



HAL
open science

Towards the understanding of the requirements of a communication language to support process interoperation in cross-disciplinary supply chains

Bishnu Pada Das, R I M Young, K Case, S Rahimifard, C Anumba, N Bouchlaghem, Anne-Francoise Cutting Decelle

► **To cite this version:**

Bishnu Pada Das, R I M Young, K Case, S Rahimifard, C Anumba, et al.. Towards the understanding of the requirements of a communication language to support process interoperation in cross-disciplinary supply chains. *International Journal of Computer Integrated Manufacturing*, 2007, 20 (04), pp.396-410. 10.1080/09511920600873741 . hal-00513382

HAL Id: hal-00513382

<https://hal.science/hal-00513382>

Submitted on 1 Sep 2010

HAL is a multi-disciplinary open access archive for the deposit and dissemination of scientific research documents, whether they are published or not. The documents may come from teaching and research institutions in France or abroad, or from public or private research centers.

L'archive ouverte pluridisciplinaire **HAL**, est destinée au dépôt et à la diffusion de documents scientifiques de niveau recherche, publiés ou non, émanant des établissements d'enseignement et de recherche français ou étrangers, des laboratoires publics ou privés.



Towards the understanding of the requirements of a communication language to support process interoperation in cross-disciplinary supply chains

Journal:	<i>International Journal of Computer Integrated Manufacturing</i>
Manuscript ID:	TCIM-2005-IJCIM-0016.R1
Manuscript Type:	Original Manuscript
Date Submitted by the Author:	06-Jun-2006
Complete List of Authors:	DAS, BISHNU; Loughborough University, Wolfson School of Mechanical & Manufacturing Engineering Young, R I M; Loughborough University, Wolfson School of Mech & Man Engineering Case, K; Loughborough University, Wolfson School of Mech & Man Engineering Rahimifard, S; Loughborough University, Wolfson School of Mech & Man Engineering Anumba, C; Loughborough University, Civil Engineering Bouchlaghem, N; Loughborough University, Civil Engineering Cutting Decelle, Anne-Francoise; Universite, Departement Organisation et Genie de la Production
Keywords:	INFORMATION SYSTEMS, SUPPLY CHAIN MANAGEMENT
Keywords (user):	Interoperability; Cross-discipline; Supply chain; Manufacturing; Construction; Communication-language.



For Peer Review Only

Towards the understanding of the requirements of a communication language to support process interoperability in cross-disciplinary supply chains

B. DAS^{*}, A. F. CUTTING-DECELLE^{*}, R. I. M. YOUNG^{*1}, K. CASE^{*}, S. RAHIMIFARD^{*}, C. J. ANUMBA^{*} and N. BOUCLAGHEM^{*}

^{*} Loughborough University, Loughborough, LE11 3TU, UK.

^{*} University of Evry / IUT-Dept Organisation and Production Management
EVRY, France.

Abstract. Many manufacturing organisations while doing business either directly or indirectly with other industrial sectors often encounter interoperability problems amongst software systems. This increases the business cost and reduces the efficiency. Research communities are exploring ways to reduce this cost. Incompatibility amongst the syntaxes and the semantics of the languages of application systems is the most common cause to this problem. The Process Specification Language (PSL), an ISO standard (18629), has the potential to overcome some of these difficulties by acting as a neutral communication language. The paper has therefore focused on exploring this aspect of the PSL within a cross-disciplinary supply chain environment.

The paper explores a specific a cross-disciplinary supply chain scenario in order to understand the mechanisms of communications within the system. Interoperability of processes supporting those communications are analysed against PSL. The paper proposes a strategy for sharing process information amongst the supply chain nodes using the “PSL 20 questions wizard” and concludes that although there is a need to develop more effective methods for mapping systems to PSL, it can still be seen as a powerful tool to aid the communications between processes in the supply chain. The paper uses a supply chain scenario which cuts across the construction and manufacturing business sectors in order to provide a breadth to the types of disciplines involved in communication.

Keywords: Interoperability; Cross-discipline; Supply chain; Manufacturing; Construction; Communication-language.

1. Introduction

The purpose of this paper is to explore from a system developers viewpoint, the requirements of a communication language, which would enable process interoperability in a cross-disciplinary supply chain system.

¹ Corresponding author: +44 1509 227662; Email: r.i.young@lboro.ac.uk

Many manufacturing organisations do business either directly or indirectly with other industrial sectors where interoperability problems are aggravated by different cultures and disciplines as well as by the broad range of suppliers and subcontractors having different business functions. This has prompted the authors to undertake this research study within a cross-disciplinary supply chain environment.

The term “interoperability” refers to the ability to share technical and business data, information and knowledge seamlessly across two or more software tools or application systems in an error free manner with minimum manual interventions (Ray and Jones, 2003). For example, a weekly production scheduling package would require customer order details, production capacity details of machinery at the shop floor and their availability, production process details etc. which would eventually generate weekly production plans as well as product delivery details for various customers. Customer order details data may come from Sales/Marketing, production capacity details may come from Manufacturing etc. Under normal circumstances for an interoperable system, the scheduling package should be able to capture those data seamlessly from the system and generate the necessary output. This does not usually happen, except where large integrated custom-made database application systems are used. In the worst case, the customer’s system cannot understand the output data relayed by the supplier as such, and cannot use this information in their system without re-inputting the data manually.

The ability to capture and share information seamlessly amongst a suite of software systems is very important as it reduces data handling errors, facilitates concurrent business activities and improves the responsiveness of an organisation. However, this feature is not always available amongst the commonly used software applications. This lack of interoperability is costly to many globally distributed industries (NIST, 1999) and this has encouraged the research community to explore ways to reduce this cost.

There are many reasons for this lack of interoperability: different software operating systems; different software development approaches; different high-level software languages for interfacing data/information, etc. The most common reason is due to incompatibility between the syntaxes of the languages and the semantics of the terms used by the languages of software application systems. This is mainly due to arbitrary definitions provided by users to the developers of the proposed systems. Therefore, there is a strong need for the development of an approach which would overcome these incompatibilities.

There are three principal approaches to handle these issues (NIST 1999, Cutting-Decelle et al. 2004). The first is a point-to-point customized solution, which can be achieved by contracting the services of systems integrators. This approach is expensive since each pair of systems needs a dedicated solution. A second approach, adopted in some large supply chains, obliges all supply chain partners to conform to a particular solution. This approach does not solve the interoperability problem since the first or sub-tier suppliers are forced to purchase and maintain multiple, redundant systems. The third approach involves neutral, open, published standards. By adopting open standards the combinatorial problems is reduced from n^2 to n , with bi-directional translators. Published standards also offer some stability in the representation they propose of the information models, an essential property for long-term data archiving.

The third approach appears to be promising suggesting development of a common shared communication language understandable to each participating software application. It is important to note that many standards approaches to integration and interoperability provide a syntactic standard but do not provide standard, interoperable, semantics. Some commercial approaches are starting to provide a level of semantic support for supply chain communication such as ebXML and RosettaNet (ebXML 2006, RosettaNet 2006). However, a critical issue is the level of rigour involved in the semantic definition. If interoperability is to be checked and confirmed by computer analysis, it is essential that sufficient mathematical rigour underlies the semantic definitions being used. It is for that reason that this paper has focused on the application of the Process Specification Language (PSL), which is based on first order logic, to the supply chain interoperability problem.

PSL (Schlenoff et al. 1996, ISO TC184/SC4 2001) has been developed to provide a common shared language to support process interoperability and is now an ISO standard, ISO 18629. PSL is formal and based on first-order logic, a mathematical set theory (Barwise and Etchemendy, 1999). The work reported in this paper has focused on exploring the potential of PSL as a shared communication language within the context of a cross-disciplinary supply chain. The particular aim of the work is to explore the use of PSL as a formal route to the comparison of potentially interoperable processes in order to identify their level of compatibility.

