Technology
Arts Sciences
TH Köln

Improving e-commerce fraud investigations in virtual, inter-institutional teams:

Towards an approach based on Semantic Web technologies

Master thesis

by

Andreas Gerlach

submitted to obtain the degree of

Master of Science (M.Sc.)

at

TH Köln - University of Applied Sciences
Institute of Informatics

Course of Studies

Web Science

First supervisor: Prof. Dr. Kristian Fischer
TH Köln - University of Applied Sciences

Second supervisor: Stephan Pavlovic
TH Köln - University of Applied Sciences

Cologne, August 2016

Contact details: Andreas Gerlach
Wilhelmstr. 78
52070 Aachen
andreas.gerlach@smail.th-koeln.de

Prof. Dr. Kristian Fischer
TH Köln - University of Applied Sciences
Institute of Informatics
Steinmüllerallee 1
51643 Gummersbach
kristian.fischer@th-koeln.de

Stephan Pavlovic
TH Köln - University of Applied Sciences
Institute of Informatics
Steinmüllerallee 1
51643 Gummersbach
stephan@railslove.com

Abstract

There is a dramatic shift in credit card fraud from the offline to the online world. Large online retailers have tried to establish countermeasures and transaction data analysis technologies to lower the rate of fraudulent transactions to a manageable amount. But as retailers will always have to make a trade-off between the *performance* of the transaction processing, the *usability* of the web shop and the overall *security* of it, one can assume that E-commerce fraud will still happen in the future and that retailers have to collaborate with relevant business partners on the incident to find a common ground and take coordinated (legal) actions against it.

Trying to combine the information from different stakeholders will face issues due to different wordings and data formats, competing incentives of the stakeholders to participate on information sharing as well as possible sharing restrictions, that prevent them from making the information available to a larger audience. Additionally, as some of the information might be confidential or business-critical to at least one of the parties involved, a *centralized* system (e.g. a service in the cloud) can *not* be used.

This Master thesis is therefore analyzing how far a computer supported collaborative work system based on peer-to-peer communication and Semantic Web technologies can improve the efficiency and effectivity of E-commerce fraud investigations within an inter-institutional team.

Keywords: peer-to-peer communication, Semantic Web, CSCW

Contents

1 Introduction

This introductory chapter of the Master thesis starts with a section showing the importance and relevance of the topic in the research area of Web Science, which is followed by a short description of the problem, that this thesis will focus on, and ends with an outline of its structure.

1.1 Motivation

> *"When it comes to fraud, 2015 is likely among the riskiest season retailers have ever seen, [...] it is critical that they prepare for a significant uptick in fraud, particularly within e-commerce channels." (Reuters 2015)*

This statement from Mike Braatz, senior vice president of Payment Risk Management, ACI Worldwide in (Reuters 2015) shows the dramatic shift in credit card fraud from the offline to the online world, that retailers are starting to face nowadays.

In general credit card fraud can occur if a consumer has lost the credit card, or if the credit card has been stolen by a criminal. This usually results in an identity theft by the criminal, who is using the original credit card to make financial transactions by pretending to be the owner of the credit card. Additionally, consumers might hand over their credit card information to untrustworthy individuals, who might use this information for their own benefit. In the real world scenario there is usually a face-to-face interaction between both parties. A consumer, wanting to do business with a merchant or interacting with an employee of a larger business, has to hand over the credit card information explicitly and can deny doing so in a suspicious situation. The criminals on the other hand must get access to the physical credit card first, before they are able to make an illegal copy of it — a process called skimming. The devices used to read out and duplicate the credit card information are therefore called skimmers. These can be special terminals that the criminals use to make copies of credit cards they get their hands on, or those devices can be installed in or attached to terminals the consumers interact with on their own (Consumer Action 2009). All of these so-called *card-present transaction* scenarios have seen a lot of improvements in

security over the last years. Especially the transition from magnetic swipe readers to EMV chip-based credit cards makes it more difficult for criminals to counterfeit them (Lewis 2015).

As a consequence criminals are turning away from these card-present transaction scenarios in the offline world. Instead they are focusing on transactions in the online and mobile world, in which it is easy to pretend to own a certain credit card. Most online transactions (either E-commerce or M-commerce) rely *only* on credit card information such as card number, card holder and security code for the card validation process; therefore these interactions are usually called *card-not-present transactions*. The credit card information can be obtained by a criminal in a number of ways. First they might send out phishing emails to consumers. These emails mimic the look-and-feel of emails from a merchant or bank, that the consumers are normally interacting with, but instead navigate them to a malicious web site with the intend to capture credit card or other personal related information (Consumer Action 2009). Additionally, criminals can break into the web sites of large Internet businesses with the intent of getting access to the underlying database of customer information that in some cases also holds credit card data (Holmes 2015). Additionally, some of the online retailers are not encrypting the transaction information before transmitting them over the Internet; a hacker can easily start a man-in-the-middle attack to trace these data packages and get access to credit card and personal related information in this way (Captain 2015).

Based on these facts it should not come as a surprise, that the growth rate of online fraud has been 163% in 2015 alone (PYMNTS 2016). This results in huge losses for the global economy every year, and it is expected that retailers are losing $3.08 for every dollar in fraud incurred in 2014 (incl. the costs for handling fraudulent transactions) (Rampton 2015). These fraudulent transactions also impact the revenue of the online retailers. Here we have seen a growth of 94% in revenue lost in 2015. Overall it is estimated that credit card fault resulted in $16 billion losses globally in 2014 (PYMNTS 2016) (Business Wire 2015).

While it is possible to prevent fraudulent transactions in the card-present, real-world scenario (mostly due to introducing better technology and establishing organizational countermeasures in the recent past), it is more difficult to do so in the card-not-present E-commerce and M-commerce scenarios, which are lacking face-to-face interactions and enable massive scalability of misusing credit card information in even shorter time frames (Lewis 2015). Large online retailers have tried to establish countermeasures and transaction data analysis technologies to lower the rate of fraudulent transactions

to a manageable amount. But this is still an expensive and inefficient solution to integrate into the retailers' business processes, and is largely driven by machine-learning techniques and manual review processes (Brachmann 2015). Additionally, it can be assumed that the online retailers are getting into a "Red Queen race" with the criminals here: with every new technology or method introduced they might just be able to safe the status quo. This is largely due to the facts, that there will be no 100% security for a complex and interconnected system such as an E-commerce or M-commerce shop, the criminals will also increase their efforts and technology skills to adapt to new security features; and most importantly retailers will always have to make a trade-off between the *performance* of the transaction processing, the *usability* of the web shop and the overall *security* of it.

1.2 Problem Definition

This Master thesis will look into a concept to optimize the collaboration between the affected stakeholders in case of an existing credit card fraud in an E-commerce system. It will *not* look into novel techniques and methods to *prevent* credit card fraud in the E-commerce world. This aspect has been seeing a lot of research in the last years.[1]

Stakeholders might include vendors and other businesses, that a retailer has a long-term business relationship with, law enforcement agencies, payment service providers such as PayPal or Visa, banks, and even competitors, that are also affected by the Internet frauds. In these cases merchants usually try to solve the issues on their own, and getting in contact with relevant parties by phone or e-mail if necessary. But these communication styles do not fit to the complexity of the task involved, and based on the media-richness model (see Figure 1.1) will result in inefficient and ineffective problem solutions.

Due to the task complexity a physical face-to-face meeting with representatives of all stakeholders involved might be a good fit, but arranging such a meeting (at the same time and on the same place) with multiple parties, that are globally dispersed, is either economically not feasible or takes a lot of time. But the more time passes for investigating a fraud, the more difficult it will become to identify the fraudsters and take legal actions against them. Acting in a timely fashion can therefore reduce the

[1] Please also note the various US patent applications of Google on that matter from 2015, e.g.: "Credit card fraud prevention system and method", "Financial card fraud alert", "Payment card fraud prevention system and method" (Google Patents).

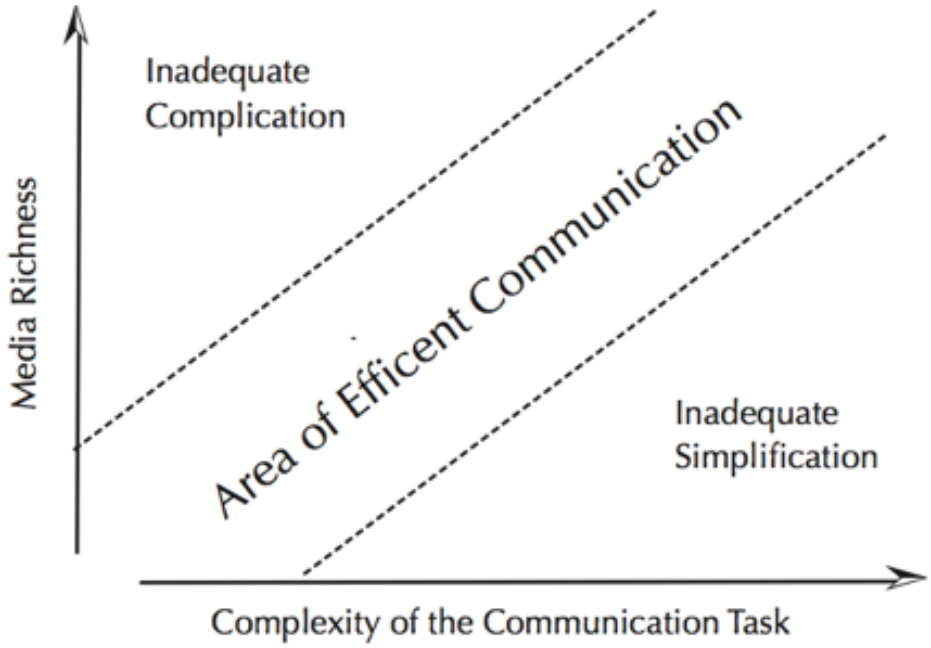

Figure 1.1: The Media Richness Model (Rice 1992)

risk of losing the money completely.

As of these conditions a computer-supported collaborative work (CSCW) system might be an alternative to *collaborate* on an incident of E-commerce fraud (at the same time, but on different places). CSCW systems can be categorized by their support for the mode of group interaction as done in the "3C model" (Koch 2008):

- **communication:** two-way exchange of information between different parties,
- **coordination:** management of shared resources such as meeting rooms,
- **collaboration:** members of a group work together in a shared environment to reach a goal.

Based on the level of support for one of these functionalities the various systems can be classified and described as shown in Figure 1.2:

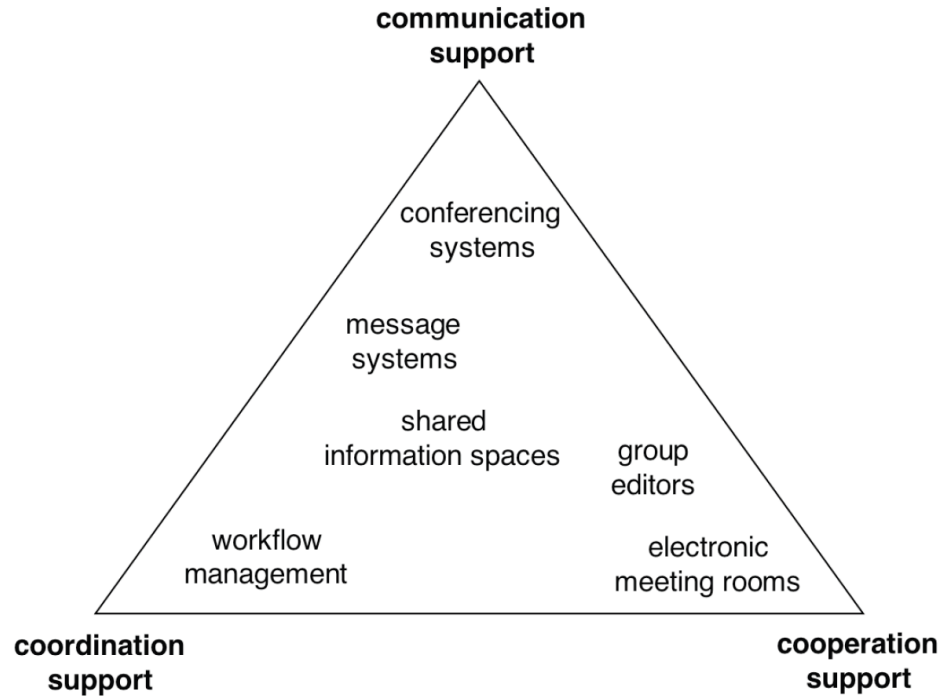

Figure 1.2: The 3C Model (Koch 2008)

A good candidate for such a collaborative system *could* be a shared information space; aka team rooms, cloud storage services or document management systems, that allow participating parties to access information at any place, any time and to share information between each other — usually with a build in versioning support for artefacts and a workflow component.

However, as some of the required information might be confidential or business-critical to one of the involved parties, a centralized system (e.g. a service in the cloud) can *not* be used in the scenario described here. Another key characteristic of the investigation of an E-commerce fraud is the fact, that it involves information sharing from many different organizations. These different aspects have to be combined into a shared information space in a meaningful way to be able to achieve a common group goal on time. Trying to combine information from different stakeholders will face issues due to different wordings and data formats, competing incentives of the stakeholders to participate on information sharing as well as possible sharing restrictions, that prevent making the information available to a larger audience.

Decentralized information sharing architectures, which utilize peer-to-peer communication technologies, are either restricted to a commonly agreed set of data entities and relations between all parties involved, or are lacking richer semantics for sharing and integrating content between the stakeholders. Semantic Web technologies can help lower the barrier to integrate information from various sources into a shared information space, and the advantages of peer-to-peer communication and Semantic Web technologies for information sharing in distributed, inter-organizational settings have been shown in (Staab & Stuckenschmidt 2006).

Still these studies concentrate on making information from different parties searchable and accessible in a distributed, shared information space, in which data can be accessed and queried at any time from any participating party. They are not solving the problem of working collaboratively on a common goal in an ad-hoc, loosely-coupled virtual team of disperse organizations by making certain (sometimes sensitive) information available in a shared environment.

Therefore, the research question for this Master thesis can be summarized as follows:

> *In how far can a computer supported collaborative work system based on peer-to-peer communication and Semantic Web technologies improve the efficiency and effectivity of E-commerce fraud investigations within an inter-institutional team?*

1.3 Master Thesis Outline

Before starting with the investigation of E-commerce fraud incidents and their possible examinations, the thesis starts with a description of related works in Chapter 2. These research papers have been evaluated during the course of this Master thesis, and have had an influence on it.

In the next part, Context Analysis in Chapter 3, the thesis discusses the E-commerce scenario in detail. It starts with a description of the E-commerce shopping process, looks into the stakeholders involved as well as shows possible kinds of E-commerce fraud incidents and how they are handled today. Based on these findings this chapter closes with a presentation of the specific scenario, that has been selected for further examination within this Master thesis.

After this initial scope setup the thesis briefly outlines the theoretical foundations required for the understanding of the concepts in Chapter 4 and design decisions in Chapter 6. This section starts with a short overview of the relevant facets of computer-supported collaborative work systems (CSCW), shows the essential specifications of the Semantic Web, and ends up with an introduction to the peer-to-peer (P2P) communication techniques and protocols.

In the main parts of this thesis (Chapter 5 and Chapter 6) the concept and design for a collaborative system, that supports the investigation of E-commerce fraud incidents, is discussed. These chapters will lay out and analyze the possibilities for designing and using such a collaborative system. The objective is to come up with an approach at the end of the discussions, that might be the best fit for the problem described in the scenario at the beginning.

To conclude the thesis also sum up the findings and give an outlook for future work on this topic.

2 Related Works

- “A Study on E-Commerce Security Issues and Solutions” (Sen et al. 2015) - “A Survey on Fraud Detection Techniques in Ecommerce” (Rana & Baria 2015)
- “Overview of E-Commerce” (Ankhule & Joshy 2015)
- “Fraud in Non-Cash Transactions: Methods, Tendencies and Threats.” (Sobko 2014)
- “Applying Semantic Technologies to Fight Online Banking Fraud” (Carvalho et al.)
- “Goodrelations: An ontology for describing products and services offers on the web” (Hepp 2008)
- “Effects of Sensemaking Translucence on Distributed Collaborative Analysis” (Goyal & Fussell)
- “Linked data-the story so far” (Bizer et al. 2009)
- “Linked data-as-a-service: the semantic web redeployed” (Rietveld et al. 2015) - “Schema.org: Evolution of structured data on the web” (Guha et al. 2016)
- “What is schema.org?” (Barker & Campbell 2014)
- “Leveraging WebRTC for P2P content distribution in web browsers” (Vogt et al. 2013b)
- “Taking on WebRTC in an enterprise” (Vogt et al. 2013b)
- “Content-centric user networks: WebRTC as a path to name-based publishing” (Vogt et al. 2013a)
- “RDFPeers: a scalable distributed RDF repository based on a structured peer-to-peer network” (Cai & Frank 2004)

3 Context Analysis

This chapter looks into the scenario of E-commerce fraud investigation in detail. It starts with an in-depth description of the E-commerce scenario followed by an analysis of the stakeholders involved. It further describes the kind of information each stakeholder has in their local context, and their objectives to take part on the information sharing and collaboration initiative. Based on the analysis of the possible kinds of E-commerce fraud incidents and the current process of their investigation, the chapter closes with a description of the specific scenario, that has been selected for this Master thesis.

3.1 An overview of E-commerce

E-commerce as a term relates to the trading of products or services utilizing a computer network such as the Internet. It is usually divided into the following four different subfields (Sen et al. 2015):

1. **Business-To-Business (B2B)**: refers to electronic trading between companies with the objective to improve their supply chain processes,
2. **Business-To-Consumer (B2C)**: refers to electronic trading between a company and its consumers (most prominent example for it is Amazon (Amazon.com)),
3. **Consumer-To-Consumer (C2C)**: refers to electronic trading between consumers (most publicly known example for that is eBay (eBay Inc)),
4. **Consumer-To-Business (C2B)**: refers to electronic trading between consumers and businesses (most notable example for this is TaskRabbit (TaskRabbit)).

Due to the problem initially sketched out in Section 1.1 this Master thesis will *solely* focus on the B2C aspect of E-commerce. In that case a consumer uses an E-commerce shop of a merchant on the Internet to order products or services online. The merchant offers a catalog of available products or services on the Web that is available and accessible by the general public and usually has a nation-wide if not global reach.

The merchant can either run the E-commerce shop software on their own servers (on-premise) or can outsource this additional sales channel to a 3^{rd} party hosting company or cloud service provider (CSP). Also, the E-commerce shop software itself can be either developed by the merchant in-house or acquired as a boxed product from an Independent Software Vendor (ISV) on the market. For business accounting purposes the merchant also runs a bank account with an acquirer (see Figure 3.1).

When placing an order with a merchant online, the consumers normally use a credit card for finalizing the transaction. These credit cards have originally been handed out to the consumers by the issuers. Additionally, in some online shops it is mandatory for the consumers to create a user account with them, while in others it is not. The former is the preferred way when consumers are repetitively buying from that merchant, whereas the latter might be used for one-time or irregular shopping trips online. To be able to connect to the Internet the consumers also rely on a service offered by an Internet Service Provider (ISP). The whole initial setup for participating in E-commerce activities is found in Figure 3.1.

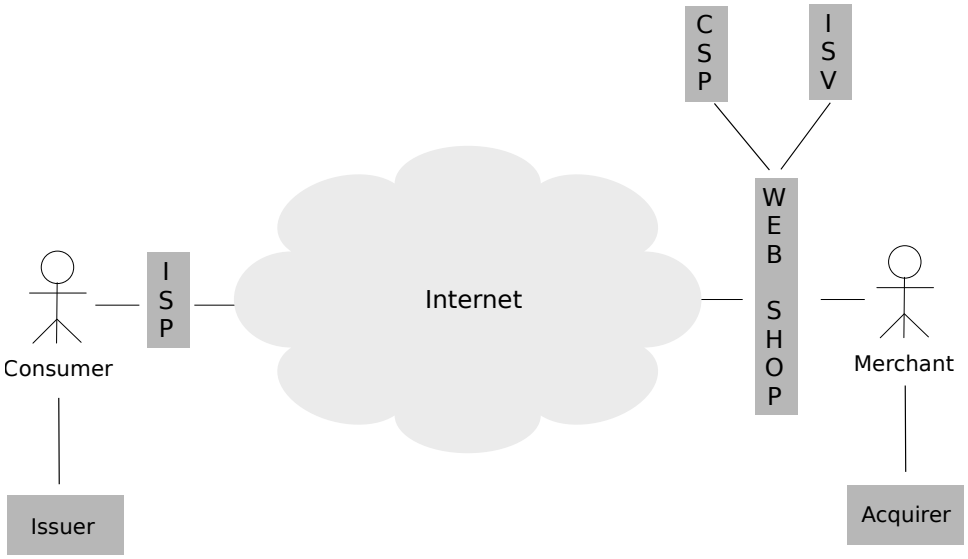

Figure 3.1: E-commerce Fundamentals

When a consumer places an order online, the merchant receives at least a list of products or services from the current shopping cart of the consumer, the identification of the consumer, as well as the delivery address to ship the physical items to. If the transaction is going to be finalized with a credit card, the consumer will have to provide additional information like the billing address and the credit card details (including

the number, the expiry date and the security code of the card).

The merchants usually do not validate the credit card information on their own. For that purpose they are relying on another 3^{rd} party service offered on the Internet by the Payment Service Provider (PSP). These providers either validate the credit card information themselves based on a user profile the consumer has with the PSP (e.g. a globally available Web service such as PayPal), or communicate with the issuer of the credit card for doing so. For initiating this validation process the merchant is handing over the billing information to the PSP incl. the credit card details given by the consumer.

Either the PSPs or the issuers validate the correctness of these information with reference to criteria such as:

- Does the billing address matches the current consumer's postal address on file?
- Is the stated credit card information correct?
- Is the credit card still valid?
- Is the credit card not marked as being blocked in the internal databases?

The merchant receives the status of the authorization as well as an unique payment token in return. If the authorization has been successful, the merchant collects the items and sends out a shipping request to one of the available Logistic Service Providers (LSP), that are capable of delivering the order. They pickup the items at the merchant's facility and ship them to the delivery address stated by the consumer. Usually at about the same time the merchant informs the acquirer about the order, amount due as well as the payment token received from the PSP. The acquirer is in charge to withdraw the amount of the order from the consumer's bank account either via the PSP or directly from the issuer, depending on which of them has authorized the initial payment request (a process called clearing) (Visa Europe 2014). The sequence of activities within an E-commerce checkout process is visualized in Figure 3.2.

3.2 Stakeholders

The following section looks at each stakeholder involved in the E-commerce scenario in detail, lists the kind of information they own or provide to others as well as describes the role of each stakeholder in the E-commerce fraud investigation process (if any).

