Modelling Business Transactions across Service Supply Chain Networks

Noel Carroll, Rafiqul Haque, Eoin Whelan, and Ita Richardson

Lero - the Irish Software Engineering Research Centre
Department of Computer Science & Information Systems
University of Limerick, Limerick, Ireland
Email :{Noel.Carroll, Rafiqul.Haque, Ita.Richardson}@lero.ie, Eoin.Whelan@ul.ie

Abstract. This paper is concerned with understanding the complex nature of service network environments with particular attention on exploring business transactions across supply chains. Although business transactions have been traditionally well documented throughout literature, what becomes apparent is that these approaches fail to capture the dynamic complexity of modern service supply chains. To address the problem, this paper introduces a method to model supply chain behaviour which is of particular interest at the network design time and offers a conceptual view of extracting network analysis and process metrics. We introduce a business transaction language (BTL) to gain insight into the business transactions while we also explore the application of social network analysis (SNA) to model the dynamism of service networks. In doing so, the research sets out to generate greater service network intelligence and extend the service network ontology while visualising the transactional interaction landscape.

Keywords: service network, business transaction, business transaction language, social network analysis, supply chain, service infrastructure

1 Introduction

In today's service-dominant business environment (Normann, 2001; Spohrer et al., 2007), harnessing innovative applications of technology is considered one of the critical factors towards organisational sustainability. In addition, with the emergence of 'Cloud Computing', understanding the application and indeed '*value'* of technology to support infrastructure as a service (IaaS) is becoming increasingly more important. Consequently, the application of technology to support services has altered our traditional understanding of the 'organisation', making it more difficult to conceptualise the paradigm of services. Thus, the emergence of 'Service Science' as a discipline, has underscored the importance of understanding the complexities of service and their intertwining properties (Normann, 2001; Chesbrough and Spohrer, 2006; Spohrer et al., 2007; Spohrer and Maglio, 2009). In doing so, the discipline of Service Science has received much interest, although little attention has been placed thus far, on exploring the contribution of information technology (IT) within a service network and the impact of IT on relational structures which support service networks. To align with this recent paradigm, we propose the need to expand on modelling techniques to accommodate for the 'modern' and digitised service environment by introducing a business transaction language (BTL). The development of BTL attempts

---

The research leading to these results has received funding from the European Community's Seventh Framework Programme FP7/2007-2013 under grant agreement 215483 (S-Cube). It was supported, in part, by Science Foundation Ireland grant 03/CE2/I303_1 to Lero – the Irish Software Engineering Research Centre (www.lero.ie).

to gain a more fundamental insight of the business transaction model which specifies the elements of a business transaction from an application perspective. As this paper will demonstrate, a business transaction is a series of collaborative activities distributed across the multiple partners in service networks that explicitly enforce the achievement of an agreed-upon business objective in end-to-end processes (S-Cube, 2010). Understanding how transactions are influenced by the service architecture and vice versa, is of critical importance in the service environment, as we discuss throughout this paper.

2 Modelling Business Transactions

Modelling business transaction denotes specifying the structural and behavioural aspects of transactions that underpin the applications such as Service Based Application (SBA). Structural aspect deals with the construction of transaction such as nested structure. In this research, we do not concentrate on structural aspect because a list of transaction models such as distributed transaction as suggested by (Gray, 1993) and open nested transaction model (Traiger, 1983) are structurally rich enough to be used for business transactions in service network environment. Our focus is rather on behavioural aspect which deals with the operations including process level operations (e.g., refund) and system level operations (e.g., commit). In this section, we discuss different issues that are related to business transaction behaviour.

