The product centric approach: a solution to supply network information management problems?

Mikko Kärkkäinen[*], Timo Ala-Risku, Kary Främling

Helsinki University of Technology, TAI Research Centre, P.O. Box 9555, FIN-02015 TKK, Finland

Abstract

The management of single-item level information has grown to be one of the greatest challenges of supply network management. The main reasons for this are the increasing product customisation and increasingly complex supply networks. Product customisation adds to the quantity and specificity of product-related information. Current mainstream solutions, based on integrating product information databases, are not suitable for complex, short-term supply networks. This paper presents an approach and a proposed system for managing single-item level information. The approach we call product centric information management is based on centralising information to the individual products. The proposed system uses software agents, peer-to-peer information sharing, and a coding mechanism that utilises the domain names of the Internet. We also review applications of the system in tracking and logistics control.
© 2003 Elsevier B.V. All rights reserved.

1. Introduction

The management of product related information, either product type or single item level, is one of the biggest challenges currently facing supply chain management [1–3]. For example, according to [4] an average of 30% of information in retailer systems is incorrect, and studies have shown that as much as 63% of product descriptions can diverge in supplier and wholesaler systems ([5], p. 37). These examples illustrate the magnitude of problems present in product type level information management. The problems to

be expected are even greater when discussing management of information at single item level [1]. The challenges are principally caused by two factors. First, the quantity of item level information is increasing, and second, the information has to be managed in supply networks of increasing complexity.

The amount and complexity of item related information is increasing due to a demand for a high number of product variants and customised products [6]. Also, stricter governmental requirements on product lifecycle management, traceability, and after-sales support, all demand information management at the level of an individual product. For example, in the European Union all consumer electronics have to be recycled, their minimum warranty time has been set to 2 years, with some member states demanding much longer warranty times [7]. Thus, companies are forced

[*] Corresponding author.
E-mail addresses: mikko.karkkainen@hut.fi (M. Kärkkäinen),
timo.ala-risku@hut.fi (T. Ala-Risku),
kary.framling@hut.fi (K. Främling).

to retain increasing amounts of data related to individual products, and also amend it as the product advances in the supply chain or is inspected or repaired after purchase. There is also increasing customer pressure for individual item level traceability in many product groups: from pharmaceuticals to grocery products [8]. All these requirements contribute to the amount and variety of product related information required to be retained, which complicates considerably the task of item level information management.

Supply networks are increasingly complex as multiple companies, as well as companies from different continents, participate in a product's delivery process. Moreover, the participating companies in the supply networks may not remain the same over the product life-cycle [9]. A growing tendency for outsourcing has led to a significant increase in the number of partners that participate in global supply networks. Furthermore, the companies that participate in the production of a product variant can change over time, depending on, for example, the fluctuations of currency rates or minor changes in the product design [10]. The increasing complexity and continuous change of supply networks present a considerable challenge for the management of product related information, as it is difficult to integrate the information systems of all the companies in the network.

The current practice for managing product related information is that individual companies store information that they create in company databases or paper-based filing systems. In such cases, retrieving the correct information may, in the worst case, require hours of work and the intervention of several people with various kinds of competence [8]. The challenge quickly grows even more severe, as the network of companies that produce components and related information grows in size and complexity. The increasing complexity of the networks leads to the dispersion of the information to increasingly many locations. The problem is intensified by the fact that companies rarely have common product codes for specific products or parts, neither on individual nor type level [11]. The increasing quantity and variety of the information to be managed makes the definition of standard transactions for exchanging the information difficult. Engineering changes and other autonomous actions by partners in the supply network induces even further complexity to information management. There is,

thus, a need for the development of more flexible information sharing approaches.

One solution for the problem of information dispersion would be the creation of a central product information database for the whole supply network. However, there are some major obstacles to doing so. First, companies in supply networks are independent entities, and it is therefore difficult to convince them to hand over their information to be managed by an external party [12]. Second, building a central database for the supply networks is a complicated task and integrating all the companies involved into the database consumes a lot of resources. This problem is multiplied by the flexible nature of the networks. Integration complexity is still increased by the difficulty of defining standard transactions due to the great variety of information to be exchanged. Third, most companies are members in several supply networks, which makes central supply network databases undesirable for them. For example, it is not feasible for a components supplier to integrate with the product information databases of several Original Equipment Manufacturers (OEMs).

Another possibility for collecting and sharing product related information is interacting directly with the products themselves, i.e. moving to "product centric" information management [13–16]. Research organisations and companies have developed approaches and systems where information can be directly linked to individual products. Most of these solutions are developed to operate at individual item level, and in this paper, we use 'item level information' to mean information relating to physically individual objects.

This paper presents a new approach, and a system utilising this approach for item related information management. Product centric information management and the system proposed in this paper were introduced on a more general level by Kärkkäinen et al. [16]. This paper adds to the work by discussing security related aspects and concentrating mainly on item level applications. It must be noted that the discussed approaches work just as well for product type level information management applications, which are further discussed in the concluding section of this paper.

In the first part of the paper, existing models and systems for product centric information management are reviewed. The second section presents our pro-

posed system, entitled Dialog system, being developed at Helsinki University of Technology (http://dialog.hut.fi/). Different application areas for the Dialog system are presented in the third part, and concluding remarks and areas for future research are reviewed in the final section.

