IBM Lotus Domino applications and the IBM Workplace technical strategy.
Introduction

In 2004, IBM, a trusted leader in middleware, introduced IBM® Workplace™ software—a family of products, technologies and solutions designed to complement, support and integrate with one another to transform the way people work.

IBM recognizes the emergence of a new work paradigm in which employees collaborate with each other in the context of business processes to increase organizational productivity. IBM Workplace software supports this paradigm by using a portal framework to tie together key applications and functions based on employees’ roles, and enhancing them to provide contextual collaboration at their fingertips wherever they work. The result is an environment that can help businesses quickly meet the demands of their customers and partners.

In January 2005, IBM announced IBM® Workplace™ Collaboration Services, targeted for availability in mid-2005. Workplace Collaboration Services integrates a range of collaborative features into a single product, providing a consistent, integrated collaborative environment delivered through open standards to a variety of client options and applications. The IBM Workplace environment also encompasses infrastructure elements provided by IBM Lotus® Notes® and IBM Lotus Domino® software, as well as IBM WebSphere® middleware, including IBM WebSphere Portal and IBM WebSphere Everyplace® software (see Figure 1).

There are different advantages to using IBM Lotus Domino, IBM WebSphere Application Server, IBM WebSphere Portal or IBM Workplace Collaboration Services for your applications:

- The strengths of Domino lie in collaborative applications, providing a hierarchical data store, rich-content handling and a robust security model.
- WebSphere Application Server offers great value for transactional applications, highly leveraging a relational data store and providing enterprise-level Java™ 2 Platform, Enterprise Edition (J2EE) application hosting.
- WebSphere Portal excels at application and data aggregation, providing a rich framework for Web-based applications.
- Workplace Collaboration Services brings together a wide range of ready-to-use collaboration capabilities into a flexible, standards-based work environment. This next-generation platform is designed to support composite applications, blending components built on a J2EE platform with those built with Domino and WebSphere Portal tools.
This paper is designed to provide developers, chief technology officers (CTOs) and other IT decision makers with information about the tools and technologies used to support an IBM Workplace environment.

**Overview of IBM Workplace software**

IBM Workplace software is an innovative portfolio of capabilities that represents the front end of computing. It is designed to help people be more productive by enabling them to make more-informed decisions and take targeted actions faster. The goals of the IBM Workplace strategy are:

- **Unified access to the tools and resources that users need to communicate, collaborate and conduct day-to-day business, customized for their unique roles in the organization.**

- **Support for open standards, including J2EE and Eclipse, allowing existing or new IT investments to be intelligently integrated.** Eclipse is an award-winning, open-source platform for the construction of powerful software development tools and rich desktop applications. Full details on Eclipse and the Eclipse Foundation are available at [www.eclipse.org](http://www.eclipse.org).

- **Network-centric delivery model that centralizes server-based deployment and management of clients and applications, and combines the productivity benefits of the desktop with the cost advantages of the network.**

- **Robust anytime, anywhere access to accommodate mobile and traveling users as well as those who use a dedicated desktop.**
Figure 1. IBM Workplace software provides a dynamic work environment that brings together the collaborative tools, applications and connections required by the business situation.

The IBM Lotus Domino application development and deployment environment enables you to develop collaborative applications quickly and to take them online, bringing people, processes and data together to facilitate productivity and quick decision making in an on demand business. For this reason, Domino applications are an integral part of the IBM Workplace environment. Existing custom applications built with Lotus products can integrate with J2EE components of IBM Workplace, allowing you to further leverage your application investments.

In the IBM tradition, our clients who have implemented Lotus Notes and Domino continue to benefit from support for the traditional Domino application development model. Over time, we expect Lotus Domino to offer increased support of J2EE and infrastructure standards, such as Java Server Page (JSP) tags, Java application programming interfaces (APIs), Lightweight Directory Access Protocol (LDAP), Web services and relational database (RDB) integration to assist developers who are interested in working in both the Domino and J2EE application environments.
There are numerous advantages to using the J2EE platform. Many industry-leading, independent software vendors (ISVs) have begun writing applications for this open development platform. This means that the J2EE-based applications you develop will have the ability to interact in a much larger environment. Using a standards-based directory and data store, J2EE helps reduce your total cost of ownership (TCO) while offering advanced scalability and flexibility. Because J2EE is a layer above the operating system, it allows for application portability by enabling a developer to build an application for the J2EE platform and not for a specific operating system or device.

IBM WebSphere Application Server leverages the J2EE specification as the application development server for IBM Workplace Collaboration Services. J2EE provides a specific architecture for building, deploying and managing applications in multiple tiers, often broken into presentation, logic and data. This architecture is designed to provide scalability, flexibility and manageability. Although J2EE is a rich application-development platform, it has few features to support collaboration, so it benefits from integrating with the rich collaborative capabilities of Lotus Domino or Workplace Collaboration Services. Applications designed for Workplace Collaboration Services, or for Lotus Domino in conjunction with WebSphere software, blend powerful collaborative features with high transactional scalability to deliver end-to-end on demand business solutions.

Services of IBM WebSphere Portal provide the framework for bringing together the modular components of Workplace Collaboration Services, and are integrated into the product. WebSphere Portal provides both Lotus Domino and WebSphere developer communities with a modular architecture for building integrated, collaborative solutions. Domino developers can use the collaborative services provided by Workplace Collaboration Services to augment and enhance their applications. WebSphere and J2EE developers can integrate collaborative capabilities from Lotus Domino and extended products through Web services. In both cases, integrating collaborative solutions becomes easier and can be accomplished with standard developer skills.

IBM is focused on developing software that not only makes new things possible, but also makes them easier and less expensive to deliver. The IBM Workplace strategy embodies collaborative solutions, integrated virtually anywhere in the enterprise, to help boost productivity and enhance communication in all areas of your business.
IBM Workplace application servers

IBM offers flexibility and choice in application server software. The options include IBM Lotus Domino, IBM WebSphere Application Server, IBM WebSphere Portal and IBM Workplace Collaboration Services.

IBM Lotus Domino: Collaboration beyond e-mail

IBM Lotus Domino is a comprehensive platform for collaboration that addresses both connected- and disconnected-mode requirements for data and applications. Most customers initially implement Lotus Domino for its built-in enterprise e-mail, calendar and scheduling functions, making those the most-widely deployed collaborative applications. However, many customers exploit the “more than mail” capabilities that support core business processes, which enable employees to work together efficiently with the protection of advanced security features. Lotus Domino is comprehensive; it provides an infrastructure to create, test, deploy and manage distributed, multilingual applications including directory, database, application server, administration, security, connectivity, Web server, e-mail server, calendaring engine and more—all in one system.

