How Architecture-Driven Modernization Is Changing the Game in Information System Modernization

Philip Newcomb, CEO & Founder
philip.newcomb@softwarerevolution.com

The Software Revolution, Inc. (TSRI)
Kirkland, Washington
www.softwarerevolution.com

Copyright 2010 The Software Revolution, Inc All Rights Reserved
**Report Documentation Page**

Public reporting burden for the collection of information is estimated to average 1 hour per response, including the time for reviewing instructions, searching existing data sources, gathering and maintaining the data needed, and completing and reviewing the collection of information. Send comments regarding this burden estimate or any other aspect of this collection of information, including suggestions for reducing this burden, to Washington Headquarters Services, Directorate for Information Operations and Reports, 1215 Jefferson Davis Highway, Suite 1204, Arlington VA 22202-4302. Respondents should be aware that notwithstanding any other provision of law, no person shall be subject to a penalty for failing to comply with a collection of information if it does not display a currently valid OMB control number.

| 1. REPORT DATE | APR 2010 |
| 2. REPORT TYPE | |
| 3. DATES COVERED | 00-00-2010 to 00-00-2010 |
| 4. TITLE AND SUBTITLE | How Architecture-Driven Modernization Is Changing the Game in Information System Modernization |
| 5a. CONTRACT NUMBER | |
| 5b. GRANT NUMBER | |
| 5c. PROGRAM ELEMENT NUMBER | |
| 5d. PROJECT NUMBER | |
| 5e. TASK NUMBER | |
| 5f. WORK UNIT NUMBER | |
| 6. AUTHOR(S) | The Software Revolution, Inc. (TSRI), 11410 NE 122nd Way Suite 105, Kirtland, WA, 98034 |
| 7. PERFORMING ORGANIZATION NAME(S) AND ADDRESS(ES) | |
| 8. PERFORMING ORGANIZATION REPORT NUMBER | |
| 9. SPONSORING/MONITORING AGENCY NAME(S) AND ADDRESS(ES) | |
| 10. SPONSOR/MONITOR’S ACRONYM(S) | |
| 11. SPONSOR/MONITOR’S REPORT NUMBER(S) | |
| 12. DISTRIBUTION/AVAILABILITY STATEMENT | Approved for public release; distribution unlimited |
| 13. SUPPLEMENTARY NOTES | Presented at the 22nd Systems and Software Technology Conference (SSTC), 26-29 April 2010, Salt Lake City, UT. Sponsored in part by the USAF. U.S. Government or Federal Rights License |
| 14. ABSTRACT | |
| 15. SUBJECT TERMS | |
| 16. SECURITY CLASSIFICATION OF: | |
| a. REPORT | unclassified |
| b. ABSTRACT | unclassified |
| c. THIS PAGE | unclassified |
| 17. LIMITATION OF ABSTRACT | Same as Report (SAR) |
| 18. NUMBER OF PAGES | 36 |
| 19a. NAME OF RESPONSIBLE PERSON | |

Standard Form 298 (Rev. 8-98)  
Prescribed by ANSI Std Z39-18
Why Modernize Legacy Systems?

Doing Nothing Is The Pathway To Decline

- Half Trillion Lines of Code Written in Obsolete Legacy Languages
- Massive Drain on National Productivity
- Many times more Costly to Maintain and Operate Than Modern Applications
- Hindering Progress And Innovation In Every IT Sector

Choosing The Wrong Pathway Is Perilous

- Massive Waste On Misguided and Poorly Executed Approaches.
- Past Replacement Efforts Have A Shocking 63% Failure Rate*
- Historic Modernization Failures Cited Time and Again
- Past Failure Should Never Be A Justification For Doing Nothing

*according to Forster
Why Architecture Driven Modernization?

- ADM Is A Body of Management Practices That Work
- Backed By the Object Management Group (OMG)
- The Only Formally Defined Set Of Modernization Standards
- Establishes Routine Predictable Repeatable Practices
- If Followed, ADM Delivers:
  - High-Quality Modernized Systems
  - Fraction The Time and Cost Of Past Methods,
  - Handles All Combinations of Source And Target Languages,
  - Applicable To All Sectors: Military, Government, Commercial
- ADM Is The Best Path To Moving Aging Systems To Agile Methods

ADM Works!!
ADM Works Across All Legacy Languages, Platforms and Applications
ADM Works at a Fraction of the Cost and Time

