for the FT case study we can observe that as in the previous work (Andrews & Polovina, 2011), the lattice has demonstrated that there ought to be a flow from an uppermost Transaction object and culminating in TRA_Inc. as the bottommost object. Interestingly, the FT_Transaction definition target attribute is the Transaction object, thus as we are describing an FT_Transaction that is correctly shown as the topmost object with Transaction below it through the FT_Transaction definition attribute. Portfolio is however also topmost, when it part of the transaction. It suggests that there is a missing downward CG relationship from Transaction to Portfolio. The bottommost concept lacks its own object, namely Investment Firm: TRA_Inc. As it is this enterprise's transaction, the intent of this object should be all the attributes in the lattice. Going the other way, the extent of FT_Transaction definition are all the objects in the lattice including TRA_Inc. It is what defines the transaction. Therefore just like the simple university case study exemplar of the previous work we need to refine the FT Transaction Graph so that the CGs arcs (the arrows connecting CG concepts through relations) cascade down from FT_Transaction to TRA_Inc. The result is shown by Figure 5.

To make the arrows point in the right direction, use was made of '..._of' relations, which read the source concept is relation_of target concept (e.g. Portfolio_Manager is manager_of Portfolio). In a conventional reading of a CG, the reading is 'the relation of a concept is a concept' (Polovina, 2007) e.g. "The manager of a portfolio is a portfolio manager". The direction of the arc would thus point in the opposite direction and cause a problem to our intended top to bottom flow. Interestingly, other work has described the reading of CGs in the alternative way, adjusting the relation names and the direc-