Domino developers can design applications for the Lotus Notes client, Web browsers, supported mobile phones and handheld devices, or a hybrid environment accessed by multiple types of clients. Domino applications can use replication and offline services for security-rich, synchronized applications that work as well in a disconnected mode as when accessed on a server over a network. Replication enables users to save a local copy of a Domino application and its data and to periodically synchronize that data, so that users can be productive and efficient—even when they are disconnected from the network.

High-value Domino solutions include document-centric processes and workflow routing. Some examples are project team rooms, document repositories, discussion forums, sales-force enablement systems and employee self-service applications. Businesses of all sizes have benefited from Domino applications. As a comprehensive application platform, Lotus Domino includes out-of-the-box application templates, a document-based object model and broad programming language support for building custom collaborative applications. With these choices, your organization can use many developer skills to develop a Domino application.
Domino applications can implement business logic through the use of Notes Formula Language or the scripting language called LotusScript, similar to Visual Basic, for interactive field validation or for running agent code on an event-driven or scheduled basis. For more-advanced solutions, developers can use Java, Microsoft Component Object Model (COM), C/C++ or CORBA.

Multiple interfaces to a single object model enable developers to choose the most appropriate language for the task and reuse their skills in new applications and solutions.

Some solutions require external data and globalization support. Lotus Domino add-on tools facilitate such solutions. Using visual data-mapping techniques, a developer can easily and quickly integrate relational data with Domino data. For example, you can use IBM Lotus Domino Enterprise Connection Services (supplied with Lotus Domino) or IBM Lotus Enterprise Integrator® (a separately orderable product) to integrate data from a wide variety of systems like IBM DB2® Universal Database™, Oracle and Microsoft SQL Server without programming. Using the Domino Global Workbench capability included with IBM Lotus Domino Designer, Domino applications that require global deployment can be translated into a variety of languages by using straightforward forms. In this way, the same application can be delivered to multiple users in their native languages.
The Lotus Domino road map for application development builds on the fundamental premise that Domino is a flexible and open platform, which is demonstrated by eXtensible Markup Language (XML) and broad programming-language support. Flexibility and openness are key to a Domino application’s ability to leverage J2EE. You can extend your Domino application investment in data and application logic in a number of ways. For example, you can expose Domino data using Web services (through LotusScript or Java) or JSP tags to integrate with IBM Workplace or WebSphere applications. Another option is to surface your Domino application directly in a portal using IBM WebSphere Portal.

Progressive innovations to Domino Designer and to the Domino programming model offer continued integration with IBM application development tools for building J2EE applications. Continued integration is not designed to replace Domino Designer, but would help facilitate teams of developers using Lotus Domino, WebSphere Application Server, WebSphere Portal and IBM Workplace Collaboration Services in their environments. These teams can benefit by using the strengths of each system when building applications.

**IBM WebSphere Application Server: The need for open standards**

In the early stages of on demand business, application servers, integration servers and customer relationship management (CRM) servers all had their own software technology stacks. Over time, it became clear that the core engine required to build, deploy and manage any application containing new business logic could be defined as a single set of application-server functions. And, if this common engine could be standards based, it could more quickly and easily become the foundation for any number of specialized software packages.
As organizations invest in a J2EE server software platform, choosing the right one for their environments can seem a daunting task. Yet, although there might be many choices, the basic functionality of all J2EE server software remains essentially the same. These core elements of the J2EE application-server model are defined in the J2EE specification and provide a vast array of services for the application that runs on it. The server handles the life cycle of the various application elements, providing them with current state and context information, authentication information about the current user, and transactional context for highly sensitive applications.

Where the J2EE specification leaves off, a J2EE server software vendor can provide enhancements that make one product more appealing or appropriate to a certain customer’s environment. This gives software vendors the ability to develop value-added functionality without impacting the standards with which the application server software must comply. Therefore, the development model for each product from various vendors is similar and allows customers to use common administration and development skills.

IBM WebSphere Application Server provides features that support dynamic applications, improved ease of management, and support for the latest J2EE and Web services standards. This application server software also addresses scalability and performance with automated performance-tuning and load-balancing services. WebSphere Application Server is available in several configurations across a wide range of operating systems to address today’s needs and provide growth options for the future.
One of the key differences between WebSphere Application Server and the Domino application server is that Lotus Domino provides a fully integrated environment—application execution, user authentication, directory services, data hosting and presentation display—all in one system.

In contrast to Lotus Domino, WebSphere Application Server does not contain its own data manager. Instead, it relies on a relational database, such as IBM DB2 Universal Database, to provide information storage for its applications. One way that WebSphere Application Server provides access to those data sources is by using the J2EE Connector Architecture (JCA). JCA defines a set of services that a connector developer can expect to be available to an adapter at application run time. The three services defined in JCA V1.0 include:

- **Connection management**
- **Transaction management**
- **Security management**

The J2EE model includes elements for providing the application with all the same information, but the J2EE server alone is not responsible for addressing all the aspects that the Domino server handles. The J2EE server calls out to different parts of the IT environment to fulfill requests for application data and directory information. For example, whereas the J2EE specification outlines how the server can pull data from a data store into the application using the JCA or Java Database Connectivity (JDBC), J2EE does not require that the server contain the database manager itself.

**IBM WebSphere Portal: Single point of access**

Business portals serve as simple, unified access points to Web applications. Portals also do much more: they provide valuable functions like security and search, and they enable collaboration and workflow. A portal delivers integrated content and applications, plus a unified work environment. Indeed, portals are the next-generation desktop, delivering on demand business applications over the Web to all kinds of client devices. A complete portal solution should provide users with convenient access to everything that they need to get their tasks done anytime, anywhere, in a security-rich manner. Portals provide the tools and user interface to access information and applications, and to manage the selection and personalization of content.
Within the portal desktop, portlets provide access to applications, Web-based content and other resources. Portlets are small, task-specific, reusable applications that allow separation of business-model functions from the presentation and control logic. Companies can create their own portlets or select from a catalog of portlets created by IBM and by IBM Business Partners. Portal administrators can configure portlets to supply credentials for the target resource on behalf of the current user, and multiple instances of the same portlet can be personalized to display different data for each portal user.

IBM WebSphere Portal provides the application connectivity, integration, administration and presentation services that are required across portal environments. IBM is building on these services to develop solutions that deliver highly personalized and context-sensitive applications, accessible from a wide range of devices. As business applications enter the on demand era, WebSphere Portal leads the way with its concepts of delegated administration, cascading page layouts, portal federation through Web services, advanced portlet application concepts, business-process integration, knowledge management and advanced personalization. Through complementary offerings, additional mobile computing functions are enabled, such as intelligent notification, offline browsing and data synchronization.