2. Existing approaches for product centric information management

As discussed earlier, traditional methods, using company specific product information storage for managing product related information, are becoming insufficient for the task. This is because of the growing amount and variety of information to be managed, and the increasing complexity and flexibility of supply networks that participate in generating that information.

A proposed solution to this problem has been linking the necessary information directly to the objects [13–16]. We call this approach product centric information management. Many different approaches and systems have been developed for product centric information management. Development efforts are based on two fundamentally different approaches: developing proprietary solutions or developing standards for creating compatible solutions [17].

Two kinds of proprietary solutions have emerged so far, solutions developed by a single company to solve their problems, and proprietary solutions of service providers. A good example of a company's internal solution is the retention of maintenance data at a consumer electronics manufacturer. The company retains the maintenance related information of all its individual products in a centralised database, where the information is organised by identity codes of the items. The information concerning a certain product can be accessed using its identity code. In practice, the products are labelled with bar codes, which can be read to access the information. The main weaknesses of this kind of solutions are the necessity of transferring all needed information to a centralised database, and that the system is operable only after an integration period. This kind of approach is undesirable for an independent maintenance service provider, because in order to operate with the products of several OEMs the company has to integrate with the systems of all the OEMs. Solutions analogous to this example are the tracking solutions of express parcel carriers [18,19], and the tracking service that Savi Technologies offers to its customers [20]. All these systems use their own item coding, and are functional only with those codes and reading points that are connected to their systems.

Different service providers have also been emerging with the offer of linking products to information network addresses based on different coding schemes [21–23]. Usually these services utilise existing code base, most often the EAN 13 or UPC code. The strength of these systems is that they are ready-made solutions and can be taken into use quickly. One weakness of these kinds of systems is that the codes used do not always give the possibility to manage item level information, i.e. they work only at the product *type* level. This is because the codes used only distinguish different types of products, not individual pieces.

An additional weakness of these proprietary solutions is that a company gets the information only through the server of a particular service provider. This is a major disadvantage to the users of the system
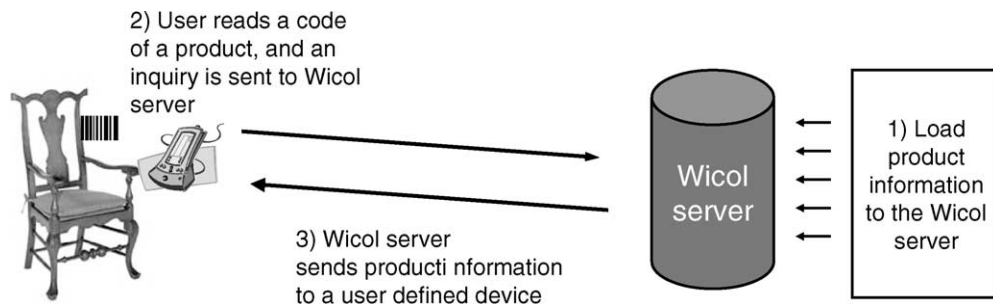

Fig. 1. The Wicol system (modified from [23]).

because, in order for the system to work, all companies in the supply network should use the same service provider. This is not feasible for companies belonging to several supply networks, and thus the solution is not scalable in an environment with complex, interlinked supply networks.

The functionality of a system created by Wicol Ltd., a company developing wireless information collector systems, is illustrated in Fig. 1. Most proprietary solutions are implemented according to this illustration.

The lack of universally readable item level coding and lack of scalability are the most severe problems with proprietary solutions. Standardisation could solve these two problems. Auto-ID Centre, a research community based jointly at Massachusetts Institute of Technology, the University of Cambridge, and the Adelaide University facilitates a large initiative developing standards and technology to enable product centric information management at item level [13,14]. It is developing a new standard code base for item level coding (the Electronic Product Code, ePC) and a method for linking item related information to the code and for transmitting it to the system that reads the code (the method is based on Object Naming Server (ONS) infrastructure). The system developed at the Auto-ID Centre is illustrated in Fig. 2.

The ePC is designed to be long enough to enable giving an unique identity to all products. Information related to individual products can then be stored in information networks and accessed with the code. An information network address is attached to each code in an object naming server (ONS) located in a predefined address on the Internet [13]. If the standardisation of the product coding is successful, and the ONS servers can be constructed with the necessary

functionality and scale, a global system will be available. This means that companies belonging to several supply networks can utilise the system with all their partners. Thus, the solution circumvents the biggest problems caused by proprietary solutions.

However, there are also some noteworthy challenges that Auto-ID Centre has to solve in order to be able to develop the global solution. Firstly, standardisation of product coding is difficult, and it is difficult to achieve a global acceptance of the coding. For example, it took over ten years for the EAN/UPC coding to achieve a strong foothold in the consumer goods industry and grocery business. The proposed solution is supposed to have an even wider appeal. Auto-ID Centre has been able to gather a significant base of companies to its network, which we think will enable the creation of the standards, but the process is not likely to be very fast.

Secondly, the approach demands central allocation of the codes. This means, that if a company wishes to start producing a new product variant, or even produce significantly more pieces of a product currently produced, it would have to request ePC codes to be allocated to those individual products. Therefore, much of the control of companies' operations is transferred to an institute responsible of the code allocation. Furthermore, the ONS network is not yet ready and the date of its possible completion is not known.

