

How Architecture-Driven Modernization Is Changing Information System Modernization



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



Local Support for TSRI
Ottawa, Ontario
www.asmg-ltd.com



- **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

- Offices in Ottawa Ontario Canada
- Providing Local Support for TSRI Capability
- Founded 1995
- Affiliations:
 - Co-chair C4I Domain Task Force
 - Co-chair Emergency, Crisis and Major Event Management Special Interest Group at OMG
- Corporate Focus
 - Information sharing, Protection and Security
 - Information Assurance
 - Cross-Agency, Enterprise, Information, Systems and Technical Architectures
 - Architecture Driven Information Solutions
 - Policy Enforcement Services
 - Open Standards
 - Shared Operational Exchange Services IEDM
 - Information Exchange Framework
 - Unified Profile for MODAF and DODAF
 - Emergency Management System Interoperability Framework
- Products
 - Common Object Interoperability Layer (COIL)

<http://www.asmg-ltd.com/>

Why Modernize Legacy Systems?

Doing Nothing Is The Pathway To Decline

- ❑ **Trillion Lines of Code – Most Written in Obsolete Legacy Languages!**
 - ❑ **Estimated Cost of \$15B in modernization costs to the Government of Canada**
- ❑ **“Supported by Declining Generation of Obsolete Developers” ***
- ❑ **“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” ***
- ❑ **“Package Implementation and Complete Rewrites Represent the Biggest Disasters in IT” ***
- ❑ **Past Replacement Efforts Have A Shocking 63% Failure Rate !!**
- ❑ **Past Failure Should Never Be A Justification For Doing Nothing**

Why Modernize Legacy Systems?

Reason	Description
Skills shortage	fewer and fewer staff and contractors have the skills and knowledge to use older programming languages and source code structures.
Vendor support	vendors (system developers and integrators) may no longer exist or no longer support older products.
Regulatory compliance	outdated systems may be hard to update to comply with changing laws, regulations, and industry standards.
Maintenance costs	costs go up because aging systems are very complex and difficult to maintain, there are few service providers, and parts are scarce and often very costly. These cost are consuming greater and greater portions of the IM/IT budgets of many organizations; leaving fewer resource to enhance capability.
Access to data	information becomes increasingly cumbersome to extract and analyze as data structures age.
Meeting client expectations	outdated systems cannot be modified to support modern technologies and meet expectations, such as 24/7 availability and workflow.
Security	Legacy outdated systems cannot always be modified to conform to changing security requirements.
Green IT initiatives	outdated IT systems are generally not energy efficient and are hard to modify to reduce their environmental impact.
Disaster recovery	the more outdated the system, the harder it is to recover data after a disaster.
Interoperability	outdated systems cannot be upgraded to support the growing needs of government to interoperate (share data in a secure and trusted manner) within and/or between government agencies, private sector agencies and the public.

What is Architecture Driven Modernization?

- ❑ ADM Is A Software Modernization Approach developed at the Object Management Group (OMG)
- ❑ ADM enables Predictable Repeatable Practices for highly-automated (model and rule based) modernization of Information Systems
- ❑ ADM is seeking to establish
 - ❖ Best Practices, standards and Guidelines information system modernization Services.
 - ❖ Interoperability Standards For Modernization Tools
- ❑ The OMG ADM Task Force is the only formal body for the specification of Modernization Standards in industry
- ❑ ADM Delivers:
 - ❖ High-Quality Modernized Systems
 - ❖ Reducing the Time and Cost of Modernization,
 - ❖ Addressed a wide range of source to target languages transformations,
 - ❖ Applicable To All Sectors: Military, Government, Commercial

ADM Is the only viable means for evolving the billions of Lines of aging information systems software in a manner that yields sustainable capability into the future (architecture, documentation and institutional memory)