WebSphere Portal supports open standards such as Java Specification Request (JSR) 168 and Web Services for Remote Portlets (WSRP). JSR 168 is a specification from the Java Community Process that addresses the requirements of aggregation, personalization, presentation and security for portlets running in a portal environment. Its purpose is to address portlet compatibility between portal servers offered by different vendors. WebSphere Portal includes a portlet run-time environment that supports a large part of the Java Portlet Specification defined by JSR 168 in addition to portlets built with the IBM Portlet API.

WSRP allows Web services providers, known as producers, to easily serve content and applications to portals, known as consumers. The objective of WSRP is to reduce integration time and storage overhead. WSRP enables consumers to invoke the remote services of the producer as a portlet without extensive configuration for interface, protocol, data and presentation. IBM WebSphere Portal supports WSRP and can serve as either a WSRP producer or a WSRP consumer.
IBM Workplace Collaboration Services: Integrated work environment

Traditional collaborative solutions encapsulate individual collaborative features into distinct product offerings, such as e-mail, instant messaging and workflow applications. Supporting many of the tenets of the IBM Workplace strategy, IBM Workplace Collaboration Services integrates a range of collaborative features within a single product.

IBM Workplace Collaboration Services provides an easy-to-use, easy-to-manage, reliable and customizable work environment that:

- Is built on the J2EE platform services of IBM WebSphere Application Server and uses the relational information management services of IBM DB2 Universal Database. Other supported relational databases include Oracle Enterprise Edition and Microsoft SQL Server.
- Uses the presentation services of IBM WebSphere Portal plus additional built-in services to deliver collaborative capabilities to the user in the context of the role the user plays.

Workplace Collaboration Services uses a simplified approach to deploying software using a network-centric delivery model that avoids touching the desktop and helps to reduce administration and systems management costs. In addition, Workplace Collaboration Services is built on an open-standards model that can be combined with industry and cross-industry solutions available from IBM and IBM Business Partners.

By providing flexible, integrated collaborative capabilities that can be used in any combination and activated on demand in response to business change, IBM Workplace Collaboration Services can help you drive down costs and increase business productivity.
Application platform considerations
The IBM Workplace portfolio provides a choice of application development models and corresponding runtime servers. The Workplace environment supports both J2EE technology-based applications and Lotus Notes and Domino platform-based applications. Each server environment and application-development model has its strengths and weaknesses. Your selection of application platform can be an important factor in the success of an application.

This section of the paper covers various topics such as the application deployment topology, the skills available in the organization, and the technical requirements of the application data and scalability. These considerations are designed to help you determine the appropriate design strategy for new applications. In many cases, the new application can combine the strengths of J2EE and Domino for high return on investment (ROI) of the application.

The considerations to guide an application platform decision are typically based on three types of requirements: business application requirements, user requirements and infrastructure requirements.

Business application requirements
The scope of business application requirements includes the nature of the business process, content creation and content storage; granularity of access control; length of time to value; and strategic compared with tactical nature.
Nature of the business process
Is the application based on a document-centric workflow or is it more transactional in nature?

A transactional application is one in which a series of steps in the application is monitored so that the process can be completely reversed if necessary. A classic example is a banking system in which an account credit must be reversed if the preceding debit step is not completed. Transaction processing is a key benefit of the J2EE application model of IBM WebSphere Application Server and IBM WebSphere Portal.

Document-centric applications often require the routing of electronic forms to users in a series of predefined steps in order to complete a workflow process. For example, in an employee self-serve help-desk application, the employee fills out a form that is routed by e-mail to the appropriate help-desk professional when submitted. Upon closure of the problem, the completed form is routed back to the originator for sign-off. After sign-off, the form is logged into a help-desk tracking database. This type of document-centric solution is easy to build and deploy on Lotus Domino.

Lotus Domino is best suited for medium-volume, transaction-oriented applications, although it can potentially handle hundreds of thousands of documents in an application. However, the collaborative services of Domino can be integrated with a J2EE application to support the human intervention necessary for resolving exceptions that occur in business transactions. A value-added solution can use a Lotus Domino user-centric, forms-based workflow application in support of transaction-processing workflows that are more suitable to be managed by WebSphere Application Server and IBM WebSphere MQ. Such end-to-end workflow solutions that merge system transactions with human processes are capabilities that few single vendors can offer.

IBM Workplace Collaboration Services is designed to help you streamline business processes where collaboration is required. Rather than having business activities take place in isolation, unrecorded and out of context, Workplace Collaboration Services provides a means to integrate collaboration into the context of the activity.
For example, while processing an expense report, the employee can see a list of experts in travel policy and human resources who are online and available to answer questions he might have. When a manager needs to approve the expense report, she can ask the employee a question using instant messaging and easily refer to travel policy documents, all without leaving the context of her work environment. IBM Workplace Collaboration Services can provide the unifying force for applications that require both transactions and collaboration.

**Nature of content creation**

*Will the application include rich-format content editing, creating and rendering?*

Many user-facing applications require the ability to display, create, store, edit and delete rich text, spreadsheets, presentations and other composite documents. Lotus Notes and Domino have built-in capabilities for handling this type of content through a built-in, rich-text editor. Lotus Notes also integrates with installed desktop applications, including Microsoft Office, to support in-place editing of attached files. Lotus Domino can also serve rich-text data stored in Domino databases to Web browsers and accepts input of attributed text from the Web browser.

The productivity components provided with WebSphere Portal and with IBM Workplace Collaboration Services allow you to create and edit rich text, spreadsheet and presentation documents from the Web browser and do not require installation of a local desktop application. The productivity tools are available for Windows and Linux® operating system users.

If users want to create and edit documents using familiar applications, including Microsoft Office or IBM Lotus SmartSuite® software, that are installed on their desktops, portal administrators can configure WebSphere Portal and IBM Workplace Collaboration Services to allow this option. The WebSphere Portal document manager stores and manages these documents within document libraries. For each document library, the library administrator can also specify user access permissions, document-versioning options, approval workflow and document locking.
WebSphere Portal document manager allows the user to search a document repository and to import files into document libraries. The administrator can also configure the document manager to allow users to view documents as HTML files. This feature is useful if the user does not have access to the application that was used to create the stored file.

