IBM e-business Infrastructure
August 2001

Facilitating the application development process using the IBM Patterns for e-business.

John G. Lord, Jr.
Certified Consulting IT Architect
IBM Software Group

Contents

The importance of best practices in application development

The Internet is an excellent medium for communication, but, more importantly, it has evolved into a powerful business tool, a place where companies get real work done, establish real competitive advantage, and generate real growth and profits. While the Internet has created entirely new paradigms for doing business, successful companies—dot-coms or otherwise—require sound, proven business models to take advantage of this new medium. In March of 2000, Lou Gerstner, the chairman of IBM, predicted a major fallout in the Internet industry, not because the technology or medium itself was inherently flawed but because many companies experimenting with e-business were not adhering to sound business practices. Many of these businesses failed not only due to unsound business practices but also due to inadequate systems—their systems were unable to handle the volume of business, to maintain consistently high-quality service or to provide customers with reasonable response times. The unfortunate changes in the industry during the past year and a half have proven Gerstner correct and underscored the importance of using best practices in the development of successful e-businesses.

This rule of using best practices also applies when developing the applications and systems that run today's e-businesses. Application developers require proven techniques for building systems to take advantage of this new communications and business medium. While many developers maintain the maxim that software engineering is a craft or an art, certainly much hard work goes into building successful applications. For instance, many Apple end users tout the relative ease in which they build applications with a Mac; however, they possibly fail to recognize the enormous number of man-hours that went into creating the programs that make the Mac so user-friendly. Getting machines to reliably and consistently do our bidding requires some

Highlights

Just as in every other profession, application developers can learn from the triumphs and failures of their predecessors.

very hard work. Often that work is based on the vision of one person or a small group of people who have the skill and knowledge to design and build a robust, reliable software product. Many software projects are never completed or do not succeed in the marketplace, and companies and developers can learn valuable lessons by studying the ways in which others have either wasted money, time and effort or been successful.

Just as in every other profession, application developers can learn from the triumphs and failures of their predecessors. Although no one right way exists for application development, as Frederick Brooks notes in his often-quoted paper "No Silver Bullet"[1] and further illustrates in *The Mythical Man-Month*,[2] Brooks also points out that successful software projects share the following characteristics:

- *A lead architect or at most a small surgical team that can completely envision what the completed system will look like and how it will work*
- *Clear, well-defined project goals*
- *Management support (for both users and information systems)*
- *A realistic schedule (Brooks is a proponent of allowing time to throw away the first version of a software product before anyone sees it.)*[3]

When companies apply these characteristics to their software projects, they have a greater chance of achieving the important objectives of minimizing risks, reducing time to market, and lowering and controlling costs. This vision and the project plan must be shared with all team members to enable their success.

Highlights

Companies need an effective plan to enable successful application development.

Even when companies seem to follow the above tenets closely, application development projects can still fail. Many companies attempt to apply existing processes ("this is the way we've always done it") and integrate existing applications ("if it's not broke, don't fix it") to the development process. Often, organizations fear straying from the existing corporate culture or introducing technology that may threaten developers. Sometimes organizations run into problems by fundamentally changing the way they do business as they integrate the Internet into their business models. Projects can fail when the initial scope is too large and when companies vastly underestimate the infrastructure needed to scale their business for rapid growth.

Companies need an effective plan to enable successful application development, which is predicated on understanding a current need, building the existing project to address that need and adding the appropriate infrastructure to allow the necessary access by customers and business (or trading) partners. Development must also properly consider data and application security and integrity. IBM, in its continued efforts to lead in the creation, development and manufacturing of the industry's most advanced information technologies, shares its best practices to help enterprises develop applications.

The IBM Patterns for e-business are a set of reusable architectures that help reduce risk, shorten time to market and generate a more significant return on investment.

The IBM Patterns for e-business

IBM has compiled the collective wisdom and experience gained from more than 20,000 successful Internet-based engagements and transformed that wisdom into the IBM Patterns for e-business. These Patterns provide the best-practice blueprints and tools to facilitate the application development process and enable companies to shorten time to market, reduce risk and, in general, see a more significant return on investment. While the IBM Patterns for e-business do not advocate one methodology over another, they help companies understand the true scope of their development projects; employ best-practice techniques; and use a manageable, repeatable, reliable approach to the development of e-business applications, significantly increasing the likelihood for success.

