Collaborating with Strategic Business Units in Support of Enterprise Applications

While the following article inevitably leads to the approach Avada Software took in creating its Infrared360 product for managing, monitoring, testing, healing, and providing statistical reporting for Enterprise Middleware Environments; it is also in part a research paper on the adaptation of market trends that lead to the innovation of such products.

The article title identifies three major subjects that are woven into this discussion. The basic foundation of the approach to be discussed requires understanding of the three subjects in order to weave them into a cohesive theme. Yes, this is a technology discussion. Yet the discussion does not begin without the observation of the environment to which that technology is applied. [*As a technologist I was trained to do things just because they were more technically sound. However, I learned early on that understanding the business that the technology supports is invariably more successful for both.]*

Enterprise applications are defined as applications that collectively model an entire business process across an organization. They are not intended for the individual or a few individuals. They are usually complex and can be leveraged across multiple departments or systems that have common requirements. Enterprise middleware is simply the technology that bridges these departments or systems. It is the 'stuff in the middle' (pipes and putty) that takes information from point A to point B.

To that end there is the Strategic Business Unit (SBU). Each SBU organizationally maps to one or more department, system, and/or applications. However, an SBU relates to an overall corporate entity distinguishable from other business activity. It has somewhat of its own autonomy because it has self contained business objectives. It usually has its own market and/or operating plan as well as its own profit and loss center. Therefore, whether providing a service or a product, it needs to sustain itself, if not grow.

Business units depend on technology, which supports business applications, which in turn supports business processes and objectives. Therefore, the business unit is dependent on technology at all large companies (enterprises). Yes, techies are still in demand! Show me an enterprise business that does not require a large quantity of development, implementation, support, or updating, and I'll show the who's who of 'what ever happened to that company'.

Herein lays the quandary. Executives consider that all underlying technology is a cost center. It costs X to deliver Y. If they could just reduce X, then Y would be more profitable! Technology is needed, yet not necessarily wanted. Reducing operational costs mean higher profits. Those SBUs that find areas for reduction, yet don't diminish services, are always the most valued.

What does this have to do with the title?

In order to maximize the use of their technology investments, business units are asked to share technology, especially Infrastructure resources; otherwise each SBU would need duplicate environments, both hardware and software. This forces your top SBU-A to share that infrastructure with another top SBU-B, with an average SBU–C, and with a lagging SBU–D, etc. This has ultimately led to the entire movement to virtual and cloud systems. We'll save *that* discussion for another day.

In 2007, McKinsey's June quarterly survey of business executives[1] found that those SBU's that were having the most positive outcomes had two key business factors in common among their executives and SBU managers:

1- They were able to leverage commonalities with other SBUs, and

2- They collaborated on initiatives

Regarding the technology area, following that approach within those same enterprises was a bit more problematic. The advent of Enterprise Middleware, Service Bus, and SOA technologies allowed different systems and applications to be interconnected and share information between applications. Yet, the technical groups creating, maintaining, and supporting these systems did not follow this model. There were, and still are, limiting factors: security, knowledge of multiple technologies, operating systems, programming languages, and data formats, etc., making it a cost prohibitive and labor intensive process. Systems were shared where they could be, but access to those systems was constrained because of insufficient subject matter expertise (SME) for those limiting factors above.

In Enterprise middleware technologies, some SME's were devoted to an operating system (OS). Some of the reason was due to skills specialization, but much of it was that *particular* applications resided in that OS. Administrators and support personnel were specialists for that OS and the only access point was via the specialist. In order to leverage or share support for the different SBUs, centers of excellence or infrastructure groups were created so that the resources could be shared, enabling a major cost reduction for the business units. However, the result of hardware farms, OS farms, and technology farms is that 'they all share the same kitchen'. An enterprise may have 100's of servers containing data or processing transactions. Each server is handling that data or those transactions for scores of different applications. This means that the SBU-A domain expert can be accessing, and supporting, and maintaining its

[1] Source June 2007 McKinsey quarterly survey of business executives

application on the same hardware or logical server (virtual server technology is only a partial solution) as SBU-B, C, D, etc.