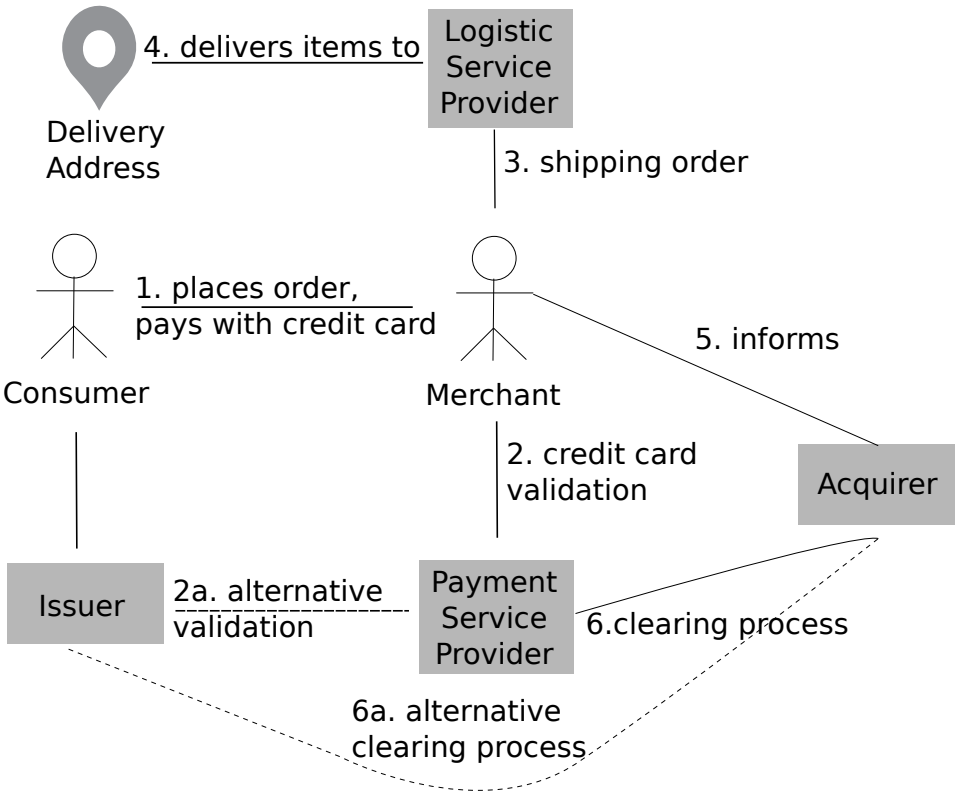

Figure 3.2: E-commerce Checkout Process in detail

3.2.1 Consumer

The consumers are the initiators of E-commerce transactions. They are using the shop of a merchant on the Internet to order products or services. For doing so they have to know the URL of the Web shop, have to be connected to the Internet via an ISP and have to use a standard software called a Web browser on their computer. For the duration of their online browsing sessions they also own a unique IP address handed out from the ISP.

They might have had a long-term business relationship with the merchant and already own an user account on the Web shop. As an alternative they might be just interested into a one-time shopping trip and might want to order the items without creating an account first — sometimes also called “anonymous” or “guest” checkout in the E-commerce shops.

The consumers are also having a bank account and at least own a debit card from that bank to get access to the money on the account. In addition to that they can

also hold multiple credit cards. A credit card can be issued by the same bank, or can be provided by another financial service institutions (e.g. American Express). In any case the organization that has handed out the credit card to the consumer is called the issuer.

If the consumers are going to order items in a Web shop, they will usually browse the product and service offerings of a merchant first and put the items of interest into the shopping cart. When finalizing the transaction they have to state the following information to the merchant:

- personal information incl. given name, family name and date of birth,
- the address the items should be shipped to,
- payment information incl. type of payment and billing address (if different to shipping address).

If they are going to end the transaction with a payment of type credit card they will also have to provide specific information of the credit card, that should be used as payment:

- the owner of the credit card (if it is not belonging to themselves),
- the unique credit card number,
- the expiry date of the credit card (in format MM/YY),
- the security code of the credit card.

The consumers have a special role in the whole scenario. As the online merchants have to deal with the consumers without any face-to-face or real-world interactions, the consumers are also the least trustworthy participants from the point of view of the merchants. As Section 3.4 will show, the consumer is the main questionable object in the case of an E-commerce fraud incident. Therefore, the consumers are not taking any active part in the fraud investigation process.

3.2.2 Merchant

The merchants offer products and services on the Internet to the general public. They might use the Internet as an additional sales channel, or rely on it solely for making any business. To provide access to the Web shop a merchant has to register a domain name and an URL with a local domain name registry. This specific URL refers to a

fixed public IP address, that the server that runs the Web shop software uses. Normally the merchants do not operate the servers themselves, but rely on a service offered by a hosting or cloud service provider for that. Also the Web shop software itself is usually not provided by the merchants, but bought from an ISV on the market. In any case the merchants have special responsibilities in the Web shop, because they have to take care to configure the products, prices, promotions, payment, and shipment services available. In addition products can be categorized by them into categories and sub-categories for easier navigating and searching the offerings in the Web shop by the consumers later.

The merchants can decide whether they restrict ordering of products to registered users only, or allows anonymous users too. The main benefit of the former is the possibility to analyze the shopping behavior of individual consumers, whereas the latter will open the business for a wider range of consumers as it includes also those, who do not want to register with any existing online shop. Nevertheless, any consumer activity on the online shop is tracked in the analytic databases of a merchant. This includes not only the items, that have been placed into the shopping cart, but also any product that a consumer has looked at during a shopping session. Even if these detailed analytic capabilities are actually synonymous for their usage in target-related advertising, they can also help to decide whether a consumer behaves normally or not within a Web shop.

Any business transaction that a consumer makes with a merchant is stored in the merchant's databases. A transaction information contains, but is not limited to:

- personal related information of the consumer,
- the address the items will be shipped to,
- a collection of products with quantities and prices,
- the total amount of the order considering promotions, taxes and fees,
- the selected payment information.

If a consumer wants to pay with credit card, the payment process is not handled by the merchants themselves, but is routed to a Payment Service Provider (PSP) on the Internet. To initiate the credit card authorization, a merchant is sending a request with the following information to the Web service endpoint of a PSP:

- consumer's billing address,

- given credit card number, expiry date and security code,
- identification of the merchant,
- final amount of the current transaction.

In return of the payment authorization a merchant receives and stores these payment-related information for the transaction:

- the type of credit card used (e.g. Visa, MasterCard, American Express, ...),
- the name of the credit card owner,
- the unique payment token received by the PSP,
- the timestamps and result code of the authorization,
- the authority, who has approved the payment (if the merchant works with multiple Payment Service Providers).

As the merchants will collect a lot of personal and payment-related information over time, they are also one of the major sources of possible data leaks in the E-commerce scenario. Due to this circumstance the Payment Card Initiative, a group of banks, issuers and PSPs, provides rules and guidelines (aka PCI/DSS standards) for securely handling these kind of information in an IT system (Virtue 2009).

The merchants are one of the main actors in the fraud investigation process. They are highly interested in figuring out whether the consumer's transactions are valid or not. That is due to the fact, that in case of an E-commerce fraud incident the merchants will mostly have to cover the costs (see Section 3.4). Also the online merchant's reputations will suffer, if private information from their databases get leaked. If a merchant falls victim to fraud incidents multiple times, the economic damages can finally result in a bankruptcy of that merchant.

3.2.3 Payment Service Provider

The Payment Service Providers offer payment-related services to online merchants. To be able to do this a PSP provides a Web service interface, that the merchants have to communicate with by sending payment authorization requests to it (see above). The PSPs might be able to authorize a payment request on their own, or might have to route that request to the corresponding issuer of the credit card in question. For the former procedure the PSPs have to run their own databases of registered users with

their credit card information (e.g. a Web service such as PayPal). For the latter they will just have to know, who has issued the credit card in question, and have to call into the Web service of that issuer for validation purposes. For verifying a credit card and authorizing the payment a merchant hands over the following:

- credit card owner incl. billing address given,
- credit card number,
- credit card expiry date,
- credit card security code,
- identification of the merchant,
- total amount of the current transaction.

In case the PSPs are authorizing the payment requests, they will have to securely process the information and return the validation results to the merchants. Each result message also contains a unique payment token that a merchant can refer to later to initiate the clearing process. As of this the PSPs have to persist the credit card and payment-related information in their own back-end databases. According to industry standards, they should also follow the PCI/DSS guidelines mentioned in the previous section.

The level of activity in the E-commerce fraud investigation process depends on whether the PSPs authorize the payments themselves, or only act as a routing service between the merchants and the original credit card issuers. In the former case the PSPs are more actively involved. In that situation they also holds more of the valuable information to analyze an E-commerce fraud incident. In the latter case they will still be required to connect the payment-related request information from a merchant with the corresponding authorization result coming from an issuer.

If the PSPs hold sensitive information in their own databases, they will also be a source of possible data leaks. In that situation they have to put the same precautions in place as issuers have to do (as explained in the next section).

3.2.4 Issuer

The issuers are the only members in the E-commerce scenario that know the owners of credit cards in person. Each individual has to register personally with an issuer to

get access to a credit card. This registration process includes providing the following information:

- personal related information such as given name, family name and date of birth,
- the currently registered home address,
- the bank account that should be used to settle credit card balances.

Even if the two parties do not really meet each other personally, individuals will still have to identify themselves with a valid ID card and bank account to receive and activate a new credit card. Beside being the single source of truth about the original credit card owner, the issuers of credit cards also collect and store all of their usages. Whereas the Payment Service Providers can only provide individual credit card usage patterns for the online shopping scenario, the issuers can also include those transactions that the credit card owners do in the real-world. Needless to say that these are valuable information for an E-commerce fraud investigation.

Still an issuer does not know any details of the transactions that have been made with a credit card yet. As shown in the Section 3.2.3 the issuers receive only an identifier of the merchants, in whose shops a credit card has been used. Based on public available information from a commercial register about merchants, the issuers could come up with at least the retail branch each merchant operates in.

Being the single source of truth about all issued credit cards, their owners and usage patterns make the issuers another high-risk candidates for possible data leaks. They should as well follow the guidelines from the PCI/DSS standards, should incorporate security standards for their IT systems and the processes of operating them, as well as monitor their back-end systems actively with an intrusion detection mechanism.

3.2.5 Acquirer

The acquirers hold the bank accounts of merchants and are responsible for withdrawing the outstanding amounts of transactions from the accounts of the consumers, or more precisely requesting it from the issuer of each consumer. Due to this an acquirer does usually not process any credit card related information from consumers directly, but refers to the unique payment tokens that have been given out by the PSPs or the issuers during the authorization processes.

Still as financial institutions acquirers (like issuers) have to comply with the rules and guidelines of the PCI/DSS and other industry standards to make sure that their bank accounts as well as the transaction processing are safe and secure. The detailed analysis of these techniques and procedures as well as possible banking fraud incidents are out of scope of this Master thesis though.

3.2.6 Logistic Service Provider

The Logistic Service Providers have two important roles in the E-commerce scenario. First, they have access to and control over the items of a merchant for the duration of the transport between the merchant's facility and the consumer's shipping address. And second, they hold the information to whom they have handed over the items at the final destination. Although the LSPs have nothing to do with any payment-related activities, they are still critical parts of the investigation of fraud incidents as they will be the last chance for a merchant to stop the delivery of an order (in case a fraud has been detected after initiating the shipment), or provide information about the person that has received the items at the shipping address — especially so for orders of high-priced goods, which usually require a recipient to identify with a personal ID card and place a signature on the delivery receipt.

For initiating the shipment procedure a merchant orders a certain transport service from a LSP and hands over the following information:

- name of the recipient,
- delivery address given by the consumer,
- list of items to be shipped,
- optionally: value of the items if an insurance policy is taken.

The LSP returns a unique tracking id for the shipment in response. This number can be used by the merchant, and the consumer, to check for the status of a shipment online.

As the LSPs do not have to deal with the payment-related activities in the E-commerce scenario, they are also not actively involved in the fraud investigation. However, they can stop the delivery of the items, or provide useful information about the recipients if an incident is found.

3.2.7 Cloud Service Provider

The Cloud Service Providers offer IT services to their customers. These IT services include hardware and software assets, that merchants can order in the E-commerce scenario to run their Web shops on the Internet. Part of the service level agreement between a merchant and a CSP is a detailed listing of the responsibilities of both parties (who has to take care of what). In most cases the merchants are outsourcing the complete operation of the hardware and software for their Web shops to the CSPs; so the CSPs are responsible for making sure that the Web shops are available and secure. The CSPs are also constantly monitoring the incoming connections to each public Internet server under their control and can provide information, whether a Web shop of one of the merchants has been compromised or not. Still the CSPs are not actively involved in the E-commerce fraud investigation.

3.2.8 Independent Software Vendor

The Independent Software Vendors design, implement and sell the Web shop software tools. They have detailed knowledge about the software components and libraries used within their Web shop products and check them regularly for security breaches or vulnerabilities. They also have to verify these software parts for vulnerabilities, that they have implemented on their own, as well as have to make sure that their implementations follow industry standards (e.g. PCI/DSS for handling person and payment-related information). Therefore they can best assert these quality criteria of a Web shop software if needed. Due to this the ISVs are not an active member of an E-commerce fraud investigation.

3.2.9 Internet Service Provider

The Internet Service Providers offer services to the consumers, so that they are able to connect to and make use of the Internet. Each Web request consumers are doing on their systems is routed to the Internet via the infrastructure of an ISP. Due to existing regulations and laws the ISPs have to store the log files of each Internet session of their customers for a certain amount of time. Especially, these log files can be helpful to decide whether a consumer was visiting pages in the dark-side of the Web, or if they fall victim to some phishing attacks (explained later in Section 3.4). Although these information can be helpful to decide on fraudulent transactions in the E-commerce scenario, the ISPs are not actively involved in the investigation of it. They are rather required for getting information about the deceivers in case a fraud is found.

3.3 Data flow for credit card transactions

As the previous chapter shows, there are a many stakeholders involved in providing IT hardware, software and services to keep the Web shops on the Internet up and running. Only a small fraction of those will have to deal with the handling of credit card payments and order fulfillments though. These are the relevant stakeholders to look at in the case of an E-commerce fraud incident. The actual flow of information between them is displayed in Figure 3.3.

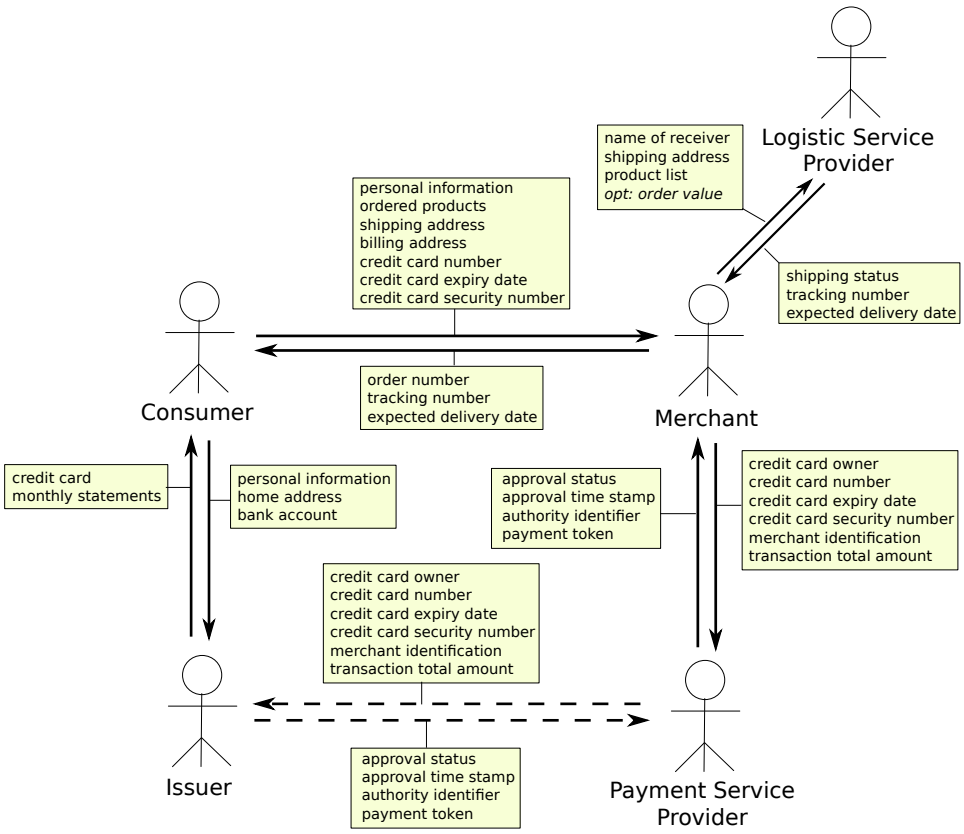

Figure 3.3: Stakeholder and Data Flow in E-commerce scenario

3.4 E-commerce fraud incidents

Based on the previous sections one can come up with strategies fraudsters might use to trick the E-commerce system. To do so the criminals will have to get access to credit card information in the first place. Therefore this section first looks into ways a criminal might get access to credit card and personal-related information in the E-commerce scenario. After that the section describes possible strategies fraudsters can

use to trick the system. The section ends with a discussion of the E-commerce fraud incident handling as it is in place today.

3.4.1 Credit Card data breaches

Based on information in the Section 3.3 one can figure out the parties, who have access to or store credit card information in the E-commerce scenario, namely:

- a consumer as owner of a credit card,
- an issuer, who handed out a credit card to a consumer,
- a merchant, if a consumer is paying with credit card,
- a Payment Service Provider, if a consumer is paying with a credit card online.

The PSPs receive credit card information from merchants with the payment authorization requests. If the PSPs do the authorization themselves, they are also the participants, who store and hold the credit card information in their back-end databases. As mentioned earlier the PSPs should follow industry standards and guidelines for storing and processing payment-related information; especially the PCI/DSS standard (Virtue 2009). In addition they are responsible for monitoring their systems with an intrusion detection program. This utility will trigger a signal as soon as an hacker got access to the internal databases. In that case the PSPs can put the leaked credit card information on an internal blacklist, so that these cards can no longer be used for further payments online. Additionally they will have to send a message to the corresponding issuers, to which the PSPs generally maintain strong business relationships. The issuers will inform the affected credit card owners and send out a new credit card to each of them. Due to this procedure in place, one can assume that the safety and security of credit card handling at the PSPs can be guaranteed.

The merchants receive the credit card information during the checkout processes from the consumers. The credit card information are transfered via the public Internet from the consumers to the merchants and could be victims to man-in-the-middle attacks, in which hackers are intercepting the communication between the consumers and the merchants with the objectives to capture the personal and payment-related information from the data transmission streams. Therefore the merchants should offer their Web shops via a secure communication channel only. For that they can use industry standards such as TLS to encrypt the information that is send between both parties. Doing so will make it more difficult for attackers to get to the plain-text information exchanged between consumers and merchants during the checkouts. As the merchants

are not processing the credit card information directly, they also do not have to store them in their own back-end databases. The merchants are asking the PSPs or the issuers of the credit cards for authorization of a payment and receive an unique payment token in response, if that authorization was successful. As stated in the PCI/DSS standard (Virtue 2009) merchants should *never* store credit card information as a whole in their own databases, but should use the unique payment tokens and shortened credit card data (especially abbreviated credit card numbers) to refer to a specific payment later. Due to this procedure in place one can conclude, that breaking into the systems of a merchant will not result in any leaked credit card information, if the merchants follow these guidelines.

The issuers are a valuable target for hacking into the back-end systems with the objective to leak a massive amount of credit card and personal related information. As financial institutions the issuers also have to follow a huge set of regulations and safety procedures to be able to participate on the market. It can be assumed that at least the same safety mechanisms are valid as are in place for the PSPs. This means constantly monitoring the internal systems with an intrusion detection mechanism and blacklisting any leaked credit card. In addition to the monitoring of all online activities (as also the PSPs are doing) the issuers can monitor activities done with the credit card in the offline world too. In case of suspicious activities the credit cards can be blocked immediately, and new ones will be send out to each affected owner.

The consumers are also a valuable target for eavesdropping on credit card and personal related information. They are also the weakest and most insecure party in the whole E-commerce scenario. As shown before a lot of the protection mechanisms of the other participants rely on following industry standards, and on constantly monitoring the own systems for malicious activities. This can not be securely said about the computers of the consumers though. Whether they are using up-to-date security programs (e.g. an Anti-virus tool and a firewall) on their computers or not is out of reach of the other actors to verify. Additionally, consumers can fall victim to phishing attacks, that will send them to malicious Web sites with the intend to get their personal related information. In some seldom cases the consumers might cooperate with fraudsters, or might be the impostors themselves with the intent to trick the system for their self-interests. Due to these facts an E-commerce fraud investigation can not rely on information from the consumers at all, but instead has to figure out if a suspicious transaction was triggered from the owner of the credit card, or if the transaction was coming from a deceiver.

3.4.2 E-commerce fraud strategies

After fraudsters have got access to leaked credit card information they can come up with the following strategies to trick the E-commerce system:

1. a deceiver owns information about **one** leaked credit card and try to use it for ordering products from **multiple** merchants on the Internet,
2. a deceiver owns information about **multiple** leaked credit cards and try to use them for ordering products from **one** merchant on the Internet,
3. a combination of the two cases above, that can also be related to as a series of the first fraud activity.

In the first scenario, in which the fraudsters try out a leaked credit card for ordering products on Web shops of various merchants, each of the merchants only sees the transaction that takes place in their systems. This will make it more difficult for merchants to detect whether there are fraudulent transactions or not, because they are not aware of the attempts the fraudsters did on other merchant's Web shops.

As each merchant will rely on a PSP or an issuer to verify the credit card payment, it is in the responsibility of these participants to recognize fraudulent transactions in this specific scenario. To be able to do so, the PSPs and also the issuers are monitoring the usage of credit cards and are actively looking for suspicious activities. The fraud prevention mechanisms in place are mostly working on rule-based, and in some cases also on score-based systems running in the internal networks of the PSPs and the issuers. These systems are fed with the information the merchants send with the payment authorization requests and will come up with a decision on each transaction, that is either:

1. Yes, this looks like a fraudulent transaction and has to be blocked.
2. No, this seems to be a valid transaction and should be acknowledged.
3. Maybe, this transaction might be valid, but there is some uncertainty in the validation of it. These edge cases are routed to a human operator of a PSP or an issuer to decide on how to proceed with them.

As a recent study shows the success rate of the fraud prevention systems heavily relies on the techniques used to validate the transaction data (Rana & Baria 2015). The outcome is that ca. 70 to 80% of the fraudulent transactions will be currently recognized as such and blocked successfully. That still means up to 30 percent of fraudulent

transactions could not be identified correctly. For handling these edge cases each organization employs special trained staff, that is operating 24/7 and 365 days a year, for handling them.

As stated in the introductory of this Master thesis in Chapter 1, there is a shift from the offline credit card fraud to the online world. This is also resembled in current figures of E-commerce fraud incidents, which show that it makes up to 85 percent of all credit card fraud attempts and have on average a transaction value of 500 to 600 EUR.

As the PSPs and the issuers do not have any order details, they can only decide on the information given during the payment authorization requests (see Section 3.2). At most they can validate the branch a merchant is operating in, and it might come as no surprise that the fraudsters are regularly using Web shops of merchants, who offer either electronics, clothings, entertainment- or travel-related products and services. These are also the most commonly used sources of *valid* E-commerce transactions, and will therefore make any fraudulent transaction very difficult to detect.

At the end it might be the owners of the credit cards, who detect suspicious activities on their credit card accounts and inform their issuers about them. Based on current regulations and laws the issuers have to rollback the fraudulent transactions on request of the consumers, which means that the merchants will have to cover the costs of the E-commerce frauds (as they are not receiving the money for the products that might have been shipped to the fraudsters already).

