Infosys®

Win in the flat world

Legacy Modernization using Service Oriented Architecture

Rajib Das Sharma
Technology Consulting Group
Infosys Technologies Limited

IT Architecture Practitioners Conference
Mumbai

Feb 2007

Decades of Legacy Assets

- New Code cost 5X more than Reusing Existing Code (*Software Productivity Research*)
- 200 Billion Lines of COBOL Code in existence (*eWeek*)
- 5 Billion Lines of COBOL Code added yearly (*Bill Ullrich, TSG Inc.*)
- Between 850K and 1.3 Million COBOL Developers with 12,000 per year attrition (*IDC*)
- Majority of customer data still on Mainframes, even though a lot of it is front-ended by Web and eCommerce applications (*Don Greb, Mellon Financial Group in ComputerWorld*)

A Gartner Study of Legacy Systems

- The mainframe environment can be divided into three general segments:
  - < 500 MIPS
  - 500-1000 MIPS
  - > 1000 MIPS
- Enterprises less than 500 MIPS are most likely to successfully migrate which is again heavily dependent on application and environmental complexities
- Lack of innovation in this market (IBM is the only vendor providing J2EE-centric solutions on mainframe)
- Mainframe Enterprises Options:
  - Determine a long term application strategy
  - Decide on the role of mainframe in this strategy – maintain, transition or migrate
  - Consider SOA for preserving and extending mainframe applications
  - Evaluate investment in skills for external staff

Agenda

- What is Legacy
- What is Modernization
- Challenges of Legacy Modernization
- Evolution of Solutions
- Service Oriented Architecture – a solution for Legacy Modernization

What is a Legacy System?

- In an IT context, Legacy Systems are defined as production enabled systems with varying ‘ages’, those support mission critical business operations to generate revenue, to save operational cost and/or to perform accounting tasks, with varying implementation and deployment platforms.
- **Characteristics:**
  - Monolithic application built over years
  - Old/outdated computing platforms, languages, databases and UI
  - Decreasing vendor support
  - Complex program flow with no modularity and hard coded logic
  - Little to no trace ability to business flows
  - Inflexible and not scalable to support mergers and acquisitions

Challenges of Legacy Systems

- Lack of application knowledge among IT staff
- Business Knowledge is ‘tribal’ (in the head of the legacy developers and not documented anywhere)
- Apply changes to redundant code
- High cost of change
- Inability to integrate with new technologies
- Difficult to extend to support new business requirements
- Limited UI leading to low end user productivity

What is Modernization?

- Web enabling
- Enabling integration of disparate systems
- Software migration
- Hardware Migration
- Business Process Re-engineering
- Process change
- People skills change

Challenges of Legacy Modernization

- **Crystallization of Business Value** – Developing a convincing business case for the Modernization. Modernization as a technology change initiative is often won't go much far ahead.
- **Cost Management** - based in assumptions and extrapolations, high initial cost, frequent surprises, very hard to show reduced TCO in short term
- **Multiple Stakeholders** : Due to the vast system footprint inside an organization, there can be large number of stakeholders, need extremely efficient program management practices
- **Managing Expectations** – "Old is gold and new should be gold or platinum " – e.g. (the monolithic) CICS applications gives sub second response, the new distributed system should do the same.
- **Testing Effort** – difficult to justify with the stakeholders if there is a lack of sufficient enhancements in the functionality or business process.
- **Managing change in the Ecosystem** - Decades old practices and usage patterns are going to change. Need extremely good change management practices
- **Need to coexist and integrate with the surroundings** : One system or a few systems are changing at a time, need to coexist and run the business as usual