Highlights

The Patterns Web site provides detailed, usable information to enhance each stage of application development.

IBM has a Web site for the Patterns for e-business (*ibm.com/framework/patterns*), which is designed as a self-help tool to assist application development teams in using the Patterns. The Patterns Web site provides detailed, usable information to enhance each stage of application development.

Application development

Methodology

Regardless of the method used for application development, or of a project's size or complexity, most projects follow, either formally or informally, a set of well-worn steps:

- *Requirements gathering*
- *Analyzing the requirements*
- *Designing a solution*
- *Building or buying the software*
- *Testing.*

Highlights

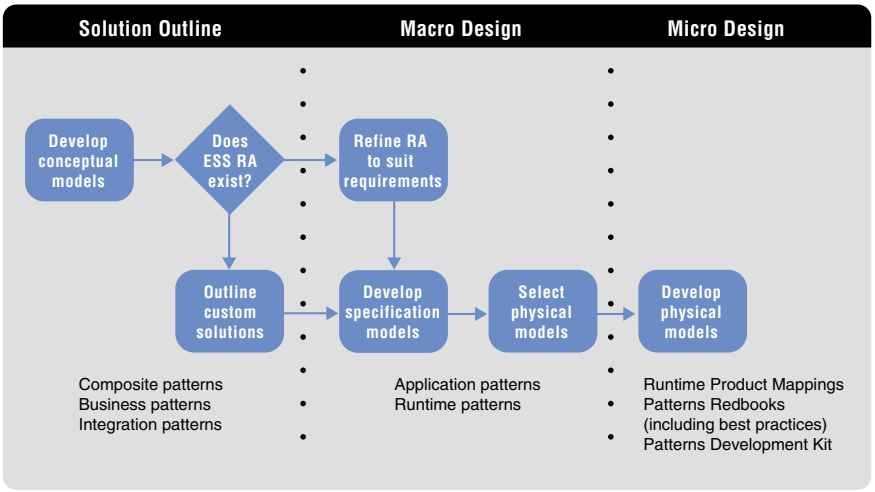
No matter the methodology driving an application development project, the major steps in successful projects are essentially the same.

Many methodologies for developing applications exist; however, most of these are probably variations of three models:

- *The data-centric model, championed by James Martin in his writing on information engineering*[4]
- *The process-centric model, based on information analysis (data-flow diagramming)*
- *Various object-based models, which incorporate a combination of components from the first two models and the documenting/modeling tool UML (Unified Modeling Language).*

No matter what methodology drives an application development project, the common thread is process, and successful development will consistently use the major steps mentioned above.

The diagram below illustrates a method for building e-business applications that IBM Global Services has refined over thousands of customer engagements. More importantly, the diagram depicts the intersection of process and methodology, and how any methodology can make use of the IBM Patterns for e-business at each major step in the development process. In this particular example, IBM Global Services combines its Enterprise System Structures (ESS)



Highlights

A well-understood methodology is a basic requirement for success.

and Reference Architecture (RA) with the Patterns for e-business to detail a complete set of proven, well-understood and reusable components for a typical application development engagement. (ESS and RA are IBM Global Services assets and are available only through IBM Global Services engagements.) A well-understood methodology is a basic requirement for success; using processes derived from the experience of many successful engagements further ensures success.

Most projects, including those using one of the three methodologies described above, begin with requirements gathering, which entails determining what the application is meant to accomplish and which existing systems, data stores and other portions of the current infrastructure will be used. This is an ideal time to clearly establish functional boundaries to limit the scope of the application or project.

The tasks of application developers and product managers generally diverge at this point. The application developers analyze requirements, develop a design based on the given constraints (such as time, budget and existing systems) and produce specifications for each of the components or subsystems to be developed. Product managers concurrently develop test cases or acceptance criteria, the information the developers can use to determine if the system is functioning properly and whether it can accomplish all of the required tasks.

The requirements-gathering stage is an ideal time to clearly establish functional boundaries to limit application or project scope.

With both the requirements and specifications in hand, developers can begin to build and unit-test the new application. Developers can analyze test results by comparing the design documents and specifications with the expected results developed by product managers. As developers complete major portions of the new application, integration testing can begin to determine if the various pieces work together and to ensure they work with existing applications, databases and infrastructure.

Highlights

Simply put, experience is the best way to learn what works and what doesn't.