We have pursued a scenario study approach to developing and understanding the requirements of a cross-disciplinary supply chain system in terms of likely communication processes. This scenario is introduced in section two. Within the context of this scenario, the paper goes on to analyse a particular cross disciplinary transaction, the "Purchase order transaction", in order to explore the detailed functions, processes and tasks that may occur in the chain. Section three, then highlights the communication processes that take place through a set of forms across the chain by following the life cycle of the purchase order from start to finish.

The analysis of the "Purchase order transaction" scenario, clearly suggests that the processes within the system are the main routes to facilitating the communication across the supply chain, which occurs through physical movement of sets of forms containing embedded information generated by those processes. It is therefore logical as the next phase of the work to explore the characteristics of those processes in order to identify their interoperability potential. The PSL_20 questions wizard tool developed by NIST (2003), provides a mechanism for identifying the interoperability characteristics of processes. How this has been applied to our scenario is explained in section four. Section five then provides a general discussion of our work and the conclusions which can be drawn from it.

2. A cross-disciplinary supply chain scenario

This section introduces a cross-disciplinary supply chain scenario with a set of processes and information flows, which are required to support a typical set of business functions. The scenario encompasses a Construction company, a Construction site, a Manufacturer, a Retailer and a Transporter. The scenario is then used as the basis for the exploration of specific supply chain processes and the applicability of PSL to process interoperability analysis.

2.1 The supply chain nodes

Figure 1 shows the supply chain (SC) network as devised for this research study. The main actor of this network is the construction company who initiates most of the activities.

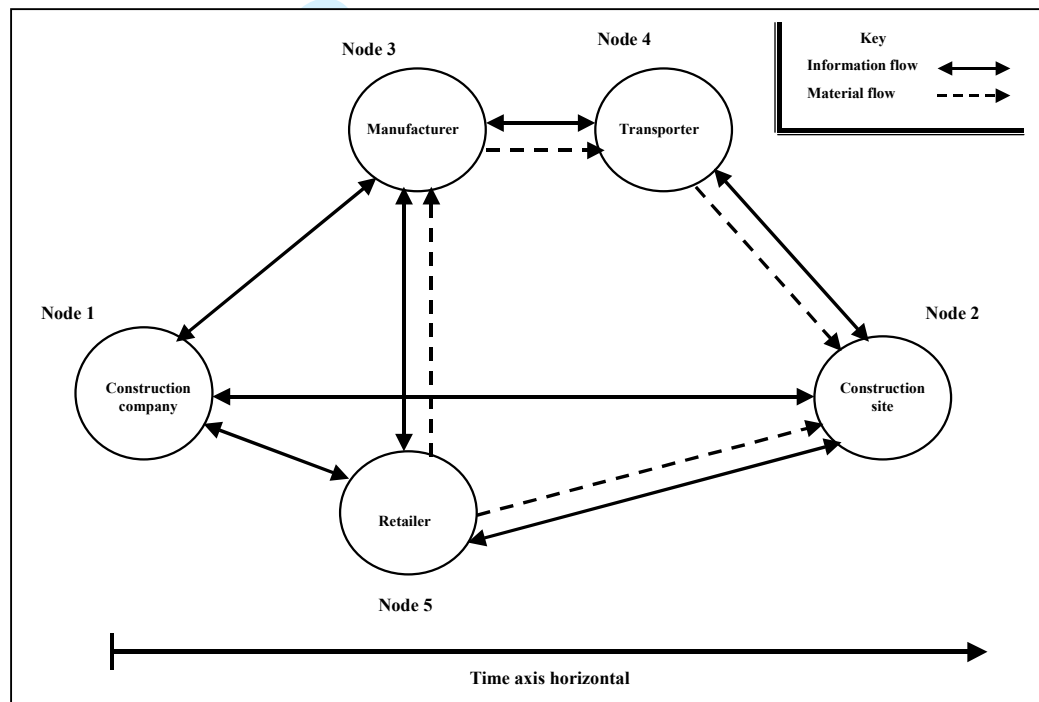


Figure 1: A typical cross-disciplinary supply chain network

2.2 Functions of the supply chain nodes

Various functions (tasks, activities, actions, processes, operations) normally performed by each node of the network within the context of processing a client's order by the construction company are briefly described below.

- **Construction company:** first receives an order from its client for implementing construction project. The Project manager in-charge of this project would ask its

various departments (see figure 2 for organisation structure) to design, cost-estimate and prepare a construction schedule for the project.

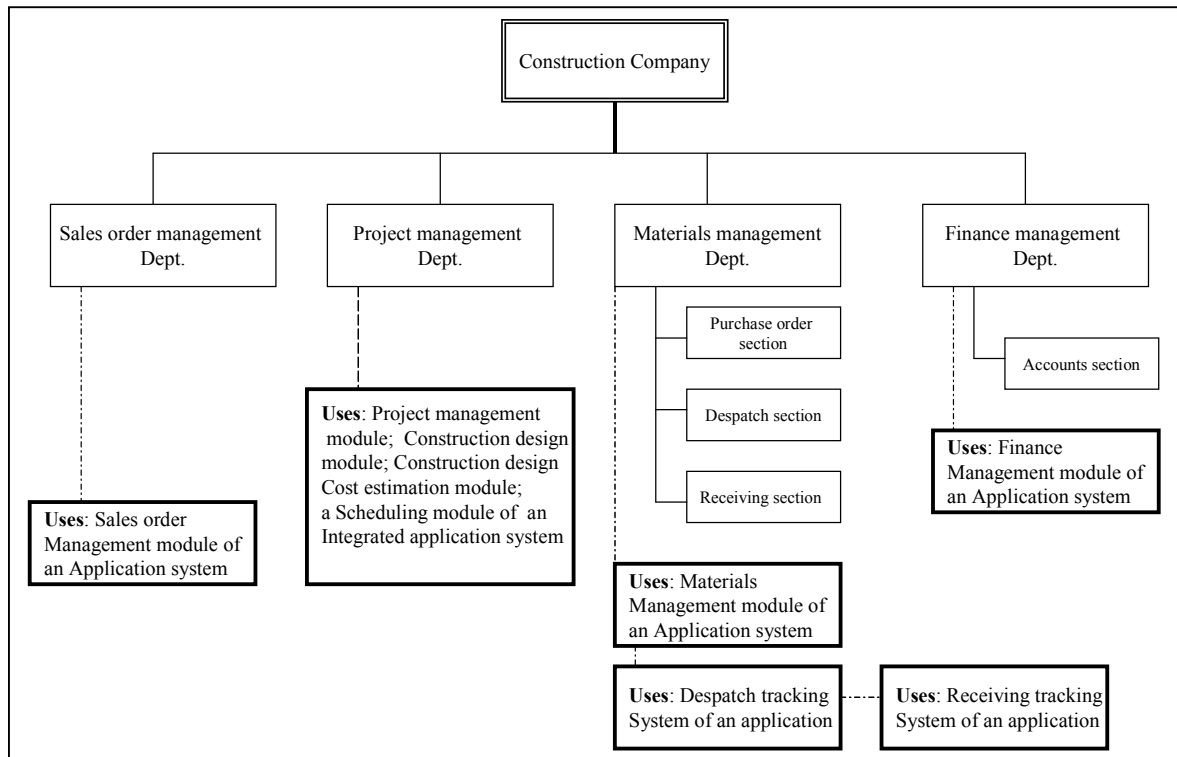


Figure 2: A typical organisational structure of the Construction Company

Based on the large list of items which need to be ordered for the construction schedule to be completed in time, the Project manager would place orders through the materials management department to the manufacturer and the retailer for the goods to be delivered at the construction site on a particular date. Once the orders have been accepted by the supplier and confirmed by the supplier, the Project manager would request its accounts department to prepare payment to the supplier for the set of goods he has ordered. However the payment will not be released until the Project manager has received confirmation of the delivery of goods at the construction site from the construction site manager and the satisfactory quality report from the Quality department. Many other functions take place within the company but they are out of the scope of this scenario.

