Modernizing Your Enterprise with Open Source

University of Minnesota
MIS Research Center
Minneapolis, MN
November 9, 2007

Uwe Vieille
Open Source Strategic Program Office
An initial thought …
How true it is!

FROM NOW ON, I WANT YOU TO USE OPEN SOURCE SOFTWARE FOR EVERYTHING WE DO. IT'S FREE.

IT'S AN EMERGENCY. I THINK HE'S BEEN READING.

WE KNOW YOU HAVE A TRADE PUBLICATION IN HERE. HAND IT OVER AND NO ONE HAS TO GET HURT.

© Scott Adams, Inc./Dist. by UFS, Inc.
Agenda

- Architecture Driven Modernization
- Service Oriented Architecture
- Application Platform Modernization
- Infrastructure Modernization
- Open Source Open Standards
- Applying Open Source to Enterprise Modernization
- Case Studies
Architecture Driven Modernization
Introduction to Architecture Driven Enterprise Modernization

- Definitions
- Business Drivers
- Benefits and Risks
Architecture-Driven Modernization is the process of understanding & evolving existing assets for purposes of:

- application portfolio management
- code improvement
- programming language transformation
- integration
- platform migration
- data migration
- Consolidation
- Virtualization
- data warehousing
- reuse
- package selection
- service-oriented architecture
- model-driven architectures

The terms “ADM”, “Enterprise Modernization”, “Enterprise Asset Modernization”, “Legacy Modernization” and “IT Modernization” are used as synonyms.
Architecture Driven Modernization (ADM) is a discipline concerned with:

- evolving existing systems to meet future needs
- identification of the gap between existing system and enterprise’s current requirements
- preserving investments in existing systems (including proven business logic and expertise of current staff)
- understanding existing software and other IT assets
Why are businesses considering IT modernization?

• Agility
  – Existing systems do not support new business models
  – Modification of existing systems is too risky and expensive

• Intellectual Property
  – People with business and IT process knowledge are retiring
  – Functional knowledge hidden in application code

• Total Cost of Ownership
  – Operational cost of legacy systems is too high
  – Complexity drives maintenance and support cost of existing software
  – Skill sets are rare and too expensive
Driving Modernization Demand…

Figure 1 IT Spending On Maintenance And Ongoing Operations Is Growing Steadily

2-3 “Over what time frame do you plan to eliminate or reduce your COBOL applications?”

Don’t know 6%
Within the next 12 months 18%
Over the next 24 to 36 months 26%
Over the next 12 to 24 months 31%
Other 20%

Base: 347 decision-makers (companies with 100 IT employees or less filtered out from the initial base of 911) (percentages may not total 100 because of rounding)

Source: Forrester’s Business Technographics® November 2005 North American And European Enterprise Software AndServices Survey

Source: Forrester Research, Inc.
ADM benefits

- Increases **COMPETITIVE AGILITY** by supporting business vision and simplified business processes
- Ensures **UNINTERRUPTED OPERATIONS** by allowing old and new applications to coexist during an evolutionary transition period
- Provides **COST REDUCTION** for operation, support and maintenance by modernizing, consolidating and simplification of legacy applications and IT infrastructure
- Provides **RISK REDUCTION** associated with legacy systems including loss of legacy skills
- Discovers and preserves **INTELLECTUAL PROPERTY** hidden in existing applications and enables its **REUSE**
- Provides a **COMPREHENSIVE END-TO-END SOLUTION** by leveraging and integrating proven innovative technologies from Unisys and its partners
## Benefits, risks, and estimated costs