Looking at the second scenario of the E-commerce fraud strategies at the beginning of this section, a merchant will receive multiple requests from a deceiver, who is trying out various leaked credit cards for finishing an order. These kind of E-commerce frauds can be recognized at the systems of the merchants based on the same source IP address of the requests, or due to having the same shipping address for orders with different credit cards. Therefore, one can conclude that also merchants must take an active role in the fraud prevention processes (if they do not do so already) and try to minimize the amount of fraudulent transactions taken place in their Web shops. As this scenario is likely be manageable with additional fraud prevention mechanisms at the merchants, and does not need to involve other parties of the E-commerce scenario to figure out the validity of the transactions, this second scenario falls out of scope of this Master thesis.

3.4.3 E-commerce fraud incidents handling

If the fraud prevention systems at the PSPs or the issuers are detecting a suspicious transaction, an operator working in a special department within the organization will be informed about that transaction via a notification on his or her computer. This operator will have to decide whether the transaction looks valid and should be acknowledged, or seems to be fraudulent and has to be denied. To be able to decide this, he or she is going to look into the recent usages of the credit card in question. Whereas it will be easy to recognize that a credit card, that was just being used in a shop in Germany, could not be used in a shop in US or Asia within a short time-frame due to physical constraints in the real world, the same consumer can order products from an US or Asian online retailer with ease within minutes. So these initial geographical constraints, that work so well with real-world usage patterns of credit cards (a proven fraud prevention mechanism called Geo-fencing), will no longer work in the E-commerce scenario.

So the operators have to found their decisions on the transaction information at hand. Initially they can check for the amount that has been paid with the credit card. One can assume that small amounts will be covered by the PSPs or the issuers, who will take over the risk for a false payment authorization. But with an increased value of the items ordered, the PSPs and the issuers are putting back the risk to the merchants in case of any consumer complaints later. At a second glance the operators can also verify whether a consumer has had any business relationship with a merchant in the past or not, as well as check for the retail branch a merchant operates in. But these are weak hints for investigating the validity of an E-commerce transaction as they can be bypassed by the fraudsters with ease (see the explanations in the previous section).

To make a solid decision the operators will have to get in contact with all the merchants a credit card has been used with recently, and have to ask for additional information such as:

- Does the consumer owns an user account with the merchant's Web shop?
- What is the consumer usually looking for in the merchant's Web shop?
- Does the shipping address matches the billing address for that order?
- If not, has the user send orders to this shipping address in the past?
- What has been ordered by the consumer, incl. detailed product information such as brand, model, product categories, ...?

In some cases the PSPs or the issuers have had a business relationship with an online merchant in the past. So the operators from the PSPs or the issuers might already know whom to contact from the support personnel of that merchant. But in most cases the contact persons might not be known to them, so they have to send a request to the general support staff via the contact forms on the merchant's Web site.

Getting the right information will still take time, because the correct addressees from the support departments of the merchants are unknown, the merchants do not have specialized staff at hand to handle these kind of inquiries, or there might be misunderstandings on handling a request due to language barriers or different incentives between the participants. Additionally the operator, who is responsible for such a case, has to collect all available information from these merchants, notes them down and tries to build a "big picture" out of them. In case the initial information received from one of the participants have not been enough, the operator will have to get in contact with the support personnel again. This can result in a lengthy sequence of communication attempts and question-response processes between an operator and the online merchants concerned. Due to this, getting an in-depth overview of suspicious credit card usages in the E-commerce scenario is likely taking hours if not days or weeks. That is definitely way to much time and effort to look into any of these fraudulent transactions in detail. Therefore one can assume that an in-depth analysis of any suspicious transaction will not take place today; instead most of these transactions will be acknowledged without any doubt after a first short look and plausibility check by the operators.

Still the merchants as well as the PSPs and the issuers have a high incentive for increasing the success rates of their fraud prevention mechanisms, and for keeping the numbers of successful fraudulent activities low. For the PSPs and the issuers there are regulations stating that at maximum only one thousands of the overall transactions[1] can be fraudulent. This keeps the pressure on these financial institutions to invest in fraud prevention techniques for being able to stay in business. For the merchants it is also of high interest, that a fraudulent transaction can be resolved before a deceiver receives the ordered products. In the worst case scenario just *one* successfully performed fraudulent transaction in an E-commerce shop will trigger hundreds if not thousands of subsequent attempts from other fraudsters, as past experiences have shown.

[1] Note: numbers stated are valid for the EU.

3.5 Scope of this Master Thesis

As laid out in the previous section, the most interesting E-commerce fraud scenario is the one, in which fraudsters use leaked credit card information to order products or services from various merchants on the Internet. This is currently most likely to be successful, because there is a lack of information on the side of the merchants as well as the PSPs and the issuers. Each of the affected merchants just noticed the transaction that takes place in their own Web shop, without knowing about the other attempts the fraudsters do on the Internet. The PSPs and the issuers will both notice the active use of a credit card on different Web shops though, but do not have any transaction details. Therefore they could not correlate the data from these transactions to check for suspicious activities.

Based on the current credit card usage patterns of the fraudsters, who will try a leaked credit card in commonly used Web shops, it is more likely that these fraudulent transactions will not be recognized on time by the existing fraud prevention techniques in place.

A simple approach to solve these issues would be to just share more information of the ongoing transactions between the merchants, the PSPs and the issuers. This approach might be subject to fail though, because adapting and harmonizing the communication interfaces between the Web shops from various online merchants and the Web Service interfaces of different PSPs and issuers are an enormous undertaking. Any attempt will likely not succeed due to different notions of the communication patterns and data structures exchanged between all relevant participants.

To solve these problems this Master thesis will look into the information sharing issues in detail and try to come up with a solution to answer the most important question of this scenario:

Is this transaction really a valid E-commerce transaction?

Looking into the stakeholders, who can provide useful information to decide it, one will come up with:

1. **Merchants**, who can provide additional information of each E-commerce transaction in question.
2. **PSPs/issuers**, that have information about the credit card usage patterns and the original credit card owners.

3. **LSPs**, who can offer information about whether an order has already been shipped or not, and in the former case to whom it has been handed over.

Its important to point out, that parts of the shared information are confidential or business-critical to at least one of the stakeholders involved. Due to this fact the data sharing has to be secured, and access to the resources has to be granted to selected participants of the scenario only. This Master thesis will focus on the data sharing, collecting and combining aspects of the collaborative system. A detailed discussion of the security aspects of it, incl. how to restrict access to the data with available techniques such as OAuth, is out of scope of the thesis though.

Additionally, as a CSCW system *always* consists of a social and a technological component introducing such a collaborative system into an existing organization will raise issues of user acceptance and adaptation to business processes that are also left out of the discussions in this Master thesis.

4 Theoretical Foundations

This chapter will lay out the theoretical foundations for the to-be-designed collaborative system. It will start with an investigation of the CSCW system theory followed by a detailed examination of the Semantic Web standards such as RDF, OWL and SPARQL. Last but not least the chapter will look into the core concepts of P2P communication technologies and protocols such as WebRTC.

4.1 Computer-Supported Cooperative Work

This section gives a short introduction to the theoretical foundations of CSCW systems. It starts with an overview of the research field itself, followed by a description of the types of CSCW systems available. After that it explains the concept of shared information spaces in detail.

4.1.1 Fundamental aspects

CSCW is "a *generic* term which combines the understanding of the way people work in groups with the enabling technologies of computer networking, and associated hardware, software, services and techniques" (Borghoff & Schlichter 2000, pg. 92). It is part of the research field of *Cooperation Systems*, which emerged during the early 1980s with the understanding that a multi-disciplinary approach for designing IT systems is needed for the success of such systems. As such the research field looks into the usage of applications to support group work in an organizational setting, the effects of such a system on individual users, as well as how the applications have to be adapted for the context of the group. Therefore the studies of Cooperation Systems are consisting of a *social* part as well as a *technical* part, and are looking into the interrelationship between them for certain aspects of work in general, and explicitly for communication and cooperation in a team (Grudin 1994).

These systems are generally focused on the concept of human-centered computing that wants to establish technology and work methods to improve processes and results of the work, while also improving the human conditions at work. A "work system" in such a sense describes the process of human work that consists of goal-directed activities in

a *professional* context. As more work has been moved to information workers another important aspect of such a system is the human cognition, which results in a need of taking human behavior as well as individual goals and knowledge into consideration. Therefore "work systems" focus on activities that is done by a group of people, a team, an organization or a society, and includes social factors like knowledge, goals, tasks and work of individuals or subgroups. These "work systems" are getting more and more complex as the problems human have to deal with are getting more difficult; in terms of their dynamic, nonlinear, interactive and simultaneous nature. Therefore humans have to continually adapt, take over different roles, and are engaged in various activities, which include management of the technologies used and handle the issues they introduce (Hoffman et al. 2009).

That said a "work system" in this sense has two possible outcomes: the products and services created together by humans and/or machines and the sociological and psychological consequences as a result of being part of the process. The objective of a *sociotechnical system* is to optimize both outcomes (see Figure 4.1).

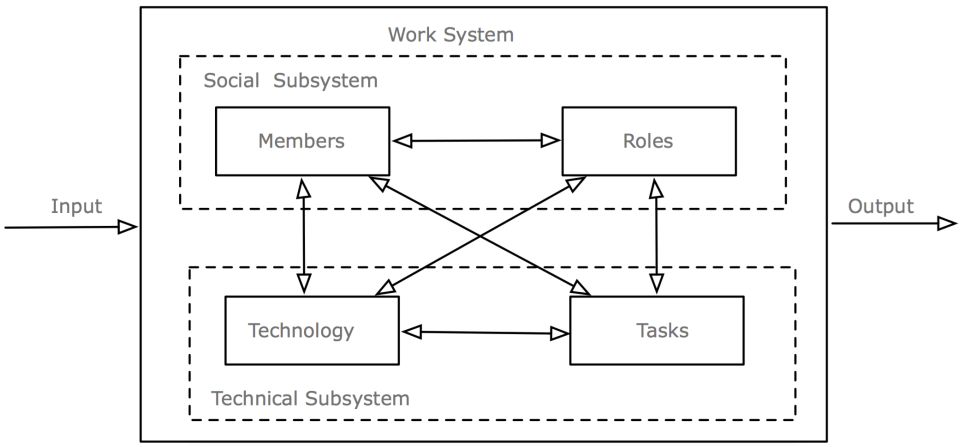

Figure 4.1: A sociotechnical work system (**?**)

To sum up *sociotechnical* refers to the interrelatedness of social and technical aspects of an organization. The sociotechnical theory is founded on two main principles (Koch 2008):

- The interaction of social and technical factors creates the conditions for successful organizational performance. This interaction consists partly of linear "cause and effect" relationships and partly from "non-linear", often unpredictable rela-

tionships. Whether designed or not, both types of interaction occur when socio and technical elements are put to work.

- An optimization of each aspect alone (socio or technical) tends to increase not only the quantity of unpredictable relationships, but those relationships are injurious to the system's performance.

The success of a *collaborative system* depends highly on the level of active use of the system by its users. To improve the situation the system has to offer a clear balance between efforts and benefits for all of its users, has to communicate those clearly to its users, and require a human-centered user interface as well as a good integration into the context of its users (Koch 2008).

4.1.2 Classification of CSCW systems

In general CSCW system can be classified based on the type of communication they support, the kind of applications they made possible as well as according to the "3C model".

In a distributed team environment the style of communication could be *synchronous* or *asynchronous* depending on the dimension of time. If the communication takes place at the same time the communication is synchronous, otherwise asynchronous. Another aspect that needs to be taken into account is the place. The team can be either co-located or geographically dispersed, which have a huge impact on the type of communication suitable. Taking both dimensions into account leads to the quadrant shown in Figure 4.2.

Additionally it is possible to group the CSCW systems based on the "3C model" as visualized in Figure 4.3 into (Borghoff & Schlichter 2000, pg. 125):

- **communication support:** for a two way exchange of information between different team members,
- **coordination support:** for management of shared resources such as meeting rooms, network printers, file shares, ...,
- **collaboration support:** to enable members of a group work together in a shared environment to reach a predefined goal.

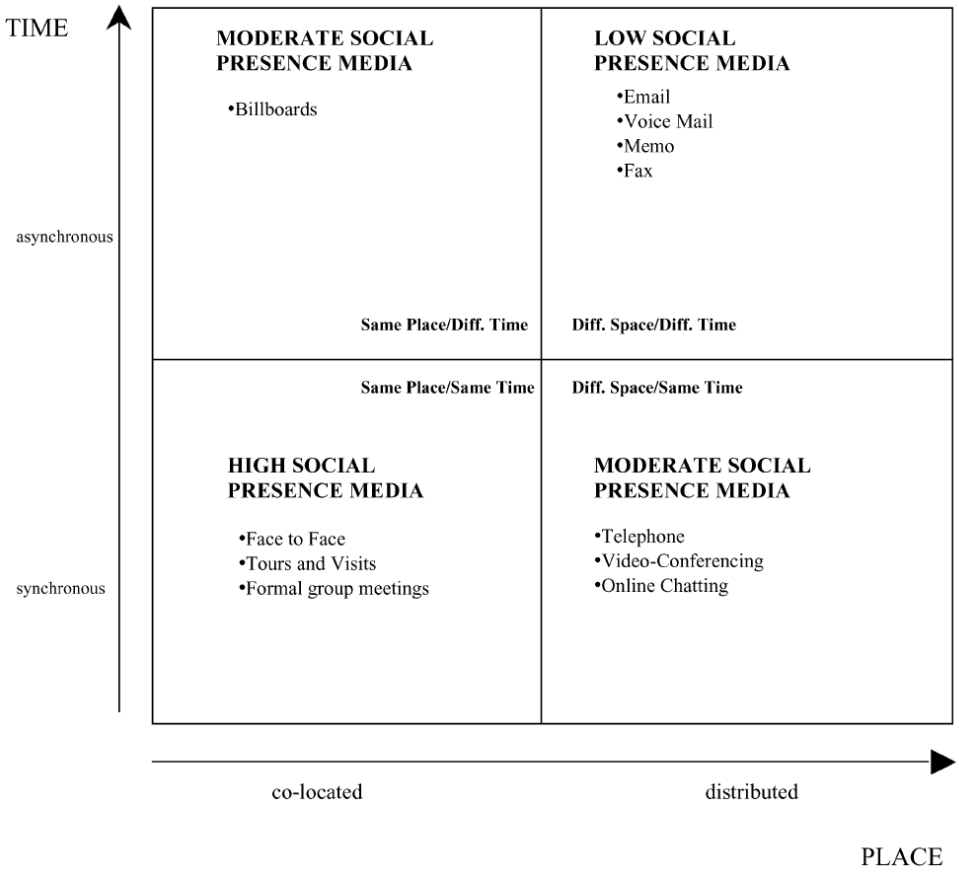

Figure 4.2: Time/Place Matrix (Robert & Dennis 2005)

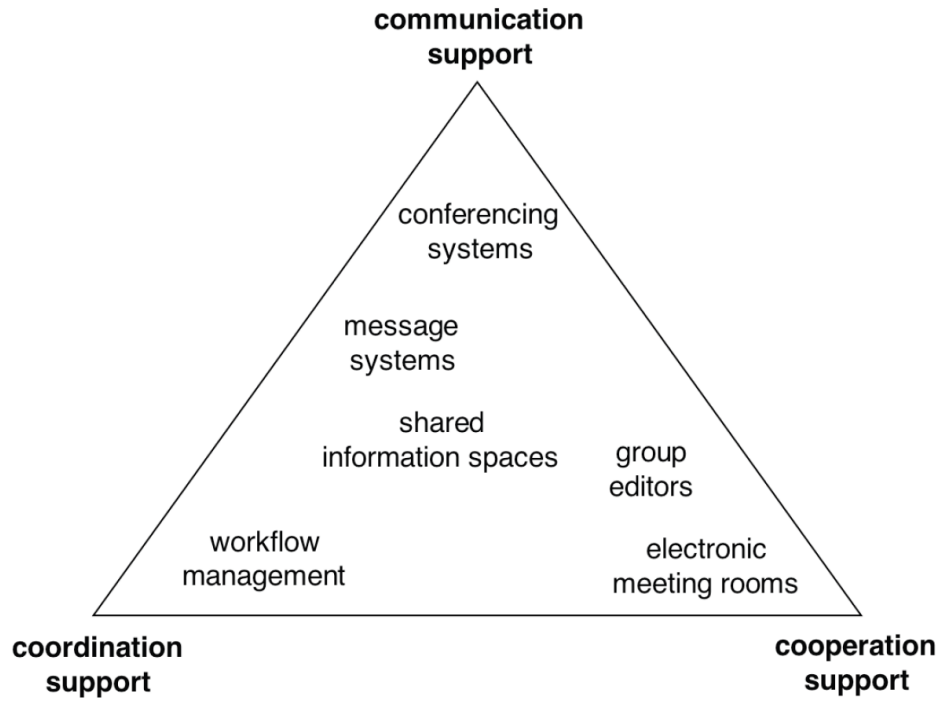

Figure 4.3: The 3C Model (Koch 2008)

Typical application classes for CSCW systems might be (Borghoff & Schlichter 2000, pg. 119-120):

- **message systems:** allow to exchange textual messages between team members asynchronously; modern systems allow sending of other digital artifacts such as images and documents as well (e.g. instant messengers such as Microsoft Skype or Slack),
- **group editors:** allow collaborated work on some kind of shared document or artifact; editing of the shared document can be either allowed synchronously at the same time, or also asynchronously at different times (e.g. collaborative word processors such as Microsoft Word Online or Google Docs),
- **electronic meeting rooms:** allow multiple participants to work within a shared workspace or on a shared whiteboard, and offer support for ad-hoc brainstorming, idea generation and group decision making,
- **shared information spaces:** allow participants to access information at any place any time as well as to share information with others (e.g. Microsoft Sharepoint)

4.1.3 Shared Information Spaces

In a CSCW system shared information can take over two important roles: on the one hand they can transfer knowledge and facts between participants and on the other hand they resemble intermediate or final results of the group works themselves. In case of business critical information the system also have to provide a log of activities as well as a history of all artifacts generated or manipulated with it over time (Borghoff & Schlichter 2000, pg. 295).

The kind of artifacts that are shared within such as collaborative system is not further specified, and can range from information, events or object representations from the real world to internal terms or objects of the working group. Whereas the former can be described extensively and usually do not need further interpretation, the latter one will need an interpretative component to define and communicate their intended meanings between group members. This common interpretation of terms and objects is even more important if the collaborative system is working across time and space boundaries, because co-located group member are generally having the same understanding of terms and objects due to being in the same (working) context and environment. Based to this fact, information that has to be shared between dispersed

group members have to be refined and packaged in a way that enable the receiver to "unpack" the information and be able to re-create the original context, in which the information was created. Therefore the information in such a system is not just coming out of a shared database, but also involves the joined interpretation of it by all the actors involved (Bannon & Bødker 1997).

If information from different sources come together in a shared information space the collaborative system should support a nonlinear, exploratory way to retrieve and navigate through the information space to enable participants to browse and ascertain the concepts and their relations individually. A valid proposal for such a system is based on *Hypertext*, because of its generic approach for the construction of nonlinear, computer-supported material that users can display and navigate on their screen in a nonlinear fashion. Hypertext systems are providing information that is distributed over a network of nodes, which make up the information space. Therefore, the information is divided into small, logical information units (aka nodes), in which references (aka links) are pointing to relevant or related units from the shared information space (see Figure 4.4). Users can navigate the information space along these links (Borghoff & Schlichter 2000, pg. 295-307).

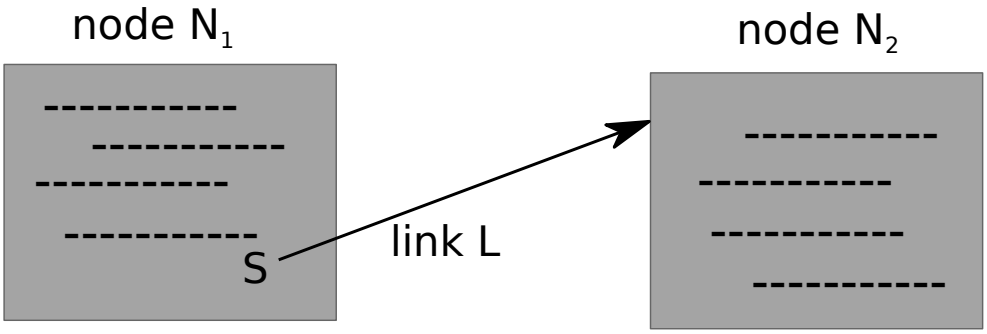

Figure 4.4: A link between two nodes (Borghoff & Schlichter 2000, pg. 303)

A Hypertext system usually consists of three layers (Borghoff & Schlichter 2000, pg. 301-302):

- **database layer:** persists and manages the Hypertext information in a way that allows fast access and retrieval of information units,
- **Hypertext abstract machine:** creates an information network from the information units and their relations (aka links between them),
- **presentation:** displays individual information units on screen and allows the user to navigate the information space (via the links)

The link specifications can be either part of the node content itself (e.g. as in the HTML standard), can be completely separated from the contents of the nodes (e.g. as navigational elements such as “previous” or “next”), or can be collected and presented in some kind of general overview such as a table of contents (Borghoff & Schlichter 2000, pg. 304-306).

4.2 The Semantic Web

The *Semantic Web* initiative strives for a better integration of distributed data from various publishers on the Web with the objective to enable new kinds of Smart Web applications. To achieve this goal, the Semantic Web delivers the infrastructure for this vision in form of various standard specifications such as RDF, RDFS, OWL, SPARQL, ..., which are introduced during the course of this section. Before going into the technical specifications of each of them, the section shows the fundamental aspects underlying the (Semantic) Web as a whole.

4.2.1 Fundamental aspects

The Semantic Web builds on the fundamentals of the existing World-Wide Web, especially (Allemang & Hendler 2011, pg. 4-11):

- **AAA-Slogan:** “Anyone can say Anything about Any topic”. The Web does not restrict or control what people can post or publish on it. It is in the responsibilities of the readers to decide whether they can trust information from a specific source or not.
- **Open World Assumption:** as the amount of information on the Web is limitless, and new information is published every day, one must always assume that there are new information available that one does not know yet. As of this one can never be sure to have all facts at hand. New information can be published at any time that can give additional insights to the topic.
- **Non-unique Naming Assumption:** there is no central authority, who is responsible for providing unique identifiers for entities on the Web. Due to this fact different URIs might refer to the same virtual entity or real-world object.

Instead of making information on the Web available for human consumption *only*, the Semantic Web is trying to make the information on the Web accessible (and readable) to machines as well. This will allow the integration of information across Web sites, and enable a distributed, interlinked “Web of Data”. The major design principles to achieve this objective are (Antoniou & Van Harmelen 2012, pg. 1-22):

1. make structured and semi-structured data available in standard formats,
2. make individual data elements and their relationships accessible on the Web,
3. describe the intended semantics of the data in a machine readable format

The data model of the Semantic Web is build upon labeled graphs with objects and their relationships. Objects are modeled as *nodes* and their relationships as *edges* between them. To express these graphs of related objects, the Semantic Web has to:

- formalize the syntax of the graph in RDF (see Section 4.2.2),
- use URIs to identify individual data items and relations,
- use ontologies to represent semantics of the entities. Ontologies can be lightweight RDFS definitions or expressive descriptions in the OWL language (see Section 4.2.3).