The main functions involved within the construction company in this business scenario are: *Order processing, Project planning and management, Designing of building, Cost-estimating, Construction scheduling, Procuring, Quality management and Accounting.*

- **Manufacturer:** based on the order received from the construction company, the manufacturer would process the order; if feasible, the goods would be produced and shipped to customer's construction site via the transporter; accounts department prepares payments to suppliers and obtains remittance from the customer. The main functions involved within the manufacturer's business scenario are: **Order processing, Manufacturing, Procuring, Goods handling and Accounting.**
- **Transporter:** handles all the tasks relating to transport and delivery of finished goods to customer's selected location. It receives the shipment order from the manufacturer; collects the goods and delivers them at the construction site; accounts department would obtain remittance from the customer for the service. The main functions involved within the Transporter's business scenario are: **Order processing, Distribution planning and scheduling, Goods handling and Accounting.**
- **Retailer:** based on the order received from the customer (manufacturer and construction company) the retailer would analyse the order; if feasible to supply, goods would be delivered; if items are not in stock, the retailer places an order to a distributor and delivers them to the customer when they are received from the distributor; accounts department prepares payment to the distributor and releases it when goods are received and also obtains remittance from the customer for the goods supplied. The main functions involved within the Retailer's business scenario are: **Order processing, Distribution planning and scheduling, Procuring, Goods handling and Accounting.**
- **Construction site:** is a part of the construction company's business until it is handed over to the client. The construction site starts building work according to a project plan and the building work schedule, which would be prepared by the construction company at its head office. The construction site receives notification from the construction company, manufacturer, transporter and the retailer; construction site notifies appropriately the construction company, manufacturer, transporter and the retailer when goods are received; all damaged goods would be sent back to relevant supplier and undamaged goods would be released to the relevant users. The main functions involved at the construction site's business scenario are: **Project management and Goods handling including Quality management.**

It is essential to mention here that it is assumed that each actor uses separate workflow engines, as each represents an autonomous company, and there is therefore bound to be a problem with interoperability.

Detailed organization structure and function analysis of all the five nodes have been carried out within the scope of the project/research work. The information presented in this sub-section represents a brief overview.

The functions described here for each node are basically inter-business processes and it is essential to understand how they are related to each other within the node as well as across the supply chain nodes towards supporting process interoperation in cross-disciplinary supply chains. These issues are addressed in the next two sub-sections (2.3 and 2.4).

2.3 Processes to support the supply chain node functions

This sub-section introduces a set of processes, which are assumed to be needed to support the various functions of the supply chain node described in sub-section 2.2. For this research work, the term “process” means a structured collection of *activities/tasks* that have sequential relationships, while an activity/task means the transformation of a set of inputs into a set of outputs.

Table 1, shows the overall list of processes that may occur within the *Construction company* node to facilitate the functions described earlier. Many other processes occur within the same node, but they are not considered here.

Table 1: Overall list of processes within the Construction company node supporting the functions

Network node number and name	Function name	Process Name
1. Construction company	1. Order Processing	1. Receive client's order 2. Process client's order
	2. Procuring	3. Raise purchase order 4. Send purchase order 5. Receive acceptance of purchase order 6. Receive rejection of purchase order 7. Raise delivery instruction for supplier 8. Send delivery instruction to supplier 9. Send goods arrival dates to Construction site
	3. Accounting	10. Prepare payment to supplier 11. Receive notification from Construction site 12. Release payment to supplier 13. Notify payment to supplier 14. Close supplier account

Similarly four other sets of processes supporting the functions of the other nodes that is: manufacturer, transporter, retailer and construction site have been identified and developed within the scope of the project work.

2.4 Information flow across the supply chain nodes

An overall information flow diagram across the supply chain to support an order processing transaction (“Client’s order”) as initiated by the Construction company is shown in figure 3. Table 2 shows the overall list of information as used in figure 3. The numbering procedure used on the information flow line of figure 3 is as follows: first digit indicates the node number, which is followed by a dot and then the next two digits indicate information list number of that particular node.

The diagram shows the nature of interactions of information that may occur across the supply chain network. Some of this information would be *going-out* from the node and some will be *coming-into* the node from another external node. Therefore visualisation of the flow of information within and across the system boundaries are important particularly from the point of view of software development as well as in understanding the requirements for better communication/ interoperability etc. It would also highlight the variations of system requirements.

This brief introduction of the functions, processes and information flow across the supply chain nodes clearly indicates the nature and the complexity of the system and emphasize that a clear understanding of the system is essential for managing the supply chain efficiently (Cutting-Decelle et al. 2004, Young et al 2004, Cutting-Decelle et al. 2005).

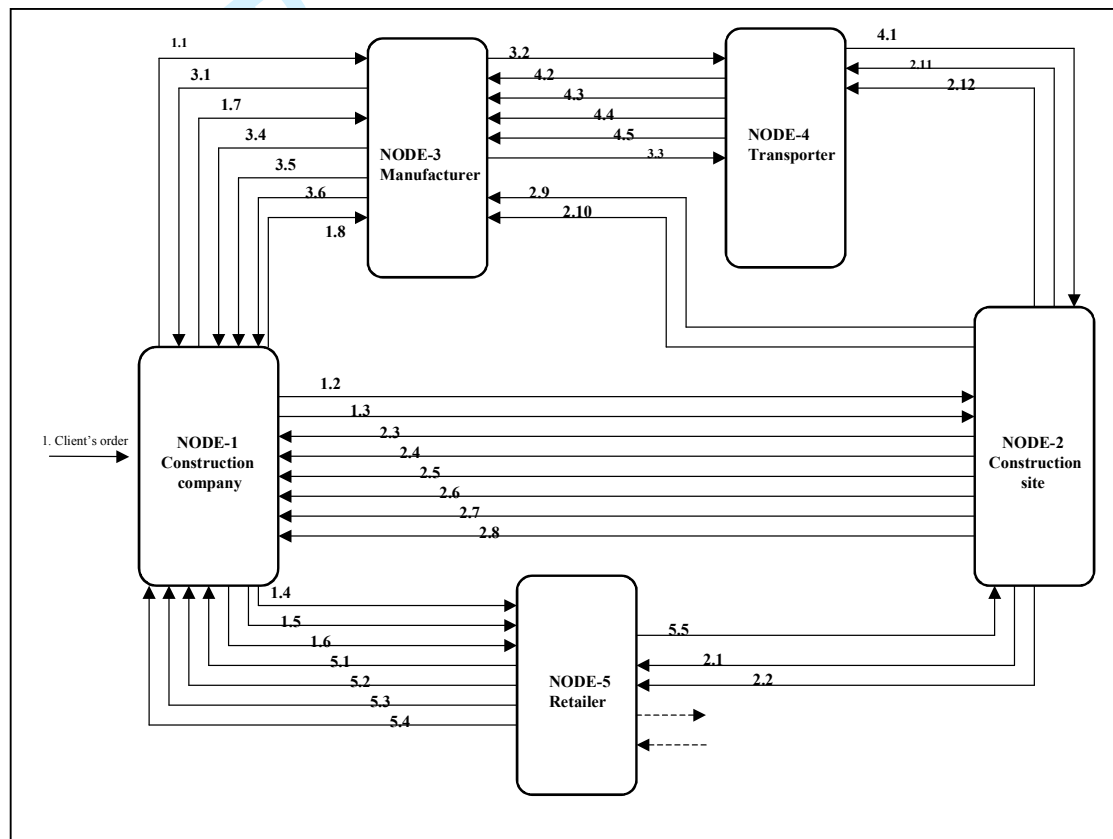


Figure 3: Overall information flow across the supply chain nodes for an order processing transaction

