A Distributed EPC Discovery Service based on Peer-to-peer Technology

Martin, Lorenz, Hasso-Plattner-Institut, 14482 Potsdam, Germany
Juergen, Mueller, Hasso-Plattner-Institut, 14482 Potsdam, Germany
Matthieu-P. Schapranow, Hasso-Plattner-Institut, 14482 Potsdam, Germany
Dr. Alexander, Zeier, Hasso-Plattner-Institut, 14482 Potsdam, Germany
Prof. Hasso, Plattner, Hasso-Plattner-Institut, 14482 Potsdam, Germany

Abstract

Supply chain visibility and real-time awareness are two of the major drivers for the implementation of Auto-ID technologies in Supply Chain Management. A prerequisite for achieving real-time awareness and company overlapping visibility is an infrastructure to enable companies to share supply chain information in a reliable and secure way. EPCglobal proposes an application layer protocol called the EPC discovery service, which is supposed to provide services to gather supply chain information from a number of independent resources, across company borders. Our investigations on pharmaceutical and tobacco supply chains revealed tremendous data volumes and network traffic, generated from RFID-enabled supply chain networks. It is highly questionable if a single global discovery service is able to cope with such requirements.

In this paper, we address the question how distributed discovery services can deliver their service in a Peer-to-Peer (P2P) based manner. For this purpose, we analyzed the applicability of distinctive distribution schemes and present an approach, which enforces a distribution scheme that allows product managers to decide, which discovery service their data is stored at. Furthermore, we present a prototypical implementation that is based on the open source P2P protocol JXTA. Our architecture utilizes an unstructured P2P network coupled with cache optimizations for lowering response times concerning the processing of a query.

1 Introduction

Supply Chain Management (SCM) is known to be one of the crucial aspects when it comes to a synchronized coordination of the acquisition, production, and deposition of goods. With the ongoing advancements in the field of RFID technology, coupled with unambiguous object identification using the Electronic Product Code (EPC), automatic real-world awareness of supply chain processes can be permitted for supporting SCM, enhancing supply chain performance and lowering overall costs [1, 2].

The vision of real-time item tracking and company overlapping tracing needs to be supported by an appropriate infrastructure. EPCglobal defines such an infrastructure, namely the EPCglobal Network Architecture [3], where the EPC discovery service marks a core component for the retrieval and acquisition of relevant visibility data. This information system provides functionality to determine a complete list of addresses (URLs) of resources, i.e., EPC information services (EPCIS), which are in possession of EPC-related information, i.e., EPC read events.

From such an infrastructure, new scenarios emerge that do not just comprise economic interests with business value, but also scenarios for improving public safety. One of those scenarios involves the pharmaceutical supply chain, where it is of high interest to combat drug counterfeiting. The World Health Organization (WHO) has estimated that five to eight percent of the total amount of the world’s total pharmaceutical sales are either counterfeit or at least of dubious quality [4]. In 2008, the EU reported to have seized 34 millions illegal medicines in two months [5].

Unfortunately, the verification from customer side whether the chain of custody for a specific product is intact, requires to know all of the supply chain partners that were in possession of this product to this date. Because the full lifecycle of the product is not known by all trading partners, a discovery service is needed to determine all needed information. Simulations have shown that the amount of generated information in RFID-enabled supply chains can reach up to several TB of data per year [6]. Additionally, since the global economy comprises of thousands of companies and involves a broad range of products from different companies, distinctive concerns arise, e.g. specific (company-based) data partitioning schemes, security facilities, or scalability issues. From our point of view, the only way to address these issues is by operating a network of federated EPCDSs, which provides means to distribute data volume and network load among a number of interconnected network nodes and at the same time enables the separation of supply chain parties with conflicting interests.

We regard the distribution of data among independent discovery services as inevitable for coping with data volumes and request load expected for RFID-enabled supply chains. Furthermore, we believe that privacy and economic factors, e.g., user fees and quality of service will influence companies, when choosing a provider for discovery service functionality. That is why we focus on a specification of a distributed EPCDS that is based on P2P technology. Choosing a P2P network structure as a basis fulfills the requirements of minimized network traffic and distributed
storage. In particular, P2P networks generally yield independence and ease of distribution, i.e. a non-hierarchical architecture is provided that removes the necessity of central servers, facilitates a high fault tolerance, and provides a high degree of scalability [7]. Our architecture design approach enables independent providers of discovery service functionality to form a federated network to achieve company overlapping collaboration. Clients and companies remain invisible to the network, because they only interact with their trusted EPCDS provider.