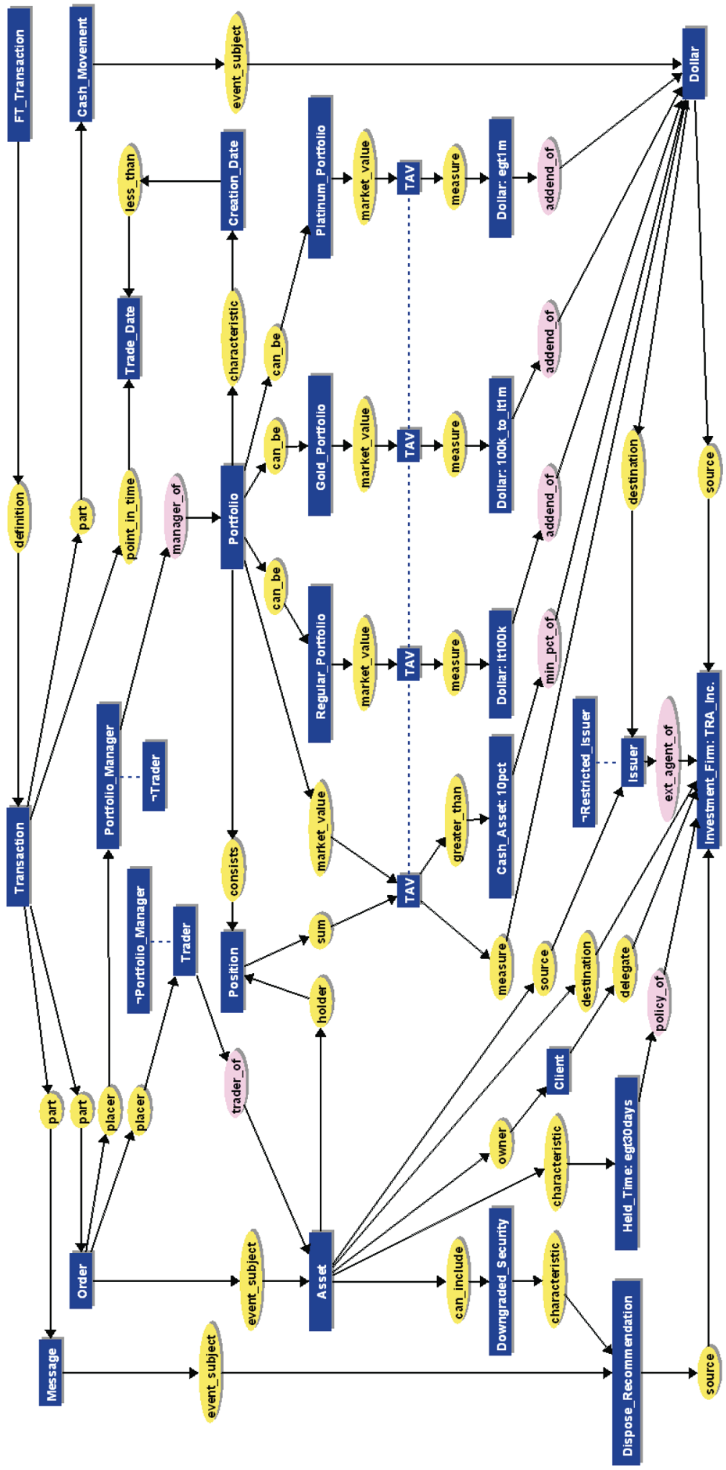


Figure 5. The transaction graph for TRA Inc., connecting FT_Transaction to TRA_Inc.

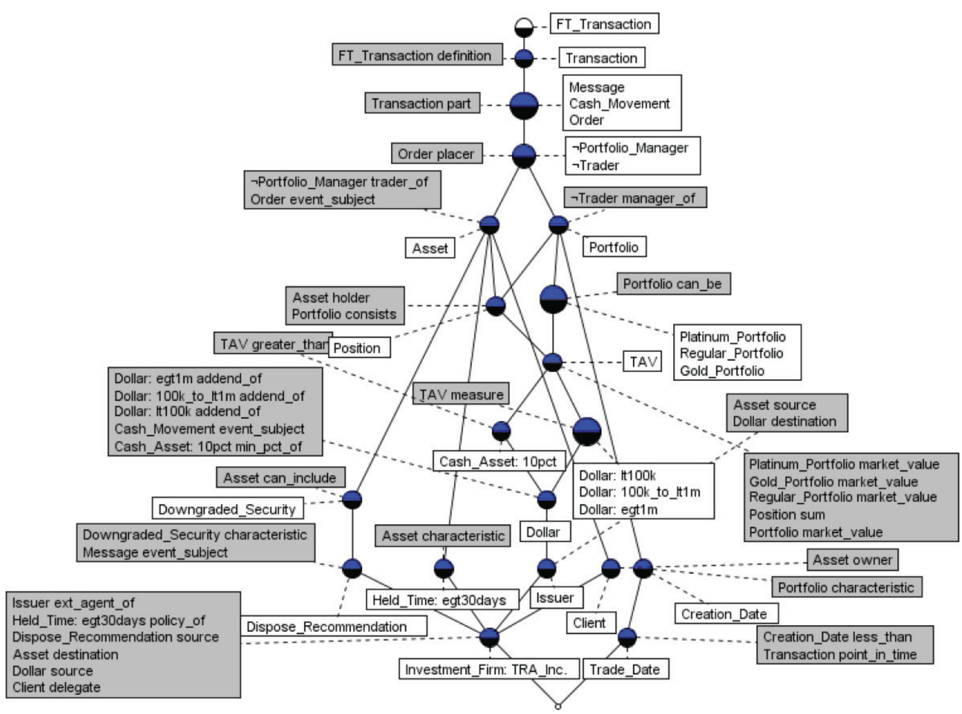
tion of the arcs accordingly (Chein & Mugnier, 2008). This also would accord with work that translates CG to RDF (Baget, Croitoru, Gutierrez, Leclère, & Mugnier, 2010).

Semantically, the direction of the arcs from top to bottom is the overarching theme in that for FT_Transaction the CG concepts are pointed to TRA_Inc. In FCA terms the intent of TRA_Inc. are all the attributes of the transaction that describe the FT_Transaction and why those attributes are core to it. The extent of the FT_Transaction is the constituent objects that together make up that transaction. Thus it is simply down to the naming of the relations in a way that supports the overall downward flow not another way around. The arcs thereby define these names; hence it is valid to use the '..._of' relations. Without FCA this would not have been highlighted. The lattice is depicted by Figure 6.