Modernization Options

- Legacy Wrapping
- Legacy Refactoring
- Legacy Integration
- Legacy Migration

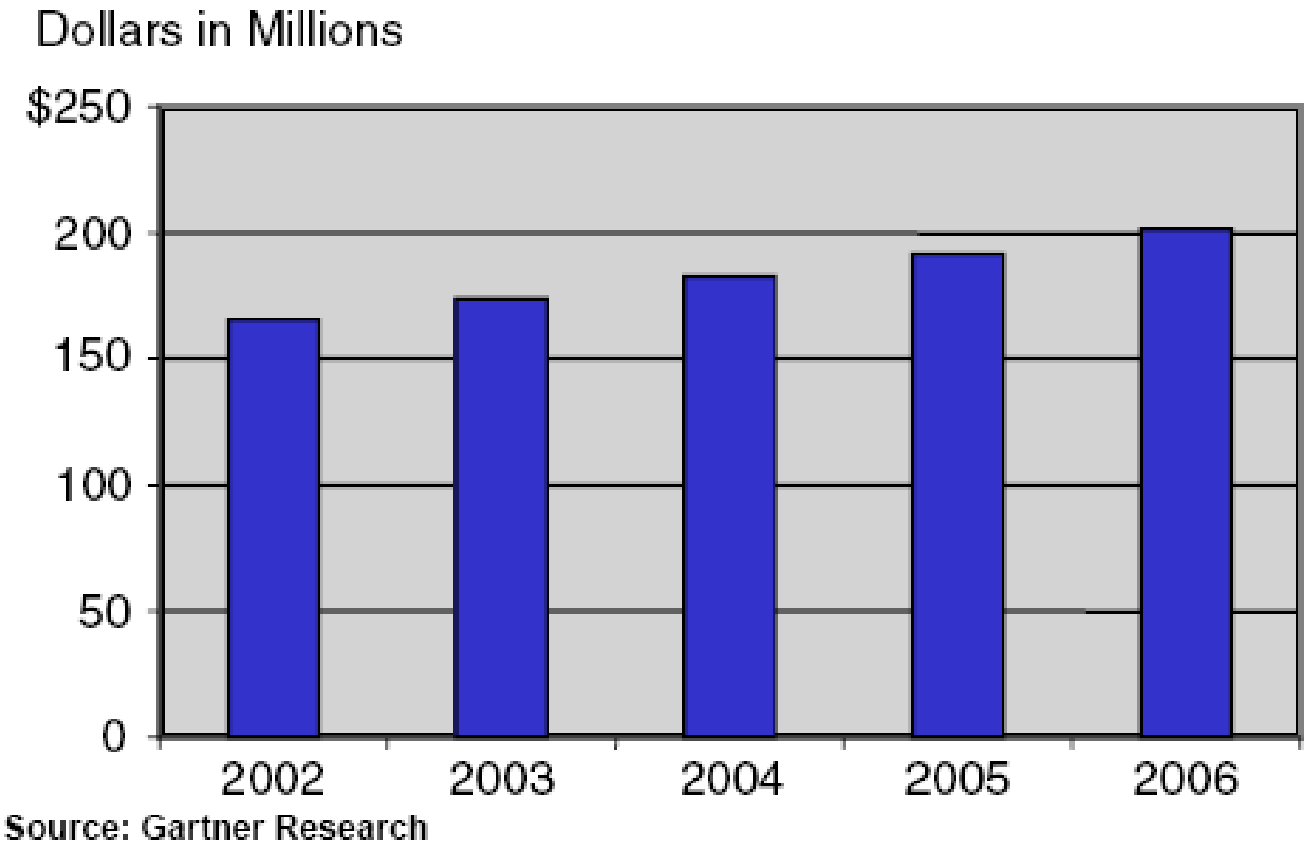
Evolution of Architecture, Tools & Technologies

- Enterprise Architecture
- Service Oriented Architecture
- Legacy Asset Extraction Tools
- Internet
- Web Services
- Enterprise Service Bus
- Grid Technology

Evolution of Solutions

- Screen Scrapping
- Connecting to mainframe components using proprietary protocols
- Wrappers/Facades over Legacy Components
- Service Oriented Architecture

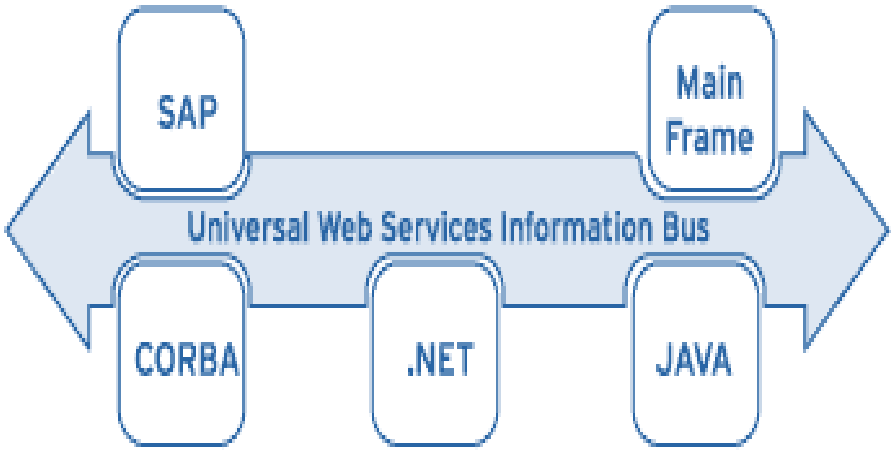
Growth of the Legacy Modernization Market as Tools and Technologies evolve