The remainder of this paper is structured as follows. First, we define and analyze different distribution schemes for EPCs, where we specifically focus on the structural features of the EPC numbering schema. Next, we define requirements for an EPCDS that can be derived from the chosen distribution scheme and literature review. Afterward, a comprehensive comparison between well-known P2P network structures is given, where pros and cons are analyzed in the context of the defined requirements. Furthermore, we present a specification and system architecture of a distributed EPCDS that incorporates our design decisions and is based on an unstructured P2P approach. In particular, a prototypical implementation is presented that builds amongst others on top of the open source P2P protocol JXTA [8], [9]. Finally, we give an outlook on future work, where we especially discuss further improvements on the topic of authorization and authentication and secure information transmission.

2 Related Work

In this section, we present previous work on the implementation of an EPCDS as well as well-known P2P systems.

One of the first implementation concepts for discovery services was introduced by Beier et. al. in [10]. They already proposed a network of a number of distributed discovery services, each serving a different type of industry. However, they do not consider any type of communication among discovery services.

The BRIDGE project, supported by the EU and coordinated by GS1, investigated designs of different discovery service architectures for the EPCglobal network, regarding functional and non-functional requirements, which have been derived from literature research, company surveys, and expert interviews. Their project report for work package two [11], which targeted discovery service designs, includes a comparison of different technologies, suitable for large scale data storage. Among others, they investigated the possibility to use a Distributed Hash Table (DHT) based on P2P network technology. In their conclusion, they judged the DHT as the most promising approach, considering scalability the predominant requirement for the discovery service design.

A prototypical approach of a discovery service implementation is described by Barchetti et. al. in [12]. In their work, they developed a discovery service prototype that builds on Fosstrak’s open source EPCIS. They tested their system in a controlled laboratory environment simulating the whole pharmaceutical supply chain. The prototype does not implement any inter discovery service communication. However, in their future work they stress the fact that they need to define and implement a discovery service mechanism based on P2P technology, to improve scalability and effectiveness.

The only prototypical implementation of an EPCglobal conform, distributed discovery service, based on P2P technology is introduced by Manzanares-Lopez et. al. in [13]. They explicitly describe the design of a distributed discovery service architecture, using a structured overlay network based on a DHT. Their main aim is to guarantee scalability. However, after carefully examining their work, we believe that there have been some misunderstandings regarding the role and functionality of some of the EPCglobal network architecture components. In detail, they suggest to use the ONS to resolve the internet addresses of EPCIS servers, which have been in contact with EPCs in question. The problem is that the EPCglobal standard explicitly defines the ONS to hold only the address of the EPCIS of the entity that has assigned the EPC to the object. Given an EPC, an ONS query would only result in a single address. No other EPCIS addresses could be resolved. A second weak point of the design is imposed by the DHT function that maps EPC codes to network nodes. There could be more than one node that satisfies the requirements of an entry. As a result, a search in the network can not determine a single node that is in possession of the information about the EPC. Hence, there is no way to determine if all resources of information, i.e. nodes in the network have been found and there is no guarantee that the answer from the discovery service is complete. This problem aggravates, when new nodes enter the network or when nodes leave the network. Furthermore, our intention is to describe a network design for independent discovery services from potentially different vendors. The design proposed by Manzanares-Lopez et. al. targets the distribution of a single discovery service in a P2P fashion to achieve scalability. Our argumentation is based on the assumption that there will exist a number of providers of discovery service functionality, which serve distinctive industries and geographical regions.

The most comprehensive statement regarding inter discovery service communication is found in the Extensible Supply-chain discovery service (ESDS) protocol, which is an internet draft managed by the Internet Engineering Task Force (IETF) [14]. Section 5.11 of the problem statement of the ESDS protocol explicitly proposes P2P technology, to provide a federated network of independent discovery services. Their argumentation is based on the assumption that the lifecycle information about trade items is potentially distributed over a number of different discovery services.
However, an interested party should be able to gather the desired information sources solely from the identifier, i.e. the EPC. The actual discovery process should be carried out by the discovery service network, which should provide functionality to determine the correct resources dynamically. For that purpose, the ESDS internet draft explicitly accounts P2P technology to be the technology of choice, to construct an overlay network for discovery services to exchange information among one another.