Collaboration

So we've now achieved economies of scale, shared technical environments, and leveraging horse power and data space. Yet, administrators and or support personnel of SBU-A's application are 'in the same kitchen' with that of the other SBU's. To complicate matters, there are now many internal and external compliance rules and laws in effect. A person is only supposed to have visibility to the information needed for their job and no more. And that person also needs to correct a problem on a server, or logical server, or application server! We need their expertise. But in the compliance world, or in the independent SBU world, why are they seeing or accessing *my* application, tools or data? If an airline ticketing application is having problems, for instance, an administrator or support person may need to look deeply at that application on a particular server. But that same server has the crew manifest or the cargo manifest applications! Do I want that person seeing information they shouldn't? Do I risk the possibility that that person could somehow mistakenly alter or modify something they should not; something that might affect another application? No!

This is complicated even more since enterprise applications have many moving parts. Joe, the WMQ administrator, might have a streamline enterprise messaging backbone system, but he likely doesn't know much about the applications putting data onto it. Doug, the ESB data transformation expert, may be great at that, but doesn't know much about the enterprise messaging backbone. Mary, the SBU executive might know everything about the business applications, but nothing much about either the backbone or the ESB.

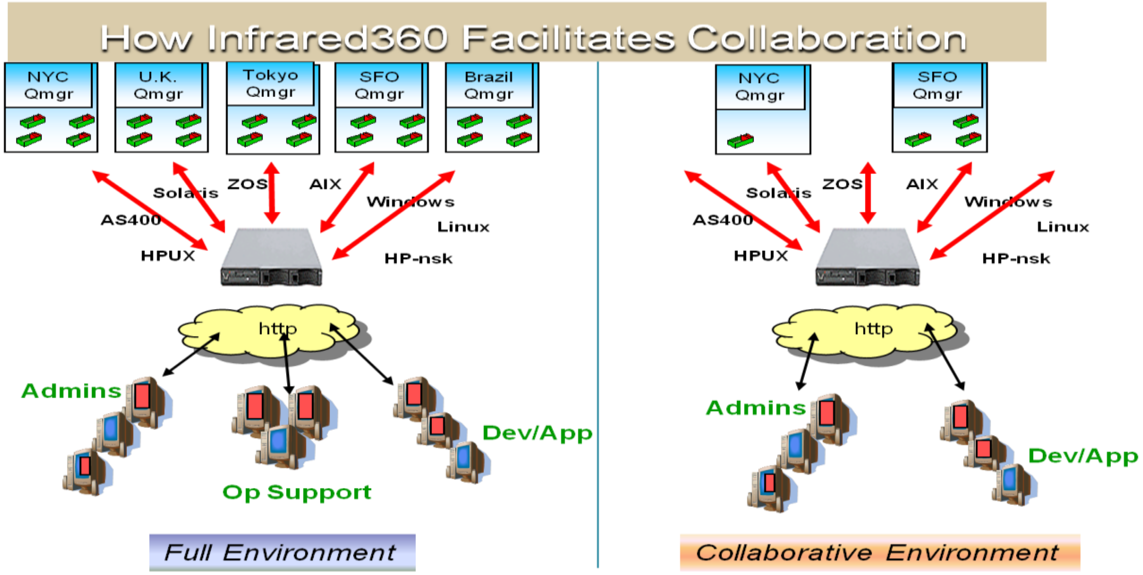
When something goes wrong, they first need to determine who, what, and where?! But given they likely have no overlapping visibility to a holistic system, they each work in their own silo researching and testing and analyzing if the issue is their domain.

So the question comes up, why? Why aren't the SBU, the ESB, the backbone administrators, and SMEs not collaborating like their business executive brethren or like the technologies they are supporting? Well, those business executive brethren do not have their hands tied as much by the technical limiting factors mentioned earlier.

And that's how the problem presented itself to the founders of Avada Software, who lived and worked in this world. What if they *could* share a problem domain, without worry of seeing what they're not supposed to see, without worry of touching what they're not supposed to touch? What if they could do that while at the same time collaborating with each other on the problem domain to which they all had visibility, but never worry about performing an action they were either not authorized to perform, did not have the expertise to perform, or was problematic and prone to error?

Joe could share his kitchen. Jane could see, but not use his appliances. Doug could use some of his appliances, but with restrictions that prevented him from changing any settings.

This was the genesis of Avada Software's Secure Collaboration™ and delegated administration.