We propose an approach conceptually similar to that of Auto-ID Centre, but which does not require establishing a new global standard for product numbering, nor building a new naming server infrastructure. This approach is based on the use of existing standards for defining an alternative way of ensuring global uniqueness of the codes used. The technical solution for accessing item related information is
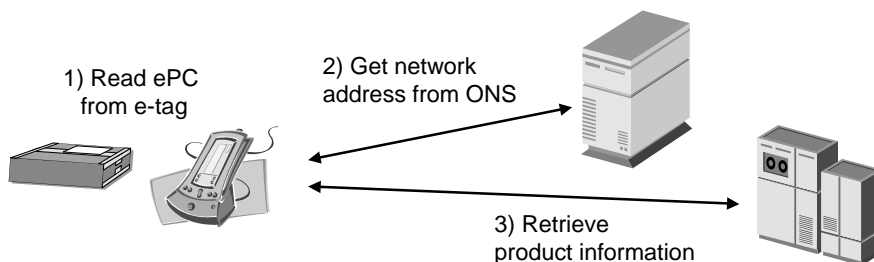

Fig. 2. The system developed at the Auto-ID Centre.

based on software agents that share information in a peer-to-peer fashion. Therefore, our solution can provide an immediately usable platform for practical applications. This means that companies can start implementing product centric information management immediately, rather than wait for solutions to challenges associated with coding standardisation and setting up object naming servers.

3. The Dialog system

The Dialog system aims at solving the challenges of item level information management without the need of developing new coding standards. In this section, the Dialog system will be presented in the following order: first, we will present the principles of the coding scheme utilised in the Dialog system, then we will proceed to review the basic technology used in developing the software agents for the system, and finally we will describe the connection and information exchange between these agents.

3.1. Coding scheme of the Dialog system

The Dialog system does not require the development of new coding standards, because it utilises the unique Internet domain names of companies to create globally unique codes for products and consignments. In our approach, the connection between a tangible object and the information network address that contains information regarding the object is defined by two pieces of information:

- Identification part (string containing numbers and/ or text of free choice).
- Uniform Resource Identifier (URI), which is the Internet address of an agent associated with the tangible object.

These two pieces of information guarantee that the resulting combination is globally unique as long as the identification part is unique at the given URI address. The organisation that owns the URI can arrange this by carefully allocating the identities inside that URI. The difference of our approach to that of the Auto-ID Centre is that our approach uses an existing method of providing physically unique objects with unique identity, namely the domain naming service. There is, therefore, no need to build a global base of codes for the products to be identified, nor is there any need for the codes to be centrally allocated. The principle of the Dialog system is illustrated in Fig. 3.

The identification part may, in principle, be of any format whatsoever. However, if RFID tags are used it is most convenient to use the fixed identification number of the tag. RFID tags normally have a globally unique identification number, such as the 64-bit code defined by the ISO standard for RFID tags working at 13.56 MHz. This code can then be linked to the internal references of the company. When operating with bar codes, it is often easiest to use identity coding already in use in the company owning the URI. Examples of this kind of codes are dispatch note references, order numbers, or combined product type and serial numbers.

One weakness of the coding scheme we propose is that it uses two separate pieces of information in order to build the code. However, the two codes can be read in a single operation using automatic reading devices. If using RFID, it is simplest to use the tag's own identity as the ID part. The tag id and the programmable data area, where the URI part can be stored, are easily retrieved in one single read operation. When using bar codes, the identification and the URI can either be coded as two separate bar codes or coded into one single bar code. For coding the two parts into one field, a predefined separator between the codes has to be used. The Dialog project proposes using a coding convention similar to e-mail addresses, i.e.
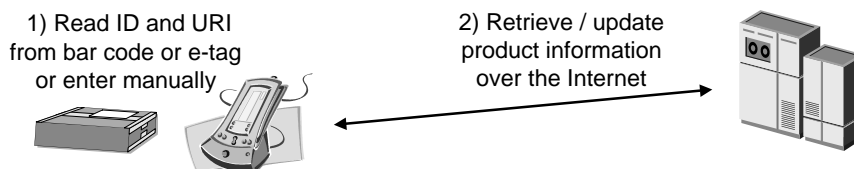

Fig. 3. The Dialog system.

identification@URI (for instance 12345@dialog.hut.fi) [24]. These ways of action enable circumventing the need for attaching and reading two separate codes.

3.2. Dialog agents

The URI part of the Dialog item code indicates the location of the tangible object's "agent". The agent is a background service running at the computer indicated by the URI. It offers various interfaces for functionalities like location updates, item information requests, maintenance information requests, etc. Each interface has its own characteristics in terms of the information to exchange, restrictions on data security, authentication, authorisation, etc. Therefore, security considerations can be treated in different ways depending on how "dangerous" the service provided by each interface is for the Dialog system itself and for the information systems of companies using the Dialog system. The current versions of Dialog agents are programmed in Java, and they provide interfaces for (1) receiving location updates, (2) for linking the Dialog identification part to internal company reference numbers and (3) for retrieving and displaying item related information.