The rich-client option for IBM Workplace Collaboration Services, called IBM® Workplace Managed Client™ software, allows users to work online or offline with documents and document libraries. Workplace Managed Client synchronizes the local copy and server copy of the document library. Users of Workplace Managed Client can also use their locally installed Microsoft Office products to edit documents and can import files from their local file system into the document library. Workplace Managed Client can run on Windows and Linux systems.

**Nature of content storage**

*Does the application involve structured data or unstructured data?*

Transactional applications are historically based on structured data. Information is stored in records consisting of fields with a specific length and identified data type (for example, text or numeric). This type of application lends itself to relational data storage. A relational database (RDB) maintains multidimensional links between related records, such as the linkage among customer demographics, invoices, purchase orders and vendor information. IBM WebSphere Application Server and WebSphere Portal use an RDB to store their configurations and contents, and provide robust connectivity to RDB systems.

Document-centric applications typically involve unstructured data, such as free-form text, lists, spreadsheets, word-processing documents and presentation files. Hierarchical storage, which uses a parent/child relationship model, is better suited for this type of information. Lotus Domino has traditionally used a built-in hierarchical, unstructured data store called Notes Storage Facility (NSF).
Modern applications often require a blend of relational and hierarchical data. This is a compelling reason for IBM and our clients to integrate Lotus Domino software and J2EE-based capabilities. IBM Workplace Collaboration Services uses a document object model with a relational database management system under the covers. Beta features for Domino 7 include a Notes application plug-in to IBM Workplace Managed Client, and an option to use DB2 Universal Database as an alternative to NSF on selected operating systems.

Enterprise application-integration tools such as Domino Enterprise Connection Services and IBM Lotus Enterprise Integrator facilitate access to relational data from Domino applications. Conversely, hierarchical Domino data can be accessed from WebSphere applications using Domino support for XML, the Domino JSP tag libraries and Domino Objects for Java.

Furthermore, Domino applications can be surfaced in WebSphere Portal through supplied portlets. And a portal document-management feature allows users to create unstructured data within the portal.

Granularity of access control
Do users need to delegate their actions and still allow an audit trail?
Does a user’s access control need to be limited to a subset of records or field?
Does the application need dynamic, role-based user identities?
The advanced authentication model of Lotus Domino is highly granular. It offers seven levels of access to application resources — manager, designer, editor, author, reader, depositor and no access. Access control begins at the server level, extends to the application, then to the document and view level, and can go all the way down to the field level with the use of encrypted fields.

Domino provides delegation support and allows access control based on roles as an alternative to lists of specific user or group names. With Domino, access control can even be altered programmatically, adjusting who sees what content on a dynamic, event-driven basis.
A role-based environment is one of the basic tenets of IBM Workplace technology. IBM Workplace Collaboration Services provides an integrated collaborative work environment that can be easily tailored to the needs of the user’s specific role in the organization.

For example, people on the floor of a manufacturing facility might require e-mail and calendaring for only occasional use. Corporate teams who write proposals and various business-level documents might routinely require team spaces, document libraries, instant messaging and Web conferences in addition to e-mail and calendaring. Based on a service-oriented architecture (SOA), any mix of the desired capabilities can be enabled and accessed from a standard Web browser, without the need for programming. If more-granular access is required, the role-based user identity model provides another integration point for Lotus Domino and J2EE applications.

Length of time to value
How quickly does your application need to be developed (or enhanced) and deployed?

The IBM Workplace portfolio includes rapid application-development tools for Lotus Domino and for IBM Workplace Collaboration Services. Two of the most significant tools are IBM Lotus Domino Designer and IBM Workplace Designer software.

IBM Lotus Domino Designer continues to set the standard for rapidly creating collaborative applications on the Lotus Notes and Domino platform. Domino Designer is easy to use, enabling even novice application developers to build applications based on supplied templates. At the same time, it provides the performance and features that the most-demanding application developers need – whether they are developing applications for a Lotus Notes client, the Web or mobile devices.

IBM Workplace Designer, targeted for availability in mid-2005, is a powerful, easy-to-use tool for the visual script developer to build J2EE-based components for IBM Workplace applications. Using an interface modeled after Domino Designer, Workplace Designer allows the creation of document-oriented, business-process applications using a flexible model based on XML with forms, views and JavaScript.
Strategic versus tactical nature

How strategic is the application? Or is it a tactical or point solution?

This question asks you to evaluate the application purpose and life cycle. If the application is expected to have a useful life cycle of five years or more, or if the application is crucial to supporting the core business, then it is strategic. Tactical or point solutions support a relatively short-term project or business initiative.

Tactical solutions can also include applications that are used infrequently. For example, a human resources employee performance-review application used only once a year is a tactical or point solution because it is infrequently used and is only indirectly tied to core business operations. If your organization has such applications deployed on Lotus Domino, it is not advisable to re-architect them on J2EE because it is unlikely that you would gain sufficient return to offset the development expense.

Strategic applications are architected to deliver maximum returns over the duration of the application life cycle, which includes unforeseen future extension and integration of the applications. Careful assessment of the application life span and consideration of the factors described in this paper should help determine the appropriate platforms for strategic solutions.

User requirements

After you have evaluated the business application requirements, rating various user requirements as to the degree of importance can further assist you in making an informed application-platform decision. User requirements include the need for capabilities such as an integrated user experience, customization for the user’s role, the desktop productivity features of a rich client and offline use with application fidelity. Application fidelity means that the application acts the same way, whether online or offline, for creating, editing or deleting data.
**Integrated user experience**

A Lotus Notes welcome page option introduced with release 6.5 mirrors a significant part of the IBM Workplace Collaboration Services experience. This welcome page provides a single, unified access point to frequently used resources such as e-mail, calendar, instant messaging, contact list, applications and Web sites. End users can benefit from an innovative collaborative work environment, with quick and easy access to multiple information sources.

IBM WebSphere Portal delivers a single, unified point of personalized interaction with applications, content, processes and people. The user interface enables collaboration in the context of the work at hand. With a single sign-on, users can quickly access the dynamic information they need, implement business processes across critical applications, and collaborate with portal users inside and outside the organization. These advantages can help your company improve employee productivity and business responsiveness, cut costs and strengthen relationships with customers and partners.

Key to IBM Workplace Collaboration Services is its portal-based user interface. By delivering capabilities as portlets, many Domino applications written by IBM clients and Business Partners can be integrated with Workplace Collaboration Services. But unlike traditional and pure portal applications, a Workplace Collaboration Services application incorporates people as a key ingredient. It is often a composite application, made up of various components, presented in a single, integrated user experience. The power to assemble the application can even lie with the business user, using the included tool called IBM Workplace Builder.

