A Mapping from Conceptual Graphs to Formal Concept Analysis

Simon Andrews and Simon Polovina

Conceptual Structures Research Group
Communication and Computing Research Centre
Faculty of Arts, Computing, Engineering and Sciences
Sheffield Hallam University, Sheffield, UK
s.andrews@shu.ac.uk, s.polovina@shu.ac.uk

Abstract. A straightforward mapping from Conceptual Graphs (CGs) to Formal Concept Analysis (FCA) is presented. It is shown that the benefits of FCA can be added to those of CGs, in, for example, formally reasoning about a system design. In the mapping, a formal attribute in FCA is formed by combining a CG source concept with its relation. The corresponding formal object in FCA is the corresponding CG target concept. It is described how a 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 $\frown$ relation) creating a formal context in FCA. An algorithm for the transformation is presented and for which there is a software implementation. The approach is compared to that of Wille. An example is given of a simple University Transaction Model (TM) scenario that demonstrates how FCA can be applied to CGs, combining the power of each in an integrated and intuitive way.

1 Introduction

Conceptual Graphs (CGs) and Formal Concept Analysis (FCA) are related disciplines in that they both aim to help us understand the world and systems within it by structuring, formalising and depicting their semantics. CGs and FCA have communities of researchers and practitioners that, although independent, share common goals of knowledge discovery and elucidating the meaning of human systems and interactions. Indeed, each community has come to be conversant in the other's discipline, not least through the shared dissemination of their work at the annual International Conference on Conceptual Structures. As a result, CGs and FCA have been linked in several works [1–5]. Although, in these cases, the powers of both conceptual disciplines have been brought to bear on a particular domain, a direct mapping from CGs to FCA is not being attempted. Such a mapping was, however, proposed by Wille to obtain a unified mathematical theory of Elementary Logic [7]. Referring to Wille's translation as our comparison along the way, we present a straightforward mapping from CGs to FCA.

2 Motivation

At Sheffield Hallam University we have established a Conceptual Structures Research Group. One of us (Polovina) has had a long-standing interest in CGs whilst the other (Andrews) has developed an active interest in FCA. Our group has applied CGs through the Transaction Model (TM) [6]. For FCA there is our contribution to the CUBIST project (www.cubist-project.eu). One of our core interests is how we can at a practical level bring CGs and FCA together. For this purpose we took a simple TM example, namely 'P-H University' to illustrate the discussion that we now present [6].

3 Conceptual Graphs (CGs)

A Conceptual Graph (CG) is a bipartite graph as shown by the general form in Figure 1 and may be read as: "The relation of a Concept_1 is a Concept_2".

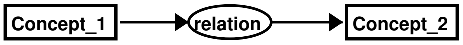

Fig. 1. General Form of a CG

3.1 Concepts and Relations

The direction of the arcs between a concept and a relation assist the direction of the reading. It distinguishes the source concept from the target concept. Here therefore Concept_1 is the source concept and Concept_2 the target concept. Alternatively to this 'display' form (produced using the *CharGer* CGs software, charger.sourceforge.net/), a CG may be written in the following 'linear' text-based form:

```
[Concept_1] -> (relation) -> [Concept_2]
```

Consider the following example:

```
[Transaction] -> (part) -> [Cash_Payment]
```

This example will form a part of an illustrative case study involving a fictitious university P-H University. The example graph reads as "The part of a transaction is a cash payment". This may create readings that may sound long-winded or ungrammatical, but is a useful mnemonic aid in constructing and interpreting any CG. It is easier in this case to state "A cash payment is part of a transaction". Furthermore, a concept has a referent that refers to the particular instance, or individual, in that concept. For example consider the concept:

`[Educational_Institution: P-H_University]`

This reads as “The educational institution known as P-H University”, where P-H University is the referent of the type label Educational Institution in this concept. A concept that appears without an explicit referent has a ‘generic’ referent, thereby referring to an individual that is implicit. Thus for example the concept `[Transaction]` simply means “A transaction” or “There is a transaction”. It could be that the transaction has a distinct reference number e.g. `#tx1` as its referent, resulting in `[Transaction: #tx1]`, if it conforms to that particular transaction.