<table>
<thead>
<tr>
<th>Approach</th>
<th>Benefits</th>
<th>Risks</th>
<th>Estimated Cost</th>
</tr>
</thead>
<tbody>
<tr>
<td><strong>Traditional (Manual)</strong></td>
<td><strong>High quality</strong></td>
<td><strong>Expensive</strong></td>
<td><strong>$10-20 per line of code (Baseline cost)</strong></td>
</tr>
<tr>
<td></td>
<td></td>
<td><strong>Labor intensive</strong></td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td><strong>Business model – extra cost</strong></td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td><strong>Requires heavy support from Subject Matter Expert</strong></td>
<td></td>
</tr>
<tr>
<td><strong>Translation (“Short Horseshoe”)</strong></td>
<td><strong>Quick turnaround</strong></td>
<td><strong>May require extensive “legacy remediation” before the process</strong></td>
<td><strong>$3-6 per line of code (Roughly 1/3rd the baseline cost)</strong></td>
</tr>
<tr>
<td></td>
<td><strong>Least expensive</strong></td>
<td><strong>May require extensive “refactoring” after the process</strong></td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td><strong>May not work well for complex or poor quality application code</strong></td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td><strong>May have high maintenance cost</strong></td>
<td></td>
</tr>
<tr>
<td><strong>Knowledge-Based (“Tall Horseshoe”)</strong></td>
<td><strong>High Quality</strong></td>
<td><strong>More expensive than Translation (“Short Horseshoe”) method</strong></td>
<td><strong>$8-12 per line of code (Roughly 2/3rd the baseline cost)</strong></td>
</tr>
<tr>
<td></td>
<td><strong>Produces business model (vocabulary, rules, processes)</strong></td>
<td><strong>Requires validation support from Subject Matter Experts</strong></td>
<td></td>
</tr>
<tr>
<td></td>
<td><strong>Generates automated test cases</strong></td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td><strong>Generates usable documentation, including training documents</strong></td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td><strong>Deploy target applications from business model</strong></td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td><strong>Low maintenance cost</strong></td>
<td></td>
<td></td>
</tr>
</tbody>
</table>
What the Analysts Say…

- “By 2010 80% of smaller mainframe environments will migrate off the platform, while large environments will SOA-enable and remain on this platform”
  – Gartner

- “Migrate away from Legacy Databases”
  “…IDMS and Datacom in next 5 years”
  “…IMS and ADABAS in next 10 years”
  – Gartner

- “Open Source has emerged as one of the most important IT trends in this young century”
  - Forrester

What are you doing with your Cobol applications?

- Keep all or most Cobol applications: 31%
- Eliminate some Cobol applications: 27%
- Eliminate all or most Cobol applications: 26%
- Don’t know: 5%
- Move all Cobol applications to Windows/Intel or Unix: 11%

ComputerWorld
April 2006
Service Oriented Architecture
SOA **MAY** Be Defined As:

“The policies, practices, frameworks that enable business functionality to be provided and consumed as sets of services published at a granularity relevant to the service consumer. Services can be invoked, published and discovered, and are abstracted away from the implementation using a single, standards-based form of interface.”

**Service**
- Capabilities performed by one for another to achieve a desired outcome

**Oriented**
- Aligning architecture to enable a collection of services to be linked together to solve a business problem

**Architecture**
- The fundamental organization of a system by its capabilities, their interactions, and the enterprise environment

SOA is an approach for creating, organizing and consuming services to support interaction between people, processes, and information assets
Business Architecture Focus: Metadata

- Security policy
- Business Process models
- Business Metadata Repository
- SOA Framework
- Regulation, compliance, privacy, security
- Organizational structure
Business services drive the agile Enterprise

Business Metadata

Business SOA

Business services

Business Process Services

Service delivery network

Flight Schedule inquiry
Seat Availability
Booking
Ticketing
Payment
Check-in

Services initiate Business processes

Composite services and information

Portal apps
Data grids
ERP
CRM
Legacy Apps
Rules engines
The SOA IT Infrastructure Model
# The Mental Model Shift

<table>
<thead>
<tr>
<th>Overall</th>
<th>From: First design your business, then design systems to support it</th>
<th>To: Concurrently design your business and the systems that embody it</th>
</tr>
</thead>
<tbody>
<tr>
<td>Business applications</td>
<td>From: Application code to serve specific business functions</td>
<td>To: Create re-usable business service capabilities orchestrated for agile business functionality</td>
</tr>
<tr>
<td>User interfaces</td>
<td>From: Creating application specific user interfaces</td>
<td>To: Optimized endpoints to connect business processes to users &amp; devices</td>
</tr>
<tr>
<td>Communications &amp; collaboration</td>
<td>From: Single, fits all network on which to deliver a message</td>
<td>To: Multi function collaborative, cross-channel network</td>
</tr>
<tr>
<td>IT infrastructure</td>
<td>From: Hosting applications on specific servers</td>
<td>To: Allocate real resources in a virtual environment to orchestrated business processes</td>
</tr>
</tbody>
</table>

