Turning Product Development Into Competitive Advantage

Mark Wasserman
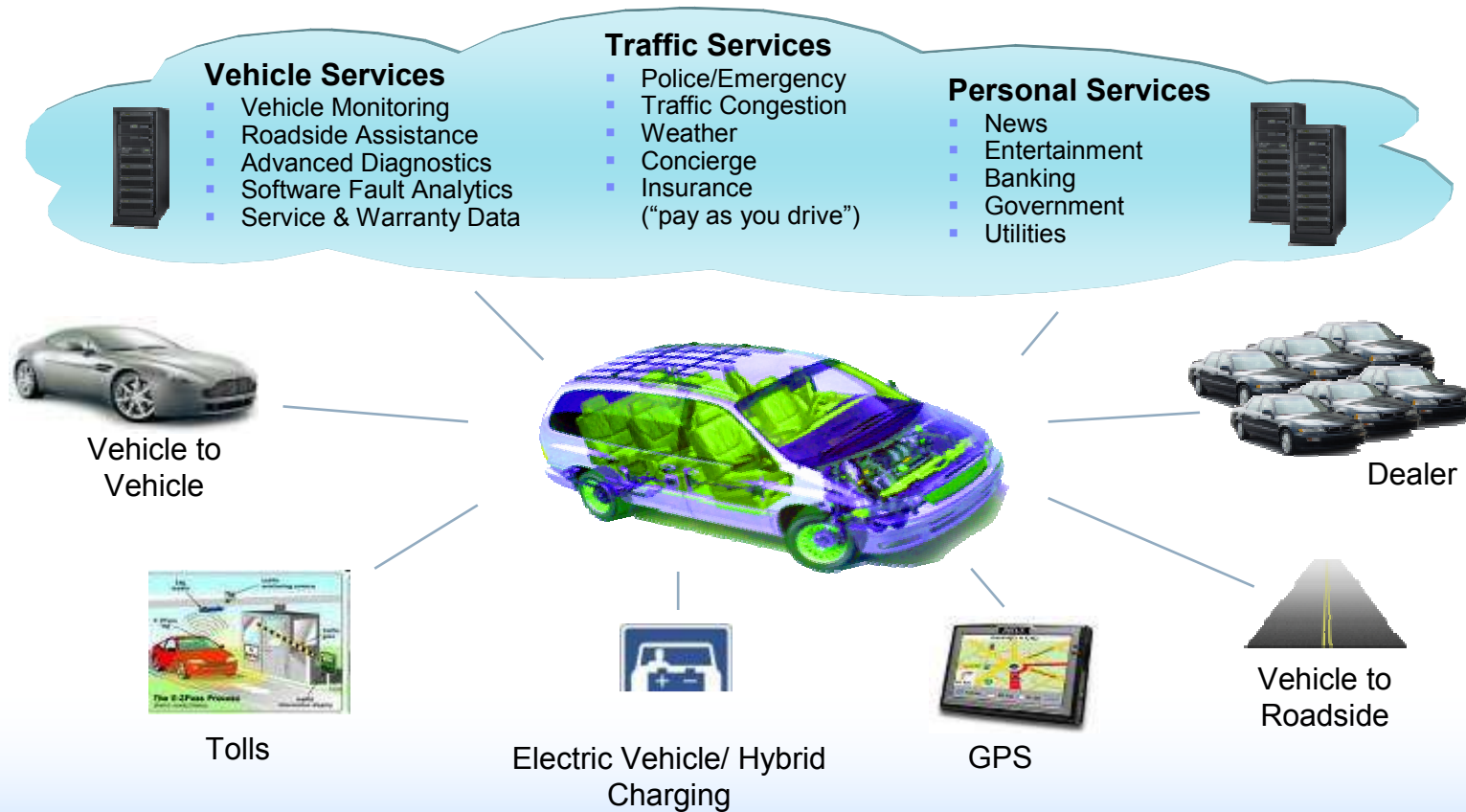**Madrid, Spain May 27th, 2009**

Rational. software

Go to IBM

Agenda

- Trends in Product Development & Delivery

- Challenges in Product Development

- Best Practices for Successful Product Development

Incremental value is created by **global interconnection** across products, systems, applications and the Internet