4 Mapping CGs to FCA

4.1 A CG to FCA Algorithm

The following algorithm takes a Conceptual Graph (CG) in the form of a set of $(SourceConcept, Relation, TargetConcept)$ triples and creates a corresponding set of binary relations of the form $(TargetConcept, SourceConcept \frown Relation)$ and thereby makes an FCA formal context whose attributes are the $SourceConcept \frown Relation$ parts and whose objects are the corresponding $TargetConcept$ parts of each binary relation.

The $SourceConcept$, $Relation$ and $TargetConcept$ parts of a CG triple, t, are denoted by $t.source$, $t.relation$ and $t.target$, respectively. I is the set of $(Object, Attribute)$ pairs making the FCA formal context, where

$$t \in T \Rightarrow (t.target, t.source \frown t.relation) \in I$$

The mapping is transitive in that the target CG concept of one relation can become the source CG concept of another relation. This transitivity gives rise to the inference of implicit mappings:

$$\forall\, t_1, t_2 \in T \bullet t_1.target = t_2.source \Rightarrow (t_2.target, t_1.source \frown t_1.relation) \in I$$

```
begin
    foreach $t \in T$ do
        $FormBinaries(t.target, t.target)$;
end
```

Fig. 2. *CGtoFCA*()

The main algorithm, *CGtoFCA* (see Figure 2) takes each triple, t, in a set of CG triples, T, and forms all binary relations associated with the target concept of the triple by calling the procedure *FormBinaries*, given in Figure 3. *FormBinaries* takes two CG concepts as arguments: *FixedTarget* and

```
begin
    foreach $t \in T$ do
        if $t.target = MovingTarget$ then
            $Attribute \leftarrow t.source \frown t.relation$;
            $Object \leftarrow FixedTarget$;
            $I \leftarrow I \cup \{(Object, Attribute)\}$;
            $FormBinaries(FixedTarget, t.source)$;
end
```

Fig. 3. *FormBinaries*(*FixedTarget*, *MovingTarget*)

MovingTarget. *FixedTarget* is used as a stem target concept so that inferred, transitive, binaries can be formed by backward chaining. This is achieved by setting *MovingTarget* to a source concept that is searched for as a target concept in an inferred relation.

The algorithm works by initially setting the target concept of each triple as both *MovingTarget* and *FixedTarget* and then calling *FormBinaries* to iterate through all triples, forming a corresponding FCA (*Object*, *Attribute*) pair each time *MovingTarget* matches the target concept of a triple. The transitive binaries are formed by recursively calling *FormBinaries*, setting *MovingTarget* to the current source concept and leaving *FixedTarget* unchanged.

To demonstrate the algorithm (and provide a partial proof) it is applied to the simple CG example shown in Figure 4. The corresponding set of triples is given in Table 1.

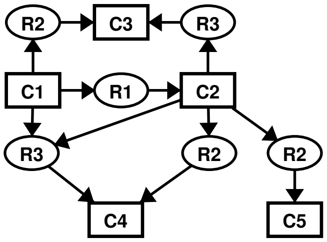

Fig. 4. A Simple Conceptual Graph

Rather than go through all of the triples in the main algorithm, for brevity it is sufficient to show the features of the algorithm by listing a sequence of steps from the call to *FormBinaries* from *CGtoFCA* when $t \leftarrow (C2, R2, C4)$.