---

**UNiSYS**
What do you need to do to implement the model?

<table>
<thead>
<tr>
<th>STOP</th>
<th>START</th>
</tr>
</thead>
<tbody>
<tr>
<td>STOP writing isolated IT requirements documents.</td>
<td>START joint business-IT solution development of business problems.</td>
</tr>
<tr>
<td>STOP building single use applications for specific business functions.</td>
<td>START building process-ready business service capabilities.</td>
</tr>
<tr>
<td>STOP designing user interface screens.</td>
<td>START creating re-usable business process service endpoints.</td>
</tr>
<tr>
<td>STOP using architecture merely for cost-saving standardization.</td>
<td>START using architecture as the basis for strategic business agility.</td>
</tr>
<tr>
<td>STOP treating business and IT as two different worlds.</td>
<td>START cross-boundary process governance, policies and rules.</td>
</tr>
</tbody>
</table>
So, SOA is the answer, right?

- Adoption of SOA is strong. But even more notably, SOA users are quite happy with it and expect it to make a big impact on their enterprises.

- Among the largest enterprises — those with 40,000 or more employees - 67% will be using SOA by the end of 2006. Among the smallest organizations - SMBs with fewer than 1,000 employees — 44% report that implementing an SOA is a high or critical priority.

- Nearly 70% of enterprise SOA users say they will increase their use of SOA,

- Nearly half of all large enterprises using SOA use it for strategic business transformation. What does all this add up to? SOA is a critical foundation for IT’s future ability to meet the strategic needs of the business, and firms are asking how to invest in it.

Forrester, May 19, 2006, SOA Investment Strategies
Case Studies On How Enterprises Are Paying For SOA
Level of SOA Penetration

“The majority are using SOA

They are quite happy and doing more SOA

They look for it to have a big impact

Source: Forrester’s Business Technographics® November 2005 North American And European Enterprise Software And Services Survey
What is so different about this?

Think you have seen this before? Didn’t CORBA, DCOM, or Java RPC promise this also?

There are some important updates to the story this time: Changes in the IT industry and business domains show that SOA has made a paradigm shift that cannot afford to be missed:

- The groundwork has been laid with nearly universal adoption of protocols such as TCP and HTTP
- A stable infrastructure agnostic information exchange foundation based on XML
- All the big infrastructure players are deeply involved creating a robust open standards based interoperability platform
- An open standards based development environment that leverages re-use through orchestration
Missteps That Can Result in SOA Strategy Failure

Stage 1 — Initial Excitement
- **Recommendation:** Set reasonable expectations about Web services development that recognize Web services as mechanisms that operate within an SOA, not SOA’s in and of themselves.

Stage 2 — Hype “Buy-In”
- **Recommendations:** Focus on debunking hype rather than believing it. Evaluate products based on requirements as well as features.

Stage 3 — Underestimating Complexity
- **Recommendation:** Consider Web services frameworks as an initial investment need for a Web services based SOA.

Stage 4 — Overestimating Maturity
- **Recommendations:** Conceptually separate the use of opportunistic Web services from the implementation of an SOA. Use opportunistic Web services for tactical success while designing a complete SOA strategy.
Stage 5 — Disenchantment and Abandonment

- **Recommendations:** Spend time evaluating SOA platforms and frameworks at the outset. Generate a corporate SOA strategy before buying into a vendor’s SOA strategy.

Stage 6 — The “Tool-of-the-Month Club”

- **Recommendation:** Focus on standards and strategic technologies rather than tactical features when crafting an SOA vision. This will help in following the evolution of the strategy toward long term usability.

**Bottom Line**

- Through 2008, 70 percent of IT organizations will fail to successfully select and implement an SOA strategy on the first try (0.7 probability).

- These organizations must be prepared to use software services tactically while planning for strategic evolution of their architectures.

Analytical source: Daryl C. Plummer, Gartner Research
Application Platform Modernization
Two Sides of Legacy Systems

It’s an asset!
- Functional
- Customized
- Well-tuned
- Supports mission critical operations
- Robust
- Secure

It’s a liability!
- Uses legacy technology
- Requires hard-to-find skills
- Lacks documentation
- Is difficult to maintain
- Resists change, integration, and replacement
- Inhibits organizational agility
- Is Unsupported

ADM preserves the asset value while eliminating the liabilities.
...but what to do...

Where to begin...

