Towards an Information Architecture to enable Cross Chain Control Centers

Simon Dalmolen
Industrial Engineering and Business Information Systems
University of Twente
Enschede, The Netherlands
s.dalmolen@utwente.nl

and
CGI Business Consulting
simon.dalmolen@cgi.com

Hans Moonen
Industrial Engineering and Business Information Systems
University of Twente
Enschede, The Netherlands
hans.moonen@utwente.nl

&
CGI Business Consulting
h.moonen@cgi.com

Jos van Hillegersberg
Industrial Engineering and Business Information Systems
University of Twente
Enschede, The Netherlands
j.vanhillegersberg@utwente
Introduction

Cross chain control centers (4C) have received increasing attention. Operating at the intersection of science and practice, Dutch Logistics Topinstitute Dinalog positioned the so-called 4C centrally in its research agenda. A 4Cs, could either physical (“an overall supply chain cockpit”) fully virtual, or a mix. However, their large scale implementation is hindered by several barriers including the lack of effective governance mechanisms, potential conflicting goals, limited willingness to share data and unclear or ill-defined business models and gain sharing mechanisms. Moreover, the lack of proper ICT support can be a key hindrance to the feasibility of a 4c. While ICT today offers many paradigms and technologies to support internal process automation or static business to business integration, the creation of a 4C requires far more advanced IT architectures. Current ERP systems are not fit to this task.

Traditional ICT support for SCM has been limited to (often cumbersome) static horizontal and vertical integration of enterprise systems. The IT links established are usually limited to coordination and control at the operational level in the context of fixed collaboration patterns. This does enable useful functionality such as tracking and tracing of goods, exceptions and alerts in case of delays and so on.

Already in 1966, Felix Kaufman published an article in Harvard Business Review that called for experiments with ICT that would cross organizational boundaries (Kaufman, 1966). However, studies have shown that decades later ICT may be both an enabler as well as a disabler to agile business networks. Enterprise Systems Integration projects may take years and huge investments to complete. Connecting legacy and ERP systems of various partners is technically highly complex. The resulting “hard-wired” links often do not enable agile business networks that allow business partners to quickly connect their business processes.

Numerous authors have investigated this issue. For example, a Delphi study by Akkermans et al. (2003) revealed that the following key limitations of ERP systems in providing effective SCM support emerge as: “(1) their insufficient extended enterprise functionality in crossing organizational boundaries; (2) their inflexibility to ever-changing supply chain needs, (3) their lack of functionality beyond managing transactions, and (4) their closed and non-modular system architecture”. In a more recent Delphi panel Daniel and White (2005) investigate the potential of improved support of inter-organizational linkages by emerging ICT. Their findings suggest that “ERP systems may be reaching a structural limit concerning their capabilities and adjunct technologies will be required to integrate multiple inter-organisational operations”. These include a combination of electronic hubs, web services, widespread adoption of common enterprise resource planning (ERP) systems and enterprise portals.

Van Hillegersberg et al. (2004) develop a typical virtual organization scenario using webservices and conclude that the technology provides clear benefits: “Webservices will truly allow straightforward B2B integration using standard and low-cost internet technology. This is a major advantage in enabling business networks, as small companies within these networks usually do not have the knowledge, time and money to implement traditional and complex Enterprise Integration technologies…network orchestration could be designed mostly separately from the various systems available in the business network”. However, the authors also stress that the orchestration technologies may have scalability and security issues. Furthermore, to truly design a collaborative and intelligent network integration, contracting and collaboration tools are required as well.

Inflexible and cumbersome integration of ICT systems does not empower a true strategic 4C, in which flexibility is a key issue. Therefore, we focus on ICT support for agile business network integration. Such ICT platforms will enable a strategic 4C in which business services of 3PL’s can be found in advanced registries, evaluated and seamlessly integrated and deployed into supply chains.

The proposed ICT architecture departs from the traditional static ICT architectures and makes use of mechanisms to enable swift service integration. Such an ICT architecture enables the 4C’s concept at both the operational level (tracking, tracing, planning and execution) as well as the tactic and strategic levels (business network formation, alliance building, service pricing and evaluations).

Key requirements for the ICT architecture are:

- Modularization of Services, Product, Process Services
- Coordination and Collaboration Capability
- Quick connect capability
• Relationship Management Capability
• Risk Management Capability

In this chapter we describe these requirements and zoom in on ICT capabilities for swift business to business integration.