$FormBinaries(C4, C4)$ //the fixed target is C4, moving target is C4
    $t \leftarrow (C1, R1, C2)$

```
    $C2 \neq C4$
$t \leftarrow (C1, R2, C3)$
    $C3 \neq C4$
$t \leftarrow (C1, R3, C4)$
    $C4 = C4$ //the target of the triple matches the moving target
        $I \leftarrow I \cup \{(C4, C1 \frown R3)\}$ //add FCA (Object,Attribute)
        $FormBinaries(C4, C1)$ //the fixed target is C4, moving target is C1
            $t \leftarrow (C1, R1, C2)$
                $C2 \neq C1$
            $t \leftarrow (C1, R2, C3)$
                $C3 \neq C1$
            $t \leftarrow (C1, R3, C4)$
                $C4 \neq C1$
            $t \leftarrow (C2, R2, C4)$
                $C4 \neq C1$
            $t \leftarrow (C2, R2, C5)$
                $C5 \neq C1$
            $t \leftarrow (C2, R3, C3)$
                $C3 \neq C1$
            $t \leftarrow (C2, R3, C4)$
                $C4 \neq C1$
            //no targets of triples match C1
//so back up one level to moving target being C4
$t \leftarrow (C2, R2, C4)$
    $C4 = C4$ //the target of the triple matches the moving target
        $I \leftarrow I \cup \{(C4, C2 \frown R2)\}$
        $FormBinaries(C4, C2)$ //moving target is now C2
            $t \leftarrow (C1, R1, C2)$
                $C2 = C2$ //the target of the triple matches the moving target
                    $I \leftarrow I \cup \{(C4, C1 \frown R1)\}$
                    //thus adding an implied FCA binary by backward chaining
                    $FormBinaries(C4, C1)$ //already completed above
```

source		target
concept	relation	concept
C1	R1	C2
C1	R2	C3
C1	R3	C4
C2	R2	C4
C2	R2	C5
C2	R3	C3
C2	R3	C4

Table 1. Concept-Relation-Concept Triples from the Simple Conceptual Graph

Simple CG	C1-R1	C1-R2	C1-R3	C2-R2	C2-R3
C1					
C2	×				
C3	×	×			×
C4	×		×	×	×
C5	×			×	

Fig. 5. The Corresponding Formal Context of the Simple Conceptual Graph

A small deficiency of the algorithm is that binaries associated with a target concept may be generated multiple times. Each call to *FormBinaries* from *CGtoFCA* generates all binaries associated with *FixedTarget*, so multiple instances of a target concept means this will happen multiple times. An implementation of the algorithm may need to take this into account (by removing repeated binaries after generation, for example). However, in practical terms, multiple instances of a binary only means that a 'cross' is entered into a table cell multiple times.

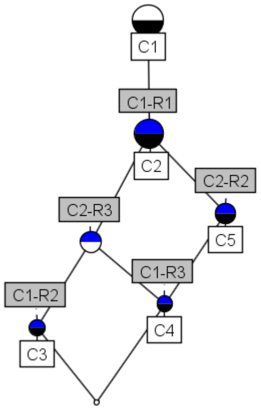

Fig. 6. The Corresponding Concept Lattice for the Simple Conceptual Graph

The algorithm will produce an infinite recursion, generating repeated binaries, if a cycle exists in the CG. The simplest example of this is a CG concept that is both the source and target of a relation, although larger cycles are easily possible. A sensible approach to implementing the algorithm could include the capture of such cycles (by noting that the program has 'been here before' in the

CG) and reporting them to the user. It may then be useful for the author of the CG to consider how desirable the cycles are in their design.

The resulting binaries generated from the simple CG triples are shown as a cross-table (FCA context) in Figure 5 and hence as an FCA concept lattice in Figure 6. Using well known notions from FCA, a number of observations can be made:

- $C1$ is not a target CG concept of any relation.
- $C1 - R1$ is the relation with the most results (four): $C2$,$C5$,$C4$ and $C3$.
- $C4$ results from the most relations (four): $C1 - R1$, $C2 - R3$, $C2 - R2$ and $C1 - R3$.

5 A Simple University Scenario Example

We now specify the example. It is a simple case study that is discussed extensively elsewhere to demonstrate the Transaction Model (TM) [6]. Essentially the case study is about P-H University, which is a fictional higher education institution. The University is not primarily a profit-making institution; rather it has to remain financially sound whilst recognising its community objectives. Key to these objectives are its research activities. P-H University thereby needs to explicate the relationship between its research and its community objectives. 40% of its staff are emerging researchers that receive time off for research instead of revenue-generating activities (e.g. teaching), added to which there is a diversion of revenue to give the established researchers' time to support their emerging colleagues. In financial terms these items represent a 'cost' to the University for which there is no corresponding revenue. Yet the financial cost saving from not investing in the emerging research staff's psychical stimulation would undermine the very purpose of the university in meeting its community objectives. To achieve the correct balance, P-H University turns to the TM. The TM shows that psychical stimulation and motivated staff are balanced with its financial obligations to sustain the University's existence.