Composite applications can combine components built on a J2EE platform with those built using Lotus Domino tools. Each of the components of a composite application can potentially be an application in itself. A composite application can be server based, running on WebSphere Portal or IBM Workplace Collaboration Services, or it can be client based, using IBM Workplace Managed Client.
Customization for the user's role
An activity-based model is an added consideration beyond the role-based capabilities outlined in the “Business application requirements” section. Beyond the traditional methods of collaboration, IBM Workplace Collaboration Services introduces the Activity Explorer feature using IBM Workplace Managed Client to more effectively track and manage multiple points of interaction (activities) that relate to a project or process.

IBM refers to these interactions as activity-centric or process-centric collaboration. This new innovation extends collaboration beyond instant messaging and e-mail by allowing teams to proactively record and organize their shared conversations and documents using an easy-to-navigate, structured approach.

Desktop productivity features of a rich client
Lotus Notes provides a graphical window user interface and an extensible client framework for running collaborative and document-centric applications. The following list includes Lotus Notes client features that differentiate it from a Web browser:

- Richly formatted content editing (tables, spell check, inline graphics, full font support, find/replace and more)
- Microsoft Office integration for attaching, storing and distributing files
- Offline and disconnected e-mail, calendaring, personal information management and applications with virtually full functional fidelity while offline and synchronization when reconnected

Consistent with the nature of a portal, WebSphere Portal is a browser-based environment. However, the product offers supplied portlets that allow access to Domino applications and mail. An integrated feature of the portlets allows the user to launch the Notes client from within the portal when viewing Domino data. This offers the benefits of application and data aggregation through portlets while preserving the user’s ability to manage Domino data from within the feature-rich Notes client environment.
IBM Workplace Collaboration Services offers the option to use IBM Workplace Managed Client, a full rich client based on IBM Workplace Client Technology™ software. When used in conjunction with Workplace Collaboration Services, IBM Workplace Managed Client offers the Activity Explorer feature, the built-in productivity tools for Microsoft Office integration described earlier in this paper, and more.

*Offline capability with application fidelity*

Application fidelity has been one of most compelling capabilities of Lotus Notes and Domino. Domino replication services can allow an application to be taken offline and used with full fidelity, through access to data in the local replica using a Lotus Notes client. When the user reconnects, the work he or she performed offline is fully synchronized with the server. Lotus Domino also extends offline support for mail, calendaring and personal information management functions to the Web browser user with IBM Lotus Domino Web Access.

This coveted offline capability is also evolving to the J2EE side of the IBM Workplace environment. With IBM Workplace Collaboration Services, the degree of disconnected support provided by IBM Workplace Managed Client initially focuses primarily on mail, calendar and document capabilities. The offline information is encrypted for high security, using the IBM Cloudscape™ relational database on the local workstation.

*Infrastructure requirements*

Existing infrastructure and organizational elements must also be factored into an application-platform decision. Such elements include enterprise standards and policies, available IT resources, the importance of operating system choice and centralized systems management capabilities, and available application development skills.
Many IT organizations are making strategic decisions regarding their application infrastructure to simplify systems management, to reduce costs of application development and administration, to streamline operations and, perhaps, to consolidate servers. For example, a company may base its strategy on the J2EE application architecture, LDAP directory services and relational data storage.

If your application architecture is governed by such an IT enterprise strategy, then it probably belongs on a J2EE platform, keeping in mind that the use of Lotus Domino might be desirable for features not yet available in the J2EE architecture. Another fact to remember is that you can configure Domino Directory to be an LDAP directory, allowing your existing Domino Directory to participate in your IT policy, IT infrastructure and support environment.

The IBM Workplace model offers several important attributes to support a long-term infrastructure decision. Its J2EE standards-based architecture is open and extensible, allowing for growth and ease of interoperability. Its network-centric application delivery model is designed to help organizations reduce systems management and administration overhead.

The wide spectrum of J2EE application development tools in the IBM software portfolio offers choice and flexibility for developers at every level. The tools range from pure Java- and Eclipse-based tools such as IBM Rational® Application Developer for WebSphere Software — to the visual scripting-level development capabilities of IBM Workplace Designer — to the application assembly of IBM Workplace Builder, designed for the business user. Application components built with these tools are reusable, and the use of open standards allows developers to incorporate components available from ISVs into their composite applications.

Varying degrees of resource overhead are associated with application development, deployment and management. Both IT resources and human resources are factors in system administration, database administration, directory services and security architecture.
Consider an application that needs to be deployed to a field site in a remote, underdeveloped country without adequate local IT support or wide area network (WAN) bandwidth. Along with such infrastructure constraints, budget constraints are another contributing factor. For example, a business unit has identified the need for a solution but is unable to secure adequate funding for IT resources.

Lotus Domino might be an appropriate solution in these instances because it is an integrated, stand-alone application server for Lotus Notes clients as well as Web browsers. If disconnected support is not a requirement, Web browser access to a J2EE environment from a remote region is also an option. However, J2EE-based applications might require the availability of other infrastructure elements in the environment, such as a relational data store that is installed and managed separately from the application server, and might require more IT resources to support the platform.

**Choice of server and operating system platforms**

Deployment of the appropriate hardware and operating system platform is critical to meeting application scalability and availability requirements. Horizontal scalability refers to distributing an application across several physical servers, several server instances in a blade configuration, or even several server instances distributed across logical partitions (LPARs) on a single physical server. Vertical scalability refers to taking advantage of multiple processor configurations by upgrading to a more scalable hardware and operating system platform. It is important to consider the scalability requirements for the future as well as for today.

IBM WebSphere Application Server, WebSphere Portal and Lotus Domino take advantage of vertical scalability because they support a broad range of platforms ranging from Intel® processor-based servers to the IBM® zSeries® mainframe, and can scale across multiple processor configurations of those systems. IBM Workplace Collaboration Services supports Linux for Intel, IBM AIX®, IBM i5/OS®, Microsoft Windows® 2000 and Windows 2003 operating systems, and the Sun Solaris 9 operating environment.
Choice of client options and low-touch client deployment
Lotus Domino supports a variety of client software options, including Lotus Notes and Web browsers. Notes and Domino 6 introduced streamlined deployment capabilities (such as Notes Smart Upgrade, Single Copy Template and Policy Based Administration) that allow you to centrally manage the installation and configuration of users’ machines without visiting a single desktop.

Similarly, IBM Workplace Collaboration Services can be experienced from either a Web browser or IBM Workplace Managed Client. Organizations that embrace Workplace Managed Client and the applications built on it can enjoy the flexibility of client-side applications with the server-side control and related cost savings traditionally associated with Web-based computing.

