Improving the Logistic Sectors Efficiency using Service Oriented Architectures (SOA)

Robbin Harleman
University of Twente
P.O. Box 217, 7500AE Enschede
The Netherlands
r.j.s.harleman@student.utwente.nl

ABSTRACT
It is often reported that a Service Oriented Architecture (SOA) can improve the efficiency of a Logistic Service Provider (LSP). However, several inefficiencies today are caused by limited supply chain integration. Therefore, this research is looking for improving the efficiency of the whole Logistics Business Network (LBN) using SOAs. In an ideal situation, organizations use a service architecture and implement one identical open communication standard. However, such standards don’t exist (yet). Therefore this research investigates the openness, chances to success and quality of services offered by software packages used in logistics, using an Open Services Quality Model.

Keywords
Service Oriented Architecture (SOA), cargo transport, transport efficiency, Supply Chain Management (SCM), SCM efficiency, open web services, open web service API, API quality, open web service quality, Open Services Quality Model.

1. INTRODUCTION
In 2007 Cruissen [6] investigated 82 Flemish road transport organisations. His conclusions were that Flemish road trucking companies are operating at an unacceptable low efficiency rate. Meanwhile, a study by the Dutch government shows that transport volumes are increasing and the sector’s CO2 footprint has to be reduced by 60% in 2035 [19].

One way to achieve this CO2 reduction, is improving the sector’s efficiency. The main cause of the low efficiency rate is the lack of information sharing and coordination between Supply Chain Partners (SCP), according to [18] and [2]. SOA is seen as a promising technology to achieve better supply chain coordination and can also be used to optimize communication between SCP.

However, implementing a SOA within a company is not enough for sharing information with SCP. To achieve effective and efficient system integration, services should be open and should support real-time data exchange. It’s our assumption that openness is not well understood by both software vendors and users and therefore we will define openness and provide a model to test the openness and quality of services. In addition; a literature study is performed to find out whether there are standards emerging supporting real-time data exchange in LBN.

The paper is structured as follows; In section 2, the research questions are explained. Section 3 and 4 describe a literature search about the use of SOAs to improve the efficiency of the logistics sector. Section 5 describes a design study to develop a model to test the openness, chances to success and quality of open services. In section 6 this model is applied to software packages used in the logistics sector.

2. RESEARCH QUESTIONS
The goal of this research paper is to improve the efficiency of Logistics Business Networks (LBN), using SOA. In addition, this research will look at the current status of openness of software packages used in the LBN.

We define two research questions:
Q1: How can SOAs improve the logistic sector’s efficiency?
Q2: How open are the services provided by software packages used in logistics?

Research question Q1 cannot be answered before more is known about the logistics BN, the efficiency of LBN and about SOA. Sub-questions are:
Q1.1: How can we measure and improve LBN efficiency?
Q1.2: What are the main differences between a SOA and traditional software?
Q1.3: Which standards are emerging for (real-time) communication in logistics using SOA?

With the 3 answers of these sub-questions together, one should be able to say how the efficiency of the LBN can be improved.

Question Q2 needs an instrument to measure the openness of software packages. The term “openness” needs to be defined, because it is our assumption that openness is not well understood by software vendors and users. For question Q2, sub-questions are:
Q2.1: What is “openness” of services exposed by software packages?
Q2.2: Can we provide a model to rate the openness of services exposed by software packages?

With the definition of openness and a model to rate their openness, a number of software packages used in logistics are tested. The model should be a helpful tool for IT-managers to test the openness of the software packages that are being used in their logistic organizations.
3. HOW CAN SOA IMPROVE THE EFFICIENCY OF LBN?

3.1 How can we measure and improve LBN efficiency?
Several researchers have suggested ways to measure the performance and efficiency of LSP and LBN. For example: The balanced scorecard by Brewer & Speh and the profit and loss statements by Lambert & Pohlen. Unfortunately, these measures can only be applied on one organization and there is no generally accepted framework for measuring the efficiency of LBN [15].

It is essential for this research to define how efficiency is measured and later on how efficiency can be optimized using SOAs. After the definitions, several methods about the optimization of LBN efficiency with SOA are introduced.

The first measurement in this research is provided by “Transport en Logistiek Nederland” [14]. This Dutch government organization investigated the value of the load factor of trucks. This measurement can also be applied on other transportation methods such as transport by boat, plane or train. The research in [14] suggests that optimizing the load factor results in a more efficient LBN.