<table>
<thead>
<tr>
<th>Integrator</th>
<th>System</th>
<th>Code</th>
<th>SLOC</th>
<th>TTC</th>
</tr>
</thead>
<tbody>
<tr>
<td>Navigant</td>
<td>Health Car Service Corp.</td>
<td>PowerBuilder &amp; Magna X to Java</td>
<td>170K</td>
<td>On-Going</td>
</tr>
<tr>
<td>Premera Blue Cross</td>
<td>Automated Doc. Assembly System</td>
<td>COBOL to C++</td>
<td>44K</td>
<td>3 mo.</td>
</tr>
<tr>
<td>AETNA</td>
<td>Proof-of-Concept</td>
<td>COBOL DOCS &amp; Re-Fact.</td>
<td>92K</td>
<td>3 mo.</td>
</tr>
<tr>
<td>Thales Air Systems</td>
<td>French Air Traffic Management</td>
<td>Ada to Java</td>
<td>495K</td>
<td>12 mo.</td>
</tr>
<tr>
<td>Thales Air Systems</td>
<td>Nordic Air Traffic Management</td>
<td>Ada to Java</td>
<td>541K</td>
<td>9 mo</td>
</tr>
<tr>
<td>Thales Air Systems</td>
<td>Australian Air Traffic Management</td>
<td>Ada to Java</td>
<td>638K</td>
<td>9 mo</td>
</tr>
<tr>
<td>Unisys</td>
<td>NY State Dept. of Criminal Justice</td>
<td>COBOL Documentation</td>
<td>308K</td>
<td>2 mo.</td>
</tr>
<tr>
<td>NEA</td>
<td>Grant &amp; Business Systems</td>
<td>COBOL to C++</td>
<td>656K</td>
<td>7 mo.</td>
</tr>
<tr>
<td>SAIC</td>
<td>Veteran's Health Administration</td>
<td>MUMPS to Java</td>
<td>300K</td>
<td>4 mo.</td>
</tr>
<tr>
<td>State of OR</td>
<td>Employee Retirement System</td>
<td>COBOL to C# .Net</td>
<td>250K</td>
<td>4 mo.</td>
</tr>
<tr>
<td>State of WA</td>
<td>Off. of Super of Public Instruct.</td>
<td>COBOL to C# .Net</td>
<td>191K</td>
<td>5 mo.</td>
</tr>
<tr>
<td>TriGeo</td>
<td>Sim v4.0 (Internal Product)</td>
<td>Java Docs &amp; Re-Fact.</td>
<td>370K</td>
<td>2 mo.</td>
</tr>
<tr>
<td>EDS</td>
<td>Proof-of-Concept</td>
<td>P/L 1 to Java</td>
<td>50K</td>
<td>7 mo.</td>
</tr>
<tr>
<td>CSC</td>
<td>Bureau of Immigration</td>
<td>COBOL to C++</td>
<td>17K</td>
<td>3 mo.</td>
</tr>
<tr>
<td>Boeing</td>
<td>ALCA - Czechoslovakia</td>
<td>Jovial to C++</td>
<td>9K</td>
<td>2 mo.</td>
</tr>
<tr>
<td>LMCO</td>
<td>P-3C</td>
<td>Ada - C++</td>
<td>656K</td>
<td>14 mo.</td>
</tr>
<tr>
<td>ITT</td>
<td>BMEWS - Cobra Dane</td>
<td>Ada/Fortran - C++</td>
<td>380K</td>
<td>8 mo.</td>
</tr>
<tr>
<td>Raytheon</td>
<td>Satellites</td>
<td>Ada/Fortran - C++</td>
<td>284K</td>
<td>5 mo.</td>
</tr>
<tr>
<td>L-3</td>
<td>VTT</td>
<td>Ada - C++</td>
<td>77K</td>
<td>3 mo.</td>
</tr>
<tr>
<td>LMCO</td>
<td>SAC Strategic Planning System</td>
<td>Ada - C++</td>
<td>40K</td>
<td>2 mo.</td>
</tr>
<tr>
<td>DSR</td>
<td>E-2C ACFT</td>
<td>Ada - C++</td>
<td>20K</td>
<td>5 mo.</td>
</tr>
<tr>
<td>USAF</td>
<td>CAMS</td>
<td>COBOL Docs</td>
<td>1M</td>
<td>6 mo.</td>
</tr>
<tr>
<td>NGC</td>
<td>REMIS</td>
<td>COBOL - C++</td>
<td>400K</td>
<td>7 mo.</td>
</tr>
<tr>
<td>Dyncorp</td>
<td>WCRS</td>
<td>COBOL - C++</td>
<td>90K</td>
<td>5 mo.</td>
</tr>
<tr>
<td>ITT</td>
<td>BMEWS - ROSA</td>
<td>Fortran/C Docs</td>
<td>2M</td>
<td>4 mo.</td>
</tr>
<tr>
<td>Raytheon</td>
<td>Patriot Missile</td>
<td>Fortran - C++</td>
<td>200K</td>
<td>6 mo.</td>
</tr>
<tr>
<td>Raytheon</td>
<td>Strategic Air Command</td>
<td>Fortran - C++</td>
<td>50K</td>
<td>4 mo.</td>
</tr>
<tr>
<td>Raytheon</td>
<td>WAC</td>
<td>Fortran - C++</td>
<td>40K</td>
<td>1 mo.</td>
</tr>
<tr>
<td>SAIC</td>
<td>EOS</td>
<td>VAX Basic - Java</td>
<td>38K</td>
<td>5 mo.</td>
</tr>
<tr>
<td>TRW</td>
<td>MILSTAR</td>
<td>Jovial to C++</td>
<td>143K</td>
<td>1 mo.</td>
</tr>
<tr>
<td>USAF</td>
<td>F-16 Decis</td>
<td>Jovial to C++</td>
<td>50K</td>
<td>4 mo.</td>
</tr>
</tbody>
</table>
ADM Works with Prevailing Engineering Processes

