

CBDI Report

Extending SOA with Web 2.0

Abstract

This paper describes some opportunities to extend SOA with Web 2.0, and outlines some of the IBM offerings that support these opportunities. This paper may be of interest both to IT managers who are tasked with meeting demands to make enterprise systems more responsive to new requirements and easier to access, and also line-of-business users looking to address new markets and interact better with their customers and partners.

Contents

Overview	1
Requirements	3
Extending SOA with Web 2.0	6
The Need for Service Architecture.....	9
Enabling Tools and Technology	12
Authors.....	14



Independent insight for Service Oriented Practice

Overview

This paper describes some opportunities for extending SOA with Web 2.0, and outlines some of the IBM offerings that support these opportunities. This paper may be of interest both to IT managers who are tasked with meeting demands to make enterprise systems more responsive to new requirements and easier to access, and also line-of-business users looking to address new markets and interact better with their customers and partners.

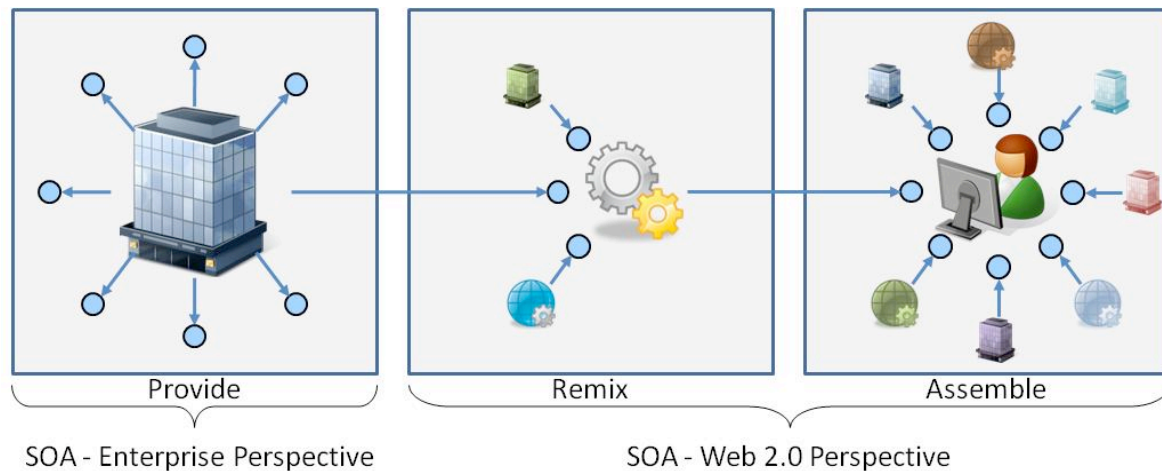


Figure 1 – SOA and Web 2.0 Perspectives

As Figure 1 illustrates, SOA and Web 2.0 can be seen as a continuum, where

Provide	<ul style="list-style-type: none"> ○ SOA is designed to make information more easily accessible via services, and to a wider audience both within and outside the enterprise
Remix	<ul style="list-style-type: none"> ○ Web 2.0 helps add value by combining enterprise services and web feeds together to help deliver richer information
Assemble	<ul style="list-style-type: none"> ○ Web 2.0 is designed to make it easier to assemble interactive solutions using services from many different sources

In the first part of this paper we focus on the remix and assemble perspectives, and explore the value of Web 2.0. In the second part, we look at the provide perspective and examine how to extend SOA with Web 2.0. Finally we consider how IBM offerings support each of these perspectives.

What is Web 2.0?