The second measurement is provided by the work of Cuijssen [6] and van Hillegersberg [15]. Efficiency of LSPs can be measured by analyzing the efficiency of the required resources needed to fulfill orders for SCP. The efficiency of a LBN depends on the utilization factor and the productivity of resources. Resources are defined as: working hours, trucks, warehouses, number of handlings etc. [6] and [15] suggested that optimizing the efficiency and productivity of required resources results in more efficient LBN.

Optimizing the efficiency of resources is not always as easy as it sounds. For example optimizing working hours. It is impossible to cut staff and without any changes accommodate the same amount of work. However, constructing more efficient (round)trips, and planning orders more efficient means that the same loads can be transported with less equipment and less labor, according to [7]. So planning is a very important factor in the logistic sector: Planning more efficient (round)trips and a more efficient order planning results in a more efficient logistic sector.

Every organization wants to, and possibly can improve his efficiency. The challenge however is to improve efficiency, and at the same time keep the required product quality and service levels. In general, optimizing efficiency is fulfilling orders requiring a minimum amount of resources and meeting the required service levels. Evaluating the service levels starts with choosing the right SCPs. Caris [4] showed that bundling loads and using multi-modal transport methods resulted in a more efficient supply chain. Because for every track where the right transport method was chosen, efficiency increased. The same arguments counts for other SCPs like warehousing partners, shipping partners etc. Conclusion: Choosing the right SCP results in a more efficient logistic sector.

The research of [7] showed that efficiency of larger organizations was in all cases higher, than the efficiency of smaller logistic organizations. This could be improved by horizontally cooperating in the sector. Bundling loads, sharing (round)trips and information about the location of trucks could optimize the LBN.

Another way to improve efficiency is the optimization of customer contact. If a LSP receives 100 calls a day and 80 of the calls are about the estimated time of arrival of the transported goods, the organization should consider implementing an IT system support these kind of questions. If the customer could check a website that confirms that the goods will be there on time, he won’t call the LSP anymore. So, sharing (the right) information results in better cooperation and coordination in the logistic sector, which results in a more efficient logistic sector.

The opposite way of providing information digitally, is achieving information digitally. Hoshino [11] showed that the planning, storing and transportation process could be far more efficient if information was available in early stages of the process. Having correct and enough information available is crucial to making the right decisions at the right moment. It can be a helpful tool to exchange information to improve LBN efficiency.

3.2 What are the main differences between SOA and traditional software?
Traditional software used in the logistic sector is usually a closed source package, designed for internal purposes (read: does not share information outside the business) and can only be extended by buying extra modules from the same supplier or paying large amounts of money to build a custom made solution. As Candido [3] concluded his research about software used in supply chains: traditional systems have four major limitations leading to unexpected underperformance:

1. Inflexibility to accommodate changes of supply chain structures
2. Lack of modular and open system architecture
3. Lack of functionality beyond managing transactions
4. Inability to share internal data efficiently with supply chain partners across organizational boundaries.

Cheng [12] and Obeidat [17] extended the research of Candido [3] and stated that ‘sharing information and integrating applications and services among supply chain members can help reduce cost, increase responsiveness and service levels, support decision making and enhance project transparency’. SOA is seen as a promising technology to achieve this integration of applications and sharing of information.

The SOA approach is very different from the traditional approach. ‘The [...] SOA application consists of services that provide service descriptions and communicate via messages.’ [3]. Every service functions as an independent component of the architecture. Such an architecture has a number of internal and external advantages for the organization (according to [3]). Internal advantages (advantages for the internal organization) of SOAs are:

- Easy updating and expanding of components because they can function independent from each other.
- Reusability of software-code.
- The ability to support Behavior Driven Design (BDD) because SOAs can easily being extended.
The differences between SOA and traditional software were explained, and several methods were pointed out to measure and optimize LBN efficiency. Section 3.2 seems to provide information about a large number of methods and measurements mentioned in section 3.1. The key of this section is to look at the advantages of SOA and to see if they effect the efficiency of LBN.

The first measurement about the optimization of the load factor and the second measurement about the optimization of the required resources are objectives for every supply chain organization. Optimizing these factors means that their organizations become more efficient and often more profitable. However, the work of [7] already indicated that the possibilities to optimize these factors were limited. [7] showed that efficiency of LBN could be optimized further by horizontally cooperating and sharing information. Section 3.2 showed that SOA is seen as a promising technology to achieve this cooperation.