Legacy Systems “As Is”

Requirements Analysis, System Architecture, Systems Engineering, Management Plan

Systems Engineering Process
ISO 9001, CMM-1 Level 3

Target System “To Be”

the software revolution, inc.

ADA, Assembler, C, CMS 2, COBOL, DCL, Fortran, Java, JCL, Jovial, Mumps, MagnaX, Natural, PL/1, PowerBuilder, SQL, Vax Basic, VB 6, + Others

ASSESSMENT

DATABASE

TRANSFORMATION

RE-FACTORIZING

WEB-ENABLED

C, C#, C++, Java + Others

VB . Net

Networking, Software, Hardware, Infrastructure, Licensing

Unit Test, System Test, Integration, Implementation

Maintenance, Support, Warranty Training, Enhancement

Copyright 2010 The Software Revolution, Inc All Rights Reserved
ADM is A Rigorous Agile Method – Component-Oriented, Model-Based, Architecture-First, UML Design
All Change Is Iterative Accomplished By Rules Applied To Models
Role Of The OMG

- The OMG (Object Management Group) chartered the ADM Task Force in 2003
- To Extend UML-Based MDA practices and standards to existing systems.

The OMG ADM TF Mission, Goals and Benefits
- Create Specifications/Promote Industry Consensus on Modernization of Existing Applications
- Leverage Existing OMG Modeling Standards and the MDA Initiative
- Consolidate and Promote Best Practices Leading to Successful Modernization
- Improve ROI in Existing Software by Extending Useful Life of Existing Applications

ADM Practices are documented via ADM Scenarios
- A Set of Guidelines for Applying ADM In Practice In Different Scenarios

ADM Standards are documented via ADM Roadmap
- A Set of OMG Modeling Specifications that Promote Interoperability between modernization tools and service providers

More Information is available at the OMG’s ADM Website:
http://adm.omg.org/
MDA Meets ADM

Model Driven Architecture (MDA) is generally green-field top-down model-driven process for new system development.

- Architectural models and tools (e.g. IBM Rational Eclipse) support agile development

Architecture Driven Modernization (ADM) top-down + bottom-up

- Bottom-up extraction of software and architectural models, followed by
- Top-down reuse for modernization.
- Delivered Modernized Systems are Agile, MDA-enabled

THEN (before ADM)

NOW (MDA + ADM)
ADM Scenarios, Specifications, Tools & Methods – How it all fits together

ADM Modernization Scenarios

I. Application Portfolio Management

II. Application Improvement

III. Language-to-Language Conversion

IV. Platform Migration

V. Non-Invasive Application Integration

IV. Platform Migration

VII. Data Architecture Migration

VIII. Application & Data Architecture Consolidation

IX. Data Warehouse Deployment

X. Application Package Selection & Deployment

XI. Reusable Software Assets / Component Reuse

XII. Model-Driven Architecture Transformation

Standards-Based Services, Tools, Tool Chains, and Tool Suites

OMG Object Services

OMG Interchange Services

ADM Modeling Specifications

KDM

AP

MP

VP

RP

TMTP

ASTM

OMG Repository Common Facility

MOF

UML

XMI
Why define ADM Scenarios?

- Helps envision all potential ADM applications.
- Helps a user determine the tasks, tools and use of the ADM.
- Provides templates for crafting project objectives, plans and related deliverables.
- Defines tasks necessary to complete a given modernization initiative and omits unnecessary tasks that would not apply to such a scenario.
- Allows a user to pinpoint the types of tools necessary to perform these tasks.
- Identifies the universe of modernization scenarios and tasks and provides a guide as to the role of the ADM within modernization in general.
ADM Scenarios and Specifications

- Modernization Projects Involve Collaborative Geographically Dispersed Teams
- ADM Projects Are Tool And Technology Intensive
- Tool Interoperability Enables More Comprehensive Solutions For Complex Problems
- To Interoperate Tools Need To Interchange Standardized Models
- To Collaborate Effectively Vendors and Customers Need Agreed Upon Practices (Scenarios)