Requirements of an Architecture to support 4c

Cross chain control towers basically enable and facilitate inter-organizational relationships between actors in the supply chain. In general, inter-organizational relationships require careful governance. A comprehensive set of joint processes and practices is needed to achieve a successful sourcing relationship. We focus here on capabilities that are key to achieving agile business networks. Several key capabilities of agile business networks have been described in literature:

Modularization of Services, Product, Process – Products and Services offered and the business processes supporting these have a modular structure. Such a modular structure enables effective sourcing, coordination and integration of logistics services (Tanriverdi, Konana, & Ge, 2007). Quality of the services can be precisely specified and assessed. Pricing schemes allow for price comparisons (Hoogeweegen, Teunissen, Vervest, & Wagenaar, 1999).

Coordination and Collaboration Capability - These are clearly key in agile business networks. As defined by (Thompson, 1967), coordination comprises the protocols, tasks and decision-making mechanisms designed to achieve concerted action between interdependent units. As outlined by (Dekker, 2004), both formal and informal control mechanisms can be applied to coordinate the inter-organizational relationship (see. Table 1)

Table 1: formal and informal control mechanisms, source (Dekker, 2004)

<table>
<thead>
<tr>
<th>Outcome control</th>
<th>Behavior control</th>
<th>Social control</th>
</tr>
</thead>
<tbody>
<tr>
<td>Ex-ante mechanisms</td>
<td>Structural specifications</td>
<td>Partner selection</td>
</tr>
<tr>
<td>Goal setting: Incentive systems/reward structures</td>
<td>Planning</td>
<td>Trust (goodwill/capability):</td>
</tr>
<tr>
<td></td>
<td>Procedures</td>
<td>-Interaction</td>
</tr>
<tr>
<td></td>
<td>Rules and regulations</td>
<td>-Reputation</td>
</tr>
<tr>
<td>Ex-post mechanisms</td>
<td>Behavior monitoring and rewarding</td>
<td>-Social networks</td>
</tr>
<tr>
<td>Performance monitoring and rewarding</td>
<td>Trust building:</td>
<td>Risk taking</td>
</tr>
<tr>
<td></td>
<td></td>
<td>Joint decision making and problem solving</td>
</tr>
<tr>
<td></td>
<td></td>
<td>Partner development</td>
</tr>
</tbody>
</table>

Quick connect capability – support integration and quick-connect and quick-disconnect capabilities to external partners. These include searching, contracting, monitoring and enacting services. Such capabilities are needed from the business contract level to the technical infrastructure level (Goldman, Nagel, & Preiss, 1995)(Konsynski & Tiwana, 2005)(Van Heck & Vervest, 2007);

Relationship Management Capability – In agile networks, there is little time to build subjective loyalty between network partners. Therefore, according to Mowshowitz (1997) there is only room for “objective loyalty that is based on reasoned self-interest”. Trust cannot be based on long term relationships and past performance either. Therefore agile business networks need to find alternative mechanisms to ensure trust and loyalty. Aziz et al. (2010) point out that capabilities such as high quality and formal communications between partners, adaptation of processes, and conflict resolution to higher performance in an inter-organisational relationship.

Risk Management Capability – The dynamically formed reciprocal relationships in agile business networks often do not have a stable history. Both at an organizational and technical level building networked relationships are high risk activities (Kumar & Van Dissel, 1996). At both technical and organizational levels semantic misunderstandings easily occur. The lack of high quality semantic standards in many industries increases this risk (Folmer, Oude Luttighuis, & van Hillegersberg, 2011).
An integration architecture to enable 4c

In this section we focus on two requirements for the ICT architecture to support 4c. (1) Coordination and Collaboration Capability and (2) Quick connect capability. Traditionally, closed and proprietary IT systems have hindered effective collaboration between business partners (horizontal collaboration) and between suppliers and customers (vertical collaboration). Recently, standard API based integration has emerged in other areas such as social media (e.g. Twitter, LinkedIn, Facebook) as a very rapid way to integrate services crossing inter-organizational boundaries. API’s also start to serve as a basis for new innovative initiatives. We see a phenomenon that the business is giving to the consumer (user) a so called application-programming interface (API). This is a set of routines and protocols, so that programmers can write applications using information from the business such as Twitter API – querying a certain hash key via programmable routines instead of using the mobile app by hand. In practice we see that the adoption of B2C is easier due to the extra given functionality of using the offered services. Therefore there is more space for innovation e.g. connecting Twitter with Facebook, and LinkedIn and doing data analysis. APIs seem to be much more than a technical issue. Still, research into their nature and impact is very scarce. In this paper we illustrate how APIs can contribute to rapid and successful innovation using. We illustrate how setting up collaboration through joint API design can facilitate information sharing agreements among supply chain partners and enable effective 4c.