Source: Gartner Research

Service Oriented Architecture

- A service-oriented approach will standardize interaction
- Allows more flexibility in the process
- The complete value chain within a company is divided into small modular functional units or services.
- Companies and sub-units can provide services
- **Other business units can use these services to implement business processes**



Service Oriented Architecture Components

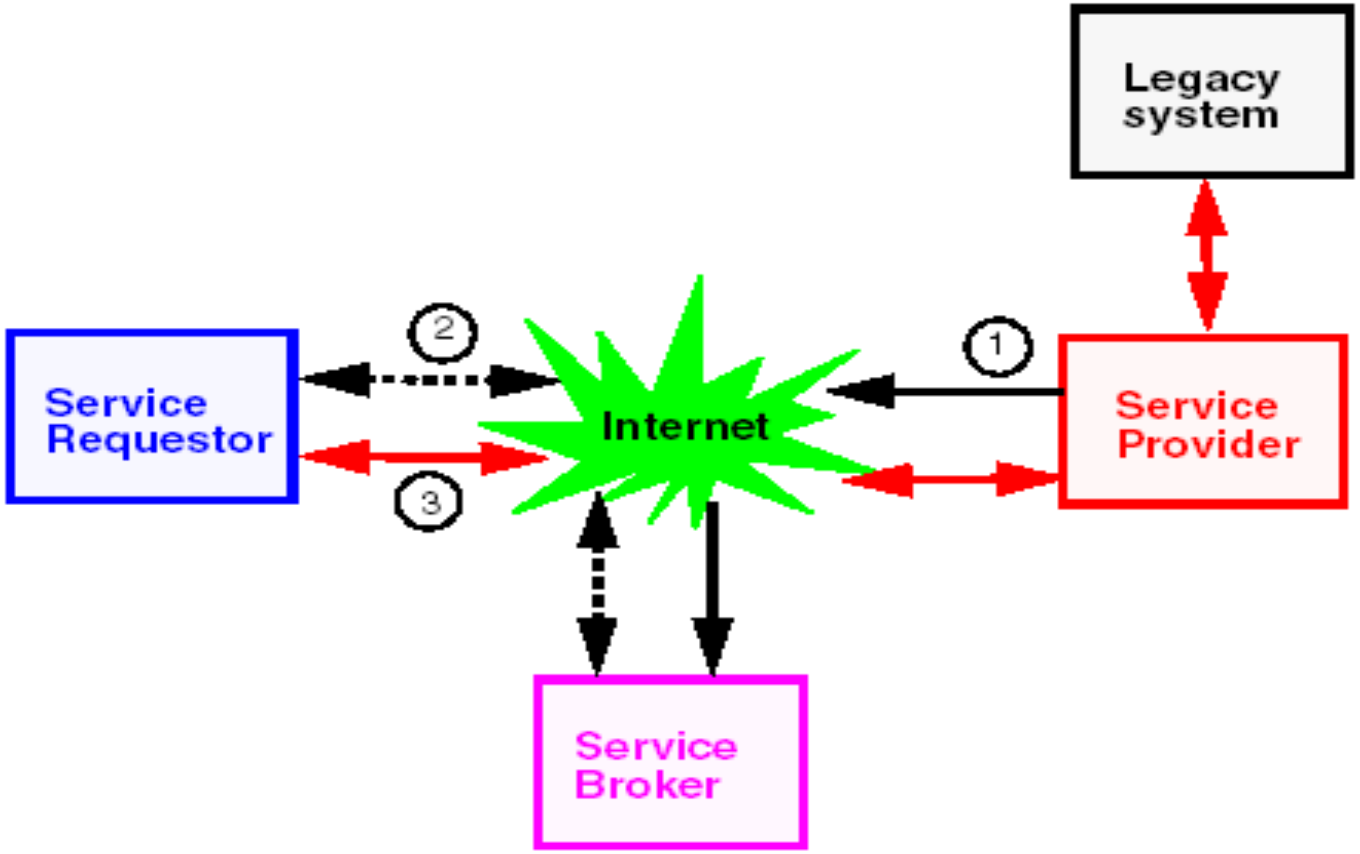
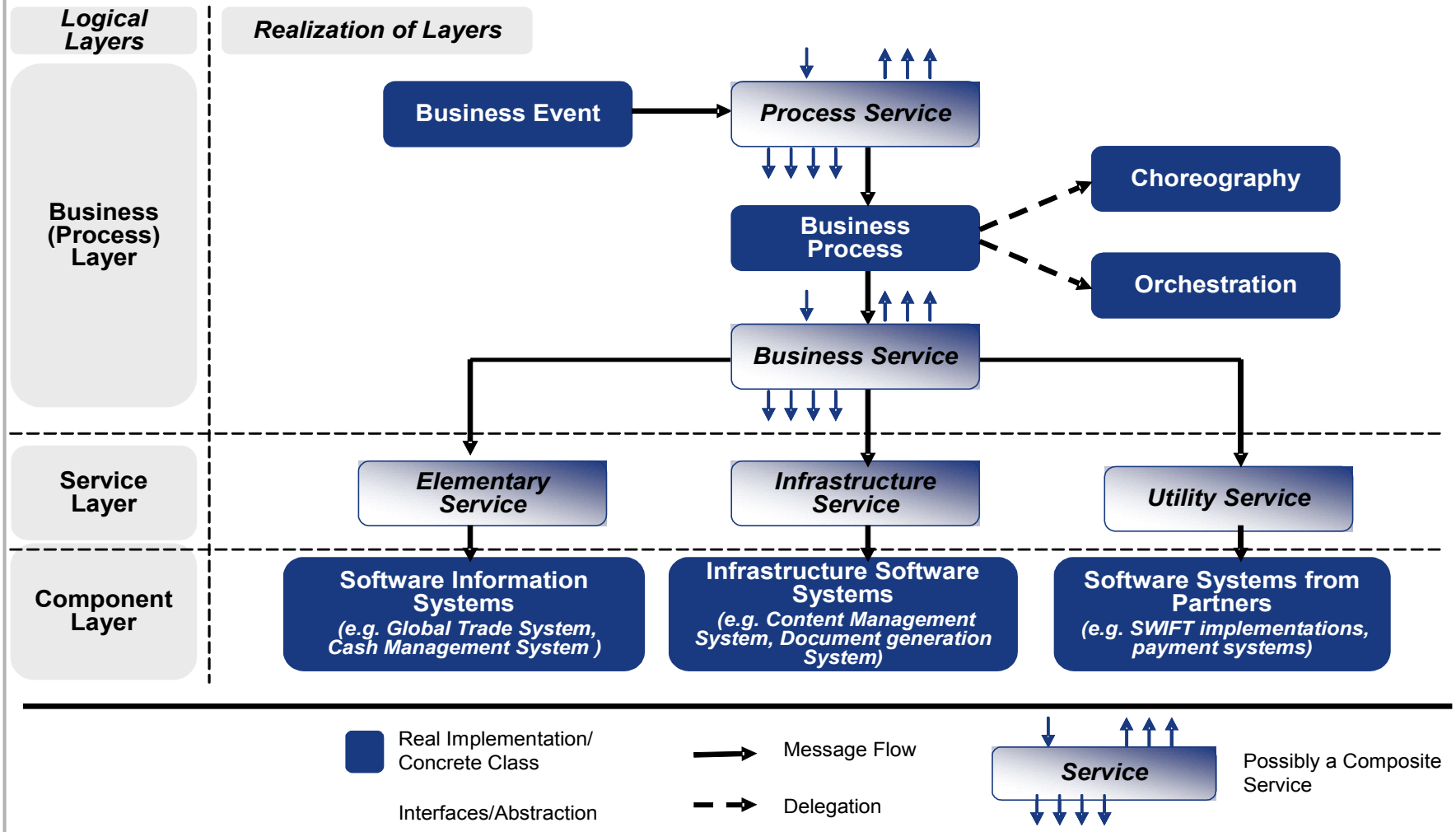