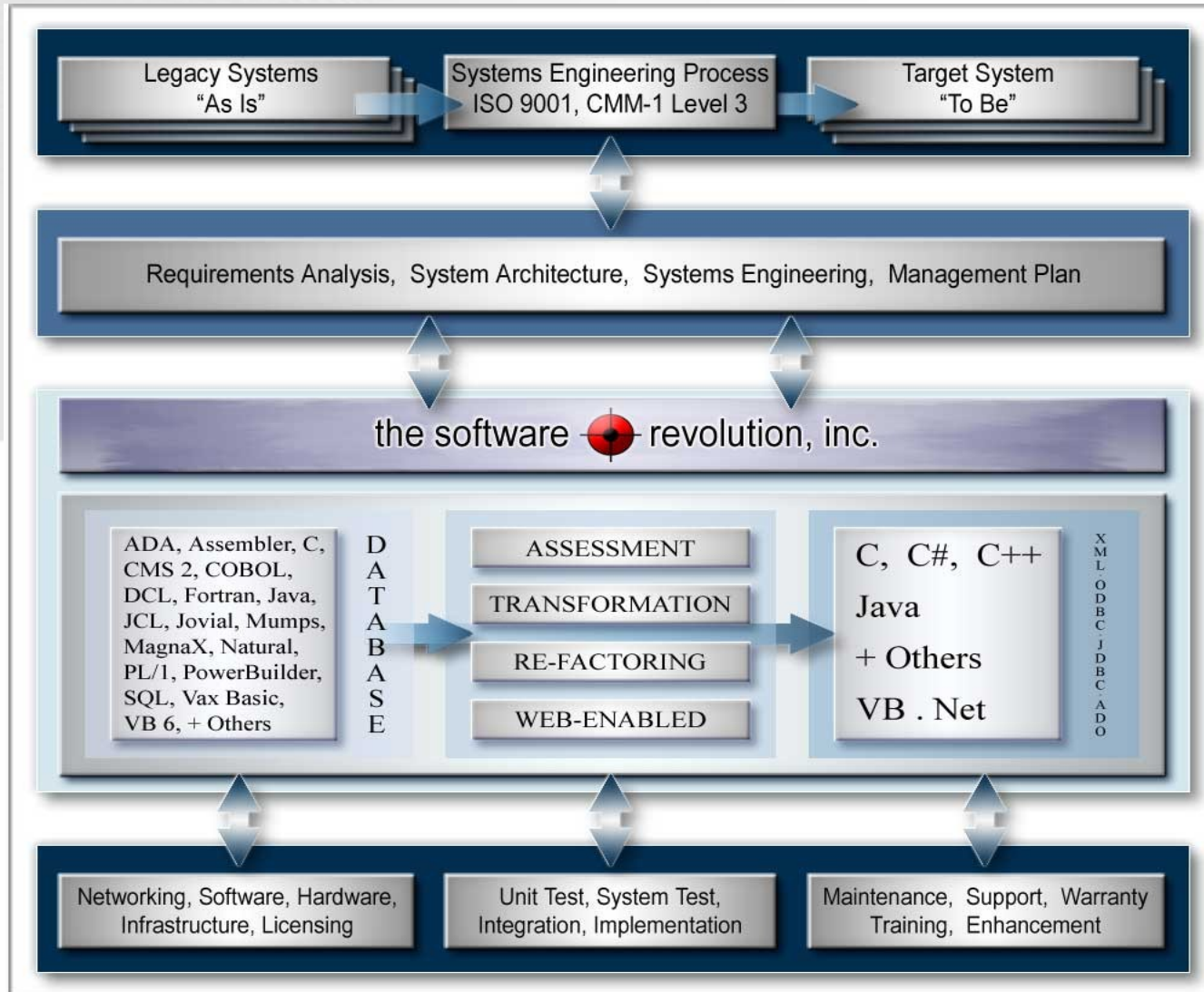
ADM Delivers Across Legacy Languages, Platforms and Applications

