



## Getting to The Real Heart of a Vendor's Software Architecture

*By Mark Fralick  
SupplyChainDigest Contributing Editor*

Does technical architecture still matter in supply chain systems? This is an interesting question to ask, for two reasons. First, one might assume that most supply chain software vendors are pretty far along in their architectural evolutions. We are well past the 1980's and early 1990s, when vendors had to figure out their own ways to do things because, frankly, nothing much existed to help them. Today, however, there are known standards, well-developed frameworks, and with advances like CE.NET there is even a standard framework for mobile devices, which have always been a challenge for vendors needing mobility, such as WMS.

Second, technical architecture has traditionally been something that only interested IT personnel – the operations/business side for the most part has only been concerned with an application's functionality, in truth often find it amusing when IT people are so concerned about the underlying technology.

I believe quite strongly that when evaluating supply chain vendors, architecture does still matter, not only to IT but to the business decision-maker as well - but not for the reasons you may be thinking. This discussion is worth considering for companies are evaluating supply chain vendors, or for those giving consideration to building their own supply chain applications.

Rather than focusing on technologies and tools, judgments on architecture should be about two things: an underlying philosophy and "connective tissue." By asking the right questions, in a controlled format, you can in fact separate much more clearly the architecture differences that really matter to your company and its supply chain needs now and over time.

### **Just what is Architecture?**

The areas generally associated with "architecture" are things such the programming languages, standards, databases, etc. used by a vendor. The reality, however, is that these are very temporal issues. What is right today may not be the best thing tomorrow. You should judge the architecture based on a philosophical approach.

I recommend addressing a vendor's architecture first with three broad questions, drilling into detail later:

- How does the architecture address making the system successful today?
- How does it protect my investment in technology (make sure my system will be successful tomorrow)?
- How does it enable me to run my business and change based on business dynamics?

By starting with these questions, you can make judgments about a vendor's technical decisions (e.g., standards, languages, database technologies) based on how they address the problems of the underlying philosophy. This changes the game from a very point-in-time question (as pure technology always is) to questions of how the vendor addresses the bigger issues of success and change over time.

## Connective Tissue

"Connective tissue" refers to the way the software interacts within the vendor's own suite of solutions and with the world outside. Connective tissue generally comes in two forms: old baggage (bad) and currently philosophy (good). Old baggage is the result of the fact that many of these supply chain vendors have grown, through the years, by acquisition. This means the "product" you are buying may have been developed by different companies using different technologies and cobbled (for lack of a better term) together.

This may not be all bad. There are good examples of this approach working well. IBM, for example, arguably had been great at this, as evidenced by the growing breadth through acquisition of its WebSphere offering. However, IBM has the resources to invest significant time and effort on technical alignment before it pulls products under the WebSphere umbrella. Supply chain vendors just don't have the same amount of resources. In their haste to get things to the market they might just slap a new splash screen on the acquired product and declare victory. In these types of situations, you might find vastly different technologies – but more importantly vastly different philosophies of implementation.

It may not be easy to dig into this old baggage because the vendors would much rather discuss their current philosophy and future direction – which will always sound great. The question is - when will they really get there – if ever. This discussion is certainly very timely now, as we see the rise of Web Services, Publish and Subscribe mechanisms, Services Oriented Architecture, and ability to hook into Business Process Integration (BPI) systems. All of these are good things. But how much of it is real in a vendor's solution? It is an important issue, and ultimately goes to Total Cost of Ownership.

A large retailer uses an approach that does as good a job as I have seen in getting under the covers of a vendors solution - to find what's real and what isn't. It is very powerful, yet absolutely brilliant in its simplicity and execution. In fact, I've adopted some of their technique for when I do technology evaluations and corporate due diligence. Here's how it works:

1. Invite the vendor to do an architectural presentation. If they are in for a functional demo, invite them to do a formal architectural breakout session.
2. This architecture presentation need not, and indeed probably should not, have sales people and their smooth-talking demonstration sidekicks present. Why? Because this conversation is all about "what" the architecture is and not about the "why" (reasons/benefits). Make sure the vendor brings someone who can well-address the "what" questions.
3. Require the vendor to prepare the answers to a small number of questions. I think this particular retailer had 12 questions. I use 15. Examples:
  - List the products typically sold in a deal. Separately list 3rd party products required or supplied.
  - For each product, indicate who developed (owns) the product.
  - Provide the number of developers for each of the product components.
  - Provide a complete diagram of the solution being proposed, and a component definition describing each module or piece.
  - Provide an interaction diagram describing the flow of data from the presentation layer, to the business layer and finally to the database layer.

There are a several keys to making this work.

- ❑ First, control the content. I permit only 15 slides – one for each of my questions. No sales or marketing materials permitted since it doesn't apply to the "what" question. Salespeople are not permitted to give an "Intro."
- ❑ Second, create a non-judgmental atmosphere. The discussion should be strictly matter of fact. Again, this is entirely about the question of 'what' the architecture is – not why or how it could or would be better.
- ❑ Finally, be a stickler about the diagram. It needs to be at a level where you can see all the components and, more importantly, all of the interactions. The diagram is the key. It will show you the baggage, if any, or at least give you a feel for how consistent their architecture is.

I've been on both sides of the table for this type of presentation. I can tell you that as a presenter it is a bit like taking a physical. You feel like you've left your fancy cloths out in the waiting room and you're just there in a hospital gown. However, from the customer's viewpoint, it is empowering. It permits you to cut through the sales hype and really understand what they are pitching and what's underneath the covers - which many companies never really get to see completely.

Keep in mind that vendors may complain. Get their presentation stack ahead of time. Require them to present only the slides that are submitted and approved by you. Control the content and remove anything not on your list of questions. Be calm and persistent about getting to the "what" question of architecture. After all, you're the one that will have to live with it.

## About the Author

*Mark Fralick is president of ROI Solutions and a SupplyChainDigest Contributing Editor. Having operational, implementation and software development experience, along with detailed understanding the workings of software and services companies, puts ROI Solutions in a unique position to stand strongly on the customer's side in the battle for ROI in system selection and implementation. Learn more at [www.GetUsROI.com](http://www.GetUsROI.com), or email Mark Fralick at [Mark.Fralick@GetUsROI.com](mailto:Mark.Fralick@GetUsROI.com).*

## About SupplyChainDigest

SupplyChainDigest™ is the industry's premier interactive knowledge source, providing timely, relevant, in-context information. Reaching tens of thousands of supply chain and logistics decision-makers each week, our flagship publications - *SupplyChainDigest* and *SupplyChainDigest – Logistics Edition*, and web site ([www.scdigest.com](http://www.scdigest.com)) deliver news, opinions and information to help end users improve supply chain processes and find technology solutions.

For more information, contact SupplyChainDigest at:

937-885-3253

[www.scdigest.com](http://www.scdigest.com)

email: [info@scdigest.com](mailto:info@scdigest.com)