The term Web 2.0 refers to new modes of user-driven Internet activity, including networking, collaboration and user-driven content. Many of the elements of Web 2.0 are highly relevant to service-oriented architecture and the service economy. We can identify four sets in particular:

Loosely coupled systems of systems	<ul style="list-style-type: none"> ○ Small pieces loosely joined – decoupling data from functionality over the internet – sometimes called “web as platform “ ○ Granular addressability of content ○ Software above the level of a single device ○ Radical decentralization
Remixability	<ul style="list-style-type: none"> ○ Repurposing services and other assets ○ Unlocking valuable business data
User-centric	<ul style="list-style-type: none"> ○ Users control their own data ○ Rich user experience, Rich Internet Applications (RIA), Ajax ○ Trust your users ○ Emergent - user behavior not predetermined ○ Customer self-service - enabling the long tail
Enterprise social software	<ul style="list-style-type: none"> ○ Networking and ad hoc computing within the enterprise.

Context	Provide	Remix	Assembly
SOA offers	Delivery of information and system solutions to internal and external users in the form of services	Availability of data to remix into new applications/services Loosely coupled abstractions to support repurposing and remixing	Consumption of third party information and system solutions (e.g. packaged applications) in the form of services
Web 2.0 offers	Extra sources of information and services Management and rendering of user content	Light-weight access protocols for SOA services	Extra channels for delivering enterprise content to consumers Channels for peer-to-peer collaboration between users

Table 1 – Context

Requirements

Collaboration → dynamic assembly → leveraging information

Business drivers

The solutions described in this report are driven by a combination of forces – both from the provider side (service providers, platform providers) and from the user side.

Provider side

Economics of scale and scope

On the provider side, organizations are generally looking for cost-effective ways to help increase the agility, power and reach of their systems and services, and provide greater value to their users, both internally and externally.

The concept of “**power to the edge**” refers to changing the way individuals, organizations, and systems relate to one another and work together.

- empowerment of individuals at the edge of an organization
- adoption of an edge organization, with greatly enhanced peer-to-peer interactions.
- moving senior personnel into roles that place them at the edge

One area of particular interest to many organizations is market expansion – developing new market opportunities (products, market sectors, channels), while finding rapid and low-cost ways of testing and exploring these opportunities, thus helping to reduce time-to-market and risk, and helping enable the user to consider a larger number of opportunities.

Organizations may also be interested in leveraging communities – for example, providing services to diverse communities and their members, encouraging communities to emerge as informal distribution channels for their products and services.

User-side

Greater functionality through remix and assembly

On the user side, there is an increasing awareness of what Web 2.0 can offer the user. Users often encounter new functionality in the consumer world before this functionality is adopted by corporate IT. Customers and employees may use a wide range of web functionality in their lives outside work, and may belong to a number of web-based communities. Examples include blogging and self-publication (Blogger, MySpace), content sharing (Flickr, YouTube), social networking (FaceBook, LinkedIn), tagging (Digg, del.icio.us), and a large range of simple web application hybrids or “mashups” (many of them involving Google Maps).

Users, both inside the enterprise and outside, are often time-challenged and information-overloaded people driven by fast changing business needs. They therefore may increasingly need this kind of functionality in their web-based dealings.

Role	Type of Interaction	Possible Interest
employees	interacting with colleagues across the enterprise	developing situational apps for short term or long tail,
consumers	interacting with financial, retail, media, and other services	obtaining “joined-up” services – services that are consistent and effective from the consumer side as well as from the producer side
professionals (e.g. doctors, lawyers)	interacting with their peers in relation to a complex network of organizations (e.g. hospitals, drug companies, police, prisons)	acting on behalf of their clients
citizens	interacting with a wide range of government and voluntary agencies	obtaining “joined-up” services
members of communities	carrying out a range of collective activities	communities wanting to build their own solutions
entrepreneurs	interacting with the Internet marketplace	looking to create the “next big thing”

Table 2 – Demand by Role

Large organizations, whether public, private or voluntary, may be increasingly driven to respond to these expectations – and particularly if their competitors are already doing so.