Legacy Source Code	System Application
COBOL (4 Dialects)	<ul style="list-style-type: none"> • Administration • Health Maintenance • Logistics • Finance
ADA	<ul style="list-style-type: none"> • Aircraft Display • Satellite Terminal • Mission Planning
Assembler	<ul style="list-style-type: none"> • Strategic Missile Defense
Jovial	<ul style="list-style-type: none"> • Satellite Tracking • Aircraft Control • Command & Control
CMS-2	<ul style="list-style-type: none"> • Tactical Weapons Display
Fortran	<ul style="list-style-type: none"> • Weather Tracking • Strategic Missile Defense • Strategic Warfare Planning
C	<ul style="list-style-type: none"> • Homeland Defense • Strategic Missile Defense
VAX Basic	<ul style="list-style-type: none"> • Shipboard Engineering
MUMPS	<ul style="list-style-type: none"> • Hospital Operations
PowerBuilder	<ul style="list-style-type: none"> • Healthcare Services
Magna-X	<ul style="list-style-type: none"> • Healthcare Services

ADM REDUCES MODERNIZATION Cost, Schedule & Risk as Compared to Alternative Methods

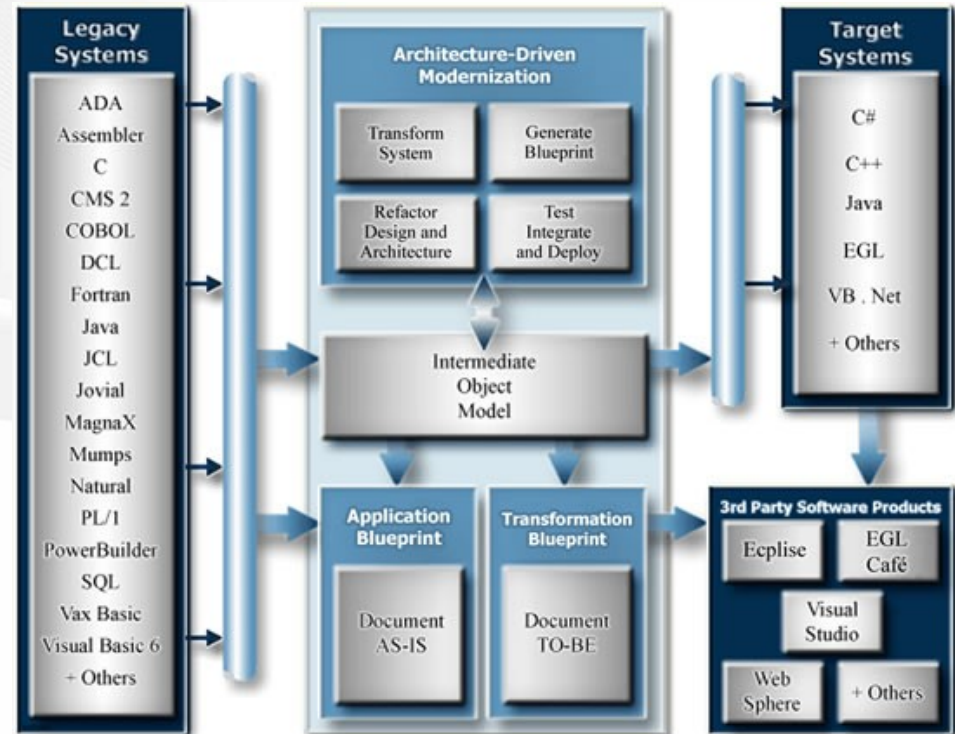
	Integrator	System	Code	SLOC	TTC
Civilian	SAIC & Open Source	Fileman, Vista Pilot and OpenVista	MUMPS to Java	2.5M	6 mo
	Health Care Insurance Co	BlueCross Provider System	PowerBuilder & Magna X to Java	170K	3 mo.
	AMDOCS	Billing System	COBOL to C	5.1M	7mo.
	Thales Air Systems	French Air Traffic Management	Ada to Java	495K	12 mo.
	Thales Air Systems	Nordic Air Traffic Management	Ada to Java	541K	9 mo
	Thales Air Systems	Australian Air Traffic Management	Ada to Java	638K	9 mo
	Unisys	NY State Dept. of Criminal Justice	COBOL Documentation	308K	2 mo.
	NEA	Grant & Business Systems	COBOL to C++	656K	7 mo.
	SAIC	Veteran's Health Administration	MUMPS to Java	300K	4 mo.
	State of OR	Employee Retirement System	COBOL to C# .Net	250K	4 mo.
	State of WA	Off. of Super of Public Instruct.	COBOL to C# .Net	191K	5 mo.
	TriGeo	Sim v4.0 (Internal Product)	Java Docs & Re-Fact.	370K	2 mo.
	EDS	Proof-of-Concept	P/L 1 to Java	50K	7 mo.
	CSC	Bureau of Immigration	COBOL to C++	17K	3 mo.
	Military	Boeing	ALCA - Czechoslovakia	Jovial to C++	9K
LMCO		P-3C	Ada - C++	656K	14 mo.
ITT		BMEWS - Cobra Dane	Ada/Fortran - C++	380K	8 mo.
Raytheon		Satellites	Ada/Fortran - C++	284K	5 mo.
L-3		VTT	Ada - C++	77K	3 mo.
LMCO		SAC Strategic Planning System	Ada - C++	40K	2 mo.
DSR		E-2C ACFT	Ada - C++	20K	5 mo.
USAF		CAMS	COBOL Docs	1M	6 mo.
NGC		REMIS	COBOL - C++	400K	7 mo.
Dyncorp		WCSRS	COBOL - C++	90K	5 mo.
ITT		BMEWS - ROSA	Fortran/C Docs	2M	4 mo.
Raytheon		Patriot Missile	Fortran - C++	200K	6 mo.
Litton PRC		Strategic Air Command	Fortran - C++	50K	4 mo.
Raytheon		WDAC	Fortran - C++	40K	1 mo.
SAIC		EOSS	VAX Basic - Java	38K	5 mo.
TRW	MILSTAR	Jovial to C++	143K	1 mo.	
USAF	F-16 Decis	Jovial to C++	50K	4 mo.	