Table 2: Overall list of information as used in figure 3

1. Client's order	2.1 Const. site's GRN to retailer	3.1 Acceptance/rejection order from manufacturer
1.1 Purchase order to manufacturer	2.2 Const. site's QR on retailer's goods to retailer	3.2 Manufacturer's Shipment plan to transporter
1.2 MGAD to const. comp	2.3 Const. Site's Transporter's GRN to const. comp	3.3 Manufacturer's Payment notification to transporter
1.3 RGAD to const. Comp	2.4 Const. Site's QR on manufacturer's goods to const. comp	3.4 Manufacturer Shipment date to const. Comp
1.4 Purchase order to retailer	2.5 Const. Site's RP notification on manufacturer's goods	3.5 Manufacturer's GDN to const. Comp
1.5 Delivery instruction to retailer	2.6 Const. Site's Retailer's GRN to const. comp	3.6 Invoice statement to const. comp
1.6 Payment notification to retailer	2.7 Const. Site's QR on retailer's goods to const. comp	
1.7 Delivery instruction to manufacturer	2.8 Const. Site's RP notification on retailer's goods to const.comp	
1.8 Payment notification to manufacturer	2.9 Const. Site's QR on manufacturer's goods to manufacturer	
	2.10 Const. site's notification to manufacturer OMG	
	2.11 Const. Site's QR on manufacturer's goods to transporter	
	2.12 Const. site's notification to transporter OMG	
4.1 Transporter's Goods delivery (GD) date to const. Site		5.1 Acceptance of order from retailer to const. comp
4.2 Transporter's Order acceptance notification to manufacturer		5.2 Shipment date to const. comp
4.3 Transporter's Goods collection date to manufacturer		5.3 Delivery notification to const. comp
4.4 GDN to manufacturer		5.4 Invoice statement to const. comp
4.5 Transporter's Invoice statement to manufacturer		5.5 GD date from retailer to const. site

3. Purchase order transaction scenario

The paper now considers a "Purchase order transaction scenario" as a basis for the detailed exploration of the communication processes across the supply chain, the understanding of the related information structure, the exploration of process connectivity and also the applicability of the PSL_20 questions wizard. For this work, the term communication process means the flow of information from the sender to the receiver and vice versa in order to execute an activity/task. The propagation of such information takes place either electronically or by some other means such as postal systems etc. In this particular scenario as shown in figure 1, the Purchase department of the Construction Company of the supply chain places an order for supplying a set of doors to a manufacturer. Doors should be delivered to its construction site on a particular date. The order may contain the following information: "Supply 10 pieces of 6.5ft x 3.5ft (2inch thick) plane varnished ply-wood door complete with lock and handle at our Loughborough construction site on 10th October 2004". The manufacturer would supply the goods at the construction site through a transporter. The supply chain schematic as presented in figure 1 is still applicable for this particular scenario, although in this case the retailer is not involved. Similarly the functions and the processes as described in sub-sections 2.2 and 2.3 for each supply chain node to support the business of those nodes are also valid for this scenario.

3.1 The Life-cycle of a Purchase order Form

3.1.1 How Forms are generated

Whenever a business transaction (for example "Receive client's order" by the construction company) takes place in a supply chain node, then based on the functions of that node, a number of processes take place within the node. These processes eventually lead to generation of set of data such as: client's order details, client number, project order number, accounting code number, start date of the project, end date of the project, location of the project etc. Ultimately, depending on the business practice of that node, these data are collated either electronically or manually and a set

of information would be produced. Again these sets of information would be collated and eventually a required Form would be generated containing some of this information, for example a *project order form* that would be ready for the next process. The next process may simply be *send completed project order form* to project management department or to a sub-contractor. Similar scenarios would exist within other participating nodes of the supply chain. However, those participating nodes would not activate/initiate their appropriate functions and processes until they receive from the sender the appropriate information through such forms. Based on this scenario, it may therefore be said that all information flow between supply chain nodes are expected to be present on such forms. Figure 4, shows the mechanism of formation of a typical form through set of processes.

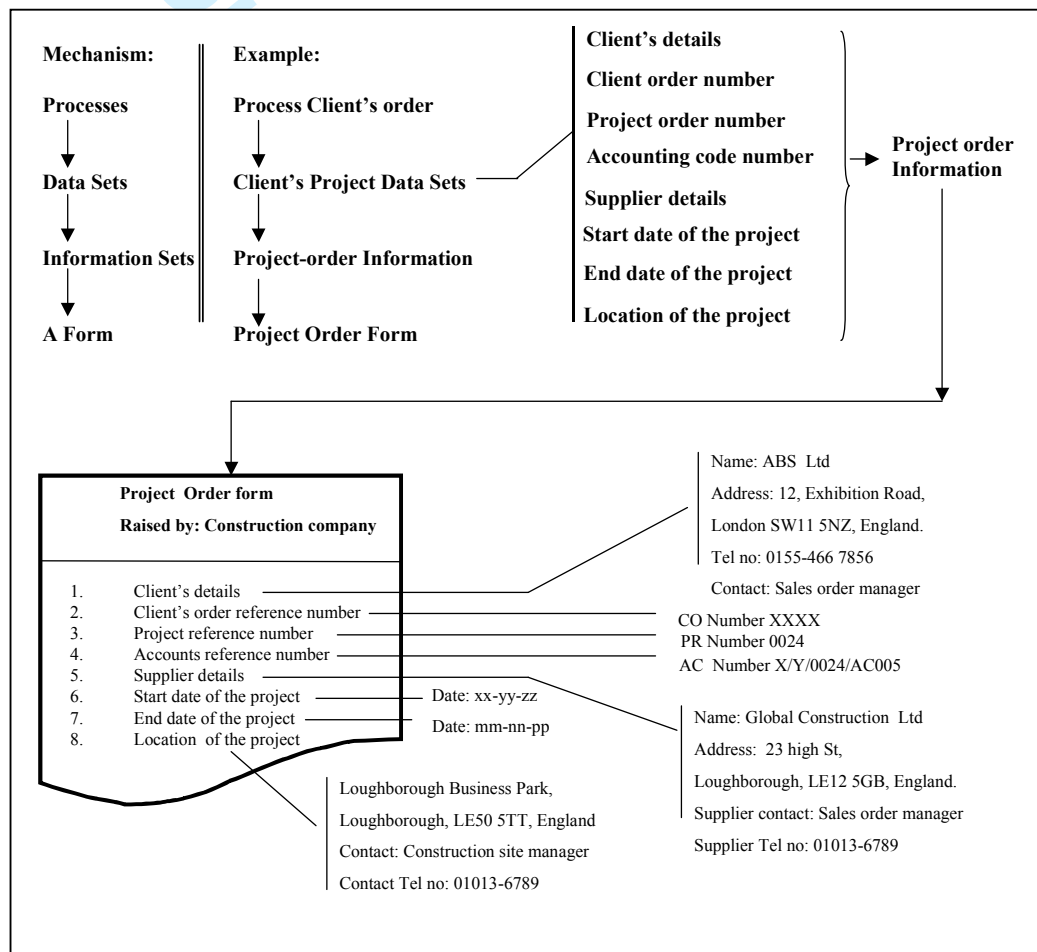
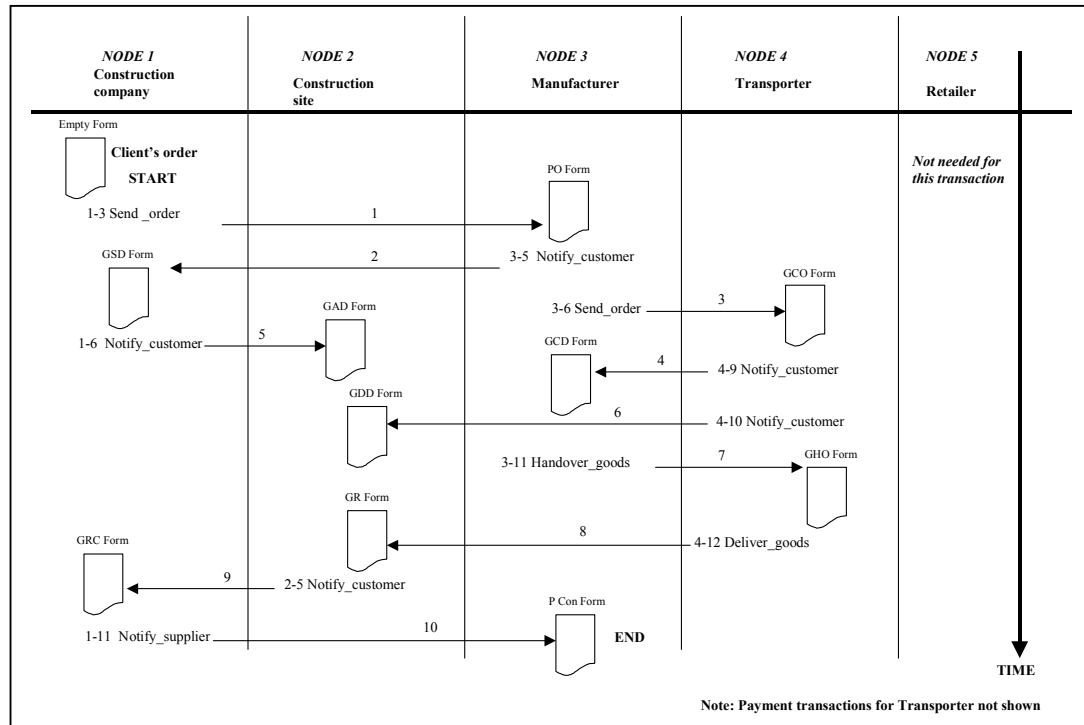