Opportunities for Innovation

Mode of Innovation	User	Community	Collaborative
WHY	Leverage and extend existing systems <ul style="list-style-type: none"> ○ to new markets ○ to new users 	Emergent Community New communities emerge from shared content and revealed affinities.	Collaborative Governance Negotiation of relationships with diverse partners. Leverage Third Party Innovation
WHAT	User-Generated Content Self-publication, individual blogs combined with corporate/supplier /enterprise information	Collective Content Wikis, interlinked blogs Community specific products/services ready for mashing	Situated Software Software designed in and for a particular social situation or context. Web Application Hybrid (Mashup) Assembly of third party services

Table 3 – Opportunities for Innovation

HOW	<p>Rapid Development</p> <p>Use of Web 2.0 tools to create rich internet applications.</p> <p>User-Driven Computing</p> <p>End-user computing is not a new idea, but Web 2.0 is designed to create new opportunities for employees, customers and partners to assemble their own solutions.</p>	<p>Social Computing</p> <p>Encourages the creation of new communities.</p> <p>Community Driven Computing</p> <p>Web 2.0 Ajax development helps enable IT to rapidly create situational applications for communities and groups of users</p>	<p>Collaborative Composition</p> <p>Planning at the edge between multiple design authorities</p> <p>Open Service Platform</p> <p>Rendering enterprise content in a form that can be repurposed and repackaged by third parties.</p>
------------	---	---	---

Table 3 – Opportunities for Innovation (continued)

Economic viability

Economic principles such as **network effect** and the **long tail** play a role in how Web 2.0 may impact your business.

Network Effect	<ul style="list-style-type: none"> ○ Value of a service in one place depends on the amount of usage elsewhere – the more the better. ○ Especially true of services that offer some form of communication – the value of the service depends partly on the number of other people you can communicate with. ○ Positive feedback loop – as more people use the service, it may increase in value, thus persuading more people to use the service. This feedback loop may result in a rapid growth curve. ○ Critical mass – the level at which the growth curve becomes self-sustaining.
Long Tail	<ul style="list-style-type: none"> ○ The traditional Pareto principle (80% of the value comes from 20% of the cases) assumes a significant proportion of fixed costs. For example, a bricks-and-mortar bookshop must devote shelf space to any book in stock, and there is a cost associated with excess inventory. ○ But in a service-based economy, it may be possible to alter the supply-side economics – for example shifting from fixed costs to variable costs – and this means that the rare items (both individually and in aggregate) could be as profitable as the popular items. ○ The economic viability of the long tail depends critically on bringing down the fixed costs of servicing the long tail, and shifting to a more flexible variable-cost regime.

Web 2.0 and SOA together can help provide a cost-effective way of accessing these network effects and long-tail benefits.

- Cost reduction – leveraging existing systems through services, helping to cut cost of customization
- Cost mitigation – providing services not solutions, shifting responsibilities to collaborating partners. For example, customer self-service, small communities providing their own solutions. Thus collaboration helps spread the cost.
- Shift from fixed cost to variable cost – for example through “pay-as-you-go” on-demand pricing – thus helping to reduce the risk and the barriers to adoption.

At the same time, IT organizations are also often looking for ways to improve the delivery of solutions to their business users. Web 2.0 technologies and approaches may help reduce developer effort and time to solution, as well as providing more agile solutions that can be rapidly remixed and re-assembled to meet changing demands.

Challenges/concerns

In order to achieve these Web 2.0 benefits, there are some questions that need to be answered:

- What are the implications for IT management? How can I guarantee the continued security of my core systems? How can I provide the necessary quality of service to my existing users as well as the new ones?
- What are the implications for business management? How can I guarantee the continued integrity of my products and services? If third parties are able to mash my services with external services, how can I retain control of my brand and corporate image?
- What are the implications for corporate governance and compliance? Do these external arrangements introduce new risks?

Clearly many organizations will not be prepared to go ahead with Web 2.0 without good answers to questions like these. This is why it will be important to have a well-structured approach to Web 2.0 and SOA.