5.1 The CG for the example

P-H University's TM is given by Figure 7. From the TM modelled in CG for P-H University we can observe the following:

1. The University's TM reveals its validity through the costs being balanced by the benefits to the university achieving its community objectives. As Community Objective refers to communit*ites*, in CG it is shown as the plural referent `{*cmtobj}`.
2. The balancing of these debits and credits denotes the exchange of resources over and above the simple monetary aspects. Thus the qualitative Psychic Enjoyment is as much a part of the transaction as the quantitative Cash Payment.

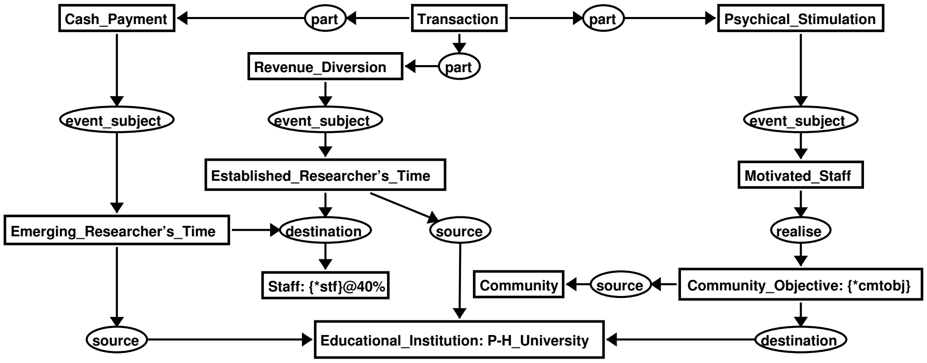

Fig. 7. CG of P-H University Scenario

P-H University's TM	Transaction part	Established Researcher's Time source	Psychical Stimulation event subject	Revenue Diversion event subject	Cash Payment event subject	Motivated Staff realise	Emerging Researcher's Time destination	Established Researcher's Time destination	Community Objective: {*cmtobj} destination	Community Objective: {*cmtobj} source	Emerging Researcher's Time source
Staff: {*stf}@40%	×			×	×		×	×			
Educational Institution: P-H University	×	×	×	×	×	×			×		×
Community Objective: {*}	×		×			×					
Motivated Staff	×		×								
Revenue Diversion	×										
Emerging Researcher's Time	×				×						
Community	×		×			×				×	
Established Researcher's Time	×			×							
Cash Payment	×										
Psychical Stimulation	×										
Transaction											

Fig. 8. Formal Context of P-H University Scenario

3. The TM shows that Cash Payment and Revenue Diversion versus Psychical Stimulation are the two complementary sets of economic events that trigger the transaction.
4. The event subject relations (i.e. the states altered by the economic events) point to the relevant economic resources that in this case are the researchers' time and staff motivation.
5. The source and destination relations (i.e. providers and recipients) of the economic resources are the agents in the transaction. These are the educational institution P-H University and the agents it transacts with, namely its staff and the community it serves.
6. The `{*stf}@40%` describes a plural of staff, specialised by the `@40%` thereby denoting the 40% of staff that are supported by emerging researchers time.
7. Motivated Staff is an economic resource of the University in that they add value to the assets of the University by being motivated.

5.2 *CGtoFCA*

Using a software implementation of the *CGtoFCA* algorithm (*CGFCA*, sourceforge.net /projects/cgfca/), and displaying the result using the Concept Explorer software (sourceforge.net/projects/conexp/), Figure 9 shows the FCA lattice for P-H University's TM in CG as shown by Figure 7. The formal context table is given by Figure 8.

As described earlier 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 `[Emerging_Researcher's_Time]`$^\frown$`(destination)` becomes the formal attribute `Emerging_Researcher's_Time destination` and `[Educational_Institution: P-H_University]` becomes the formal object `Educational_Institution: P-H_University`.