A Complete View from a Single Portal

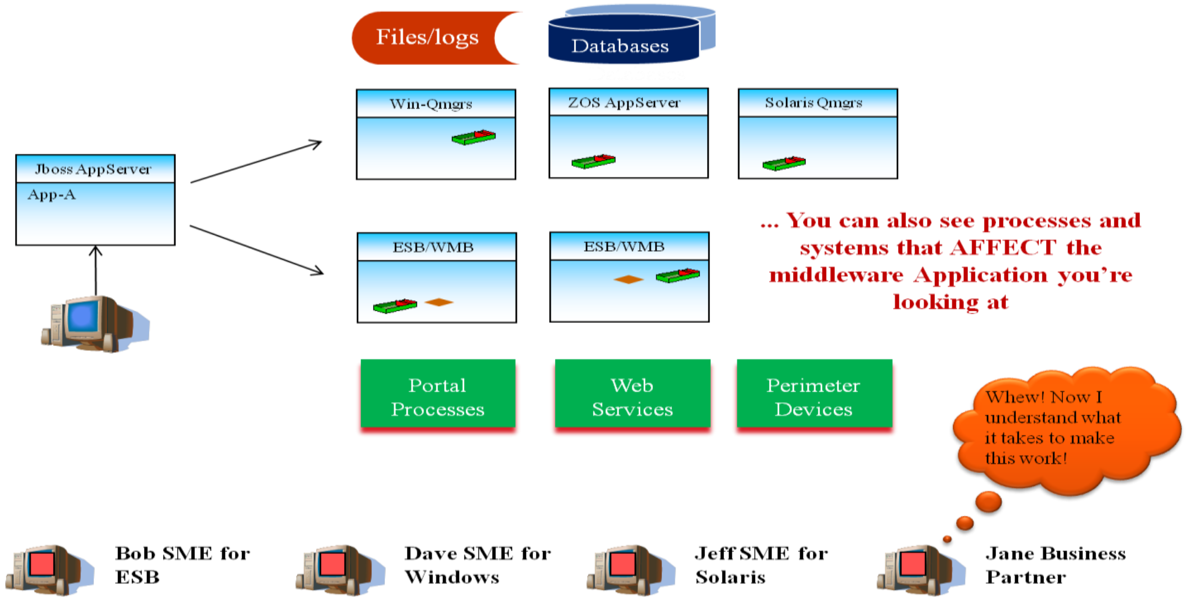
But it wasn't enough to provide just that collaborative ability. The servers, objects, and configurations that people had access to were on different operating systems. The product needed to provide ONE UI that would be platform independent and could reside in the enterprise ecosystem it supports. In this way, SMEs and support personnel from different areas and expertise could look at their application infrastructure intuitively, no matter what OS it was running on. Jane, Joe, and Doug could share the problem domain without the OS-dependent syntactical challenges. But Jane could see only her SBU while Joe and Doug had visibility to others for which they were responsible. Further Doug could make changes to configurations in his domain expertise but not to Joe's, and Joe could do likewise in his. If they needed someone from Mary's group to help expedite a problem resolution, they could give that person visibility without permission to take any action.

The idea and the solution as implemented by Avada Software in the form of Infrared360 is now managing enterprise middleware environments for many of the largest Fortune 1000 financial, pharmaceuticals, manufacturing, utility, insurance, retail, B2B, and transportation organizations.

It wasn't enough to just provide the solution above, but to provide it in a manner that aligns with mandates of cost averse executive sponsors. So Infrared360 is implemented as a private cloud solution. It sits on ONE server or VM image. It has no other deployed parts – none!

An SBU Focus

Now sit back and think about what it looks like to quickly and immediately create SBU-focused logical sandboxes for joint/collaborative use by application SMEs, administrators, and support personnel. Each SBU, or even the business applications within an SBU, will have its own contained environment to review, inspect, or share (based upon corporate compliance and permissions). The problem domain will contain all the artifacts needed to review: the middleware servers, and the objects within them (queues, connections, processes, transactions, etc.), the associated technologies for persistence of the transaction data (files, databases), the tangent technologies that either drive or receive the transactions (application servers, ESBs, web services, perimeter devices (like IBM's DataPower); all of this presented as a virtual, interrelated environment on the Infrared360 screens.



Unlike environments where a monitor solution stands on its own to alert and notify, then an analysis team uses other tools to do problem determination, then an administrative team uses another tool to make necessary changes, then a forensic team uses yet another to review statistical reports. Infrared360 allows all of that to happen in a comprehensive, holistic SBU-focused, virtual environment.