<table>
<thead>
<tr>
<th>ADM Scenarios</th>
<th>ADM Specifications</th>
</tr>
</thead>
<tbody>
<tr>
<td>I. Application Portfolio Management</td>
<td>1. Knowledge Discovery Metamodelling Standard</td>
</tr>
<tr>
<td>II. Application Improvement</td>
<td>2. Abstract Syntax Metamodelling Standard</td>
</tr>
<tr>
<td>III. Language-to-Language Conversion</td>
<td>3. Analysis Package</td>
</tr>
<tr>
<td>IV. Platform Migration</td>
<td>4. Metrics Package</td>
</tr>
<tr>
<td>V. Non-Invasive Application Integration</td>
<td>5. Visualization Package</td>
</tr>
<tr>
<td>VI. Services Oriented Architecture Transformation</td>
<td>6. Refactoring Package</td>
</tr>
<tr>
<td>VII. Data Architecture Migration</td>
<td>7. Target Mapping and Transformation Package</td>
</tr>
<tr>
<td>VIII. Application &amp; Data Architecture Consolidation</td>
<td></td>
</tr>
<tr>
<td>IX. Data Warehouse Deployment</td>
<td></td>
</tr>
<tr>
<td>X. Application Package Selection &amp; Deployment</td>
<td></td>
</tr>
<tr>
<td>XI. Reusable Software Assets / Component Reuse</td>
<td></td>
</tr>
<tr>
<td>XII. Model-Driven Architecture Transformation</td>
<td></td>
</tr>
</tbody>
</table>

Copyright 2010 The Software Revolution, Inc All Rights Reserved
Scenario I. Application Portfolio Management

Objective: Establish multi-dimensional knowledge base for managing & transforming applications.

Vertical Product-Lines for Scenario I
- Catalog Application Inventory, Structure and Relationships between Components
- Catalog Architecture of Application Layers, Flow and Interactions among Components and Application Boundaries
- Catalog Dead Code, Obsolete Code and Dead-End Code
- Catalog Definitions, Facts and Rules about Data
- Catalog Derivation, Triggering and Validation Rules
- Catalog Business Processes and Usage Scenarios
- Catalog Flaws and Vulnerabilities
- Group Applications with Like Data, Functionality and Requirements

Vertical Product-Line Support

ADM Repository Technologies

GASTM : Language Neutral Support

SASTM : Horizontal Language Support
### Scenario II. Application Improvement

**Objective:** Create a stable foundation for managing, enhancing or modernizing systems. Reduce application fragility, increase quality.

#### Vertical Product-Lines for Scenario II
- Remove Dead and Redundant Code
- Restructure Poorly Structured Code
- Clean-up Poorly Structured Data
- Evaluate Coupling and Refactor
- Evaluate Cohesion and Refactor
- Evaluation Modularity and Recomponentize
- Evaluate Software Flaws and Vulnerabilities and Refactor

#### Vertical Product-Line Support

<table>
<thead>
<tr>
<th>Standards-Based Services, Tools, Tool Chains, and Tool Suites</th>
</tr>
</thead>
<tbody>
<tr>
<td>ToolA</td>
</tr>
<tr>
<td>ToolB</td>
</tr>
<tr>
<td>ToolC</td>
</tr>
<tr>
<td>ToolD</td>
</tr>
<tr>
<td>ToolE</td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>OMG Object Services</th>
</tr>
</thead>
<tbody>
<tr>
<td>OMG Interchange Services</td>
</tr>
</tbody>
</table>

**ADM Repository Technologies**

- **ASTM**
- **AP**
- **MP**
- **SW/A**
- **VP**
- **RP**
- **TMTP**
- **KDM**

**GASTM : Language Neutral Support**

**SASTM : Horizontal Language Support**

<table>
<thead>
<tr>
<th>C++</th>
<th>Java</th>
<th>Ada</th>
<th>C#</th>
<th>VB.Net</th>
<th>C</th>
<th>COBOL</th>
<th>FORTRAN</th>
<th>Jovial</th>
<th>VMS</th>
<th>VAX</th>
<th>BASIC</th>
<th>PL/1</th>
<th>JCL</th>
<th>Etc.</th>
</tr>
</thead>
</table>

---

Copyright 2010 The Software Revolution, Inc All Rights Reserved
Scenario III. Language-to-Language Conversion & Scenario IV. Platform Migration

Objective: Convert applications to new language and / or run time platform. Scenarios can be performed separately or together.