Vehicle Services
- Vehicle Monitoring
- Roadside Assistance
- Advanced Diagnostics
- Software Fault Analytics
- Service & Warranty Data

Traffic Services
- Police/Emergency
- Traffic Congestion
- Weather
- Concierge
- Insurance
 ("pay as you drive")

Personal Services
- News
- Entertainment
- Banking
- Government
- Utilities

Vehicle to Vehicle

Dealer

Tolls

Electric Vehicle/ Hybrid Charging

GPS

Vehicle to Roadside

Changes are being driven across the entire supply chain - even to commodity parts that now require sophisticated software & electronics

What is the value of the System?
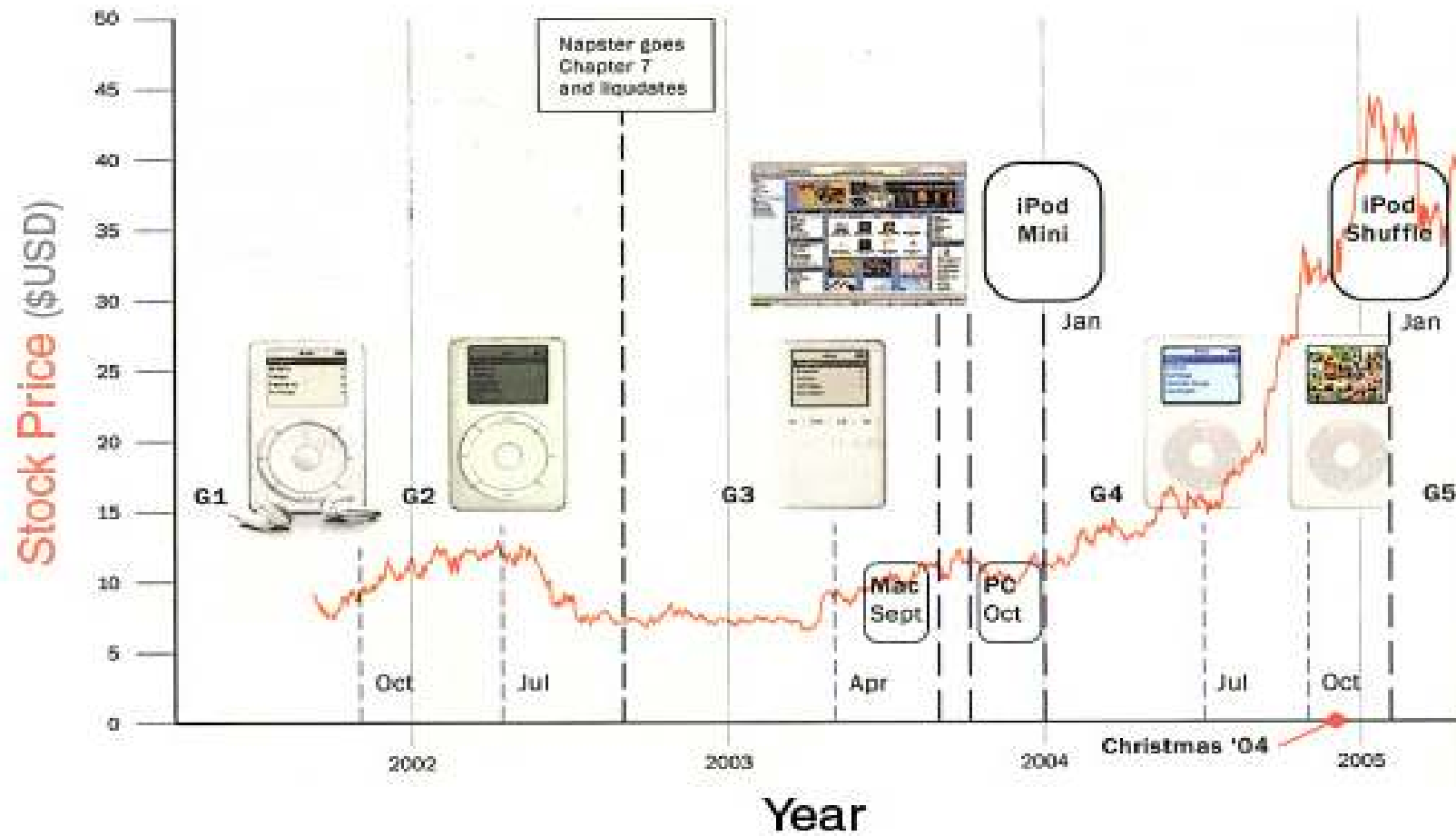
Transition from manufacturer to systems assembler
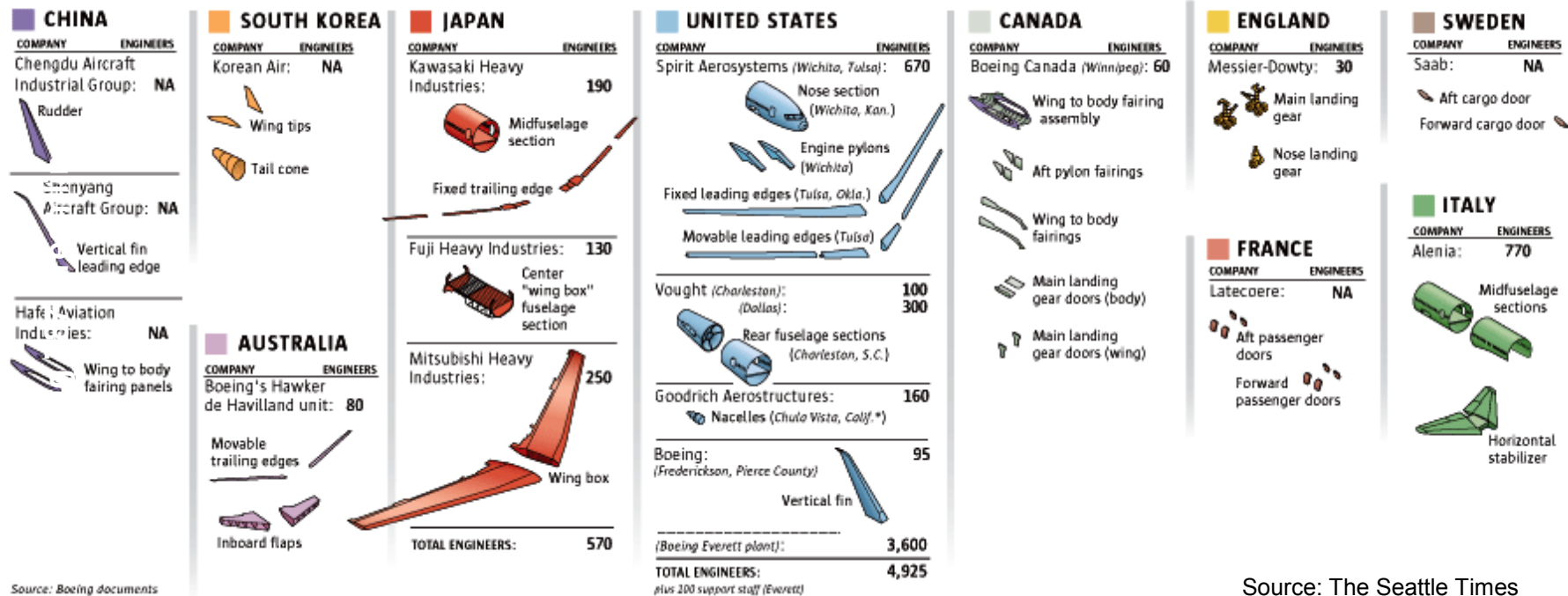## *Boeing Commercial Aircraft: 787 Development Program*