As all URI addresses, the one used in the Dialog system can also contain other information concerning the agent service, namely a protocol specification, desired port number and a directory tree. The company that has issued the tangible object is normally the owner of the URI it is carrying, so the protocol (RMI or other), port, etc. to be used can be decided by the company without preliminary agreements with other actors. Therefore, issues like data security and firewall management remain under the control of the companies themselves instead of being imposed by third-party software companies.

In order to connect to existing information systems, the agents use Java Database Connectivity (JDBC) [25], which offers a standardised interface to most existing database products using the standardised query language (SQL). The Dialog system does not require modification of existing information management systems, since it creates its own data structures and links to existing company data when installed. In the event that similar data structures already exist in the company system, there is a possibility to parameterise database, table and field names so that the Dialog

system can use them directly. However, existing systems will normally need some new functionality so that they forward and display the new information to the users of the system.

The strength of Dialog agents is their lean composition. They can be downloaded from the Internet and have a low installation overhead, which is important for scalability. A typical installation of current software components take less than 10 min, including configurations for database and reading device connections. Dialog agents exchange information in a peer-to-peer fashion, which also increases scalability, and the degree of scalability should be good enough to allow companies of any size to use them. The system also supports direct data exchange by hand-held devices.

3.3. Dialog connection protocols

In the Dialog system, connections between a Dialog agent accessing information related to a tangible object, and the Dialog "tangible object" agent managing the information, are made in a peer-to-peer[1] fashion over the Internet. The Dialog system utilises methods of distributed information management, where information is transmitted directly between the place where it is needed and the place where it is stored. In a peer-to-peer system, the information is usually stored by the party that has created it, so unnecessary copies of information are not made to the companies in the supply network or to intermediate databases operated by third-party companies.

Furthermore, Dialog always opens up a bi-directional communication between the agents exchanging information. Bi-directional information exchange is needed in a variety of situations. For example, when maintenance operations are carried out, it is important that the maintenance workers receive the necessary information to perform the work, and that their operations are recorded in the data of that specific product.

In Dialog, the requirements of real-time, bi-directional information exchange are dealt with using synchronised, distributed programming technologies.

---

[1] Peer-to-peer computing is not a very well defined concept, but its main idea is to use direct connections between computing devices of any size, thus avoiding the use of intermediate server machines and buffering techniques.

The current versions of Dialog agents are programmed in Java and use the RMI protocol [26]. Java and RMI have been selected mainly because they make it possible to develop lightweight software components, which give little network overhead and are easy to implement and install.

However, the system supports heavy traffic. Tests performed with the current software components indicate that the scalability of the system is sufficient. A low-range computer (Pentium 233 MHz, 80 Mbytes RAM, running Linux with Kernel version 2.2.14 and the MySQL database) was capable of handling 100 simultaneous threads making 10 sequential information requests in 76 s. A typically configured server machine is likely to be able to handle requests at least tens of times quicker, so in most cases system delays should not become greater than those caused by the Internet connections themselves. Computers hosting very heavily used services can currently receive hundreds or even thousands of information requests per second. In cases where the number of connections grows too big, requests can be distributed internally either by using several URI addresses for different products or by normal load balancing performed in server clusters.

In the future, alternative technologies may also be supported. Examples of such technologies are Corba [27] and SOAP [28]. Corba is programming language-independent and of comparable rapidity as RMI, but not as straightforward to use as RMI. SOAP is XML-based and is also programming language-independent, but it is considerably slower than RMI (about five times according to tests performed) and gives more complicated installation procedures. Capabilities to handling other Java-specific technologies like Jini [29] and JXTA [30] are also considered. Jini allows for dynamic service lookup, which is necessary for agents that actively need to discover services, like transportation or assembly services for instance. Communication based on established information exchange standards, at least XML, and standards using XML like RosettaNet and ebXML, might also offer interesting solutions. The choice of technology mainly depends on (1) if several programming languages need to be supported, (2) if open XML standards are available for the communication, (3) if communication needs to be bi-directional in real time or not and (4) performance issues.

3.4. Strengths and limitations of the Dialog system

The Dialog system is designed to be quick to take into use, scalable, and flexible enough to be used in many different problem areas. Also, being a peer-to-peer system, Dialog facilitates equality between participating companies.

Dialog can be taken into use quickly, because there is no need for the standardisation of the coding used to connect tangible items to their agents. There is, also, no need to individually establish connections between agents, as agents are able to discuss over the Internet when the URI is obtained. This increases the scalability of the system, as does the fact that Dialog agents are rapid to install. For example, for installing a checkpoint agent for a tracking application it is enough to download an archive file from the internet, decompress it to a new directory and start it by double-clicking the start-up script (agents can be obtained from http://dialog.hut.fi/).

As the item identification and coding is founded on existing standards and the installation of the software is rather effortless, the Dialog system can be implemented incrementally, starting with applications that benefit most from product centric information management. Thus, the system avoids one of the biggest challenges for network systems: the system does not require a large user-base in order to be economically feasible and create value for the users [17]. To further alleviate the implementation, the Dialog system is intended to be as open as possible. Therefore, project results, software specifications and demonstration software are made publicly accessible at the address http://dialog.hut.fi/.

