



IBM Software Group

# IBM WebSphere Application Server V7.0 Feature Pack for Service Component Architecture V1.0.1

## *SCA feature pack introduction*



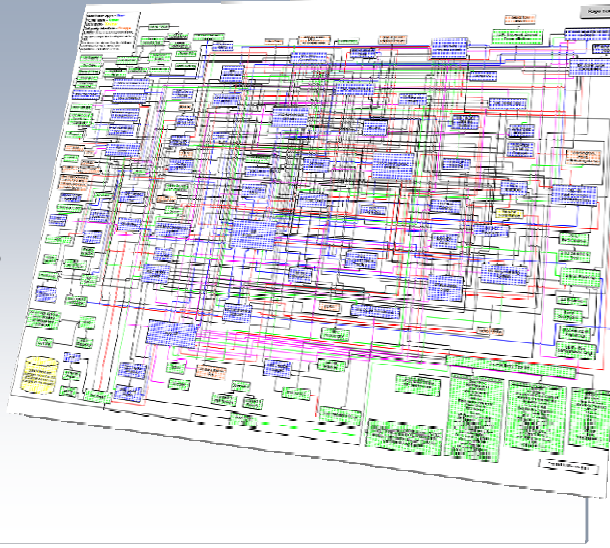
@business on demand.

© 2009 IBM Corporation  
Updated November 2, 2009

This presentation will cover the introduction of Service Component Architecture.

## What are the barriers to business flexibility and reuse

- **Lack of business process standards**
- **Architectural policy limited**
- **Point application buys to support redundant LOB needs**
- **Infrastructure built with no roadmap**



To introduce SCA, this section will address the need for flexibility and reuse and some of the barriers that sometimes prevent companies from getting to where they want to be. Business flexibility requires IT flexibility. IT flexibility is very difficult and expensive with today's systems that look like the diagram. Businesses can only be as flexible as the IT systems that support them. On the right, you'll see an actual application architecture for a consumer electronics company. Current approaches to IT architecture do not support these drivers.

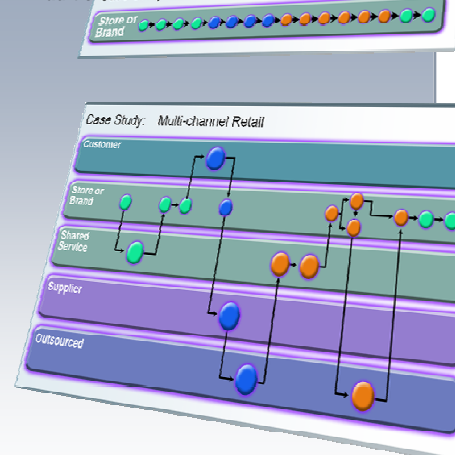
The text on the left talks about IT infrastructure that has grown bit by bit, over the years to handle focused issues with no recognizable roadmap. Linkages between pieces of this infrastructure tend to be inflexible and very difficult, expensive, and time-consuming to change.

## Why SOA for business flexibility and reuse?

### Flexible business requires flexible IT

- **Economics: globalization demands greater flexibility**
- **Business processes: daily changes versus yearly changes**
- **Growth through flexibility is at the top of the CEO agenda**
- **Reusable assets can cut costs by up to 20%**
- **Crucial for flexibility and becoming an On Demand Business**

### Tradition Case Study: Multi-channel Retail



Today's World-Class Business\*

\*Sources: CBDi



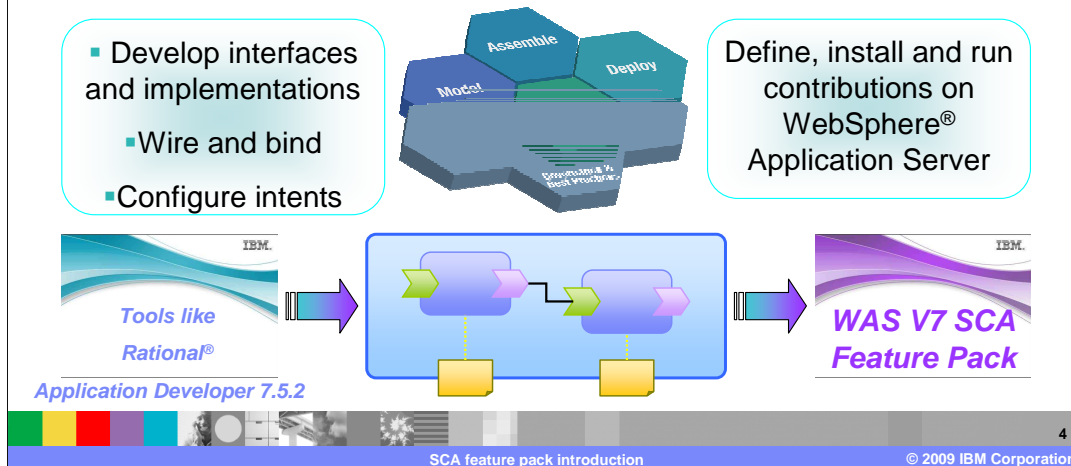
This slide addresses some of the reasons why businesses are taking steps to increase their flexibility and reuse through SOA.