Summarizing, we can state that discovery service design proposals value the existence of political and scalability issues by considering the presents of a number of independent discovery services. However, only the ESDS protocol explicitly considers an inter discovery service communication, to create a federated network of independent discovery services that collaborate to fulfill their service.

3 Conceptual Design

In this section, we present our design concept of an overlay network of independent discovery services based on P2P technology.

3.1 Distribution Scheme Definition

The definition of a distribution scheme is essential for the subsequent selection of a P2P network structure, in particular the distinctive distribution of EPCs among network nodes is about to reflect the system configuration and communication architecture. Following considerations only concern how discovery services are distributed and EPC read events can be mapped from a logical perspective. The following enumeration lists, distinctive concerns that arise, when thinking about how a possible distribution scheme for EPCs can look like:

1. Control of mapping: To which degree producers or managers are able to control the mapping process of a specific EPC to an EPCDS, and how political concerns for the distribution of EPC read events in foreign countries are handled.
2. Control of scaling: To which degree a scaling for balancing the load across discovery services is controllable.
3. Degree of spread: To which degree the spread of EPC read events across multiple discovery services applies.

It is clear that these three concerns stand in conflict to each other. Prioritizing Control of mapping the most, may result in an unequal distribution of read events, which in turn causes a higher degree of spread. The same accounts for scalability. Companies may favor discovery services from certain providers, resulting in an increased load on their servers, while servers of other providers may still have plenty capacity. However, we believe political issues and self-determination to be the pre-dominant factors when it comes to the participation in company overlapping collaboration using sensitive supply chain data. That is why we consider Control of mapping as the biggest concern for managers. Being able to control the distribution of data among nodes, directly influences Control of scaling and Degree of spread.

Now that we determined Control of mapping as the top priority for the distribution scheme, we need to define a way to enforce this scheme based on existing technologies. Fortunately, the EPC numbering scheme introduced by the EPCglobal tag data standard [15] already defines a possible distribution scheme, which provides a way for companies to control the distribution of their EPCs on a fine-granular level. The two types of EPCs used for the identification of items and logistic units, e.g., boxes, pallets, are the Serialized Global Trade Item Number (SGTIN) [15] and the Serial Shipping Container Code (SSCC) [15]. Figure 1 depicts the structure of both EPC types. As we can see,

SGTIN Syntax:

\[
\text{urn:epc:id:sgtin:CompanyPrefix.ItemReference.SerialNumber}
\]

Example: \( \text{urn:epc:id:sgtin:0614141.112345.40} \)

SSCC Syntax:

\[
\text{urn:epc:id:sscc:CompanyPrefix.SerialNumber}
\]

Example: \( \text{urn:epc:id:sscc:0614141.123456789} \)

4 Operational Usage Scenarios

The last section prioritized a distribution schema that favors self-determination of companies regarding the choice where to store the data that is related to a group of EPCs. In this section, we exemplify the usage scenarios defined for our distributed discovery service. The practicality of the scenarios is based on the following assumptions.

- All supply chain participants have interest in sharing their information.
- All EPCDSs in the federated network are trustworthy.
- The P2P network provides secure communication for all members

Derived from the notion of P2P, our EPCDS network acts as one logical unit, distributed over a number of peers, sharing the information for the entire network. For the client that is requesting specific EPC information, it is transparent where the actual information of interest is stored. The client only needs to know the interface of a single EPCDS to get all the information needed. Any peer in the
network has access to all the information in the network. Each operator of an EPCIS has to register the EPCIS at any arbitrary EPCDS. Once an EPCDS gets a registration of a new EPCIS, it registers a standing query at that EPCIS, requesting information about any incoming EPCs. That way the information is sent automatically to the EPCDS, where it is being processed.

To fulfill its tasks, an EPCDS needs to be assigned an EPC pattern that it is responsible for, e.g., all EPCs with a particular company prefix. EPCs passed to that EPCDS will be checked if they match the pattern defined for this peer. If so, they will be stored locally at that EPCDS. Otherwise, the EPCDS determines the peer in the network that is responsible for the EPC and sends a storage request to that peer.