Scenario III:
Converts source code to new language or language level with run time environment

Scenario IV:
Migrates application to new hardware and / or operating system

Vertical Product-Lines for Scenario III
Top Down Language To Language (L2L) Conversion Scenarios
5GL to 4GL
5GL to 4GL to 3GL
5GL to 4GL to 3GL to 2GL
4GL to 3GL
4GL to 3GL to 2GL
3GL to 2GL

Bottom Up Language To Language (L2L) Conversion Scenarios
2GL to 3GL
2GL to 3GL to 4GL
2GL to 3GL to 4GL to 5GL
3GL to 4GL
3GL to 4GL to 5GL
4GL to 5GL

Vertical Product-Lines for Scenario IV

Vertical Product-Line Support
ADM Repository Technologies
ASTM AP MP SW/A VP RP TMTP KDM

GASTM : Language Neutral Support

SASTM : Horizontal Language Support

Copyright 2010 The Software Revolution, Inc All Rights Reserved
Objective: Create the option of accessing host applications & data via Web-based interfaces*.

Vertical Product-Lines for Scenario V
- **Partition by Product (what system produces)**
  - Wrap Functions that Create Products with SOA Interfaces
- **Partition by Application Layer**
  - Wrap Application Layers with SOA Interfaces
- **Partition by Business Process**
  - Wrap Functions supporting Business Process with SOA Interfaces
- **Partition by Data Sets**
  - Wrap Legacy Data Bases with SOA Interfaces

*Non-invasive approach only impacts front-ends. Underlying architecture remains intact. Courtesy Bill Ulrich

**Vertical Product-Line Support**

**ADM Repository Technologies**
- **ASTM**
- **AP**
- **MP**
- **SW/A**
- **VP**
- **RP**
- **TMTP**
- **KDM**

**GASTM : Language Neutral Support**

**SASTM : Horizontal Language Support**
- **C**
- **C++**
- **Java**
- **Ada**
- **C#**
- **VB.Net**
- **C**
- **COBOL**
- **FORTRAN**
- **Jovial**
- **VMS**
- **VAX**
- **BASIC**
- **MUMPS**
- **PL/1**
- **JCL**
- **ETC.**
Scenario VI. Services Oriented Architecture (SOA) Transformation

**Objective:** Create a framework for constructing and interlinking back-end systems with the goal of making applications more agile.

Vertical Product-Lines for Scenario VI

- **Transform by Product Lines (what system produces):**
  - Modernize Functions that Create Products into SOA

- **Transform by Application Layers (how system is layered):**
  - Modernize Application Layers into SOA

- **Transform by Business Process:**
  - Modernize Functions supporting Business Process into SOA

- **Transform by Business Rules:**
  - Modernize Application Logic into Business Rule Engines and SOA

- **Transform by Data Sets (how data is organized):**
  - Map Legacy Data Bases into SOA Enabled Repositories

Existing applications

- Identify, extract, document & publish services within SOA framework

Repository identifying reusable services

- ADM Repository Technologies
  - Standards-Based Services, Tools, Tool Chains, and Tool Suites

- OMG Object Services
  - OMG Interchange Services

- ADM Roadmap
  - ASTM AP MP SW/A VP RP TMTP KDM

- OMG Repository Common Facility
  - MOF UML XMI

GASTM : Language Neutral Support

SASTM : Horizontal Language Support
Scenario VII. Data Architecture Migration

Objective: Consolidate, cleanse, redesign & migrate existing data structures to relational database. Refactor applications as required.

- Rationalized data structures
- Physical database
- Data cleansing process
- Physical data design

Existing data file / database formats
- Relational data model
- Refactored applications

Standards-Based Services, Tools, Tool Chains, and Tool Suites
- Tool A
- Tool B
- Tool C
- Tool D
- Tool E

OMG Object Services
- OMG Interchange Services

ADM Roadmap
- KDM
- AP
- MP
- VP
- RP
- TMTP

OMG Repository Common Facility
- MOF
- UML
- XMI

Vertical Product-Lines for Scenario VII
- Model Existing Data Files and Database Formats
- Map to Modern RDBMS & Rationalize and Normalize Schema
- Map Legacy Data To Modern RDBMS Repository
- Cleanse Data to Remove Duplication & Standardize Formats
- Refactor Data Manipulation Code in Application To Use Modern RDBMS

Vertical Product-Line Support
- ADM Repository Technologies
  - ASTM
  - AP
  - MP
  - SW/A
  - VP
  - RP
  - TMTP
  - KDM

GASTM : Language Neutral Support
- SASTM : Horizontal Language Support
  - C
  - +
  - Java
  - Ada
  - C#
  - VB.Net
  - COBOL
  - FORTRAN
  - JOVIAL
  - VAX
  - BASIC
  - MUMPS
  - PL/1
  - JCL
  - Etc.
Objective: Consolidate multiple redundant or related systems into a common data & application architecture.