Number of parts: 6 million
Peak number of suppliers: 2,600

CAD models: 20,000
Design changes per year: 150,000

Who makes the parts and where the engineering jobs are:

Boeing 787: # of engineers are 2005 projections and may not include all engineering specialties. Production workers are not included.

CHINA
COMPANY	ENGINEERS
Chengdu Aircraft Industrial Group:	NA

Rudder

Shenyang Aircraft Group:	NA

Vertical fin leading edge

Hafei Aviation Industries:	NA

Wing to body fairing panels

Source: Boeing documents

SOUTH KOREA
COMPANY	ENGINEERS
Korean Air:	NA

Wing tips

Tail cone

AUSTRALIA
COMPANY	ENGINEERS
Boeing's Hawker de Havilland unit:	80

Movable trailing edges

Inboard flaps

JAPAN
COMPANY	ENGINEERS
Kawasaki Heavy Industries:	190

Midfuselage section

Fixed trailing edge

Fuji Heavy Industries:	130

Center "wing box" fuselage section

Mitsubishi Heavy Industries:	250

Wing box

TOTAL ENGINEERS:	570

UNITED STATES
COMPANY	ENGINEERS
Spirit Aerosystems (Wichita, Tulsa):	670

Nose section (Wichita, Kan.)

Engine pylons (Wichita)

Fixed leading edges (Tulsa, Okla.)

Movable leading edges (Tulsa)

Vought (Charleston):	100
(Dallas):	300

Rear fuselage sections (Charleston, S.C.)

Goodrich Aerostructures:	160

Nacelles (Chula Vista, Calif.*)

Boeing: (Frederickson, Pierce County)	95

Vertical fin

(Boeing Everett plant):	3,600
TOTAL ENGINEERS:	4,925

plus 200 support staff (Everett)

CANADA
COMPANY	ENGINEERS
Boeing Canada (Winnipeg):	60

Wing to body fairing assembly

Aft pylon fairings

Wing to body fairings

Main landing gear doors (body)

Main landing gear doors (wing)

ENGLAND
COMPANY	ENGINEERS
Messier-Dowty:	30

Main landing gear

Nose landing gear

FRANCE
COMPANY	ENGINEERS
Latecoere:	NA

Aft passenger doors

Forward passenger doors

SWEDEN
COMPANY	ENGINEERS
Saab:	NA

Aft cargo door

Forward cargo door

ITALY
COMPANY	ENGINEERS
Alenia:	770

Midfuselage sections

Horizontal stabilizer

Source: The Seattle Times

Agenda

- Trends in Product Development & Delivery

- **Product Development Challenges**

- Best Practices for Successful Product Development

Systems Engineering

The New York Times

Top Engineers Shun Military; Concern Grows

By PHILIP TAUBMAN
Published: June 25, 2008

 … the percentage increase in projected cost overruns for 95 major weapons programs in 2007 … A $295 billion overrun projected for 2007 was 26 percent above initial estimates …

… the central problem is a breakdown in the most basic element of any big military project: accurately assessing at the outset whether the technological goals are attainable and affordable, then managing the engineering to ensure that hardware and software are properly designed, tested and integrated.

The technical term for the discipline is *systems engineering*. Without it, projects can turn into chaotic, costly failures.

CNN Politics.com

Military procurement reform sweeps through Senate
May 7, 2009

….. "This really a situation of near-crisis proportions," McCain said. "We just cannot have the kinds of cost overruns that are associated with literally -- with one or two exceptions -- literally every new weapon system that the Department of Defense acquires."……

….." the Pentagon relies on unreasonable cost estimates, has unrealistic expectations for performance and "immature" technologies, and changes requirements and production levels at great cost. **…"**

What's Behind These Failures and Increasing Costs?

Business View

Product missed customer needs	46%
Late to market/missed demand	33%
Poor commercialization / promotion	26%
Product quality	24%
Pricing	23%
No clear product differentiation	19%

The CIO's Guide to the PERFECT Launch: Translating Innovation to Business Benefit, AMR Research, 2005

Engineering Opportunity

Improve communication and collaboration across disciplines	**71%**
Increase visibility into status of requirements	**49%**
Increase ability to predict system behavior prior to testing	**46%**
Implement or alter new product development processes for a multi-disciplinary approach	**43%**
Increase real time visibility of product Bill of Materials (BOM) throughout the development process	39%

Aberdeen Group, System Design: New Product Development for Mechatronics, Michelle Boucher, David Houlihan, January, 2008