According to [11], more efficient planning can be done when more information is available. The greater ability of SOAs (compared to traditional systems) to be able to exchange information with SCP results in better cooperation and coordination and planning between SCP, according to [21].

Section 3.2 was closed with the statement that IT could be a helpful tool to support information exchange. In addition, this section also showed that information exchange was necessary to optimize the efficiency of LBN. However, in this situation as first SCP have to be selected and after that an information exchange is possible. A more favorable situation is when SOAs are able to exchange information automatically without the costs of building a new B2B-integration for every supply chain partner. Therefore standards for exchanging information in the logistic sector is needed. With such standards every logistic organization is able to communicate with partners using the language defined in the standard.

Such standards should make it possible to exchange information with SCP, make organizations more efficient and possibly the SCM-process becomes cheaper [13]. In addition, it should be possible to select the right SCP, due to the automatic alignment of the requirements and possibilities of the SCP communicated through the standard.

3.3 Can SOA improve the efficiency of LBN?

In the previous sections, the differences between SOA and traditional software were explained, and several methods were pointed out to measure and optimize LBN efficiency. This system aims to exchange data. A few of these standards are EDIFACT, RosettaNet, STEP and AnsiX12; all of these are EDI-based standards [1] [16]. An Electronic Data Interchange (EDI) message is pushed from one organization to another on a predefined time. Therefore, this standard is not suitable for real-time information exchange. Data might be outdated because information is updated, and is not directly send in an EDI message.

Another disadvantage of EDI messages is that the sender is not able to detect how the information is being used. The receiver might be interested in more data or can use only half of the information send in an EDI message. When the receiver was able to pull messages, instead of waiting until a new message is pushed to him, the data provider is able to see what information is used by whom and when the information is requested.

An interview with Modality (software company building terminal management software) pointed out that terminals are continuing integrating EDI connections, because it’s better than having no interconnectivity at all. There are terminals nowadays that are still implementing this technology, even though lots of researchers pointed out that newer alternatives like XML messages requested from open web services are widely available and can easily be implemented.

The work of [22] showed that organizations are scared to implement new technologies before there is an official standard. It is far from optimal to implement and support multiple standards. Therefore, organizations might decide not to implement new technologies at all.

Conclusion: we need to have standards defined before large numbers of organizations decide to switch from the EDI push standard to some pull standard (most likely in an open format, like XML).

There are a few XML standards available for subsectors of the logistics sector. For short sea operations, the short-sea XML standard is developed [20]. This standard is developed with the purpose of: reducing EDI implementation costs, improving vessel utilization, saving man hours, reducing administrative costs etc. It is unknown how many organizations use this standard.

Another example is the River Information System (RIS) developed by the European Union. This system aims to collect data about vessels, and tracking and tracing vessels, based on the Automatic Identification System (AIS) on board of ships. Another example in this direction, is AISHUB. Members of AISHUB need to buy a shore station to monitor passing ships. They record the ships identity, destination, speed, course and other relevant information. This information is send to AISHUB. In return members can use all other data available at AISHUB for free.

So a number of standards are available for subsectors of the logistics sector, but it seems that not all sectors are covered.

4. WHICH STANDARDS ARE EMERGING FOR (REAL-TIME) COMMUNICATION IN THE LOGISTICS SECTOR?

It seems that there are a number of standards developed in subsectors of the logistic sector with the purpose to exchange data. A few of these standards are EDIFACT, RosettaNet, STEP and AnsiX12; all of these are EDI-based standards [1] [16]. An Electronic Data Interchange (EDI) message is pushed from one organization to another on a predefined time. Therefore, this standard is not suitable for real-time information exchange. Data might be outdated because information is updated, and is not directly send in an EDI message.

Another disadvantage of EDI messages is that the sender is not able to detect how the information is being used. The receiver might be interested in more data or can use only half of the information send in an EDI message. When the receiver was able to pull messages, instead of waiting until a new message is pushed to him, the data provider is able to see what information is used by whom and when the information is requested.

An interview with Modality (software company building terminal management software) pointed out that terminals are continuing integrating EDI connections, because it’s better than having no interconnectivity at all. There are terminals nowadays that are still implementing this technology, even though lots of researchers pointed out that newer alternatives like XML messages requested from open web services are widely available and can easily be implemented.