Developer skills
Many applications need developers to build and maintain them, and system administrators to manage them. An application architecture decision is often determined by the available skills in the organization. Making an assessment of which resources are currently available or obtainable through hiring or contracting is an important factor in determining which platform you choose. Although a good starting point is the type of skills that you have available in-house, it is also important to keep skills up-to-date, which also includes cross-training of skills.

Domino developers can benefit from knowledge of the role that Java can play, and Java developers can benefit from understanding the collaborative power that Domino brings to an application. The third part of this paper describes existing and upcoming tools to help developers cross-train and build integrated applications that take advantage of the benefits of both the Domino and J2EE environments.
Leveraging your investment in Domino

One of the greatest benefits of developing applications for the Domino platform is that many applications written for Notes Version 1 or 2 can be run with little or no modification on a Domino 6.5 server. That means your company might still realize value from an investment that you made 15 years ago, and those applications can now be extended for access by Web browsers, Java APIs and Web services.

As Lotus Domino moves forward as one of the application server platforms for the IBM Workplace portfolio of products, it is important that IBM continues to protect our clients’ investments. The following sections provide information about how to leverage your Domino investment from J2EE applications running on WebSphere Application Server, WebSphere Portal or IBM Workplace Collaboration Services.

Lotus Domino integration with WebSphere Application Server

Within IBM, the Lotus and WebSphere software development teams have coordinated efforts to develop and to deliver key integration points that clients have requested. Because WebSphere Portal and IBM Workplace Collaboration Services are based on WebSphere Application Server, these integration points apply to those two products as well. Each of the requested elements stems from the need for an excellent user experience, easy administration and overall infrastructure integration.
Whereas Java and J2EE standards make it possible to integrate disparate applications at a fairly broad level, IBM has developed powerful integration points for use specifically between Lotus Domino and WebSphere Application Server. For example:

- LDAP has become the de facto standard for directory serving, and is also used by WebSphere Application Server for authenticating users. Because Lotus Domino can be an LDAP server, WebSphere can use this directory protocol, allowing you to manage user identities for Lotus Domino and for J2EE applications in one location, easing the administration skills needed in your IT environment.

- Where WebSphere Application Server and Lotus Domino use the same LDAP directory, you can choose to configure single signon (SSO) between Lotus Domino and WebSphere applications. SSO allows a Web browser user to enter his or her authentication information once. This information is stored in a secure context that both WebSphere and Lotus Domino share and reference as the user navigates between WebSphere and Domino applications.

- In addition, the Java APIs of Lotus Domino allow a Java developer to use the SSO token to create a secure session with Lotus Domino from WebSphere Application Server on behalf of the current Web application user. This eliminates the need for an application to know the specific user name and password when accessing Lotus Domino, and also minimizes a security risk of user passwords being handled by any given application.

Leveraging your Domino data

If you choose to have IBM WebSphere Application Server, IBM Workplace Collaboration Services or IBM WebSphere Portal handle the presentation and application logic, you can still leverage the data that you have in Domino applications. Depending on the J2EE application needs, this can be done using an end-user tool, using a rapid application-development tool or using XML and Java to provide more granular control.
Notes application plug-in for IBM Workplace Managed Client
At the time of this writing, the next major release of Lotus Notes and Lotus Domino is in its third beta release, and is planned to be generally available in the third quarter 2005. Lotus Notes beta 3 includes a technology preview of a Notes application plug-in for IBM Workplace Managed Client. The plug-in is designed to let users run existing, native Notes and Domino applications within the context of IBM Workplace Managed Client, thereby extending the reach and viability of those applications without design modifications.

WebSphere Portal tools
WebSphere Portal supplies a number of portlets, packaged as IBM WebSphere Portal Collaboration Center, that allow you to integrate existing Domino applications into the WebSphere Portal environment. WebSphere Portal Collaboration Center allows personnel with basic WebSphere Portal administration skills and little-to-no programming skills to configure the portlets quickly and effectively. These portlets can display data from the Domino Mail, Calendar and To Do views as well as views from other Domino applications. Using these collaborative portlets, companies can easily extend commonly used Domino applications to the WebSphere Portal environment.

A specialized portlet called IBM Portlet Builder for Lotus Domino adds Domino capabilities to the existing WebSphere Portal Application Integrator framework. Also supplied with WebSphere Portal, the Portlet Builder for Domino enables line of business (LOB) users to build portlets quickly and easily based on Lotus Domino. This wizard-like portlet eliminates the need for programmers to write low-level functions and allows developers to easily bring key Domino applications and data into the WebSphere Portal environment. This portlet builder also includes online awareness and click-to-action capabilities for sharing information between portlets, providing immediate productivity gains for portal users.
Domino application development features that enable integration with J2EE

Lotus Domino also supplies features that facilitate the integration of Domino data with J2EE applications. You can benefit from many application benefits by combining Domino and J2EE applications, most importantly, the addition of human interaction into scalable and transactional applications built on J2EE. When integrating Lotus Domino with a J2EE-based application, there are two ways to access Lotus Domino programmatically:

- **Domino Objects for Java**
- **Domino 6 custom JSP tags**

Domino Objects for Java comprise the Java API of Lotus Domino. These objects allow for object-based access to the data store for essential database activities (create, read, update and delete). The API allows access not only to Domino data but also to some important services, such as registering users, running agents and sending mail. The Domino 6 custom JSP tags are designed for use only from within a JSP. These tags are XML tags embedded in a JSP that provides data access, data input and process control. JSP tags abstract the Domino Objects for Java and help reduce the development time to build a J2EE application that uses Domino data and services.

IBM Rational Application Developer for WebSphere Software enhances support for Domino

IBM Rational Application Developer for WebSphere Software is the primary development tool for IBM WebSphere software-based applications. Rational Application Developer 6.0.0.1 added support for IBM Lotus Domino. Using these enhancements, developers can quickly build a rich Web user interface with reusable, drag-and-drop JavaServer Face components that connect to Domino. Connectivity is handled by the Domino Service Data Object (SDO) mediator. Rational Application Developer includes a set of visual portal development tools and a WebSphere Portal unit test environment, allowing you to build and test individual portlets and entire portal applications.
XML capabilities
Lotus Domino also has extensive XML capabilities that can be used from a J2EE application. A J2EE application can use the XML interface to Domino to access unstructured Domino documents without knowing details about the data. Then, the J2EE application can process the resulting XML and transform it to the specific format needed by the application. When the application is ready to update Lotus Domino, the same XML interface can be reused to update the database, thus providing for a round trip of data in and out of Lotus Domino.