Connecting FT_Transaction to TRA_Inc., Corrected

It is obvious from the lattice that Investment_Firm: TRA_Inc. is not at the bottommost concept. Why? Immediately this points to fact that even within the modified Transaction Graph of Figure 5 the arcs are still not all pointing collectively in the downward direction. From the lattice it is easy to identify that the culprit is Trade_Date as this is not in TRA_Inc.'s extent. As we have seen, the direction of the arcs in CGs is rather informal, relying on the names of the relations to give a sense of this direction. Indeed it is easily possible to mix up the direction even in introductory CGs as one paper's typos in its CGs will reveal on a careful examination (Polovina, 2007). FCA eradicates this issue; the focus is on the direction and the concepts, and informs the names of the relations. The result is shown by Figure 7 and the lattice is shown by Figure 8.

Figure 6. Connecting FT_Transaction to TRA_Inc.



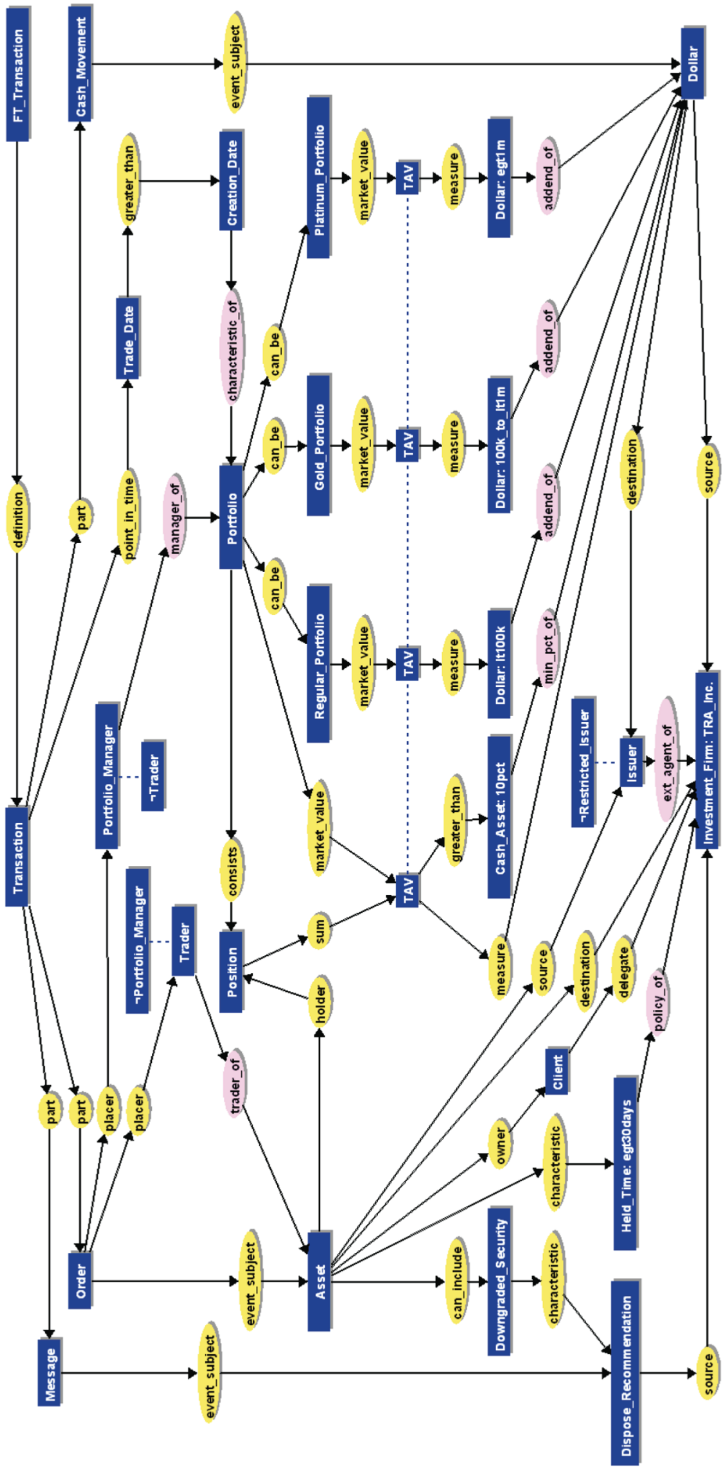
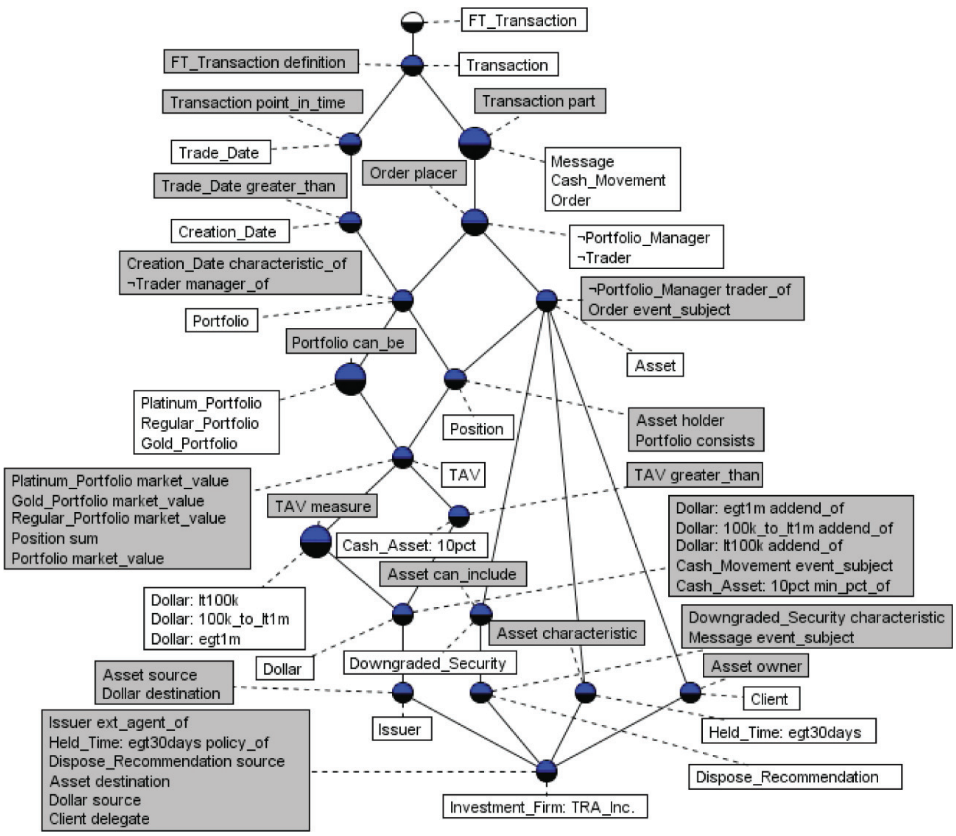