Behavioural aspect of business transaction in service network involves business perspective that entails real-world business elements. Within service network environment, businesses are collaborative representations among partners which are achieved through a mutual understanding on obligations and guarantees in terms of qualities, actions etc. This coalition (mutual relationship) is formally known as *agreement* which is an element of business perspective. The collaborative partners have to comply with the agreement while carrying out the transactions. Thus, agreement is considered as one of the primary determinants of transaction behaviours in service network environments. In particular, the obligations in terms of quality of service contained in agreements, may promote the failure atomic behaviour of transactions. For example, a buyer may cancel an order if the seller fails to satisfy the agreed delivery lead-time; this results the failure of list of transactions that have already committed successfully. This behaviour is similar to the notion of atomicity. The example shows how agreement influences the failure atomic behaviour of business transaction. Additionally, obligations in terms of actions (service operations) are a set of rules that impose constrains on service transaction. These rules must be satisfied while performing transactions independently by collaborating partners.

Furthermore, vitality is an attribute of business process activities (note that activity specifically, a composite activity is also called *service*) which influences the behaviour of business transaction in service networks. Some participating services in an SBA may be vital that must commit successfully; because, the successful commit of a business transaction depends on the commit of each these vital services. For example, the successful commit of a business transaction depends on the commit of *order processing*, *payment*, and *delivery* services. Failure of any of these activities fails the transaction entirely. In addition, if the transaction of a vital service fails, transactions of all the services that are *dependent* on the vital one also fail. This indicates that dependency is an important issue for business transaction behaviour.

Transaction behaviour is also determined by a list the properties. The widely known transaction properties are atomicity, consistency, isolation, and durability, shortly called ACID (Harder, 1983). The database application relies heavily on these properties but they (specifically atomicity and isolation) are not suitable for complex applications (Gray, 1993; Chrynsanthis, 1990) such as SBAs. Thus, the transaction models (e.g., Nested, Multilevel, and Flat) that are built on these properties lack the efficiency when used for new applications (Chrynsanthis, 1990; Garcia-Molina, 1987) such as SBAs. The strictness of these properties is the primary reason for inefficiency. More specifically, atomicity does not allow any failure in transaction and isolation uses two-phase locking protocol, which degrade the performance of business transaction (Gray, 1993). Nonetheless flexibility is an important metric to measure efficiency in business transaction. In a word, flexible behaviour is highly desired in business transaction for service network.

Decidedly, a business transaction model should (i) incorporate business elements (ii) facilitate specifying the attributes of business process activities (iii) ensure flexible behaviour. How to define such business transaction model? Traditional approach in particular, hardwiring business elements and attributes in application logic is not feasible for transactions in service network environment due to many reasons predominantly managing the runtime changes. In service network environment, services may need to be integrated with running application *on the fly* because of the demand; this is called Just-in-Time (JIT) integration. JIT integration is advantageous for SBAs that support end-to-end business processes in service network because it is hard to perceive the service demand in advance. However, the integration of new service may require changes in transactional attributes. For instance, if a shipper is replaced by a new shipper, the transactional attributes (e.g., delivery time) may need to be changed. In addition, traditionally transactions are defined by the application programmer but the participation of business personnel (e.g., business analysts) is enormously important especially for business transactions in service network environment. Taking these issues into consideration, we proposed a design-time Business Transaction Language (BTL) (S-Cube, 2010) in the earlier phase of this research. In this paper, we present the recent development of BTL.

Before going into the details about BTL, we discuss our contribution to ensure flexible transaction behaviour. For flexible transaction behaviour, we refine the classical transaction properties in particular, atomicity and isolation. We do not consider consistency and durability within the scope. Table 1 shows these properties with a brief description of each.

Table 1 List of business transaction properties

Transactional Properties		Description
Behavioural	Flex-Atomicity	Atomicity cannot be fully compromised due to the fact that complete non-atomicity may increase the transaction failure rate exponentially. But flexibility guarantees that a failure of a transaction can be repaired using contingent transaction. We also call this flex-atomicity as *eventual failure atomicity* which denotes a transaction is failed entirely only when it is beyond to be repaired.
	Relaxed-Isolation	Relaxed-isolation determines the visibility of outcomes. The classical isolation property do not allows externalizing the outcome of a child

		transaction until the parent transaction commits. Relaxed-Isolation ensures the externalization of every change that happens after operation. For instance, the transition of the state of transactional activity is notified to all of its collaborating partners at once. Relaxed-isolation is prone to inconsistency. In fact, it is the trade-off between consistency and performance. In service network environment, performance is a primary factor to gain customer satisfaction. Thus, relaxed-isolation may be more suitable than the classical isolation by tolerating inconsistency at lower extent.