From the lattice for P-H University we can observe the following:

1. A node's own attribute(s) and its own objects follow the triple structure of CGs i.e. source concept → relation → target concept. For example:
   The `Cash_Payment event_subject` is `Emerging_Researcher's_Time` and `Motivated_Staff realise Community_Objective: {*cmtobj}`
2. The formal attributes and objects of a concept automatically 'collect' the the dependencies in the CG. For example:
   `Community` is dependent on `Community_Objective: {*cmtobj} source`, `Motivated_Staff realise`, `Psychical_Stimulation event_subject` and `Transaction part`.
3. The dependencies culminate in `Transaction`.

5.3 An Integrated, Interoperable Conceptual Structure

The lattice reveals that each CG concept is dependent on the CG concepts in its formal attributes and for the TM their reliance on the Transaction CG concept,

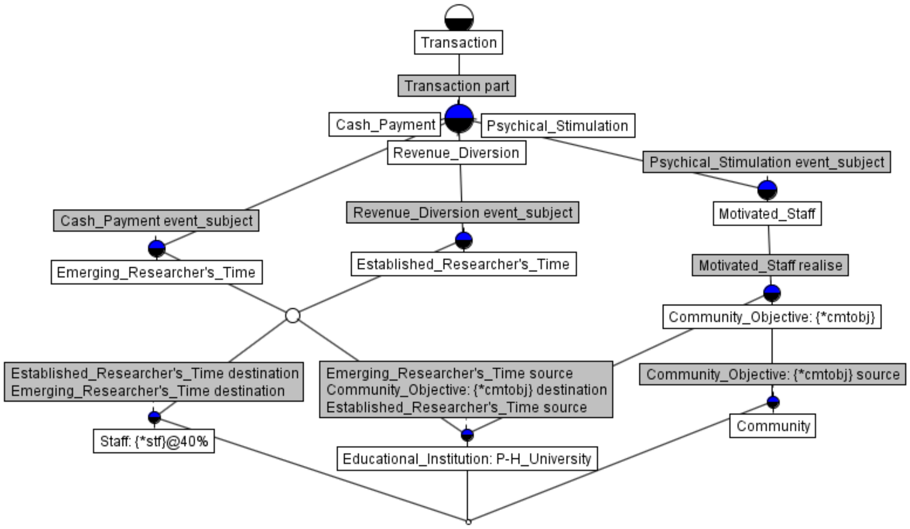
Transaction
Transaction part
Cash_Payment
Psychical_Stimulation
Revenue_Diversion
Psychical_Stimulation event_subject
Cash_Payment event_subject
Revenue_Diversion event_subject
Motivated_Staff
Emerging_Researcher's_Time
Established_Researcher's_Time
Motivated_Staff realise
Community_Objective: {*cmtobj}
Established_Researcher's_Time destination
Emerging_Researcher's_Time destination
Emerging_Researcher's_Time source
Community_Objective: {*cmtobj} destination
Established_Researcher's_Time source
Community_Objective: {*cmtobj} source
Staff: {*stf}@40%
Community
Educational_Institution: P-H_University

Fig. 9. Concept Lattice of P-H University Scenario

which is the epitome of the TM. In simple terms *all* the objects further down from `Transaction part` describe the extent of the transaction; without which the transaction cannot exist and the lattice shows the hierarchical interdependencies of the CG concepts and their relations. The direction of the arcs of the CGs model are preserved in the FCA lattice. As for each concept's referent, they too are preserved in exactly the same way as they appear in the CG.

As the inherent nature of CGs are preserved in *CGtoFCA*, CGs operations such as conformity, projection and maximal join can still be performed and iterated with those of FCA (e.g. attribute exploration). Thus to give one simple scenario, P-H University could model its TM in CGs, exploring the dimensions of its model through such CG operations. Using *CGtoFCA* it could simultaneously translate that TM into an FCA lattice and bring the power of FCA to bear on its TM too. Any enhancements that FCA identified could be translated back into CGs. Models could then be round-tripped between FCA and CGs. Put simply, through *CGtoFCA* we have merged CGs and FCA into a *single* and *interoperable* conceptual structure that provides a superset of operations by combining those of CGs with FCA.