Peer-to-peer systems provide more equality between the companies participating in the network. This is because all parties participate on equal terms, no matter what their size is. The biggest companies often dictate information exchange standards and centralise the information in their systems in current collaboration networks, but with peer-to-peer communication this is not the case. Every company retains the ownership of the information they have created, but may exchange the information in a standardised fashion.

A significant limitation in implementing the Dialog system is that it encompasses only the information exchange between network participants. The content and syntax of the information must be defined

separately, which may be a rather demanding task, depending on the network of companies and planned use of the system. The efforts in standardisation of information exchange syntax, e.g. the Product Mark-up Language of Auto-ID Centre and XML-based standardisation may enormously ease the implementation of certain applications.

The peer-to-peer inspired principles used in Dialog partially help to improve data security since data does not need to be copied between companies and/or passed by third-party companies. However, due to the growing number of intrusions into protected systems and virus attacks, data security and firewalls have become frequent issues. Firewalls are a potential problem for the information exchange between different systems, since they may block communication. This is not a major problem for the Dialog system than it is for other data communication methods. The current implementation that uses RMI requires opening ports in the firewall, which is the reason Dialog agents should normally be installed on computers in the Demilitarised Zone (DMZ). The DMZ is outside the company Intranet, but partially protected from the public Internet. Dialog agents can also use direct XML or SOAP exchange, which makes it possible to use standard web ports and thus avoids opening new port numbers in the firewall. However, it is far from obvious which solution (RMI or direct XML) is safer in reality since security normally depends more on the services themselves than on the number of open ports.

Data exchange itself is secured by using existing authentication and encryption technologies. The main issue for safe data exchange is how to authenticate trusted parties. This can be done either by directly exchanging authentication keys and storing them locally for every Dialog agent as in [31]. In bigger collaboration networks, it is more appropriate to use the existing Public Key Infrastructure (PKI), where third-party certification authorities manage and certify the authenticity of public keys as described in [32].

Server or network down-time are also important issues, especially in tracking applications, where no location update should be allowed to disappear. This is why message persistence [33] is implemented for all applications requiring it, to guarantee that no messages are lost due to network or server downtime, so they can always be retrieved and sent when the system is operational again.

4. Applications of the Dialog system

In this section, we present the first two application areas of the Dialog system. To-date, the system has been piloted on tracking deliveries for international investment projects. It is used for the tracking of transport units, and the system works with both bar codes and RFID tags. A heavy industry company providing components for international investment projects will take the tracking functionality into operational use in the fall 2002. Also, the extension to the system that enables delivering transportation-related information between the consignors and consignees without the need for traditional systems integration was tested in spring 2002. This enables decreasing the costs related to automation of data exchange in transportation. These applications are briefly presented in this section. Both of the applications work on an individual item level, as logistics execution calls for control of individual parcels, and is less concerned with the product types transported.

4.1. Forwarder independent tracking

Customers are increasingly demanding transportation service providers to provide them with information concerning the whereabouts of their goods in transit. Consequently, many companies have invested tens of millions of dollars in developing tracking systems to keep track the location of consignments [18–20]. However, these systems have two basic weaknesses. First, they work only when the goods are handled inside the company's handling system, and second, the information is usually only visible on a www-page, and delays are not updated in the recipient's information system unless time consuming integration efforts are carried out.

Tracking is especially important in project deliveries. This is due to the deliveries of project components are always time specific, delays in their delivery may, at the worst, hinder the whole project if the knowledge of the delay is not obtained pro-actively [15]. Furthermore, the supply networks of investment projects are often so large and complex that it is difficult to use only one forwarder to deliver all the components for the project. The first application area of the Dialog system is in forwarder-independent tracking of investment project deliveries. It is also

an interesting application from a systems point of view, since the location of the tangible object is a basic requirement for many other services enabled by the Dialog agents. Such services could include informing the construction site of delivery delays, intentional delay of deliveries due to schedule changes on the construction site, and changing the transportation mode to speed up a delivery [15].

The tracking application has been demonstrated to industrial project partners in November 2001. In April 2002, the system was used in its first pilot implementation. This pilot project involved a producer of heavy machinery, which was the company where tracking information was updated and who owned the URI address programmed into the RFID tags attached to the shipments. Two checkpoints were installed along the delivery chain. The first checkpoint was the packaging department situated in the same country as the heavy machinery producer but located in a different town at a distance of hundreds of kilometres. The second checkpoint was at the destination (construction) site situated in another country. The aim of this pilot was to ascertain the functionality of the system, and to detect the practical problems associated with forwarder-independent tracking.

The codes were programmed into RFID tags, because bar codes can lose their readability in the harsh environment of the project delivery chain. Since RFID tags had been used, their pre-programmed serial numbers were used as the identification part of the product code. The checkpoints communicate the location of deliveries directly to the producers system every time an RFID tag is read.

Conclusions from the pilot project were very positive, taking into consideration that it was the first installation in an industrial environment. In addition to the default system set-up time, firewall configuration and database protocol issues introduced extra installation delay of a couple of days, but these delays were expected beforehand. Main validated features of the pilot installation are that firewalls are usually not a problem at least at the checkpoint end and that the system operates well even through modem Internet connections. Once all three software components were running, the tracking system has been working perfectly and no problems were encountered during the trial that lasted for a month. The successful pilot implementation convinced the case company to take the system to operative use starting from fall 2002.