The work of [22] showed that organizations are scared to implement new technologies before there is an official standard. It is far from optimal to implement and support multiple standards. Therefore, organizations might decide not to implement new technologies at all.

Conclusion: we need to have standards defined before large numbers of organizations decide to switch from the EDI push standard to some pull standard (most likely in an open format, like XML).

There are a few XML standards available for subsectors of the logistics sector. For short sea operations, the short-sea XML standard is developed [20]. This standard is developed with the purpose of: reducing EDI implementation costs, improving vessel utilization, saving man hours, reducing administrative costs etc. It is unknown how many organizations use this standard.

Another example is the River Information System (RIS) developed by the European Union. This system aims to collect data about vessels, and tracking and tracing vessels, based on the Automatic Identification System (AIS) on board of ships. Another example in this direction, is AISHUB. Members of AISHUB need to buy a shore station to monitor passing ships. They record the ships identity, destination, speed, course and other relevant information. This information is send to AISHUB. In return members can use all other data available at AISHUB for free.

So a number of standards are available for subsectors of the logistics sector, but it seems that not all sectors are covered.
A LBN can only be successful if the full network is covered and standardized. Therefore, more standards need to be developed.

5. CAN WE PROVIDE A MODEL TO RATE THE QUALITY AND OPENNESS OF SOFTWARE?
Openness and quality are terms used a lot nowadays, but the words itself have no meaning. As an example: Microsoft used the term openness on their product Microsoft Word because their product was widely available [9]. In section 5.1 we will attach some definitions to openness and quality of services. In section 5.2 we will introduce the Open Services Quality Model (OSQM), verify the model in section 5.3 and test it on logistic software packages in section 6.

5.1 Defining openness and quality
Openness in this research is meant as opening software or being able to exchange data with the software. The goal of the this definition is to help IT managers making better decisions on buying software that helps not only their own business, but helps the sector by making standardized B2B implementations.

If every IT-Manager makes the right decision for the right software for his organization, while using valid arguments; not only his own organization should profit, but the sector as a whole. The efficiency of all businesses together is eventually equal to the efficiency of the sector. Section 3 already proved that the total efficiency of the logistics sector - when there is efficient data exchanged - can be higher, than the cumulative efficiency of all individual organizations.

The right decision in this case is a decision where the IT manager chooses:
A. A high-quality\(^1\), open software package\(^2\);
B. Using open services to support efficient communication with customers, colleagues and partners\(^3\);
C. Using a standard and open communication-protocol\(^4\).

1 The quality of the software is tested in the model itself. Various quality measurements are introduced in section 5.2.
2 Open software in this context means opening the software, or being able to exchange data with the software. In the case of a SOA through open services.
3 Supporting communication with external parties is necessary to improve the efficiency of transport organizations. This was argued earlier in section 3.
4 A standard and open protocol is part of the quality model, because the more open the software is, the more flexible it will be.

The OSQM is based on the research of Folmer [8], the ISO 9000 standard on software quality, the work of Krehmer on open software requirements, the presentation of Even Cooke about building great web APIs and the presentation of Zach Maier on the Google I/O conference about how Google builds web service APIs [10].

The OSQM is based on the assumption that the most qualitative model will succeed. This does not imply that the most qualitative model (according to the OSQM) is of superior quality. The guidelines of the OSQM should help managers to make fundamented decisions when choosing software to support open services. Because this is one of the first works on the quality of open services future work has to be done to prove the model’s completeness and quality.

5.2 The Open Services Quality Model (OSQM)
To achieve the business goals as stated in section 5.1 the OSQM is divided in four parts; respectively the service (and the attached service API) in general, technology, (implementation)-support and versioning. Asking the following questions makes IT-managers think critical about a product before buying it. If multiple service APIs are checked, a valid comparison can be made.

5.2.1 The service in general
- Is the service API published online?
- Business value for all parties using the service and thus supporting B2B integration?
- Pricing terms.
  - What is the business model? Pay per request/per month? 
  - Who has to pay for (using) the service?

To start with the last argument, pricing terms are important for every investment. However, decisions about choosing one service provider or another can only be made if clear pricing terms are available.

The most important part of quality is that the service should do what is has to do: offer business value for both the provider of a service and the service-users. In the logistics sector the most important business value for achieving greater efficiency is information sharing or profound B2B-integration.

