SUPPLY CHAIN NETWORKS REVEALED

SUPPLY CHAIN NETWORKS

Executive Summary

No matter your industry, the very nature of competition has changed. The break-through business strategy that has emerged today is the networked enterprise.

In this new paradigm, competitive advantage has as much to do with your network of trading partners and the technology platform your enterprise and ecosystems, as the actual product or service you offer the market. And we know, for supply chain professionals, solving the multi-party challenge has always been the essence of supply chain.

These issues are driving the high level of interest in supply chain networks. Yet these networks are not well understood, in spite of the current hype, which has not provided the right framework for a clear understanding of what supply chain networks are, what they can provide, how they work, and what the different offerings in the market are differentiated.

In this report series, *Supply Chain Networks Revealed*, we provide some answers. This executive summary focuses on the key issues required to understand and evaluate a supply chain network.

ChainLink Research 2019

Supply Chain Networks Revealed

This report series will provide a differentiated discussion on supply chain application networks from a process, functional and technical perspective.

This series consists of three parts:

- **Executive Summary**: A strategic overview of Supply Chain Application Networks and why they are an essential component of competitive advantage, along with a summary of key points from the report.
- **Part One: What They Do**: What is a supply chain network? Why is it the platform for the inter-enterprise supply chain process and functional perspective?
- **Part Two: How They Work**: Understanding the supply chain application network technology and examples of different types of network offerings in the technology market.

Table of Contents: Supply Chain Networks Revealed

What's Your Strategy?

For this decade of the 21st century and beyond, the winning business model is the networked enterprise. Supply chain has emerged as the key enabler of networked enterprises, where the imperative is designing and managing federated partner relationships with dynamic and fluid connections with customers. The technology platforms that enable them is the connective tissue of these networks. Cases in point:

- Product companies today are rapidly becoming *smart connected enterprises* that provide product-as-a-service via a Performance-Based Logistics/Power-by-the-Hour model. They leverage their smart connected products with a continuous stream of information to sustain high performance.
- Manufacturers who become "network enterprises" can and have changed their business models, becoming information companies—beyond just providing service or hardware.
- Retailers often become resellers, recrafting the store-within-a-store for the digital age, providing infinite catalogs and direct-to-consumer delivery anywhere in the world.
- The rate of mergers has steeply accelerated as companies leverage their physical, service and web presence to provide global end-to-end service to their customers.
- Customers engage with multiple sellers and do business, not just based on competitive price, but based on *availability and service/supply chain performance.*
- To accelerate growth and performance, networked enterprises address their talent shortages using effective partnering strategies and support remote workers with advanced technologies.

For supply chain executives, transitioning to this new model requires implementing a supply chain trading partner operating model that supports the networked enterprise, providing interoperability and visibility with reduced complexity. The supply chain trading partner operating model is a codification and automation of the key processes, policies, performance metrics and technology that govern your supply chain partner relationships.[1]

21st Century Business Model

- Enterprise strategy: *Networked Enterprise*
- Supply chain strategy: *Supply Chain Trading Partner Operating Model*
- Information strategy: *Supply Chain Network*

To meet the instant access/instant response cadence of today's sales and fulfillment lifecycle, companies need extremely granular data from all of the demand and supply points across their supply chain, as well as the ability to rapidly and accurately analyze that data. Customers expect not only accurate price, availability, and reliable delivery promises (down to the hour) while they are shopping, but often they also demand visibility to source markets and accurate tracking throughout the whole process.

[1] Examples included compliance and data standards (types of technology expected and formats), contracts (pricing, volumes, guarantees and so on) and agreements

This is BIG data. This is connected. This is 24/7. And this is extremely smart. This requires moving beyond incremental thinking. This demands transformative, ever-fresh business models, with broader and bolder approaches. And all of this requires digital connective tissue. Often when we hear about digital supply chains, the advice given is to automate more processes. Automation as a goal is insufficient if it introduces more layers between you and your objective. So we need to move beyond enterprise-centric approaches that introduce more layers in the supply chain. A networked enterprise with a networked supply chain requires reducing the layers of complexity, providing more clarity, and the cutting away of the labyrinth and the time required for multiple stovepipe applications and enterprises.

As the phrase 'form follows function' implies, the technology we deploy must follow and support these needs. We need to be closer to our partners and customers, with more visibility, more sharing, and common goals. This is why supply chain executives are turning to supply chain network providers for solutions. But as we know, the tech market is notorious for buzzwords that often don't really describe accurately and assist the buyer in understanding and choosing the appropriate solutions.

Understanding Supply Chain Networks—What to Look For

To fulfill these goals, the supply chain network solution should include all elements needed to achieve an intelligent real-time supply chain. Integration or visibility platforms, without applications, are frequently marketed as 'supply chain networks' by themselves.[2] But, a supply chain network needs both the integration/visibility capabilities and multi-enterprise supply chain applications, together on one process-wide platform, a supply chain application network. In the marketplace there are two basic types of supply chain application network[3] offerings. We call them Integrator Networks (IN) and Real-time Single Version of the Truth Networks (RSN).

Definition: Supply Chain Network

A multi-party network for trading partners, which is a single-platform, single-instance of supply chain applications, data, and services.

Members can share data, processes, and applications for the purpose of achieving their mutual and their enterprise business goals.

[2] Examples are business to business (B2B) and application integration offerings such as Informatica, OpenText, IBM's B2B Integrator, TIBCO, SPS Commerce, etc.

[3] You may also hear more generic terms such as *supply chain business networks*. However, the definition for those offered up in the media includes B2B/EDI integration solutions. *B2B is a feature* needed by supply chains, but is insufficient by itself to support the physical *supply chain network*. Another term you may hear is *supply chain commerce networks* which includes transactional as well as integration capabilities.

Integrator Network—These are networks that have been built up over time through acquisitions. Thus within their offerings are a set of separate networks that are integrated between applications and users/tenants. A common integration platform is provided. Each tenant has their own database[4] (in the cloud) supported by common master data.[5]

Real-time SVoT (Single Version of the Truth) Network—These are networks that have generally been organically developed with a single database and single processing engine. This is a many-to-many approach in which all tenants share the same ledger-like,[6] multi-enterprise data store—as well as sharing the same multi-party process execution.[7] Figure One shows an overview comparison between these two supply chain application network approaches.