Initially it was tried to solve the data integration aspect on the Web with the exchange of XML-based messages, but though the XML format is more machine readable as HTML it still lacks the semantic of the data transmitted. Therefore the Semantic Web defines the RDF format as the basic data exchange format of it. Still the RDF format was initially based on the XML specification. To formally describe the existing terms and their possible relationships within a domain the Semantic Web relies on an ontology specification. These specifications are either expressed in RDFS or uses the more expressive OWL language; both of them are meta-description languages, which allows the definition of domain-specific knowledge representations based on the concepts found in RDF itself.

As such the Semantic Web is a layered approach as depicted in Figure 4.5.

4.2.2 A Resource Description Framework

When trying to come up with a specification on how to integrate data on a globally dispersed platform such as the World-Wide Web, one will have to answer the following questions:

- **syntax:** How to serialize the data?
- **data model:** How to structure and organize the data?
- **semantics:** How to interpret the data?

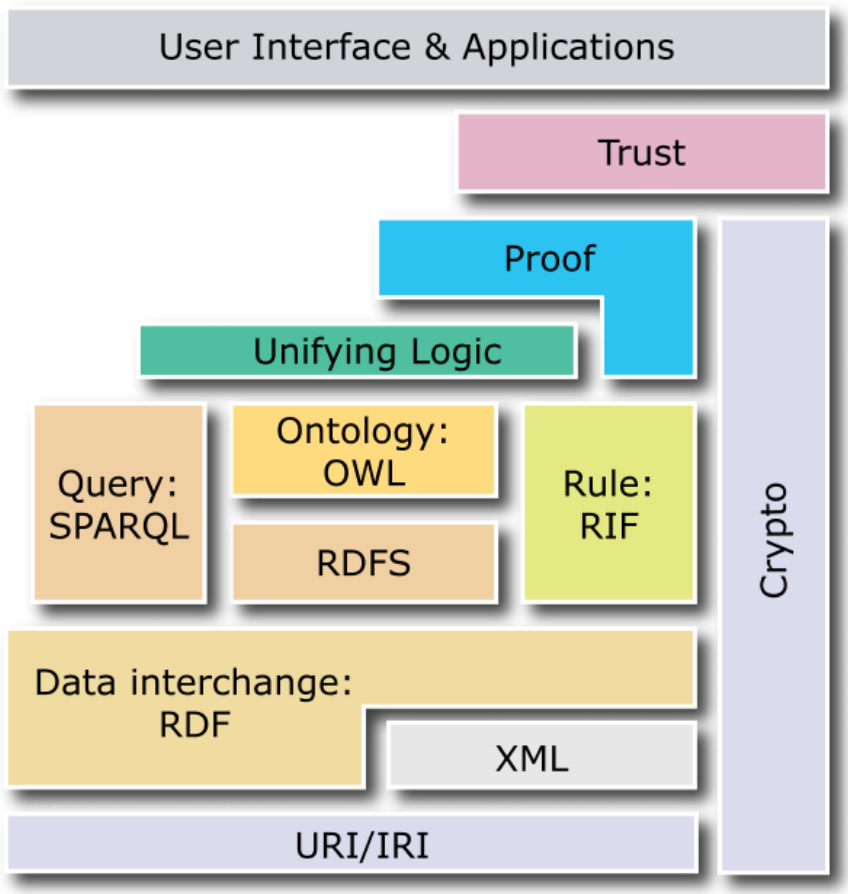

Figure 4.5: The Semantic Web Model (W3C 2013)

Whereas the World-Wide Web is made up from interlinked documents in the HTML format that are specifically designed for rendering information on screen, and will be consumed by a human, the RDF brings a highly flexible data model to the Web. Its basic building block is the *triple*, that is a statement consisting of an entity, an attribute and a value. The individual parts of a statement are also known as subject, predicate and object, and make up a directed graph as shown in the example in Figure 4.6:

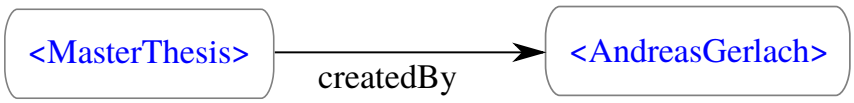

Figure 4.6: A basic example for a triple-statement

In this example the triple consists of the entity “MasterThesis”, the assigned attribute “createdBy” and the value “AndreasGerlach”. The value-part of a triple can contain

either a literal value or refer to another entity (as in the example shown above). Still a problem with this example statement is that the entities are not unique. Based on the given information it is not clear, which specific "MasterThesis" is meant and to whom the value "AndreasGerlach" refers. Additionally the predicate used can have multiple meanings. These ambiguities have to be resolved on the Semantic Web to be able to make these information understandable by machines. To solve these issues the Semantic Web standard specifies that names of entities and predicates have to use a URI to make their intended meanings clear. Literals that can be also used as values, such as numbers, dates and strings, borrow their data type specifications from the XML standard (Wood et al. 2014, pg. 15-38).

Based on this description the foundational elements of RDF can be summarized as:

- **entities:** aka resources or "things of interest" that are identified via URIs,
- **predicates:** aka attributes or properties that specify the relations between resources and are also identified by URIs,
- **literals:** integral values such as numbers, dates and strings that are based on the XML data type specification,
- **statements:** assign a value (either another entity or a literal) to a "entity-predicate" relation,
- **graphs:** are the data model behind RDF and enable a distributed, interlinked "Web of Data".

RDF triples can be serialized into four different syntax formats (Wood et al. 2014, pg. 43-54):

- **RDF/XML:** the original format of the RDF data sets is based on the XML specification. Because of their complexity they are best used with a parser program. For an example see Listing 1.
- **RDFa:** describes how to embed RDF information into existing HTML documents. It allows authors to enrich their Web pages with semantic information by adding a set of predefined HTML attributes to important items within the document. Listing 2 shows a basic example.
- **JSON-LD:** a recent initiative to allow JavaScript developers to use JSON documents to express a RDF data set, see Listing 3 for an example.

- **Turtle:** a human readable serialization format for RDF statements. URIs can be shortened with a predeclared prefix, statements have to be finished with a period. Statements referring to the same entity can be abbreviated via a colon, which repeats the subject from the previous statement, or a comma, which repeats subject and predicate from it. For an example see Listing 4.

```
<?xml version="1.0"?>
<rdf:RDF xmlns:rdfs="http://www.w3.org/2000/01/rdf-schema#"
         xmlns:ex="http://www.example.com/">
  <rdf:Description rdf:about="http://www.example.com/MasterThesis">
    <ex:createdBy rdf:resource="http://www.example.com/AndreasGerlach"
    ↪ />
  </rdf:Description>
</rdf:RDF>
```

Listing 1: A triple statement expressed in RDF/XML format

```
<div about="http://www.example.com/MasterThesis">
  <span rel="http://www.example.com/createdBy"
  ↪ resource="http://www.example.com/AndreasGerlach">
</div>
```

Listing 2: A triple statement expressed in RDFa format

```
{
 "@context": "http://www.example.com/",
 "@id": "http://www.example.com/MasterThesis",
 "createdBy": "http://www.example.com/AndreasGerlach"
}
```

Listing 3: A triple statement expressed in JSON-LD format

```
1 @prefix ex:  <http://www.example.com/> .
2 ex:MasterThesis ex:createdBy ex:AndreasGerlach .
```

Listing 4: A triple statement expressed in Turle format

Interestingly, these different serialization formats for RDF data sets are fully interchangeable. A RDF data set can be easily converted from one serialization format to the other, and merging RDF data sets work with sources expressed in different formats as well.

Coming back to the initial questions that have to be solved for data integration on Web scale, the section showed how the Semantic Web initiative tries to solve them, such as this:

- **syntax:** support for the following formats: Turtle, RDFa, RDF/XML and JSON-LD,
- **data model:** use the graph-based data model of RDF,
- **semantics:** express the semantics of the data in RDFS. This will be the topic of the next section.

4.2.3 RDF vocabularies and Web Ontologies

For describing domain specific semantics of the data in a RDF data set one can either use the lightweight RDFS standard to define available entities and their relationships, or use the more expressive OWL specification from the W3C.

Both specifications are based on the RDF data model, and make use of the following predefined URIs to declare their meanings, see Table 4.1:

Name	Prefix	Used for	Namespace URI
RDF	rdf:	Core RDF framework	`http://www.w3.org/1999/02/22-rdf-syntax-ns#`
RDFS	rdfs:	Define RDF vocabularies	`http://www.w3.org/2000/01/rdf-schema#`
Web Ontology Language	owl:	Define ontologies	`http://www.w3.org/2002/07/owl#`

Table 4.1: RDF vocabularies specified by the W3C (Wood et al. 2014, pg. 41)

An important step in the definition of a RDF vocabulary or Web ontology is to analyze the domain of interest in detail, and come up with a list of objects and their possible relations. One can use the following step-by-step guide as a reference (Antoniou & Van Harmelen 2012, pg. 40-55):

1. Specify the *things* to talk about. These have to be divided into *objects* (aka real entities) and *classes* (aka set of entities). A specific statement containing the predicate "rdf:type" assigns individual objects to their classes.

2. Set up relations that are available between these classes. The relations can either be of type inheritance or composition.

3. Define existing properties (aka predicates) and their hierarchical relationships (if appropriate).

4. Impose restrictions on the kind of properties that can be used on objects. These can include restrictions on the *values* a predicate might take (aka "rdfs:range" restrictions), and restrictions on the possible *subjects* of a predicate (aka "rdfs:domain" restrictions).

The fundamental concepts of the RDFS specification, which are used to model entities and their relations within a domain, are listed in Table 4.2.

Classes	**Used for**
rdfs:Resource	individual resources
rdfs:Class	classes
rdfs:Literal	literals
rdfs:Propery	properties
Predicate	**Describes**
rdf:type	kind of class
rdfs:subClassOf	inheritance between classes
rdfs:subPropertyOf	inheritance between properties
rdfs:domain	restrict the subjects of a property
rdfs:range	restrict the values of a property

Table 4.2: RDFS axioms commonly used to define RDF vocabularies (Antoniou & Van Harmelen 2012, pg. 46-49)

As an example the Listing 5 describes a RDF data set, which is also displayed in Figure 4.7, like that:

1. there is a class “Song” that is derived from the general class “Audio”
2. a class of type “Song” can have a predicate named “title”
3. the predicate “title” is a subproperty of “attribute” and can contain a literal value

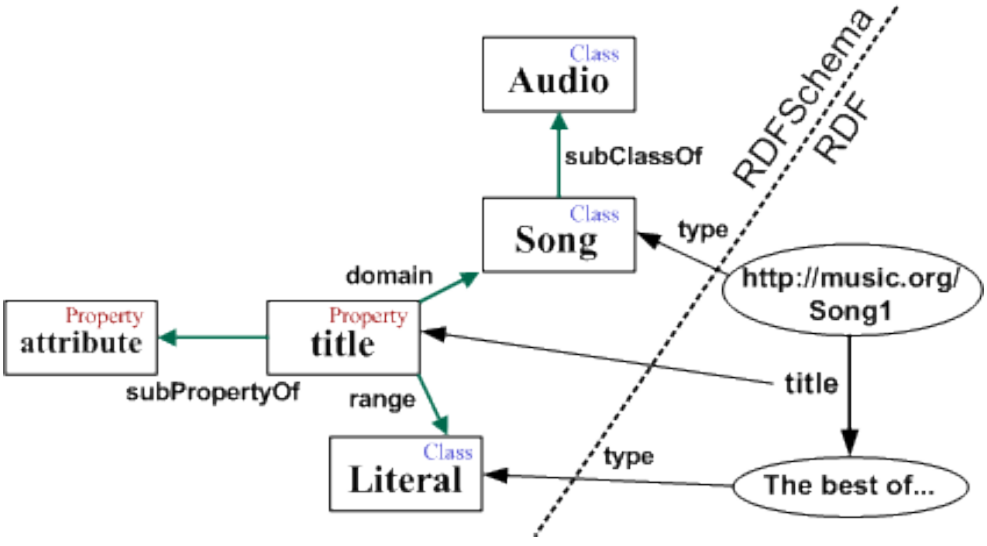

Figure 4.7: RDF Schema sample

```
# define prefixes for URIs
@prefix ex: <http://www.example.com/> .
@prefix rdf: <http://www.w3.org/1999/02/22-rdf-syntax-ns#> .
@prefix rdfs: <http://www.w3.org/2000/01/rdf-schema#> .

# define available classes and their hierarchy
ex:Audio rdf:type rdfs:Class .
ex:Song  rdf:type rdfs:Class;
         rdfs:subClassOf ex:Audio .

# define available properties, their hierarchy and restrictions
ex:attribute rdf:type rdfs:Property .
ex:title rdf:type rdfs:Property;
         rdfs:subPropertyOf ex:attribute;
         rdfs:domain ex:Song;
         rdfs:range rdfs:Literal .

# add specific instances of classes
<http://music.org/Song1> rdf:type ex:Song;
                         ex:title "The best of..." .
```

Listing 5: A sample RDF data set based on Figure 4.7

As is also shown in Figure 4.7 the majority of the domain specific knowledge is described in the RDFS part of the RDF data set. It holds all available classes, predicates, restrictions, ..., and is therefore more abstract and expressive. The RDF part containing the specific instances might be rather small and concrete, consisting of only existing resources with their known predicates. Usually each of the real entities from the RDF part refer to its best matching and most descriptive entity from the RDFS part (e.g. the entity `http://music.org/Song1` in the sample is referring to the *Song* class rather than the *Audio* class).

Its important to note that the meaning of the RDFS predicates "rdfs:subClassOf", "rdfs:subPropertyOf" as well as "rdfs:domain" and "rdfs:range" is *not* to restrict or validate the proper usage of them in a RDF statement, but is rather used to *infer* additional statements in a RDF data set based on their usage. So in the example above, one can infer that the resource found at `http://music.org/Song1` is not only

a “Song”, but also an “Audio” based on the “rdfs:subClassOf” relationship between “Song” and “Audio”. This kind of propagation of RDF statements based on the usages of those RDFS predicates fits well to the Open World Assumption of the Semantic Web (Allemang & Hendler 2011, pg. 125-152).

In addition to the axioms shown above the RDF and RDFS specifications contain further useful entities and predicates, such as shown in Table 4.3:

Classes	**Used for**
rdf:Bag	unordered list of entities
rdf:Seq	ordered list of entities
rdf:Alt	list of alternatives or choices
rdf:Container	superclass of all containers
Predicates	**Describes**
rdfs:seeAlso	links to an external resource that contains additional information about it
rdfs:isDefinedBy	links to the original definition of a resource
rdfs:comment	comments and notes on entities
rdfs:label	human friendly label for entities

Table 4.3: RDF and RDFS supplemental axioms (Antoniou & Van Harmelen 2012, pg. 46-49)

As the two tables (Table 4.2 and Table 4.3) show the RDFS specification contains only basic axioms to describe an entity and its possible relationships as well as the hierarchy between these entities and the relations. It is the fundamental set of constraints that is needed to start with publishing resources on the Semantic Web. As of this it does not include more advanced features that can be of use for combining and reasoning on distributed RDF data sets, such as:

- **Equality/Inequality:** Neither RDF nor RDFS provide a way to specify that two resources or properties coming from different RDF data sets are the same, or are not the same. Though it might be possible to model the aspect of equality with a combination of “rdfs:subClassOf” or “rdfs:subPropertyOf” statements the results are usually not as desired due to the inferencing nature of those axioms.
- **Cardinality:** Predicates defined in RDFS can be used multiple times in statements referring to the same subject or object. This is not desired in situations, in which there is only one possible instance of a “subject-predicate-object” relationship.

- **Transitivity:** Whereas it is possible to express the hierarchy of classes and predicates in RDFS it is not feasible to do so for individual instances. This might be useful when building up a hierarchy of people, e.g. a statement such as "S3 has an ancestor S2, who has an ancestor S1" can lead to the inference "S3 has an ancestor S1".

- **Property Restrictions:** To define whether a predicate is referring to an object or a literal as value is not possible with RDFS, but may be useful for modeling and editing tools.

Therefore, in case there is more expressiveness required to model a domain one can use additional concepts from the Web Ontology Language specification (OWL). It is build on the RDFS specification, but contains additional constraints and classifiers, some of the most commonly used ones are listed in Table 4.4.

Classes	**Used for**
owl:Class	all classes in OWL
owl:FunctionalProperty	allow only one value
owl:InverseFunctionalProperty	allow only one source
owl:TransitiveProperty	build chains of relationships
owl:ObjectProperty	property can hold a resource as value
owl:DataProperty	property can hold a literal as value
Predicates	**Describes**
owl:equivalentClass	equality of classes
owl:equivalentProperty	equality of properties
owl:sameAs	equality of individual resources

Table 4.4: Commonly used OWL axioms (Allemang & Hendler 2011, pg. 153-185)

The Table 4.4 is only showing a small subset of the axioms available in OWL. The Web Ontology Language can be used to express a wide variety of constraints and classifiers on predicates, and the whole specification is based on three parts that are build upon each other (W3C 2004). Still, with an increase of expressiveness in the model the complexity of the reasoning and inferencing engine also grows up, because most of these axioms are used to express inferences that can be drawn on the RDF data set. This Master thesis is restricting the usage of axioms of the OWL specifications to ones shown in Table 4.4.

4.2.4 SPARQL protocol and query language

A kind of query language is required to be able to access specific information in a RDF data set on the Semantic Web. Additionally, the query language has to support the distributed nature of information on the Semantic Web, as well as be suitable for asking information from the graph-oriented data model of RDF. The W3C proposes the SPARQL protocol and query language as a standard way to access and query for information on the Semantic Web. As the name already implies the specification contains two parts: a *protocol* and a *query language*.

SPARQL requires the RDF data set in a *triple store*, which is a kind of database containing RDF statements (another name for it might be *graph store*). The RDF data set is usually inserted into the triple store via bulk load operations or single SPARQL update statements, which like in SQL for relational databases can manipulate RDF statements in the data store. A SPARQL query is usually expressed in a Turtle-like syntax and is send to an HTTP endpoint of the triple store by a client application. The result of the query can contain either a single value, a tabular data stream, or a subgraph of the RDF data set depending on the type of query issued. The structure of a Semantic Web application that uses a triple store (aka RDF store) and the SPARQL query engine is depicted in Figure 4.8 (Allemang & Hendler 2011, pg. 51-60):

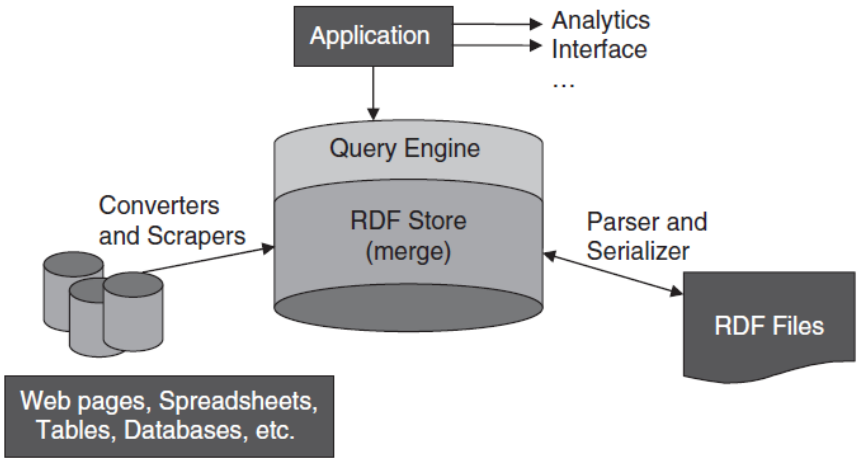

Figure 4.8: Semantic Web application architecture (Allemang & Hendler 2011, pg. 57)

The SPARQL query language has a lot of similarities with the SQL used for querying relational databases. This design decision will ease the transition to the Semantic Web for application developers familiar with accessing data from a relational database. The main difference though is the way used to specify the conditions for a query. This is

largely due to the differences in the underlying data model, which is relational in SQL versus graph-oriented in SPARQL. As of this, a WHERE clause in a SPARQL query has to contain a graph-based representation of the query conditions that have to be matched in the RDF data set. Parts of these conditions can be marked as placeholders, which are referred to in a SELECT statement for generating a tabular data output. These placeholders are marked with a question-mark in the beginning of their name and can be used as placeholder for any part of a triple statement (subject, predicate, object) (Allemang & Hendler 2011, pg. 66-112). E.g. querying a RDF data set containing triples such as the one described in Listing 5 for the titles of all available songs, one has to write a SPARQL SELECT query as shown in Listing 6:

```
# define prefixes for URIs
PREFIX ex:  <http://www.example.com/> .
PREFIX rdf: <http://www.w3.org/1999/02/22-rdf-syntax-ns#> .

# choose to output any title found in graph-pattern
SELECT ?title
WHERE {
  # describe the conditions for the query
  # as graph-patterns that have to match
  # here: it has to be a Song, which has a predicate title
  ?song rdf:type ex:Song .
  ?song ex:title ?title .
}
```

Listing 6: Selecting the title from all songs with SPARQL

Please note that this query defines two placeholders in the WHERE clause: "?song" and "?title", but only uses the "?title" as an output criteria in the SELECT statement. The placeholder "?song" is *just* required to refer to the same node when specifying the conditions in the WHERE clauses.

The WHERE clause in a SPARQL query can include additional conditions that have an effect on the returned information, such as (Allemang & Hendler 2011, pg. 66-112):

- **LIMIT:** specifies the upper limit of results that should be returned from a SPARQL query; e.g. LIMIT 100

- **FILTER:** express additional filter conditions on the result set of a SPARQL query; e.g. FILTER(?releaseDate >“1980-01-01”)
- **UNION:** combine the result sets from different graph patterns into one; e.g. { ex:Song ex:title “A” } UNION { ex:Song ex:title “G” }

Beside the SELECT statement, that returns tabular data, SPARQL also supports the ASK type of query, which checks for the existence of graph patterns stated in the WHERE clause, and will return a boolean value (true or false). This kind of query is commonly used to assert the presence of certain triples in a RDF data set.

Another kind of query is the CONSTRUCT statement, that is used to retrieve a subgraph from a RDF data set, and can also be used to harmonize graphs from different sources, that use different schemata. As a query of type CONSTRUCT will return a new RDF graph it can also be used for basic reasoning functionality ala “if this graph-pattern is found, assume that . . . ”, as well as to help with resolving issues of identifying entities that are described with different URIs (Allemang & Hendler 2011, pg. 88-98). E.g. querying a RDF data set containing triples such as the one described in Listing 5 and mapping the song information to the DublinCore specification (Initiative 2012), one has to write a SPARQL CONSTRUCT query as shown in Listing 7.

```
# define prefixes for URIs
PREFIX ex:  <http://www.example.com/> .
PREFIX dc:  <http://purl.org/dc/elements/1.1/> .
PREFIX dct: <http://purl.org/dc/terms/> .
PREFIX rdf: <http://www.w3.org/1999/02/22-rdf-syntax-ns#> .

# create a new graph with the mapped song information
# here: there is an entity of type dc:Sound that have a title
CONSTRUCT {
  ?song rdf:type dc:Sound .
  ?song dct:title ?title .
}
WHERE {
  # describe the conditions for the query
  # as graph-patterns that have to match
  # here: it has to be a Song, which has a predicate title
  ?song rdf:type ex:Song .
  ?song ex:title ?title .
}
```

Listing 7: Mapping custom song information to the DublinCore vocabulary with SPARQL

Please note that the CONSTRUCT statement consists of a set of triple statements that will make up the resulting RDF graph.