Extending SOA with Web 2.0

Decoupling and abstraction → ubiquity and grid → power-to-the-edge

Provide - Unlocking Information and Processes for Web 2.0

In many systems today there are technical and behavioral dependences between resources placing constraints on both the service provider and service consumer’s ability to change resources, or respond to new opportunities. A key principle of SOA is to decouple the service

provider and service consumer's resources to help remove these constraints and improve flexibility.

Today, solutions are normally assembled to address a specific business requirement, such as automating a business process. However, whilst the solution might subsequently be made accessible to a wide community both internal and external via a web portal, the solution is still limited in terms of the behavior it offers. Users see the solution, not the underlying services, and the ability to respond to different user requirements is limited to what is built into the solution. Hence a key aspect of decoupling is to also separate service provision from solution assembly.

Most organizations look to SOA as a way to help them better leverage their existing enterprise systems and data stores by making it easier to access them in new business solutions. At the same time, decoupling existing systems from the new solutions helps provide greater flexibility, both in the systems that provide the services, and in the business processes that use them.

SOA therefore is designed to enable Web 2.0 through the following features:

Easy to Access	<ul style="list-style-type: none"> ○ It is simpler to access resources that are provided as services. All the service consumer needs to know is the address of the service, the information they must provide, and what they will get in return. For example, simply send a product code to http://anystuff.com/stocklevelservice and get a stock quantity in return. ○ REST (Representational State Transfer) in particular is a simple protocol to use, though Web 2.0 tooling hides complexity in using services presented in the protocols referenced earlier. ○ Services provided as a façade across enterprise systems may also be made available for use in Web 2.0 scenarios. The SOA infrastructure may provide access to real-time operational data that is normally locked away inside internal systems, yet at the same time still making it easy to use.
Decoupling	<ul style="list-style-type: none"> ○ SOA encourages the separation of service provision from solution assembly to help provide greater flexibility ○ Web 2.0 users may more easily access enterprise systems, but are not tightly coupled to them
Quality of Service	<ul style="list-style-type: none"> ○ The investment in the SOA infrastructure to enable these services may also be used to help ensure that quality of service (QoS) and security of those enterprise systems is not compromised by exposure to Web 2.0 – or conversely, this may be seen as bringing enterprise levels of QoS and security to Web 2.0.

Remix and Assemble - Extending SOA with Web 2.0

The same properties of services that make them easy to access also make them easy to remix. That is, information retrieved from services provided by many different organizations may be mashed together and re-presented as a new service to provide a solution that was never

envisaged by the original service providers. For example, the now familiar scenario of an address provided by one service used to retrieve the relevant map provided by another, simply by mapping data output from one service to the input required by the other.

So, not only can users personalize their solution by assembling information from many sources, but they can also personalize the services and pass them on within their community to others with similar needs but who still wish to personalize their own solution.

Though Web 2.0 is often seen as enabling end-users and customers to remix and assemble their own solutions, where appropriate IT developers may also use these technologies themselves to help speed delivery of applications they are building on behalf of the business.

Consequently, Web 2.0 may be seen as extending the value of SOA investments. For example

Leveraging SOA Investments	<ul style="list-style-type: none"> ○ Platform for business innovation and exploration – which may lead to new ways of exploiting SOA, thus providing an evolutionary path – for when short term, situation applications need to evolve into long term mission critical systems when a new market or opportunity takes off. This helps preserve the investment in service provision ○ New mechanisms for differentiation and integration, helping to enable the personalization and simplification of service oriented solutions. The broader applicability of the services may make the investment more attractive, for example on a per user basis <p>Collaborative composition - The overall cost of solution may be effectively shared between the service provider and Web 2.0 user, in that the enterprise isn't having to bear the whole cost of both providing services and assembling the solutions</p>
Extending Reach	<ul style="list-style-type: none"> ○ REST, ATOM, and RSS are designed to provide new, simpler ways to access SOA services, that are accessible by a wider audience
Easy to Remix	<ul style="list-style-type: none"> ○ Organizations can make their services themselves accessible via the web, not just their pre-assembled solutions. The Web 2.0 user may then further remix those services to help provide new capabilities ○ Web 2.0 technologies are designed to make it easier to add value by remixing enterprise services with web feeds and other information sources
Reduce IT Developer Effort	<ul style="list-style-type: none"> ○ Web 2.0 technologies may also be used by IT developers to help rapidly assemble solutions for their business users. Where appropriate, the services provided to support Web 2.0 may benefit IT developers by helping to reduce the coding required by Ajax scripting and simplify access through protocols such as REST