Web services
Emerging Web services standards simplify application integration by providing a standardized access protocol known as Simple Object Access Protocol (SOAP). To do so in a Lotus Domino 6 environment requires LotusScript or Java development skills, knowledge of the Domino object model and writing code to simulate a SOAP engine. Enhanced support for hosting Web services is planned for Lotus Domino 7 which adds a SOAP engine that is used with the Domino HTTP server capability to provide the run-time support.

In addition, beta 3 of Lotus Domino Designer 7 includes a new Web service design element that gives Domino developers the option to code the Web service using either their familiar LotusScript interface or Java. Lotus Domino Designer 7 also includes built-in support for Web Services Description Language (WSDL). With this feature, Domino developers can build a new Web service based on existing WSDL. This can help them to save time and to ensure that the Web service will provide the correct interface for the Web service consumer.

Domino support for IBM DB2 Universal Database
Another significant feature of Domino 7 beta 3 is support for IBM DB2 Universal Database as an alternative to NSF on selected operating system platforms. With this option available on a per-database basis, enterprises can choose to leverage this relational repository for common storage of data for Domino and J2EE applications.
In addition, designers can build applications that blend collaborative services and relational data using Standard Query Language (SQL). Another enhancement allows developers to create *Query views*—a new type of Domino view that uses SQL statements to populate the view. By providing access to Domino data through a relational database interface, leveraging your investment in Domino data can be easier than ever.

**Projecting Domino applications through portlets**

Portlets are the component model for the assembly of IBM Workplace applications. Domino applications can be developed for access through a portlet. From the developer's perspective, this is similar to enabling a Notes client-based application to be accessed from a Web browser. Because IBM Workplace Collaboration Services is built on the WebSphere Portal framework, interest in integrating Domino applications by projecting them in a portlet is likely to increase with new implementations of Workplace Collaboration Services.

Domino Web applications might have various design elements that do not map well to portlets or do not adhere to the portal framework guidelines. Some of these include frames and frame tags, JavaScript namespace issues and URL redirects targeted at Domino and not the portal server. Some of these issues can be addressed by using the iFrame or Web clipping portlets.

The IBM Lotus Domino Application Portlet addresses many of these issues by using the Domino view presentation and application logic, allowing a full Domino Web application to be surfaced in a portlet. This tool directs all client requests through WebSphere Portal, a capability known as *reverse proxy*. The Domino Application Portlet is delivered through the WebSphere Portal Catalog—either as a stand-alone portlet or as a part of the Lotus Notes and Domino Extended Product Portlet package.

Because developing Domino applications for the portal environment is similar to developing Domino Web applications, enhancements to tools used to develop and serve Web applications are also planned. These include enhancements to Domino Designer to allow Domino application development suitable for the portal framework as well as enhancements to the Domino Web engine. See Table 1 for application development objectives and available options.
<table>
<thead>
<tr>
<th>Application integration objective</th>
<th>Available options</th>
</tr>
</thead>
<tbody>
<tr>
<td>Access applications through Notes client</td>
<td>• Use IBM Lotus Domino Designer</td>
</tr>
</tbody>
</table>
| Access applications through Web browser | • To develop applications for Domino HTTP server, use Domino Designer.  
• To develop applications that integrate IBM WebSphere Application Server with Lotus Domino:  
  • Use IBM Rational Application Developer for WebSphere Software  
  • Use Domino custom JSP tags  
  • Use Domino Objects for Java  
  • Use Domino XML support |
| Access applications through WebSphere Portal | • Use the Notes and Domino portlet of WebSphere Portal Collaboration Center  
  • Develop portlets using IBM Lotus Portlet Builder for Domino  
  • Use Domino Application Portlet |
| Access application data | • Use Domino XML support  
• Use Domino custom JSP tags  
• Use Domino Objects for Java |
| Access applications through Web services | • Use Domino to host Web services written in LotusScript or Java  
• Use WebSphere Application Server to host the Web service and use Domino XML support or Domino Objects for Java |
| Move data to relational data store | • Use feature planned for Domino 7 that allows DB2 Universal Database to be an alternative to NSF  
• Use Lotus Enterprise Integrator to transfer data to a relational data store |

*Table 1. Application development objectives and available options*
The evolution of rapid application development

With the introduction of the IBM Workplace family, blended applications that combine capabilities of IBM Lotus Domino, IBM WebSphere Portal and IBM Workplace Collaboration Services will become increasingly common. The mix will vary from company to company, and will be based on your specific business needs. With this in mind, three objectives provide the basis for IBM’s strategy for application development tools:

- Offer tools appropriate to the user task, ranging from Web-browser-based wizards to visual script tools for quick development, to API toolkits and samples for the developer who needs to code directly to the platform.
- Optimize tools for the target server platform when used by itself (Lotus Domino, WebSphere Portal or IBM Workplace Collaboration Services).
- Support real-world environments where integration is important.

The proven rapid application-development capabilities of Domino Designer make it well-suited to continue supporting Domino as it moves forward in the evolution of the IBM Workplace family. IBM Lotus Domino Designer is the premier application-development tool for building Lotus Notes and Domino applications. As stated previously in this paper, the planned future of Domino Designer is one of continued enhancement, especially focusing on features that make it easier to integrate with J2EE applications. These plans include enhancements for Web services, features to take advantage of DB2 Universal Database as a data store, and applications customized for access through the portal environment.

With the growing adoption of the J2EE architecture, the need to provide application tools that appeal to the application development communities beyond Java technology-skilled developers is clear. This requires the delivery of application-development tools that empower Web application developers to build and connect components, data and application business logic, while insulating them from the complexities of the J2EE architecture.

Among Domino Designer, Workplace Designer and the various tools for portlet development, developers have a comprehensive portfolio of rapid application development tools available to them (see Table 2). These tools enable them to build the modular components of composite IBM Workplace applications.
## At a glance: Tools for accessing Domino applications