5.2.2 Technology:
- Which protocol is being used for exchanging messages?
  - Is the protocol open?
  - Are additional modules needed for de-encrypting response messages?
- Which message formats are supported?
- Are standards being used inside the service?
  - For example time formats in GMT, or locations by GPS coordinates?
- What is the maximum response time (according to vendor or by testing the service)?
- Is there a maximum amount of queries per hour being supported?
- How is the communication being secured?
  - Is there a security policy?
  - For example firewalling, blocking accounts in case of a suspected hack or expiring access tokens over time?

On operational levels one would measure the quality of a service in terms of technical quality and costs of supporting/implementing the service. Therefore one would like to know which technologies are used for exchanging data
and what message format is supported. The most favorable case is that the HTTP protocol is being used with a (mainstream) open message format like XML or JSON. This architecture is future-proof and easy to translate, and connect to already installed systems due to its openness.

Another type of quality measurement are limitations that may be available in terms of number of requests or maximum response time. In case of large response time, a callback URL is required to not slow down the users application [5].

The last item of technical quality measurement is the security policy. The challenge in security is finding a balance between maximum secured connection versus easy implementation. A possible solution to fix this issue (according to Evan Cooke [5]) is setting up a testing environment where the user can test the service API without the “limitations” of security. If test results are good, the provider can help the user by giving code examples to facilitate easy and quick implementation.

5.2.3 (Implementation) support:
- Is there a testing environment?
  - What has the service-provider done to help you debug?
  - Is there a history of performed requests?
  - Support department available?
- What is the completeness and quality of the documentation?
  - Is every function documented?
  - Is every input documented?
  - Is every output documented?
  - Are code examples (how to use the functions) available?
- Estimate learning time?

Let’s now make another step further and assume that the service is tested positive on offering business values and technical quality. It’s just a matter of implementing the service and the B2B connection is up and running.

As mentioned in the previous section, a testing environment would really help the user with his implementation. Other things a provider could do to help implementing is offering a history of performed requests. With this history and extensive documentation, errors could be easily resolved. If error codes or service functionality are not documented this model would automatically discourage the use of the service(s). The golden rule is: the more documentation, the better!

5.2.4 Versioning
- Is there a version?
- Is there a version policy?
  - How long are older versions being supported?
  - Are deprecated functions being announced? How long in front of deprecation?
- Is it clear how version(s) are being implemented in the service (API)?
- Number of changes per version release.

Assuming that services are used in the logistics sector, the OSQM should be able to measure the future proofness of the service. The only way of being future proof is having a clear policy on versioning. A version policy should describe how long older versions are supported and what the policy is regarding (announcing) deprecated functions. Most favorable is to support older functions and that every new release has its own URL or parameter in the service API.

Folmer’s research [8] operationalized version quality in terms of the number of changes per release. Too much changes indicated that the product wasn’t stable. Too little changes was an indication of a lack of support and innovation in the product.

5.3 Verification

Before using the model on services used with software packages in the logistics sector, the model needs to be checked on completeness, usability and correctness. To test the model high quality service APIs that are being used a lot on the Internet are checked on openness, completeness and quality. The OSQM should be able to test major service APIs, but should also support developing services and give some guideline to develop high quality open service.

5.3.1 Test results on high-quality services on the Internet using the OSQM.

Two service APIs are tested on openness and quality: the Twitter REST API and the Google Maps Javascript API. The evaluation of the Google service API can be found in Appendix A.

To prove the internal validity of this model, the test results on the Twitter rest API are discussed in the next section. Not many software packages in the logistics sector support open services (prove follows in section 6), and therefore probably need some guideline in order to achieve high-quality service APIs. Because this is the first version of the Twitter API, web services developed in the near future (for the logistics sector) evolve through the same stages as the Twitter API. Therefore the Twitter API was chosen to test the validity of the OSQM. The Google Maps API exists for 7 years now, so the Google API was used to test the OSQM on a mature API.

5.3.1.1 Twitter REST service API.

Business value for each party?

Yes, there seems to be a business value for Twitter and also for the programmer. Twitter wants to grow as a platform. The programmer wants to make use of the (real-time) information on the platform and has access via the platform. The service API is also an ideal solution for platform- and device in dependability. It’s now the users choice how he’d like to use the platform Twitter.

Pricing terms. What is the business model? Pay per request/per month?

The Twitter service API is free to use, without any costs. Twitter does earn money with promoted tweets (adds). It is not clear if promoted tweets are also pushed through the service API. I suppose they are.

5.3.1.2 Versioning

Is there a version?