Figure 4: Mechanism of formation of a typical Form through set of processes

3.1.2 Lists of Forms relating to the purchase order transaction scenario

Based on this understanding, the set of forms supporting the “Purchase order transaction scenario” have been identified by following the “Life-cycle” (start-end) of a Purchase order Form as shown in diagram 5.



Abbreviations as used in above figure are:

PO form = Purchase order form;
 GCO form = Goods collection order form;
 GCD form = Goods collection date form;
 GHO form = Goods handover form;
 GRC form = Goods receipt confirmation form;
 P Con form = Payment confirmation form;

GSD form = Goods supply date form;
 GAD form = Goods arrival date form;
 GDD form = Goods delivery date form;
 GR form = Goods receipts form;

Figure 5: Life cycle of a Purchase order Form within the supply-chain as viewed from the Construction company node

This diagram represents the construction company’s point of view. It shows how the purchase order form progressed through the system across the participating supply chain nodes, from the moment it is sent out by the construction company to the relevant node (in this case the manufacturer), all the way until the ordered item relating to this form reaches the construction site through the transporter. The diagram also shows the other associated forms generated by various processes of other nodes needed to react to the information content of the purchase order form. The long solid lines with arrows and numbers on top of them indicate the sequence and direction of movement of these forms from node to node.

As explained earlier (see figure 4), these forms could be linked to the particular processes of the nodes of the supply chain involved in the transaction. They are also shown in this diagram. For example, in the diagram, the process “1-3 Send_order” relates to the purchase order form sent out by the construction company. Similarly, the process “3-5 Notify_customer” relates to “Goods supply date form”, sent out by the manufacturer to the construction company.

Table 3, shows the overall connectivity between the processes and the forms generated by the supply chain nodes to affect the “Purchase order transaction scenario”.

Table 3: Overall connectivity between the processes and the forms supporting the “Purchase order transaction scenario”

Process Id	Process	Name of the Form
1	1-3 Send purchase order to manufacturer	Purchase order form;
2	3-5 Notify goods supply date to customer	Goods supply date form;
3	3-6 Send goods collection order to transporter	Goods collection order form;
4	4-9 Notify goods collection date to customer	Goods collection date form;
5	1-6 Notify goods arrival instruction to site	Goods arrival date form;
6	4-10 Notify goods delivery date to site	Goods delivery date form;
7	3-11 Handover goods to transporter	Goods handover form;
8	4-12 Deliver goods to customer’s chosen site	Goods receipt form;
9	2-5 Notify receipt of undamaged goods to construction company	Goods receipt confirmation form
10	1-11 Notify payment to supplier	Payment confirmation form;

Note: The first digit of the process represents the supply chain node number, which is followed by a dash and the corresponding process list number of that node.

3.2 Information definitions of the Forms

In the previous sub-section, we have just discussed the list of forms that are generated by the “Purchase order transaction scenario” and their connectivity with the relevant processes of the node. However to share the information as represented in these Forms, we need to know the information content of the forms. For example, “*items on order*” information may contain data on: quantity, item name, item type, item size, accessories, special requirements etc. This would enable us to analyse which parts of the information content of the form are shareable and vice versa. The information definitions of these forms are derived from detailed analysis of functions, processes

and activities/tasks of the supply chain nodes relevant to the “Purchase order transaction scenario”.

Figure 6, shows the possible information of the “Purchase order Form (PO form)” relating to the “Purchase order transaction scenario”. It is sent out by the construction company to the manufacturer. The process linked to this form is: “1-3: Send purchase order to manufacturer”.

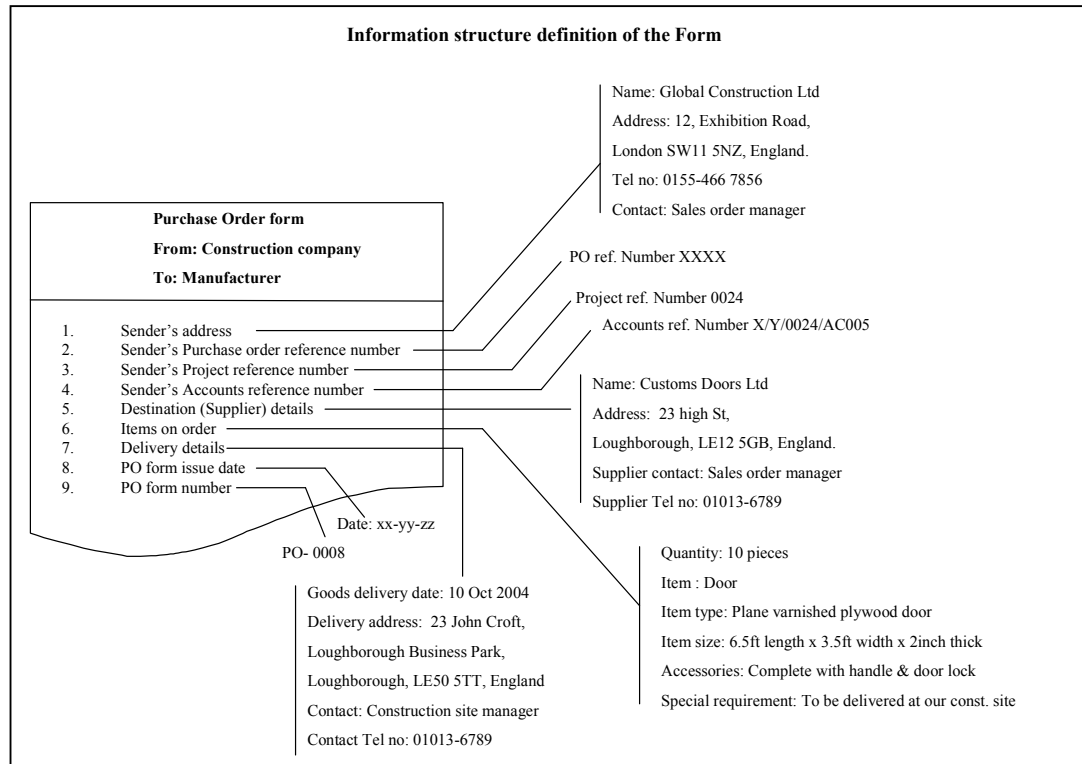


Figure 6: Typical information in a Purchase order Form

Similarly, information definitions of the other nine forms (see table 3) have been developed through detailed analysis of functions, processes and activities/tasks of the supply chain nodes relevant to the “Purchase order transaction scenario”.

