WebSphere® Studio Workbench

Leveraging WebSphere technology to design, develop and deploy extensible Web application tools.

October 2001

e® business software

Contents

Preface

The IBM WebSphere® Studio Workbench is the foundation for developers and IBM Business Partners to build the next generation of application development tools that integrate with the WebSphere software platform. WebSphere Studio Workbench offers state-of-the art tool technology that is designed to enable an unprecedented level of inter-tool integration.

Keep up with e-business evolution

Advancements in technology—and the maturity of market strategy—in the field of application development continue to redefine the nature of e-business. With that change comes increased pressure on business planners and application developers to find solutions that meet system demands and business objectives.

The rapid, ongoing evolution of e-business has changed the need for business application integration. To survive in the competitive e-business marketplace, you need applications that interoperate seamlessly. To create that environment you need an integrated set of development tools to understand and simplify the complex relationships between interoperable components, while preserving their functionality (Figure 1).

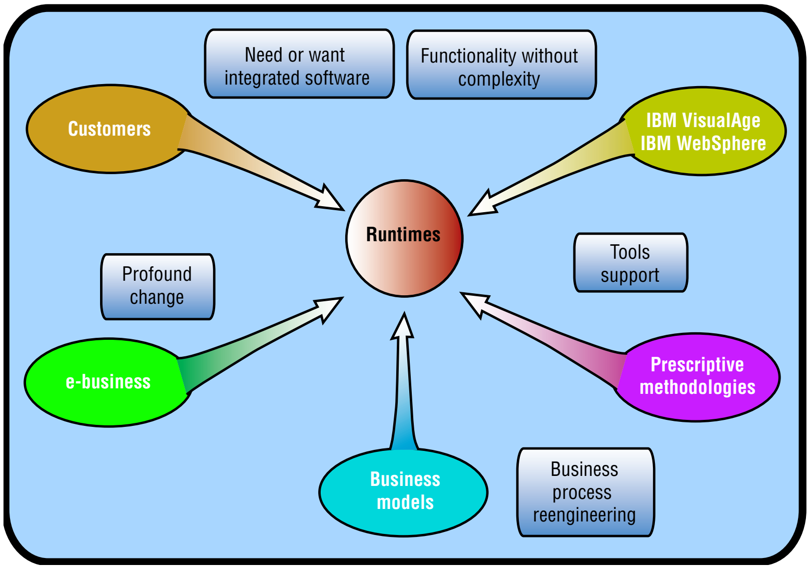

Need or want integrated software
Functionality without complexity
Customers
IBM VisualAge IBM WebSphere
Runtimes
Profound change
Tools support
e-business
Prescriptive methodologies
Business models
Business process reengineering

Figure 1. *Pressures on contemporary software development tools*

Simplifying the e-business environment

A typical Web-based transaction originates with a client, travels through a communication channel to a server and is queued to a back-end processor for data access or integration with existing applications. The result then travels back through the same path to the originating client.