Yes, we are testing version 1.0 of the Twitter REST service API.

Is it clear how version(s) are being used?
Yes, the version should be the first parameter in the URL the requests are being send to.

Is there a version policy?

No, since this is the first version of the service API, there is no official policy. They promise on the support blog they will support older versions, but do not clarify how many and how long older versions are being supported.

Number of changes per version release.

Not tested since this is the first version of the service API.

5.3.1.3 Technology

Is an open standard being used for sending and receiving data via the service API?

Yes. The user can choose the format. Options are: RSS, JSON, atom or XML.

Are standards being used inside the service API (for example time formats in GMT, or locations by GPS coordinates)?

Yes, times are in GMT with the corresponding time zone or as time expired since the tweet was posted.

Maximum response time (according to vendor or by testing the service API)

Twitter doesn’t specify the maximum amount of requests per hour. If the user experiences slow responses when working with large amounts of data, one is advised to work with the streaming API. This API is also advised to use when real-time data is needed.

Tested response times are more than good. Large amounts of data (hundreds of lines of text) are send/received in less than a second.

How is the communication being secured?

Twitter is using oAuth 1.0. Optional is the use of an SSL secured connection.

Security policy?

Twitter doesn’t have a security policy other than an abuse department for spam-complaints. Access tokens do not expire.

Testing environment

There also isn’t a testing environment. Twitter has 14 classes available for different programming languages to help you setup the oAuth connection with the service API.

5.3.1.4 Support

What has the service API-provider done to help you debug?

The error messages are very extensive and clearly documented.

Is there a history of performed requests?

No, there is no history.

Completeness and quality of documentation

The documentation is complete and of high-quality. Every function, input parameters and output parameters (with examples) are documented. There are also code examples of how-to use the functions.

Estimate learning time?

Twitter doesn’t indicate learning time. They’ve done everything to help you getting started, so it shouldn’t take longer than necessary.

Support department available?

No, there is no support department. Just a forum where you are being helped by other programmers.

5.3.2 Verification conclusions

The OSQM is now tested in two high-tech often used services and looks operational. However, the services used for the tests aren’t built for B2B integration, but just for opening a platform or offering a service. Therefore the test only pointed out that the technical aspects of the model are all right.

Interviewing an operations manager on a Dutch short-sea terminal, the following points were concluded:

1. For B2B integration it is highly important that both parties (the service API provider and the service API user) are profiting on the use of the service API because both parties have to invest in implementing the service API.

2. If the service API can be bought as part of a transport management software package pricing terms are important. It should be clear what the business model is. For example, pay per request, fixed term per month etc.

6. HOW OPEN ARE SOFTWARE PACKAGES IN THE LOGISTICS SECTOR?

Several packages used in the logistics sector still (proudly) support EDI connections. These packages were not suitable to be tested with the OSQM. It was hard to find packages that did support open service APIs to test with the OSQM. In table 1 a quick scan can be found on checking software packages that might be suitable for testing with the OSQM. Table 1 only contains software vendors that mention the openness of their software online. Various vendors could be found that don’t even mention if they support EDI implementations or other forms of openness.

Table 1. Quick scan openness logistic software

<table>
<thead>
<tr>
<th>Service API available?</th>
<th>SOAP</th>
<th>CSV</th>
<th>XML</th>
<th>MySql</th>
<th>Others</th>
</tr>
</thead>
<tbody>
<tr>
<td>TomTom</td>
<td>Yes</td>
<td>Yes</td>
<td>SOAP</td>
<td>/ No</td>
<td>CSV</td>
</tr>
<tr>
<td>Carrierweb</td>
<td>Yes</td>
<td>No</td>
<td>XML</td>
<td>/ Yes</td>
<td>MySql</td>
</tr>
<tr>
<td>Cargonaut</td>
<td>No</td>
<td>-</td>
<td>-</td>
<td>-</td>
<td>Yes</td>
</tr>
<tr>
<td>PortBase</td>
<td>No</td>
<td>-</td>
<td>-</td>
<td>-</td>
<td>Yes</td>
</tr>
<tr>
<td>Centric SCM</td>
<td>Yes</td>
<td>No</td>
<td>XML</td>
<td>/ Yes</td>
<td>SOAP</td>
</tr>
</tbody>
</table>

Tomtom is a relatively new vendor of vehicle management and navigation software. The weblfleet connect package is sold as a new and innovative transport management system (TMS). New and innovative functions such as vehicle tracking, communication with drivers, optimizing business efficiency and others are included in the package. Such an
interesting innovative package is a good one to test with the OSQM. The full test results are added in appendix B.

