

# SOA Based Integration for Creation of Collaborative Supply Chain in Automotive Domain

*By Krishnendu Kunti, Terance Dias and Ashwini Jeksani*

*Enable collaboration among supply chain  
partners using SOA-based integration platform*

**A**utomotive industry has one of the most complex supply chains primarily because of the complex nature of the end product, number of supply chain partners involved and the geographical dispersion of supply chain partners. A typical vehicle requires somewhere around 5,000 to 7,000 finished parts; these finished parts in turn consume nearly 130,000 subparts [1]. These parts are either manufactured in-house or sourced from multiple supply chain partners across the globe resulting in complex supply chains. Supply chain in automotive industry has matured over time and is still in a flux, driven by business needs.

We discuss the nature of supply chain in automotive industry, the present business drivers and resulting demands on supply chain integration. We also propose on how a SOA based integration platform is best suited to address these issues. Subsequently, we discuss the required functionalities to be supported

by the SOA based integration platform and highlight the major pain points addressed by such a platform.

## **EVOLUTION OF SUPPLY CHAIN AUTOMOTIVE INDUSTRY**

During the early 20th century the modern automotive industry came into existence. One of the events that brought automobiles within the reach of general consumers was the introduction of Ford T model [2]. The T Model was based on mass production of single base model with zero customization resulting in huge economies of scale. Every component required in Ford T model was manufactured in-house resulting in a simple supply chain. However this mode of manufacturing resulted in huge amount of safety stock for individual components. Also it did not allow desired customization to the base model and did not outsource better engineering and manufacturing skills from other organizations.

During the early 1970's Toyota introduced an alternative manufacturing model --- a model that brought together advantages of economies of scale and customization as per customers' demands. Toyota's manufacturing model was based on principles of just-in-time (JIT) manufacturing and keiretsu [3]. JIT is based on pull system that provides information on exactly what components and sub-assemblies need to be produced by means of paper based cards known as kanbans [4]. Keiretsu relies on collaboration among multiple business stakeholders towards achieving common business goals. When applied to automotive supply chain this results in outsourcing of some functions to other business

Automotive majors have realized that in order to survive they should strive to get the best of technology at the lowest of cost and in the fastest time to market possible. In order to achieve the target they need to collaborate with multiple supply chain partners across the globe.

### EMERGENCE OF COLLABORATIVE SUPPLY CHAINS

In a collaborative supply chain scenario, supply chains of stake holders are intricately interwoven. Supply chain partners not only collaborate towards exchanging data for final goods delivered (e.g., sending information on the items to be procured) but also share information

---

*Huge synergies can be gained by different partners in an intricately woven collaborative supply chain scenario*

---

entities that are better equipped to execute the function. For instance, an automotive design-house could design multiple variants of the same base model faster than an in-house design team. Extending JIT to supply chain partners resulted in even better inventory control and smother response to variations in demand. Using the Japanese manufacturing system, companies were able to meet varied customer demands, with lesser inventory in a shorter period of time. In recent years apart from reasons of technical expertise manufacturing has also been outsourced to other business entities because of price differentials arising out of location advantages.

leading to visibility across each other's supply chains. Virtually collaboration can happen at any level of business, for instance, OEM getting discounts for a medium scale supplier or dealers deciding on promotion in conjunction with OEM to push a particular make of a vehicle. The final realization is that business is driven by efficiency of the whole supply chain rather than efficiency of a single organization. Any gains made by one supply chain partner will be passed on to other members of the supply chain and vice versa. Essentially participants of a collaborative supply chain can be seen as a single distributed business entity that collaborates at multiple levels of business requirements. The success

towards creation of collaborative supply chain depends on the ability to share information across multiple supply chain partners with the least cost and effort. Given the number of supply chain partners in a collaborative supply chain and the fact that often a single entity takes part in more than one supply chain (e.g., single supplier supplying to more than one OEM) it becomes extremely difficult to meet the integration requirements in such scenarios.