4.3 Peer-to-peer communication

This section explains the core concepts of P2P communication technologies. It begins with a comparison of the benefits and disadvantages of centralized and decentralized Web architectures. After that it shows how P2P communication networks can be structured, different ways to initiate a communication session as well as how data can be transmitted between peers.

4.3.1 Centralized vs. Decentralized Web Architectures

In classical client-server applications the information is stored on a central system (aka server). Clients have to connect to the server and ask for the information. The server

handles the requests from the clients and deliver the information in case a request was valid. Prominent examples of centralized Web architectures are Social Networks such as Facebook or Twitter, in which clients, such as a Web browser or Mobile application, communicate with a Web service, which runs on a server of the organization providing these Social Networks, to access and retrieve Web documents (e.g. HTML, images, audio, video, ...) via the HTTP protocol, as shown in Figure 4.9.

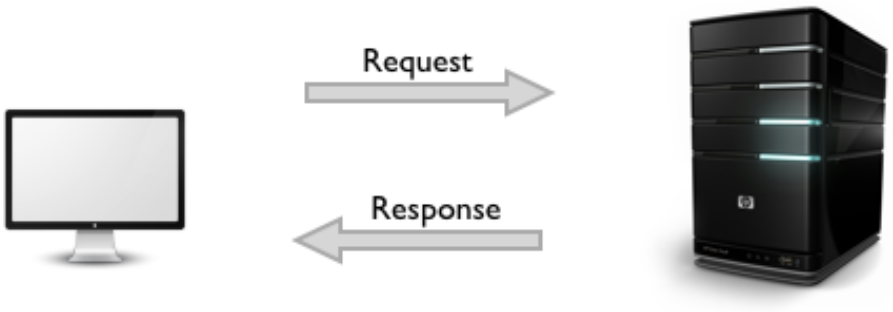

Figure 4.9: Centralized Web architectures as used by prominent Social Networks (Pavan Podila 2013)

As a consequence of this architecture all of the information are centralized and under control of the provider of the (Web) service. This can lead to a variety of problems, including serious issues such as unreliable or no longer existing services will result in a dismissal of all the information stored on them, or privacy concerns for user-generated content stored on those central servers.

In opposite to that a P2P network considers all nodes as equal. This offers the benefits that information can be kept on each node, and each node can provide access to its information to any other node on the network. Due to this the P2P system has an high degree of decentralization, is not owned and controlled by a specific company, and therefore tends to me more resilient to faults, outages and attacks. But due to the distributed nature of it, looking for and accessing information is more difficult. Information in a P2P system have to be indexed in a way so that the correct node is queried for it. Still this index has to be stored somewhere in the system, and the optimal solution for the indexing problem depends on the type of P2P system used (see next section). Additionally, the way new nodes get connected to the system is depending on the type of P2P system used, and might lead to the introduction of special bootstrap or super nodes into the system as shown in Figure 4.10 (Parameswaran et al. 2001).

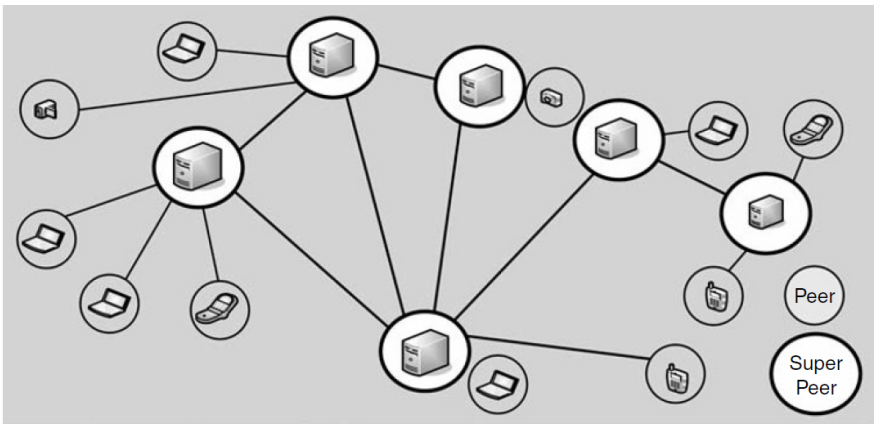

Figure 4.10: A P2P overlay network (Buford et al. 2009, pg. 9)

4.3.2 Classification of P2P systems

P2P system architectures can be classified based on their degree of centralization into:

- **partially centralized P2P system:** rely on a dedicated controller node that maintains the set of participating nodes, host the index of the information available in the system, and controls the overall operation of the network,
- **decentralized P2P system:** does not use any dedicated controller node, but may need to introduce bootstrap and super nodes for maintaining the list of participating nodes and the index of the information available depending on the size of the network.

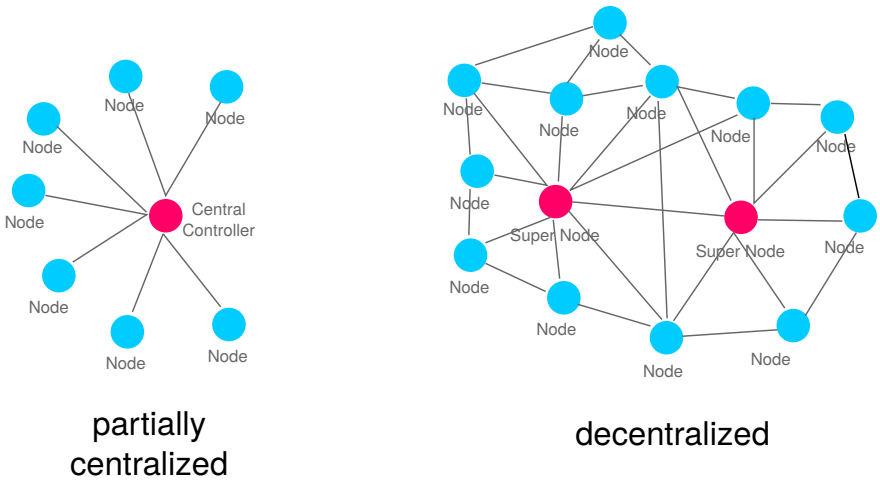

Figure 4.11: Classification of P2P networks

4.3.3 Communication in a P2P network

The procedure to establish a P2P communication depends on the structure of the P2P system. In a *partly centralized P2P system* new nodes join the network by connecting to the central controller first. This central controller has a well-known IP address and maintains the operation of the whole P2P network. Due to this, any new node has to register with the central controller to get introduced to the P2P network. The controller also maintains the information about the overlay network as well as the information about each object and on which node(s) it resides within the network. The overlay is typically following a star-shaped topology with the central controller at the center, see Figure 4.11.

In a *decentralized P2P system* new nodes are expected to obtain the IP address they have to connect to initially via a separate channel (e.g. as a link on a Web site). Depending on the size of the P2P network additional bootstrap or super nodes, which help to set up a new node, are available on the network. These special nodes are generally also consolidating information about the objects available on the peers nearby, which helps speeding up searching and accessing required information. The overlay information of such distributed network can be either *structured*, in which each node receives a unique identifier from a numeric keyspace resembling the responsibilities of that node, or *unstructured*, in which there is no particular network structure, and no constraints are assigned to the nodes of the network.

A *structured overlay* maintain the information within the network more efficiently, because it uses a distributed hash-table to maintain a distributed index and decides the location (aka node) of an object in the network based on its hash-value. In an *unstructured overlay network* the information is typically stored on the node that introduces it. To locate an object a query request is typically broadcasted through the overlay network. Based on the size of the network and the distance between the node asking for and the node holding the information querying and accessing an information on an unstructured overlay network can take some time, and can also flood the whole network with query requests. Therefore, requesting nodes often set the scope of the request, which limits the number of hops that should be done on the network. This will reduce the communication overhead on the whole system. Additionally, introducing super nodes that collect and maintain indexes of their peers nearby can further reduce the number of hops necessary to find the required information, see Figure 4.11 (Rodrigues & Druschel 2010).

4.3.4 Using WebRTC for P2P communication

"Web Real-Time Communication (WebRTC) is a collection of standards, protocols, and JavaScript APIs, the combination of which enables peer-to-peer audio, video, and data sharing between browsers (peers)" (Grigorik 2013, pg. 307). Although this new W3C standard usually stands for in-browser video or audio conferencing without the need of proprietary browser extensions, it also offers ways to exchange arbitrary messages or binary data between participating peers in a distributed Web application. Due to being an open Web standard WebRTC is available in many current Web browsers directly, and is widely adopted as a standardized and open way to establish a P2P communication between clients of a Web site, or from within a Web application. The standard wraps a lot of the complexities of establishing peer-to-peer communication channels and transmitting data into three primary APIs (Grigorik 2013, pg. 307-308):

- **MediaStream:** for acquiring access to and retrieve data from local audio and video devices,
- **RTCPeerConnection:** for establishing a peer-to-peer connection between clients,
- **RTCDataChannel:** for transmitting arbitrary application data

To establish a data connection between peers a Web application has to create a RTCPeerConnection object first, before it can create a RTCDataChannel to exchange messages on it. Establishing a P2P connection between globally dispersed peers on the Web is not a trivial task and has to provide fallback solutions in case of P2P connectivity issues due to firewall or NAT services used by some of the peers, which usually prevent clients to connect to each other directly. Fortunately, the W3C standard is taking care of these steps during the initiating of a WebRTC connection by utilizing the ICE protocol. After being able to open a connection to another peer, a communication session has to be created. For that the communicating peers have to negotiate on protocols, encodings, and additional functionality required for the P2P communication tasks at hand. The WebRTC uses the SCTP to exchange application data between peers (Grigorik 2013, pg. 315-330). It has the following set of features (Grigorik 2013, pg. 342):

- **Reliability:** the data channel can be configured to use either reliable or unreliable delivery of packages,
- **Delivery:** the data channel can be also configured to support either in-order or out-of-order delivery of packages,
- **Transmission:** the transport of data is message-oriented,

- **Confidentiality/Integrity:** all application data transmitted between the peers is encrypted to guarantee confidentiality and integrity of the data exchanged.

For a purely data transmission channel one can also disable any audio and video transfers during the setup of the communication session (see Listing 8).

```
// create signaling channel for negotiating between peers
var signalingChannel = new SignalingChannel();
// create p2p connection object
var pc = new RTCPeerConnection(iceConfig);
// create a named data channel with unreliable transfer option
var dc = pc.createDataChannel('namedChannel', { reliable: false });
// set media constraints to disable audio and video transfers
var mediaConstraints = {
  mandatory: { OfferToReceiveAudio: false, OfferToReceiveVideo: false }
};
pc.createOffer((offer) => { ... }, null, mediaConstraints);
```

Listing 8: Establishing a pure WebRTC data connection (Grigorik 2013, pg. 349)

Once a data channel has been established between the peers application data can be exchanged between them via message passing, as shown in Listing 9:

```
// initial channel and session setup and negotiation
...
// register callback for handling remote data channels
pc.ondatachannel = handleChannel;
function handleChannel(dc) {
  dc.onerror = (error) => { /* handle error event */ }
  dc.onclose = () => { /* handle close event */ }
  // exchange application information with peer
  dc.onopen = (evt) => { dc.send(msg); }
  // act on data received by another peer
  dc.onmessage = (msg) => { console.log(msg.data); }
}
```

Listing 9: Message-oriented communication via a WebRTC data channel (Grigorik 2013, pg. 346)

5 Concept for a system supporting E-commerce fraud investigations

This chapter looks specifically into the concept of a collaborative system, that will improve the situation described in the scenario in Section 3.5. To do this, the chapter will discuss the overall concept of such a system on an high level, without going to much into implementation specific details. At the end, the chapter will have answered the question of what the system is and what should be able to achieve with it. In addition to these discussions, the chapter will further look into existing design approaches and analyses why they are of no use for the specific scenario of this Master thesis.

5.1 Collaboration on E-commerce fraud incidents

Based on the explanations in Chapter 3, and especially the scope definition for this Master thesis in Section 3.5, the collaborative system for investigating E-commerce fraud incidents have to answer the central question:

Is this transaction really a valid E-commerce transaction?

Looking into the stakeholders, who can provide useful information to decide it, one will come up with:

1. **Merchants**, who can provide additional information of each E-commerce transaction in question.
2. **PSPs/issuers**, that have information about the credit card usage patterns and the original credit card owners.
3. **LSPs**, who can offer information about whether an order has already been shipped or not, and in the former case to whom it has been handed over.

Ideally each of those participants would make parts of their internal databases available for the others to access and query for information in a shared information space. That would allow those stakeholders, who have to authorize or validate a suspicious credit card transaction, to analyze all available information as depicted in the Figure 5.1.

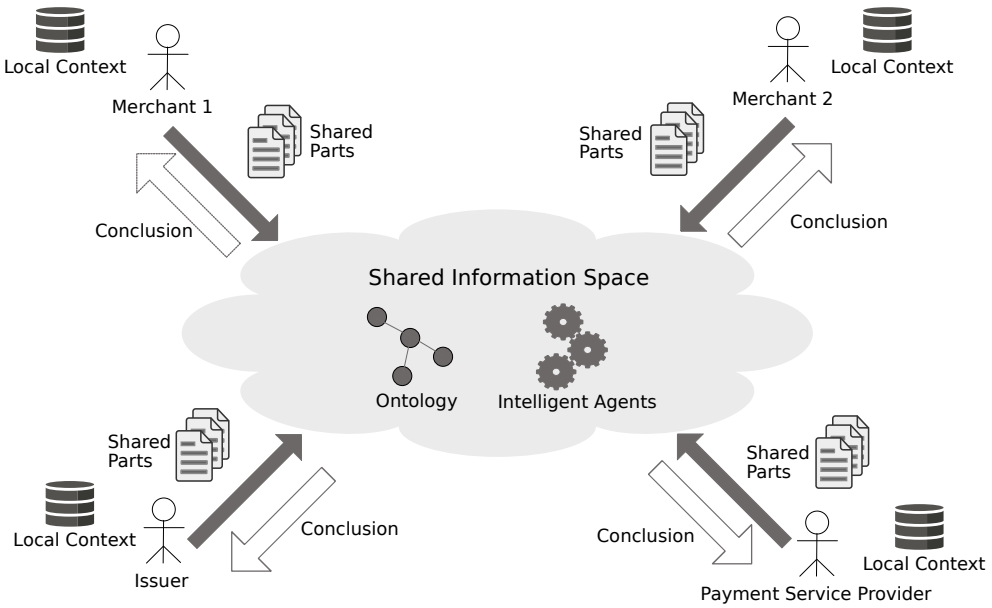

Figure 5.1: High-level concept of the system

In this figure one can see that the relevant parties are providing access to parts of their internal local context information within a shared information space. The collaborative system should allow participants to communicate and collaborate on the E-commerce fraud incidents from different places at the same time (see Section 4.1). Due to the fact, that data from various sources have to be combined into a shared understanding of the E-commerce activities of a consumer, there is a need to harmonize and transform the information from each participant into a common data model to be able to analyze the combined data set. Based on the shared understanding of the E-commerce activities, that have been done with a credit card recently, a set of intelligent agents (aka analysis software) can assess them and present their findings, which can be valuable to any of the participants of the collaborative system.

5.2 An ER model for E-commerce transactions

Based on the analysis of the information each stakeholder holds and transmits to others in Section 3.2, the following ER model can be conducted for E-commerce transactions (see Figure 5.2). This figure shows not only the relevant information from the local contexts of each stakeholder, but also how they can be combined within a shared information space.

As the figure also shows there are *shared information tokens* that will be exchanged between various stakeholders. Those can be used in the collaborative system as a

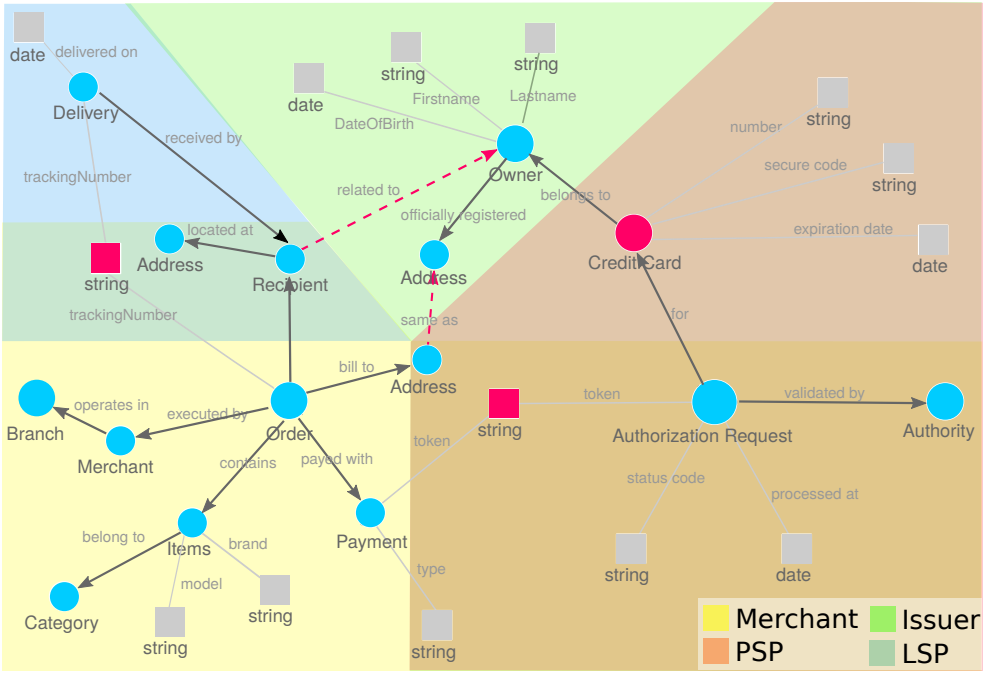

Figure 5.2: Data relations in the E-commerce scenario

reference for joining the distributed pieces of information into a combined view of an E-commerce transaction. There are actually three important tokens:

1. **payment token**: shared between merchants and PSPs,
2. **tracking number**: shared between merchants and LSPs,
3. **credit card**: shared between issuers and PSPs

In addition to these tokens Figure 5.2 also shows the important validation criteria. These are connections that have an influence on the decision whether an E-commerce transaction is evaluated as suspicious or not. The two main criteria are:

1. **billing address-to-owner address**: the billing address of the order has to match the registered address of the credit card owner
2. **recipient-to-owner**: the recipient of the delivery has to be related to the owner of the credit card

Whereas the first criteria can be examined during the payment authorization process of an E-commerce transaction based on the information transmitted between merchants and PSPs or issuers, the second one is more difficult to validate (or can not be verified at all). The only check the LSPs are able to do, before they are handing over the

packaged items to the recipients, is to verify that they are the ones mentioned in the shipping address information of the order. If a recipient is somehow related to the owner of the credit card used for paying an order, or just a deceiver misusing a credit card can not be confirmed by the LSP.

Also merchants, PSPs and issuers have no possibility to check for this criteria. Whereas the merchants are able to validate whether a consumer has send items to a shipping address before, they can not restrict consumers to choose only validated recipient addresses for their orders. Doing so would have negative impacts on the business success of the online merchants. The PSPs and issuers can not analyze this situation either, because both participants will not receive any information about the delivery address of an order with the payment authorization request from a merchant.

But just sharing the fact whether the shipping and billing address of an order is different or not between the relevant stakeholders is not enough. Although this information is necessary, it is not sufficient to make a decision about suspicious transactions. Other necessary information are whether the consumer has send orders to this shipping address before, and the information about the content of the current order. Nevertheless, as mentioned in Section 3.5 looking at the transactions of just one of the online merchants is not enough either to solve the E-commerce fraud scenario, that this Master thesis looks at. More sophisticated analyzing capabilities are required for the collaborative system to be helpful for the E-commerce fraud investigation.

5.3 Analyzing E-commerce transactions

Based on the explanations in the previous section the idea is to link the transaction information from various merchants, LSPs, PSPs and issuers together into a shared information space to be able to analyze if there are any orders that look extraordinary, and are likely not being made by the owner of the credit card to a certain extend. Due to this proposal the collaborative system will also have to use statistical evaluations and probabilities to find and rate suspicious activities. Starting with the credit card in question an issuer can query for the order details of all the transactions that have been done with the credit card online recently. To be able to do that an issuer will likely have to query the PSPs for the payment tokens first, before asking the affected merchants for order details to any of those payment tokens. At the end each online transaction can be mapped into an ER model like the one shown in Figure 5.2, building up a large graph of entities and their relationships, which has the specific credit card

in the center of it. An abbreviated sample graph of this procedure can be seen in Figure 5.3.

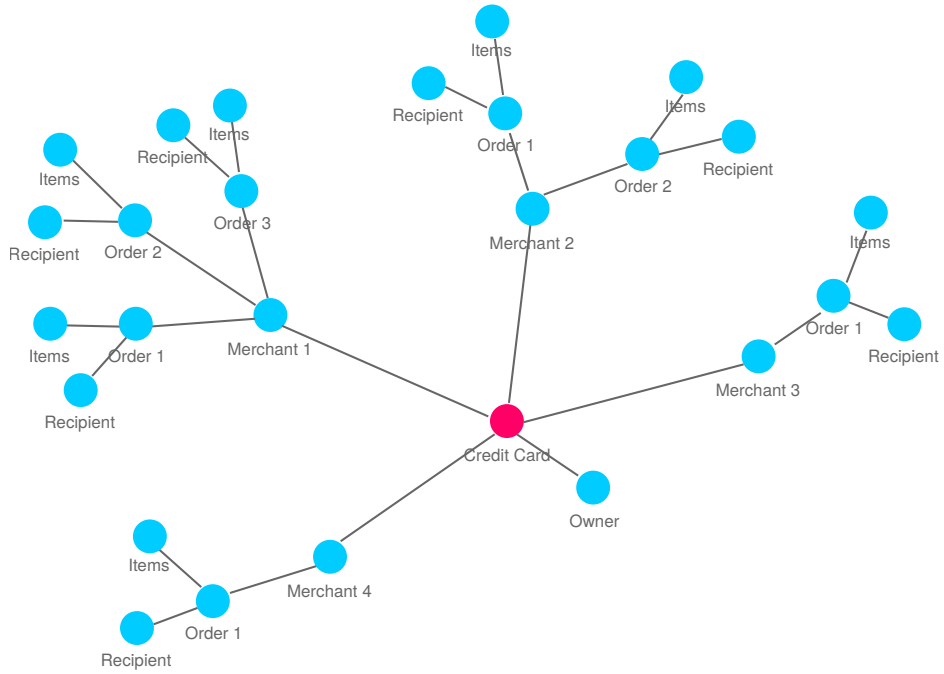

Figure 5.3: Building clusters of E-commerce transactions by merchant

As shown in this figure the transactions will be clustered by merchants first. But collecting the various order information into one combined data set is just the beginning of the E-commerce fraud incident analysis. Based on the information received an issuer can already filter out transactions that have been shipped to different addresses than the one the credit card owner is registered for. Particularly for those edge cases it might be worth to ask for additional information from the affected merchants to be able to figure out if the consumer has used one of these shipping addresses before. As a result the existing data set can be further enriched with supplementary transactional information from merchants at any time if needed. In addition to the address information an issuer can also analyze the item information (incl. category, brand and model) of each order to check for malicious activities.