- Integrate, automate common processes
- Consolidate & redesign cross-functional data
- Migrate baseline systems to new architecture
- Identify & consolidate redundant logic
- Create & phase in reusable modules

Vertical Product-Lines for Scenario VIII

Identify Like-Applications Requirements, Data and Functions
Rationalize Data and Functions To Facilitate Comparison
Score Card Similar & Redundant Functionality & Data
Modernize Best-Of-Breed Baseline System
Cherry-Pick and Modernize Useful Logic and Data From Weakest Links and Integrate with Best-of-Breed Baseline

Existing Applications
Consolidation tasks
Consolidated data & application architecture

Standards-Based Services, Tools, Tool Chains, and Tool Suites
OMG Object Services
OMG Interchange Services

ADM Roadmap
OMG Repository Common Facility

Vertical Product-Line Support
ADM Repository Technologies
ASTM AP MP SW/A VP RP TMTP KDM

GASTM : Language Neutral Support
SASTM : Horizontal Language Support
Scenario IX. Data Warehouse Deployment

Objective: Create non-operational, consolidated view of data & make this data available to business users in ad hoc formats.

Vertical Product-Lines for Scenario IX
- Model Existing Data Files and Database Formats
- Map Model of Existing Data Formats To Consolidated, Standardized Warehouse Model
- Map Existing Data To Common Warehouse Metamodel Repository

Vertical Product-Line Support

ADM Repository Technologies

GASTM : Language Neutral Support

SASTM : Horizontal Language Support
Scenario X. Application Package Replacement

**Application Package Selection & Deployment – phase one**

**Objective:** Provide objective analysis of how well various packages meet strategic information requirements.

1. **Vertical Product-Lines for Scenario X**
   - Identify Like-Applications Requirements, Data and Functions
   - Rationalize Data and Functions To Facilitate Comparison
   - Score Card Similar & Redundant Functionality & Data
   - Install COTS as Baseline
   - Cherry-Pick and Modernize Useful Logic and Data From Existing Application and Integrate with COTS Baseline

2. **Vertical Product-Line Support**
   - **ADM Repository Technologies**
   - **GASTM : Language Neutral Support**
   - **SASTM : Horizontal Language Support**

---

**Copyright 2010 The Software Revolution, Inc All Rights Reserved**
Objective: Create a repository of reusable components.

Existing Applications

Data / Rule Consolidation

Business Rule & Data Extraction

Reuse Repository

Component Repository

Development Projects & Web Services

Vertical Product-Lines for Scenario XI

Mine Applications For Reusable Requirements, Data and Functions

Rationalize Data and Functions To Facilitate Comparison

Score Card Similar & Redundant Functionality & Data


Transform Selected Functions, Logic and Data From Existing Applications and Integrate Into Reuse Repository expressed as Componentized Intermediate Language, Modern Code, or as Rule Sets.

Vertical Product-Line Support

ADM Repository Technologies

ASTM AP MP SW/A VP RP TMTP KDM

GASTM : Language Neutral Support

SASTM : Horizontal Language Support

OMG Interchange Services

OMG Repository Common Facility

MOF UML XMI
Scenario XII. Model-Driven Architecture Transformation

Objective: Migrate existing applications to an environment in which systems are maintained in models.

- Extract business rules & data definitions
- Purge implementation dependent logic
- Rationalize/consolidate data & logic as needed
- Migrate/merge extracted logic & data definitions into applicable models

Vertical Product-Lines for Scenario XII

- Document Application Requirements, Data and Functions as UML 2.0, BSBR or other Applicable OMG models
- Transform and Refactor Application To Cleanse and Modernize Application Code, Data & Logic and Validate In Target Implementation
- Separate “Business” From “Technical” Models via Domain Analysis
- Generate Application (in whole or piecewise) from Architecture-Level Models
- Validate Accuracy of Architecture Models By Comparing To Transformed Code (where applicable)

ADM Repository Technologies

<table>
<thead>
<tr>
<th>ASTM</th>
<th>AP</th>
<th>MP</th>
<th>SW/A</th>
<th>VP</th>
<th>RP</th>
<th>TMTP</th>
<th>KDM</th>
</tr>
</thead>
</table>

GASTM : Language Neutral Support

SASTM : Horizontal Language Support

| C++ | Java | Ada | C# | VB.Net | C | COBOL | FORTRAN | JOVIAL | VMS | VAX | BASIC | MUMPS | PL/1 | JCL | Etc. |
|-----|------|-----|----|--------|--|-------|----------|--------|-----|-----|-------|-------|------|-----|-----|-----|

Sample models

Existing applications & data definitions

Standards-Based Services, Tools, Tool Chains, and Tool Suites

OMG Object Services

OMG Interchange Services

ADM Roadmap

OMG Repository Common Facility

MOF < UML < XMI
Architecture Driven Modernization
Standardization

- **Standards-Based Tools**
  - Store and Access Models in Industry Standard Formats (XMI™)
  - Manage IT Artifacts as High Fidelity Models in Repositories.
  - Interoperate With Other Tools By Interchanging Models

- **Standards-Based Tool Chains**
  - Series of Tools will Cooperatively Produce A Work Product
  - Cooperate By Interchanging Models.