BTL allows specifying the above properties. BTL is a declarative language that is used to model business transactions in service networks. It is declarative because it describes *what* (transactional elements) should be defined in a business transaction model but not *how* they should be implemented. The implementation of BTL is not within the scope of this paper. However, BTL comprises of three main perspectives embodying *business perspective, functional perspective,* and *protocol perspective.*

```
BTL>
       <BusinessPerpsective>
             <! -- Business perspective contains the
             business related elements -->
             <element> name = "" </element>
       </BusinessPerpsective>
       <FunctionalPerpsective>
             <! -- Business perspective contains
             function related elements such as
             activities -->
             <element> name = "" </element>
       </FunctionalPerpsective>
       <ProtocolPerspective >
             <! -- protocol description contains list
             of Protocols including business and
             technical -->
             <element> name = "" </element>
       </ProtocolPerspective>
 </BTL>
```

Listing 1: The code snippet shows thee aspects of BTL

BTL facilitates capturing business, functional, and protocol related elements through these perspectives. The code snippet in listing 1 shows the fundamental structure of BTL. Business Transaction is the central entity that is lined with all these perspectives using different relational semantics. We claim BTL a novel approach owing to its ability to capture the business elements that are critical for business transactions in service network.

SLA is the primary element of business perspective, which influences the behaviour of business transactions in service network. It is formal agreement between two to many partners. SLA contains a set of obligations and service guarantees. *Policy* and *Quality of Service (QoS)* are the constructs used to capture the obligations and service guarantees respectively. The participants are agreed on such obligations and guarantees that must be satisfied while performing business transactions. The functional perspective of BTL entails business functions (which are in fact composite services or processes). Business functions are composed of events and activities. An activity is an abstraction of a piece of work that's has to be performed within a process whereas an event denotes an occurrence or something has happened in a process (Leymann, 2000). Since flex-atomicity is a desired property of business transaction in service network, BTL allows specifying contingency activity and compensating activity. The key purpose of these activities is to prevent the total failure of business transactions. Listing 2 shows the code snippet of business and functional elements.

In a service network, business transactions happen through executing business functions in an ordered manner. For instance, a purchase order business transaction may follow the execution order of business functions in following sequence: order registration, payment and delivery. In contrast, the delivery transaction may happen before payment or payment can be split into before and after delivery. The execution order is important to the business transactions in service network since they ensure correctness of the business transactions. This order is maintained using business protocols. Business protocol defines the ordering in which a particular partner sends messages to and expects messages from its partners based on an actual business context and captures all behavioural aspects that have cross-enterprise business significance (Papazoglou, 2003). BTL includes protocol perspective that allows specifying the business protocol, in other words, specifying the order of business functions, which determines the order of business transactions. See the code snippet in listing 2.

In this section, we have discussed the issues that need to be considered, suggested a list of properties that are important for modelling business transactions in service networks, and presented the core XML syntax of BTL.We provide a detailed description of transaction behaviour of business transactions in section 3 using a scenario.