No company should expect to implement a single modernization strategy across the entire portfolio. For many applications, the modernization approach is to eliminate them. For others, replacement with packaged solutions may be more appropriate. In some cases, SOA can provide an approach to leveraging the inherent business value in these applications, while opening up access to the business functionalities in these systems to new constituents.²

Think strategic, act tactical

Organizations can get experience exposing existing applications as services for a particular project. However, focusing solely on the success of the immediate project does not position the organization for SOA as a long-term modernization strategy¹.

• Define the role of the current IT infrastructure in future SOA architecture
• Build a reuse culture and technology infrastructure
• Use code understanding / inventory / restructuring tools to improve service granularity³

¹Gartner, Dale Vecchio, Best Practices for Mainframe SOA, January 23, 2006
²Gartner, Dale Vecchio, Leverage Your Mainframe Applications With SOA, October 25, 2005
...and where to start...

1. \( \text{...begins with mining of knowledge from existing infrastructure...} \)
   - Understanding legacy in all its complexity is a key to secure legacy transformation, Web enablement and SOA re-architecture. Enterprises should implement legacy analysis. We recommend automating that process\(^1\)

2. \( \text{...and heads towards Service Oriented Architecture (SOA)} \)
   - SOA will be used in part in more than 50% of new, mission-critical applications, business processes and infrastructures designed in 2007, and in more than 80% by 2010\(^2\)

More than 80% of current (2006) production-deployed applications will be partly or fully re-engineered by 2011 to participate in the then-prevailing SOA-style business-driven application composition\(^3\)

---

\(^1\)Gartner, Joseph Feiman, *Implementing Security for Mainframe Legacy Applications Worth the Investment*, September 29, 2005


\(^3\)Ibid
Blueprinting Enterprise Modernization
Applies Concepts of Traceability and Visibility to Challenge of Modernization

As-Is Business Model

New business needs

To-Be Business Model

Business Domain

IT Domain

As-Is Applications

To-be applications

Applying concepts of business modeling to modernization
1. Discovery Reports Against Code

New Business Requirements

As-Is Business Rules Model

To-Be Business Rules Model

ADM

Business Domain

IT Domain

1. Dynamic Analysis

Discovery of Boundary Elements

Documentation

SMEs/interviews

COBOL

Rule Engine

J2EE or .Net Apps

UNISYS
2. Discover BR Implementations

As-Is Business Rules Model

To-Be Business Rules Model

Seed Business Vocabulary

New Business Requirements

2. Unisys Rules Modeler

Business Domain

Discover BR Implementations

3. Relativity

IT Domain

Documentation
SMEs/interviews
COBOL

Rule Engine
J2EE or .Net Apps

UNISYS
3. Map BR Implementations to BR Model

3.1 As-Is Business Rules Model

3.2 To-Be Business Rules Model

3.3 ADM

3.4 Unisys Knowledge Modeler

New Business Requirements

3. Map BR Implementations to BR Model

New Business Requirements

As-Is Business Rules Model

To-Be Business Rules Model

Map Relativity BR Implementations to KM BRM

Discover BR Implementations

Relativity

4. Unisys Knowledge Modeler

Documentation

SMEs/interviews

COBOL

J2EE or .Net Apps

Rule Engine

Business Domain

IT Domain

UNISYS
IT Infrastructure Modernization
# Benefits, risks, and estimated costs

<table>
<thead>
<tr>
<th>Approach</th>
<th>Benefits</th>
<th>Risks</th>
<th>Estimated Cost</th>
</tr>
</thead>
<tbody>
<tr>
<td>Traditional (Add new Technology platform)</td>
<td>➢ No interruption of current environment</td>
<td>➢ Expensive&lt;br&gt;➢ Labor intensive&lt;br&gt;➢ Business model – extra cost&lt;br&gt;➢ Requires additional maintenance and support resources</td>
<td>$$$$$</td>
</tr>
<tr>
<td>Consolidate and Migrate (Integrate and Replace Legacy Systems)</td>
<td>➢ Relatively Quick ROI&lt;br&gt;➢ Less complex environment</td>
<td>➢ Requires extensive planning&lt;br&gt;➢ May be disruptive&lt;br&gt;➢ May not work well for complex or diverse SLA application / DB systems</td>
<td>$$$</td>
</tr>
<tr>
<td>Virtualize and Replace Legacy Infrastructure</td>
<td>➢ Lowest maintenance and support cost&lt;br&gt;➢ Highest Level of IT and Business Agility&lt;br&gt;➢ Resources can be re-positioned to create business value functionality</td>
<td>➢ More expensive than Consolidation&lt;br&gt;➢ Will be disruptive&lt;br&gt;➢ Requires different Skill Sets</td>
<td>$$$$$</td>
</tr>
</tbody>
</table>
...and where to start...