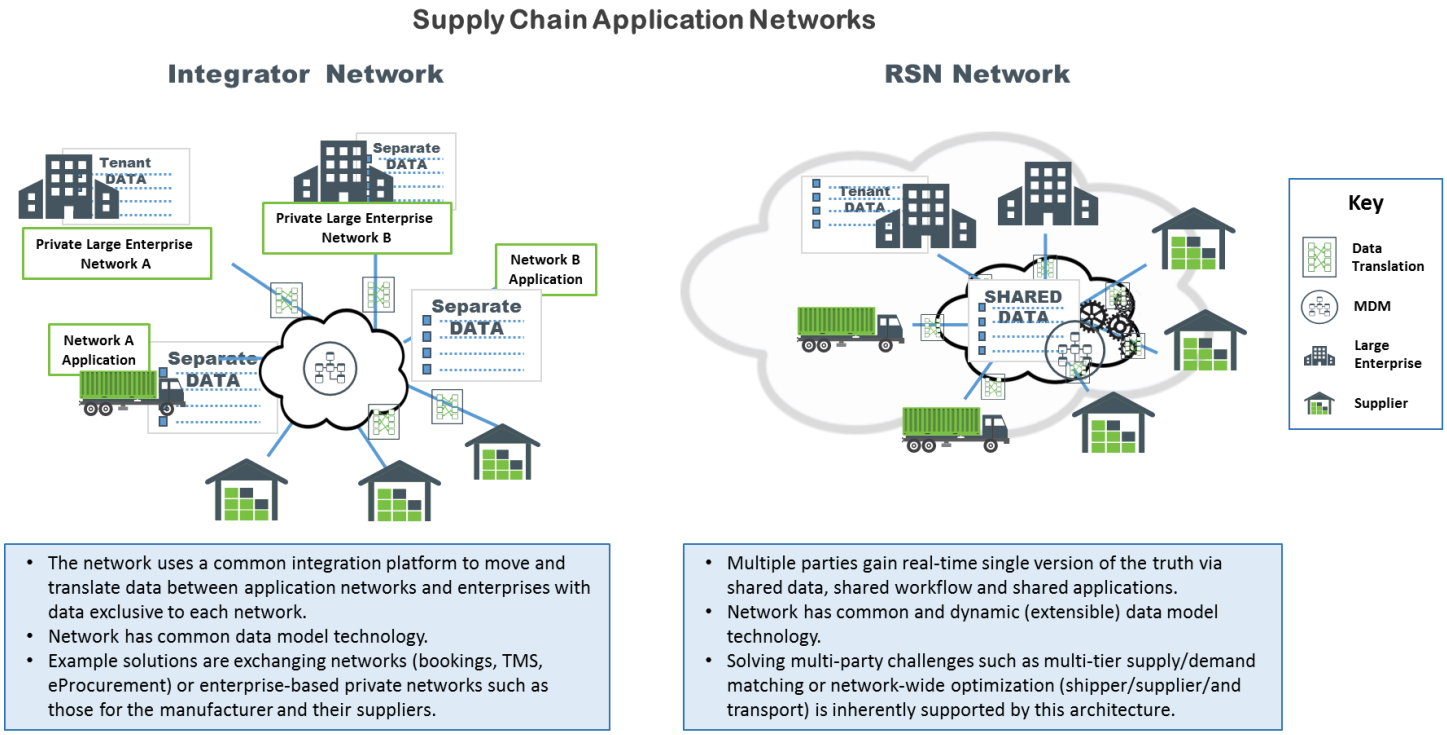


Figure One: Comparison of Application Networks

A central goal for supply chain application network providers is to provide an accurate and timely picture of the supply chain—a real-time single version of the truth and the ability to act upon it. So many terms are babbled about that sometimes their meaning gets lost. With some simple clarifications, we can better understand these terms and therefore what is required to run a multi-party supply chain.

[4] Generally, these are called *hubs.* When supporting entities (partners such as suppliers or transport) connect, they are called *spokes.*
[5] Examples are Descartes and E2open
[6] Not to be confused with Blockchain
[7] Examples are Infor Nexus and One Network

Single version of the truth (SVoT):

- ✓ A single *shared* data base where all the participants can see the transaction *in progress.*
- ✓ A single version of the truth can be multi-functional/network-wide as in the case of the RSN or per application in the case of the IN.
- ✓ Granular inclusive data—Single version of the truth also means supply chain-wide scope and granularity—at all the demand and supply points. This allows users to 'see' and include all the relevant data points needed to plan and execute accurately, again and again. Gone are the days of multi-layered endless steps from general to specific planning approaches. We can't fulfil orders end-to-end unless we know all the elements that need to be included in each and every shipment, no matter how large or small.
- ✓ Scale—Even for a small company there can be an overwhelming amount of data. Systems need the scale including all that granular SVoT data—forecasted, in progress, and history.
- ✓ Extensible shared master data management—Data across the supply chain is industry specific, always changing, and often unique to a particular enterprise or trading partner. In order to trade and support customers and partners, data has to be commonly understood and usable by all involved parties. The platform needs the ability to respond to the dynamism of continuously evolving data between multiple parties.
- ✓ Modern supply chain data—By this we mean social, geospatial, location-based data, unstructured data from the web and devices (conveyances, goods in motion, equipment) and mobile connectivity and data from customers, employees, and service providers. All these are included in today's supply chain analysis and execution.

Think of this:

A mid-size manufacturer can have tens or hundreds of thousands of supply and processing points.

A mid-size retailer can have hundreds of thousands of demand points—SKUs at each location.

In a supply chain, then, just these two entities can have millions of demand/supply points. That is incalculable for the average user to understand or legacy system to manage!

Real-time:

- ✓ Real-time means continuous refresh of information from data sensing, planning, through execution, *as events happen.*
- ✓ The ability to immediately *act* upon new circumstances. The always-on, 24/7 expectations do not leave time for elaborate orchestration or planning meetings. Systems must have the *applications and the intelligence* to alert users to critical issues in a timely manner and then to make recommendations and/or bring together all the necessary key data so that the best decisions can be made and executed on time.
- ✓ Real-time also means the system needs to support the dynamism of constant data, compliance, standards, and regulatory changes.

Why we need Real-time Single Version of the Truth to span planning through execution

Often supply chain decisions, such as those made in Sales and Operations Planning (S&OP), never make their way into the actual execution. And conversely, all the important lessons from successful or ineffective execution are not fed back and captured in the planning models.

Technology Prerequisites

High-level technology capabilities[8]:

- ✓ *Native/In-network Applications*—Often called *network services*. Rather than pulling together separate runs from individual application networks or off-network systems, the platform runs the relevant applications (such as demand, supply and logistics) *in* the network, using the shared SVoT data. In this way, achievable decisions that are optimized across a multi-enterprise supply chain are derived.[9]
- ✓ *Extensible Master Data Management*—The ability to support continuous, dynamic updating, appending data models and definitions (syntax and semantics) which include the rich, high-fidelity, modern supply chain data.
- ✓ *B2B Integration*—Supports the required data movement, standards, and translations between trading partners, within an industry or eco-system.
- ✓ *Interoperability*—Process synchronization between partners, through data and process inheritance, workflow management, and shared data
- ✓ *Elastic Cloud Services*—The ability to scale resources up or down as needed. Data and processing loads often change dramatically from season to season or for short term surge. Services should support rapid growth, increasing granularity, and volumes of data.
- ✓ *Machine Learning*—With the enormous scale of data and the dynamics of rapid change, we cannot expect users to see and understand every pattern buried in the sea of data. Machine learning can monitor, identify/derive patterns, and learn from the oceans of data, creating data-validated insights and make recommendations for action.
- ✓ *Intelligent Agents*—These programs take action based on explicit requests, rules-driven triggers, or machine learning-based pattern recognition. As organizations gain knowledge and experience with machine learning, intelligent agents can operate more autonomously.
- ✓ *End-to-end Process Management*—Supports the whole 'work' cycle of the supply chain user's tasks: visibility, planning, decision making, execution, analysis, and dynamic report creation.
- ✓ *Control Towers*—These monitor and control an enterprise's supply chain domain, built using both in-network and off-network data. These towers should not be merely visibility systems. Control means the ability to predict, alert, decide, and execute. Advanced control towers have intelligent agents to provide autonomous execution.
- ✓ *Trading Partner Services*—Methods for customers and partners who may be infrequent users and/or who have limited technology capabilities[10] to very easily get onboarded, communicate, and participate in planning and transaction management.

All these create more inclusion which allows multi-enterprise optimization—to discover cross-function/network-wide solutions that are feasible and optimized for all the parties involved, rather than optimizing just for the big anchor tenant and pushing the burden to smaller suppliers.

[8] For further definitions of these terms and additional essential features, please read *Part Two—How They Do It*

[9] See *Part One—What They Do* for supply chain process examples

[10] Often system services are offered at low or no charge for smaller partners.

Conclusions

Improving overall performance across the whole chain is the *quintessence*[11] of network objectives today. Thus our technology strategy and platform should be one that enables new levels of performance—growth, with profit, for all the partners.

We know our physical network includes all the elements that support fulfillment—seen and unseen. It's not just our partners, but also the environment and events happening around us that have an impact on the flow and integrity of product. Thus, we have to turn to networks.

By creating a network-wide digital network, we can gain exploitable details—not just estimated time of arrival, but the actual time; not just which ocean carrier, but which container at each stage as it moves from port to over the road.[12] We may need sensor data, such as temperature, vibration, and directionality, to ensure our product integrity. We want to see clearly into the selling channel—our product in distribution, at the store, and with the customer.