The most important improvements TomTom should make are:

- The service sAPI is a good first step to optimize the internal organization. However, improvement could be made to optimize the whole sector’s efficiency. These improvements include almost all measurements / efficiency improvements suggested in section 3.1.
- The package isn’t complete. The user has to do things the service API could do.
- There is no (clear) version policy, so future proofness is not guaranteed.
- The service API demands the user to comply to certain standard while the service API explicitly doesn’t implies these standards.

Although important improvements are suggested, TomTom is far more innovative than other (transport management) software-vendors. This service API is a good first step in opening TomToms software.

7. CONCLUSIONS

Q1: How can SOAs improve the logistic sector’s efficiency?

Service Oriented Architectures (SOAs) can improve the efficiency of the logistics sector due to facilitating cooperation and coordination in the sector. A number of measurements and optimizations are introduced for achieving more efficiency in LBN. Main reason for achieving higher efficiency levels is the exchange of information between SCP. Information can easily be exchanged when SOAs support open services. However, standards to exchange information needs to be developed. When a standard for exchanging real-time information is developed, the result is that the total efficiency of the logistics sector, assuming that cooperation and coordination is highly efficient, can be higher than the cumulative efficiency of all individual organizations.

Q2: How open are the services provided by software packages used in the logistics sector?

In this research openness was defined as: “Opening software to achieve B2B-integration”. Software may be closed source, but in that case a well-documented service API is needed. Openness of software in the logistics sector is (mainly) interpreted as “supporting EDI connections”, but EDI is not suitable for real-time information exchange. Most important conclusion on this question is that “openness” of software is not well understood.

In order to help software vendors build more open software resulting in a more efficient logistics sector, an OSQM is developed. This model is a first tool to evaluate the openness, completeness, quality and future proofness of software supporting a web service API, designed using the SOA approach.

8. FUTURE WORK

Because this research provided the first work on completeness, quality, openness and future proofness of web services future work needs to be done. The OSQM needs to be extensively tested and possibly complemented.

The OSQM is developed to test web services in the logistics sector. However, it can also be applied to other sectors to improve their efficiency using SOAs and (web) services. Most favorable is a sector were standards exist for information exchange.

At last, more research can be done in the logistic sector to optimize the efficiency. The sector has challenging goals (60% reduction of the CO2 footprint) and a lot of work needs to be done to achieve this. One research example in this context is to improve the load factor by bundling loads using services to exchange information between supply chain partners.

9. ACKNOWLEDGMENTS

Special thanks go out to my supervisor at the university, prof. dr. Jos van Hillegersberg, who provided valuable feedback for writing the report and conducting the research. He also managed to send me in the right direction and provided lots of helpful resources to find important information.

10. REFERENCES


APPENDIX

A. GOOGLE MAPS JAVASCRIPT API TESTED WITH THE OSQM.

10.1.1 The service API in general
Business value for all parties using the service API and thus supporting B2B integration?
I don’t really get why Google offers the Google Maps API. The business model of the Google maps website is to earn money when showing adds on the map, but Google doesn’t show adds via the API. The only business value for Google supporting the maps API is to earn money from paying customers with large amounts of requests.

Pricing terms:
- What is the business model? Pay per request/per month?
- Who has to pay for (using) the service API?
When the user needs more than 25000 requests per day, every 1000 extra requests costs $4 - $10 (depending of the version of the service API is being used).

10.1.2 Technology:
Which protocol is being used for exchanging messages?
- Is the standard open?
- Are additional modules needed for de-encrypting response messages?
- Which message formats are supported?
Messages are exchanged in JSON or XML format over the HTTP protocol. The message exchange is fully open, without extra requirements for encryption.

Are standards being used inside the service API?
- For example time formats in GMT, or locations by GPS coordinates

Yes, Google uses geo coordinates to show an address on the map.
- What is the maximum response time (according to vendor or by testing the service API)?

Google doesn’t indicate a maximum response time, but tests indicate that a few milliseconds are the average for locating an address on the map. Another advantage is that Google allows the use of callback urls for large queries.
- Is there a maximum amount of queries per hour being supported?
Yes, the maximum is 25000 requests. If one likes to use more, 1000 requests costs between $4 and $10 depending on the version that is being used.
- How is the communication being secured?
Only OAuth 2.0 + Use can be limited to one site (or ip address) + 14 libraries for different programming languages available + SSL optional
- Is there a security policy?
  - For example firewalling, blocking accounts in case of a suspected hack of expiring access tokens over time?