Process

Methodologies generally specify the steps to be taken and enumerate the work products to be produced. In general, they do not specify the actual process to be used to complete the specified tasks nor do they specify how a project is to be managed. A methodology may or may not include the specific process for each of the outputs to be produced. This is the process of using the methodology. In addition, successful projects will have other processes in place to handle tasks such as change management, progress reporting and tracking completion of assigned tasks. It is this aspect of a methodology, the details of how the work is to be done and how the required ancillary tasks are to be completed, that provides the additional detail necessary for success.

Simply put, experience is the best way to learn what works and what doesn't. As Watts Humphrey advocates in the book *Managing the Software Process*, part of the Software Engineering Institute Series on Software Engineering, an information technology organization must be able to evaluate its processes and learn as it goes about the business of constructing software.[5] Organizations should also employ internal mentorship and process reviews to help educate junior members and adjust procedures for continued improvement; learn from outside consultants, who can kick-start the process; and look outside the enterprise to learn from the experiences of others who have worked or are working to solve similar problems.[6]

An organization that does not learn from its mistakes is bound to repeat them.

An organization that does not learn from its mistakes is bound to repeat them. Initiating a system in which developers, designers and analysts can learn from each other and others creates the foundation for a more effective application development process. Hiring outside consultants is another excellent way of gaining additional knowledge and experience to help guide the process. However, many consultants use only proprietary methodologies and tools, and may or may not have the appropriate experience to lead a large development project. The most reputable organizations can provide invaluable assistance and experience based on their many successful engagements.

Highlights

IBM has taken learning from the experience of others to an entirely new level.

Learning from the experience of others

No matter the methodology, the entire application development process is fraught with challenges. Many companies embark on application development projects feeling as though their situation is wholly unique. This may be true in some ways, but wisdom comes with time and experience. When a consultant hears a question for the first time, his or her answer may not hold much value, but after the consultant hears the same question 25 times, audiences will marvel at the value of his or her wisdom. This is not to say that application development projects do not differ, but rather to emphasize how much they all have in common. IBM research shows that in almost all cases, only 20 percent of an application development project is unique; 80 percent of a project can be approached with well-proven software and techniques. Therefore, learning useful techniques that increase the chance for success can save significant time and resources.

IBM has taken learning from the experience of others to an entirely new level. The Patterns for e-business, created from the documentation and analysis of thousands of successful IBM application development projects, give businesses a set of proven, reusable architectures that can guide the design, development, implementation and extension of e-business applications. They match business challenges with Business and Integration patterns, use proven Application and Runtime patterns, populate the Runtime patterns with pretested Runtime Product Mappings, and establish best-practice guidelines for application design, development and management.

The four basic Business patterns provide tangible solutions to the most frequently encountered business challenges.

The four basic Business patterns

IBM has distilled its extensive application development knowledge into four primary Business patterns. These patterns, organized by application function, provide tangible solutions to the most frequently encountered business challenges by identifying common interactions among users, business and data. The four primary Business patterns are also detailed in a new book entitled *Patterns for e-business: A Strategy for Reuse* (available from the Patterns Web site).

Highlights

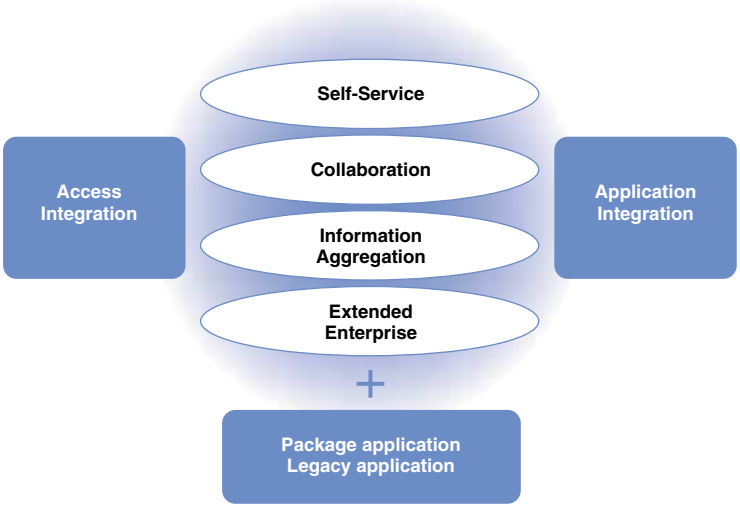
Developers can match business challenges to the most appropriate Business pattern.