And we want to act on it. That is network-wide optimization. It goes beyond changing local parameters, such as resetting a safety stock level. It involves mutual-optimization across multiple functions and multiple enterprises. Ultimately, solutions that optimize across functions and enterprises will produce superior results across the chain.

All this data enriches us, makes us smarter, and enables us to be more competitive. This kind of real-time data and intelligence opens up greater possibilities for all parties involved. Now is the time to get closer to our customers and partners. They need us—and we need them—now more than ever.

[11] Definition: typical example of a quality; exemplar; stereotype; epitome; paragon; picture; prototype.

[12] Users have been tackling the consolidation and reconciliation of multiple data sources—carrier, 3PL, freight forwarder, an onboard GPS or RFID device, and/or one of many new ocean tracing or container tracking companies. Sadly, often this data does not agree and users are left to develop additional methods to try and accurately extrapolate the current reality on the ground … or on the water or in the air.

SUPPLY CHAIN NETWORKS REVEALED

SUPPLY CHAIN NETWORKS—Part One

What They Do.

Solving the multi-party challenge is the essence of supply chain.

Think of this: much of our operations are beyond our immediate control, and *no one organization is responsible for the entire workflow*. Along with this, we are drowning in a world of disparate data, with highly complex workflows and relationships.

Paradigms for how we operate and use technology are rapidly changing. We need to understand and leverage them now.

Those are the issues central to the growing wave of interest in supply chain networks. Yet networks—what they can provide and how they are different from traditional systems—are little understood, in spite of the current proliferation of articles and reports about them. Understanding these differences is fundamental when evaluating technology, if we are to make good choices.

This report series will provide a differentiated discussion on supply chain networks from a functional and technical perspective.

ChainLink Research 2019

Supply Chain Networks Revealed

This report series will provide a differentiated discussion on supply chain application networks from a process, functional, and technical perspective.

This series consists of three parts:

- **Executive Summary**: A strategic overview of Supply Chain Application Networks and why they are an essential component of competitive advantage, along with a summary of key points from the report.
- **Part One: What They Do**: What is a supply chain network? Why is it the platform for the inter-enterprise supply chain process and functional perspective.
- **Part Two: How They Work**: Understanding the supply chain application network technology and examples of different types of network offerings in the technology market.

Table of Contents

Introduction

It's All About the Architecture!

There is an architecture for each epoch. In the 70s and 80s, there was limited computing power and no internet;[1] EDI was used to transmit data point-to-point—**one-to-one**. Within the enterprise, departmental system modules[2] were strung together like a vast spaghetti of custom-code libraries. Things moved pretty slowly. System maintenance was a bear!

Today, we are living in a world of complex relationships with billions of connected nodes: organizations, processes, people, places, and *things*. This is a world of **many-to-many**. Our customers and partners require *always-on, instant response*, and with that expectation of instant responsiveness, the lines have been blurred between planning and execution. As we share processes, information, and goals with our partners, we need a technology architecture that supports these complexities, this need for *speed* and the supply chain's collaborative requirements. That architecture is a *shared operating*[3] platform to enable supply chain partnerships.

Supply Chain Networks: A Definition

When we say *supply-chain network,* what are we really talking about? (See definition in the side box.) EDI, visibility platforms, and commerce sites, though they may be *used* by supply chain operators, are merely *features* of a supply chain network. They do not constitute a supply chain network in and of themselves, as they provide only one function. The supply chain you manage every day includes a lot more than one capability.

There are significant differences in the market among supply chain networks, which are important to understand if we are to make good choices. In previous reports, we covered the breadth of networks. However, in this report, we focus on networks that support processes—that is, the *applications* that run our supply chain.

Definition:
Supply Chain Network

A multi-party network for trading partners, which is a single-platform, single-instance of supply chain applications, data, and services.

Members can share data, processes, and applications for the purpose of achieving mutual and private enterprise business goals.

We will explore and explain the different approaches to supply chain networks from a process and technology perspective, providing some examples that help differentiate and clarify how the technology enables specific outcomes. We will then paint a picture of the future, which is rapidly becoming part of our present.[4]

---

[1] Though the internet started in the 70s, it was used almost exclusively by government and academia until the 90s when businesses and consumers started connecting and using it. Before that, businesses used point-to-point, teletype, and dial-up, which replaced the physical delivery of magnetic tapes, punch cards, paper, or snail mail.

[2] which have grown into ERP and Supply Chain suites

[3] By operating platform, we mean a platform on which to do work at any level in the enterprise.

[4] NOTE TO READERS: In this report, we have attempted to *not* use deep technology terminology wherever possible, so supply chain professionals with only some technology understanding will be able to read this.

Solving the Multi-Party Challenge

There are various ways in which solution providers attempt to address the multi-party challenge. We present a quick overview of the basic approaches offered in the supply-chain market today.

Enterprise-centric Applications and Suites—today, these may be offered in the cloud or on premise, but fundamentally, these are not networks. They try to solve supply chain visibility problems by layering a control tower on top of an enterprise instance, using B2B messaging to collect supply-chain data. In the enterprise-centric world, though, we are forced to go it alone a lot, connecting and reconnecting to a variety of systems and trading partners with varying protocols. This can lead to a very unsatisfying scenario[5] with challenges in synchronizing partner data.

B2B Networks—these are the EDI, AS2, and other vehicles for B2B communication (API, workflows, and web services tools that are internet resident). B2B networks are all about *moving* data.

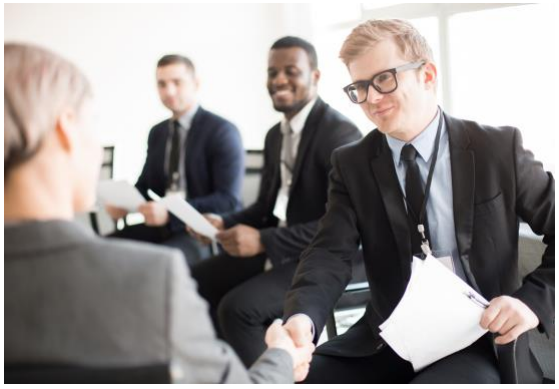
Integrator Network—these are *application* networks that are based on a messaging paradigm. The Integrator Network (IN), as distinguished from a B2B network, is an advanced approach whereby there are shared business applications and in-cloud business data. Examples are eProcurement, Transportation Management, ecommerce and so on. These web-resident solutions provide a significantly more consistent way for trading partners to work their processes. Each tenant will have their own database[6] (in the cloud or on premise) supported by common master data (MDM) and/or canonical data models.

Networks

Integrator Networks are based on messaging—a send/receive paradigm, to the network and within the network between each enterprise data store. Historically, these attempted to synchronize various partner data.

Real-time SVoT Networks are based on publish/subscribe to a ledger-like data store paradigm where all partners publish to the same database. This provides a single source of data for all to share.

Real-time SVoT Network—the real-time application network (RSN) is based on a single database/single processing engine (single version of the truth, SVoT). This is a many-to-many approach in which all tenants share the same data store—ledger-like,[7] as well as sharing the same multi-party process execution. Partners can create/share a *network-wide workflow*, and all the particulars for that transaction are visible (given permission).[8] For example, multiple locations of inventory, multi-mode location of conveyances, changing data, changing conditions, and so on across multiple legs or multiple parties are visible. Figure One shows an overview comparison between these two *supply chain application network* approaches.



[5] For more on supply chain network evolution and approaches, you can read *Multi-party Solutions for Supply Chain*.
[6] Generally, these are called *hubs.* When supporting entities (partners such as suppliers or transport) connect, they are called *spokes.*
[7] Not to be confused with Blockchain
[8] When we talk about shared data, visibility and so on, in all cases, it is based on the granting of permission by the relevant parties. We will discuss the concept of *permissions* in Part Two.