<table>
<thead>
<tr>
<th>Developer skill level</th>
<th>Type of access</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td>Business user</td>
</tr>
<tr>
<td>IBM Lotus Domino tools</td>
<td></td>
</tr>
<tr>
<td>IBM Lotus Notes (From the action bar, click File Database New and choose an out-of-the-box template)</td>
<td>X</td>
</tr>
<tr>
<td>IBM Lotus Domino Designer (separate product)</td>
<td></td>
</tr>
<tr>
<td>IBM Lotus Workflow™ (separate product)</td>
<td></td>
</tr>
<tr>
<td>Domino Objects for Java (supplied with Domino)</td>
<td></td>
</tr>
<tr>
<td>Domino 6 custom JSP tags (supplied with Domino)</td>
<td></td>
</tr>
<tr>
<td>Lotus Collaborative Components API (supplied with Domino)</td>
<td></td>
</tr>
<tr>
<td>IBM WebSphere Portal tools (supplied with WebSphere Portal or available from WebSphere Portal Catalog)</td>
<td></td>
</tr>
<tr>
<td>Notes and Domino portlet in IBM WebSphere Portal Collaboration Center</td>
<td>X</td>
</tr>
<tr>
<td>IBM Portlet Builder for Domino</td>
<td>X</td>
</tr>
<tr>
<td>Domino Application Portlet</td>
<td>X</td>
</tr>
<tr>
<td>IBM Workplace tools</td>
<td></td>
</tr>
<tr>
<td>Notes application plug-in (planned to be supplied with Notes 7)</td>
<td>X</td>
</tr>
<tr>
<td>IBM Workplace Builder (supplied with IBM Workplace Collaboration Services)</td>
<td>X</td>
</tr>
<tr>
<td>IBM Workplace Data Access (available to users accessing IBM Workplace Collaboration Services through the optional rich client, IBM Workplace Managed Client)</td>
<td>X</td>
</tr>
<tr>
<td>IBM Workplace Designer (planned for availability in mid-2005-delivery vehicle to be determined)</td>
<td></td>
</tr>
<tr>
<td>Workplace Collaboration Services API Toolkit (available at ibm.com/lotus/ldd/wpsapi for use with Workplace Collaboration Services)</td>
<td></td>
</tr>
</tbody>
</table>
IBM software development platform tools

<table>
<thead>
<tr>
<th>Developer skill level</th>
<th>Type of access</th>
</tr>
</thead>
<tbody>
<tr>
<td>Business user</td>
<td></td>
</tr>
<tr>
<td>Visual script developer</td>
<td>• Build business logic that can be used to integrate with Domino.</td>
</tr>
<tr>
<td>Highly skilled programmer</td>
<td>• Develop applications based on IBM Workplace Client Technology.</td>
</tr>
<tr>
<td>X</td>
<td>Build portlets that can be used to integrate with Domino.</td>
</tr>
</tbody>
</table>

Table 2. When to use which tool

Conclusions

Many IBM Workplace offerings are already, or soon will be, available in the market. It is not too soon to learn how to leverage them in your organization. You are likely concerned about making an appropriate decision that can result in a high return on your investment. Consider that some underlying elements of the architecture that supports IBM Workplace software are applicable to any platform that you choose.

For example, LDAP has become the de facto standard for corporate directories. If you are using IBM Lotus Domino today, Domino Directory can serve as your corporate LDAP directory. It can also serve as the directory used by IBM WebSphere Application Server (either stand-alone or as the integrated J2EE platform for IBM Workplace Collaboration Services) and IBM WebSphere Portal.

Web services and XML are also an increasingly important part of a forward-looking IT infrastructure. WebSphere Application Server can be your Web service application-hosting platform. Using the integration options described in this paper, Domino can provide data and content to Java technology-based Web services.
WebSphere Portal offers the WSRP capability, which allows a portlet to act as a Web service. Therefore, any application served through the portal can act as a Web service to another application. XML is the language of choice for Web service data markup, and Lotus Domino, IBM Workplace Collaboration Services, WebSphere Application Server and WebSphere Portal all support robust XML functionality.

When looking to leverage your existing Domino investment, there are many options. Simple configurations that enable Domino to interoperate with WebSphere Application Server and WebSphere Portal, such as a common LDAP directory and SSO, are a great start. From there, application integration through the use of Domino data within a WebSphere platform-based application can help an organization standardize on an application platform. If you are still running an older version of Lotus Domino, such as Release 5, it is time to consider an upgrade. More current releases include enhancements to ease integration.

IBM recommends that your Domino and J2EE developers work together to build applications. Domino developers bring unique skills and experience in building user-centric collaborative applications. Java developers have experience in systems programming that can be valuable to extend Domino applications to larger-scale deployments beyond the enterprise. These teams can tap each other’s strengths to integrate collaboration into more areas of the business and beyond the corporate boundaries. Your company may benefit more quickly from the power and flexibility of composite applications in an IBM Workplace environment.

For more information
To learn more about IBM Lotus Notes and Domino, visit:
ibm.com/lotus/notesanddomino
To learn more about how Lotus Notes and Lotus Domino fit in the IBM Workplace strategy, visit the IBM Lotus Software Advantage page at
ibm.com/lotus/byproducts
To learn more about IBM Workplace, visit:
ibm.com/software/workplace
To learn more about WebSphere Portal:
ibm.com/software/websphere/portal

© Copyright IBM Corporation 2005
Lotus Software
IBM Software Group
One Rogers Street
Cambridge, MA 02142
U.S.A.
Produced in the United States of America
5/05
All rights reserved

References in this publication to IBM products or services do not imply that IBM intends to make them available in all countries in which IBM operates.

AIX, Cloudscape, DB2, DB2 Universal Database, Domino, Domino Designer, @server, Everyplace, i5/OS, IBM, the IBM logo, Lotus, Lotus Enterprise Integrator, Lotus Notes, Lotus Workflow, Rational, SmartSuite, Workplace, Workplace Client Technology, Workplace Collaboration Services, Workplace Managed Client, WebSphere and zSeries are trademarks of International Business Machines Corporation in the United States, other countries or both.
Java and all Java-based trademarks and logos are trademarks or registered trademarks of Sun Microsystems, Inc. in the United States, other countries or both.
Microsoft and Windows are trademarks of Microsoft Corporation in the United States, other countries or both.
Intel is a trademark of Intel Corporation or its subsidiaries in the United States, other countries, or both.
Linux is a trademark of Linus Torvalds in the United States, other countries or both.
Other company, product, and service names may be trademarks or service marks of others.
The information contained in this documentation is provided for informational purposes only, and is provided as is, without warranty of any kind, express or implied.
In addition, the information is based on IBM’s current product plans and strategy, which are subject to change by IBM without notice. IBM shall not be responsible for any damages arising out of the use of, or otherwise related to, the information provided herein or any other materials.
Nothing contained in this documentation is intended to, nor shall have the effect of: 1) creating any warranties or representations from IBM or its suppliers or licensors, or altering the terms and conditions of the applicable license agreement governing the use of IBM software, or 2) stating or implying that any activities undertaken by you based in whole or in part on the information provided herein will bring about any specific benefits or other results. Any action undertaken by you based on these materials is at your sole discretion and risk.
All statements regarding IBM future direction or intent are subject to change or withdrawal without notice and represent goals and objectives only.