3.3 Case for exploring interoperability of Processes

The analysis of the “Purchase order transaction scenario”, suggests that the processes within the system are the main sources in facilitating the communication across the supply chain. The next phase of the work explored the interoperability characteristics of those processes before attempting to develop mechanisms to share information amongst the nodes of the chain. This had been done through the use of the PSL_20 questions wizard tool as presented in the next section.

4. Application of PSL_20 question wizard to the understanding of process interoperability

This section introduces a short overview of PSL, the PSL_20 questions wizard tool and then explains its application to our scenario.

4.1 Concepts in PSL

PSL is based on a lexicon or a set of logical and non-logical symbols as well as a grammar i.e. a specification of how these symbols can be combined to make well-formed formulae. The underlying grammar used for PSL is based on the Knowledge Interchange Format, a formal language based on first-order logic developed for the exchange of knowledge among computer programs with disparate representations. This provides a rigorous foundation for the formal definition of the concepts in the language.

PSL provides an extensive set of some 350 concepts for process description. The key classes of concepts can be listed as:

- *Activity concepts* e.g. 'activity', 'activity_occurrence', 'object', 'occurrence_of', 'participates_in', 'primitive'
- *Time and state concepts* e.g. 'timepoint', 'between', 'exists_at'; 'state', 'holds', 'prior', 'trigger, launch
- *Ordering concepts* e.g. 'earlier', 'initial', 'precedes', 'successor'
- *Resource requirements concepts* e.g. 'resource', 'requires', 'resource_point', 'demand', 'agg_demand'

These provide comprehensive coverage for process relationships and include concepts for concurrency, preserving order and repetitive processes. It also provides concepts to capture state and/or time effects, either those which are pre-conditions for an activity or those which are the effects of an activity.

By formally defining supply chain communication processes using PSL concepts it should be possible to identify the extent to which two processes are the same. For example we could have two processes, both called "send", but can we tell if they mean the same? The following section describes a tool which enables PSL concepts called 'complex activities' to be analysed and compared

4.2 Overview of the PSL_20 questions wizard

The PSL_20 questions wizard has been developed by NIST as a tool facilitating the selection of the most appropriate PSL concepts corresponding to the set of 10 generic properties set-up for a process. These concepts are selected by the tool based on the activities inherent within the process and the answers chosen by the user to the questions proposed by the tool in a questionnaire. A screen shot of the PSL_20 questions wizard questionnaire is shown in figure 7.

The screenshot shows a web browser window with the URL <http://www.mel.nist.gov/psl/complex.html>. The page title is "20 Questions" and the breadcrumb trail is "home >> 20 questions >> complex activities". The main heading is "Classes of Complex Activities" with a text input field containing "Send_purchase_order".

Constraints on Branch Structure

- Relationships among activity occurrences that are elements of the same activity tree**
Are the activity occurrences isomorphic to permutations of each other?
 - The set of atomic subactivity occurrences in an activity occurrence is isomorphic to a permutation of the atomic subactivity occurrences in every other activity occurrence in the same activity tree.
 - The set of atomic subactivity occurrences in an activity occurrence is isomorphic to a permutation of the atomic subactivity occurrences in some other activity occurrence in the same activity tree.
 - The set of activity occurrences that are in the same activity tree is isomorphic to the union of occurrences that are permutations of other occurrences, and occurrences that are not permutations of any other occurrences.
 - None of the activity occurrences in the same activity tree are isomorphic to permutations of each other.
- Relationships among activity occurrences that are elements of the same activity tree**
Does there exist an activity occurrence whose atomic subactivity occurrences are the combination of the atomic subactivity occurrences in other activity occurrences in the same activity tree?
 - There exists an activity occurrence whose set of atomic subactivity occurrences is a combination of the atomic subactivity occurrence of every other activity occurrence in the same activity tree.
 - There exists multiple activity occurrences in the same activity tree whose set of atomic subactivity occurrences is a combination of the atomic subactivity occurrence in other activity occurrence in the same activity tree.
 - The set of activity occurrences that are in the same activity tree is the union of those whose atomic subactivity occurrences are part of a combination and those that are not combined with any

Left Sidebar:

- All About PSL
- PSL 2.1
- How Does It Work?
- FAQ
- PSL Tools & Articles
- 20 Questions
- Exchange
- Ontology
- PSL-EXPRESS
- PSL-XML
- More on PSL
- People
- Publications
- Related Projects
- Your Role in PSL

Footer:

Privacy/Security Notice/Disclaimer.
 NIST is an agency of the U.S. Commerce Department's Technology Administration.
 Technical inquiries: PSL Team
 Website comments: Webmaster
 Date created: 5/10/2003
 Last updated: 06/03/2003

Figure 7: First page of the questionnaire proposed by the PSL_20 questions wizard

This questionnaire has ten questions with multiple answers from which the user would choose/tick only one answer for each question. These questions have been set-up around three topics: Constraints on branch structure, Variation of branch structure and Distribution of complex activity occurrences. Question number 1 and 2 cover the first topic; Question number 3 – 6, cover the second topic and finally Question number 7 – 10, cover the third topic. Once the questionnaire has been completed and sent by the user to the system, a response is provided on the screen, as illustrated in figure 8.

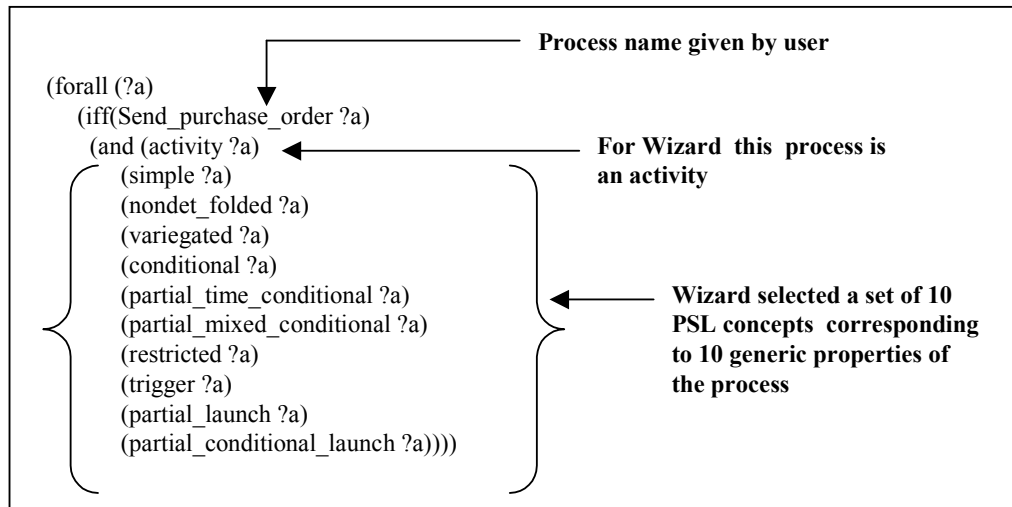


Figure 8: Answer proposed by the PSL_20 questions wizard for the process “Send_purchase_order”

Each of the ten PSL concepts chosen by the wizard from a set of concepts, directly links to one of the ten generic properties of the process. That is “Parent-children” relationship exists between the generic properties of a process and the corresponding PSL concepts. The list of these ten generic properties of a process is given in table 4 below.