The Patterns document proven designs and new directions in reusable e-business architectures:

- *Self-Service (also known as User-to-Business)—applications where users are interacting with enterprise transactions and data*
- *Collaboration (also known as User-to-User)—applications where tools facilitate communication among users*
- *Information Aggregation (also known as User-to-Data)—applications where tools extract information from other data sources.*
- *Extended Enterprise (also known as Business-to-Business)—applications that integrate programmatic interactions among organizations.*

Developers first match a business challenge to one of these high-level Business patterns. For the development of more in-depth systems, the Patterns have an elegance that allows highly flexible combinations. Some complex systems could use all four patterns or a combination of these patterns, plus one or more Integration patterns. IBM has also developed two common Integration patterns to better enable end use and to leverage the value of existing applications or systems:

- *Access Integration—applications that enable access from multiple channels (devices) and integrate common services required to support a consistent user interface*
- *Application Integration—applications that call for the integration of Web-based solutions to core business systems and databases.*



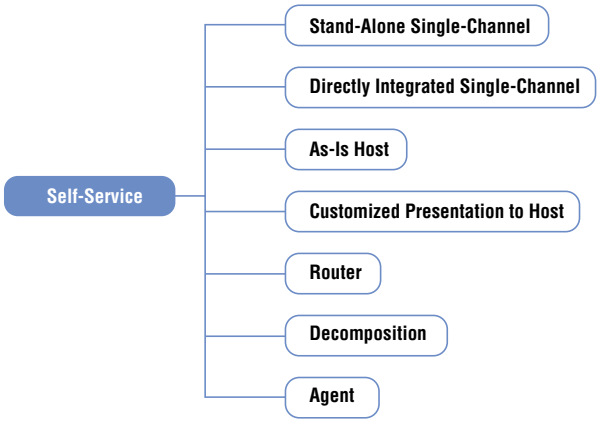
Highlights

Companies can use the Patterns to scale applications to specific business demands with relative ease.

A common development scenario using the Patterns proceeds as follows: A team is building an online sales system. Thus, they begin with the Self-Service pattern to set up the initial interface that enables potential customers to take orders. Now, the team needs the system to integrate with existing applications to process orders. They then, using the Collaboration pattern, add capabilities to provide direct customer service support and communication within the enterprise. If the team wanted to allow customers to access data that would allow them to check on the status of an order or make inquiries directly to other data stores and applications, they would use the Information Aggregation pattern, quite possibly coupled with the Application Integration pattern. The team also wants to support supply chain management, so they use the Extended Enterprise pattern. Lastly, because they want to allow access from a variety of devices, the team turns to the Access Integration pattern to develop an end-to-end solution.

Lower-level patterns provide the in-depth, nuts-and-bolts roadmaps for each process step.

This high-level example illustrates the simplicity of the use of the Patterns and points to the relative ease in which companies can use them to scale applications to business demands, leveraging the ability to "start slow, grow fast." The above example depicts only the skeleton of what the Patterns and Patterns Web site provide. Clearly, developers recognize that the devil is in the details. Once a development team selects the appropriate Business pattern, the Patterns offer an index into a set of lower-level patterns that provide the in-depth, nuts-and-bolts roadmaps for each step in the process. Each pattern breaks down into an increasingly complex series of patterns as shown in the Self-Service example below:



Highlights

The Application pattern can be used to develop how application components and data interact.

The Patterns provide assistance in all phases of the development process, beginning with requirements gathering. As a development team puts together requirements, the Patterns Web site helps match those requirements to the appropriate pattern. As the team refines the requirements and determines which existing systems, data stores and infrastructure will be integrated into the system, they can use the Application pattern to develop how application components and data within a business solution interact.

After choosing the Application pattern, the team can match Runtime patterns topology based on the existing environment and business needs. The Runtime pattern establishes the components needed to support the chosen Application pattern. Without advocating a particular vendor, this pattern defines the logical middleware nodes, their roles and the interfaces among these nodes in order to meet business requirements. The Runtime pattern documents what must be in place to complete the application but does not specify product brands.