No. There is no security policy.

10.1.3 (Implementation) support:
Is there a testing environment?
Yes, for private purpose. Payed environment for businesses.

What has the service API-provider done to help you debug?
- Is there a history of performed requests?
  - No
- Completeness and quality of documentation
  - Complete service API, every function, input parameter, output
    - No request examples
- Support department available?
  - FAQs, Forum support, getting started manual.
Google support for bugs. + release notes per mail.

Estimate learning time?
Not indicated.

10.1.4 Versioning
Is there a version?
Yes, the newest (and tested) version is 3.8.

Is there a version policy?

- How long are older versions being supported?
- Are deprecated functions being announced? How long in front of deprecation?
  - Yes. Deprecation policy: older functions are being supported till one year after deprecation announcement or till 20-04-2015. Version policy: New or deprecated functions: version + 0.1

Is it clear how version(s) are being implemented in the service API?

Version call in url.
Number of changes per version release.
2-5 fixes per release
1-6 noticeable changes per release
4 version upgrades in 2011

B. OSQM ON TOMTOM WEBFLEET
.CONNECT API
Business value for each party?
No. TomTom assures their product is ideal for the logistics sector and is causing decreased fuel usage and optimized efficiency (in terms of avoiding traffic jams, order management, vehicle tracing and time management).
However, TomTom does not support an open service API in terms of creating new B2B relationships. Major keypoint of the Tomtom API is optimizing the internal organization.

Pricing terms. What is the business model? Pay per request/per month?
To use the TomTom WebFleet API, one needs a WebFleet Connect account and pays a monthly fee to use this. Pricing terms are not available online.

### 10.1.4.1 Versioning

**Is there a version?**

Yes, current version seems to be 1.12.0.

**Is it clear how version(s) are being used?**

No, there are versions, but no versions are mentioned in the service API itself.

**Is there a version policy?**

No, a version policy is not available (online). Another downside on versioning is that functionalities seem to be added and suspended in the same version.

**Number of changes per version release.**

AVG: 5.5 updates per year
AVG: 6.5 functions updated per release

### 10.1.4.2 Technology

**Is an open standard being used for sending and receiving data via the service API?**

There are two types of requests available for the user; SOAP or CSV. However, TomTom thinks CSV stands for Character separated values instead of comma separated. They also use an MTOM extension for SOAP messages for which a proprietary library is required.

**Are standards being used inside the service API (for example time formats in GMT, or locations by GPS coordinates)?**

No, explicitly not! Tomtom recommends users to use the ISO8601 standard for exchanging date/time values. However, Tomtom explicitly mentions that a return value of date-time together with the time zone is NOT guaranteed. It can’t be worse! Don’t ask a user to imply a standard and return messages that do not imply the standard!

The IETF RFC 2616: HTTP 1.1 protocol for exchanging data is used for sending/receiving messages over the HTTP protocol.

Tomtom does not explicitly mention that they imply the RFC 4080 standard for CSV messages or the W3C standard for SOAP (but it looks like they do).

**Maximum response time (according to vendor or by testing the service API)**

A maximum response time is not available. The number of queries is limited, but the limit is not given.

### How is the communication being secured?

An API key, username and password are required to use the API. An SSL is optional (for CSV). SSL connection is required when the SOAP protocol is being used, but encryption is not allowed in both cases.

**Security policy?**

Not available.

**Testing environment**

Not available.

**Is the service API complete (doesn’t let the user do anything the API can do)?**

No.

- The ordering of return messages is random. The user has to order the messages.
- Tomtom doesn’t imply the standards the user is required to implement, so the user has to check every response.

### 10.1.4.3 Support

**What has the service API-provider done to help you debug?**

A support department is available (and they’ll possibly help with the debugging).

**Is there a history of performed requests?**

No, there is no history.

**Completeness and quality of documentation**

The documentation is complete: all functions and their parameters are described well. However, there are response codes that are undocumented, or have a description like “general error”.

**Estimate learning time?**

Not indicated, but is certainly not optimal because of the weird (or the lack of) implementation of standards and the random ordering of return messages.

**Support department available?**

Yes.