### IT LANDSCAPE SUPPORTING SUPPLY CHAIN MANAGEMENT

Typically supply chain applications were either home grown or procured from supply chain vendors. These applications addressed specific supply chain functions such as demand planning, factory planning, production scheduling, transportation management etc. Within an enterprise if a supply chain suite is procured from a single vendor, the vendor, more often than not, provides an integration application to coordinate among its own suite of applications e.g., i2 operational data store [5], or uses EAI tools for integration. However, if within a supply chain, applications are procured from multiple vendors then there does not exist any standard means of integration between these applications and in such a scenario integration is done by batch mode data extracts or using EAI tools. Integration based on batch mode data extracts induces latencies inherent to batch processes and creates point to point connections which are hard to maintain. While EAI tools might provide real time integration they create vendor lock-in and are often expensive to implement. The other issue faced in such scenarios is standardization of exchanged data format. EDI [6], an existing approach for data interchange in the automotive industry, is often expensive both in terms of cost and effort to implement and hence small and

medium size vendors are reluctant to accept such technology. Also, EDI is not sufficient to capture different kind of data interchanges required for creation of collaborative supply chain.

Requirements to support different data format across multiple applications that communicate using different protocols and are deployed in disparate systems poses considerable challenge in integration of collaborative supply chain .

- Supporting multiple data formats (e.g., EDI, flat file, any business specific defined schema like STAR etc) both within enterprise and external to enterprise (supply chain partners)
- Exposing business interfaces locked in supply chain applications as services using standards based syntax (e.g., WSDL)
- Supporting multiple types of end points e.g., JMS, HTTP, .NET, EJB, RMI, databases etc
- Support for widely adopted standards for business document interchange e.g., RFC4130 for MIME-based HTTP EDIINT (aka. AS2) transfers (use for EDI) and STAR specification for Business Object Documents (XML) transfer using ebXML and web services
- Defining common business schema, that maps to different application specific schemas and required transformations
- Providing means to create process from services using workflow languages like BPEL etc
- Manage authentication and authorization across multiple business domains
- Integration of supply chains with least cost and minimal effort.

Often organizations handle integration of supply chain in piecemeal manner using custom code (e.g., ABAP for data extraction from SAP ERP to be used in planning applications) resulting in point to point connections that cannot be reused by other applications requiring the same data.

### SOA BASED PLATFORM FOR SCM COLLABORATION

A SOA based platform to manage supply chain integration should handle integration needs at multiple levels of enterprise IT viz., data access layer, application layer, integration layer and process layer.

based format (e.g., Xquery, OQL) and exposing them to standards based interface definition language (e.g., WSDL) or using API. Data services layer also provides value added features like caching, profile based data filtering and meta data management. Output from data access services can be translated and transmitted over required protocol to destination systems using SOA based integration backbone.

**Application level services:** Application level services are provisioned from application API's. However application level API can only be accessed on a particular platform. At this