5.4 An Enhanced TM

As a simple illustration let us re-examine Figure 9. Whilst it has identified Transaction as the overarching superconcept, interestingly there is no object identified with the bottommost subconcept. This prompts P-H University to re-examine its CGs TM. It sees that whilst the other concepts in this TM eventually point

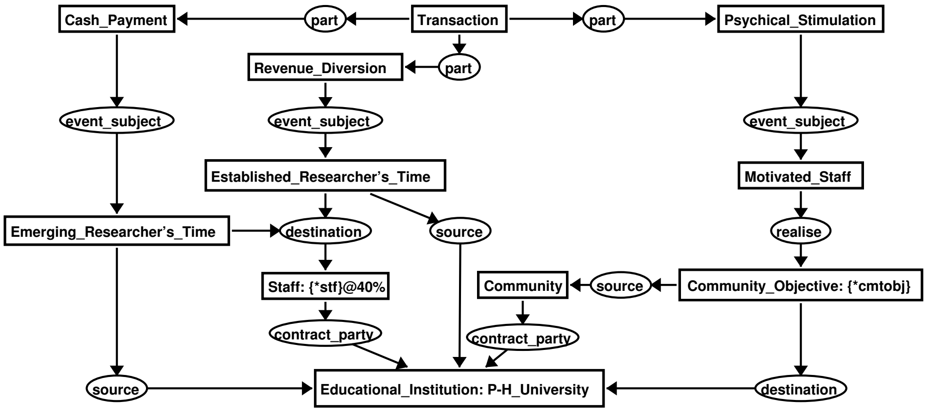

Fig. 10. CG of P-H University Scenario v2

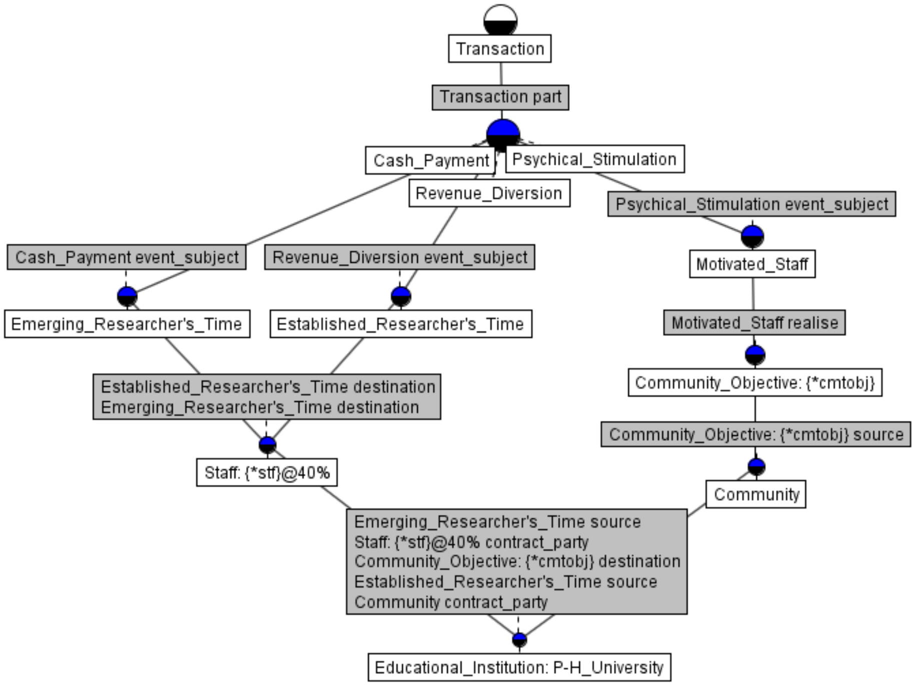
Transaction
Transaction part
Cash_Payment
Psychical_Stimulation
Revenue_Diversion
Psychical_Stimulation event_subject
Cash_Payment event_subject
Revenue_Diversion event_subject
Motivated_Staff
Emerging_Researcher's_Time
Established_Researcher's_Time
Motivated_Staff realise
Established_Researcher's_Time destination
Emerging_Researcher's_Time destination
Community_Objective: {*cmtobj}
Community_Objective: {*cmtobj} source
Staff: {*stf}@40%
Community
Emerging_Researcher's_Time source
Staff: {*stf}@40% contract_party
Community_Objective: {*cmtobj} destination
Established_Researcher's_Time source
Community contract_party
Educational_Institution: P-H_University