- **Standards-Based Tool Suites**
  - Collections of Tools that Interoperate Using Models

- **Standard-Based Service Providers**
  - Offer Services Based Upon Tools, Tool Chains and Tool Suites
  - Collaborate with Customers and Other Services Providers Via Interchange of Industry Standard Models (e.g. Eclipse, MOF™, XMI™, UML™)

- **Standard-Based Enterprise**
  - IT Artifacts (such as Data and Applications) are Managed In Repositories As Standards-Based Models
  - IT Artifact Management employs Service Providers, Standards-based Tools, Tool Chains, and Tool Suites
Ten Groundbreaking Architecture Driven Modernization Projects
Case Study Overviews
TSRI Case Study: ITT Corporation SENSOR

Project Summary
ITT Corporation awarded a sole-source contract to The Software Revolution, Inc. (TSRI) for modernization of the COBRA DANE Radar Calibration System (SCRS) of the Ballistic Missile Early System (BMEWS) under the Air Force’s System Engineering & Sustainment Integrator (SENSOR) program.

Services
- Code Transformation
- Automatic Re-Factoring
- Semi-Automatic Re-Factoring
- System Integration, Testing & Implementation
- Final “To-Be” Documentation

Results
- Project successfully completed in 8 months
- Created a new JANUS™ Ada parser
- JANUS™ rules engine parser was tuned to address the SENSOR Ada & Fortran
- Fully modernized 380,300 LOC of highly complex mission-critical Ada & Fortran to C++ & Java
TSRI Case Study: Raytheon Corporation TCS/MCS

Project Summary
Raytheon awarded a sole-source contract to The Software Revolution, Inc. (TSRI) for modernization of the Navy’s Terminal Control and Modem Control (TCS & MCS) satellite system modules.

Services
- Code Transformation
- Automatic Re-Factoring
- Semi-Automatic Re-Factoring
- System Integration & Test Support

Results
- Project successfully completed in 5 months
- JANUS™ rules engine parser was tuned to address TCS’ & MCS’ Ada & Fortran, respectively
- Fully modernized the 190,772 LOC of Ada & 91,949 LOC of Fortran
- Met stringent Raytheon/Navy coding standards
- Received Raytheon & U.S. Small Business Administration awards of excellence
Project Summary
Lockheed Martin Corporation awarded a sole-source contract to The Software Revolution, Inc. (TSRI) for modernization of the Acoustic Signal Processing Suite aboard the Navy’s P-3C submarine hunting aircraft.

Services
• Legacy “As-Is” Documentation
• Code Transformation
• Automatic Re-Factoring
• Semi-Automatic Re-Factoring
• System Integration, Testing & Implementation
• Final “To-Be” Documentation

Results
• Project successfully completed in 14 months
• JANUS™ rules engine parser was tuned to address the P-3C’s Ada
• Fully modernized 650,000+ LOC of highly complex mission-critical Ada to C++
TSRI Case Study: Thales EATMS

Project Summary
Thales awarded a sole-source contract to The Software Revolution, Inc. (TSRI) for a modernization of several modules of the European Air Traffic Management System (EATMS).

Services
- Legacy “As-Is” Documentation
- Code Transformation
- Automatic Re-Factoring
- Semi-Automatic Re-Factoring
- System Integration & Test Support
- Final “To-Be” Documentation

Results
- Addressed the Flight Profile Library, Minimum Safe Altitude Warning System, and Air-Ground Data Processor modules of EATMS
- JANUS™ rules engine parser was tuned to address the legacy Ada code of these EATMS modules
- Fully modernized nearly 250,000 LOC of Ada code to both C++ & Java
- Effort included extensive re-factoring to meet precise, mission-critical coding standards
- Ultra High Assurance Air Traffic Control Software
Project Summary

Raytheon awarded a sole-source contract to The Software Revolution, Inc. (TSRI) for a modernization of several modules of the Patriot Missile.

Services

- Legacy “As-Is” Documentation
- Code Transformation
- System Integration & Test Support
- Final “To-Be” Documentation

Results

- Addressed the Fire Platoon Simulation System Preprocessor, and the Battalion Simulation Support System and its preprocessor (PRED) modules of the Patriot Missile
- JANUS™ rules engine parser was tuned to address the legacy Fortran code of these missile modules
- Fully modernized nearly 600,000 LOC of Fortran code to C++
TSRI Case Study: SAIC EOSS

Project Summary

Science Application International Corporation (SAIC) awarded a sole-source contract to The Software Revolution, Inc. (TSRI) for modernization of the Navy’s Engineering Operational Sequencing System (EOSS).