ADM Delivers Modernization using Best Industry Practices



ADM is a Rigorous, Iterative and Agile Method

- Component-Oriented,
- Model Drive Architecture (MDA)
- Modeling Standards (UML)
- Extensible Modernization Rules



All Change Is Iterative, accomplished By Rules Applied To Models.

ADM delivers Highest Quality Modernized Code at 100% Automation Levels.

ADM is the OMG Sanctioned Approach To Software Modernization

- ❑ 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 Is:
 - ❖ 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 in ADM Meta-model Specifications
 - ❖ 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/>

ADM Extends and Completes OMG's Model-Driven Architecture (MDA)

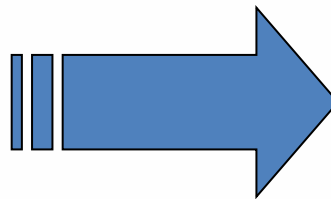
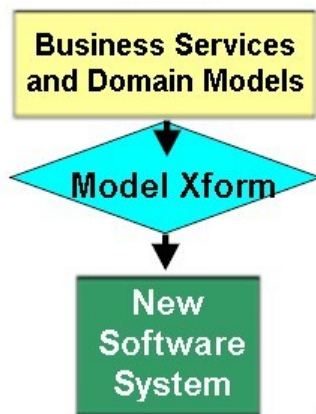
Model Driven Architecture (MDA) is generally used as a green-field top-down model-driven process for *new system development*.

