

ZAPTHINK ZAPNOTE™

HOW SERVICE-ORIENTED ARCHITECTURE TRANSFORMS THE MEANING OF LEGACY

Analyst: Ron Schmelzer

Abstract

To many technologists in the IT organization, the word “legacy” connotes a negative meaning, but not because the systems they are dealing with provide little value. Rather, on the contrary, these systems are of tremendous value to the organization, but interacting with them comes at such cost and with such complexity that little remains of this value to the organization.

Service-Oriented Architecture (SOA) promises to turn this problem on its head. One of the most powerful concepts is the notion that a business can achieve true reuse of its assets through SOA, and thus achieve the flexibility and agility it so desires. The whole idea of reuse is to get more value from what has originally been built. As such, the ideas of reuse and legacy are really one and the same thing! How can a company even hope to achieve any aspect of reuse if they are continually rebuilding their existing capabilities? Indeed, true reuse demands the continuation of legacy, and thus transforms the concept of legacy from a problem-spot to the core asset of every enterprise.



All Contents Copyright © 2006 ZapThink, LLC. All rights reserved. Reproduction of this publication in any form without prior written permission is forbidden. The information contained herein has been obtained from sources believed to be reliable. ZapThink disclaims all warranties as to the accuracy, completeness or adequacy of such information. ZapThink shall have no liability for errors, omissions or inadequacies in the information contained herein or for interpretations thereof. The reader assumes sole responsibility for the selection of these materials to achieve its intended results. The opinions expressed herein are subject to change without notice. All trademarks, service marks, and trade names are trademarked by their respective owners and ZapThink makes no claims to these names.

Rethinking Legacy in the Service-Oriented Enterprise

In the technology world, the word legacy has an often negative connotation. Most technologists deride a system or application as legacy because a newer technology or a newer approach has supplanted it, but not because it is no longer providing value. This negative connotation comes about mostly through our experience with legacy systems. Many technologists have found it difficult to build or run new applications on legacy platforms, and many times older operating systems, enterprise applications, or middleware fail when made to perform new tricks they were not originally designed for.

But outside the jargon-rich world of IT, the word “legacy” is not only a positive term, but a source of pride. Indeed, Roget’s Thesaurus lists as synonyms of legacy: heritage, birthright, estate, gift, tradition, endowment, and heirloom. Who would not want to have an heirloom or a birthright? Why wouldn’t the technology that companies have invested time, money, and their reputations in have the same sort of worth and value as an endowment?

For the information technologists whose job it is to retain the ongoing value of their existing systems, they often are critical of their legacy systems because the technological complexities of dealing with legacy, for these individuals, are significant and can overshadow the value that legacy technology provides the business. There’s no doubt that technologists recognize the value of legacy systems as well - but they also recognize the challenges in addressing new requirements from the business users. If companies can simply shift the balance of technology such that the care, feeding, and growth of legacy assets requires little added investment, then an organization can not only realize continued value from their legacy, but also demand that the legacy remains in their business. Indeed, in this environment the entire meaning of legacy has changed from one of being old and potentially obsolete to being continuously renewed and empowering.

“The legacy systems that companies have wanted to replace, and that were easy to replace have already been replaced,” says Ivo Totev, vice president of the crossvision™ product line for Software AG. “The remaining legacy systems are of high-value and still provide unique and significant competitive advantage for the organization. Service-orientation helps to transform those systems into more user-friendly and agile IT assets so that their value can be extended.”

Rethinking the way in which existing technology assets can meet new business requirements is transforming legacy, not just as a technology but as a concept. Leading the movement to rethink the position of legacy in the enterprise is the rise of Service Orientation. Fundamental to Service Orientation is a separation between the business requirements and logic, and the technology. In today’s non-Service oriented applications, business logic and implementations are tied together, meaning that most legacy applications have business logic locked up inside them. In a properly architected Service-Oriented Architecture (SOA), business Services represent the data available to the business and the core functionality of the underlying systems. Business people then compose those Services into Service-Oriented Business Applications (SOBAs) that implement Service-Oriented processes, create metadata and policies that further configure the SOBAs based upon the applicable business constraints, and then expose those SOBAs as Services that other people can compose into new SOBAs.