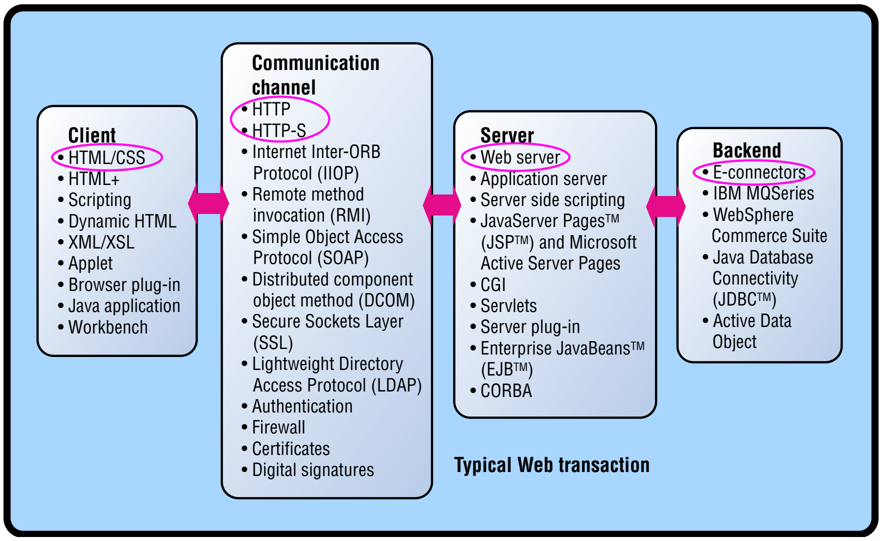


Figure 2. *Web development technology choices*

For most early adopters of Web technology, the entry point was from a business-to-consumer (B2C) perspective. The goal was to create a Web presence that presented information – most often with static HTML pages – to tell customers who you were and what you offered. For illustration, a typical transaction could be defined as a horizontal slice through various technology choices to satisfy a request (Figure 2). The tools and skills you needed were geared to support the HTML-based environment.

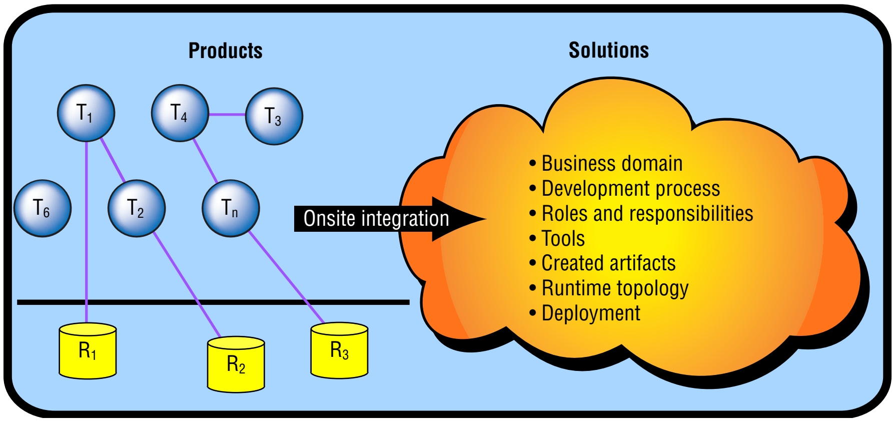


Figure 3. *Required onsite integration by the customer with individual shrink-wrapped tools*

As the focus shifted from simple, static Web page development to delivering business processes to the Web, early adopters had to migrate to a different, more complex scenario. They began to appear in the business-to-business (B2B) space. As their application requirements evolved, the tooling required to support these new requirements had to follow suit.

The jump in technology and complexity in the transition from B2C to B2B models is significant. Businesses need to buy or rapidly develop new tools and then retrain their employees to use them efficiently and effectively. Tool vendors have traditionally taken a product focus by providing individual tools designed to address specific needs. The result: tools restricted by programming language and operating system stacks that lack the integration customers require (Figure 3).

That means customers – whose expectations have been set high by the rich functionality offered by modern development tools like Lotus® SmartSuite® and Microsoft® Office – have to use their own resources or pay for consultants to integrate their tools. To make integration easier for customers, and to help reduce their development costs, tool vendors must change the way they deploy their tools in the marketplace.

Application development tools like those available from the IBM VisualAge® and WebSphere brands provide rich functionality that can support customer applications. However, these functions can be complex to adopt. Customers need application development tools that are truly integrated with one another to hide this complexity.