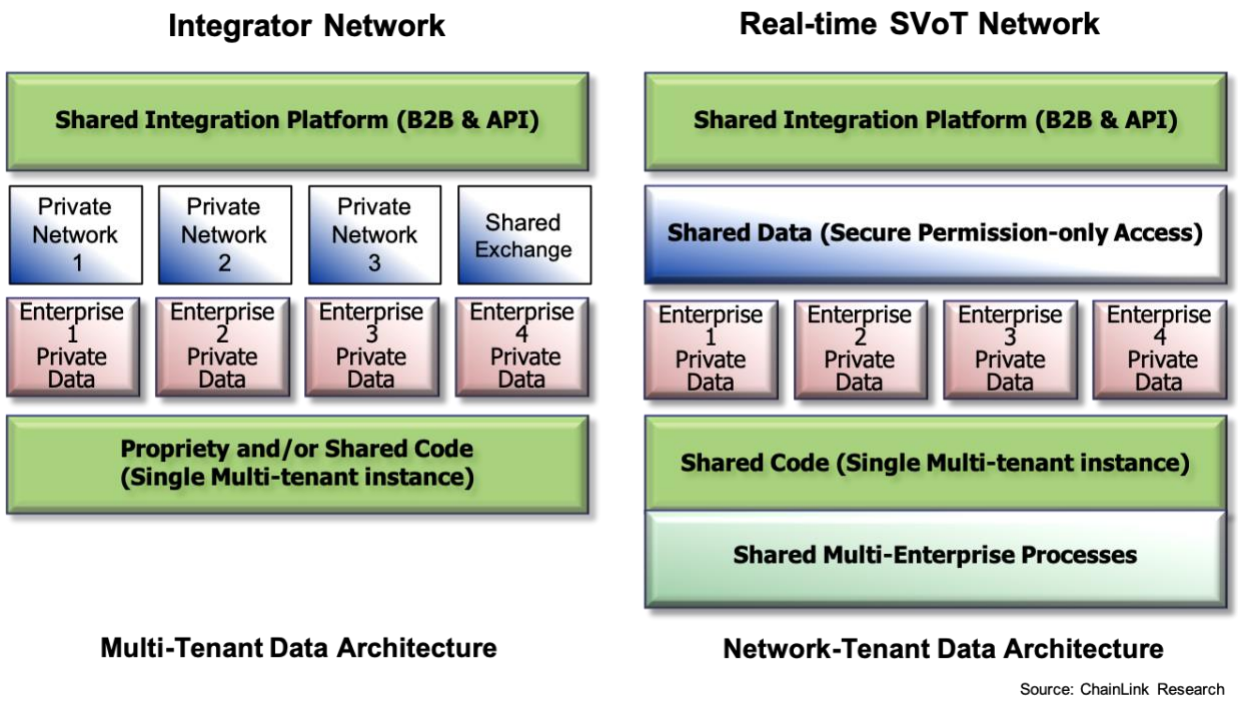


Figure One: Comparison of IN and RSN Networks

As we continue through the report, we will clarify the terms used in this figure to more deeply define the various attributes. Going forward, this report focuses on the IN and RSN supply chain application networks.

Now we will look at why supply chain application networks are important and examine some simple examples of how they support advanced supply chain objectives.

What Do Supply Chain Application Networks Do? What Approach Do I Need?

As mentioned above, there are some major differences in how different supply chain application networks work. Hence, the question becomes, which option applies to my circumstances? Here we will review specific processes.

Building Our Network

We will start with some basic supply chain tasks and build the transactions as we go. Figure Two and Figure Three highlight an end-to-end process of demand through fulfillment.

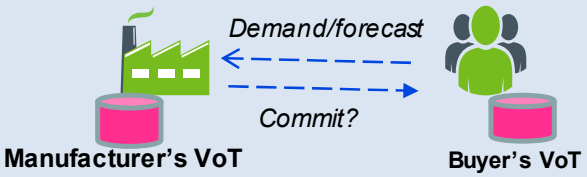
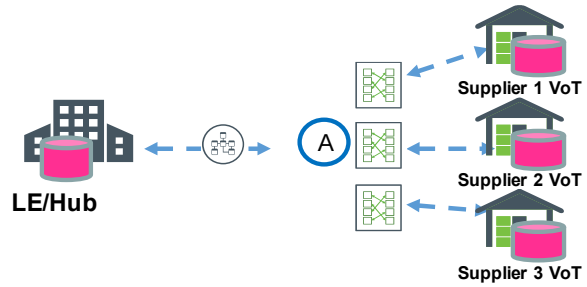
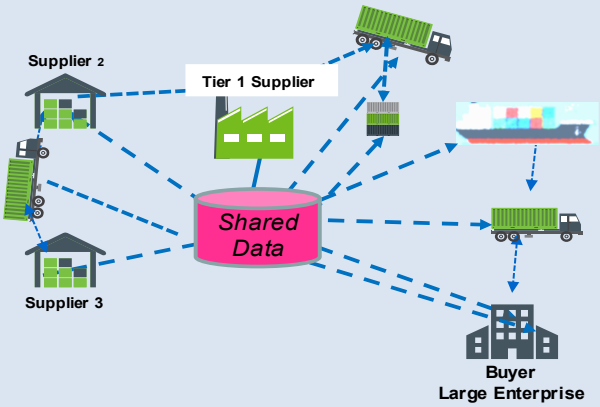
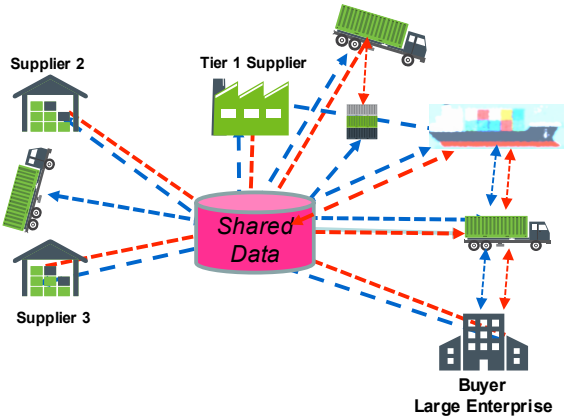
1	 ***Single Task: One-to-one***	In a one-to-one relationship, a single task (for example, sending a forecast to one supplier and getting a reply) is easily accommodated by any number of market offerings including traditional enterprise solutions supported by a B2B.[9] However, as each enterprise has its version of the truth (VoT), the challenge is synchronizing partners as things change.
2	 ***Search and Transact: One-to-many***	Compounding our challenge a bit, our standard supplier cannot meet our needs, so we are looking to consider, *simultaneously*, several suppliers who can. Immediately, we see the network option provides an obvious advantage. The customer can search and send a request to several providers, receive their responses, assess the best option, and then send an order to the selected supplier. Much better than proprietary procurement systems that require us to search multiple sources to *find* these suppliers, while, also, *we have to maintain all the data*.
3	 ***Simple Fulfillment: Many-to-many***	Execution compounds complexity further as we actually seek to *fulfill the order*. Our selected supplier is in Asia. This sets in motion significant *changes in control* across the chain. *The buyer is not in control of most of this workflow*. The *seller, de facto, owns the fulfillment process*, but also is not in control of all the steps and handoffs. In fact, *no one enterprise can orchestrate the whole workflow.* *Everyone,* not just the buyer and supplier, *needs access to the same data*. Although all parties are trying to control their own tasks, they also need to coordinate across their peers—supplier to carrier, carrier to carrier, carriers to 3PL, and so on. Everyone's decisions are interdependent. As things change—and they will—everyone needs near real-time access to that current *state of single truth* to stay on plan or to coordinate changes.
4	 ***Changing Demand: One-to-one, One-to-many-to-many***	Now what if demand changes? Upstream partners need early warning so they do not incur unnecessary expenses—excess inventory and/or misallocated capacity. Again, the network really provides an advantage. Rather than waiting or having things lost in translation, all important current events that impact demand can be "sensed" and critical plans can be responded to in near real-time. Given that no fulfillment plans have been yet created, one-to-one communications *may* be sufficient. But if fulfillment entities have already been aligned, now we need to communicate many-to-many. Resetting the network frequently can be a good or bad thing.

Figure Two: Building the Supply Chain Processes with Technology

[9] Most companies use one of the many B2B networks, or log into customer's site, or use on premise B2B.