2.1 Social Network Analysis and BTL

Since we have established that a service comprises of complex exchanges of resources and competencies it is important to understand how service interactions deliver these. SNA presents the technique to investigate network on analysis (for example, structure of service relationships) and explore other network characteristics (Carroll et al., 2010). The term 'relational structure' may be used to describe a "bundle of intuitive natural language ideas and concepts about the patterning" (Freeman et al., 1992; p. 12) in service relations among organisations to deliver a service. For example, Freeman et al., (1992; p.12) describes a 'social network' is "a collection of more or less precise analytic and methodologically concepts and procedures that facilitate the collection of data in a systematic study of such patterning". Thus, we can develop a greater understanding of a service network by describing its structure (nodes and links) and its behaviour (what the network does as a result of interactions among the nodes and links). This also supports the execution of a business transaction. This has an important application to the management, engineering, and design of service networks, i.e. service science.

```
<BTL>
      <BusinessPerpsective>
            <KPI indicators = ""/>
            <Agreement>
                  <participant name =" Participant Name
                  " type ="xs:string" minOccurs ="2"
                  maxOccurs="n"/>
                  </participants>
                  <Policy type = type = "xs:string">
                  <expression>.....</expression>
                  <Policy>
                  <QoS>
                        <QoSAttribute name ="" type =
                        "xs:string"/>
                        <QoSMetric name =""/>
                         <QoSTargetValue value = ""
                         valueType = "xs:string"/>
                  </QoS>
            </Agreement>
      </BusinessPerpsective>
      <FunctionalPerpsective>
            <BusinessFunction>
                  <event name = "" type = "xs:string" />
                   <activity name = "" type = "xs:string"
                   vital = "xs:boolean"/>
            </BusinessFunction>
      </FunctionalPerpsective>
      <ProtocolPerspective>
            <BusinessProtocol>
                  <sequence>
                  ................................
                   </sequence>
            </BusinessProtocol >
      </ ProtocolPerspective >
</BTL>
```

Listing 2: The code snippet shows the elements that are used to model business transactions in SNs.

Network analysis emphasises the relations which connect the node positions within a system, and “offers a powerful brush for painting a systematic picture” of a service infrastructure and their interaction (Knoke and Kuklinski, 1991; p. 173). Simply put, Salancik (1995), explains that “a network theory about organisations should also be able to say how network properties themselves generate the properties of organisations” (p. 349). The major characteristics of attribute analysis are that the unit of analysis is the individual actor and the variable describes the behaviour of the network actor. We can apply Lewis (2009 p.20-21) list of the key characteristics of network science of the modern service era (Table 2):

Table 2 General Principles of a Network (Lewis, 2009)

Characteristic	Description
Structure	Not a random collection of nodes and links and have a distinct format or topology which suggests that function follows form.
Emergence	Network properties are emergent as a consequence of a dynamic network achieving stability.
Dynamism	Dynamic behaviour is often the result of emergence or a series of small evolutionary steps leading to a fixed-point final state of the system.
Autonomy	A network forms by the autonomous and spontaneous action of interdependent nodes the "volunteer" to come together (link), rather than central control or central planning.
Bottom-up Evolution	Networks grow for the bottom or local level up to the top or global level. They are not designed and implemented from the top down.
Topology	The architecture or topology of a network is a property that emerges over time as a consequence of distributed and often subtle forces or autonomous behaviours of its nodes.
Power	The power of a node is proportional to its degree (number of link connecting to the network), influence (link values), and betweenness or closeness; the power of a network is proportional to the number and strengths of its nodes and links.
Stability	A dynamic network is stable if the rate of change in the state of its nodes/links or its topology either diminishes as time passes or is bounded by dampened oscillations within finite limits.

Applying these principles to a service network, SNA may be simply described as an x-ray of the organisational service structure which highlights the importance relational structures to support service performance (Carroll et al., 2010). According to Tichy et al., (1979), network analysis is concerned with the structure and pattern of these relationships and seeks to identify both their causes and consequences (p. 507). This may be mapped onto the business transaction to allow the transactions 'tell the story' of service interactions. Tichy et al., (1979) provides an overview of network concepts and network properties as listed in table 2 which are considered fundamental to the business transaction and service network performance. As identified in Table 3, we can borrow network analysis concepts to extend BTL development. Within a service context, these modifications are often initiated by a customer and influenced by business transaction properties, for example an SLA which impact the relational content. SNA provides us with the opportunity to visualise the dynamic infrastructure of the transactional network and SNA properties (table 3) can provide greater semantics to BTL and the nature of the service infrastructure. Therefore the orchestration of structural relations (emergent property of the connection, the exchange process) or attributes (intrinsic characteristics, e.g. value of an exchange) becomes a central concept to analyse a networks structural properties.