In the long term, the only way to gain organizations’ required business agility is by making this shift from tenuously-connected islands of IT functionality to an environment where requirements are met in this flexible and continuously adaptable manner. This change in thinking will transform the concept of legacy in the Service-Oriented enterprise, because Service Orientation offers a way for companies to think differently about legacy in the first place.

If companies can easily evolve or replace SOBAs whenever they desire, then they'll never become legacy. Because SOBAs are based on compositions of individual Services, it is easy to evolve or replace a service within the SOBA without impacting the whole application. Indeed, if legacy means that a technology is difficult to replace, then certainly this doesn't apply to SOBAs since people are supposed to alter or replace them. Services, and the SOBAs built with them, need never become legacy because SOA allows us to replace applications without the pain and cost associated with replacing legacy.

If You Want Reuse, You Need Legacy!

One of the most important benefits of SOA is that users can create new SOBAs from existing Services. In other words, Service reuse becomes the mantra, rather than application integration. As they create new Services that they can in turn reuse for new SOBAs, companies can realize significant return from their SOBA development investment. As a result, the economics of SOBA improve over time, as companies build and reuse an increasing number of Services.

It may even be possible to shift the 70% now spent on integration and existing application maintenance to new SOBA development. The returns companies can realize in this asset reuse scenario for SOA are not simply the cost reduction of simplified integration, but also improved time-to-market, more responsive customer service, reduction in overall staffing, and a greater ability to outsource or offshore Service creation, implementation, and even composition. Any ROI calculation that leverages the asset reuse benefits of SOA must therefore take into account all of these parts of the SOA value proposition.

"The best thing we can do to embrace the value of SOA is to get more value from the systems and investments we've already made," explains Joe Gentry, Software AG's vice president of Enterprise Transaction Systems.

While increased business agility is one of the most important business motivations for Service Orientation, reuse of existing assets is equally powerful. By building inherently flexible systems, SOA enables organizations to build composable Services that they can reuse in new business process scenarios. As a result, companies get more value out of their legacy systems because it is more practical to keep such systems around for longer. Even more powerful than the idea of obtaining flexibility is the notion that a business can achieve true reuse of its assets through SOA. The whole idea of reuse is to get more value from what has originally been built. As such, the ideas of reuse and legacy are really one and the same thing! How can a company even hope to achieve any aspect of reuse if they are continually rebuilding their existing capabilities? Indeed, true reuse demands the continuation of legacy!

As Software AG's Gentry says, "Reusability has long been a goal of application development for the last 30 years. With SOA this goal of reusability becomes much easier to attain. Organizations can now benefit from tremendous business agility, simplified maintenance requirements and an increased return on their investments in valuable legacy systems that Service-orientation enables."

Organizations spend precious little of their time and budget building new applications. Part of the reason for the lack of emphasis on new application development is the fact that they must create each new application in isolation from the previous applications they've built, resulting in a new piece of the IT puzzle that they must integrate in turn with other components. It's clearly important to build new applications in such a way that not only reduces the cost of development, but also maintenance, over time. With Service Orientation we can keep the old stuff around longer, but the new stuff we build is flexible enough never to become old stuff.

Avoiding New Troubles in SOA

Since SOA abstracts the underlying implementation of a Service, the primary assets in a Service-oriented environment are metadata – lots of metadata. Metadata define Service capabilities through contracts, control Service behavior through policies, and compose Services together through business process and semantic metadata. As such, a Service-Oriented Enterprise risks turning metadata into a new kind of legacy, even more so than the underlying implementations of the Service. Companies will continue to invest in processes and metadata until they no longer need them or their costs outweigh the benefits of their use.

Many Service-oriented organizations might indeed continue investing in legacy processes beyond their anticipated shelf-life, shifting the idea of legacy from the underlying systems to the processes themselves. But then in this case, we're dealing with a different concept of legacy. Rather than the systems or implementations becoming legacy, it's the processes and metadata that may become difficult to change, if not technically, than at least from a human behavior perspective.