Table 4: The ten generic properties of a process for which the wizard has been set-up

Property No	Generic properties of a process
1	Activity occurrences
2	Atomic sub-activity occurrences
3	Constraints on the set of minimal activity trees for the complex activity
4	Constraints on the set of minimal activity trees for the complex activity based only on the state
5	Constraints on the branch structure of occurrences of the complex activity based on the time
6	Constraints on the branch structure of occurrences of the complex activity based both on state and time
7	Constraints on the occurrence of the complex activity
8	Constraints on the occurrence of the complex activity based only on the state
9	Constraints on the occurrence of the complex activity based only on the time
10	Constraints on the occurrence of the complex activity based both on the state and the time

It must be emphasized here, that these concepts are rigorously/mathematically defined within the PSL (Knutilla et al. 1998, Schlenoff et al 1999, Gruninger, 2003) and hence there should be no scope for misunderstanding their meaning and using them incorrectly. For example, unless all the conditions embedded within the definition are

satisfied, the wizard would not select these concepts. The wizard will only select these concepts provided the user has selected the correct answer box from the multiple choice answer lists. The output as shown in figure 8 could be treated as a PSL translation definition of the process or the mapping of the process in terms of the PSL concepts. This mapping could later be analyzed to identify the interoperability issues amongst other processes of the system.

4.3 Application of the PSL_20 question wizard tool

4.3.1 Example

As an example, the PSL_20 question wizard has been applied to the process called "Send_purchase_order" of table 3. Figure 8 shows the response provided by the wizard. This translation definition highlights 10 properties of the process in terms of 10 PSL concepts such as: simple, nondet_folded, variegated, conditional etc. Repeating the PSL mapping method with other supply chain processes, allows us to identify where semantically similar processes exist. If the semantic translations of two processes are very similar, then there is likelihood that these two processes are shareable or interoperable.

Therefore based on this argument, the PSL_20 questions wizard has been applied to the set of 10 processes that are identified in our scenario (see Table 3) and the translation results are analyzed in the next subsection.

4.3.2 Evaluation of the set of processes as translated by the PSL_20 questions wizard

The analysis of the scenario and the list of 10 processes identified and shown in table 3 prompted us to classify these processes into four generic process types. These are:

SEND – formulated around activities involving monetary transactions.

NOTIFY – formulated around activities involving response by supplier to customer's requirements.

HANDOVER – formulated around handling activities of goods, which are finished, semi-finished or raw materials, and responded by supplier based on customer's requirements.

DELIVER – formulated around logistics/transporting activities of finished goods, semi-finished goods or raw materials and responded by supplier based on customer's requirements.

The ten specific processes have been allocated under the relevant generic process class as shown in the table 5.

Table 5: Allocation of processes of the scenario under the generic processes

Generic Process class	Processes of the scenario
SEND	Process 1: Send purchase order to manufacturer Process 3: Send goods collection order to transporter
NOTIFY	Process 2: Notify goods supply date to customer Process 4: Notify goods collection date to customer Process 5: Notify goods arrival instruction to site Process 6: Notify goods delivery date to site Process 9: Notify receipt of undamaged goods to construction company Process 10: Notify payment to supplier
HANDOVER	Process 7: Handover goods to transporter
DELIVER	Process 8: Deliver goods to customer's chosen site

We might reasonably expect that all the processes, which are allocated under any of these four generic processes as shown in table 5, should be semantically similar. Also, the processes belonging to different generic process class would not be semantically similar. This is not always the case, which will be evident from the following analysis. The translation results of the 10 processes in our scenario are analyzed for semantic similarities by comparing the 10 generic PSL properties of each process with the corresponding properties of the other processes. This leads to four groupings of these 10 processes as shown in tables 6.1 – 6.4 below.

Table 6.1: SEND processes against their PSL concepts

Property No	Process 1: Send purchase order to manufacturer	Process 3: Send goods collection order to transporter
1	Simple	Simple
2	Nondet_folded	Nondet_folded
3	Variegated	Variegated
4	Conditional	Conditional
5	Partial_time_conditional	Partial_time_conditional
6	Partial_mixed_conditional	Partial_mixed_conditional
7	Restricted	Restricted
8	Trigger	Trigger
9	Partial_launch	Partial_launch
10	Partial_conditional_launch	Partial_conditional_launch

Table 6.2: NOTIFY processes with same PSL concepts

Property No	Process 5: Notify goods arrival instruction to site	Process 6: Notify goods delivery date to site	Process 9: Notify receipt of undamaged goods to construction company
1	Partial_permuted	Partial_permuted	Partial_permuted
2	Partial_folded	Partial_folded	Partial_folded
3	Variegated	Variegated	Variegated
4	Partial_conditional	Partial_conditional	Partial_conditional
5	Partial_time_conditional	Partial_time_conditional	Partial_time_conditional
6	Partial_mixed_conditional	Partial_mixed_conditional	Partial_mixed_conditional
7	Restricted	Restricted	Restricted
8	Partial_trigger	Partial_trigger	Partial_trigger
9	Partial_launch	Partial_launch	Partial_launch
10	Partial_conditional_launch	Partial_conditional_launch	Partial_conditional_launch

Table 6.3: NOTIFY processes with partially different PSL concepts

Property No	Process 2: Notify goods supply date to customer	Process 4: Notify goods collection Date to customer	Process 10: Notify payment to supplier
1	Partial_permuted	Partial_permuted	Nondet_permuted
2	Partial_folded	Partial_folded	Partial_folded
3	Variegated	Variegated	Variegated
4	Conditional	Conditional	Conditional
5	Partial_time_conditional	Partial_time_conditional	Rigid_time_conditional
6	Partial_mixed_conditional	Mixed_conditional	Rigid_mixed_conditional
7	Restricted	Restricted	Local
8	Trigger	Partial_trigger	Trigger
9	Partial_launch	Partial_launch	Launch
10	Partial_conditional_launch	Partial_conditional_launch	Unconditional_launch

Table 6.4: HANDOVER & DELIVER processes against their PSL concepts

Property No	Process 7: (HANDOVER) Handover goods to transporter	Process 8: (DELIVER) Deliver goods to customer's chosen site
1	Partial_permuted	Partial_permuted
2	Partial_folded	Partial_folded
3	Variegated	Variegated
4	Conditional	Conditional
5	Partial_time_conditional	Partial_time_conditional
6	Mixed_conditional	Mixed_conditional
7	Restricted	Restricted
8	Trigger	Trigger
9	Partial_launch	Partial_launch
10	Partial_conditional_launch	Partial_conditional_launch

These groupings can be used to explore the potential for share-ability of the processes. For example, Table 6.1 above shows the translation results and the grouping of the Process 1 and the Process 3 under the generic process class “SEND”. Both processes are semantically similar in terms of the ten PSL concepts. We can conclude that there is a scope of sharing these two processes which is in line with our expectations. Similarly we can expect the “NOTIFY” processes in table 6.2 to be sharable.