SOA and Web 2.0 - Working Together

The aggregation and remixing of services serves two different purposes:

- In the service provider's space the focus is on aggregating a service where many of the internal sources may not themselves be service enabled, and on delivering a more durable service that represents some core activity or information of the business
- In the Web 2.0 user's space the focus is on aggregating a service from many other services both internal and external, and on delivering a more dynamic service that may only have a short-term value

Normally organizations would think of the platforms as those which host the systems that lie behind the service façade. Now however, instead of looking at the Web as the solution, it may become a platform itself providing a world wide grid of virtualized resources that may be assembled and remixed into new solutions.

Thus we can see that extending SOA with Web 2.0 helps provide a stronger value proposition. For example

Extending SOA with Web 2.0 Value Proposition	<ul style="list-style-type: none"> ○ Services may be of more value to a wider community than pre-assembled solutions ○ New personalized and situational solutions may more easily access enterprise information and operational systems ○ The availability of services may establish the Web as a platform.
Ubiquity	<ul style="list-style-type: none"> ○ Both the service provider and the Web 2.0 user may benefit from the ubiquitous availability of protocols such as Web Services, REST, ATOM and RSS, helping to ensure that decoupling at the technology level is easy to achieve on any platform ○ Combined with the ubiquity of the web, services that use these protocols may be accessed by Web 2.0 users virtually anywhere in the world, with almost any type of device

The Need for Service Architecture

Strong architecture and design help further enhance the value of extending SOA with Web 2.0. Since new solutions that were not envisaged by the service provider may be created, then it may be necessary for the service provider to deliver services that are suitable for this goal. Many of the following aspects reflect SOA best practices regardless of Web 2.0.

Decoupling Considerations	<ul style="list-style-type: none"> ○ As well as using appropriate protocols to help remove technology dependencies, organizations also need to consider how they might minimize other dependencies between service provider and consumer, such as: <ul style="list-style-type: none"> ○ Semantic coupling that assumes common data formats. For example, using XML in preference to binary formats
----------------------------------	---

Decoupling Considerations (cont)

- Behavior coupling that assumes a common workflow or process. The service provider assumes the service consumer is following the same process, whereas in Web 2.0 they may have a totally different activity in mind.
- Goal alignment that assumes common goals. For example the service provider might want to commit changes at what they believe is the end of the process, whereas it might only be a small part of the broader service consumer's process.

Broadening Applicability

- So that services may be used in unforeseen solutions, as well as helping to ensure decoupling, service providers should also consider:
 - Granularity. A fine-grained, modular service architecture that is designed to provide for rapid configuration to help meet new requirements
 - Independence. There should be no dependencies between services other than via published service interfaces. Service A should not access service B's database directly.
 - Generalization. To help enable repurposing, services should be generalized.
- Rather than just taking the usual perspective of the services provided and consumed by their own enterprise, organizations should consider it from the Web 2.0 viewpoint where the user may have a very different view of a the solution that addresses their particular needs.