Cost of Quality

	Requirements	Analysis/Design	Development	Deployment
Present	3%	27%	55%	15%

Systems Engineering

Leading Practice	20%	13%	22%	5%

30-50% Time saving

University of West Virginia and The United States Air Force Academy 2001
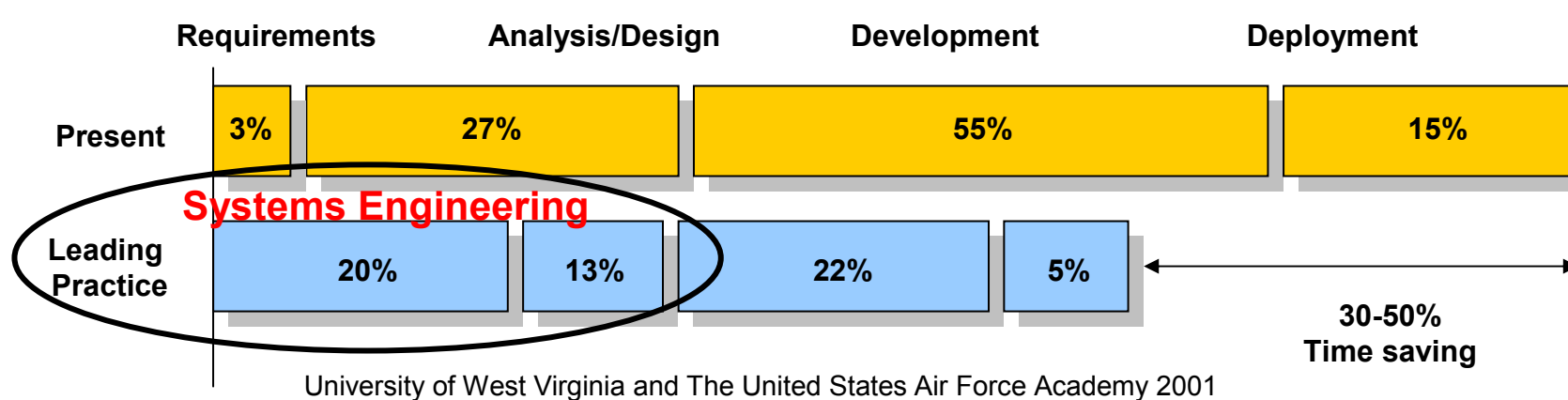
- The gap in benefits between the highest and lowest IT spenders for computing power was only 4% without a good management system.

- With a good governance system
  - Productivity improvement is 25%
  - Capital improvement is 70%

Source: Stephen J. Dorgan and John J. Dowdy - The McKinsey Quarterly, 2004 Number 4

Cost of Poor Quality
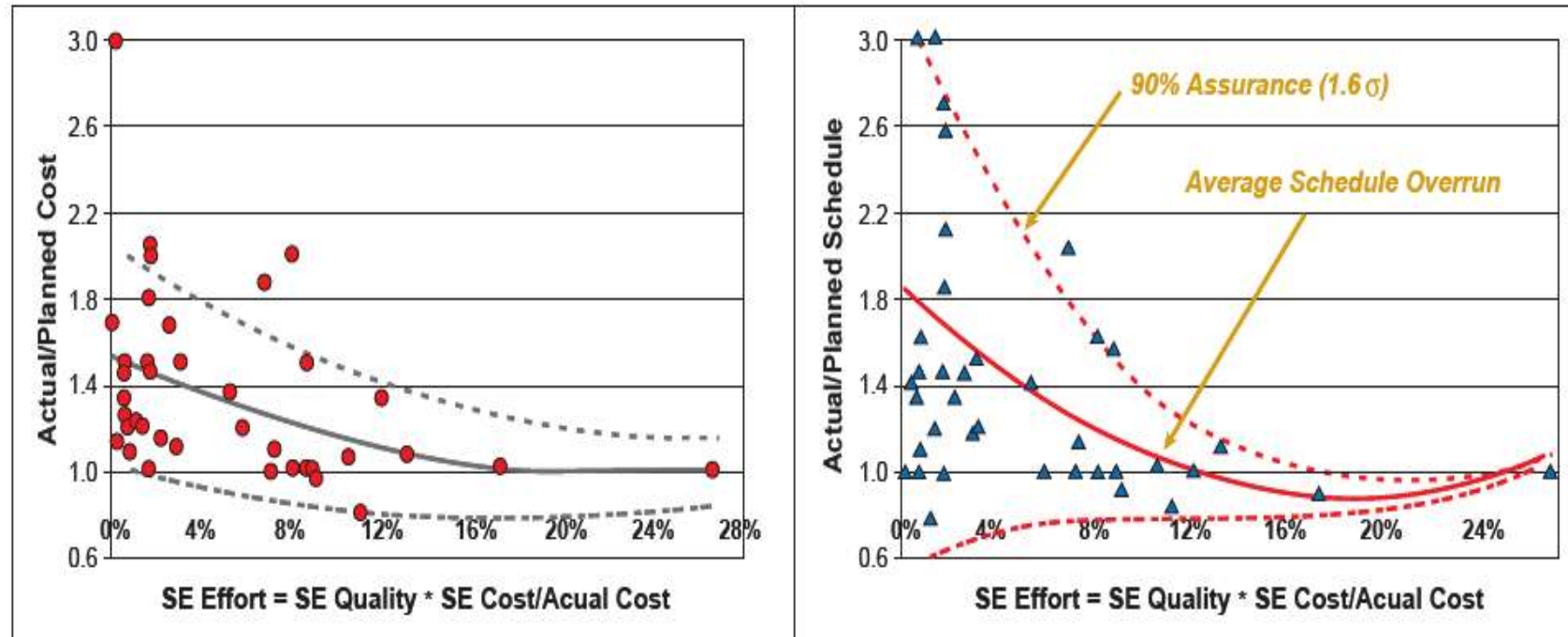
Figure 2-5 Cost and schedule overruns correlated
with systems engineering effort[12]