The information lookup is similar to the storage of information. A client interested in the information about an individual EPC can query an arbitrary EPCDS. The EPCDS checks, whether the EPC pattern matches the pattern, the peer is configured for. If it is a match, it will perform a local lookup in order to find the URLs related to that EPC. If it is not responsible, the EPCDS determines the peer in the network that is responsible for the EPC and queries it for the information about the respective EPC.

5 Requirements Analysis

The previous section described the processes for our discovery service design. In this section, we define the functional requirements, needed to enact these processes. They are mandatory for the implementation of an EPCDS.

We agree with the authors of the works, presented in Section 2 that a discovery service should be designed as a lightweight referral service, whose sole purpose is to help a client to find one or more sources of detailed information. In accordance with this idea, we enumerate functional requirements for our EPCDS. It needs to provide functionality to:

1. capture notifications about the observation of EPCs
2. store data locally
3. capture query requests for EPCs
4. lookup data locally

Security considerations, e.g. authentication and authorization, are not subject to this work. The research community provides a number of publications explicitly investigating security mechanisms for internet applications, e.g., Public Key Infrastructure (PKI) [16, 17]. That is why we expect this topic can be covered by previous works, whose concepts can be applied to our design.

Apart from the functional requirements needed to operate the discovery service as a single source of information, we also need to define requirements to provide functionality for the communication between discovery services. The enumeration below lists these requirements:

1. bootstrapping responsible peers from an EPC
2. send storage request to remote peers
3. send look-up requests to remote peers

6 Architecture Design Proposal

In this section, we depict an architecture design proposal for a distributed discovery service for the EPCglobal network. For our design proposal, we assume EPCglobal conform EPCIS server as resources, which store the actual EPC read events, captured by the RFID readers. We expect two types of clients for our approach. The first is clients, who are registering the EPC patterns and resource addresses at the EPCDS. The second group are query clients, which are interested in resource addresses for particular EPCs.

Based on the requirements stated in Section 5 and the usage scenarios described Section 4, we define a system architecture and reason on our design decision. We also developed a prototype to proof the practicality of our approach and to study the behavior of a distributed discovery service in real world scenarios. However, due to size limitations of this paper, we will leave out implementation related information. Figure 2 illustrates the system design. We introduce and explain the components of the architecture from top to bottom, starting with the Web Frontend.

The purpose of the Web Frontend is to provide a means to administer and monitor the EPCDS. Primarily it is to provide an interface to manage EPC patterns that should be handled by this EPCDS. It should be possible to register and unregister EPC patterns during runtime. Additionally, it is necessary to be able to register resource, i.e. EPCIS servers, which should be connected to this EPCDS. We expect this process to be performed by managers, whose companies choose this discovery service as their entry point to the network.

The Web Frontend is directly connected to the Request Processor. This component contains the logic of the system. It manages the registered EPC-patterns and handles incoming storage or look-up request. In detail, the Request Processor matches incoming requests against the registered list of EPC patterns. Depending on the outcome of the matching process, the component decides whether to execute the request locally or to relay it to a remote peer. If the request needs to be processed locally, the Request Processor creates a storage/look-up request, which is being delegated to the Persistence Unit. If the request is a look-up request, the Request Processor awaits the result from the Persistence Unit and returns it to the requesting client. The Persistence Unit itself is an abstraction layer, to encapsulate the communication with the underlying storage technology, e.g., a database management system (DBMS). The QueryClient registers standing queries at EPCIS servers that have been registered as resources. The registration is a standard conform invocation of the EPCIS’s QueryInter-
Since we choose Java as programing language for the implementation of our prototype, we use JXSE, an open source Java implementation of the JXTA protocols, as the basis for our P2P Module. We restrict the description of our prototype to a brief introduction of the main concepts used. JXTA uses so called Endpoints that symbolize potential connection points to peers in the network. Our prototype implements an Endpoint interface to establish communication to other peers. A peer in this scenario is a single discovery service instance, which is responsible for a number of EPCs. In order to exchange messages, two Endpoints need to create a JXTA Pipe, which is a bidirectional channel between two Endpoints. This channel can then be used to exchange application specific messages. The discovery of peers in the network is managed via so called Advertisements. These Advertisements are a central concept in JXTA. Every resource in the network, e.g., Endpoints, Pipes, Peer Groups, etc., can be made available to the network via Advertisements. Peers that want to offer their resources to the network create Advertisements, which contain resource-specific metadata, e.g., id, type, name, etc. Once a peer creates an Advertisement it publishes it to the network, so other peer can discover the peer’s resources. For our prototype we extended this concept. We created a new type of Advertisement, which defines an EPC pattern in its metadata. Furthermore, this customized Advertisement contains a pointer to an Endpoint, which belongs to a peer that is responsible for the EPC pattern defined in the Advertisement. That way, peers in the network can search for Advertisements that contain the EPC pattern that matches the EPC they have information about or which the want to have information about. When they find the particular Advertisement, they can establish a Pipe to the Endpoint of the peer, to invoke look-up or storage functionality. The Advertisements have one big advantage. In case a discovery service in the network tries to establish a connection to another discovery service for the first time, it performs a broadcast into the network, to
find the Advertisement of interest. Fortunately, this only has to be done the first time. After that, the discovery service peer caches the found Advertisement. The next time it wants to perform a look-up or storage request for that particular EPC pattern, the Advertisement is already in the local cache and the discovery service peer can directly create a Pipe to the peer’s Endpoint.