So what does this chart above say to us?

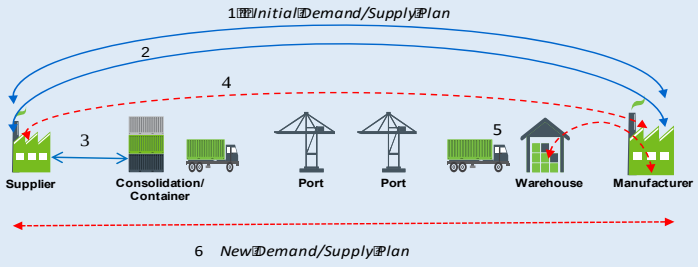
An activity like demand sensing is a good example where application networks really provide an advantage. Demand sensing is an example that *blurs the lines between planning and execution,* as it requires a continuous ecosystem of the freshest, most accurate information, to be acted on swiftly.[10]

In a supplier/buyer relationship, *changing demand should reset the network*, with simultaneous signals and subsequent mutual recalibration of everything from scheduling dates to logistics routes, prices, and so on.

But for this to work, all network partners need to be included—not just as spokes receiving serial messages, but as *participants in a multi-directional negotiation* to create an achievable response.

In a participative, reciprocal relationship between buyer and supplier, both parties can benefit!
In a participative network, multiple and mutual objectives may be achieved. That is network-wide optimizing. It goes beyond changing local parameters such as resetting a safety stock level. It involves *mutual-optimization across multiple functions and multiple enterprises*. There is a widely accepted axiom that local optimization is sub-optimization. A solution that optimizes across functions and enterprises will produce superior results across the chain, rather than just pushing the problem to different systems and players, up or down stream. For that, we need readily available, simultaneously *shared* data. Let's look at an example of network-wide optimization here (Figure Three).

5



1. Initial demand and response signal.
2. Supply-side works with logistics to determine schedules and container costs and sends to customer. **Schedules work, but analytics highlight opportunities to reduce cost.**
3. Analyze opportunity to optimize container load to reduce moves and costs.
4. Propose increase in product output to fill container.
5. Evaluate whether increase in order quantity will meet long-term demand. Can warehouse hold inventory? Can this reduce subsequent shipments/moves? **System analyzes the trade-off in costs and coordination. Cost reductions are reflected in reduced logistics costs, as well as improvement in warehouse utilization and warehouse labor costs.**

Network-wide Optimization: Many-to-many

In order to improve overall performances, we need to solve for multiple challenges. Costs are often incurred due to demands placed on partners (changes, additional coordination, extra handling, and so on). Hence, we need an overall optimized plan beneficial to multiple players.

For example, containerization is expensive, so we want to build a full container load, *without* unusable/excess inventory, and, hopefully, avoid another trip requiring more coordination, labor, paperwork, and transport costs.

Many separate applications with many messages between too many data stores will not be able to find a *network-wide* option. Here is a case that requires all the data to be in a single data store.

The additional benefit of this single approach is that with the use of machine learning, analytics can be constantly evaluating better opportunities. Applying that network wide vs. one enterprise at a time can produce a win/win.

This kind of result also motivates partners to participate in a common platform.

Figure Three: Network-wide Optimization Example

[10] For example, retailers often change ticket prices, ship-to addresses, quantities, and so on, up to and *after* ship dates.

Application Considerations

In a supply chain application network, the *applications, as well as how those applications are integrated or shared, both need to be considered.* In Part Two we will discuss 'the how'—the technology capabilities that set networks apart. Here we provide a brief comparison of several leading supply chain application network providers and their functional capabilities. Along with considering which applications a solution provider offers, we need to understand whether the applications are provided as a single *shared* network service (the RSN) vs. a set of standalone networks integrated by the common platform (the IN). Figure Four and Figure Five provide these views. We provide further comparisons in Part Two of this report.

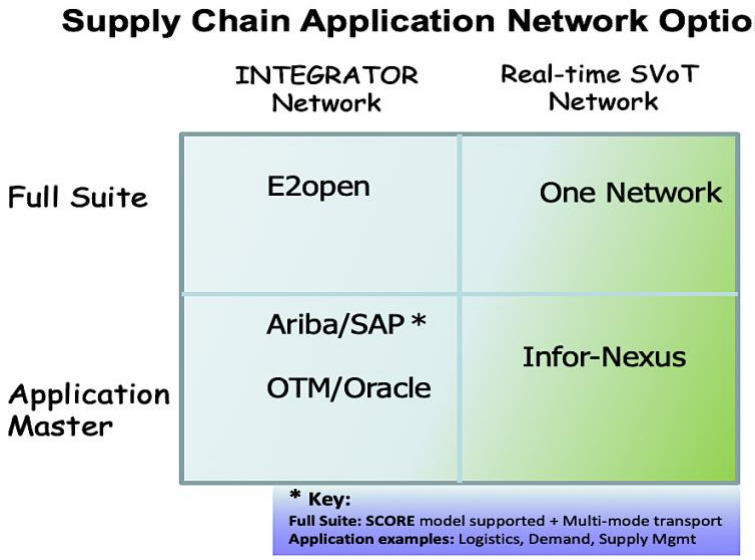


Figure Four: Network Options for Supply Chain Application Networks

Adopters of these solutions can take a functional application approach or alternatively first consider the network as their new end-to-end operating model. That choice will vary depending on their business. For example, a carrier may just need their own cloud TMS or more broadly a networked TMS to have their entire transportation and logistics business model supported. As well, the shipper may also just be looking to upgrade one function. Bear in mind that for trading partner processes to fully interoperate with one another, there can be dozens of processes that ultimately need to be shared and interoperate. This transcends a single functional application. And historical trends in tech acquisition and adoption have shown that users tend to look for more functionality on the same platform over time. Over the years, our view of supply chain has expanded from limited one-up/one-down[11] visibility in our supply chain to a multi-functional, multi-stage view. As the demands of supply chain expand to include more and more partners and modern data, the complexities of synchronization become overwhelming. Networks can provide the supporting services required. So the question then, is *how* is this done. We answer that in Part Two of this report.

[11] Such as customer/supply product/inventory planning, *or* logistics, coordination of transport and inbound receiving.
*SAP and Oracle have pursued a strategy of both acquisitions as well as organic development resulting in many network solutions. As examples, SAP has Concur, for expense/financial, and Ariba for eProcurement (which uses SAP Ariba Cloud Integration Gateway to integrate to other SAP or external systems; plus, there are many third party integration platforms that also integrate to Ariba); Oracle has OTM for transportation management, and Oracle Supplier Network which works with and between various Oracle ERPs and external suppliers (these use Oracle's AP libraries or rely also on third party integration platforms.) There are many other Application Master networks. These are just two prominent examples.

Provider	Overview The functionality listed here is based on what is *in* each solution provider's network, not what they integrate to through partners or other corporate assets.[12]	Demand	Order Management	Production/MP	D/S Matching *	Inventory Management	Warehouse Mgmt.	Channel Management	Sourcing/Procurement	Transportation	Telematics	Optimization *	Trade Finance	IoT/Geospatial	AI/IA	Control Tower	Blockchain
SAP Ariba	Founded as an eProcurement solution, and acquired by SAP in 2012, the SAP Ariba Network has grown to include four million companies, providing source-to-settle functionality on the buy side and some selling/RFx response and fulfilment on the sell side. *What you can do with this network:* Supplier management, sourcing, contract management, purchasing and payables automation (procure-to-pay), spend management, supply chain finance, supply chain collaboration.		P			P			√			P	P		P		P
E2open	Founded as a marketplace but has recently and rapidly acquired a broad set of supply chain applications—demand through transportation—each integrated with E2open's canonical data model. All application networks share the E2open integration network. E2open is taking a leadership position in channel management. ***What you can do with this network:*** Just as your own enterprise solutions leverage integration tools, E2open uses its Harmony platform to integrate between their various application products. Exchange/one-to-many by applications such as forecasting and procurement (products or transportation) are effective, but many-to-many processes, such as demand/supply balancing and multiple functional/network-wide optimization, are limited.	√	√	√	P	√		√	√	P		P	P		P	√	
Infor Nexus	GT Nexus (now Infor Nexus) was organically developed as a many-to-many RSN, providing applications for the physical and financial supply chain. Strong focus in ocean transportation. A growing, sophisticated Machine Learning foundation utilizes both operational and geospatial information. ***What you can do with this network:*** Infor Network orchestrates complex activities across their network and can integrate to other supply chain applications from Infor and others such as demand forecasting, TMS, and WMS.					P	P		√	P		T	√	√	P	√	
One Network	Organically developed as a many-to-many RSN to support supply-chain-wide applications. Other innovations include early adoption of IoT; Blockchain; as well as Machine Learning and Intelligent Agents which enable autonomous operation for some functions. Recently released Global Logistics Gateway. ***What you can do with this network:*** One-to-many, many-to-many demand/supply matching and network-wide optimization.	√	√	√	√	√	√		√	√	√	√	P	√	√	√	√