4.2. Product centric logistics information management

Another investigated application area for the Dialog system is using it to transmit logistics information, such as dispatch notes or packing lists, related to individual deliveries. More precisely our research has focused in the use of the product centric approach in merge-in-transit distribution, which presents some of the largest challenges in logistics information management.

Merge-in-transit is a distribution model, where shipments originating at different dispatch departments are consolidated into a single customer delivery within the delivery chain [34–36]. This enables fulfilling a customer order with only one drop-off, without the need for warehousing all the goods in one location. The benefits obtainable with the merge-in-transit model include improved customer service, reduced inventory, reduced transportation costs, and improved supply chain transparency [34]. For example, the distribution model can save up to 20% of distribution costs of a maintenance, repair and operating (MRO) goods wholesaler [37].

Merge-in-transit applications are usually built between a few large companies, because managing the information flows associated with merge-in-transit distribution demands very effective information systems integration [36,38]. The costs of the integration efforts often prevent smaller companies from participating in merge-in-transit distribution. Thus, larger-scale operations with a high number of suppliers, for example in a wholesaler distribution chain, have yet to address the presented challenges.

In order to make the implementation of a large-scale merge-in-transit system feasible, the delivery information needs to be transferred in electronic form. EDI is a proven concept for transferring such logistics messages, but it is not adopted in smaller companies due to high implementation costs [39]. It also requires one-to-one connections to be formed between all parties needing to exchange data, which makes it not feasible for large supply networks. The prices of EDI-integration between business applications range from a few hundred euros to tens of thousands

[40–42], depending on the applications to be integrated. Also, when using the batch-oriented EDI, there are severe problems in matching the received information to the physical delivery, i.e. in synchronising material and information flows, in situations where many small deliveries are processed [43].

The Dialog system offers a low-cost alternative for enabling a common platform of information exchange and management for multiple companies, and simultaneously ensures that material and information flows are synchronous. Shipment identification can be done with the Dialog coding scheme, and the required delivery data can be passed over the Internet between the supplier's and handling facility's agents with the Dialog system. The delivery data include the information needed for the handling of the parcel, namely the customer order number that the parcel belongs to and which parcels it is to be merged with, as well as the information included in normal shipping notes. Using the information received via Dialog, the warehouse management system can automatically instruct operators on where to transfer the parcel to wait for consolidation.

The implementation costs of the Dialog system are likely to be much lower than those of EDI-messaging. No connection needs to be created between the parties, as the identities of the parcels provide all the information needed to create the connection on the Internet. The only implementation costs of the Dialog system are the costs related to installing the software agent and linking it to the database of the business application.

This makes a merge-in-transit network operating with the Dialog system also more scalable than one based on one-to-one connections. A new member, for example an additional supplier for a wholesaler, can be added simply by installing an agent at her server. The existing members need not make any changes in their application or create any connections with the new member.

The Dialog system was tested in a limited pilot installation with a MRO wholesaler and a logistics service provider that practise merge-in-transit distribution. In the pilot, a client agent was installed on a computer of the logistics service provider, and delivery documents were retrieved from a remote server over the Internet. The installation (including the Java Runtime Environment) took seven minutes, after which the agent was operational.

The representatives of the wholesaler and the logistics service provider saw the following to be the most promising benefits achievable with the Dialog system: (1) removal of costs with EDI transactions, (2) facilitation of electronic communications, especially the opportunity for automated communications with smaller partner companies not capable of implementing an EDI-connection, (3) availability of up-to-date information everywhere in the distribution network.

5. Discussion

Product centric information management is an interesting option for managing product related information. In this paper, we have presented a new system for product centric information management, specifically the Dialog system being built at the Helsinki University of Technology. The basic functionalities of the Dialog system are operational, and applications using it are already being developed. This rapid implementation is possible because the product coding used is based on existing coding standards and therefore need no new central authority to ensure uniqueness and scalability.

The leanness and simplicity of installation of the software components is also important for rapid implementation in the logistics network. This relative maturity signifies that a natural next step is to proceed to extensive piloting and field-testing and simultaneously to offer the Dialog system to standardisation bodies.

With respect to application, two distinct areas are targeted by our future research efforts. First, we are studying how the product centric system can alter supply chain management, as a majority of the information needed to provide value-adding services to products can be communicated via the products themselves. Second, we are trying to find out what information should be managed and exchanged with the system. More specifically we are trying to address the question: "What information is important to different supply network members and how the product centric approach could help in managing that information?"

Even though the Dialog system is currently focusing mainly on supply chain management related issues, dealing with information management at the level of physically unique items, it is not in any way

limited to it. The proposed architecture should also be usable in for example in the finding of information services for tangible objects and information management at product type level. Actually most single object level applications have a product type level extension. A good example of such an application is using product centric information management for the support of maintenance service provision. There it is important to keep a log of the maintenance and operating history of that specific product, but equally important would be to give general maintenance instructions for that product type. Also, information of general breakage of the product type could help the maintenance person to speed up operations.