Services

- Legacy “As-Is” Documentation
- Target Architecture & Interface Design
- Code Transformation
- Automatic Re-Factoring
- Database Conversion
- System Integration, Testing & Implementation
- Final “To-Be” Documentation
- Support to DoN “Opportunity Analysis for Legacy System Modernization”

Results

- Project successfully completed in 5 months
- JANUS™ rules engine parser was created to address the EOSS VAX-Basic
- Design new target architecture & fully modernized 37,483 LOC of highly complex mission-critical VAX-Basic to Java
- Converted EOSS database structure from multi-ISAM ASCII to Oracle 9i
- Navy calculated that effort provided a 2.47:1 ROI with much higher ROI expected for languages already addressed by TSRI

Project Contacts:

Upon Request
TSRI Case Study: SAIC VHA

Project Summary
Science Application International Corporation (SAIC) awarded a sole-source contract to The Software Revolution, Inc. (TSRI) for modernization of the Voluntary Time Keeping module of the Veterans Health Administration’s (VHA’s) VistA system.

Services
- Code Transformation
- Automatic Re-Factoring
- Semi-Automatic Re-Factoring
- System Integration, Testing & Implementation
- Final “To-Be” Documentation

Results
- Project successfully completed in 4 months
- Created a new JANUS™ MUMPS parser
- JANUS™ rules engine parser was tuned to address the VHA MUMPS
- Fully modernized 300,000+ LOC of highly complex MUMPS to Java

Project Contacts:
Upon Request
TSRI Case Study: TRW Milstar Satellite

Project Summary

TRW awarded a sole-source contract to The Software Revolution, Inc. (TSRI) for modernization of the Advanced Extremely High Frequency module of the Milstar satellite

Services

- Code Transformation
- Semi-Automatic Re-Factoring
- System Integration & Test Support

Results

- Project successfully completed in 1 months
- JANUS™ rules engine parser was tuned to address MILSTARS’ Jovial
- Fully transformed the 143,000 LOC of J3 Jovial
- Replaced 16-Bit Big-Endian with 32 Bit Little Endian Environment
- Code fully integrated into Milstar & awaiting final system launch

Project Contacts:
Upon Request
TSRI Case Study: National Endowment for the Arts

Project Summary

The National Endowment for the Arts (NEA) awarded a sole-source contract to The Software Revolution, Inc. (TSRI) for modernization of the NEA’s business systems (Financial Management – Grants Management – Automated Panel Bank)

Project Contacts:
Upon Request

Services
• Legacy “As-Is” Documentation,
• Code Transformation
• Automatic Re-Factoring,
• Semi Automatic Re-Factoring
• System Integration, Testing & Implementation
• Final “To-Be” Documentation
• Future System Enhancement Development
• Database Conversion: OpenVMS RMS Flat Files to a Microsoft SQL environment

Results
• Project successfully completed in 3 fiscal phases over 24 months
• TSRI performed successfully after another contractor using a manual approach had failed & wasted $1.6M
• JANUS™ rules engine parser was tuned to address NEA’s Wang COBOL
• Fully modernized the 656,000 LOC of Wang-COBOL & RMS flat files to C++ & SQL Server environment & 3,270 screens into a MS Windows environment
**Project Summary**

Northrop Grumman awarded three separate sole-source contracts to The Software Revolution, Inc. (TSRI) for a transformation demonstration and subsequent modernization of the Increments 1 & 3 of the Air Force’s REMIS system.

**Services**

- Legacy “As-Is” Documentation
- Code Transformation
- Automatic Re-Factoring
- Semi-Automatic Re-Factoring
- System Integration & Test Support
- Final “To-Be” Documentation

**Results**

- Transformation demonstration & Increment 1 successfully completed in 1 month & 7 months respectively, with Increment 3 currently on-going
- JANUS™ rules engine parser was tuned to address REMIS’ Tandem COBOL code
- Fully modernized over 400,000 LOC of Tandem COBOL to both C++ & Java code
- Supported development of the API layer to allow system interface to the Air Force GCSS Integration Framework
Offices in Kirkland, Washington

Technology Rooted in Early Artificial Intelligence Projects
- 1983 USAF Knowledge Based Software Assistance (KBSA) Program
- 1988-1994 Boeing Artificial Intelligence Lab

Member company OMG Architecture Driven Modernization ADM Task Force participating in:
- Abstract Syntax Tree Meta-Modeling (ASTM)
- Knowledge Discovery Meta-Modeling (KDM)
- Structured Metrics Meta-Model (SMM)

Industrial Awards
- Northrop Grumman “Small Business of Year” Award 2002
- Raytheon “Supplier of Value” Award 2005
- Small Business Administration’s 2005 “Administrators Award of Excellence”

Over 75 Automated Modernization Projects since 2000
- 100% Project Successfully Completed
- References Upon Request