1. ...begins with mining of knowledge from existing infrastructure...

2. ...and heads towards Service Oriented Architecture (SOA) And Virtualization

Understanding legacy in all its complexity is a key to secure legacy transformation, Web enablement and SOA re-architecture. Enterprises should implement legacy analysis. We recommend automating that process.

SOA and Virtualization will be used in part in more than 50% of new, mission-critical applications, business processes and infrastructures designed in 2007, and in more than 80% by 2010.

1 Gartner, Joseph Feiman, Implementing Security for Mainframe Legacy Applications Worth the Investment, September 29, 2005
3 Ibid
Open Source and Open Standards
Open Source and Open Standards Adoption Lifecycle: Where is the Market Today?

- Linux and JBoss have “Crossed the Chasm”
- Other OSS is in the Innovator Phase of Market Adoption – Rapidly Gaining Acceptance
- Linux Economics and Dependability are driving Data Center Consolidation
- Web Services and SOA are Driving an Appl-structure Transformation
Applying Open Source to Enterprise Modernization
### The Open Source Stack Components

<table>
<thead>
<tr>
<th>Components</th>
<th>Proprietary</th>
<th>Pure OS</th>
<th>Hybrid Stack</th>
</tr>
</thead>
<tbody>
<tr>
<td>Browser</td>
<td>Internet Explorer</td>
<td>Firefox</td>
<td>Firefox or IE</td>
</tr>
<tr>
<td>Web Server</td>
<td>MSFT Commerce Server or Oracle</td>
<td>Apache, PHP</td>
<td>Apache, PHP</td>
</tr>
<tr>
<td>Middleware</td>
<td>BEA WebLogic</td>
<td>JBoss</td>
<td>BEA IBM</td>
</tr>
<tr>
<td>Database</td>
<td>IBM DB2, Oracle</td>
<td>MySQL, PostgreSQL</td>
<td>MySQL, PostgreSQL</td>
</tr>
<tr>
<td>Operating System</td>
<td>Solaris, HP-UX, AIX</td>
<td>Red Hat Novell SUSE</td>
<td>Red Hat, SUSE</td>
</tr>
<tr>
<td>Hardware</td>
<td>Sun, HP, IBM</td>
<td>unisys</td>
<td>unisys</td>
</tr>
</tbody>
</table>

*Note: The diagram illustrates the components and their corresponding stack options. The proprietary and open source stacks are clearly distinguished.*
Legacy System Integration

Integration Trough Modernization

- Portal such as LifeRay
- Development Platform such as Eclipse
- App Server such as Apache
- Database such as MySQL and Postgres
- Middleware such as JBoss

Integration Trough ESB

- Reports such as JasperSoft
- Service Bus such as IONA ServiceMix
The ROI of Open Source

• Yankee Group, JupiterResearch, Forrester Research and others, have focused on the ROI of upgrading a Windows installation versus switching to Linux and have concluded that it is less expensive to stick with Windows. But the reports miss a critical point: Switching from Windows to Linux is the worst-case ROI scenario.

A more important question is, can open source generate real ROI elsewhere?

• ABB is an $18 billion Swiss industrial company. It needed to integrate new features into its software infrastructure. By using a popular open-source tool called JBoss, ABB estimates it can save $1.1 million in just its first five factories, with further savings to come as it rolls out to more of PTPR’s 52 locations.