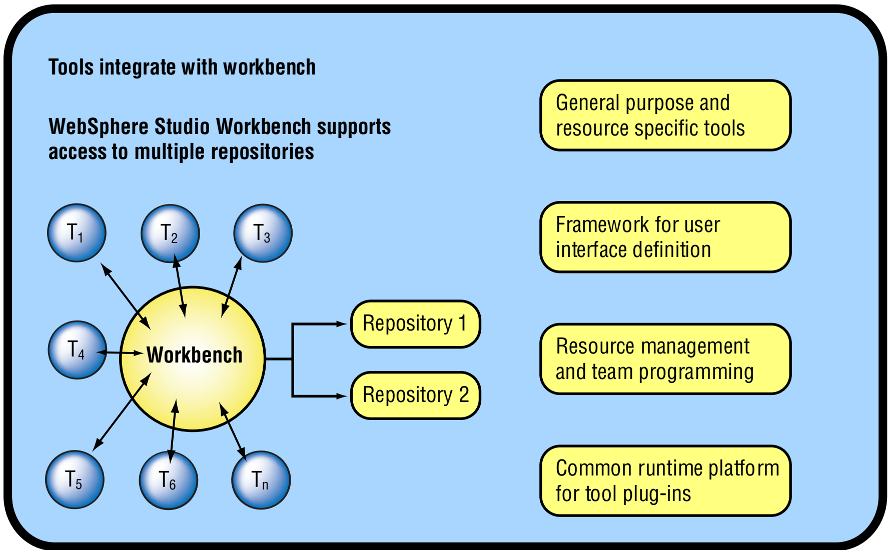
Using the WebSphere Studio Workbench platform, IBM is making a significant transition in tool strategy toward a more solution-based model of application development, driven by customer usage scenarios. IBM and IBM Business Partners can use WebSphere Studio Workbench technology to better meet customer business growth objectives, help reduce development costs and help customers get to market faster.

Access common services and frameworks

The strength of IBM application development tooling is the concept of a tool integration platform – WebSphere Studio Workbench – that provides the base for accessing common services and frameworks. And, it includes well-defined extension points that allow you to add specialized tool function with separate plug-in components (Figure 4). A common set of services and frameworks can help reduce the need for significant on-site integration of diverse tools. In this way, software developers can focus their resources on what they do best: provide solutions for their customers.

Leverage existing tools and extend workbench technology

IBM developers are currently creating plug-in components by mining functionality from our existing tool technology. And, IBM has established an independent software vendors (ISVs) program that encourages IBM Business Partners to integrate their technology with WebSphere Studio Workbench. To start, IBM currently makes WebSphere Studio Workbench available to IBM Business Partners at no charge, and now allows the technology to be redistributed without royalty charges.


Tools integrate with workbench

WebSphere Studio Workbench supports access to multiple repositories

General purpose and resource specific tools
Framework for user interface definition
Resource management and team programming
Common runtime platform for tool plug-ins
T1
T2
T3
T4
Workbench
Repository 1
Repository 2
T5
T6
Tn

Figure 4. *A common workbench with plug-in components*

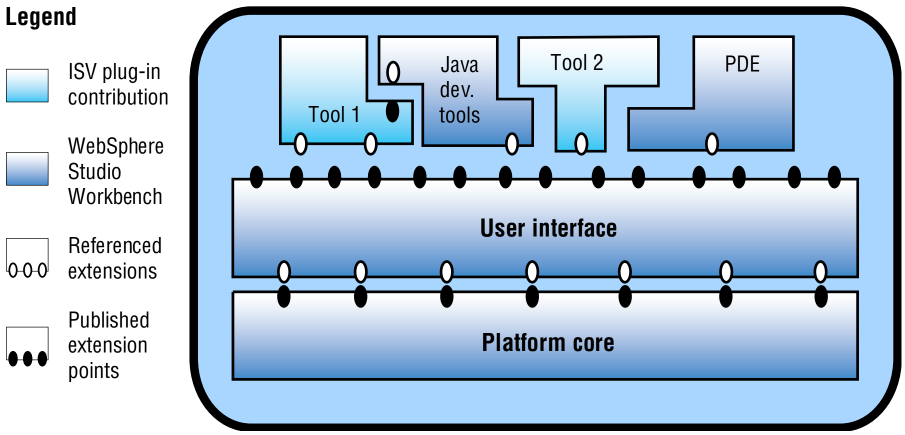


Figure 5. *The WebSphere Studio Workbench platform*

An open, extendable environment

WebSphere Studio Workbench is a standards-based tool platform and integration technology (Figure 5) that allows tool providers to build platform-neutral, professional, enterprise-ready application development tools optimized for the WebSphere software platform.