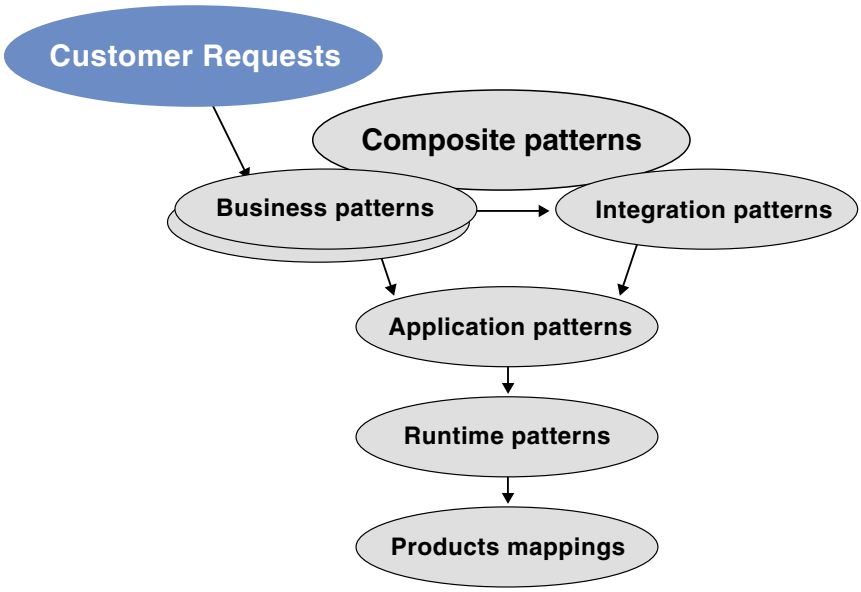
The Runtime pattern defines the logical middleware nodes without advocating a particular vendor.

Developers must now determine which products to use for the actual development. At this point, the Patterns provide a wide range of options and lend the developer significant assistance. Developers can access from the Patterns Web site Runtime Product Mappings that identify tested, optimal software implementations for each Runtime pattern. These include implementations on Microsoft® Windows NT® and IBM AIX®, AS/400® and OS/390® platforms. The Patterns Web site details the products needed for each of the implementations. Developers should also consult the IBM WebSphere® Developer Domain High Volume Web Sites Zone for additional guidance on planning, designing and building e-business solutions for Web sites supporting high-volume workloads and dynamic, volatile data (www7b.boulder.ibm.com/wsdd/zones/hvws).

Highlights

Developers can access the Patterns Web site for a wealth of resources about other, similar development efforts.

Associated with each Runtime Product Mapping on the Web site are best-practice application, design, development and management guidelines that have been gleaned in the process of developing these patterns. Developers can use them to access a wealth of information about other, similar development efforts. The Patterns Web site also provides links to suggested documents on the IBM Redbooks Web site (*www.redbooks.ibm.com/*). The Redbooks site offers a great deal of knowledge and provides developers with access to the sources they need for their project. The site also lists a large offering of available books covering many other topics.



Highlights

The IBM Patterns Development Kit (PDK) is a step-by-step guide that enables the quick creation of Web applications.

As an additional aid for developers, the Patterns Web site provides access to the IBM Patterns Development Kit (PDK) and a tutorial CD. The tutorial provides information about the Patterns and answers questions about the PDK. The PDK is a step-by-step guide that enables developers to quickly create Web applications based on the Self-Service Business pattern—an application design that allows business users to interact with enterprise transactions and data. The PDK provides a functional, best-practice implementation of the Stand-Alone Single-Channel Application pattern—and includes sample application code, installation scripts and wizards to help reduce development time and risk. This Application pattern provides a structure for applications that currently do not need to be integrated with other systems and that focus on only one delivery channel. While this Application pattern can be used to implement any one of the delivery channels, the focus is primarily on the Web channel.

The IBM Patterns for e-business provide the following layers of reusable assets:

Business problem	• Business Pattern
Business drivers	• Application Pattern • Runtime Pattern
Products—any platform	• Runtime Product Mappings • Performance Considerations • Technology Options • Application Design Guidelines • Application Development & Deployment Guidelines • SM Guidelines • Practical References
Code	• Pattern Development Kit (including skeleton demo)

Highlights

The IBM Patterns for e-business establish a common terminology to facilitate better communication.

The Patterns development process is a live project, ever-evolving and being updated as new products are released and are used in the building of real applications. IBM has constructed the Patterns and the Patterns Web site to enable development teams to work through the development process using their preferred methodology or the methodology suggested by consultants engaged to assist in the project. IBM expects development teams to customize the Patterns for the specific objectives of the application in development.