This chapter will discuss a case study where logistic service providers (LSP) are willing to collaborate instead of competing with each other. However in practice, setting up a horizontal collaboration is challenging (Cruijssen, 2006). Still, nearly 70% of LSP companies in Benelux indicated that they have either already implemented horizontal cooperation or plan to do so within the next 4-5 years (EyeForTransport, 2010).

Ecosystem enabling supply chain collaboration

In optimizing supply chains, sharing data and information should be more supported. A virtual ecosystem in which each organization uses procedures, rules, and standards for sharing information could provide a solution to these challenges.

![Virtual Ecosystem Diagram](image_url)

The virtual ecosystem supports cross chain collaboration in transportation of containers and products, see Figure 1. It is possible for one or more containers to create a digital shadow where the information can be categorized virtually. This virtual container uses an e-dossier. It contains the characteristics of the load, the location, but also the limitation/boundaries of the cargo, such as to maintain constant temperature and humidity. The partners are allowed to access the data, depending on their access rights in the dossier or certain parts of the dossier. This digital shadow can also be used for trucks and ships. This means that organizations have the control of who may view (access) their data, preventing their information from falling into the wrong hands.
An advantage of this entity-centric approach is that it is about control physical freight flows instead of focusing on all kind of processes. By organizing information via a smarter way, there is now a one version “truth” about the status of the container, truck or ship. Organization can (un)-subscribe their self on a digital shadow, whereby the owner of the data can give this organization access or not. This stands in sharp contrast to the current information sharing mechanisms, many organizations manage the same order, but the data is redundant and inaccurate. Companies have the ability of sharing crucial information through the ecosystem. This information can be used for making better decisions in optimizing the supply chain. For example when it comes to truck utilization and CO2 reduction across organizational boundaries.

Within the Netherlands LSPs are looking for opportunities to collaborate with each other. The main reason for collaboration is achieving higher sustainability, increasing profit margins. For LSPs in practice this means reducing empty running, improving vehicle utilization, lowering handling time and costs, increasing on-time performance so there is less waiting time by the truck drivers. Figure 2 gives an overview of horizontal collaboration, where normally LSP A is executing his own trip, and the same for LSP B. By combining these trips less kilometers are necessary and the truck utilization will be increased.

LSPs gained functionality over the years, besides transporting orders. Both companies offering services such as warehousing (management), Vendor-Managed Inventory (VMI), outsourcing orders to charters. The core function of an LSP is planning orders in to trips and executing these trips at most efficient and effective way with respecting the quality of service.

We studied an effort of 2 LSPs to start a collaboration using control towers concepts. Due to the complexity and the organization having cold feet about starting horizontal collaboration, the companies started with setting up a foundation so information can be exchanged based on trust and commitment. In the design, there was one challenge prescribed by anti-trust regulation. In short it is not allowed to directly share information such as prices between competitors.

The first step towards horizontal collaboration the LSPs initiated was planning together transportation. The collaborative planning is a flexible and complex process. This is because orders are translated into a trip. One trip (can) consist multiple orders. When orders are executed at day X, the intake will take place X-1, X-2 or before that period. X-1 means execution day minus 1 day. Collaborative planning will have a big impact in the planning process of both organizations, because trips are set up in the warehouse for a truck driver one day before execution (X-1). During the day X-1, both companies need to upload their data to the platform. The planning-algorithm will plan the trips at X-1 and at twelve o’clock both organizations need to give an acknowledgement about the new trip planning. This requires a collaborative planning-platform. This platform will be able to import trips from both organizations; all the trips will incorporate into a new overall transportation plan. Where trips are shared among the organizations in becoming more efficient and effective.

We identified the following four main functionalities for this platform:
**Trip & equipment intake** - the system must be able to import trips and equipment such as trucks and drivers from both organizations. Each organization should provide the data to the system. It is not allowed due to cartel agreements to share data among each other. The system must be able to handle data in a secure and confidential manner.

**Input data checker** - all the data should be checked on completeness and errors. Otherwise, the planning-algorithm will use the wrong input.

**Collaborative trip planning** - the system must be able to produce a combined trip planning where all trips are assigned to a truck and driver.

**Sharing the collaborative trip planning** - the system must be able to export the new trip planning where each company receives its own trip planning. It is not allowed for both organizations to see the whole trip planning. In practice, organization A sees their own trip planning including their own trips and the new shared trips from organization B.

In addition to these functional requirements, several non-functional requirements apply:

**Extensibility** - the system must be extensible. Adding new features e.g. at this moment trips are shared among the partners via this system. In the future, it is desired to share orders instead of trips. Adding a new or improved planning-algorithm.