Application development tools, based on WebSphere Studio Workbench, are sets of components, or plug-ins, that integrate and interoperate in realtime with other WebSphere Studio Workbench tools. This is similar to the role plug-ins play with Web browsers. Plug-ins extend the WebSphere Studio Workbench through the use of defined extension points.

A tool provider delivers one or more tool plug-ins that add provider-specific tool capability to the WebSphere Studio Workbench. The WebSphere Studio Workbench user interface (UI) and core components provide the runtime platform and integration capability to be used by installed plug-ins. As the core around which ISVs can build products, WebSphere Studio Workbench also includes a simple environment to build Java™ technology-based tools and plug-in development environment (PDE) tools.

Platform core

The WebSphere Studio Workbench platform core (Figure 6) enables the plug-in runtime platform and defines a general extension architecture to access publicly available extension mechanisms. Tool developers can create plug-in extensions to the platform core and to the WebSphere Studio Workbench UI. The platform offers support to install and register tool extensions, and multiple versions of the same plug-in can coexist to help simplify version management and updating. Additional plug-ins can be quickly integrated to extend the workbench environment. Your plug-ins can include extension points to accept other plug-ins, allowing increased variability and customization. The platform core allows controlled access to certain open application programming interfaces (API) and facilities of the core.

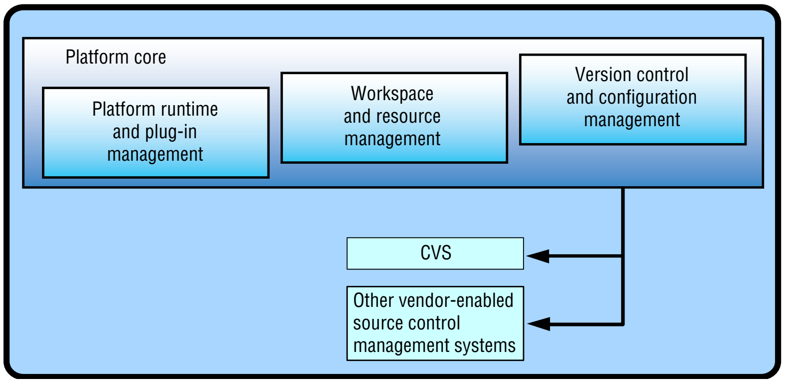

Platform core
Platform runtime and plug-in management
Workspace and resource management
Version control and configuration management
CVS
Other vendor-enabled source control management systems

Figure 6. *Platform core component*

WebSphere Studio Workbench provides easy implementation of many of the common services required by an application development tool, including:

- *Plug-in registry. Defines the registry for plug-ins, their extension points and extensions*
- *Workspace and resource management support. Defines the project model used by the workbench; allows resources to be edited in the local file system in the user's private workspace and can be integrated with a workgroup versioning server*

- *Workspace model. Facilitates version control and configuration management of workspace resources; provides support for managing assets in a multiuser, distributed or disconnected environment; and allows a project to be associated with a supported repository*

WebSphere Studio Workbench also includes access to repository function with support for systems, such as Concurrent Versions System (CVS). CVS is an open source repository used widely on UNIX® platforms for both proprietary and open source development. Source code management vendors can add plug-ins to WebSphere Studio Workbench to gain direct access to their implementation of repository function.

WebSphere Studio Workbench user interface

The WebSphere Studio Workbench UI (Figure 7) can help you reduce the learning curve as developers move from tool to tool and role to role in the development cycle, by providing a common look and feel across all tools. They have the tools they need, a rich set of reusable plug-ins, a set of UI frameworks that facilitate plug-in development and a widget toolkit that supports a portable API to host native widgets. The UI also offers key interaction features to simplify artifact-based work, such as a navigator, content outliner, property sheet, samples, tasks, wizards, a content editor and a debugger.