Figure 7. The transaction graph for TRA_Inc., correctly connecting FT_Transaction to TRA_Inc.

Figure 8. Concept Lattice for TRA_Inc. from connected CGs



Two changes were made. The first one, changing the 'less_than' relation name to 'greater_than' was relatively straightforward (and had the interesting side-effect of pondering over if one should use less_than or greater_than; FCA immediately took care of that). The other was harder in that the characteristic relation looked as if it had to be 'fabricated' just to fit the model; that relation demands that the arcs point in its original direction especially as Creation_Date is a characteristic of Portfolio that is being described! However, following the principle of using `..._to' relations and as a reminder that the simple naming of relations was secondary to the concepts and the directions of the arcs, the relation was renamed 'characteristic_to' accordingly. The semantics were thus preserved.

TOWARDS A TRANSACTION CONCEPT

The case study has demonstrated the interoperability of CGs with FCA, highlighting how the latter can add rigour to the former. Whereas we previously referred to the Transaction Model, or Transaction Graph, based on its CGs credentials we now have a Transaction Conceptual Structure. That epitomises that CGs and FCA have enhanced the Transaction Graph beyond the individual merits of CGs or FCA. CGs provide a convenient conceptual modelling environment where even inferencing can be visualised as part of the same model through Peirce's cuts. Duly transformed into simple CGs there was a valuable mapping to FCA, which immediately added value to the model as described. Given that CGs and FCA define the element 'Concept' in their respective names, it is appropriate to refer to the Transaction Graph (CGs) and, as we might define, the Transaction Lattice (FCA) in a new term. That is the Transaction Concept. It epitomises the hitherto hidden harmony between CGs and FCA, thanks to *CGtoFCA*.

CONCLUDING REMARKS

Whilst we have referred how CGs and FCA can be combined to enhance the Transaction Graph into the Transaction Concept, thus having potential beneficial applications in business computing that we have yet to articulate fully, the *CGtoFCA* algorithm is not restricted to it. The mapping is principled in its own right. Moreover *CGtoFCA* is essentially an algorithm that can apply to triple structures in general. That not only allows CGs and FCA to interoperate and enhance each other, but to bring FCA into the remit of wider technologies of this nature such as Linked Data. Added to the business scenario that has been the theme of the present discussion, thereby finding mainstream applications, it brings FCA and CGs as Conceptual Structures into a much wider remit that smart applications can build upon.

REFERENCES

Andrews, S., & Polovina, S. (2011). A mapping from conceptual graphs to formal concept analysis. In S. Andrews, S. Polovina, R. Hill & B. Akhgar (Eds.), *Conceptual Structures for Discovering Knowledge, the 19th International Conference on Conceptual Structures (ICCS 2011),* Derby, UK (Lecture Notes in Computer Science, Vol. 6828 (Subseries: Lecture Notes in Artificial Intelligence), pp. 63-76). Heidelberg, Germany: Springer.

Baget, J., Croitoru, M., Gutierrez, A., Leclère, M., & Mugnier, M. (2010). Translations between RDF(S) and conceptual graphs. In M. Croitoru, S. Ferré & D. Lukose (Eds.), *Proceedings of the 18th International Conference on Conceptual Structures: From Information to Intelligence (ICCS'10)* (Lecture Notes in Artificial Intelligence ed., pp. 28-41). Springer.

Chein, M., & Mugnier, M. (2008). Graph-based knowledge representation: Computational foundations of conceptual graphs (Advanced Information and Knowledge Processing Series ed.). Heidelberg, Germany: Springer.

Heaton, J. E. (1994). *Goal driven theorem proving using conceptual graphs and peirce logic.* Loughborough University). Doctoral Thesis, Loughborough University.

Launders, I. (2011a). Evaluating the transaction graph through a financial trading case study. In S. Andrews, S. Polovina, R. Hill & B. Akhgar (Eds.), *Conceptual Structures for Discovering Knowledge, the 19th International Conference on Conceptual Structures (ICCS 2011),* Derby, UK (Lecture Notes in Computer Science, Vol. 6828 (Subseries: Lecture Notes in Artificial Intelligence), pp. 145-158). Heidelberg, Germany: Springer.

Launders, I. (2011b). *The transaction graph: Requirements capture in semantic enterprise architectures*. LAP Lambert Academic Publishing.

Peirce, C. S., & Sowa, J. F. (2010). *Existential graphs: MS 514 by Charles Sanders Peirce, with commentary by John F. Sowa.* Retrieved February 7, 2013, from http://www.jfsowa.com/peirce/ms514.htm

Polovina, S. (2007). An introduction to conceptual graphs'. In U. Priss, S. Polovina & R. Hill (Eds.), *Proceedings of the 15th International Conference on Conceptual Structures (ICCS 2007): Conceptual Structures: Knowledge Architectures for Smart Applications,* Sheffield, UK (Lecture Notes in Artificial Intelligence (LNAI 4604) ed., pp. 1-15). Heidelberg, Germany: Springer.