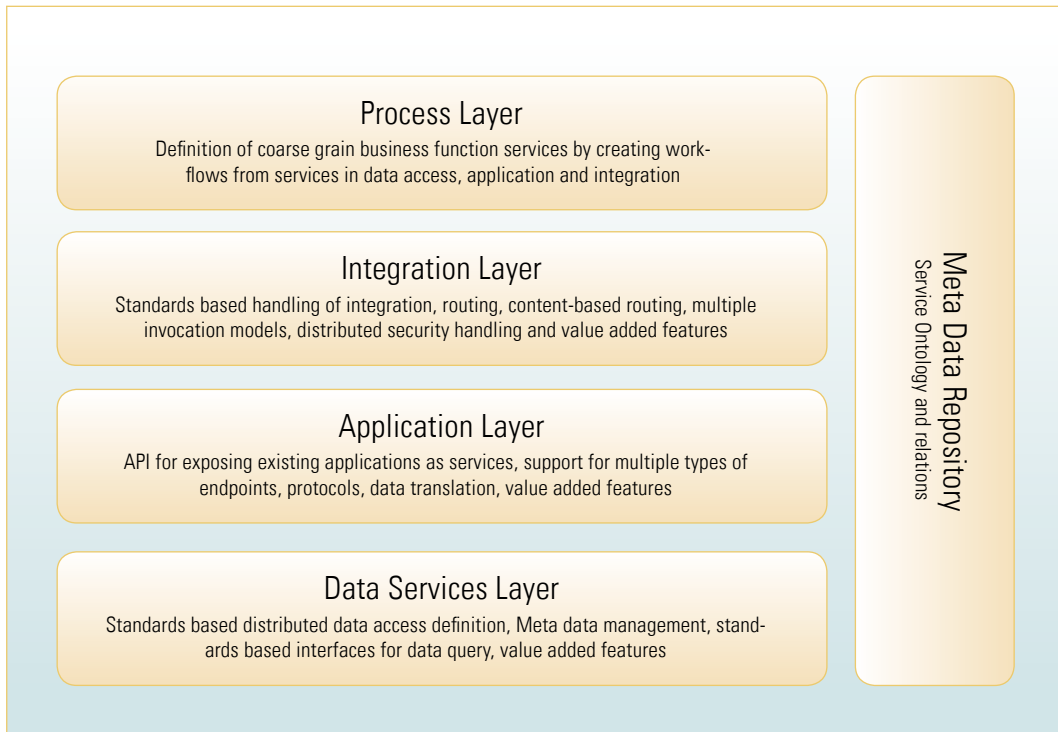
---

*A highly distributed architectural platform like SOA should be able to address supply chain integration at all levels of enterprise IT to achieve maximum SC collaboration*

---

**Data services layer:** In enterprise supply chain application integration, integrators are often faced with a scenario where in the absence of API for data access, point to point custom code needs to be written for data movement. For instance, if data needs to be moved from one or more manufacturing applications to one or more destination planning applications requiring different formats, writing point to point data extraction code can be time consuming and cost intensive and extremely hard to maintain in the long run. In such scenario shared data services[7] platform can be used for defining distributed queries in a standards

layer a SOA based integration platform provides tools for rapid generation of services from platform dependent API (JAVA, .NET, Legacy etc). Tools enable faster migration of application interfaces to SOA and allow incorporation of value added features like caching and security. These tools can also be used to auto generate service interfaces (e.g., web services artifacts), provide mapping between service schema and business data definition schema, provide value added features like caching, authentication, etc., and finally deploying generated artifacts. However it is important to note that an application level API can be directly consumed at integration layer as well. The



**Figure 1:** SOA Based Integration Platform for Supply Chain Integration

**Source:** Infosys Research

only differentiator being, if an application interface on itself qualifies to be exposed as reusable component rather than just a part of flow towards creation of reusable service, then it is better to expose the application interface as service.

**Integration level services:** Services at integration layer create coarser grained business functions from services at data access layer, application layer and native application API. Integration layer is used for transport bridging, content based routing, translation, provisioning of distributed security [8], supporting different invocation and communication model. For example, an engineering change might be required to be propagated to planning and inventory systems

— these systems might exist across multiple domains and accept data in different formats. ESB (Enterprise Service Bus) can be used to handle such scenarios with minimal effort [9]. Finally an integration layer acts as a gateway for external services and provides / receives data from external service in their supported formats over required transports.

**Process level services:** These services are coarse grained and are similar to integration layer services. They can be implemented at integration layer or using workflow language based orchestration layer. These services usually represent business document interchange interface that internally drive a number of steps in a process. For instance, submission of demand

order function can be exposed as process interface where a submitted demand is processed by a set of services at integration, application and data access layer using production planning, scheduling, manufacturing, purchasing and finance functions.

**Meta data repository:** In enterprises, IT systems often represent a single business entity differently based on different ontological classifications. For instance, a unique part identifier might be referred to as partid in one system and partno in another system. In SOA

individual services. Hence it is important to manage non functional requirements such as business activity monitoring, distributed security, service look up etc.

#### A BUSINESS USE CASE FOR SOA BASED COLLABORATION IN SCM

Let us consider a business scenario that involves introduction of a new vehicle for a particular segment of buyers viz., young professionals, elderly, householders etc. The process typically starts with a market survey across multiple geographies carried out by the OEM along with