**Scalability** - the system should be scalable in the sense of multiple partners joining this platform. Or the planning-algorithm needs to handle big chunks of data.

**Security** - the system must be secure in a way that data is encrypted. Organizations that have access to the platform are not allowed to see each other’s data, except the exchanged trips.

---

Planning algorithms need certain parameters to plan. These parameters were designed by knowledge from the different planning departments. Figure 3 gives an overview of offering data to the platform and creating a planning. The outcome (planning) must be incorporated in the existing transport management systems (TMS) or advanced planning systems (APS) of both companies. Currently both companies use a different system. We have investigated how to return the planning back to the IT systems of both organizations. We found out that both systems were not able to import trips back into the system. The current functionality of the local systems is that can handle orders as input data and not a complete trip planning. Furthermore, the local systems were not able to import trucks and driver data, that is linked to a trip. Besides these challenges all the data needs to import manually. At this moment, there is not an interface for automatic import. In this cycle, all the participants learned that current IT systems lack of functionality to achieve the benefits of flexible horizontal integration.
**API based integration**

The main challenge of the new system/platform is the extensibility. This is due to the form of collaboration, where both companies have cold feet and a lack of knowledge about the new processes and required tools. Therefore the platform needs to provide functionality and information on an agile and flexible manner. This can be done via offering an API. Besides the functional requirements is enabling supply chain innovation only possible via the non-functional requirements.

At this point the platform should provide the following POST routines. Where POST stands for sending data to the horizontal collaboration platform.

**POST**

- `-CP1.0/vehicles_drivers_by_date` : available equipment
- `-CP1.0/locations` : location information, required by the organization who will execute the trip.
- `-CP1.0/trips_by_date` : send trips for a specific date. e.g. execution date X
- `-CP1.0/trips_acceptance` : give an acknowledgement, formal check - auditable

**GET**

- `-CP1.0/planning_trips_by_date` : trip planning included assigned equipment for execution date X.
- `-CP1.0/plannings_locations_by_date` : retrieve location information for execution date X, this is only required for the incoming trips of the other organization.

By setting up and API the participating LSPs were forced in thinking differently. We have seen that most people were mostly thinking in existing processes for transportation planning. Creating an API is all about thinking what information is important for the platform and the other way around. It was a big eye-opener that they had to think about services, and what information in setting up a collaborative transportation service.

Via an iterative approach and adding functionality. We have seen that this can help supply chain collaboration. Involved people where not aware of the complexity of combining two similar processes in to one overall process. Using an agile design approach, we have seen that this helps people in changing their behavior and be aware of how IT can help/support collaboration. Current and future supply chain will change and the business networks should become more agile. They should be able to add or remove existing functionality e.g. changing trips or orders among partners within an ecosystem.

It is all about connectivity via a flexible and secure manner and connecting organizations who are willing to collaborate. Connectivity is the main pre-condition for cooperation and collaboration between organizations. In practice this is very difficult. The discussion goes beyond lack of functionality of IT systems. For example, the interpretation of data is also key. Organizations that want to collaborate should be aware of how to present their data, e.g.:

**Interpretation of data:**

- What is an order? Size of the pallet - Chep or EURO, ETA is that when an order is at the location itself or when the order will be unloaded?

**Accessibility of data**

- How can the data be accessed?
- What will be the format of the data?
- Who is allowed to get access to the data?

**Organization of Data**

- Will the data be copied every time per process where the data is required?
- Or are we going to store the data and so that the process can use the data?

**Conclusions**

There is a large need for more coordination in supply chains and 4c are a promising concept. However, the current IT systems are not very suitable to enable 4c effectively. We know that often IT investments are lost, as
creating a 4c on top of current legacy systems and methods is very challenging. Key requirements for the ICT architecture are support for (1) Modularization of Services, Product, Process Services, (2) Coordination and Collaboration Capability (3) Quick connect capability (4) Relationship Management Capability (5) Risk Management Capability. In this chapter we focused on (2) and (3) by illustrating how APIs can be used for faster and more controlled integration.

Future IT systems or platforms can offer an API to achieve more flexibility extendibility. Using an API will help other organizations in implementing their information needs for their own process, instead of building hard to maintain point to point interfaces. We have seen that there is no silver bullet for collaborative planning. In this case we only discussed trip planning, however the system should also be able to plan orders. Or start to re-plan at day X for example. This will require more flexibility of the current IT infrastructure. The future supply chain processes will become more agile and by using an API this can be a first step in becoming more flexible

References