Table 3 Organisational network analysis concepts and network properties

Property	Explanation
Transactional Content	Four types of exchanges: 1. Expression of effect (e.g. initiate a transaction). 2. Influence attempt (e.g. negotiating a SLA). 3. Exchange of information (e.g. terms and conditions). 4. Exchange of goods and services (e.g. payment).

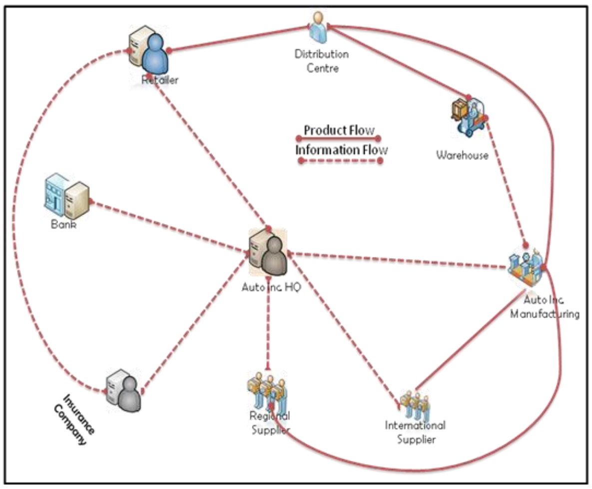
Nature of links	
1. Intensity	The strength of the relations between individuals (i.e. intensity of service interactions).
2. Reciprocity	The degree to which a relation is commonly perceived and agreed on by all parties to the relation (i.e. the degree of symmetry).
3. Clarity of Expression	The degree to which every pair of individuals has clearly defined expectations about each other's behaviour in the relation, i.e. they agree about appropriate behaviour between one another (i.e. SLA).
4. Multiplexity	The degree to which pairs of individuals are linked by multiple relations. Multiple roles of each member (e.g. consumer, supplier, negotiator, etc) and identifies how individuals are linked by multiple roles (the more roles, the stronger the link).
Structural	
1. Size	The number of individuals participating in the network (i.e. service eco-system).
2. Density (Correctedness)	The number of actual links in the network as a ratio of the number of possible links (can determine service opportunity).
3. Clustering	The number of dense regions in the network (i.e. network positioning, structural holes which may identify service threats).
4. Openness	The number of actual external links of a social unit as a ratio of the number possible external links (e.g. level of outsourcing).
5. Stability	The degree to which a network pattern changes over time (i.e. level of innovation and service evolution).
6. Reachability	The average number of links between any two individuals in the network (e.g. service brokerage).
7. Centrality	The degree to which relations are guided by the formal hierarchy (may determine power within a service network).
8. Star	The service with the highest number of nominations (can influence the level of success a service enjoys).
9. Liaison	A service which is not a member of a cluster but links two or more clusters.
10. Bridge	A service which is a member of multiple clusters in the network (linking competencies and industry-specific services).
11. Gatekeeper	A star who also links the social unit with external domains (i.e. knowledge diffusion and service network analyst).
12. Isolate	A service which has uncoupled from the network (service termination).

To describe our approach to date, we discuss a simple scenario which examines how service transactions consist of complex relations which allows us to apply BTL to understand how business transactional applies to a service model. In addition, we demonstrate how SNA provides a technique to model transactional relational flows to further extract service metrics within the network.

3 Scenario – Business Transactional Analysis

Service networks are not independent, isolated entities, but act in a concerted manner to survive in an ever increasing dynamic business environment. Therefore, interacting organisations build networks to serve their customers in a dynamic manner (S-Cube, 2009). In this section, we analyse business transactions using a supply chain scenario of Auto Inc. is located in South East Asia (S-Cube, 2008). Auto Inc. is a local branch of a large enterprise in the automobile industry in Europe. The Auto Inc. supply network comprises of a regional headquarters in Singapore, a manufacturing factory in Vietnam, local and international suppliers, distribution centre, and several warehouses located in different countries in South East Asia. This demonstrates the distributiveness of the Auto Inc. network which has been spanned across wide territories. We develop a service network scenario including bank and insurance as service partners with the existing business partners. Figure 1 depicts the service network scenario of Auto Inc.