---

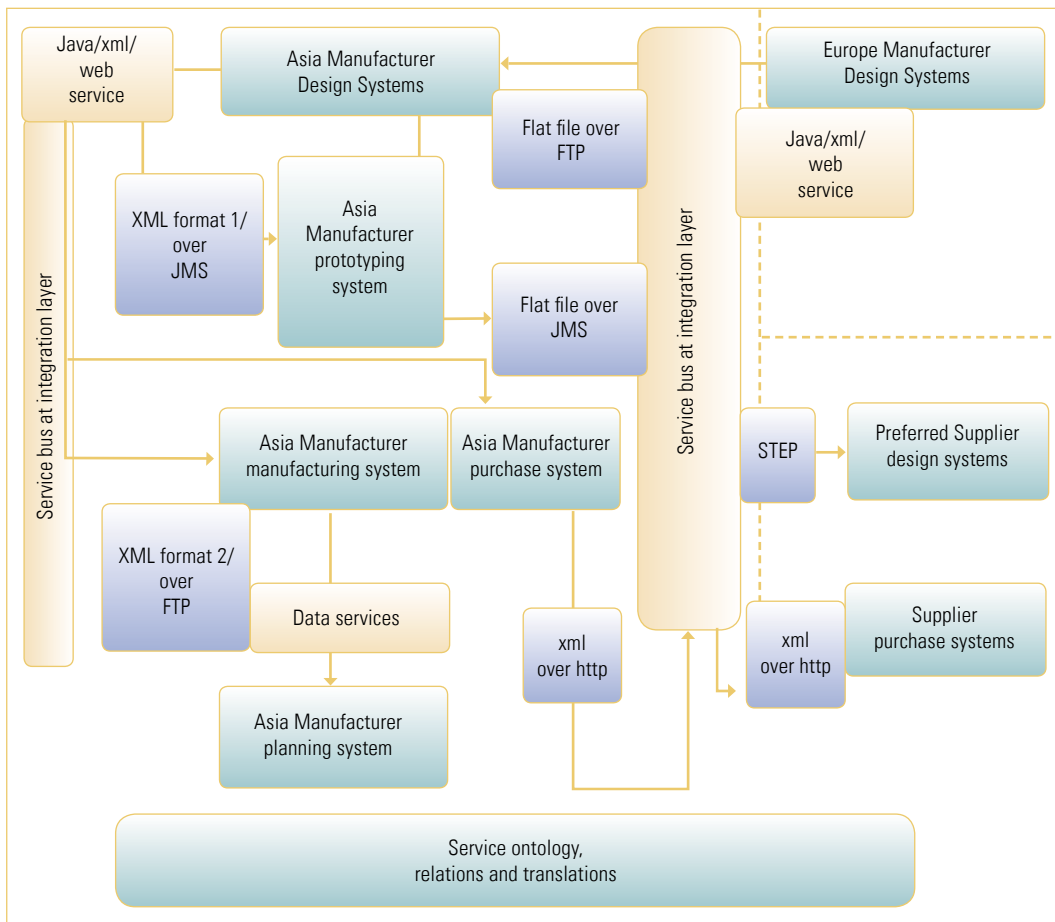
### *Maintenance of service ontology in a metadata repository can help negotiate confusion related to multiple naming of a single entity*

---

where a process is created from a number of services, managing disparities of data format pose a major challenge. This issue can be addressed by maintaining service ontology and semantic relations in a metadata repository. These ontology and semantic relations can be used at multiple levels of SOA. For example, service bus can use the semantic relations from provisioning of translation or shared data services platform can use these semantic relations for part lookup where a part identifier field is known by different names in different systems.

SOA is loosely coupled and highly distributed architectural paradigm. In SOA a process might span across multiple user domains and finally success of a process depends on

other stakeholders like dealers, advertisement agencies, etc. The outcome is refined into broad level features and pricing. The concept is then communicated to styling for creation of 3D, 2D and clay model. Once there is mutual agreement on the looks and features by stakeholders — like engineering, marketing, purchase, senior management, the concept is said to be finalized. A new vehicle is either designed from scratch or created by reusing existing platform and components with some modifications. The design and engineering process necessitates availability of systems that allow creation and sharing of documents, CAD integration, versioning of product design information, work flow management, role-based access and API for sharing/ accessing data with other applications.



**Figure 2:** Typical Integration Scenario in Supply Chain Collaboration

**Source:** Infosys Research

Sharing information with other systems is one of the key requirements for design and engineering systems. For example, an OEM decides to introduce a variant of an existing platform of one of its subsidiaries in Europe to another subsidiary in Asia after incorporating some customizations. In such a scenario the Asian subsidiary of the OEM will need access to design systems of the European subsidiary. In addition the design systems of the Asian subsidiary will also need to share information with production planning,

purchase, finance as well as preferred suppliers design systems.