The value of common terminology

An additional benefit of the IBM Patterns for e-business is that they establish a common terminology from the project's onset. Based on his engagements leading and teaching others how to develop systems, Kent Beck has emphasized the importance of using common terminology.[7] It is critical to the process that all team members understand specifically what is meant by a term or phrase. When the entire team knows exactly how to describe the completed system and its components, they can help educate management about the magnitude of the effort and the associated costs. Also, analysts, designers and developers all know exactly what is implied and required when describing and discussing the work, providing tremendous value by eliminating misunderstandings and facilitating better group communication and design efforts.

Building applications with proven experience

As part of the IBM Framework for e-business, the Patterns allow the initial development of a strategic infrastructure.

The Patterns are encompassed within the IBM Framework for e-business, which comprises standards, proven practices and a portfolio of products for developing, deploying and managing e-business applications. As a component of the Framework, the Patterns further enhance the development process by helping ensure that the application supports business objectives, significantly reducing cost and risk. The Patterns also help accommodate scaling business needs by allowing the initial development to form a strategic infrastructure that serves as the foundation for subsequent, more complex application development projects.

Highlights

The Framework provides a vendor-neutral, scalable, open-standards-based platform.

Additionally, the IBM Framework is based on widely accepted open standards and therefore does not preclude the use of software supplied by other vendors, nor is it tied to proprietary hardware. Because most enterprises have multiple operating systems with disparate devices and platforms, this vendor-neutral, scalable, open-standards approach is ideally suited for today's complex enterprise environments. Enterprises and development teams looking for an effective strategic plan and platform that leverage the value of existing infrastructure while effectively planning for the future will find immense value in the IBM Patterns and the Framework for e-business.

The IBM Patterns and the Framework for e-business have translated invaluable experiences into practical tools for any application development project. Most development projects will fall into one of the following three categories:

- *Tactical projects—budget and time constraints dictate the use of as much vendor-supplied content as possible*
- *Major projects—the high level of complexity dictates the approach*
- *Pathfinder for strategic projects—tactical decisions about how Web applications will be built and incorporated in the future help determine the methodology, process and infrastructure for a current project.*

A proven, reliable framework for application development can expand to handle both larger requirements and higher customer volumes.

No matter what type of application project an enterprise is working on today, it is prudent to base the initial project on a proven reliable framework that thoroughly considers security and integrity and can expand to handle both larger application requirements and higher customer volumes.

Several companies promise out-of-the-box solutions for a beginning, tactical project; however, often the solution puts the application into a box and fails to provide the best-practice processes for successful development. Development teams building their first Web-based application will find a veritable library of tools, code and best-practice assistance in the IBM Patterns and on the Patterns Web site. Using the provided Patterns and IBM WebSphere tutorials coupled with the PDK, a self-configuring, end-to-end skeleton Web application, developers can easily deploy a best-practice implementation of an initial user-

Highlights

IBM offers everything a development team needs for building beginning, tactical applications that can serve as the basis for larger, more complex projects.

to-business application. IBM also provides the intellectual capital, servlets and Enterprise JavaBeans™ necessary to link to IBM DB2®, CICS® and IMS™, as well as MQSeries® and Lightweight Directory Access Profile (LDAP) directories. This can help a team construct applications with user authentication and authorization, and links to back-end systems. Developers will find even more in-depth information about user-to-business applications in the IBM Redbooks, accessible through the Patterns Web site. Simply put, IBM offers everything a development team needs for building beginning, tactical projects that can serve as the basis for more complex projects.

For more complex application development projects, IBM offers a wide range of resources.

- *Development projects using WebSphere will benefit from valuable information available in the additional WebSphere tutorials on the WebSphere Developer Domain (see* IBM Version 3.5 Self-Study Guide: VisualAge® for Java™ and WebSphere Studio *at* www7.software.ibm.com/vad.nsf/Data/Document4488*).*
- *A great deal of education is also available via the developerWorks site, which offers a considerable amount of free education (*www-106.ibm.com/developerworks/training*).*
- WebSphere Version 4 Application Development Handbook *includes a complete description of a WebSphere-based application development process, which is suitable for more complex application development, in Chapter 3 (*http://publib-b.boulder.ibm.com/Redbooks.nsf/RedpieceAbstracts/sg246134.html*).*

Highlights

To aid rapid application development, IBM Global Services is reusing components and frameworks based on the Patterns.