Figure 1. Auto Inc. service network scenario comprises of partners entailing buyers, sellers, banks, insurance company, warehouse, distribution centre, and suppliers.



Auto Inc. supply chain network embodies suppliers from different regions and continents in particular, from Europe. Auto Inc. does not have logistics partner because the suppliers provides logistics support. In addition, the distribution centre is responsible for the transportation of tangible service (product) to the retailers. This setting manifests that the inter-organisational service network structure is highly complex since it connects many disparate and spatial service nodes that fire up numerous interactions among them.

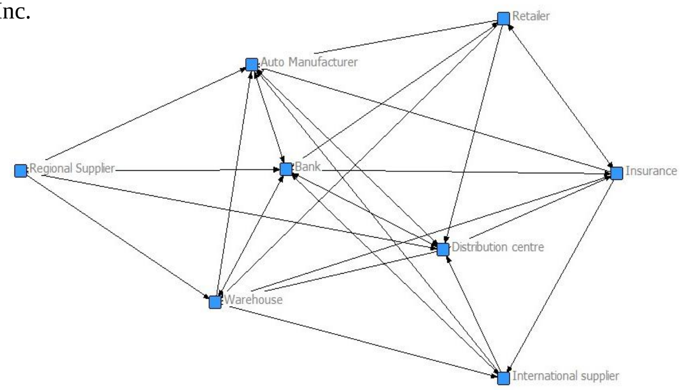
The SBA that supports the end-to-end processes of the Auto Inc. service network comprises order processing, delivery and payment services. These are composite services encapsulating operations that perform the requests triggered by the partners across the network. The transactional properties are blended with these services, for which they are treated as transactional services. For this scenario, we consider that BTL has been used to define the transactional properties of these services at design-time. These properties determine the behaviour of business transactions in Auto Inc. service networks. We discuss the three most important properties including business behavioural consistency, flex-atomicity, and reflexivity (we defined in section 2) in this section. We discuss these properties to analyse the behaviours of transactions at runtime.

Business transactions in service networks (figure 1) are said to be consistent if and only if the business transactions satisfy the obligations and guarantees such as response time, processing lead-time, delivery lead-time, and data privacy. We name this consistency as *business behavioural consistency*. The obligations and guarantees that determine business behavioural consistency are stipulated in SLA, an element of the business perspective of BTL. A transaction monitor is employed to monitor SLA at runtime. The monitor flags the transactions as inconsistent if they fail to satisfy SLA during as well as after the execution. The business behavioural inconsistency may result failure of a transaction. For instance, if a delivery from international supplier is delayed, it is the behavioural inconsistency of the supplier. It affects the transaction in several ways such as Auto Inc. may cancel the order which fails the transaction between Auto Inc. and the supplier. The effect of this failure may propagate throughout the network if Auto Inc. fails to deliver the cars ordered by retailers. The retailer cancels the order, which fails the transaction entirely for this particular instance (purchase order).

Furthermore, if a regional supplier fails to deliver the raw materials (e.g., CD player) the transaction between regional supplier and Auto Inc. fails. According to the notion of classical atomicity, the transactions that are linked to this failed transaction for instance, the transaction between retailer and Auto Inc. should be aborted immediately. This atomicity has been relaxed in business transaction in Auto Inc. service network by invoking the compensation activity as well as contingent activity that are supplied by BTL (note that, the transaction manager integrates BTL). We call it flex-atomicity. The compensation activity undoes the results of the transaction between regional supplier and Auto Inc. and contingent activity initiate an alternative transaction which repairs the failed transaction. This resists the failure of business transaction entirely.

Business in the given scenario (figure 2) is collaborative. Thus, it is important that any change in end-to-end process in Auto Inc. service network should be externalized immediately to the other partners that are part of the collaboration. This is *relaxed-isolation* property of business transaction. Note that *relaxed-isolation* allows exceptions. In particular, the failure of a transaction may not always be broadcasted across the partners in the service networks to resist the total failure of transactions. We suggest masking these failures using some techniques. The current version of BTL does not facilitate defining *relaxed-isolation* property. This work is in progress which we will add in the next version of BTL.