- ❑ Top-Down Architectural modeling and development tools (e.g. IBM Rational Eclipse) support agile development

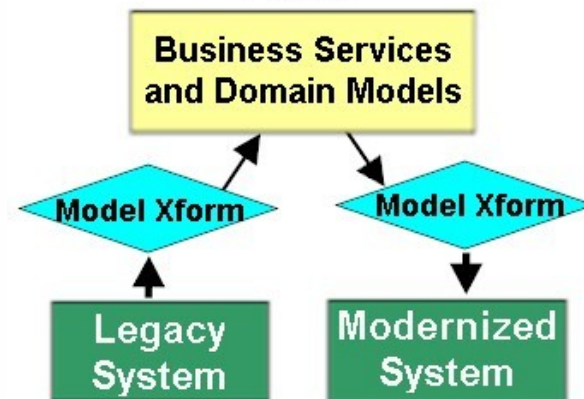
Architecture Driven Modernization (ADM) is both bottom-up + top-down

- ❑ Bottom-Up extraction of software models (code + design + architecture)
+ Top-Down reuse to create modernized systems.
- ❑ Delivers Agile, Modern Object-Oriented Modernized Systems along with UML Design

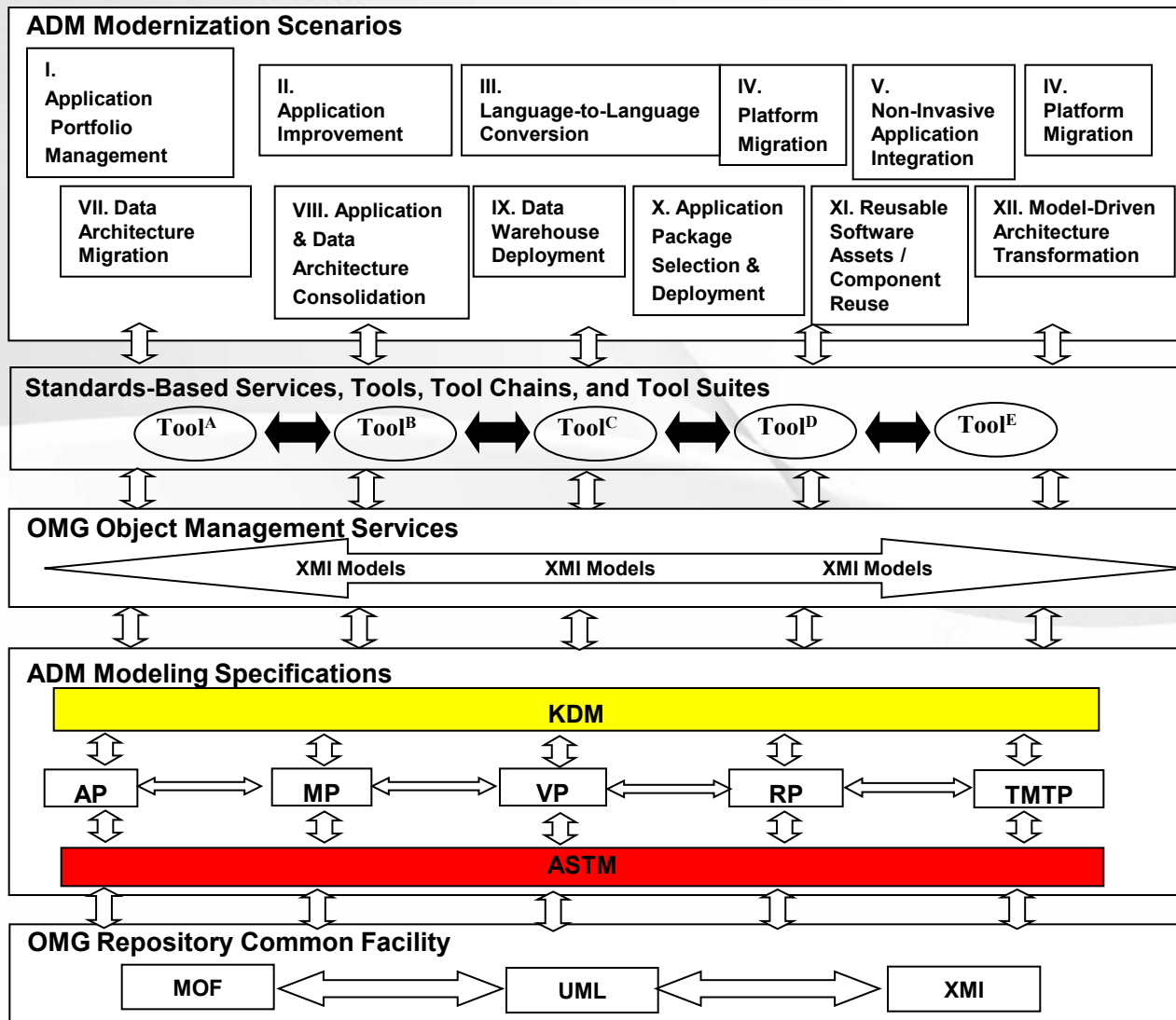
THEN (before ADM)