Polovina, S. (2007). An introduction to conceptual graphs. In U. Priss, S. Polovina & R. Hill (Eds.), *Proceedings of the 15th International Conference on Conceptual Structures (ICCS 2007): Conceptual Structures: Knowledge Architectures for Smart Applications,* Sheffield, UK (Lecture Notes in Artificial Intelligence (LNAI 4604) ed., pp. 1-15). Berlin/Heidelberg, Germany: Springer.

Polovina, S., & Andrews, S. (2011). A transaction-oriented architecture for structuring unstructured information in enterprise applications. In V. Sugumaran (Ed.), *Intelligent, adaptive and reasoning technologies: New developments and applications* (pp. 285–299). Hershey, PA: IGI-Global Publishing. doi:10.4018/978-1-60960-595-7.ch016.

Sowa, J. F. (2008). Conceptual graphs. In F. van Harmelen, V. Lifschitz, & B. Porter (Eds.), *Handbook of knowledge representation, foundations of artificial intelligence* (Vol. 3, pp. 213–237). Amsterdam, Netherlands: Elsevier. doi:10.1016/S1574-6526(07)03005-2.

CALL FOR ARTICLES

International Journal of Conceptual Structures and Smart Applications

An official publication of the Information Resources Management Association

ISSN 2166-7292
eISSN 2166-7306
Published semi-annually

MISSION:
The mission of the **International Journal of Conceptual Structures and Smart Applications (IJCSSA)** is to harmonize the creativity of humans with the productivity of computers. CS recognizes that organizations work with concepts. The journal advances the theory and practice in connecting the user's conceptual approach to problem solving with the formal structures that computer applications need to bring their productivity to bear. The goal of the journal is to bring together the world's best minds in information technology, arts, humanities, and social science to explore novel ways that information technologies can be used to leverage tangible business and social benefits. The journal thus integrates the creativity of individuals and organizations with the productivity of computers for a meaningful digital future.

COVERAGE/MAJOR TOPICS:

- Conceptual Structures: Theory, Applications, and Practices
    - Conceptual graphs
    - Formal concept analysis
    - ISO common logic
- Knowledge Architectures
    - Enterprise knowledge systems
    - Mobile, ubiquitous, or embedded systems
    - Metaphoric, cultural or semiotic considerations
    - Multi-agent systems
    - Ontologies and the their effective implementation
    - Post-syntactic, semantic, or pragmatic interoperability
    - Requirements engineering
    - Security and trust
    - Standards and recommendations
    - Transaction-oriented architectures
- Smart Applications: Science, Technology and Systems
    - e-Science
    - e-Medicine
    - Forensic computing
    - Grid computing
    - Natural language systems
    - Robotics
    - Semantic web
    - Pragmatic web
    - Topic maps
    - Web 2.0
- Smart Applications: Enterprise, Education, Society and Government
    - Augmenting collective intelligence
    - Business intelligence
    - Discovering misuse and fraud
    - e-Learning, smart VLEs (Virtual Learning Environments)
    - e-Social science
    - Folksonomies
    - Government accountability and e-democracy
    - Intellectual property management
    - Legal analysis
    - Healthcare management
    - Knowledge discovery
    - Knowledge management
    - Managing ambiguity, nonsense or contradictions
    - Teaching and learning of logic
    - Supporting social action
- Interaction Design, Visualization, Creative Industries
    - Cultural, localization and internationalization
    - Entertainment
    - Linguistics
    - Peirce's extential graphs logic
    - Portal development
    - Semiotics, signs and meaning making
    - Supporting sensory disabilities, dyslexia or other differently abled
    - Supporting alternative (e.g., lateral) thinking
    - Thinking with diagrams
    - User modeling
    - User personalization
    - Visual analytics

All submissions should be e-mailed to:
Simon Polovina and Simon Andrews
Editors-in-Chief
ijcssa@shu.ac.uk

Ideas for Special Theme Issues may be submitted to the Editor-in-Chief.

Please recommend this publication to your librarian. For a convenient easy-to-use library recommendation for. please visit: http://www.igi-global.com/IJCSSA