Simplifying Use

- To simply use in Web 2.0, services should work on the principle of "weak pre-conditions, strong post-conditions".
 - Rather than have rigid message structures, services should be designed with more "forgiving" interfaces that accept unstructured data. For example, instead of forcing an address to follow normal number, street, city, state, zip/postcode, country structure, and rejecting any input that doesn't match, a service could have a simple single address field and then parse the text.
 - This approach helps enable information to be mashed up more easily, and new service providers to be quickly included; whereas normally each service might have a different address structure, forcing the service consumer to painstakingly map them each time.
-

Layered Architecture	<ul style="list-style-type: none"> ○ SOA best practice is to use a layered service architecture where services are classified according to their specific role within the overall architecture. To provide flexibility and sharing, CBDI Forum recommends for example the separation of <ul style="list-style-type: none"> ○ process services that enable interaction with a process (e.g. order fulfillment) ○ from core business services that provide information that is recorded about a core business type (e.g. delivery) ○ Normally these services would then be consumed within solutions. For Web 2.0, organizations might choose to apply the service characteristics outlined previously to a separate layer of services specifically designed to address Web 2.0 users, or instead design them into the process and core business services directly so that no additional layer is required. For example, where services already exist, adding an additional layer might be preferable to redesigning the existing ones
Process Considerations	<ul style="list-style-type: none"> ○ Use of information modeling to provide a common basis for the services offered by an organization, and to understand differences between the demand-side and supply-side models ○ The delivery of full and precise service specifications, that are unambiguous, easy to comprehend, and ideally automatable. We believe that WSDL (Web Service Description Language) does not have constructs to provide the full understanding of the behavior a service offers, such as detailed pre-post conditions, and should be complemented with additional specifications. ○ The need for automated service publication and discovery, so that Web 2.0 users may easily locate services, and their tooling may automate the discovery and usage processes
Protocol considerations	<ul style="list-style-type: none"> ○ For Web 2.0, the use of ubiquitous web-based protocols such as Web Services, REST, ATOM and RSS. The protocols used may be determined by factors such as the level of complexity ○ Web Service protocols for transactions, security and reliable messaging make them a comprehensive solution for the delivery of complex federated services where information integrity between participants is essential. Even so, whilst tools greatly simplify their use, they can still be relatively complex to use. ○ For read-only requirements, ATOM and RSS though less comprehensive than Web Services, generally provide a simpler approach that has been more widely adopted by web developers for Web 2.0. ○ Organizations should consider providing bindings for multiple protocols if possible to help provide maximum scope for service usage. In this case, platforms that may expose many protocol variations of the same underlying business services may be key.

Enabling Tools and Technology

To understand the range of tools and technologies needed to support the combination of SOA and Web 2.0 Table 4 sets out some of the capabilities required to support the provide, remix and assemble perspectives.

Provide (P)	Remix (R)	Assemble (A)
Designed to enable services Access to existing Information and systems Service Bus for mediation (transformation and routing) and messaging	Combine, filter and enrich information from different sources such as services and Web feeds Provide new remixed services Ease of use	Consume services and Web Feeds Assemble solution Ease of use

Table 4 - Required Capabilities

Table 5 then maps relevant IBM tools to the perspectives (abbreviated to P/R/A). This isn't a full list of all tools that might be used to extend SOA with Web 2.0, but represents those that have been designed to have specific capability in the context of SOA and Web 2.0 together.

A core value proposition in this paper has been to use SOA as a mechanism to leverage existing application and information assets in new Web 2.0 scenarios. In addition to this, these products are also designed to enable the IBM user to leverage their existing investment in IBM infrastructure, which in turn provides access to their heterogeneous application portfolio. For example, a key role in these products is to provide a "gateway" between the world of service provision and the world of Web 2.0. Several IBM products are designed to provide capabilities in this layer. The choice of gateway may be determined by which products the organization already has. For example, an existing Tivoli® user would continue to use Tivoli software for many of the management capabilities required in the gateway, and a user with an investment in IBM® WebSphere® MQ software would likely use the HTTP bridge for the protocol conversion. Whilst a user without that investment might instead use IBM DataPower® software to provide a self-contained gateway that comes in a convenient appliance form-factor.