The various entities involved in this scenario are shown in Figure 2 which depicts a typical integration challenge in supply chain collaboration and the way it can be addressed using an SOA based integration platform. In most scenarios, the design systems of a manufacturer in different geographies will be different. In the stated scenario data needs to be moved from design system in Europe to a design system in Asia.

Design system in Europe provides application interfaces for data extraction as XML. However, data needs to be provisioned as a flat file over FTP to the design system in Asia. In order to transfer data between concerned applications, individual pieces of code can be written that invoke API in Europe design system that obtains data as XML, formats it to a flat file and saves it in a required location in Asia subsidiary's system. Such point to point connection based approach will require application to invoke the native API (e.g., JAVA). However, any other application that desires to get the same data will need to replicate the same function.

The solution lies in exposing native API as services (e.g., web services using SOAP over HTTP) that adhere to enterprise level data formats. Once created, these services can be used by any other applications. In the absence of required API, data services can be directly generated from the concerned data sources.

Data services are generated by definition of standards-based query in a centralized data services platform and exposed either using standards based interfaces like web services or some API. The retrieved data is translated to the required flat file format — where the source and destination ontology and required translations are drawn from a metadata repository — and transmitted to a physical location using enterprise service bus (service at integration layer). This approach allows standards-based definition of service interface that exchanges data in mutually understood format and introduces a mediation layer that handles translation, routing and transport bridging (HTTP to FTP, in this case).

In certain scenarios there might be a need for designers at Asian subsidiary to

lookup for a particular part in the European OEM's design systems. However, due to different naming conventions it might be difficult to locate the exact required part. In such scenarios, shared data services platform can use semantic definitions in metadata repository to perform lookup for parts that follow different naming conventions.

The design system at Asian OEM needs to share data with other systems like manufacturing systems, prototyping systems and purchasing systems, all of which require different data formats over multiple protocols. Also there is a need to share design data with preferred supplier design system using STEP (ISO 10303) for PLM (Standard for the Exchange of Product Model Data) [10].

A service bus can be used for achieving the desired integration and for provisioning of federated security while exchanging data with preferred suppliers' systems located across different domains. The planning system at Asian OEM requires data from one or more manufacturing systems. However, these manufacturing systems do not provide any API for retrieval of such data. Traditional means of achieving the result would be by writing custom JDBC or application API centric code (for e.g., ABAP in case of SAP). However, the resulting point to point connections are hard to maintain and are replicated for every new application.

The solution lies in definition of distributed queries in standards-based manner in SDS platform and exposing these queries as services to be re-used. SDS platform takes care of distributed connection management, transaction management, caching and tool based generation of service interfaces.

Finally the submission of purchase order from Asian manufacturers' purchase

system to suppliers' purchase system can be seen as process level service that internally drives manufacturing process at suppliers end and provides asynchronous callback handlers to service consumers for checking status of purchase order.

## CONCLUSION


Collaboration in auto supply chain involves participation of multiple stakeholders across the supply chain towards sharing information at different levels of supply chain management activities. Traditional means of point to point application integration falls short of achieving the objective of collaboration and is not suited for agile business processes. SOA provides a new approach to address integration needs in auto supply chain management towards enabling collaboration.

An SOA based integration platform allows standards based integration of multiple types of interfaces that communicate using different data formats over different protocols. Hence such a platform is ideally suited to handle fluid integration requirements in a collaborative supply chain.

