Java Project Management: Agenda

- Java: An Enabling Technology
- Is Java a Fad?
- Converting Existing Business Systems
- Managing a New Technology
- Is Java Right for Everyone
- Bridging Procedural and OO Styles
- Building Execution Architectures
- From Applets to Multi-Tier Systems
- Building Complex Java-Based Systems

Java Project Management

- Help business and technical managers understand the advantages and challenges in planning, developing, and deploying Java-based business solutions
Java: An Enabling Technology

• Through multiple deployment options, companies have more flexibility in meeting the needs of a varied user or customer community
• The promise of low-cost network computers is enabled by Java’s ability to bring business application functionality and capability to this platform

Java: An Enabling Technology (continued)

• Through platform independence, Java gives the opportunity to reduce operational cost and improve service delivery by migrating the business application to the best price/performance platform
• Through Java technology’s simple integration with Internet technologies, companies can easily and effectively develop rich and flexible intranet apps

Is Java a Fad?

• Adoption of Java technology has mirrored the rise in interest in the Internet
• Software companies have incorporated Java products into their strategic plans
• Java computing has been embraced by a wider community of users and developers than has any other development technology
Is Java a Fad? (continued)

- Time-to-market reductions can be achieved by leveraging the experience and tools built from different parts of the Java community

Converting Existing Business Systems

- Most business systems require costly and time-consuming upgrades sometime during their lifecycles
- The most successful and least expensive conversions begin with a “clean” or well-written existing system

Converting Existing Business Systems (continued)

- A structured approach should be developed and used when converting legacy systems
- There are several advantages of converting existing systems to Java, but they must be weighed against the costs of conversion
- Java’s similarity to C and C++ makes conversion from these languages relatively straightforward
Converting Existing Business Systems (continued)

• Conversion of application code to Java takes significantly less time than building the original system. The time required to test the converted system may be similar to the amount of time spent in the original application build.

Managing a New Technology

• Despite the many advantages of Java computing, there are risks resulting from the emerging nature of the technology.
• Java technology is popular. Because of the extreme interest in the technology, companies must carefully plan and implement a strategy of support from technology vendors.

Managing a New Technology (continued)

• Java development tools are emerging and may not be equal (in terms of features and functionality) to development tools for other languages currently in use by the company.
• The integration of third-party products, especially those based on C/C++ may be a challenge as vendors begin to adopt Java technology. Such integration requires skilled resources if performed internally.
Managing a New Technology (continued)

• Owing to the pace of its acceptance and release cycle, companies should accept and plan for challenges in the Java implementation. Although these problems can be overcome, they will result in some delays in application development and will require skilled resources.

Managing a New Technology (continued)

• Deep Java skills are scarce. As Java technology is adopted by more and more companies, the skill base will increase but the demand will still be great. A company should carefully consider where it will obtain the necessary skills to complete its project.

Managing a New Technology (continued)

• Java technology is well suited for building and deploying business systems; however, the overall experience in this area is immature when compared with technologies such as C, C++, and COBOL. This will result in application development delays as companies are forced to learn from experience in evolving their initial Java-based business applications.
Managing a New Technology (continued)

- Companies should develop a risk management strategy to mitigate the risks of using Java technology and ensure successful development of Java-based business systems.

Is Java Right for Everyone?

- Has Java technology matured to a point to which the company can successfully implement a Java-based business system?
  - How important to the company is it to be a technology leader?
  - Have developers been experimenting with Java applications?
  - Do strong third-party products exist to meet business requirements?

Is Java Right for Everyone? (continued)

- Has Java technology matured to a point to which the company can successfully implement a Java-based business system?
  - Does the company have a culture of “filling the gaps” with new technologies?
  - Does the company have the fortitude to be working in a fast-paced rapidly changing environment?
  - Does the company have a “green field” or new project?
• Does the company have the related technology skills and infrastructure to deliver, deploy, and operate a complex, critical Java-based business system
  – client/server systems
  – Object-oriented development
  – Intranet technology
  – Web technology
  – Internet technology

• What is the best way to develop the right Java skills?
  – Building related skills
  – Leveraging a “green field” application
  – Developing iteratively

• In the business world, procedural-based systems are older and more pervasive than object-based systems
• Object-based systems are more difficult to build, but the promise of improved productivity, reusability, and maintainability is powerful
• Java can be used to support either a procedural or an object approach
Bridging Procedural and OO Styles (continued)

• For companies with a significant base of procedural developers, an approach worth considering is building the system in a way that leverages elements of both procedural and object styles.

Building Execution Architectures

• Companies building large or multiple business systems can benefit by leveraging an execution architecture that addresses common technology requirements.
• Execution architectures simplify and standardize the development of business applications for developers. These architectures can be used across multiple projects.

Building Execution Architectures (continued)

• The cost and time required to create an execution architecture must be justified by the estimated savings in development and operation costs.
• Java simplifies, enhances, and improves the implementation and operation of execution architectures.
### Building Execution Architectures (continued)

- The use of Java in building an execution architecture faces some challenges (for example, interfaces to third-party products), but the advantages generally outweigh the disadvantages.

### Considerations in Moving from Applets to multi-tier systems

- Consider the emerging nature of Java technology
- Consider availability and performance requirements of server-side services
- Consider reliability, security, and logical state mgmt challenges of deploying Internet-based apps
- Consider challenges of competing DOC standards (CORBA / COM+)

### Considerations in Building Complex Java-Based Systems

- Java supports end-to-end functionality (complex server-side service applications and client-side apps loaded from Web pages)
- System developer should provide a layer of isolation between the business system logic and unproven, rapidly evolving network computing technology
- DOC does not provide all the answers. “Gap filling” is needed in the infrastructure.
Overcoming Performance Challenges

- Java technology has been improving
- Computing platforms have been improving
- Portability is a tradeoff
- High-quality, predictability, and consistent behavior are other tradeoffs
- Typical improvements: efficient design / implementation, business processes streamlining, etc.

Making Applications Secure

- No need for dynamic loading of Java applications
- Only HTTP is allowed in some cases through corporate firewalls

Other Areas of Concern

- Highly Available Systems
- Motivating application developers
- Learning the technology
- Improving developer productivity
  - memory management, compiler, base classes, systems programming, platform independence, object programming, error handling
Other Areas of Concern
(continued)

• New development tools and processes
  – editors, compilers, debuggers, GUI painters,
  standards, source code control, hardware and
  OS, etc.