IBM WebSphere software consists of a comprehensive array of products designed to support SOA. WebSphere software offerings are designed to provide service hosting, orchestration, Enterprise Service Bus (ESB) capabilities, as well as integration with existing systems. IBM WebSphere software already provides comprehensive support for Web Service protocols, but as discussed earlier, it is valuable to also provide support for REST, ATOM and RSS protocols to help broaden service usage. So to further leverage this infrastructure, new IBM WebSphere extensions for Web 2.0 include

- IBM WebSphere Application Server Feature Pack for Web 2.0
 - <http://www.ibm.com/websphere/was/web20>
- IBM WebSphere DataPower Integration Appliance X150
 - <http://www-306.ibm.com/software/integration/datapower/xi50/>
- IBM WebSphere Commerce 6.0
 - http://www-306.ibm.com/software/info1/websphere/index.jsp?tab=landings/commerce_enhancements_v6

IBM Tools and New Web 2.0 Extensions	P	R	A
<p>IBM WebSphere MQ HTTP Bridge. As its name suggests, it acts as a bridge between the service provider using MQ as their internal messaging and integration backbone, and the typical Web 2.0 solution that does not have a MQ client. This is designed to enable Ajax and other HTTP clients to read and write messages to MQ queues using a REST programming style.</p>	✓		
<p>IBM WebSphere Application Server Web 2.0 Feature Pack. Designed to help simplify interoperability with Ajax clients. It includes,</p> <ul style="list-style-type: none"> • Ajax Messaging provides a publish/subscribe messaging service so a server may stream data to an Ajax client from a Service Bus. For example frequently updated data like stock quotes. <p>Web 2.0 to SOA Connectivity that is designed to provide simplified access to SOA services and</p> <ul style="list-style-type: none"> • Java™ 2 Platform, Enterprise Edition (J2EE) assets through REST, ATOM, and RSS. Including an Ajax Proxy that helps to eliminate browser security concerns when combining internal and external services. • Ajax Development Toolkit based on Dojo, an Open Source JavaScript toolkit, with IBM extensions for ATOM and SOAP Web Services. Provided for the enterprise developer who needs to add the rich user experiences provided by Ajax clients to existing applications. 	✓	✓	✓
<p>IBM WebSphere DataPower. A range of appliances may be used by organizations as an effective SOA gateway for XML and Web Services, providing security, logging, XML acceleration and transformation of non-XML messages. New capabilities in the WebSphere DataPower Integration Appliance X150 are designed to enable it to leverage those capabilities when acting as gateway to Web 2.0.</p> <ul style="list-style-type: none"> • Web 2.0 to SOA Connectivity that bridges Ajax, REST and ATOM into service provider's underlying service architecture • Aggregation of ATOM and RSS web feeds 	✓		
<p>IBM WebSphere Commerce 6.0. Includes enhancements for both SOA and Web 2.0 with</p> <ul style="list-style-type: none"> • Adapters for Web Services plus reference applications and pre-defined Web Services for integration with ERP and other back-office systems. • Capabilities designed to support new models of commerce using Web 2.0 including support for ATOM and RSS to help improve partner integration, for example, and Ajax to help build a more interactive shopping experience 		✓	✓
<p>IBM WebSphere Portal Framework with SOA-based composite application includes support for</p> <ul style="list-style-type: none"> • Ajax and REST capabilities to help deliver rich internet applications • IBM Portlet for Google Gadgets for integration of Google Gadgets into portal-based mashups 		✓	✓
<p>IBM Mashup Starter Kit. Designed to enable organizations to easily remix and catalog a wide variety of data sources and content feeds. The aim is to enable line-of-business end-users to quickly "mash-up" their enterprise data sources as well as external web data and content. Consisting of,</p> <ul style="list-style-type: none"> • An interactive drag-and-drop Web editor that is designed to enable information sources such as XML, ATOM, RSS support, and Excel spreadsheets, to be remixed and published in new formats, as well support for other common Web 2.0 functions such as tagging or user ratings. • A catalog of the Web 2.0 resources such as the XML, Atom and RSS feeds, to foster reuse and collaboration. 		✓	✓
<p>IBM QEDWiki. A browser-based environment that is designed to allow users to quickly assemble new Web 2.0 solutions from different information services and feeds</p>			✓