If you look at the graphic at the right, what you'll see is what a sample business process (in this case, multi-channel retail) looked like in a traditional business compared to today's world-class businesses. In the past, you saw companies with very linear business processes that were handled by an individual department within a company. As sophistication increased, you saw the same business processes being broken up and pieces of it being performed in different places. In this example, customers placed orders directly through the Web, shared services in different parts of the company like merchandising or supply chain took over steps of the process. Suppliers contributed vendor managed inventory; shipping was outsourced, and so on and so forth. This kind of desegregation takes a lot of flexibility to establish and even more to change once it's established. It's important to remember that the business process in this example and very nearly every other business process in today's companies are very dependent on the information technology (or IT) systems that support them. Flexible business requires flexible IT.

Economics is a force driving the need for this kind of flexibility. As the marketplace globalizes, new markets, new workforces, and new competitors are making companies look for ways to adapt more quickly. The cycle time has been observed to be shrinking between changes in business processes. While you might have seen companies make significant changes yearly in the past, you'll see the same level of change on a monthly or even weekly or daily basis. While business leaders were focused mostly on cost-containment in the past, we're seeing that growth is back at the top of the today's CEO's agenda and that growth demands the flexibility to be more nimble than competitors. This isn't to say that cost reduction has lost its importance. On the contrary, businesses are looking for ways to make better use of the investments that they already have. IBM's experience shows that companies are seeing savings of about 20% through re-use. Other sources put that figure even higher. Finally SOA and the flexibility it brings is crucial for becoming what IBM calls an on-demand business.

## Open service oriented architecture

*An open, emerging standard programming model  
for assembling flexible SOA business solutions  
from diverse, reusable service enabled IT assets*



Now that you have seen some background material on the need for flexibility in business, the next couple of slides will introduce SCA.

SCA is an open (OSOA), emerging (OASIS) standard programming model. It's an assembly of flexible business solutions following SOA principles. It's an exploitation of diverse, reusable IT assets.

Service component Architecture (SCA) defines a 'simple' service-based model for construction, assembly and deployment of services (existing and new ones). Service component architecture is a set of specifications which describe a model for building applications and systems using a Service-Oriented architecture. SCA extends and complements prior approaches to implementing services, and SCA builds on open standards such as Web services. SCA is the central programming model of the SOA foundation and an open, leading industry standard for service composition.

## What is SCA?

- A concrete manifestation of an SOA way of thinking
- Designed for building agile service oriented applications
- A framework for implementing, assembling, composing and deploying services
- Supports loose coupling of coarse grained services
- Extends, exploits and complements existing technologies and standards
- Language, Application Environment, Framework and Vendor neutral
- Supports Java™ and Web services, and more



SCA stands for Service Component Architecture as indicated earlier. SCA is a concrete manifestation of an SOA way of thinking. It's designed for building agile service oriented applications. SCA is a framework for implementing, assembling, composing and deploying services and supports loose coupling of coarse grained services. It extends, exploits and complements existing technologies and standards and it is language, application environment, framework and vendor neutral. SCA supports Java and Web services, and more.

## What is SCA (continued)

- An extensible set of:
  - ▶ Protocol bindings (eg. SCA, WS, RMI, ... )
  - ▶ Implementation languages (e.g: Java)
  - ▶ Interface definitions (eg. WSDL, Java, ... )
  - ▶ Pluggable Data bindings (eg. PoJo, JAXB, SDO...)
  - ▶ Policies and Intents (eg. Integrity, Confidentiality).
- “Classic SCA”
  - ▶ Service Component Architecture as defined and built by IBM supported in a variety of WebSphere Family products starting with V6
- “Open SCA”
  - ▶ Service Component Architecture as defined by the industry at both the OSOA collaboration



SCA is also an extensible set of protocol bindings (for example: SCA, WS, RMI ) and implementation languages such as Java. It's also an extensible set of interface definitions, pluggable data bindings and policies and intents. There are two types of SCA infrastructure; classic SCA and open SCA.

“Classic SCA” refers to Service Component Architecture as it is defined and built by IBM. Classic SCA was first introduced by IBM and other vendors in 2005 and it's what is known as SCA 0.5. It's supported in a variety of WebSphere family products starting with WebSphere Application Server V6.

“Open SCA” refers to Service Component Architecture as defined by the industry at both the OSOA collaboration which was released to OASIS for standardization in 2007. It's what is referred to as SCA 1.0. The current SCA feature pack is based on open SCA.