Fig. 11. Concept Lattice of P-H University Scenario v2

to it via its arcs and relations in that TM, there are none pointing from [Staff: {*stf}@40%] or [Community]. There should be some explicit relationship that points from these 'outside' agents in this transaction to P-H University, which is the 'inside' agent given it is P-H University's TM. Each of these outside agents is a contract party to P-H University. It therefore joins the following CG to its TM:

```
[Educational_Institution: P-H_University]-
 (contract_party)<-[Staff: {*stf}@40%]
 (contract_party)<-[Community]
```

The result is the CG TM Figure 10 and the lattice Figure 11. In comparison with Figure 9 we can now observe that the extent of the bottommost concept in Figure 11 is P-H University with the contract party relations duly added in the description of its attributes. Additionally, all the concepts contain own objects. This reveals that all the key concepts have been identified in both the CGs TM and the lattice TM; it transpires that there should not be a formal concept without at least one own attribute and at least one own object. Though such a simple change, it was not obvious in the previous work on the TM using CGs alone [6].

5.5 Wille's CG to FCA Approach

To evaluate the comparative value of *CGtoFCA*, the P-H University's TM in CG as originally depicted by Figure 9 was also converted into an FCA formal context and concept lattice based on the translation given by Wille [7]. Wille remarks that CGs capture knowledge at the logical level and FCA adds mathematical rigour, terming it as the 'mathematization' of conceptual structures. We have seen an illustration of the enhancement that FCA brings through the P-H University scenario. It is therefore particularly interesting to peruse Wille's own translation between CGs and FCA. The results according to Wille's translation are shown by Figures 12 and 13 respectively.

As described in the introduction and shown by these figures, Wille's translation essentially takes the referents of CG concepts, as aggregated by the relation that links the concepts. The resulting aggregated referents are then presented as formal objects with their formal attributes as the given relation. This is followed by the source and target concept's type label. As we have seen, the source type label is given the index '1' and the target is indexed as '2'. In line with Wille's translation for each object we have had also to give the referents an explicit identifier e.g. (#tx1,#cp1) to support the attributes part, Transaction 1 and Cash Payment 2.

Wille's translation also touches upon CG type and relation hierarchies in his examples, though co-referent links between concepts. Even allowing for this consideration, which is not replicated for the P-H University example, fundamental to Wille's translation is the choice of ordered pairs of instances of CG concepts as formal objects and the choice of both CG relations and CG concepts as formal attributes. Although the resulting concept lattice provides useful insights

P-H University's TM	part	event subject	source	destination	realise	Transaction 1	Cash Payment 1	Cash Payment 2	Emerging Researchers Time 1	Emerging Researchers Time 2	Educational Institution 2	Revenue Diversion 1	Revenue Diversion 2	Established Researchers Time 1	Established Researchers Time 2	Staff 2	Psychical Stimulation 1	Psychical Stimulation 2	Motivated Staff 1	Motivated Staff 2	Community Objective 1	Community Objective 2	Community 2
(#tx1,#cp1)	×					×		×															
(#tx1,#ps1)	×					×												×					
(#tx1,#rd1)	×					×							×										
(#cp1,#emrt1)		×					×			×													
(#rd1,#esrt1)		×										×			×								
(#ps1,#ms1)		×															×			×			
(#emrt1,P-H University)			×						×		×												
(#esrt1,P-H University)			×								×			×									
(#ms1,{*cmtyobj})					×														×			×	
({*cmtyobj},#cmty1)			×																		×		×
(#emrt1,#esrt1,{*stf}@40%)				×					×					×		×							
({*ctmyobj},P-H University)				×							×										×		