Table 5 – Summary of IBM Support for SOA and Web 2.0 Perspectives

Authors

Lawrence Wilkes and Richard Veryard, CBDI Forum

lawrence.wilkes@cbdiforum.com, richard.veryard@cbdiforum.com

This report was commissioned by International Business Machines Corporation.

Trademarks:

DataPower, IBM, Tivoli, and WebSphere are registered trademarks of International Business Machines Corporation in the United States, other countries, or both.

Java and all Java-based trademarks 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.

DISCLAIMERS

This document was researched and written by CBDI Forum, UK, under a contract with International Business Machines Corporation. The analysis, views, conclusions, and recommendations in this document are those of CBDI and not necessarily those of International Business Machines Corporation. CBDI AND INTERNATIONAL BUSINESS MACHINES CORPORATION DISCLAIM ALL WARRANTIES, EXPRESS OR IMPLIED, INCLUDING THE IMPLIED WARRANTIES OF MERCHANTABILITY AND FITNESS FOR A PARTICULAR PURPOSE.

Any references in this document to non-IBM Web sites are provided for convenience only and do not in any manner serve as an endorsement of those Web sites. Use of those Web sites is at your own risk.



Independent Guidance *for* Everware-CBDI Service Architecture and Engineering



Subscribe to the CBDI Forum

The CBDI Journal is published monthly with a combined July/August edition.

An annual corporate subscription includes access to all back numbers plus access to Powerpoint Libraries and the CBDI Hot Line Service. In addition Corporate Subscribers are encouraged to participate in Special Interest Groups (SIGs), Reviews and general Forum Meetings.

*For more details see:
www.cbdiforum.com*

CBDI Objectives

CBDI Forum aims to provide independent, action oriented practice guidance on Service Oriented Architecture and Component Based Development for architects, business analysts, project managers, designers and others involved in creating and delivering advanced architectures.

CBDI Delivery Channels

CBDI Forum provides:

- Subscription services – continuous practice guidance published in the *CBDI Journal* every month (with July/August combined into one volume)
- Workshops and Seminars – providing indepth education on architecture, process and practice. Public and In-house classes are available.
- Consulting – specific guidance on adoption roadmap including status assessments, methodology customization, architectural guidance including reference architecture development, governance reviews, business design and strategy development.

CBDI Background

CBDI Forum is the Everware-CBDI research capability and portal providing independent guidance on best practice in service oriented architecture and related delivery processes. Working with F1000 enterprises and governments the CBDI Forum research team is progressively developing structured methodology and reference architectures for all aspects of service oriented architecture.

A CBDI Forum Subscription provides a corporation or government department with access to a unique knowledgebase, ongoing continuous practice research guidance materials and hotline access to CBDI Forum experts. The monthly *CBDI Journal* provides in-depth treatment of key practice issues and guidance for architects, business analysts and managers. Forum Meetings are held periodically in Europe and North America allowing peers to engage and exchange experience and best practices.

Contact Us

For further information on any of our services contact us at: info@cbdiforum.com or +353 28 38073 (International)

IMPORTANT NOTICE: The information available in CBDI publications and services, irrespective of delivery channel or media is given in good faith and is believed to be reliable. CBDI Forum Limited expressly excludes any representation or warranty (express or implied) about the suitability of materials for any particular purpose and excludes to the fullest extent possible any liability in contract, tort or howsoever for implementation of, or reliance upon, the information provided. All trademarks and copyrights are recognised and acknowledged.