"Governance is key to avoiding the pitfall of turning today's agile systems into tomorrow's maintenance nightmare," says Software AG's Totev. "Correct management of the full Service lifecycle – including people, policies, processes, and organizational change will free companies from simply re-encoding inflexible logic into metadata. Simply governing Services is not enough; organizations need policy, process, and metadata reuse just as much as service reuse. Therefore, companies must implement comprehensive governance technologies, processes, and organizations in order to avoid this potential rat's nest."

Governance enables companies to avoid creating new problems in their Service-oriented enterprise – the same sort of problems they had attempted to solve in the first place with SOA. Companies want to benefit from their legacy, but not at the expense of creating new integration and change management headaches. They want stability without brittleness, and flexibility without chaos.

Legacy systems are of tremendous value to the organization, but interacting with them comes at such cost and with such complexity that little remains of this value to the organization. SOA promises to turn this problem on its head. One of the most powerful concepts is the notion that a business can achieve true reuse of its assets through SOA, and thus achieve the flexibility and agility it so desires. For most organizations, then, successful SOA initiatives will include the ongoing preservation of legacy for the foreseeable future. And this preservation transforms the very concept of legacy.

Profile: Software AG	September 2006
Funding: Trades on Frankfurt Stock Exchange	
CEO: Karl-Heinz Streibich	
Employees: 2700	
Address:	
US Headquarters:	
11700 Plaza America Drive	
Suite 700	
Reston, Virginia 20190	
Worldwide headquarters:	
Uhlandstrasse 12	
64297 Darmstadt	
Germany	
URL: http://www.softwareag.com	
Phone: 877-724-4965 / +49-6151-92-0	
Contact: info@softwareag.com	

Related Research

- *Software AG ZapNote (ZTZN-1156)*
- *NetManage ZapNote (ZTZN-1174)*
- *Seagull Software ZapNote (ZTZN-0160)*



This ZapNote was commissioned by Software AG



About ZapThink, LLC

ZapThink is an IT advisory and analysis firm that provides trusted advice and critical insight into the architectural and organizational changes brought about by the movement to XML, Web Services, and Service Orientation. We provide our three target audiences of IT vendors, service providers and end-users a clear roadmap for standards-based, loosely coupled distributed computing – a vision of IT meeting the needs of the agile business.

ZapThink helps its customers in three ways: by helping companies understand IT products and services in the context of Service-Oriented Architecture (SOA) and the vision of Service Orientation, by providing guidance into emerging best practices for Web Services and SOA adoption, and by bringing together all our audiences into a network that provides business value and expertise to each member of the network.

ZapThink provides market intelligence to IT vendors and professional services firms that offer XML and Web Services-based products and services in order to help them understand their competitive landscape, plan their product roadmaps, and communicate their value proposition to their customers within the context of Service Orientation.

ZapThink provides guidance and expertise to professional services firms to help them grow and innovate their services as well as promote their capabilities to end-users and vendors looking to grow their businesses.

ZapThink also provides implementation intelligence to IT users who are seeking guidance and clarity into the best practices for planning and implementing SOA, including how to assemble the available products and services into a coherent plan.

ZapThink's senior analysts are widely regarded as the "go to analysts" for XML, Web Services, and SOA by vendors, end-users, and the press. Respected for their candid, insightful opinions, they are in great demand as speakers, and have presented at conferences and industry events around the world. They are among the most quoted industry analysts in the IT industry, and their recent book, *Service Orient or Be Doomed!*, is the leading book on the business concept of Service Orientation.

ZapThink was founded in October 2000 and is headquartered in Baltimore, Maryland. Its customers include Global 1000 firms and government organizations, as well as many emerging businesses. Its analysts have worked at such firms as IDC, marchFIRST, and ChannelWave, and have sat on the working group committees for standards bodies such as RosettaNet, UDDI, and ebXML.

Call, email, or visit the ZapThink Web site to learn more about how ZapThink can help you to better understand how SOA will impact your business or organization.

ZAPTHINK CONTACT:

ZapThink, LLC
108 Woodlawn Road
Baltimore, MD 21210
Phone: +1 (781) 207 0203
Fax: +1 (786) 524 3186
info@zapthink.com
www.zapthink.com