But as already stated analyzing the cluster of transactions on a merchant by merchant basis will not be sufficient to come up with a solid decision about a suspicious transaction. This is mostly due to the usage pattern of the fraudsters that have been explained in the scenario in Section 3.5. Based on this description the various order details from

the merchants have to be mapped and linked against each other, so that the initial graph of transactions, which is clustered by merchant, can be easily transformed into complementary representations, which use different criteria to cluster the transactions, such as recipient addresses, branches of merchants, or product-related information.

This reshaping of the transactional details can lead to new insights about the "normal" shopping behavior of a credit card owner, and can make deviations from this behavior visible. By using a clustered graph to visualize the combined data set on screen the exploratory nature of knowledge generation and perception will be supported, and therefore this kind of representation can help speed up the investigation of E-commerce fraud incidents. An example visualization of a clustered graph, which groups information together based on a chosen criteria, is depicted in Figure 5.4. The different colors in this figure can represent different sources of information (e.g. E-commerce transactions from various merchants). In this example information that stands out from the "normal behavior" can be found in the right area of the figure.

Figure 5.4: An example visualization of a clustered graph (Vis.js)

In addition to these clustered graph visualizations the collaborative system can also support the E-commerce fraud investigation by switching the type of representation based on the chosen criteria; e.g. when clustering transaction details based on location

information such as shipping addresses the system can present the information as a heat map on a chart as is displayed in Figure 5.5.

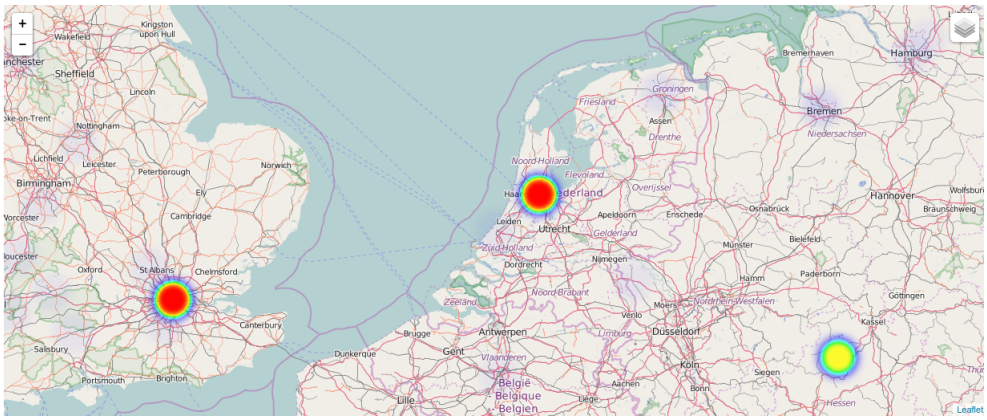

Figure 5.5: Heatmap displaying clusters of location-based information

To conclude the system have to support the collection and combination of E-commerce transaction information from various sources into a linked data set using a graph-oriented data model. This graph can further be analyzed from multiple view points to validate if there are any transactions that stand out from the "normal" shopping behavior of the credit card owner. The starting point for the investigation is a sequence payment tokens of recent credit card activities that an issuer can provide to the PSPs. The linked data set will initially collect and cluster the information from each merchant based on this list. In case there are already suspicious information in one of these clusters, an issuer can ask for further details and enrich that specific cluster with additional order information for this consumer and that merchant. In the final step the system has to do the mapping and linking of the order detail information between each merchant to allow subsequent analyzing and clustering of the transaction details based on various criteria.

5.4 Evaluation of existing design approaches

When trying to solve issues of information integration between organizations there are already existing solutions, that have to be examined whether they might fit the E-commerce fraud investigation scenario or not. This section looks into common existing approaches to collect and integrate information between IT systems.

5.4.1 ETL processes

To begin with, retrieving, transforming and combining data from multiple dispersed data sources is not a completely new problem, and is actually part of "Extract-Transform-Load" (ETL) processes *within* an organization. The basic idea is very much the same as in the concept shown in this thesis; namely to get as much information as possible from the various databases that are in use within a company, harmonize (aka transform) the data from each of them into a shared data model, and use the cleaned up and combined information repository for doing advanced business analytics and predictions later. Data within an organization is created and maintained by different business-related software tools. Each of these will usually store the information into their own database using a vendor-specific database schema. Other business-relevant information might be stored in structured files, sometimes using a proprietary format such as Microsoft Excel. Each of these data sources have to be accessed, the valuable information have to be extracted and mapped against each other, before the analysis of it can begin in a separate data store that holds the combined data set. The whole process is visualized in Figure 5.6.

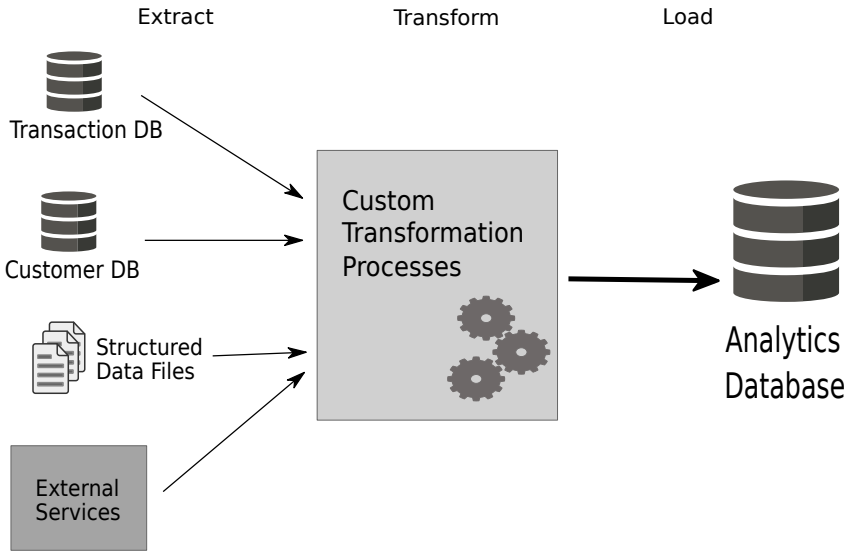

Figure 5.6: ETL process within a company (Wood et al. 2014, pg. 165)

Although this description basically resembles the required activities as explained in the conceptual overview of the E-commerce fraud investigation system before, these ETL processes generally rely on an in-depth knowledge of the data structures that are used in each of the information sources, as well as require a direct access to the databases

and files for retrieving the information. Although these preconditions are not cumbersome to work with *within* an organization, they are not suitable for situations, in which one has to integrate data sources across company boundaries. As the integration of the information takes place on the database level, grant external partners access your internal databases will not only open up access to your business internals, but will also make it much more complicated to change the underlying database structures and business-related software tools. Any changes to one of these would require an elaborate negotiation between the owner of a data source and all of the external partners depending on it.

Beside these drawbacks, which make the ETL approach unsuitable for the E-commerce fraud investigation scenario as a whole, one can assume that these ETL processes are still in use for operating the daily business of each stakeholder involved. They can be helpful in the discussion later (see Section 6.2) when a decision has to be made about how each stakeholder can prepare and transform his internal data sources for external consumptions.

5.4.2 Web Services

With the development of the E-commerce scenario there was also a need to integrate business functionalities from various service providers on the Internet. Valid examples for these kind of integrations are the usage of the PSPs for doing the payment as well as the LSPs for handling the shipping process. These approaches resulted in the "Service Oriented Architecture" paradigms, which enable application services provided by different vendors to talk to each other via a public facing programming interface (aka API). The only requirement for such interoperability to work properly is that each public interface follows some standardized or commonly agreed upon guidelines to be vendor-, platform- as well as programming language-agnostic. One possible implementation of these concepts are the so-called *Web Services*, which use the WS* protocols and standards from the W3C with the extensible markup language (aka XML) and the HTTP protocol at their core (Josuttis 2007).

Like the HTML format, which is used to represent Web pages on the Internet, XML is originally based on SGML, but instead of formalizing markup tags for structuring and styling textual content it is a meta-language allowing everyone to define their own markup languages. In this matter it doesn't dictate what tags are available to structure the information; instead it includes some basic guidelines for creating well-formed and valid documents that uses domain-specific tags, which can be freely defined

and structured by the creator of the XML document. Therefore it is better suited in situations, in which a computer has to parse and evaluate the content of a message; assuming the computer program knows the structure of that message. In an additional step the author of the API could also specify an XML schema for each message, which describes the structure of the message with all the possible elements, their ordering, nesting level and data types in detail. By doing so the XML parser program can later verify the content of a message received against the XML schema, and validate if it is a valid document related to that schema definition. XML schemata are also expressed in XML format and have been standardized by the W3C.

Being able to create custom markup languages via XML has a huge benefit for machine-to-machine communication and is the basis for integrating Web Services (via the WS* protocols), but it still has limitations when it comes to figure out the semantics of those XML messages. This is mostly due to the fact that each XML document represents a new markup language and needs a specific XML parser to be understood by the machine; also to distinguish commonly used tag names in an XML document the creator has to place them into specific namespaces (aka XML namespaces). But these XML namespaces further complicate the automatic processing of XML documents and increase the necessity to have custom instances of XML parsers for each XML document (Taylor & Harrison 2008).

An integration of information exchanged via Web Services is therefore handled separately for each Web Service interface. Looking at the payment service integration as *one* possible example, the following steps are necessary to allow a merchant to interact with the Web Service of a PSP:

- the PSP has to define and implement an interface (aka API) that a merchant can use for exchanging information
- the API includes a set of request/response messages that hold the data being exchanged, usually specified in XML format, as well as a list of operations that the interface supports
- the PSP has to document each of these messages and operations, incl. their intended structures and semantics
- the PSP has to provide access to the API via an HTTP endpoint running on a server at a specific URL
- the PSP usually restricts access to this interface for registered partners only; for doing so they have to provide a registration and identification mechanism

- the merchant has to register with the PSP to be able to call into the Web Service API
- the merchant receives some kind of token that can be used to identify with the Web Service later
- the merchant has to implement an API-specific client-side wrapper that knows how to talk to the interface; incl. calling one of the available operations as well as serializing and deserializing the messages, which will be transmitted between the Web Service and the client program
- the client program from the merchant has to understand the structures and semantics of the messages exchanged with the Web Service and react on them accordingly

Although other merchants, who want to use the same API from the PSP, can use the same client-side wrapper, which is sometimes also provided by that PSP for convenience, to be able to send/receive messages to/from this specific Web Service, they still have to make the API-specific integrations into their own Web shops. Also these integrations are only done in an one-way direction. To allow the merchants to provide information from their own databases, the merchants have to do likewise and provide an API that others can use to query for information by following the same steps as mentioned above.

Additionally, as the structures and semantics of the messages and operations of each Web Service interface are not standardized, integrating with APIs from other PSPs or issuers result in doing the same integration steps again and again. To make things worse, the mapping and linking of the information coming from different APIs have to be implemented by each client to be able to analyze the combined data sets. It becomes clear soon that these necessary tasks will increase the time and efforts with each additional stakeholder, who wants to participate in the collaborative system, see Figure 5.7.

As conclusion one could say that integrating information between a larger group of participants is very limited with the existing Web Service approach. The steps necessary for exchanging information result in huge efforts on all participating parties. As there is no common way to access and combine the information from each of the participants, beside using the fundamental HTTP protocol and XML data format, there have to be a lot of collaborative work between each of them upfront to come up with an approach for integrating the available APIs, and provide the rules for combining the different data structures. Due to these restrictions one can assume that an integration

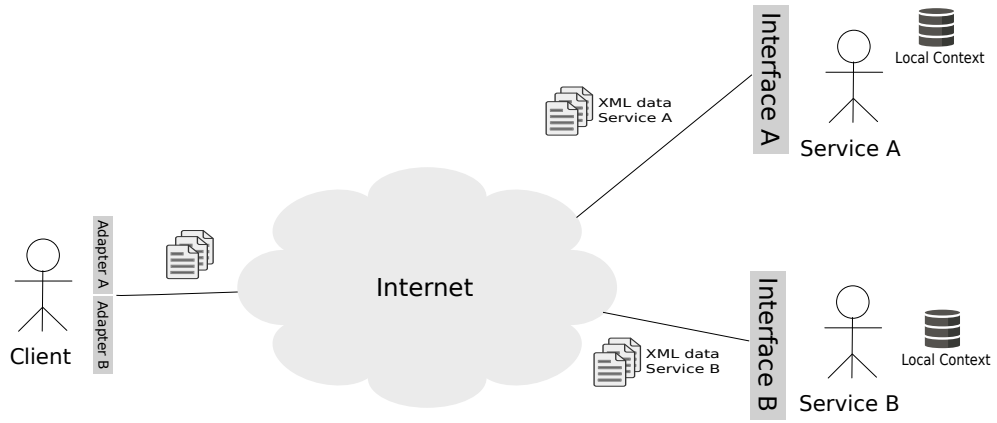

Figure 5.7: Data integration within the Web Service approach

based on the Web Service approach will only work well with a limited number of participants. This might lead to a collaborative system that will only include larger online merchants, PSPs and issuers as participants, and therefore left out smaller companies from the E-commerce fraud investigation process. For a solution of the problem described in Section 3.5 this is not sufficient. Due to this limitations one will need other technologies that provide a better scalability and integration ability for the exchange of information between various, otherwise not strongly related organizations.

5.4.3 Semantic Web

"The Web is full of intelligent applications, with new innovations coming every day" (Allemang & Hendler 2011). But each of these intelligent Web applications are *solely* driven by the data available to them. Information that are likely coming from different places in the global information space, accessible usually via a custom API on a server hosting those resources (see Section 5.4.2). The more consistent the information available to the smart Web application are, the better the service and its result will be. But to support an integration of the data from various Web services the semantics of the information delivered by each of them have to be available, and there has to be a generalized, formalized way to express the semantic of that data. The focus on a standard that enables Web services to express the semantics of the data, also allows for global scalability, openness and decentralization, which are the key principles of the World-Wide Web. The *Semantic Web* tries to give a solution for this problem by providing the Resource Description Framework (aka RDF) and related technologies (e.g. RDF schema, SPARQL, ...) for describing, linking and querying the data that a Web service delivers. But it doesn't reinvent the wheel; instead the Semantic Web builds upon existing, proven technologies such as XML, XML namespaces, XML schemata,

and the URI to uniquely address resources on the Web (Allemang & Hendler 2011).

The main benefits of the Semantic Web approach are the specification of a standardized and generalized format to exchange information on the Web (aka RDF) as well as a commonly agreed way to access and query for them (aka SPARQL). The RDF data format does not only specify the syntax of the information exchanged, but also include the semantics (aka meanings) of them. Due to this fact resources described in RDF format are consistent and semantically self-contained. These characteristics are achieved by providing information as a "triple"; that is a statement consisting of the resource in question (aka subject), a predicate and the specific value (aka object) for it. To be able to unambiguously identify the meaning of these statements, each part of such a "triple" is usually expressed with a unique URI. These URIs can be abbreviated via "prefix" definitions to make the whole statement easier to read (see also Section 4.2). To specify that there is an order "12345" from a "merchant1", one can come up with the following RDF statement, which uses the Schema.org RDF vocabulary (Schema.org b) to describe an order:

```
1 @prefix rdf: <http://www.w3.org/1999/02/22-rdf-syntax-ns#> .
2 @prefix rdfs: <http://www.w3.org/2000/01/rdf-schema#> .
3 @prefix xsd: <http://www.w3.org/2001/XMLSchema#> .
4 @prefix schema: <http://schema.org/> .
5 @base <http://www.merchant1.com/orders/> .
6
7 <12345>   rdfs:type schema:Order;
8          schema:orderNumber "12345"^^xsd:string .
```

Listing 10: An order specification in RDF

An RDF file can contain one or more of such "triples" describing the resources of interest in detail. Usually these "triples" are visualized as directed graph, in that subjects and objects are displayed as nodes and their predicates as edges between them. The order resource shown in the Listing 10 above can also be visualized as graph as shown in Figure 5.8.

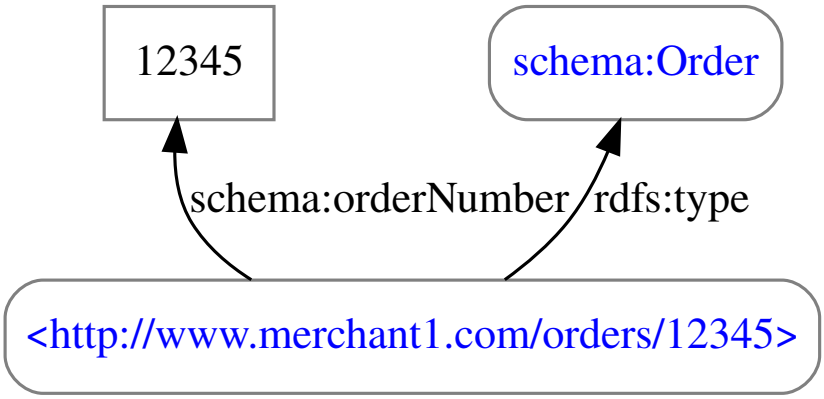

Figure 5.8: Graph-based visualization of the order from Listing 10

Additionally, the RDF format has build-in support for merging information from different data sources. This functionality is only working as expected if the "triples" in the dispersed data stores are using the same URIs to refer to the same subjects or objects. In that situation merging the "triples" from different RDF data sets will result in a locally linked data set holding the combined information as shown in Figure 5.9.

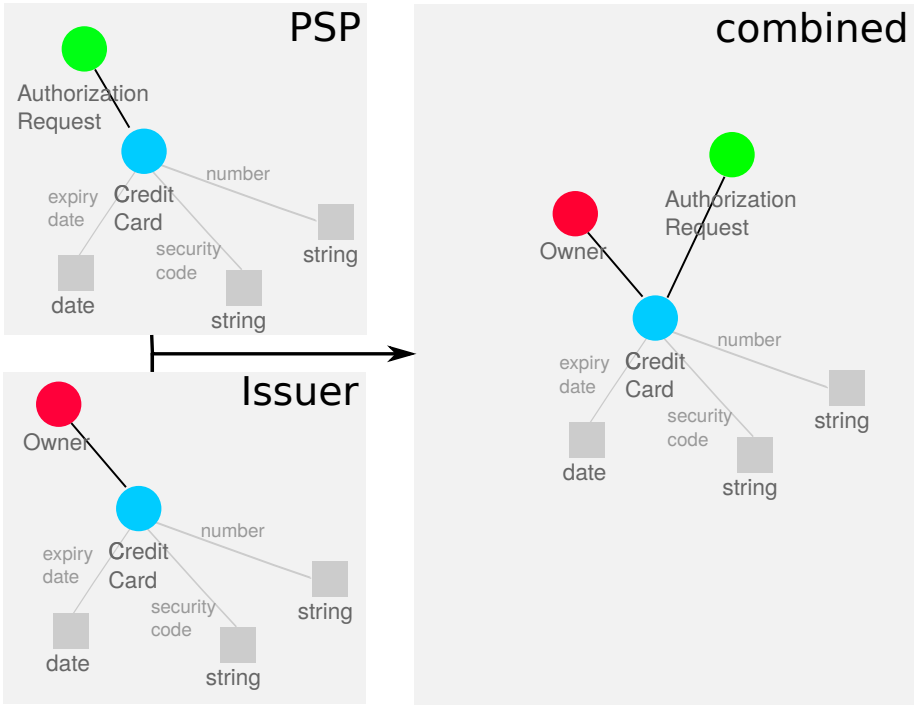

Figure 5.9: Combining two RDF files containing the same credit card entity

Beside being able to provide internal resources in an understandable RDF format for external consumption, the Semantic Web also specifies how to query and access these "information databases" on the Web. For that purpose the SPARQL protocol and query language has been defined. It does not only describes a language to query for information located in RDF data stores, but also specifies how to setup an HTTP endpoint on a server to make the RDF data set publicly available on the Internet.

Following the specifications of the Semantic Web standards each relevant participant of an E-commerce fraud investigation system will have to transform the information from their internal databases into a set of "triples" with commonly agreed upon URI references and persist them into a RDF data store. For this transformation process an extension of the existing ETL processes in an organization can be used. Additionally, these RDF data sets will be made available publicly on the Web for information retrieval via the SPARQL protocol and query language. Each participant of the collaborative system will only need to know the specific addresses of these HTTP endpoints to be able to query them for information. The results of each query can be easily combined into the local RDF data set based on the merging capabilities of the RDF standard. This will decrease the efforts for integrating the data from various external sources drastically. Also communicating with the different HTTP endpoints to access and query for information is being done in a much more efficient way based on the SPARQL protocol and query language, see Figure 5.10.

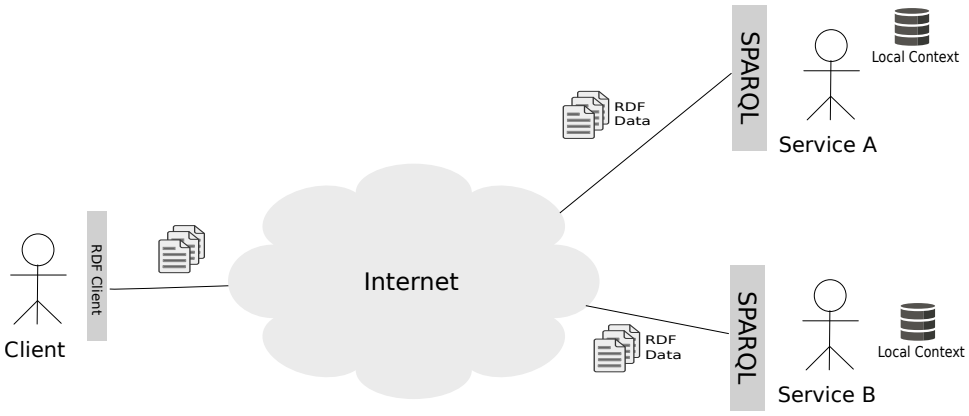

Figure 5.10: Data integration within the Semantic Web approach

As the underlying model of a RDF data set is resembling a graph-based data model it will fit the concept of the proposed system from Section 5.3 perfectly. Still requiring every participant to setup and operate a public available SPARQL server will limit the use of this approach for the solution of the E-commerce fraud investigation scenario. As

parts of the information that have to be exchanged between the relevant participants are confidential and/or business-critical, requiring a public SPARQL endpoint on the Internet is a high security risk. Additionally the SPARQL protocol and query language does not offer a way to restrict access to only a subset of the information in the RDF data stores. Any party, who is aware of the URL of a SPARQL endpoint, have access to all the information that are in the underlying RDF data stores and can easily retrieve them with a single SPARQL query (see Listing 11). It is therefore no surprise that there are only a small set of publicly available SPARQL endpoints on the Internet — with the most commonly used one from DBpedia.org (DBpedia), which offers publicly available information from Wikipedia articles in RDF format.

```
1 PREFIX rdf: <http://www.w3.org/1999/02/22-rdf-syntax-ns#>
2 PREFIX rdfs: <http://www.w3.org/2000/01/rdf-schema#>
3
4 SELECT ?s ?p ?o    # select every subject-object-predicate triple found
5 WHERE {
6  ?s ?p ?o          # do not specifying a condition returns everything
7  }
```

Listing 11: Retrieving all information in an RDF store using SPARQL

To conclude, one can assert that the fundamental technologies of the Semantic Web standards are a good fit for exchanging and merging information between different stakeholders. But the usage of an all or nothing approach for querying the RDF data stores via the SPARQL protocol and query language is way to open for the E-commerce fraud investigation system.