In addition, IBM Global Services is harvesting components and frameworks to aid rapid application development. The project is titled Enterprise Application Development Frameworks for Java (JADE). Because the IBM Global Services team not only uses the Patterns for e-business to define the project but also offers a wealth of hardened, proven J2EE™-compliant code and complex pieces culled from previous engagements, developers can more efficiently build larger, complex applications. This enables the completion of projects in less time and at a lower risk to the customer. IBM Global Services licenses the components and provides the source code to licensed customers, allowing future enhancements as needed. In other words, IBM Global Services, via the JADE project, is able to bring best practices to code development by the reuse of known components and frameworks that are based on the Patterns for e-business.

Finally, IBM has harvested a great deal of knowledge from many of the largest most complex Web sites in the world and has published this information on the High Volume Web Site Zone in the WebSphere Developer Domain (*www7b.boulder.ibm.com/wsdd/zones/hvws*). This site provide development teams with a wealth of information on how to design for performance, evaluate existing Web sites for performance bottlenecks and modify existing Web sites to maximize performance. In addition, tools needed to evaluate the performance of existing Web sites are also available on alphaWorks (*www.alphaworks.ibm.com*).

Conclusion

The application development process greatly benefits from a best-practice approach.

Application development, regardless of the complexity of the project or the methodology, will benefit greatly by employing a best-practice approach to the process. While some aspects of an application development project may be unique, all development process generally follow the same major steps. Through more than 20,000 successful e-business engagements, IBM has learned that 80 percent of all application development is essentially the same, and in its mission to lead in the creation, development and manufacturing of the industry's most advanced information technologies, IBM has analyzed the best-practices to help enterprises develop applications successfully. This in-depth analysis has led to the IBM Patterns for e-business, a set of proven, reusable architectures that can guide the design, development, implementation and extension of e-business applications.

Highlights

The IBM Patterns for e-business provide the tools, knowledge and guidance for success.

The IBM Patterns for e-business leverage the value of others' experiences, provide a common reference terminology and facilitate the application development process, saving valuable time and resources, reducing time to market, helping ensure success and generating a better return on investment. These Patterns match high-level business objectives to four essential patterns that provide more in-depth sets of topologies for a comprehensive approach. The four Business patterns can be used alone, combined with each other and the two Integration patterns, or modified to build simple tactical projects or the most advanced, integrated applications. The Patterns Web site, PDK and additional resources provide access to tutorials, basic code and intellectual capital, so a development team can confidently address any project.

As part of the IBM Framework for e-business, the IBM Patterns set the foundation for scalable e-business objectives. Companies can find the tools to build their first Web-based application, or link to additional services and products that will enhance even more complex projects. Using open-source standards, the Patterns take a vendor-neutral approach that accommodates today's complex environments and leverages the value of a company's existing infrastructure investment. Therefore, whether a company needs an efficient off-the-shelf solution or a comprehensive, integrated end-to-end set of solutions, the IBM Patterns for e-business provide the tools, knowledge and guidance for success.

IBM offers industry-specific knowledge, thought leadership skills and the ability to provide sound business solutions.

Relying on the experience of a global team

IBM offers industry-specific knowledge, thought leadership skills and the ability to provide sound business solutions. We pride ourselves on our extensive e-business engagement experience, expertise in managing information technology infrastructures, teamwork with alliance and business partners, and global presence and reach. We help companies worldwide design and implement real-world solutions that achieve measurable business results.

For more information

For more information on the IBM Patterns for e-business, visit:

ibm.com/framework/patterns

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

Printed in the United States of America
08-01

Footnotes

1. Brooks, F. P.; "No Silver Bullet—Essence and accidents of software engineering," *Information Processing 86*; H.J. Kugler, ed. Amsterdam: Elsvier Science (North Holland); 1986, pp 1069 -1076.
2. Brooks, F. P., Jr.; *The Mythical Man-Month—Essays on Software Engineering, Anniversary Edition;* Addison-Wesley, 1995.
3. *Ibid.*
4. Martin, James; Information Engineering: Introduction, Vol. 1; Prentice Hall; March 1989.
5. Humphrey, Watts S.; *Managing the Software Process;* from the Software Engineering Institute Series in Software Engineering; Addison-Wesley, 1990.
6. *Ibid.*
7. Beck, Kent; presentation at OOPSLA, San Francisco; 1996.

GC22-0252-00