• Interestingly, the Integration Framework runs on Windows and uses SQL Server as its data store, belying the perception that moving to open source is a massive rip-and-replace operation.
Enterprise Modernization using Open Source

Case Studies
Case Study: Leading Airline Alliance
The Business Challenge

Upgrade a seat on any member airline on any available flight from anywhere in the world using the Internet, Mobile Phone, fixed line phone or the airport check-in counter.
Case Study: Leading Airline Alliance

- Customer experience is perceived single airline – upgrade, payment and check-in

- Real life environment is
  - operating airline has physical seat inventory,
  - marketing codeshare airline has reservation and revenue,
  - loyalty program airline has FF miles as form of payment for upgrade, and
  - third party airport operator manages airport check-in
The IT Challenge

• Each airline has unique IT Infrastructure
  ➢ IBM and Unisys legacy infrastructures
  ➢ Web interfaces
  ➢ Proprietary links between airlines and service providers
  ➢ Proprietary processes
  ➢ Proprietary and legacy external interfaces

• Unique airline environment needs to be preserved
  ➢ Easy entry, easy exit
  ➢ All process and transaction normalization will occur outside airline domains
Case Study: Leading Airline Alliance

Solution

- Created common business metadata repository
- Created common business process services – inventory availability, eligibility, accounts payable, accounts receivable, etc.
- Created common application services from existing legacy applications and new web server applications
- Created Web Services infrastructure
- Created common messaging Interface
- Implemented Open Standards and Open Source

*Leading Market Research Company estimated US$60 M+ in additional Revenue in Year 1 of full Deployment*
Business Architecture Focus: Metadata

- Security policy
- Business Process models
- Business Metadata Repository
- SOA Framework
- Regulation, compliance, privacy, security

Organizational structure
Business services drive the agile Enterprise

Service delivery network
- Flight Schedule inquiry
- Seat Availability
- Booking
- Ticketing
- Payment
- Check-in

Business SOA
- Business services
- Business Process Services

Composite services and information
- Portal apps
- Data grids
- ERP
- CRM
- Legacy Apps
- Rules engines

UNiSYS
Application

Air Travel Shopping Engine fare search used by Travelocity and others moved from mainframe to Linux, MySQL with over 4 TB of data.

Key Business Benefit

More than $10 million in savings while scaling to millions of transactions daily.

Why MySQL? Speed and scalability

“MySQL ran faster or as fast as any commercial database we tested. It never crashed. It was the fastest to get working.”

Alan Walker
Systems Architect
Sabre Holdings
Sabre Holdings and MySQL

IBM Naming Service and Load Balancing

Schedule and Availability Updates
Fare and Rule Updates

Shopping Transactions
Air Shopping Transactions

MVS
Linux

Linux Server Farm

Fare and Rule Updates

DB Image Load & Updates E/R

Logging and Billing

Shopping Transactions

Availability Requests

Load Information

Naming Service And Load Balancing

Load & Updates

59 4-way Servers for CPU & memory-intensive search
45 4 way Servers for MySQL Database
Orbitz’s Commitment to Open Source Software

• Where? In every environment
• What? Software at (almost) every layer of the stack.
• Who? All of
  – Orbitz.com
  – Cheaptickets.com
  – Travelport.com
  – Lodging.com
  – Orbitzforbusiness.com
  – Parts of American Airlines’ website
  – Parts of United Airlines’ website
  – Parts of Northwest Airlines’ website
• Open source software has played an important role in the ability of Orbitz to compete successfully with entrenched companies that are much larger and much better funded.
Where is OSS used at Orbitz?

- Open source software is pervasive throughout the architecture
- Linux and Java are used everywhere
- OSS libraries and tools in use have many origins
- The catalog of open source software used in Orbitz systems is constantly growing

Source: Orbitz
Airline client Unix Server Consolidation and Virtualization

Currently 817 HP and Sun Unix Servers Deployed

- Utilizing toolsets to analyze and profile applications, it could be reasonably expected that client airline could reduce the Unix Server footprint from 817 systems down to 426 Unix and 56 Linux servers.

- This would drive savings of $13.247M over three years
Open Source Virtualization: How You Could Do It

Examine and profile 817 Unix servers

- Eliminate 426 servers as unsuitable consolidation candidates for various reasons (SLA’s, Applications etc.)

Collapse the remaining 391 servers on a 7:1 ratio to 56 servers

- Reduced total footprint is now 482 servers
- Every 50 servers costs $1.5M over 3 years to operate

Total savings over three years of $13.247M
Currently 222 BEA Tuxedo/Weblogic maintenance and support licenses deployed

- Replace 222 BEA Tuxedo/Weblogic maintenance and support licenses with JBoss TS/AS for a cost reduction of $1.519M over 3 years
Questions?
Thank You.