## REFERENCES

1. Laurie Sullivan, Auto-Parts Makers Open Their Networks, Information Week, Available at [http://www.agile.com/news/2004/infoweek\\_041904.pdf](http://www.agile.com/news/2004/infoweek_041904.pdf).
2. Ford T Model. [http://en.wikipedia.org/wiki/Ford\\_Model\\_T](http://en.wikipedia.org/wiki/Ford_Model_T).
3. Peter Wad, Workers in the Supply Chain of the Global Automobile Industry. <http://www.amrc.org.hk/5404.htm>.
4. Kanban, University of Cambridge. <http://www.ifm.eng.cam.ac.uk/dstools/process/kanban.html>.
5. I2Operationaldatastore.I2technologies LTD. [http://www.i2.com/assets/pdf/PDS\\_ods\\_v61\\_pds7262\\_091304.pdf](http://www.i2.com/assets/pdf/PDS_ods_v61_pds7262_091304.pdf).
6. John Leslie King and Kalle Lyytinen, Automotive Informatics: Information Technology and Enterprise Transformation in the Automobile Industry. <http://www.si.umich.edu/~jllking/Auto-MIT-Final.pdf>.
7. V Niranjana et.al., Shared Data Services: An Architectural Approach. <http://portal.acm.org/citation.cfm?id=1090954.1092078>.
8. Bijoy Majumdar et.al., SOA Security Modelv2.0. [http://whatis.techtarget.com/tip/0,289483,sid92\\_gci1168738,00.html](http://whatis.techtarget.com/tip/0,289483,sid92_gci1168738,00.html).
9. Enterprise service bus, <http://www-306.ibm.com/software/info1/websphere/index.jsp?tab=landings/esbbenefits>.
10. STEP (ISO 10303). <http://www.steptools.com/library/standard/>.
11. The American Automotive Industry Supply Chain- In the Throes of a Rattling Revolution. <http://www.ita.doc.gov/td/auto/domestic/SupplyChain.pdf>.
12. Luis Araujo, Modularity, Systems Integration and Supply Chain Leadership. <http://www.ht2.org/conference/pdf/SYM1.pdf>.
13. Ricardo UL Ltd, Skills4Auto Ltd. Vision of UK Automotive industry in 2020. [www.ricardo.com/media/medialibrary/Vision\\_for\\_the\\_UK\\_Automotive\\_Industry\\_in\\_2020.pdf](http://www.ricardo.com/media/medialibrary/Vision_for_the_UK_Automotive_Industry_in_2020.pdf).
14. Where is the auto industry headed? <http://www.deloitte.com/dtt/article/0,1002,cid%253D120316,00.html>.
15. Andrew Cummins, Supply Chain vs.

Supply Chain - automotive industry  
distribution - Brief Article. <http://>

[www.findarticles.com/p/articles/  
mi\\_m3012/is\\_6\\_181/ai\\_76855607](http://www.findarticles.com/p/articles/mi_m3012/is_6_181/ai_76855607). 

---

## *Author profile*

---

### **KRISHNENDU KUNTI**

Krishnendu Kunti is a Senior Technical Specialist with the Web Services CoE, SETLabs, Infosys. He can be reached at [krishnendu\\_kunti@infosys.com](mailto:krishnendu_kunti@infosys.com).

### **TERANCE DIAS**

Terance Dias is a technical specialist with the Web Services/SOA Center of Excellence in SETLabs, Infosys Technologies Limited. He can be reached at [terance\\_dias@infosys.com](mailto:terance_dias@infosys.com).

### **ASHWINI KUMAR JEKSANI**

Ashwini Kumar Jeksani is a Software Engineer with the Web Services/SOA Center of Excellence in SETLabs, Infosys Technologies Limited. He can be reached at [Ashwini\\_Jeksani@infosys.com](mailto:Ashwini_Jeksani@infosys.com).

*For information on obtaining additional copies, reprinting or translating articles, and all other correspondence, please contact:*

Telephone : 91-20-39167531

Email: [SetlabsBriefings@infosys.com](mailto:SetlabsBriefings@infosys.com)

© SETLabs 2007, Infosys Technologies Limited.

Infosys acknowledges the proprietary rights of the trademarks and product names of the other companies mentioned in this issue of SETLabs Briefings. The information provided in this document is intended for the sole use of the recipient and for educational purposes only. Infosys makes no express or implied warranties relating to the information contained in this document or to any derived results obtained by the recipient from the use of the information in the document. Infosys further does not guarantee the sequence, timeliness, accuracy or completeness of the information and will not be liable in any way to the recipient for any delays, inaccuracies, errors in, or omissions of, any of the information or in the transmission thereof, or for any damages arising there from. Opinions and forecasts constitute our judgment at the time of release and are subject to change without notice. This document does not contain information provided to us in confidence by our clients.

**Infosys**<sup>®</sup>

POWERED BY INTELLECT  
DRIVEN BY VALUES