Operators of EPCISs can register their EPCIS at any arbitrary discovery service and manufacturers register their assigned range of EPC numbers at any discovery service. Our distribution scheme, combined with the extended JXTA framework take care of the correct distribution and look-up of EPCs within the network, following the operational scenarios described in Section 4.

7 Summary and Future Work

In this paper, we introduced a partitioning scheme for EPCs to distribute them among independent discovery services. The biggest advantage of this scheme is that it provides companies with ability, to decide where they publish their information to and manufacturers of products with the ability to decide where the information of their products will be stored. That way, we address both scalability and political issues, related to the management of RFID-data form different supply chains. Based on the described distribution scheme, we developed a prototype of a distributed EPC discovery service. The design of this prototype is also presented in this work.

For future work, we concentrate on the implementation of security and privacy, using the different security measures provided by the JXTA framework. We mentioned the concept of peer groups in the context of JXTA. Our prototype is still missing any peer group functionality. We will investigate possible usage scenarios and an integration of the concept into our prototype. A second field of future work is the analysis of our prototype, regarding the behavior of the network under different load situations and network configurations.

8 References

Biographies

Martin Lorenz
Hasso Plattner Institute
Enterprise Platform and Integration Concepts Group
August-Bebel-Str. 88,
14482 Potsdam, Germany,
martin.lorenz@hpi.uni-potsdam.de

Martin Lorenz is a research assistant at the Hasso Plattner Institute in Potsdam, Germany at the chair of Prof. Hasso Plattner. He graduated from the Hasso Plattner Institute with a Master’s degree in IT-Systems Engineering. His research interests are RFID in Supply Chain Management and in-memory technology. Martin was employed as an intern at the Daimler-Chrysler AG in 2006 and spent 6 month as an intern at the SAP Research Labs at the University of St. Gallen, Switzerland. He directly started to work as a research assistant after finishing his Master’s studies in 2010.

Matthieu-P. Schapranow
Hasso Plattner Institute
Enterprise Platform and Integration Concepts Group
August-Bebel-Str. 88,
14482 Potsdam, Germany,
matthieu.schapranow@hpi.uni-potsdam.de

Matthieu-P. Schapranow received the BSSE and MSSE degrees in software engineering from the Hasso Plattner Institute, Potsdam, Germany in 2006 and 2008, respectively. Currently, he is Ph.D. candidate and SAP NetWeaver Expert at the chair of Prof. Plattner at the Hasso Plattner Institute. Besides his activities in the field of main memory databases for enterprise applications his interests focus on security and data protection in context of RFID technology and sensor data.

Jürgen Müller
Hasso Plattner Institute
Enterprise Platform and Integration Concepts Group
August-Bebel-Str. 88,
14482 Potsdam, Germany,
juergen.mueller@hpi.uni-potsdam.de

Jürgen Müller is a research assistant at the Hasso Plattner Institute in Potsdam, Germany at the chair of Prof. Hasso Plattner. He graduated from the elite University of Göttingen with a diploma in Business Information Systems. His research interests are Discovery Services in the EPC Network, RFID, Supply Chain Management, ERP Systems, and Software Engineering. In 2006, he spent half a year in Shanghai to improve his Chinese.