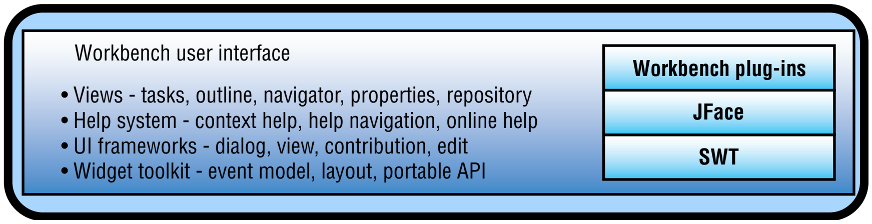

Workbench user interface
• Views - tasks, outline, navigator, properties, repository
• Help system - context help, help navigation, online help
• UI frameworks - dialog, view, contribution, edit
• Widget toolkit - event model, layout, portable API
Workbench plug-ins
JFace
SWT

Figure 7. *Workbench user interface component*

A number of common, reusable plug-ins are included in WebSphere Studio Workbench and can be used by other plug-ins to help developers view resources and edit project work in progress:

- *Outline view. Allows logical viewing of resource contents using an extension of the repository resource navigator*
- *Repository resource navigator. Helps create, rename, delete, organize, navigate and synchronize solutions, projects, folders and files*
- *Tasks view. Shows user-created or plug-in-created tasks that use the marker framework to open and position editors at appropriate points*
- *Properties view. Provides an editor or viewer of resource properties*
- *Help system. Implements a platform-optimized help Web server and document integation facility*
- *Content area. Offers a managed area of the desktop, shared by resource editors*

All content-specific functionality is provided by plug-ins that integrate with the desktop and with each other. WebSphere Studio Workbench provides the central integration point for plug-in UIs through a common view of the complete application across all components. You can improve project management, make the navigation of shared resources more efficient and leverage a wider range of skills across your entire team.

The UI frameworks provide the infrastructure required to build workbench plug-ins, such as editors, views, dialogs and integrated help. Java programs can use the UI frameworks and the standard widget toolkit to implement UI processing. With UI frameworks, you can refocus tool developer skills to mine tool functionality by reducing the time developers spend building a tool-specific infrastructure.

The WebSphere Studio Workbench UI also supports the integration of ActiveX controls and OLE Documents on Microsoft Windows® platforms.

Support for workbench-based tool development

WebSphere Studio Workbench includes development tools to help create Java applications and PDE tools to extend its application development strength and support the development and integration of additional plug-in extensions. The WebSphere Studio Workbench development tools for Java technology include:

- *Project nature builders to invoke the compiler in response to changes in source files*
- *Application and debug perspectives to organize WebSphere Studio Workbench UI for editing or debugging applications*
- *Views, wizards, preferences and property sheets to create, view and edit resources*
- *Packages and hierarchy views*
- *Text editor with full syntax highlighting and complete code assist based on visibility in the project's classpath, including elements in the current edit file*
- *Ability to view the source one method at a time or one compilation unit at a time (the entire file)*
- *Configuration support for program invocation and Java Development Kit (JDK™) level options*
- *Refactoring tools to reorganize applications*
- *Intelligent search, compare and merge tools for source files*

The PDE provides:

- *Views and editors that allow you to visually create the plug-in manifest file (plugin.xml)*
- *Ability to generate code templates for Java applications that establish a basis for the implementation of desired tool extensions*
- *Support for self-hosting that can enable WebSphere Studio Workbench to launch an additional instance of the workbench to support plug-in component testing during tool development*

With the PDE, you can specify your plug-in runtime processing attributes and identify other required plug-ins, define extension points, associate XML schema files and create extensions on existing plug-in points.