Figure 2. Auto Inc. through SNA



From the above discussion, it is clear that transactional properties play the pivotal role to controlling the runtime service behaviours. Thus, we analyse the behaviour (level of interaction) of transactions in the Auto Inc. service network through its interactions as we employ SNA. As figure 2 illustrates, SNA allows us to visualise the service network through the level of interaction (for more information on the application of SNA to service networks refer to Carroll et al., 2010). Figure 1 illustrates the service network from a process perspective while figure 2 highlights the importance of actors such as the bank to support the service operations. The bank is a key actor within the service network since it finances many of the service operations and consequently is positioned within the centre of the service network SNA map. This is true from a business perspective and a customer perspective, as it relies on finance to, for example, avail of car loans. In this simple example, there are 32 ties within the network and a density (proportion of ties relative to the number of possible ties) of 0.5714 which suggests that there is opportunity to improve the service network cohesion. In addition, there are several key metrics which were outlined in table 3 which support the development of service analysis through BTL.

Table 4 Service Supply Chain Density Measures

Density Measures

		1 Size	2 Ties	3 Pairs	4 Densit	5 AvgDis	6 Diamet	7 nWeakC	8 pWeakC	9 2StepR	10 ReachE	11 Broker	12 nBroke	13 EgoBet	14 nEgoBe
1	Regional Supplier	4.00	10.00	12.00	83.33	1.17	2.00	1.00	25.00	100.00	24.14	1.00	0.17	0.00	0.00
2	International supplier	5.00	15.00	20.00	75.00	1.25	2.00	1.00	20.00	100.00	20.00	2.50	0.25	0.67	3.33
3	Auto Manufacturer	7.00	23.00	42.00	54.76			1.00	14.29	100.00	17.07	9.50	0.45	0.58	1.39
4	Warehouse	7.00	24.00	42.00	57.14			1.00	14.29	100.00	16.67	9.00	0.43	1.17	2.78
5	Distribution centre	7.00	23.00	42.00	54.76			1.00	14.29	100.00	16.67	9.50	0.45	2.50	5.95
6	Retailer	5.00	15.00	20.00	75.00	1.25	2.00	1.00	20.00	100.00	20.00	2.50	0.25	0.33	1.67
7	Bank	7.00	19.00	42.00	45.24			1.00	14.29	100.00	16.67	11.50	0.55	10.75	25.60
8	Insurance	6.00	20.00	30.00	66.67	1.33	2.00	1.00	16.67	100.00	17.95	5.00	0.33	1.17	3.89

1. Size. Size of ego network.
2. Ties. Number of directed ties.
3. Pairs. Number of ordered pairs.
4. Density. Ties divided by Pairs.
5. AvgDist. Average geodesic distance.
6. Diameter. Longest distance in egonet.
7. nWeakComp. Number of weak components.
8. pWeakComp. NWeakComp divided by Size.
9. 2StepReach. # of nodes within 2 links of ego.
10. ReachEffic. 2StepReach divided Size.
11. Broker. # of pairs not directly connected.
12. Normalized Broker. Broker divided by number of pairs.
13. Ego Betweenness. Betweenness of ego in own network.
14. Normalized Ego Betweenness. Betweenness of ego in own network.

For each pair of nodes within the service network SNA calculates the number of edges in the shortest path between them. The average distance (among reachable pairs) is 1.347 while the distance-based cohesion ("Compactness") within the network is 0.723. In general, the range is 0 to 1 where larger values indicate greater cohesiveness. The distance-weighted fragmentation ("Breadth") of the network is 0.277. As suggested above, the density of the network allows us to identify the number of actual links in the network as a ratio of the number of possible links. This can also suggest possible opportunities. These density measures, for example, are summarised in table 4. These measures assist us gain insight as to the structure of the network upon which BTL is deployed to model service behaviour as discussed above.