Key: **√**=Full Functionality **P**=Partial Functionality
T=Transportation Optimization Only
Green=Organically Developed **Purple**=Acquired

Figure Five: Functional View of the Supply Chain Application Networks Offerings

** For Demand/Supply Matching and Optimization we mean a network-wide capability as described in Figure Three. Optimization can be for transportation such as routing (T in this chart), inventory, space utilization in warehouse, and so on or inclusive of all elements (network-wide).*

[12] And a reminder that INs are based generally on acquisitions and RSNs are mostly organically developed. We have color coded the functionality indicators above, showing acquisitions in purple text and organically developed in green text.

SUPPLY CHAIN NETWORKS REVEALED

SUPPLY CHAIN NETWORKS—Part Two

How They Work.

In this report series, *Supply Chain Networks—What They Do and How They Work*, we have discussed the new paradigm of competitive advantage where markets can be won or lost as much by the technology platform your enterprise and supply chain runs on, as the actual product or service you offer to the market. We offered some supply chain process and functional examples to highlight how a supply chain application network can drive better performance for the whole trading partner network.

In this last segment, *How They Work*, we focus on the technology and offer clear differentiation between supply chain application networks in the market. Understanding these differences is essential when evaluating technology if we are to make good choices.

ChainLink Research 2019

Supply Chain Networks Revealed

This report series will provide a differentiated discussion on supply chain application networks from a process, functional and technical perspective.

This series consists of three parts:

- **Executive Summary**: A strategic overview of Supply Chain Application Networks and why they are an essential component of competitive advantage, along with a summary of key points from the report.
- **Part One: What They Do**: What is a supply chain network? Why is it the platform for the inter-enterprise supply chain process and functional perspective?
- **Part Two: How They Work**: Understanding the supply chain application network technology and examples of different types of network offerings in the technology market.

Table of Contents

How Do Networks Do It? What Are the Differences?

Supply chain application networks in the market today have some history, which is important to understand because it is foundational to their approach to development and architecture. As discussed, there are two fundamental approaches, Integrator Networks (INs) and Real-time SVoT Networks (RSNs).

Networks

Integrator Networks are based on messaging—a send/receive paradigm, to the network and within the network between each enterprise data store. These attempted to synchronize various partner data.

Real-time SVoT Networks are based on publish/subscribe to a ledger-like data store paradigm where all partners publish to the same database. This provides a single source of data for all to share.

Networks: Two Histories

INs may have gotten started as *exchanges or private networks (hubs)*. Examples are portals, industry-specific exchanges, or a commerce site for ecommerce or procurement.[1] This means that although they share a common application, each hub is a private network isolated from the other tenant networks.

To grow, INs often make acquisitions. This adds functionality and customers to the fold.[2] However, the applications acquired were built with different architectures, data models, and security protocols. Thus, these firms face the challenge of integrating the new data, code, and customers into their infrastructure. They address this problem by "wrapping" the applications with modern technology. Applications are invoked through application-to-application software and a common, modern UI that provides a consolidated user experience. They also leverage the B2B services *within* the platform to facilitate integration between tenants on the platform as well as external entities.[3]

In this scenario, there are usually several processing engines, integrated collections of applications, or communities beneath the UI. For example, the demand application will solve for accuracy and trade-offs between customer service and inventory levels, then pass a message to the logistics application to look for carriers.

An IN that has acquired several applications within the same functional domain may re-architect the adjacent applications into a richer, single application (for example, expanding from one transportation mode to multi-mode) rationalizing the data and code base. This provides opportunities to migrate to more modern technologies and eases the burden and cost for support.

[1] For more on these architectural approaches with web systems read: Multi-party Solutions for Supply Chain
[2] The last few years have also seen huge investments in development by ERP and supply-chain firms to develop integration layers that include A2A and B2B.
[3] E2open is an example of this, where the many networks use B2B.

The RSNs, in contrast, have standardized on a single development architecture with a philosophy of single instance/shared data and a single processing engine/code base.[4] A tradeoff has been made between the short-term acquisition of customers and the supportability and holism of a single-engine approach.[5]

These distinctions are important, as explained in the examples above. As we move from single to multi—multi-mode, multi-partner—we expand our view and, thus, need more and more *inclusion.* Everyday examples abound in multi-party objectives. For example, drop ship, which requires precise coordination, has three or more parties involved in the process: the customer, the seller, and fulfillment entities such as warehouse, manufacturer, and the actual transport provider.

So, what key architectural/technical elements should the buyer look for?

Network Architectures

Since the raison d'être of networks is the ability to operate an optimized set of activities *between* parties, it can take several forms, which are reflected in the underlying structure and services of a particular network and the way it may be implemented in an instance. There are several important concepts at work here that we will explore:

- The Database and Master Data Management (MDM)—the data model, how it is created and managed
- The Processing Engine(s)—planning, executing, optimizing, and analyzing
- Integration—how it is achieved, both business to business and application to application
- Capabilities—tools, applications, and the needed data, itself

These and other attributes are summarized in Table One on page 6.

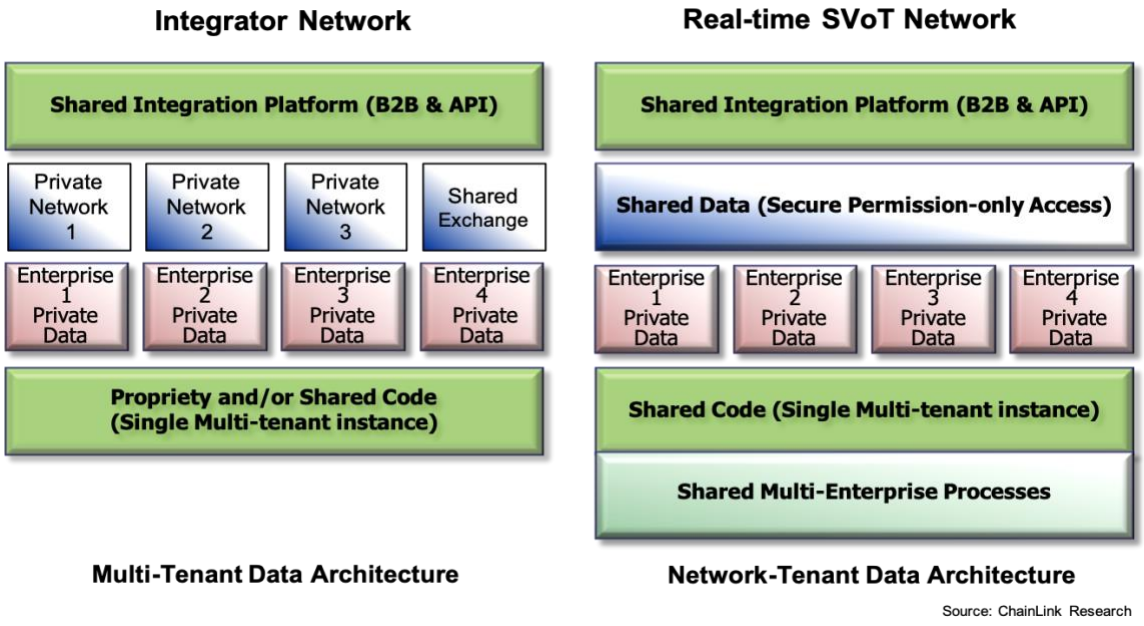


Figure One: Comparison of IN and RSN Networks

[4] The leaders in this sector use intelligent agents to act on the data, complementing the algorithms.
[5] One Network and GT Nexus, now called Infor Nexus, are examples of this.