As an example, the SCA implementation in WebSphere Process Server and WebSphere Enterprise Service Bus V6 is based on SCA 0.5 (Classic SCA). Alternatively, WebSphere Application Server V7 Feature Pack for SCA is based on SCA 1.0 (open SCA).

So how do you integrate the two if one has WebSphere process server applications that they want to run on SCA feature pack? The solution to this is through Web services binding. Customers who want to create SCA 1.0 composites on WebSphere Application Server that wire to Classic SCA services on WebSphere Enterprise Service Bus or WebSphere Process Server can do so through the Web services binding.

## A brief history of SCA



Looking at a brief history of SCA, Service Component Architecture is not new – WebSphere Process Server pioneered SCA delivering a precursor to the open version in its V6.0 release of runtime and tools. This pioneering technology called “Classic SCA,” was used by IBM to seed open specifications endorsed by several vendors at the Open Service Oriented Architecture organization. As you saw earlier, SCA represents the SOA programming model of the IBM SOA Foundation. It is a strategic technology of the IBM application server and BPM software portfolio. The Initial SCA code contribution and collaboration started back in December 2003 with contribution from BEA and IBM.

The real SCA Project was created in December 2005 in Apache incubator and this version was (SCA 0.9). Next came SCA version 0.95 OSOA which was a collaboration and an interim delivery implemented by Tuscany and WebSphere Application Server version 6.1 SOA Feature Pack Beta1. The next release SCA version 1.0 came in March 2007 which was an OSOA (Open Service Component Architecture) Collaboration and implemented by Tuscany. This is the version that SCA feature pack is implemented on mentioned earlier.

## Nature and benefits of Open SCA (1)

- Open, multi-vendor programming model designed for SOA
- Emerging standard through OASIS
- Extends, exploits and complements existing standards
- Unifying framework for assembling services from diverse assets
- Formal representation of services and their interdependencies
- Embraces heterogeneity – Strength and agility through diversity
- Rapid integration of components into applications
- Loose Coupling of coarse or fine grained components

The benefits of SCA Feature pack include the ability to perform basic service composition in new POJO-based SCA applications. (Reuse). Another benefit is to develop and deploy services that use SCA natively to simplify access from different application types. (Reuse, Connectivity)

Business pressures from competition, marketplace, mergers and acquisitions, and so on requires applications to adapt to rapid change.

SCA applications are resilient to change and are insulated from technology and infrastructural changes. SCA is architected to accommodate new technologies as they emerge.



## Nature and benefits of open SCA (2)

- Flexible assembly model – Swap implementations in or out
- Separation of concerns in multiple dimensions
- Diverse, extensible implementations, bindings and data
- Inversion of Control through dependency injection and reflection
- Asynchrony built into the programming model
- Declarative abstract Intents and concrete Policy
- Simplification for developers, integrators and deployers



Other open SCA benefits include flexibility in the assembly model and separation of concerns in multiple dimensions

Open SCA also offers diverse, extensible implementations, bindings and data, inversion of control through dependency injection and reflection and asynchrony built into the programming model. It also provides declarative abstract intents and concrete policy

Simplification for developers, integrators and deployers.

## What SCA is NOT

- SCA does not choreograph services or model workflows
- SCA is not Web services
- SCA is not tied to a specific runtime environment
- SCA does not force specific languages and technologies



Now that you have learned what SCA is, you will now learn what it is not. SCA does not choreograph services or model workflows, instead BPEL or other workflow languages can be used. SCA is not Web services. Web services are a binding technology to SCA. WSDL is an interface description language to SCA. SCA is not tied to a specific runtime environment. It can be applied to any kind of environment – Distributed, heterogeneous, large or small. SCA does not force specific languages and technologies. It's open to many languages, frameworks, technologies. It embraces, not replaces, assets and other frameworks and it engages new technology.

## Section

# *Summary and references*



The next slide provides a summary of this presentation.

## Summary

- Business applications are required to adapt to rapid change
  - ▶ SCA applications are resilient to change
  - ▶ Insulated from technology and infrastructural changes
  - ▶ Architected to accommodate new technologies as they emerge



Business pressures from competition, marketplace, mergers and acquisitions, and so on. requires applications to adapt to rapid change.

SCA applications are resilient to change. They are also insulated from technology and infrastructural changes. SCA is also architected to accommodate new technologies as they emerge.

## Resources

- Open Service Oriented Architecture Web site for SCA v1.0 Specifications



▶ <http://www.osoa.org/>

- OASIS Open CSA Web site for SCA v1.0



▶ <http://www.oasis-opencsa.org/sca>

- Apache Tuscany Web site



▶ <http://incubator.apache.org/tuscany/>

- SCA feature pack support Web site



<http://www-01.ibm.com/support/docview.wss?rs=180&context=SSEQTP&dc=DB600&uid=swg21329175>

Listed here are some resources that you might find valuable.