Fig. 12. Formal Context of P-H University Scenario after Wille

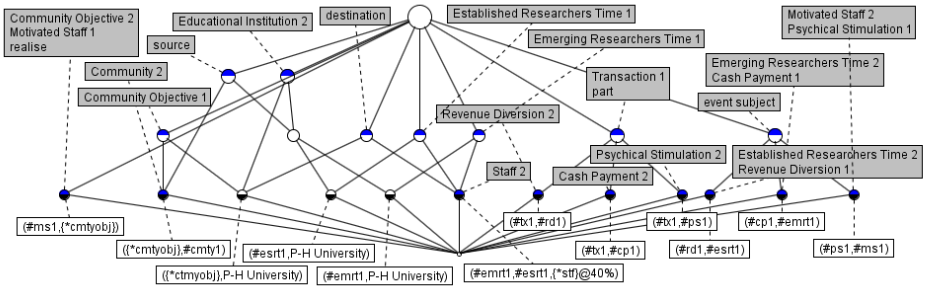

Fig. 13. Concept Lattice of P-H University Scenario after Wille

into the underlying CGs, it can be argued that this approach leads on the one hand to lists of object pairs that share the same relation but on the other hand lead to rather more complex lattices that show a hierarchy of separated out CG source and target concepts and relations. This is evidenced by the outcomes demonstrated by Figures 12 and 13 for P-H University. The elegant interdependence and simpler lattices as shown in Figure 9 is not as easily discerned and the insight that lead to Figure 11 is not evident. Notably, Transaction as the key concept in the TM is not at the head of the lattice. For the TM it is the relationship of other concepts to the Transaction concept that is the nub of the TM as we have seen.

6 Conclusions

We have demonstrated *CGtoFCA* as a straightforward mapping from Conceptual Graphs (CGs) to Formal Concept Analysis (FCA). CGs and FCA can thus interoperate in a practical way, combining their power in an integrated and intuitive conceptual structure. The simple case study (P-H University) has shown an enhancement to the Transaction Model (TM) modelled in CGs alone. It thus illustrates the benefits of *CGtoFCA*, which will spur the wider development of conceptual structures' and their practical applications.

References

1. Dau, F., Klinger, J. (2005). “From Formal Concept Analysis to Contextual Logic” Lecture Notes in Computer Science 3626, Berlin: Springer. pp. 81-100.
2. Delugach, H., Lampkin, B. (2000). “Troika: Using Grids, Lattices and Graphs in Knowledge Acquisition”. In: *Working with Conceptual Structures: Contributions to ICCS 2000*, G. Stumme, Ed. Aachen, Germany: Shaker Verlag, pp. 201-214.
3. Ganter, B., Rudolph, S (2001). “Formal Concept Analysis Methods for Dynamic Conceptual Graphs”. In: H. Delugach and G. Stumme (Eds.): ICCS 2001, LNAI 2120. Springer-Verlag Berlin Heidelberg. pp. 143-156.
4. Kuznetsov, S. O. (1999). “Learning of Simple Conceptual Graphs from Positive and Negative Examples”, Lecture Notes in Computer Science 1704, Berlin: Springer. pp. 384-391.
5. Mineau, G., Stumme, G., Wille, R. (1999) “Conceptual Structures Represented by Conceptual Graphs and Formal Concept Analysis”, in *Conceptual Structures: Standards and Practices*, Tepfenhart, W., Cyre, W. (Eds.), Lecture Notes in Computer Science 1640, Springer Berlin / Heidelberg. pp. 423-441.
6. Polovina, S., Hill, R. (2009) “A Transactions Pattern for Structuring Unstructured Corporate Information in Enterprise Applications”, *International Journal of Intelligent Information Technologies*, April-June 2009, Vol. 5, No. 2, IGI Publishing, 34-47.
7. Wille, R. (1997). “Conceptual Graphs and Formal Concept Analysis”. In: Lukose, D.; Delugach, H.S.; Keeler, M.; Searle, L.; Sowa, J.F. (eds.) *Conceptual Structures: Fulfilling Peirce's Dream, Fifth International Conference on Conceptual Structures, ICCS 1997*, Seattle, Washington, USA, August 3-8, 1997, Lecture Notes in Computer Science 1257. Berlin: Springer. pp. 290-303.