NOW (MDA + ADM)



How Do The ADM Scenarios, Specifications, And Vendors Tools & Services Fit Into the OMG Ecosystem?



How Can You Use the ADM Scenarios To Successfully Modernize Your Legacy System?

Use ADM scenarios to plan your modernization strategy.

- Envision your Modernization Project from End-to-End.**
- Use ADM Scenarios as templates for crafting project plans, objectives and related deliverables.**
- Understand typical modernization scenarios and tasks**
- Plan Tasks and Choose the Right Tools and Methods.**
- Avoid Waste by knowing what tasks are necessary and omit unnecessary tasks that do not apply.**
- Pinpoint the products services and service providers that can best support the scenario you select.**

ADM Scenarios + Specifications

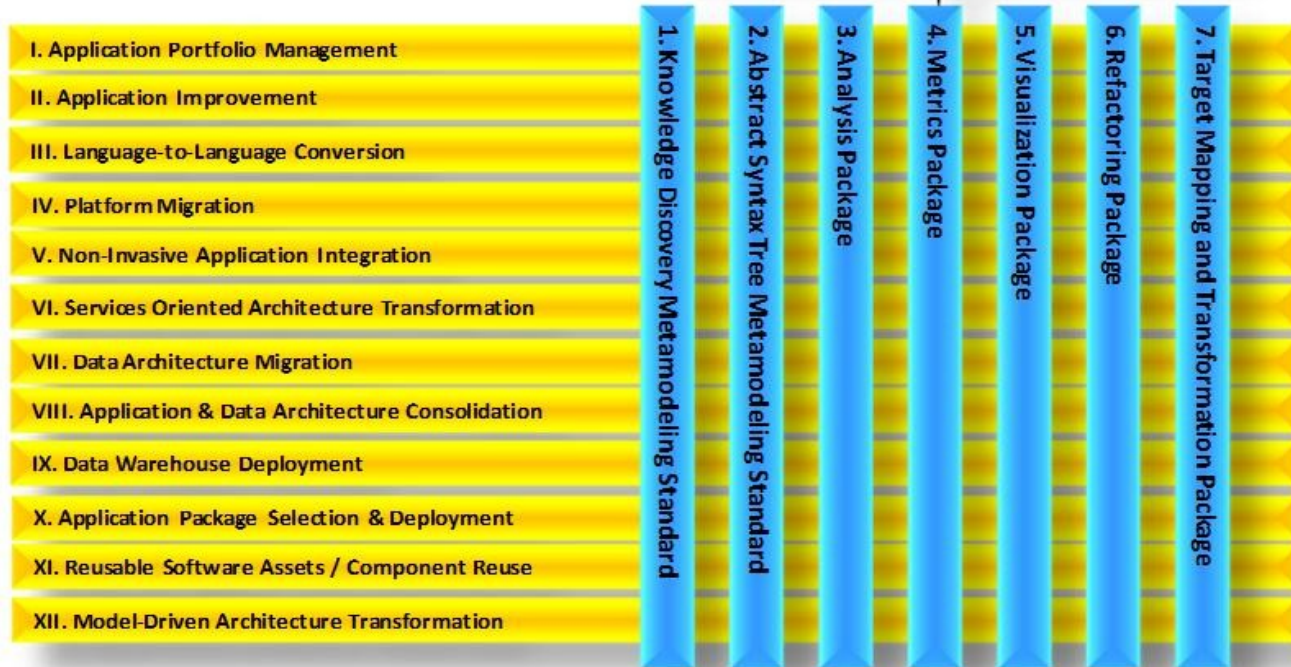
ADM Modernization Projects are Growing In Size and Complexity, and Technology-Intensive, Involving Dispersed Collaborative Teams.