The Data: Our Greatest Challenge, Our Greatest Asset

Sharable and *usable* data across the chain is essential to communicate and operate. Yet, it remains one of the biggest challenges in modern supply chains. Foundational, then, are the techniques we use to create, access, cleanse, and understand data.

Common Master Data Management—supply chain management cannot be achieved without consistent data; however, *consistent data has been the specter of the inter-enterprise world*, standards not withstanding!

Master Data Management has been difficult to achieve even within a single enterprise. It is just too hard to keep up with all the changes in day-to-day business data, what to say of changing regulations, standards, and the constant expansion of what we consider supply chain data.

This issue has been particularly burdensome for suppliers and service providers. Having the supply-chain network provider manage the industry data model[6] on behalf of the whole community as well as the translation between participants is a real boon.

Importantly, modern dynamic networks accept that there may be unique requirements and changes, so *extensibility* of the data model and a dynamic request/consent handshake or *permissions* should be included in the services provided by the MDM tools.

There is so much dynamism in the chain today that formal data change management is not always practical. A transaction—order, confirmation, and so on—may have within it a unique data format. Rather than rejecting the transmission, a *permission* can accompany the transaction, requesting the use of or change to a specific element. This can be a one-time or an on-going exception.[7] In this way, the business flow is not disrupted.

Definition: Master Data Management in the supply chain

MDM is a method for creating, standardizing, and managing data, creating a common data "language" that is sharable across systems and entities.

In addition, flexibility of data (*extensibility*) is crucial, since supply chain is a fluid environment where products, people, and situations often change. Networks need to be architected to support a shared common data model, while allowing for the dynamic creation of community-specific or company-specific customizations of and extensions to that data model.

Modern Data for a Modern Supply Chain—*What Data Do I Need*?

- Breadth, depth and granularity
- Community connections
- Public data—schedules, certifications, rates
- Trading-partner data
- Unstructured data
  - Video/voice
  - Social/search
  - RF, Mobile, GPS
- IoT/device and equipment conditions/sensor data: e.g. vibration, temperature
- Temporal Data:
  - Geospatial/weather
  - Location directionality
- Goods, assets and people

[6] Supporting the standards by which an industry and/or ecosystem operate, for example, automotive, defense, retail and so on. These can include master data, business logic, and specific rules and algorithms common to that industry.

[7] For example, "please use this data format for this order," or "every time you do this particular task, please substitute this definition," and so on.

Database—This contains the actual data built on the MDM. Data instance options offered today are the *single-tenant* (private data), *multi-tenant data* (shared database), or an overarching *network-tenancy* (shared database, plus tenants have private data). For enterprise-oriented challenges such as finance or proprietary design, the private, single-tenant option may be desired. In one-to-one communication of plans in the supply chain, we can usually do pretty well. *However,* add dynamism—*many* changes—and partners are left with debates on whose version speaks the truth.[8]

Definition: Permissions

A system service that allows the use of a process, data or access; a secure request/commit.

RSNs are network tenant[9] and their key differentiator from the IN is that essential data associated with a process or transaction are posted to that same database.[10]

Integration or Interoperability

Integration is an elusive concept. One of the obvious challenges in supply chain is that everybody has so many systems to support their unique needs that it can restrict what they are capable of—or willing to do. And that is not likely to change any time soon.[11] Enterprise systems have been developed around a construct of process (codified in the software)—do this, then do that. This rigidly reflects the "one" philosophy of the enterprise. They use APIs, workflows, and translation to bridge between agreed-to processes.

Definition: Interoperability

Not just the ability to make use of each other's data, but to operate in common.

Nonetheless, in a world of infinite connections, we need the ability to connect and understand the other's data or intent. For some activities, however, we need *more*. Partners may need to *participate* in a common process. That requires a lot more than translation. In highly complex processes where all participants need to conform to the agreed-to process, they need to *interoperate*. Technically, how is this achieved? A technique called *process inheritance* may be used (see definition in side box).

Definition: Process Inheritance

A process or object can inherit the definitions/rules of another. This supports more process flexibility as partners change rules, applications, and so on. (There are multiple layers of inheritance: data, process, rules, and/or code.)

---

[8] Debates on service levels, chargebacks, pricing and costs abound in most industries.

[9] INs and RSNs also provide enterprise control towers, since, in practice, most companies have on- and off-network applications, data, and partners. In practice, the major supply chain application networks will support various customer or industry requirements.

[10] You may ask, if the data is in a real-time database, how is it kept up to date if sources are not part of the network? Of course, most regular sources will update their partners when things change. Beyond that, to ensure the best possible data, techies have several techniques, such as machine learning, that they use to evaluate data quality and regularity of feeds. A source system can be unavailable or of poor quality, so a method called **mean perturbation** can be used. This means the software knows that a change in the state of the data should be occurring and assesses what the data *should be* (determined over time with experience, i.e., machine learning). These techniques are used to analyze the data and supply the database with improved data quality.

[11] Beyond software, each equipment provider (transport, manufacturing, or warehouse) uses their own standards plus many types of devices—mobile, GPS, sensors, RFID, ELD, and so on. All these need to be integrated together. Equally important is the explosion of new data sources from the web, as well as nonstandard data.

Extending the Solutions

We often need to customize some of the services and integrations. We extend applications by integration to external apps and/or configuration, customizations, and development on the network platform.

APIs and Workflow Engines: For most tech companies, application expansion is enabled by providing APIs to allow customers to develop *external* code that can integrate to the application. These API libraries reduce the work involved (since someone else probably had that need and all sorts of objects have already been developed). Technologists' goals are to create a low- or no-code environment to ensure standardization and make it easy for customers.

Workflow engines ensure that the human tasks occur in an agreed-to script. This needs to be easily adaptable for users to manage.

All supply-chain application networks will have at least some APIs and workflows, but may differ significantly in the type, breadth, and granularity of objects and functionality exposed.

SDK/aPaaS: Another approach is a software development environment that lets you develop *within the platform*. This is often called *aPaaS* (application platform as a service). Here, as well, the low- or no-code goal is sought, but the same development tools the network developers use are provided so that the final product is compliant and can work *in* the network. There can be multiple layers of capabilities within the SDK (software development kit) which allow for routines, workflows, and data model extensions, again, specifically compliant with the network architecture.

Summarizing the Networks—Capabilities and Examples

For quick reference, we will summarize some examples of supply chain application networks. Table One offers a comparison of the technology options.