As previously discussed there are severe problems in the management of product type level information. These problems are well illustrated in the problems associated with developing and maintaining electronic catalogues. Electronic catalogues were not considered a problematic issue at all in the eve of e-business. It was considered clear that much of the inefficiencies in procurement and sales could be eliminated by business-to-business e-business applications [44–46]. Later companies started to notice that well managed electronic catalogues actually represent much that is expected of e-commerce, and the value of the electronic catalogue comes from the fact that it is easier to keep up to date [47,48]. Finally, it has been realised that the building and even more importantly the maintaining of electronic catalogues is onerous and expensive [49,50]. Current efforts to solve the problem include building an industry specific standard numbering database that would contain product information of all suppliers, and the suppliers would also constantly have to update their information [51].

We believe that product centric information management could help solve the problem with less effort, as information in supplier databases could be linked to the product object on the catalogue. We have not developed specifications for this type of services, but argue that it as a relevant field of further study.

In a recent interview with representatives of SAP AG, we realised the global appeal of the product centric approach [52]. One of SAP AG's future goals is to promote product centric data management at the product type level in multi-company business networks. In practice, they aim to offer solutions that enable several companies to use common product master data that is residing in the information systems of one company. This would remove compatibility problems, reduce the effort needed to update information when some parameters of the product change, and speed up processes as unnecessary data transfer is minimised [53]. At SAP AG it is realised that product centric master data management is not likely to become common in a short time-span due to mistrust and the power struggle between companies [52]. However, the potential benefits with product centric information management seem attractive at the product type level.

Product centric master data management could also help construct new solutions for communicating product related information in the retail industry. It is an important field of future research because of the extent of the problem area of retail information management. We believe that the product centric approach at a business network level could provide even bigger benefits than industry specific product information databases that are being constructed in several countries.

Acknowledgements

References

[1] M. Kärkkäinen, J. Holmström, Wireless product identification: enabler for handling efficiency, customisation, information sharing, Supply Chain Management: An International Journal 7 (4) (2002) 242–252.

[2] A. Tuttle, Who do you trust, Industrial Distribution 91 (3) (2002) 17, 20, 22.

[3] P. Zipkin, The limits of mass customisation, MIT Sloan Management Review 42 (3) (2001) 81–87.

[4] UCCnet, The Value of Clean Data, An UCCnet White Paper, 2003, Available at http://knowledgebase.uccnet.org/press-Room/publications.asp?n=6-7.

[5] B.H.P.J. Vermeer, Data Quality and Data Alignment in E-Business, Eindhoven University Press Facilities, Eindhoven, 2001, 219 pp.

[6] D. Bowersox, D. Closs, S. Stank, Ten mega-trends that will revolutionize supply chain logistics, Journal of Business Logistics 21 (2) (2000) 1–16.

[7] J. Hamilton, The European Union's consumer guarantees directive, Journal of Public Policy & Marketing 20 (2) (2001) 289–296.

[8] I. Töyrylä, Realising the Potential of Traceability—A Case Study Research on Usage and Impacts of Product Traceability, Finnish Academy of Technology, Espoo, 1999.

[9] F. Hewitt, Demand satisfaction communities: new operational relationships in the information age, International Journal of Logistics Management 11 (2) (2000) 9–20.

[10] T. Andel, The consumer electronics challenge, Material Handling Management 56 (3) (2001) 48–49.

[11] C. Wilder, Purchasing Analyzer, InformationWeek, 17 May 1999, p. 79.

[12] M.J. Euwe, J.C. Wortmann, Logistics control enabled by IT—a window in the future, Human Systems Management 17 (2) (1998) 123–133.

[13] Auto-ID Centre, The Home Pages of the Auto-ID Centre of Massachusetts Institute of Technology, http://www.autoidcenter.org/main.asp, Last visited 10 June 2002.

[14] K. Ashton, Internet Things—MIT, Embedded Technology and the Next Internet Revolution, Tag 2000, Baltic Conventions, The Commonwealth Conference & Events Centre, London, 25 May 2000.

[15] M. Kärkkäinen, J. Holmström, K. Främling, K. Artto, Intelligent products—a step towards a more effective project delivery chain, Computers in Industry 50 (2) (2003) 141–151.

[16] M. Kärkkäinen, T. Ala-Risku, K. Främling, Integrating material and information flows using a distributed peer-to-peer information system, in: Proceedings of the International Conference on Advanced Production Management Systems (APMS), Eindhoven, The Netherlands, 8–13 September 2002.

[17] C. Shapiro, H. Varian, Information Rules, Harvard Business School Press, Boston, 1999, pp. 227–296.

[18] E. Booker, Service Maps To Needs—DHL Lets Customers Dictate Which Services The Company Delivers, InternetWeek, 12 July 1999, p. 17.

[19] A. Coia, Express services unite Europe, Logistics management and distribution report, vol. 40, no. 9, 2001, pp. E3–E5+.

[20] K. Dierkx, Taking Supply Chain Visibility to the Next Level: Powering the Smart Supply Chain, Council of Logistics Management Annual Conference, 26 September 2000, Council of Logistics Management, New Orleans, presentation slides.

[21] AirClic, The Home Page of AirClic, Inc., http://www.airclic.com/, Last visited 14 January 2002.

[22] J. Stebbins, Motorola to Use Barcode Scanner to Net Connect, The Australian IT, 20 June 2000, p. 47.