4 Conclusion

This paper is the result of the first phase of our research work on business transaction language and service network modelling analysis. BTL as described in this paper is a specification containing expected details of transaction and as part of our future work; we are currently working on mapping the transaction properties (defined using BTL) to runtime language which will realise these properties. We believe that there is a list of works to be carried to successfully complete the BTL. In this paper we emphasise the need for BTL to enhance performance reporting mechanisms. From our perspective, the requirement of a language must be well-justified before the development of the language; otherwise, it can be obsolete like many other languages. In addition, we identify the suitability of SNA to support our quest to examine behavioural and structural aspects of service networks and its influence on service performance. As part of our future work we are continuing to develop BTL and apply this within a real world case study. In addition, we plan to merge SNA analysis metrics to enhance service network modelling analysis techniques which is sought in service management and service computing fields.

References

Carroll, N, Whelan E, Richardson I. (2010). Applying Social Network Analysis to Discover Service Innovation within Agile Service Networks. Service Science. 2(4): pp225-244.

Chesbrough, H., and Spohrer, J., (2006). A research manifesto for services science, Communications of the ACM, July, Volume 49, Number. 7, pp. 35-40.

Chrysanthis K.P. and Ramamritham K., (1990). ACTA: A framework for specifying and reasoning about transaction structure and behavior. In Proc. ACM SIGMOD Symp. on the Management of Data.

Freeman, L. C., White, D. R., and Romney, A. K., (1992). Research Methods in Social Network Analysis. Transaction Publishers, New Brunswick, New Jersey.

Garcia-Molina, H. And Salem, K., (1987). "Sagas", Proceedings of ACM SIGMOD International Conference on Management of Data.

Gray, J. and Reuter A., (1993). Transaction Processing: Concepts and Techniques. s.l. : Morgan Kaufmann, San Francisco, CA.

Harder, T. and Reuter, A., (1983). Principle of transaction oriented database recovery. ACM computing Survey. Vol. 15(4), pp. 287-317.

Knoke, D., and Kuklinski, J. H., (1991). Network Analysis: Basis Concepts, Chapter 14. In Thompson, G., Frances, J., Levačić, R., and Mitchell, J., (1991). Markets, hierarchies and networks: the coordination of social life. Sage Publications.

Leymann, F., Roller, D(2000). Production Workflow: Concepts and Techniques. Prentice Hall PTR, Upper Saddle River, NJ, U.S.A.

Lewis, T. G., (2009). Network Science—Theory and Application, Hoboken, NJ: John Wiley & Sons.

Normann, R., (2001). Reframing business: when the map changes the landscape. Chichester, New Sussex: Wiley

Papazoglou, M. P., (2003). Web Services and Business Transactions. World Wide Web: Internet and Web Information Systems, Volume 6, Number 1, pp. 49-91.

Salancik, G. R. (1995). Wanted: A good network theory of organization. Administrative Science Quarterly, Volume 40, pp. 345-349.

S-Cube, (2008): S-Cube project website. http://www.s-cube-network.eu/

S-Cube,(2009): Initial models and mechanisms for quantitative analysis of correlations between KPIs, SLAs and underlying business processes. Available Online: http://www.s-cube-network.eu/results/deliverables/wp-jra-2.1

S-Cube, (2010). Business Transaction Language. Available Online: http://www.s-cube-network.eu/results/deliverables/wp-jra-2.1

Spohrer, J., Maglio, P. P., Bailey, J., and Gruhl, D., (2007). ‗Steps Toward a Science of Service Systems,‘‘ IEEE Computer, Volume 40, Number 1, pp. 71-77.

Spohrer, J. and Maglio, P. P. (2009). Service Science: Toward a Smarter Planet. In Service Engineering, ed. Karwowski & Salvendy. Wiley. New York, NY.

Tichy, N. M., Tushman, M. L., and Frombrun, C., (1979). ―Social Network Analysis for Organizations, Academy of Management Review, Volume 4, pp. 507-519.

Traiger L.I., (1983). Trends in System Aspects of database Management. In Proceeding 2nd International Conference on Databases (UCOD-2).