To Collaborate Effectively These Teams Need Agreed Upon Practices, and Tools That Can Work Together To Solve Bigger Problems

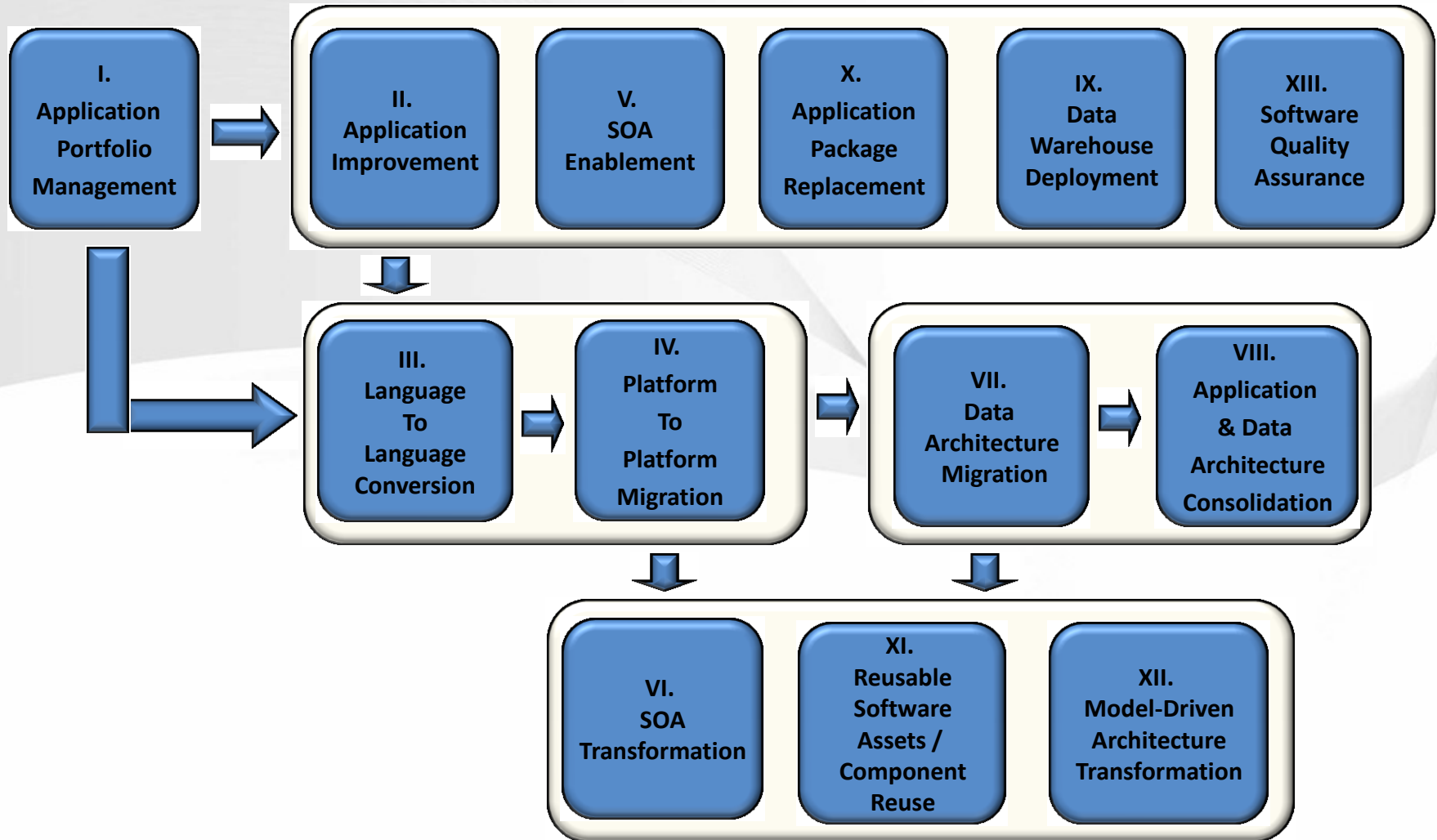
ADM Scenarios + ADM Specifications

ADM Specifications

ADM Scenarios

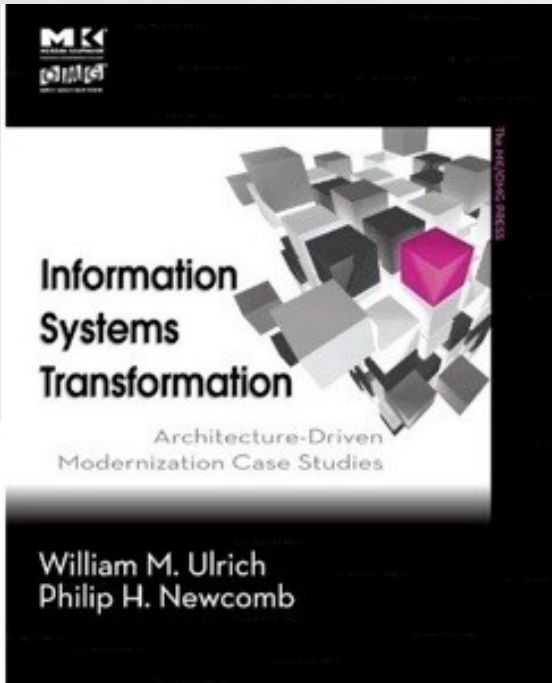


Combine Modernization Scenario *



* Object Management Group (OMG) Architecture Driven Modernization Task Force (ADMTF) Modernization Scenarios

Book Information



[Information Systems Transformation: Architecture-Driven Modernization Case Studies](#)

By William M. Ulrich and Philip H. Newcomb

Published by Morgan Kaufmann

- Acts as a one-stop shopping reference and complete guide for implementing various modernization models including core concepts, common scenarios, and a guide for getting started.
- Concepts are illustrated with real-life examples from various modernization projects, allowing you to immediately apply tested solutions and see results.
- Ten chapters containing in-depth modernization case studies, covering multiple platforms, industries and government agencies from four different countries.

Grady Booch - Book Review

Ulrich and Newcomb's book offers a comprehensive examination of the challenges of growing software-intensive systems. Every system has an architecture, but as the authors explain, it is only by continuous, intentional architectural transformation that one can attend to costs while simultaneously creating a mechanism wherein innovation may flourish.

I enjoyed the many case studies. Every circumstance is unique, but the authors have offered up best practices for systems modernization from their experience. Their focus on architecture as an artifact for governance is sound, but they go far beyond the technical issues and address many of the social and economic practices that help one evolve a good technical architecture.

From the moment one writes a line of code, it becomes legacy, and that legacy accumulates. Whether it becomes a cause of innovation inertia or a source of future value is a factor of how it is continuously modernized.