## More resources

- DeveloperWorks

- ▶ <http://www.ibm.com/developerworks/websphere>

- Exploring the WebSphere application Server feature pack for SCA

- [http://www.ibm.com/developerworks/websphere/library/techarticles/0812\\_beck/0812\\_beck.html](http://www.ibm.com/developerworks/websphere/library/techarticles/0812_beck/0812_beck.html)

- WebSphere Application Server V7 service component architecture FAQs

- <http://www-01.ibm.com/software/webservers/appserv/was/featurepacks/sca/faq.html>



More resources.

## Feedback

### Your feedback is valuable

You can help improve the quality of IBM Education Assistant content to better meet your needs by providing feedback.

- Did you find this module useful?
- Did it help you solve a problem or answer a question?
- Do you have suggestions for improvements?

Click to send e-mail feedback:

[mailto:iea@us.ibm.com?subject=Feedback\\_about\\_WASv7SCA101\\_Overview\\_SCAfepIntroduction.ppt](mailto:iea@us.ibm.com?subject=Feedback_about_WASv7SCA101_Overview_SCAfepIntroduction.ppt)

This module is also available in PDF format at: [../WASv7SCA101\\_Overview\\_SCAfepIntroduction.pdf](mailto:iea@us.ibm.com?subject=Feedback_about_WASv7SCA101_Overview_SCAfepIntroduction.ppt)



You can help improve the quality of IBM Education Assistant content by providing feedback.

## Trademarks, copyrights, and disclaimers

IBM, the IBM logo, ibm.com, and the following terms are trademarks or registered trademarks of International Business Machines Corporation in the United States, other countries, or both:  
Rational                      WebSphere

If these and other IBM trademarked terms are marked on their first occurrence in this information with a trademark symbol (® or ™), these symbols indicate U.S. registered or common law trademarks owned by IBM at the time this information was published. Such trademarks may also be registered or common law trademarks in other countries. A current list of other IBM trademarks is available on the Web at "Copyright and trademark information" at <http://www.ibm.com/legal/copytrade.shtml>

Rational is a trademark of International Business Machines Corporation and Rational Software Corporation in the United States, Other Countries, or both.

Java and all Java-based trademarks and logos are trademarks of Sun Microsystems, Inc. in the United States, other countries, or both.

Other company, product, or service names may be trademarks or service marks of others.

Product data has been reviewed for accuracy as of the date of initial publication. Product data is subject to change without notice. This document could include technical inaccuracies or typographical errors. IBM may make improvements or changes in the products or programs described herein at any time without notice. Any statements regarding IBM's future direction and intent are subject to change or withdrawal without notice, and represent goals and objectives only. References in this document to IBM products, programs, or services does not imply that IBM intends to make such products, programs or services available in all countries in which IBM operates or does business. Any reference to an IBM Program Product in this document is not intended to state or imply that only that program product may be used. Any functionally equivalent program, that does not infringe IBM's intellectual property rights, may be used instead.

THE INFORMATION PROVIDED IN THIS DOCUMENT IS DISTRIBUTED "AS IS" WITHOUT ANY WARRANTY, EITHER EXPRESS OR IMPLIED. IBM EXPRESSLY DISCLAIMS ANY WARRANTIES OF MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE OR NONINFRINGEMENT. IBM shall have no responsibility to update this information. IBM products are warranted, if at all, according to the terms and conditions of the agreements (for example, IBM Customer Agreement, Statement of Limited Warranty, International Program License Agreement, etc.) under which they are provided. Information concerning non-IBM products was obtained from the suppliers of those products, their published announcements or other publicly available sources. IBM has not tested those products in connection with this publication and cannot confirm the accuracy of performance, compatibility or any other claims related to non-IBM products.

IBM makes no representations or warranties, express or implied, regarding non-IBM products and services.

The provision of the information contained herein is not intended to, and does not, grant any right or license under any IBM patents or copyrights. Inquiries regarding patent or copyright licenses should be made, in writing, to:

IBM Director of Licensing  
IBM Corporation  
North Castle Drive  
Armonk, NY 10504-1785  
U.S.A.

Performance is based on measurements and projections using standard IBM benchmarks in a controlled environment. All customer examples described are presented as illustrations of how those customers have used IBM products and the results they may have achieved. The actual throughput or performance that any user will experience will vary depending upon considerations such as the amount of multiprogramming in the user's job stream, the I/O configuration, the storage configuration, and the workload processed. Therefore, no assurance can be given that an individual user will achieve throughput or performance improvements equivalent to the ratios stated here.

© Copyright International Business Machines Corporation 2009. All rights reserved.

Note to U.S. Government Users - Documentation related to restricted rights-Use, duplication or disclosure is subject to restrictions set forth in GSA ADP Schedule Contract and IBM Corp.