[23] Wicol, www.wicol.net, The www-Home Page of Wireless Information Collector WICOL Ltd., Last visited 10 June 2002.

[24] K. Främling, Tracking of Material Flow by An Internet-based Product Data Management System, Tieke EDISTY Magazine, No. 1, 2002, in Finnish.

[25] Sun Microsystems, JDBC^{TM} Data Access API, http://java.sun.com/products/jdbc/, Last visited 13 March 2002.

[26] Sun Microsystems, RMI Specification, http://java.sun.com/products/jdk/1.2/docs/guide/rmi/spec/rmiTOC.doc.html, Last visited 14 March 2002.

[27] Orfali, Robert, Dan Harkey, Jeri Edwards, Instant CORBA, Wiley, New York, 1997, 313 pp.

[28] W3C, Simple Object Access Protocol (SOAP) 1.1, http://www.w3.org/TR/SOAP/, Last visited 14 March 2002.

[29] S. Oaks, H. Wong, Jini in a Nutshell, O'Reilly & Associates, USA, 2000, 400 pp.

[30] JXTA, Project JXTA, http://www.jxta.org/, Last visited 14 March 2002.

[31] K. Främling, J. Holmström, A distributed software for collaborative sales forecasting. in: Proceedings of the Management and Control of Production and Logistics MCPL'2000 Conference, 5–8 July 2000. Editor: Binder, IFAC Publications, Elsevier, Grenoble, France, 2000.

[32] Biennier, Security Integration in Inter-Enterprise Business Process Engineering, in: Proceedings of the International Conference on Advanced Production Management Systems (APMS), 8–13 September 2002, Eindhoven, The Netherlands.

[33] R. Monson-Haefel, D. Chappell, Java Message Service, O'Reilly & Associates, USA, 2000, 220 pp.

[34] P. Bradley, J. Thomas, T. Gooley, J. Cooke, Merge-in-transit yields benefits, Logistics Management and Distribution Report 37 (10) (1998) 30.

[35] R. Dawe, Move it fast . . . eliminate steps, Transportation and Distribution 38 (9) (1997) 67–74.

[36] M. McLeod, Cutting both ways, Supply Management 4 (7) (1999) 24–25.

[37] M. Kärkkäinen, M. Punakivi, T. Ala-Risku, Merge in transit—a key for effective order fulfilment in B2B e-commerce, in: Proceedings of the Twelfth International Working Seminar on Production Economics, 18–22 February 2002, Igls/Innsbruck, Austria.

[38] H. Richardson, Planning for the year 2000, Transportation & Distribution 35 (11) (2000) 73–76.

[39] R. Tadjer, Shopping around for the best Internet EDI deal, Network Computing 8 (14) (1997) 26–27.

[40] J. Lankinen, Telephone Conversation with a Salesman of a Finnish EDI Service Provider Oy Datatie Ab, 5 December 2001.

[41] S. Seikkula, Telephone Conversation with a Salesman of a Finnish EDI Service Provider Elma Oyj, 28 February 2002.

[42] C. Wilde, New Life for EDI? InformationWeek, Issue 622, 17 March 1997, pp. 65–67.

[43] R.B. Johnston, A.K.C. Yap, Two-dimensional bar code as a medium for electronic data interchange, International Journal of Electronic Commerce 3 (1) (1998) 86–101.

[44] L. Cohn, D. Brady, D. Whelch, B2b: The Hottest Net Bet Yet? Business Week, 17 January 2000, No. 3664, Industrial/Technology Edition, p. 36.

[45] P. Dittrick, Purchasing savvy to boost the bottom line, Oil & Gas Investor 19 (1) (1999) 46–49.

[46] J. Wetuski, Taking "E" to level B, Oil & Gas Investor, May 2000, On-line E&P Marketplace Supplement, pp. 5–6.

[47] J.P. Baron, M.J. Shaw, A.D. Bailey, Web-based e-catalog systems in B2B producrement, Association for Computing Machinery, Communications of the ACM 43 (5) (2000) 93–100.

[48] E. Morris, New B2B player spreads benefits of e-commerce, Communication News 38 (1) (2001) 98–99.

[49] J.E. Frook, Laying it all on-line, B to B 86 (11) (2001) 20–21.

[50] H. Wraige, Take your mouse shopping, Professional Engineering 15 (10) (2002) 46.

[51] Anonymous, Industry database gets a big boost, Supply House Times, vol. 45, no. 5, 2002, pp. 29–30.

[52] H. Himanen, A. Kojola, An Interview with Henry Himanen and Antti Kojola from SAP Finland, 30 January 2003.

[53] SAP, SAP Develops Collaborative Master Data Management Solution, SAP Press Release, 5 September 2002, Available at: http://www.sap.com/company/press/press.asp?pressID=1411&query=collaborative%20master%20data, Last visited 19 February 2003.

Mikko Kärkkäinen

Timo Ala-Risku

Kary Främling

Mikko Kärkkäinen, **Timo Ala-Risku**, and **Kary Främling** are researchers at the Helsinki University of Technology. Their recent work on developing an information management practices and applications for multi-company supply networks is a part of a research program that aims to identify and develop supply chain solutions that enable productivity and service improvements on the business network level.