Capabilities	Enterprise Application	Integrator Network	Real-time SVoT Network
Data model/base	**Enterprise-centric database**	**Private Data Networks** for each enterprise • + Private Enterprise data Data is transmitted between networks on the platform across multiple applications via B2B messaging • Shared MDM and canonicals	**Network-tenancy** • Multi-party shared/network-wide "ledger," shared data instance • + Private enterprise data • Shared extensible/network-wide MDM
Processing Engine	One per application	• One per application • Multi-tenant/single instance applications	• Single processing engine leverages machine learning and intelligent agents
Communication Paradigm	**Paradigm: send/receive** • One-to-one • Point-to-point	**Paradigm: send/receive** • One-to-one, one-to-many • Connections may be built for each network application offered by the vendor • For more modern INs, connections are built only once	**Paradigm: publish/subscribe to ledger** • Many-to-many, one-to-many, one-to-one • Connections are built only once
Visibility	• Enterprise control tower and/or data warehouses built on B2B communications data acquired from external sources • A2A to keep in sync with other applications	• Enterprise control tower and/or data warehouses built on B2B communication in their internal network as well as external sources • A2A to keep in sync with other applications	• Leverages intelligent agents on a shared database to provide intelligent triggers and alerts • B2B for external sources • Control towers to support unique user needs
How do they optimize?	**Solution domain: each application optimizes independently** Data needs to be sent from multiple sources, translated, and loaded into optimizer. • Large optimization runs • Each application optimizes independently • Conflicting or irreconcilable optimization results need to be resolved manually	**Solution domain: each enterprise tenant and each application optimizes independently** • Send requested changes to other networks or enterprise tenants • Depending on optimization engine design, conflicting or irreconcilable optimization results may be resolved by back-and-forth communication between engines and re-runs of optimization	**Solution domain: cross-application and cross-enterprise optimization** • Conflict resolution happens within the engine • Can be incremental and/or continuous optimization
Application Integration	**Paradigm: integration** • Vendor specific • Custom code • Integration package	**Paradigm: integration** • Sharable integration library • Web services	**Paradigm: interoperable** • Sharable integration library • Web services • SDK
Process Flexibility	• Customizations	• Adaptable workflows, APIs	• Adaptable workflows and APIs • Process inheritance

Table One: Comparison of Technology Options in Supply Chain

Additionally, buyers should consider the solution domain/focus of the network provider. Table Two highlights key examples of application networks we mentioned in this report. Figure four provides a market overview of various types of network options.

Company	Network	B2B /Integration	Solution Domain	Platform	Advanced Capabilities
GT Nexus (Infor Nexus)	*RSN*	GT Nexus Commerce Network	• Ocean transportation • Logistics • Trade finance	Organic	• Control Center • Coleman AI—Infor's AI engine • SDK
E2open	*IN*	• E2net—B2B layer • DX/IDM—data integration layer • Harmony—UI	• Supply chain-wide applications	Acquisitions and Organic	• Harmony UI consolidates user experience for multiple applications • Control tower
One Network	*RSN*	Real-time Value Network	• Supply chain-wide applications • IoT per industry	Organic	• Autonomous network • Control tower • NEO—One Network's intelligent agents • Blockchain • SDK
SAP	*IN*	NetWeaver	SAP's Ariba is their largest business network support sourcing and procurement	Ariba network was development before SAP acquired it.	• Supplier spend analysis • Strategic sourcing

Table Two: Examples of Supply Chain Application Networks

NETWORK OPTION EXAMPLES

Application Integration Platforms	B2B/EDI Cloud	Visibility Track and Trace Solutions	Supply Chain Application Networks/ Functional	Full Suite Supply Chain Application Networks
Libraries of integration code and techniques to support major commercial applications	EDI, AS2, data translations and nonrepudiation of transactions	Built on B2B, IoT/ device integration (equipment, RFID, sensors, GPS, Mobile, etc.), Blockchain	Include Applications. B2B, API, Visibility with functional focus area	Full supply chain coverage + common MDM, B2B/API platform
Informatica MuleSoft TIBCO	Cleo DiCentral Open Text TrueCommerce SPS Commerce	A2B Navis Four Kites Savi TraceLink TradeLens Zest Labs	Elimica Exostar Ariba Infor Nexus Oracle OTM	E2open-**Integrator Network** One Network **Real-time SVoT Network**

Figure Two: Comparison of Typical Networks in all Market Sectors

Is Autonomous Our Future?

There is an ongoing discussion about autonomous supply chains. No, that does not mean robots will be sitting at our desks. But it does mean that some of the drudgery of dealing with disparate and overwhelming data volumes can be reduced, opening up our time and minds to explore more possibilities.

Machine Learning, IoT, Blockchain and other methods have become part of the common language at supply chain conferences. However, the road to making them a reality varies according to their value and the community's readiness.[12]

For example, Machine Learning/AI requires a deep base of data over a long enough period to actually *have something to learn from*. Then the appropriate Intelligent Agents (IA) need to be available to take action based on the best possible solution.[13]

IoT requires an *instrumented supply chain* and interoperability to rapidly access and utilize device, machine level, and location data. IoT can draw on machine learning to monitor and proactively avoid negative events, or at least apply predictive analytics based on device/equipment performance history.[14]

For many, the promise of a *tunable system* is important. Supply chain is getting so complex, and with expertise limitations, many companies want a lot more automation. We want to 'tune' routines and process flows to reflect exceptions or to evolve quickly with as little human intervention as possible. Within environments with isolated or restricted data instances, a mutual tuning to maintain interoperability may be challenging.

As we hope we have hit home by now, the inclusion of a network-wide ecosystem of participants and their data is foundational to achieving a smarter supply chain. With a smarter supply chain, we can leverage these more advanced capabilities, which can *learn*. With learning, the community can use and *trust* the system. With that trust, we *can* let the system make some decisions and operate autonomously across the network.

The technology is ready. The question is, are we?

Let the System Decide—Intelligent Agents

In a multi-party process, there can be hundreds of variables that influence even the most common decisions. Weather can impact independent demand—and dependent demand. A closed bridge in a foreign land can affect on-time delivery a world away.

One can imagine the cascading dependencies and changes that might have to be made. Humans, who are already managing hundreds of shipments or thousands of products, may not be able to manage it all.

Autonomy does not have to be intimidating or beyond the reach of our comprehension. Simple but powerful improvements, like autonomously updating and improving lead-times or safety stock levels, can provide huge value.

Machine learning today can sit in the background watching, analyzing, and understanding. Intelligent agents that can replan, alert, reorder, and so on, based on massive data analyses, can determine the best option. This can take the drudgework away from users who can then focus on unique situations and person-to-person interactions.

[12] For example, there are several initiatives for Blockchain, but rollout has become problematic. Some network providers do offer Blockchain capabilities and, if called upon, they can rise to the occasion and provide them.
[13] IA (Intelligent Agents) act on operational data just as traditional application code does. The methods and deployment of IA is an important topic but is beyond the scope of this report.
[14] IoT is also a substantial topic, beyond the scope of this report.

Conclusions—Going It Alone or in a Network

Today, many users are still just trying to optimize *their* enterprise task using stove-piped systems such as inventory or TMS, working on each task independently. The rare enlightened supply-chain department might even have the logistics person *talk* to the procurement person and try to create a balanced plan.

Yet, over the years, our view of the supply chain, which encompassed limited functional views of one up/one down[15] has needed to expand to a multi-functional, multi-stage view.[16] We know our *physical network* includes all the elements that support fulfillment—*seen and unseen*. It's not just partners, but also the environment that can have an impact on the flow and integrity of product.

As the demands of supply chain expand to include more and more partners and modern data, the complexities of synchronization become overwhelming.

Some organizations still harbor the 'go it alone' philosophy of building their own environment with 'piece parts' of technology—B2B/EDI, gobs of custom application interface programs, database tools and multiple applications. The challenge is that in the supply chain arena there is constant change and ever-expanding data. *You are not in control of much of this.* Thus, you need to work with organizations who have the scale and depth to support you. But this is not technology as usual, where the customer signs up for months of customizations and lots of new integration code to support for years, as well as managing all that data, while continuing to have detached/disjointed information with trading partners.

Improving overall performance across the whole chain is the quintessence[17] of network objectives today. Leading companies—and that is not necessarily determined by the size of the company—who want to continue to innovate in their supply chains, participate in truly collaborative interactions and processes, and open themselves to more customers, partners, and markets, can leverage the power that networks have to offer.

To meet the supply chain challenge and seize the opportunities, there hardly seems a choice anymore. We *need* networks.

[15] Such as customer/supply product/inventory planning, *or* logistics, coordination of transport and inbound receiving.
[18] Most systems today are still designed, sold, and implemented in these functional stovepipes, but that approach won't get us to the goal.
[19] Definition of quintessence: typical example of a quality; exemplar; stereotype; epitome; paragon, picture; prototype.

321 Walnut Street, Suite 442

Newton, MA 02460-1927

617-762-4040

Email: info@CLResearch.com

Web: www.ChainLinkResearch.com