Figure 1-1 Web services roles and operations

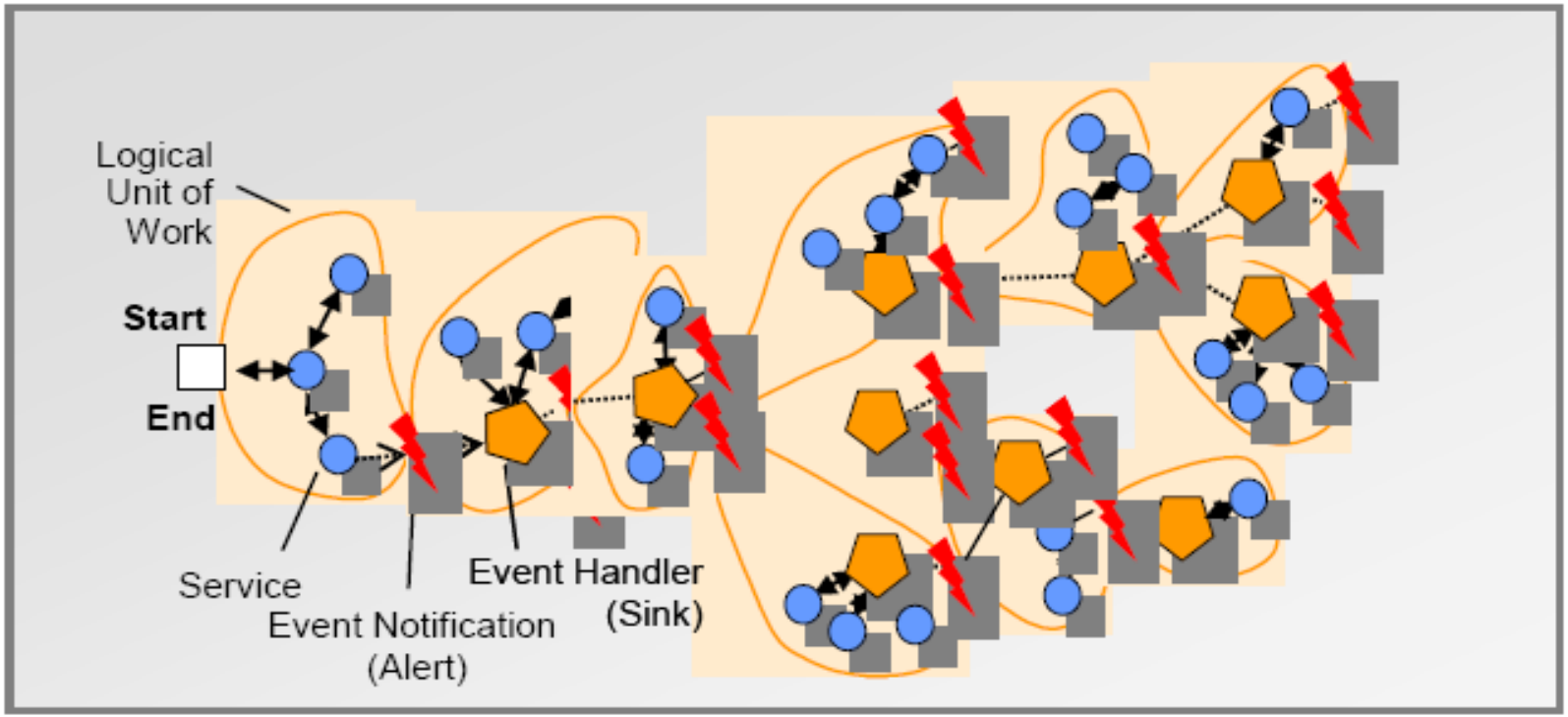
Advantages of Service Oriented Architecture

- Better Return On Investment
  - *The creation of a robust service layer has the benefit of a better return on the investment made in the creation of the software. Services map to distinct business domains.*
- Location Independence - *the client does not care where the service is located*
- Focused Developer Roles
- Support for Multiple Client Types
- Service Assembly
- Better Maintainability
- More Reuse
- Better Parallelism in Development
- Better Scalability

Service Design Framework



EDA and SOA – an advanced software topology



Source: Gartner (July 2006)

Thank You

Questions ?