However, when we analyze the “NOTIFY” processes in table 6.3 we find that these have a range of different PSL properties. For example if we examine Process 2 and Process 4 as presented in Table 6.3 it may be seen that these two processes are semantically different, because property numbers 6 and 8 of these two processes are different. Similarly if we consider Process 4 and the Process 10 of the same table, they are also semantically different, because their properties:1, 5, 6,7,8,9 and 10 are all different.

There are many reasons for these differences which can be better understood by further investigation. For example, if we look more closely at property number 8 for process numbers 4 and 10. For process 10, the value assigned to property 8 is “trigger”. The reason for allocating this property to this process is because this process is triggered once all the relevant information are received from all the 4 participating nodes of the supply chain. On the other hand, the “Partial_trigger” chosen by the PSL wizard for the Process number 4, is due to fact that it waits for information only from within the node itself, in this case the Transporter. In this manner, analytical reasons for selecting the properties of the process by the wizard can be developed.

Another interesting observation from table 6.4, is that all the 10 properties of the Process 7 and Process 8 are same, although they are allocated under different generic process class in this case – “HANDOVER” and “DELIVER”. In this case, although the names used are different the processes are semantically the same.

Analysis of this nature indicates the importance of properties of processes in terms of the PSL concepts and also suggests that although we can allocate several processes of

a scenario under one generic process term, it is not necessary that they will be semantically similar.

Again, evaluation of the set of processes as translated by PSL_20 question wizard eventually leads to overall groupings of processes as shown in table 7 below.

Table 7: Overall grouping of processes

Semantically similar	Semantically different
Process 1 and 3 Process 5, 6 and 9 Process 7 and 8	Process 2, 4 and 10

From this evaluation, we can say that the PSL tool, has enabled us to analyze and classify these 10 processes into semantically similar and non-similar categories as shown in table 7, which, we would not have been able to do by simply observing the contents of table 5. Analyses of this nature, leads us to an understanding of the transactions which have similar processes. This in turn provides a critical contribution to identifying the interoperability requirements between supply chain processes.

5 Discussion and conclusions

The identification of processes is very important for the development of software system. Many processes occur within supply chain nodes and the names of processes can be very similar across supply chain nodes. However, their requirements and function may be different, which creates the potential for interoperability problems amongst the nodes and leads to the cross-disciplinary communication problems in supply chain management.

This paper has shown an approach to the evaluation of processes in a supply chain in terms of the semantic compatibility. This has been achieved by identifying the nature of information flows, business nodes and processes. For a particular node, some information is generated within the organisation and others are received from external sources. Sometimes the names of information processes used by different nodes are the same but the information content may be different. This necessitates the formal analyse of information processes to identify where semantic similarities or differences exist.

We have shown in this paper that a communication language which can support process interoperation is required to have the following features:

1. sufficient mathematical rigour to enable computational comparisons of the meaning of terms
2. formal definition of process concepts as well as the more generally accepted entity information concepts

3. process concepts should include a broad range of constraints which may be applicable, such as sequencing, state, time and resource.

PSL, although developed for discrete manufacturing processes, appears to provide sufficiently generic and rigorous foundation ontology from which to compare the process semantics between supply chain processes. Although there is a need to develop more effective methods for mapping system specific terminology to PSL, it can still be seen as a powerful tool to aid the communication between processes in the supply chain. This is because of the combination of underlying mathematical rigour in the definition of its concepts with the comprehensive range of process definitions and constraints which it offers. The key problem appears to lie in the effective mapping from supply chain concepts to PSL concepts which requires a mapping from engineering expertise to the expertise of a logician.

The PSL wizard provides a current method available which aims to support the identification of mappings between engineering processes and PSL concepts. While this is very useful it does have some significant drawbacks in providing an effective and reliable route to achieving these mappings. Firstly, to answer the questions presented by the wizard, the user must have a clear grasp of set theory and logic, as the questions are written in terms which are linked to the mathematical theories on which PSL is based. This is inappropriate for people who are not completely familiar with these. A rewording of the wizard, with supporting explanations of the questions being asked would be helpful. This is particularly important as a slight misunderstanding of the question can lead to different experts providing different answers and hence nullifying the value of the mapping process. For the commercial use of PSL in the future, it is necessary to develop translation rules or guidelines to help the user to identify the appropriate concepts of the language. It is likely to be necessary to develop a range of wizards for different supply chain domains e.g. different wizards for manufacturing supply chains and tourism supply chains. There is also an issue in terms of the granularity with which a wizard will be able to effectively operate e.g. will a manufacturing supply chain wizard be able to support all type of manufacture or will different wizards be needed for automotive, electronic or food supply chains?

In summary, while there is great potential for improved interoperability in the approach taken, there is still substantial research needed before effective commercial solutions are available.

Acknowledgements :

The work reported in this paper has been conducted as part of the X-Slang project. The authors would like to acknowledge the support of the EPSRC which funded this project through Loughborough University's "Innovative Manufacturing and Construction Research Centre", under the Grant n° GR/N22526.

References:

- Barwise, J., and Etchemendy, J., 1999, The language of First-Order logic, 3rd edition, CSLI lecture notes.
- Cutting-Decelle, A. F., Young, R., Das, B., Anumba, C., Baldwin, A., and Bouchlaghem, N., 2004, A Multi-disciplinary Representation of the Supply Chain Information in Construction: An Innovative Approach to Project Management. Proceedings of the TMCE 2004, April 12-16, Lausanne, Switzerland.
- Cutting-Decelle, A. F., Das, B., Young, R. I. M., Case, K., Rahimifard, S., Anumba, C. J., and Bouchlaghem, N., 2005, A review of approaches to supply chain communications: From Manufacturing to Construction. Submitted to the IT-AEC Journal, UK.
- Cutting-Decelle, A. F., Pouchard, L. C., Das, B. P., Young, R. I., Michel, J. J., 2004, Utilising standards based approaches to information sharing and interoperability in manufacturing decision support. International Conference FAIM04, Toronto, Canada.
- ebXML., 2006, <http://www.ebxml.org/>
- Gruninger, M., 2003, A guide to the ontology of the Process Specification Language. In "Hand-book on Ontologies in Information Systems", R. Studer and S. Staab Eds, Springer-Verlag.
- ISO TC184/SC4/JWG8, ISO DIS 18629-1, 2001, Industrial automation system and integration – Process Specification Language: Part 1: Overview and basic principles.
- Knutilla, A., et al., 1998, Process Specification Language: An Analysis of Existing Representations. NISTIR 6133, National Institute of Standards and Technology, Gaithersburg, MD, US.
- NIST., 1999, Interoperability Cost Analysis of the U.S. Automotive Supply Chain (Planning Report #99-1). Available at <http://www.nist.gov/director/prog-ofc/report99-1.pdf>
- NIST., 2003, <http://www.mel.nist.gov/psl>
- Ray, S. R., and Jones, A. T., 2003, Manufacturing interoperability. Concurrent Engineering: Enhanced Interoperable System: Proceedings of the 10th ISPE International Conference, pp.535-540.
- RosettaNet, 2006, <http://www.rosettanet.org>
- Schlenoff, C., Kuntilla, A., and Ray, S. R., 1996, Unified Process Specification Language: Requirements for modelling process. Technical Report NISTIR 5910, National Institute of Standards and Technology, Gaithersburg, MD, US.
- Schlenoff, C., Gruninger, M., Ciocoiu, M., 1999, The Essence of the Process Specification Language. Transactions of the Society for Computer Simulation vol.16 no.4.
- Young, Bob., Guerra, David., Gunendran, George., Das, Bishnu., Cochran, Sean., and Cutting-Decelle, Anne-Francoise., 2004, Sharing Manufacturing Information and Knowledge in Design Decision Support. IDMMME2004, April 5-7, Bath, UK.