5.5 Conclusion

To support the investigation of E-commerce frauds as described in Section 3.5 the collaborative system has to collect and combine transactional information from various online merchants, in which Web shops a credit card in question has been used recently. The system has to support the combination and linking of the transactional details by utilizing a graph-based data model. Doing so will allow the system to classify and cluster the transaction information based on various criteria, which can help the investigator to figure out abnormal behavioral patterns in the credit card usage on the Internet. Visualizing the combined data set can make use of the graph-based data model and present the transaction details as a clustered graph on screen. Addition-

ally, the representation of the information can change based on the requirements of the investigators.

As the previous section showed in detail, existing approaches are of limited use for the collection and combination of dispersed transactional details in this scenario. The leading approach for the E-commerce fraud investigation system will have to combine the best characteristics from the Web Service and the Semantic Web designs.

As for the Web Service approach, the most valuable aspects of it are:

- access to the HTTP endpoints can be limited to a certain set of communication partners
- these partners have to authenticate with each Web Service first
- based on the identification of the partners only certain aspects of the information can be exchanged, and execution of Web Service operations can be restricted

Looking at the Semantic Web approach, it's most interesting functionalities are:

- providing information in a semantically self-contained way
- the ability to merge and link together information from different RDF data stores locally
- the graph-based data model underlying the RDF data stores
- the usage of SPARQL to query and analyze the locally combined data sets

In the following Chapter 6 the Master thesis will come up with an approach that uses the fundamental technologies from the Semantic Web for information sharing and integration as well as peer-to-peer communication technologies for securing and restricting access to the RDF data sets from relevant participants of the E-commerce fraud investigation scenario.

6 Design of a collaborative system

This chapter is about the design of a collaborative system that supports the investigation of E-commerce fraud incidents. It starts with a discussion of the semantics of the underlying RDF data sets, and how these can be combined across various organizations. After that it shows how these information can be provided to the relevant parties based on the E-commerce fraud investigation scenario described in Chapter 3. For this purpose it looks in detail into the partially centralized P2P communication architecture and shows how that can be used for securely sharing the relevant information between the stakeholders.

6.1 RDF vocabularies and Web Ontologies for E-commerce

As a major objective of the E-commerce fraud investigation system is to collect the various transactional information from online merchants, PSPs and issuers, combine and link them together, and analyze the resulting data set from different view points to find abnormal activities, the information exchanged between the relevant participants either have to follow commonly available RDF vocabularies, have to be based on a custom shared RDF vocabulary that has been specifically designed for this system, or have to be mapped and linked against each other from different RDF schema specifications.

6.1.1 Using a common RDF vocabulary

One valid approach to come up with a data schema for the collaborative system is to take a look into commonly used RDF vocabularies and Web ontologies, and try to figure out whether they can be used for describing the information that need to be exchanged between participants of the E-commerce fraud investigation system. When consulting the Semantic Web community for commonly agreed upon and highly used RDF schema specifications, one will come up with the following list (see Table 6.1):

Name	Prefix	Describes	Namespace URI
Dublin Core	dc:	Meta data	http://purl.org/dc/terms/
FOAF	foaf:	People	http://xmlns.com/foaf/0.1/
Geo	pos:	Positions	http://www.w3.org/2003/01/geo/wgs84_pos#
Geo Names	gn:	Locations	http://www.geonames.org/ontology#
Good Relations	gr:	Products	http://purl.org/goodrelations/v1#
RDF	rdf:	Core framework	http://www.w3.org/1999/02/22-rdf-syntax-ns#
RDFS	rdfs:	RDF vocabularies	http://www.w3.org/2000/01/rdf-schema#
Schema.org	schema:	Schema.org vocabularies	http://schema.org/
SKOS	skos:	Controlled vocabularies	http://www.w3.org/2004/02/skos/core#
vCard	vcard:	Business Cards	http://www.w3.org/2006/vcard/ns#
Web Ontology Language	owl:	Ontologies	http://www.w3.org/2002/07/owl#
XML Schema Datatypes	xsd:	Data types	http://www.w3.org/2001/XMLSchema#

Table 6.1: Commonly used RDF vocabularies on the Web (Wood et al. 2014, pg. 41)

Based on these existing schema specifications describing a fictive consumer named "Max Mustermann" incl. his home address can be done by combining data utilizing the FOAF and vCard vocabularies into a RDF data set such as described in Listing 12 and visualized as directed graph in Figure 6.1. The described resource can be uniquely identified by the URI `http://www.merchant1.com/customers/MaxMustermann`. Additionally one can see, that these vocabularies use expressive names for their entities and predicates, which make it easier to understand their intended meanings (e.g. "foaf:givenname", "vcard:locality", ...).

```
@prefix rdf: <http://www.w3.org/1999/02/22-rdf-syntax-ns#> .
@prefix rdfs: <http://www.w3.org/2000/01/rdf-schema#> .
@prefix xsd: <http://www.w3.org/2001/XMLSchema#> .
@prefix foaf: <http://xmlns.com/foaf/0.1/> .
@prefix vcard: <http://www.w3.org/2006/vcard/ns#> .
@base <http://www.merchant1.com/customers/> .

<MaxMustermann> rdf:type foaf:Person;
                rdfs:label "Max Mustermann";
                foaf:family_name "Mustermann";
                foaf:givenname "Max";
                foaf:gender "Male";
                foaf:title "Mr.";
                vcard:adr [
                 rdf:type vcard:Home;
                 vcard:street-address "Mustermannstr. 12";
                 vcard:locality "Musterstadt";
                 vcard:region "North-Rhine Westfalia";
                 vcard:postal-code "33123";
                 vcard:country-name "Germany"
                ] .
```

Listing 12: Personal related information about a fictive consumer in RDF

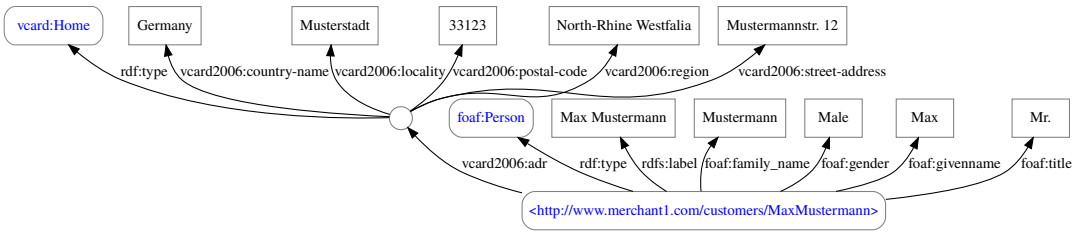

Figure 6.1: Graph representation of consumer information from Listing 12

Still, being able to describe persons and their addresses is just a subset of the entities and relations found in the E-commerce scenario. When looking back to the initial ER model of an E-commerce transaction as shown in Section 5.2 one can map the information, which are currently available in the E-commerce scenario, to the existing RDF vocabularies such as follows (see Table 6.2):

Information	**RDF vocabulary**
Consumer	FOAF
Credit Card Owner	FOAF
Billing Address	vCard
Shipping Address	vCard
Location Information	Geo Names
Merchant	GoodRelations
Items	GoodRelations
Item Categories	GoodRelations
Brands	GoodRelations
Payment Types	GoodRelations

Table 6.2: Possible usage of RDF vocabularies for E-commerce transaction information

As this table shows there are some parts of the E-commerce ER model that can be expressed with existing RDF vocabularies extensively — such as personal related information via FOAF and vCard, whereas other parts can not be stated in-depth (e.g. credit card information), or are not specified at all (e.g. tracking of the delivery). Due to these circumstances one usually have to build an own RDF vocabulary or Web ontology that fills in the missing pieces and refers to the existing concepts whenever appropriate.

When trying to model the information of a credit card as displayed in Figure 5.2 a possible result will be the RDFS specification shown in Listing 13. This definition of a credit card resource explicitly reuses specifications from the FOAF and GoodRelations ontologies by defining that:

- the owner of a credit card has to be of type “Person” from the FOAF ontology
- the type of a credit card has to be an instance of the type “PaymentMethodCreditCard” from the GoodRelations ontology

As most of the parts of the E-commerce data model shown in Figure 5.2 can not be expressed directly with the existing RDF vocabularies, filling in the gaps would mean to come up with a large set of custom entities and relationships, which will limit the usage of the system as explained in Section 6.1.2.

```
@prefix rdf: <http://www.w3.org/1999/02/22-rdf-syntax-ns#> .
@prefix rdfs: <http://www.w3.org/2000/01/rdf-schema#> .
@prefix xsd: <http://www.w3.org/2001/XMLSchema#> .
@prefix foaf: <http://xmlns.com/foaf/0.1/> .
@prefix gr: <http://purl.org/goodrelations/v1#> .
@base <http://www.example.com/ecommerce#> .
# define the subject "CreditCard"
<CreditCard>      rdf:type rdfs:Class;
                  rdfs:comment "Describes a credit card in the
  ↪  E-commerce scenario";
                  rdfs:label "A credit card" .
# define the property "ExpirationDate" on subject "CreditCard"
<ExpirationDate>  rdf:type rdfs:Property;
                  rdfs:domain <CreditCard>;
                  rdfs:range xsd:date;
                  rdfs:label "Expiration Date" .
# define the property "SecureCode" on subject "CreditCard"
<SecureCode>      rdf:type rdfs:Property;
                  rdfs:domain <CreditCard>;
                  rdfs:range xsd:string;
                  rdfs:label "Security Code" .
# define the property "Number" on subject "CreditCard"
<Number>          rdf:type rdfs:Property;
                  rdfs:domain <CreditCard>;
                  rdfs:range  xsd:string;
                  rdfs:label "Credit Card Number" .
# define the property "BelongsTo" on subject "CreditCard"
<BelongsTo>       rdf:type rdfs:Property;
                  rdfs:domain <CreditCard>;
                  rdfs:range <foaf:Person>;
                  rdfs:label "Credit Card Owner" .
# define the property "Type" on subject "CreditCard"
<Type>            rdf:type rdfs:Property;
                  rdfs:domain <CreditCard>;
                  rdfs:range <gr:PaymentMethodCreditCard>;
                  rdfs:label "Type of Credit Card" .
```

Listing 13: A specification for a credit card in RDFS

When analyzing the list of existing and actively used RDF vocabularies and Web ontologies in Table 6.1, one will also find the Schema.org initiative (Schema.org b). This meta data vocabulary was initially designed by the leading search engines (e.g. Google, Microsoft and Yahoo!) to allow authors of Web sites to markup their HTML documents in a way so that they are better understood by these search engines. The Schema.org vocabulary is actively maintained by its community, includes new concepts with each release, and also offers an extension mechanism to implement additional vocabularies with terms that are not part of the core specifications (Schema.org a) yet. In one of the past releases the maintainers also introduced all of the existing concepts of the GoodRelation ontology into the Schema.org vocabulary (R.V. Guha).

As online merchants will likely provide semantic meta data for their products and offerings in the vocabulary of Schema.org already to improve their listings on search engine results (also known as SEO), one can reuse parts of these information for the E-commerce fraud investigation scenario. Additionally, the wide-ranging scope of aspects declared in the Schema.org vocabulary can make it a good fit for the collaborative system of the E-commerce fraud investigation scenario. When trying to map the initial ER model from Section 5.2 to the Schema.org core specifications, one will basically come up with a schema as displayed in Figure 6.2.

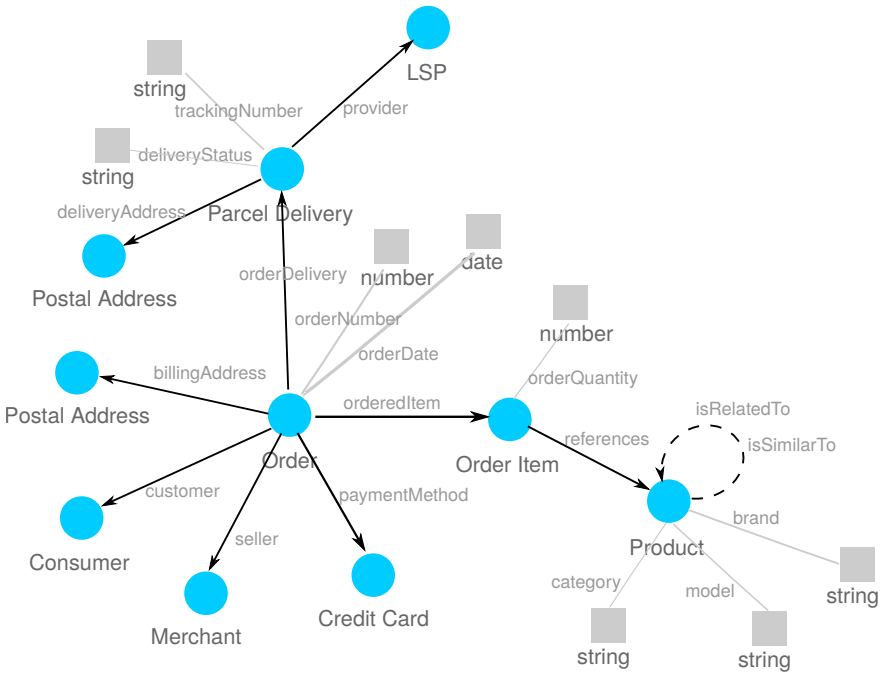

Figure 6.2: Schema.org based mapping of an E-commerce transaction

6.1.2 Creating a custom RDF vocabulary

Another possible approach to harmonize information in the collaborative system is to define a completely new RDF vocabulary or Web ontology for the proposed E-commerce fraud investigation system and share that with every possible stakeholder. This specification will have to define all the entities and relations known to the collaborative system and describe them in RDFS format (see Section 4.2).

A major drawback of this approach is that new participants of the system will have to implement the conversion of their internal data structures to a RDF data set that follows the predefined schema definition first, before even being able to join in. This will limit the general usage of the collaborative system, and will therefore not further considered in detail.

6.1.3 Mapping and Linking between RDF vocabularies

Although it is possible to model an E-commerce transaction solely with the Schema.org specifications as shown in Figure 6.2, the collaborative system likely has to take care of the mapping of the transactional information coming from various sources to be able to combine them later. As the Semantic Web does not restrict how organizations structure and express their information, and due to the "AAA slogan" (see Section 4.2), there are likely different RDF representations of an E-commerce transaction in-use and have to be brought together.

The W3C standards for the Semantic Web also include support for these mapping issues, because they will also come up when trying to combine semantic information available around the Web. The following axioms are available in the RDFS and OWL specifications explicitly for that purpose:

- **rdfs:subClassOf:** a relation of type "rdfs:subClassOf" defines a specialization of a class, in which the child class inherits all the properties of the parent class,
- **rdfs:subProperyOf:** a relation of type "rdfs:subPropertyOf" defines a specialization of a property, in which the child property inherits all constraints of the parent property,
- **owl:equivalentClass:** a relation of type "owl:equivalentClass" specifies the equality of classes coming from different RDF vocabularies or Web ontologies,
- **owl:equivalentProperty:** a relation of type "owl:equivalentProperty" specifies the equality of classes coming from different RDF vocabularies or Web ontologies

If a merchant wants to state that a product-related information, which is delivered as resource using the GoodRelations vocabulary, is equal to product information that can be found in the Schema.org specification, he or she can do so as follows (see Listing 14):

```
@prefix rdf: <http://www.w3.org/1999/02/22-rdf-syntax-ns#> .
@prefix rdfs: <http://www.w3.org/2000/01/rdf-schema#> .
@prefix owl: <http://www.w3.org/2002/07/owl#> .
@prefix schema: <http://schema.org/> .
@prefix gr: <http://purl.org/goodrelations/v1#> .

# mapping classes and properties between GoodRelations and Schema.org
gr:ProductOrService owl:equivalentClass schema:Product .
gr:category owl:equivalentProperty schema:category .
gr:color owl:equivalentProperty schema:color .
gr:description owl:equivalentProperty schema:description .
gr:hasBrand owl:equivalentProperty schema:brand .
gr:hasEAN_UCC-13 owl:equivalentProperty schema:gtin13 .
gr:hasGTIN-14 owl:equivalentProperty schema:gtin14 .
gr:hasGTIN-8 owl:equivalentProperty schema:gtin8 .
gr:name owl:equivalentProperty schema:name .
[...]
```

Listing 14: Mapping product-related information from GoodRelations to Schema.org

These mapping statements from one RDF vocabulary to another can be either created and injected into a RDF data store by the party who is going to merge information from different sources according to needs, or can also be part of the resource specification coming from an external source. In the former case the stakeholder who is collecting and combining the information from various sources has to maintain the additional "triples" to map information between each RDF vocabulary used and include them in the processing of the statements within the combined RDF data store. With an increased number of participants, who are using disjunct RDF vocabularies, the effort and time to manage and create these mapping instructions on the collectors side will increase tremendously. Therefore the second approach is the preferred one. In that situation the RDF description of an entitiy coming from an external source is already stating the mapping to one or more well-known RDF vocabularies (e.g. the Schema.org specification mentioned above). This will reduce the effort and time to combine the

information from different sources, and will only slightly increase the effort on the side of the external partner to prepare their resources for external consumptions.

6.2 Combining RDF data sets in the E-commerce scenario

With these methods in place one can now specify how the relevant participants have to provide their information so that they can be combined and analyzed in the collaborative system. This section explains how the different participants might prepare their local context information for external consumption, and how the transactional details from various online merchants can be combined on an individual resource level to be able to analyze and cluster the transactions by different criteria as described in Section 5.3.

6.2.1 Preparing internal information for external consumption

As explained in Section 5.4.1 there are likely ETL processes in-use within the IT operations of every stakeholder. These processes are usually collecting and combining information from internal data sources for *internal* business analytics, but can also be used to prepare internal data for external consumption. In the latter case the parts of the relevant information for the E-commerce fraud investigation have to be extracted from the internal databases and encoded in a RDF data set incl. the RDFS vocabulary used and required mapping statements to a well-known vocabulary such as the one from Schema.org as shown in Figure 6.2.

Merchant

The merchants should be able to provide RDF encoded information for their orders based on a given payment token or based on a consumer identification. The former selector is likely used in the initial phase of collecting all required information of orders that have been done with a credit card recently. The latter one is of interest if there are malicious transactions found for a consumer and a merchant will have to provide additional order details coming from that individual.

The merchants provide the following information in RDF:

- **product-related information:** as part of the order details the merchants have detailed information about the products that have been bought by a consumer. These information include the brand, model as well as product categories of each item within an order. These are likely of interest in the E-commerce fraud

investigation. If merchants have these information in a RDFa format on their Web sites already, they can refer to those data via the "rdfs:seeAlso" predicate, which holds a URI to an external resource that contains additional information for the subject (see Listing 15 for an example). Additionally, the product-related information might be available in different languages on the Web shop. The merchants should use the English expression for each textual identifier in a RDF data set and express language-dependent terms via the "rdfs:label" predicate that is used for a human-friendly name of the resource and supports language specifiers (see Listing 16 for an example).

- **consumer-related information:** as part of the order details the merchants also have the personal related information of the buyer incl. billing and shipping addresses.
- **merchant-related information:** the merchants can also provide information about themselves such as the retail branch they are operating in.

```
1  @prefix rdf: <http://www.w3.org/1999/02/22-rdf-syntax-ns#> .
2  @prefix rdfs: <http://www.w3.org/2000/01/rdf-schema#> .
3  @prefix xsd: <http://www.w3.org/2001/XMLSchema#> .
4  @prefix schema: <http://schema.org/> .
5  @base <http://www.merchant1.com/orders/> .
6
7  <O12345>  rdfs:type schema:Order;
8            schema:orderedItem [
9              rdfs:type schema:Product;
10             schema:name "Self-cleaning refrigerator";
11             rdfs:seeAlso <http://www.merchant1.com/catalog/P12345> .
12           ] .
```

Listing 15: Specifying a link to a Web site for looking up product-related information in RDF[1]

[1]Please note that the application has to resolve the URI and embed the external RDF data set at this position.

```
@prefix rdf: <http://www.w3.org/1999/02/22-rdf-syntax-ns#> .
@prefix rdfs: <http://www.w3.org/2000/01/rdf-schema#> .
@prefix xsd: <http://www.w3.org/2001/XMLSchema#> .
@prefix schema: <http://schema.org/> .
@base <http://www.merchant1.com/catalog/> .

<P12345>  rdfs:type schema:Product;
          schema:name "Self-cleaning refrigerator";
          rdfs:label "Selbstreinigender Kühlschrank"@de;
          rdfs:label "Self-cleaning refrigerator"@en;
          rdfs:label "refrigerador autolimpiable"@es .
```

Listing 16: Specifying a product with labels in three different languages in RDF[2]

Payment Service Provider

The PSPs provide information about a payment token and the authorization request that belongs to it. These information are required to link the credit card to the order information from a merchant. Due to this the PSPs can act as a broker between the issuers and online merchants. On the one hand they have a strong relationship with the issuers for any payment related activities, and on the other hand they have an integration of their Web service APIs at the merchants (see Section 5.4.2). The benefit of this is that the issuers do not have to know about any online merchant operating on the Internet, because they can get contact information to any of these merchants from the PSPs. In a distributed P2P communication scenario the PSPs can also use the RDFS predicate “rdfs:seeAlso” to provide links to the affiliated online merchant of a payment authorization in their RDF data set as shown in Listing 17.

[2]Please note that the name of the product has been stated without a language specifier, which makes it valid globally.

```
@prefix rdf: <http://www.w3.org/1999/02/22-rdf-syntax-ns#> .
@prefix rdfs: <http://www.w3.org/2000/01/rdf-schema#> .
@prefix xsd: <http://www.w3.org/2001/XMLSchema#> .
@prefix schema: <http://schema.org/> .
@base <http://www.payservices.com/payments/> .

<C12345>  rdfs:type schema:PayAction;
          schema:recipient [
            schema:name "ACME Corp.";
            rdfs:seeAlso "http://www.acme.com/about"
          ] .
```

Listing 17: Linking to an online merchant in the RDF from a PSP

Logistic Service Provider

The LSPs provide information about a tracking number and the delivery status of an order. If the recipients have to show their ID card or have to place a signature on the delivery receipt, the LSPs can also hand over personal related information about them. The information will be requested based on the tracking number that is shared with a merchant. In these situations the merchants will be acting as a broker between the issuers and the LSPs due to the strong business integrations between merchants and LSPs.

Issuer

The issuers holds information about credit cards and their owners incl. personal related information. They are the ones who are usually initiating the E-commerce fraud investigation by asking the PSPs for detailed order information to a payment authorization request. They will also collect all the information from the different stakeholders and have to combine and analyze them to be able to validate a credit card transaction. To be able to do so they will have to use a RDF data store, in which the dispersed RDF data sets are imported and linked against each other (see next section).