WebSphere Studio Workbench platform-based solutions

A common, standards-based tool integration platform can provide value to customers who buy tools to use and the ISVs that form relationships with IBM to develop tools for WebSphere Studio Workbench. The goal of the WebSphere Studio Workbench project is to provide ISVs with an integrated development environment to help them reduce time to market and reuse development tool components profitably. You can leverage the WebSphere Studio Workbench platform with a set of plug-ins from one or more ISVs to create solutions for customer scenarios.

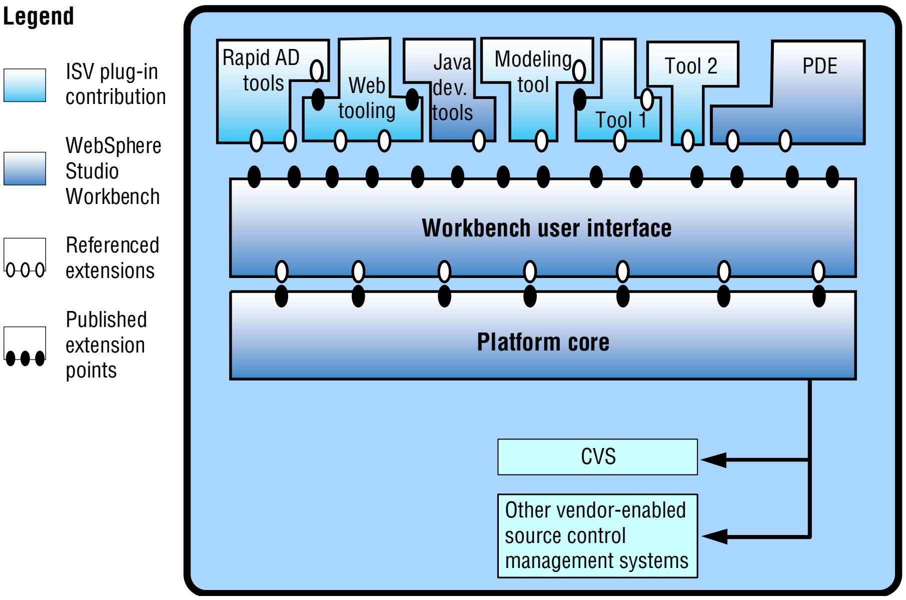


Figure 8. *WebSphere Studio Workbench platform-based solutions*

Gain access to a broad range of integrated tools

With tool integration support and unique class-loading mechanisms, the WebSphere Studio Workbench platform can help customers gain better access to a broad range of integrated application development tools. And, the versatile, flexible platform and tool components can be configured to support the various roles and responsibilities of your entire development team. WebSphere Studio Workbench provides an orderly progression of tool features and functions when different versions of tools are integrated on the same instance of the platform (Figure 8). That can mean fewer conflicts between components, faster, tighter integration and less time spent writing code and debugging. With the confidence that everything will work and integrate easily from the start, you can reduce the costs of managing tool inventories for your developers.

You can also realize faster construction and deployment of high-performance applications using a set of integrated tools that support the full development lifecycle. WebSphere Studio Workbench enables an integrated approach to the development and management of many different types of application resources.

Reach new and growing markets faster with help from IBM

IBM Business Partners can leverage WebSphere Studio Workbench technology as the base of their tool development activity to reach new and growing markets. Drive e-business at the leading edge – as the WebSphere Studio Workbench platform becomes widely adopted – by supporting the development of applications for IBM WebSphere Application Server and IBM MQSeries® runtime environments. Respond on the fly to changing markets with the flexibility of WebSphere Studio Workbench integration points that can allow you to quickly and easily add and remove tool components. You can reduce development costs, and focus resources on core competencies by leveraging the robust functionality of WebSphere Studio Workbench and IBM tools. And, give your marketing resources a boost by taking advantage of IBM marketing programs when you develop tools that integrate with IBM tools.

For more information

To learn more about IBM WebSphere Studio Workbench technology, visit:

www.developer.**ibm.com**/welcome/wstools/workbench.html

IBM Corporation
Software Group
Route 100
Somers, NY 10589
U.S.A.

Produced in the United States of America
10-01
All Rights Reserved