6.2.2 Merging transactional information from various sources

Still, if the transactional details from various online merchants have to be linked together on the individual entity level to support analyzing and clustering the information on different aspects, the build-in merging capabilities of the RDF specification will rely

on unique URIs used for the same entities found in different RDF data sets as shown in the Section 5.4.3. These unique URIs are used to identify the resources in a dispersed RDF data set.

- **owl:sameAs, schema:sameAs**: the "owl:sameAs" as well as the "schema:sameAs" relations are providing an unique URI that unambiguously define the subject (see Listing 18 for an example)

```
@prefix rdf: <http://www.w3.org/1999/02/22-rdf-syntax-ns#> .
@prefix rdfs: <http://www.w3.org/2000/01/rdf-schema#> .
@prefix xsd: <http://www.w3.org/2001/XMLSchema#> .
@prefix schema: <http://schema.org/> .
@base <http://www.merchant1.com/customers/> .

<MaxMustermann> rdf:type schema:Person;
                rdfs:label "Max Mustermann"@en;
                schema:adress [
                  rdf:type schema:PostalAddress;
                  schema:addressLocality "Cologne"@en;
                  schema:sameAs <http://dbpedia.org/resource/Cologne>
                ] .
```

Listing 18: Specifying a link to a DBpedia resource to uniquely identify an entity in RDF

When looking at the E-commerce transaction schema as defined in Section 5.2 the following information must be uniquely identified and mapped within the RDF data sets coming from the participants of the collaborative system:

- **personal related information** such as the consumer, recipient and credit card owner
- **location based information** such as the billing and shipping address as well as the location a credit card owner is registered for
- **product related information** such as the categories, subcategories, brand, model and item description
- **merchant related information** such as the branch of a merchant

To support the unique identification of entities in the RDF data set of an E-commerce transaction, one can refer to publicly available RDF data sets on the Internet, such as GeoNames or DBpedia. These can provide an unique URI for locations and named places. A product-related RDF data set was available in form of the ProductDB initiative until recently (Bouzidi et al. 2014). Due to the shutdown of it, the mapping of products can no longer be done by referencing unique URIs from the Web, but will have to be based on the global trade item number (aka GTIN) of each product. Additional aspects of an item, such as brand, categories and subcategories, can be found on DBpedia as well.

A problem, that will come up, is the unique addressing of personal related information, such as identifying the consumer. The collaborative system can not rely on mapping the personal related information based on properties such as familyName and givenName alone (see Listing 12). There could be typos in the information coming from various RDF data sets, and different individuals can still have the same name information. One possible approach to bring these information together would be the mapping based on the e-mail address of the individual. An e-mail address like an URI is a globally unique addressing scheme, and one can assume that two entities, who are using to the same e-mail address, are referring to the same entity. Still this is only a weak hint as an individual can have more than one e-mail address, and could use different e-mail addresses for the online shopping trips at different merchants. Therefore a more sophisticated mapping algorithm for personal related information is needed in the collaborative system. This algorithm may take into account the combination of familyName, givenName, dateOfBirth as well as location-based information to uniquely identify an individual. To sum up, the identification of important entities from an E-commerce transaction can be based on the following aspects (see Table 6.3):

Entity	Unique Identifier	Public Data Set	Example
Person	eventually e-Mail address	n/a	`mailto:max.mustermann@t-online.de`
PostalAddress	Location, Position	Geo Names	`http://sws.geonames.org/2886242/`
Item	GTIN, ISBN	n/a	`gtin:9781617290398`
Brand	Name	DBpedia	`http://dbpedia.org/resource/Samsung`
Organization	Web Site URL	n/a	`http://www.samsung.com`

Table 6.3: List of possible criteria to uniquely identify entities of an E-commerce transaction

6.3 Using a partially centralized P2P system

For the E-commerce fraud scenario, that has been selected for this thesis in Section 3.5, one can say that the issuer of a credit card is the party who initiates the collaborative fraud investigation. They are recognizing the active use (and likely misuse) of a credit card in the online and the offline world first, and are also getting a notification about any suspicious transactions made with it from their fraud prevention systems. Due to this fact, one can come up with a partially centralized P2P architecture for the E-commerce fraud investigation system, in that the issuer of a card in at the center and acts as a trusted party in this system.

6.3.1 Analyzing information at the issuer

This issuer will initiate a collaborative session with the other required stakeholders based on the usage history of the credit card in question. During this P2P communication session the merchants, PSPs and LSPs will share the required information with the issuer. In this process the data from the other stakeholders will be replicated to the issuer, who will build up a networked graph based on the Schema.org specification. So the main work will be on the issuer's side, who is the major driving party in the system, as depicted in Figure 6.3.

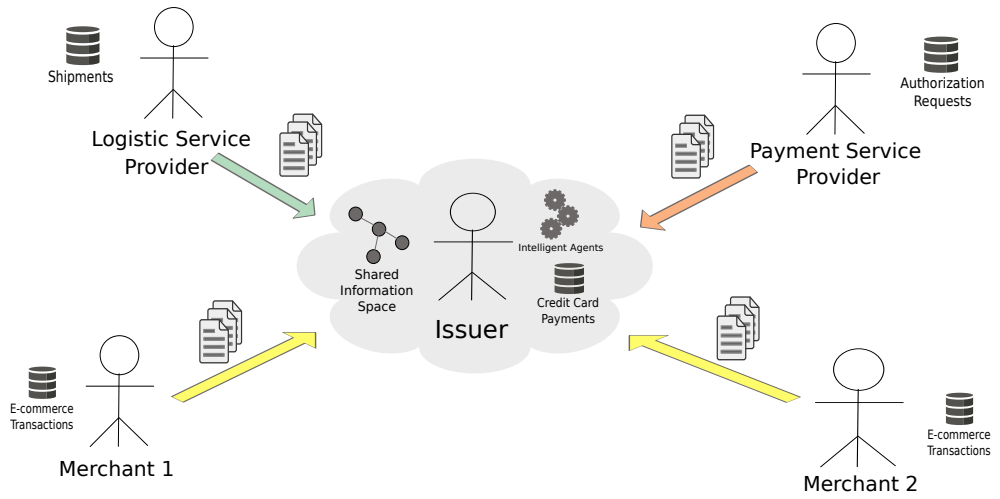

Figure 6.3: Collaborative system using a partially centralized P2P architecture

Using the WebRTC communication protocol for initiating the P2P session will allow the issuers to setup a communication between the relevant stakeholders directly from within an application running in their Web browsers. The application can visualize the connectivity status of the participants, the progress of their data sharing efforts as well as offer direct face-to-face communication possibilities in case of misunderstandings or further requests.

One of the major issues with the above mentioned system architecture is, that the merchants, PSPs and LSPs have to hand over all of their relevant information to the issuer of a credit card for the analysis.

6.3.2 Dealing with privacy and security concerns

...

7 Conclusion and Future Work

7.1 Towards a decentralized P2P system

In the decentralized P2P system architecture each node is equal and keeps their local data ready for analysis if the node is online. If the issuer will have to figure out, whether a transaction is fraudulent or not, she is going to send out various queries to all the available nodes in the P2P cluster asking for certain information that help investigating the case. The other nodes, whose reside on each stakeholder involved, will answering the queries based on the common Schema.org data mapping shown above and send back the results to the issuer bank. The issuer will collect all the results from the various parties and combine them to be able to analyze the issue and come up with a conclusion. The main benefit of this architecture is, that there is no need to duplicate the data from the other stakeholders to the issuer. Due to this it can also be a better suited solution if data sharing faces restrictions due to law or regulations. On the other hand this architecture will depend on the nodes being online all the time so the issuer can query for information at any time. So this works only in synchronous communication mode. Additionally there are efforts spread around all the stakeholders to set up and maintain a system for secure data querying functionality, please see Figure 7.1.

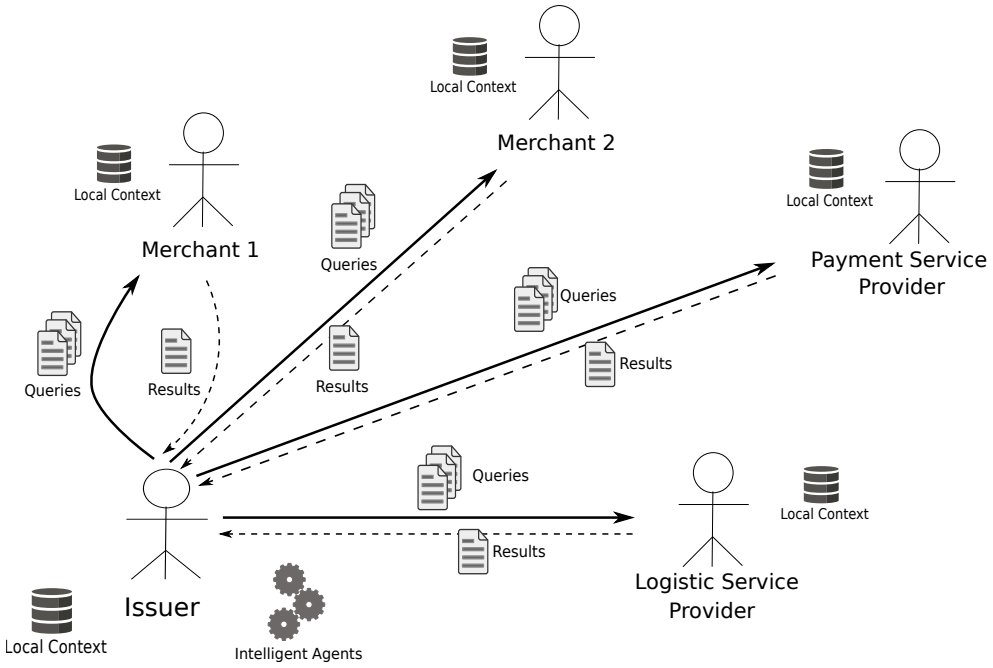

Figure 7.1: Decentralized P2P system architecture

List of Figures

List of Tables

List of Listings

Glossary

API	Application Programming Interface.
B2B	Business-To-Business.
B2C	Business-To-Consumer.
C2B	Consumer-To-Business.
C2C	Consumer-To-Consumer.
CSCW	computer-supported cooperative work.
CSP	Cloud Service Provider / Hosting Service.
E-commerce	Electronic trading over a network such as the Internet.
EMV	Europay, MasterCard and Visa defined security standard for credit and debit cards.
ER	Entity-relationship.
ETL	Extract-Transform-Load.
FOAF	Friend-of-a-Friend: commonly used RDF vocabulary to describe people.
GTIN	Global Trade Item Number.
HTML	Hypertext Markup Language.
HTTP	Hypertext Transfer Protocol.
ICE	Interactive Connectivity Establishment.
IP	Internet Protocol.
ISBN	International Standard Book Number.
ISP	Internet Service Provider.
ISV	Independent Software Vendor.
IT	Information Technology.
JSON	JavaScript Object Notation.
JSON-LD	JavaScript Object Notation for Linked Data.
LSP	Logistic Service Provider.
M-commerce	Electronic trading via mobile computers such as smartphones and tablets.
NAT	Network Address Translation.

OAuth	An open protocol to allow secure authorization on the Web.
OWL	Web Ontology Language.
P2P	Peer-To-Peer.
PCI/DSS	Payment Card Industry Data Security Standards.
PSP	Payment Service Provider.
RDF	Resource Description Framework.
RDFa	Resource Description Framework in Attributes.
RDFS	Resource Description Framework Schema.
SCTP	Stream Control Transmission Protocol.
SEO	Search Engine Optimization.
SGML	Standard Generalized Markup Language.
SPARQL	SPARQL Protocol and RDF Query Language.
SQL	Structured Query Language.
TLS	Transport Level Security.
URI	Uniform Resource Identifier.
URL	Uniform Resource Locator.
vCard	vCard: commonly used RDF vocabulary to describe contact information.
W3C	World-Wide Web Consortium.
WebRTC	Web Real-Time Communication.
XML	Extensible Markup Language.

Bibliography

Allemang & Hendler 2011
Allemang, Dean; Hendler, James: *Semantic web for the working ontologist: effective modeling in RDFS and OWL.* Elsevier, 2011

Amazon.com
Amazon.com: *Amazon.com: Online Shopping for Electronics, Apparel, Computers, Books, DVDs & more.* `https://www.amazon.com/`

Ankhule & Joshy 2015
Ankhule, Gayatri R.; Joshy, MR: Overview of E-Commerce. In: *International Journal of Electronics, Communication and Soft Computing Science & Engineering (IJECSCSE)* (2015), pages 196

Antoniou & Van Harmelen 2012
Antoniou, Grigoris; Van Harmelen, Frank: *A semantic web primer.* 3rd. Edition. MIT Press, 2012

Bannon & Bødker 1997
Bannon, Liam; Bødker, Susanne: *Constructing common information spaces.* In: *Proceedings of the Fifth European Conference on Computer Supported Cooperative Work* Springer, 1997, pages 81–96

Barker & Campbell 2014
Barker, Phil; Campbell, Lorna M.: What is schema.org? In: *LRMI. Retrieved April* 21 (2014), pages 2015

Bizer et al. 2009
Bizer, Christian; Heath, Tom; Berners-Lee, Tim: Linked data-the story so far. In: *Semantic Services, Interoperability and Web Applications: Emerging Concepts* (2009), pages 205–227

Borghoff & Schlichter 2000
Borghoff, UM; Schlichter, JH: *Computer-Supported Cooperative Work: Introduction to Distributed Applications. Secaucus.* NJ, USA: Springer-Verlag New York, Inc, 2000

Bouzidi et al. 2014
Bouzidi, Sabri; Vandic, Damir; Frasincar, Flavius; Kaymak, Uzay: *Product Information Retrieval on the Web: An Empirical Study.* In: *The 8th International Conference on Knowledge Management in Organizations* Springer, 2014, pages 439–450

Brachmann 2015
Brachmann, Steve: *In the face of growing e-commerce fraud, many merchants not prepared for holidays - IPWatchdog.com |patents & patent law.* http://www.ipwatchdog.com/2015/11/22/growing-e-commerce-fraud-merchants-not-prepared-for-holidays/id=63271/. Version: 11 2015

Buford et al. 2009
Buford, John; Yu, Heather; Lua, Eng K.: *P2P networking and applications.* Morgan Kaufmann, 2009

Business Wire 2015
Business Wire: Global card fraud losses reach $16.31 Billion — will exceed $35 Billion in 2020 according to the Nilson report. In: *Business Wire* (2015), 08. http://www.marketwatch.com/story/global-card-fraud-losses-reach-1631-billion-will-exceed-35-billion-in-2020-accor

Cai & Frank 2004
Cai, Min; Frank, Martin: *RDFPeers: a scalable distributed RDF repository based on a structured peer-to-peer network.* In: *Proceedings of the 13th international conference on World Wide Web* ACM, 2004, pages 650–657

Captain 2015
Captain, Sean: These are the mobile sites leaking credit card data for up to 500, 000 people A day. In: *Fast Company* (2015), 12. http://www.fastcompany.com/3054411/these-are-the-faulty-apps-leaking-credit-card-data-for-up-to-500000-people-a-day

Carvalho et al.
Carvalho, Rodrigo; Goldsmith, Michael; Creese, Sadie; Police, Brazilian F.: Applying Semantic Technologies to Fight Online Banking Fraud.

Consumer Action 2009
Consumer Action: Questions and answers about credit card fraud A Q & consumer aCtion A consumer action publication. Version: 2009. http://www.consumer-action.org/downloads/english/Chase_CC_Fraud_Leaders.pdf. http://www.consumer-action.org/downloads/english/Chase_CC_Fraud_Leaders.pdf, 2009. – Forschungsbericht

DBpedia
DBpedia: *Online Access.* http://wiki.dbpedia.org/OnlineAccess#1.1%20Public%20SPARQL%20Endpoint

eBay Inc
eBay Inc: *eBay: Company Information.* https://www.ebayinc.com/

Google Patents
https://patents.google.com/?q=credit+card+fraud+prevention&after=20150101

Goyal & Fussell
Goyal, Nitesh; Fussell, Susan R.: Effects of Sensemaking Translucence on Distributed Collaborative Analysis.

Grigorik 2013
Grigorik, Ilya: *High Performance Browser Networking: What every web developer should know about networking and web performance.* " O'Reilly Media, Inc.", 2013

Grudin 1994
Grudin, J.: Computer-supported cooperative work: history and focus. In: *Computer* 27 (1994), May, Nr. 5, pages 19–26

Guha et al. 2016
Guha, RV; Brickley, Dan; Macbeth, Steve: Schema.org: Evolution of structured data on the web. In: *Communications of the ACM* 59 (2016), Nr. 2, pages 44–51

Hepp 2008
Hepp, Martin: Goodrelations: An ontology for describing products and services offers on the web. In: *Knowledge Engineering: Practice and Patterns*. Springer, 2008, pages 329–346

Hoffman et al. 2009
Hoffman, R. R.; Norman, D. O.; Vagners, J.: "Complex Sociotechnical Joint Cognitive Work Systems"? In: *IEEE Intelligent Systems* 24 (2009), May, Nr. 3, pages 82–c3

Holmes 2015
Holmes, Tamara E.: *Credit card fraud and ID theft statistics*. `http://www.creditcards.com/credit-card-news/credit-card-security-id-theft-fraud-statistics-1276.php`. Version: 09 2015

Initiative 2012
Initiative, Dublin Core M.: *DCMI Metadata Terms*. `http://dublincore.org/documents/dcmi-terms`. Version: 06 2012

Josuttis 2007
Josuttis, Nicolai M.: *SOA in practice: the art of distributed system design.* " O'Reilly Media, Inc.", 2007

Koch 2008
Koch, Michael: *CSCW and enterprise 2.0 - towards an integrated perspective*. In: *BLED 2008 Proceedings*, 2008

Lewis 2015
Lewis, Len: *More vulnerable than ever?* `https://nrf.com/news/more-vulnerable-ever`. Version: 12 2015

Parameswaran et al. 2001
Parameswaran, Manoj; Susarla, Anjana; Whinston, Andrew B.: P2P networking: An information-sharing alternative. In: *Computer* (2001), Nr. 7, pages 31–38

Pavan Podila 2013
Pavan Podila: *HTTP: The Protocol Every Web Developer Must Know - Part 1.* http://code.tutsplus.com/tutorials/http-the-protocol-every-web-developer-must-know-part-1--net-31177. Version: 04 2013

PYMNTS 2016
PYMNTS: *Hackers and their fraud attack methods.* http://www.pymnts.com/fraud-prevention/2016/benchmarking-hackers-and-their-attack-methods. Version: 02 2016

Rampton 2015
Rampton, John: How online fraud is a growing trend. In: *Forbes* (2015), 04. http://www.forbes.com/sites/johnrampton/2015/04/14/how-online-fraud-is-a-growing-trend/#16ffc0ec349f

Rana & Baria 2015
Rana, Priya J.; Baria, Jwalant: A Survey on Fraud Detection Techniques in Ecommerce. In: *International Journal of Computer Applications* 113 (2015), Nr. 14

Reuters 2015
Reuters: *Fraud rates on online transactions seen up during holidays: Study.* http://www.reuters.com/article/us-retail-fraud-idUSKCN0T611T20151117?feedType=RSS&feedName=technologyNews. Version: 11 2015

Rice 1992
Rice, Ronald E.: Task Analyzability, use of new media, and effectiveness: A multi-site exploration of media richness. In: *Organization Science* 3 (1992), 11, Nr. 4, pages 475–500. http://dx.doi.org/10.1287/orsc.3.4.475. – DOI 10.1287/orsc.3.4.475. – ISSN 1047–7039

Rietveld et al. 2015
Rietveld, Laurens; Verborgh, Ruben; Beek, Wouter; Vander Sande, Miel; Schlobach, Stefan: *Linked data-as-a-service: the semantic web redeployed.* In: *European Semantic Web Conference* Springer, 2015, pages 471–487

Robert & Dennis 2005
Robert, Lionel P.; Dennis, Alan R.: Paradox of richness: A cognitive model of media choice. In: *Professional Communication, IEEE Transactions on* 48 (2005), Nr. 1, pages 10–21

Rodrigues & Druschel 2010
RODRIGUES, Rodrigo; DRUSCHEL, Peter: Peer-to-peer systems. In: *Communications of the ACM* 53 (2010), Nr. 10, pages 72–82

R.V. Guha
R.V. GUHA: *Good Relations and Schema.org*. http://blog.schema.org/2012/11/good-relations-and-schemaorg.html

Schema.org a
SCHEMA.ORG: *Schema.org Extensions*. http://schema.org/docs/extension.html

Schema.org b
SCHEMA.ORG: *Welcome to Schema.org*. http://schema.org

Sen et al. 2015
SEN, Pritikana; AHMED, Rustam A.; ISLAM, Md R.: A Study on E-Commerce Security Issues and Solutions. (2015)

Sobko 2014
SOBKO, Oleg V.: Fraud in Non-Cash Transactions: Methods, Tendencies and Threats. In: *World Applied Sciences Journal* 29 (2014), Nr. 6, pages 774–778

Staab & Stuckenschmidt 2006
STAAB, Steffen (Hrsg.); STUCKENSCHMIDT, Heiner (Hrsg.): *Semantic web and peer-to-peer*. Springer Science + Business Media, 2006. http://dx.doi.org/10.1007/3-540-28347-1. http://dx.doi.org/10.1007/3-540-28347-1. – ISBN 9783540283461

TaskRabbit
https://www.taskrabbit.com/about

Taylor & Harrison 2008
TAYLOR, Ian J.; HARRISON, Andrew: *From P2P and grids to services on the web: evolving distributed communities*. Springer Science & Business Media, 2008

Virtue 2009
VIRTUE, Timothy M.: *Payment card industry data security standard handbook*. Wiley Online Library, 2009

Visa Europe 2014
VISA EUROPE: *Processing e-commerce payments*. https://www.visaeurope.com/media/images/processing%20e-commerce%20payments%20guide-73-17337.pdf. Version: 08 2014

Vis.js
VIS.JS: *vis.js showcase*. http://visjs.org/showcase/index.html

Vogt et al. 2013a
VOGT, Christian; WERNER, Max J.; SCHMIDT, Thomas C.: *Content-centric user*

networks: WebRTC as a path to name-based publishing. In: *Network Protocols (ICNP), 2013 21st IEEE International Conference on* IEEE, 2013, pages 1–3

Vogt et al. 2013b
Vogt, Christian; Werner, Max J.; Schmidt, Thomas C.: *Leveraging WebRTC for P2P content distribution in web browsers*. In: *Network Protocols (ICNP), 2013 21st IEEE International Conference on* IEEE, 2013, pages 1–2

W3C 2004
W3C: *OWL Web Ontology Language Guide*. `https://www.w3.org/TR/2004/REC-owl-guide-20040210/`. Version: 02 2004

W3C 2013
W3C: *W3C semantic web activity*. `https://www.w3.org/2001/sw/`. Version: 06 2013

Wood et al. 2014
Wood, David; Zaidman, Marsha; Ruth, Luke; Hausenblas, Michael: *Linked Data*. Manning Publications Co., 2014

Declaration in lieu of oath

I hereby declare that this master thesis was independently composed and authored by myself.

All content and ideas drawn directly or indirectly from external sources are indicated as such. All sources and materials that have been used are referred to in this thesis.

The thesis has not been submitted to any other examining body and has not been published.

Place, date and signature of student
Andreas Gerlach

Appendix