Historical Areas of Risk to Program/Product Success…..
In aerospace and defense, software content has increased 10X

**Relative Distribution
of Software/Hardware Costs**
(Percent of total cost)

Hardware

Development

Software

Maintenance

100

60

20

1955   1970   1985
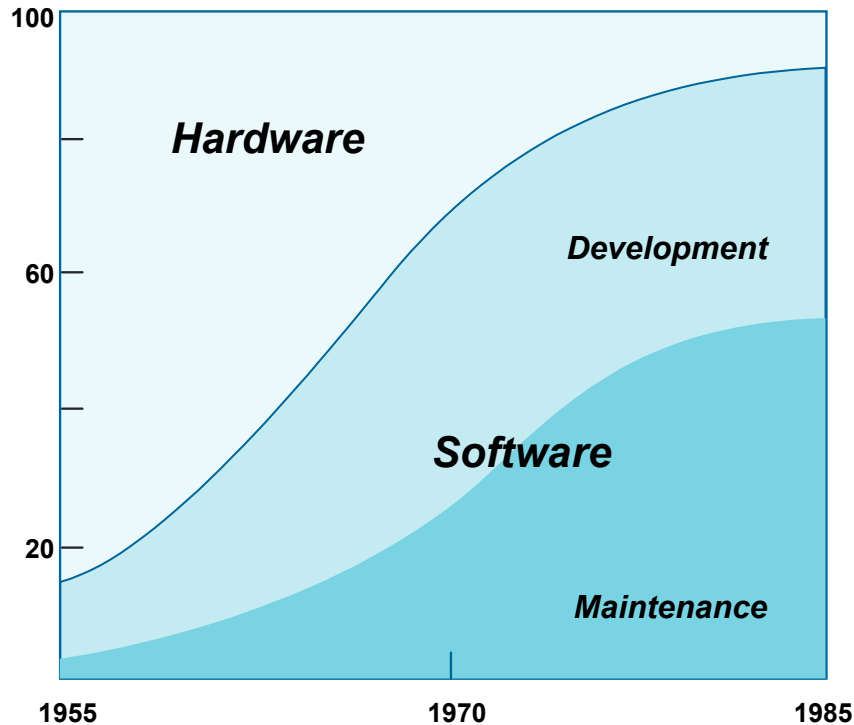
Source: Software Engineering, IEEE Transactions on
Computers December 1976

Platform	Year	Percent of Specification Requirements Requiring Software Control
F-4	1960	8%
A-7	1964	10%
F-111	1970	20%
F-15	1975	35%
F-16	1982	45%
B-2	1990	65%
F-22	2000	80%

Source: The Australian Software Acquisition Management Course,
Defense Systems Management College, March 2000

Agenda

- Trends in Product Development & Delivery

- **Best Practices for Successful Product Development**
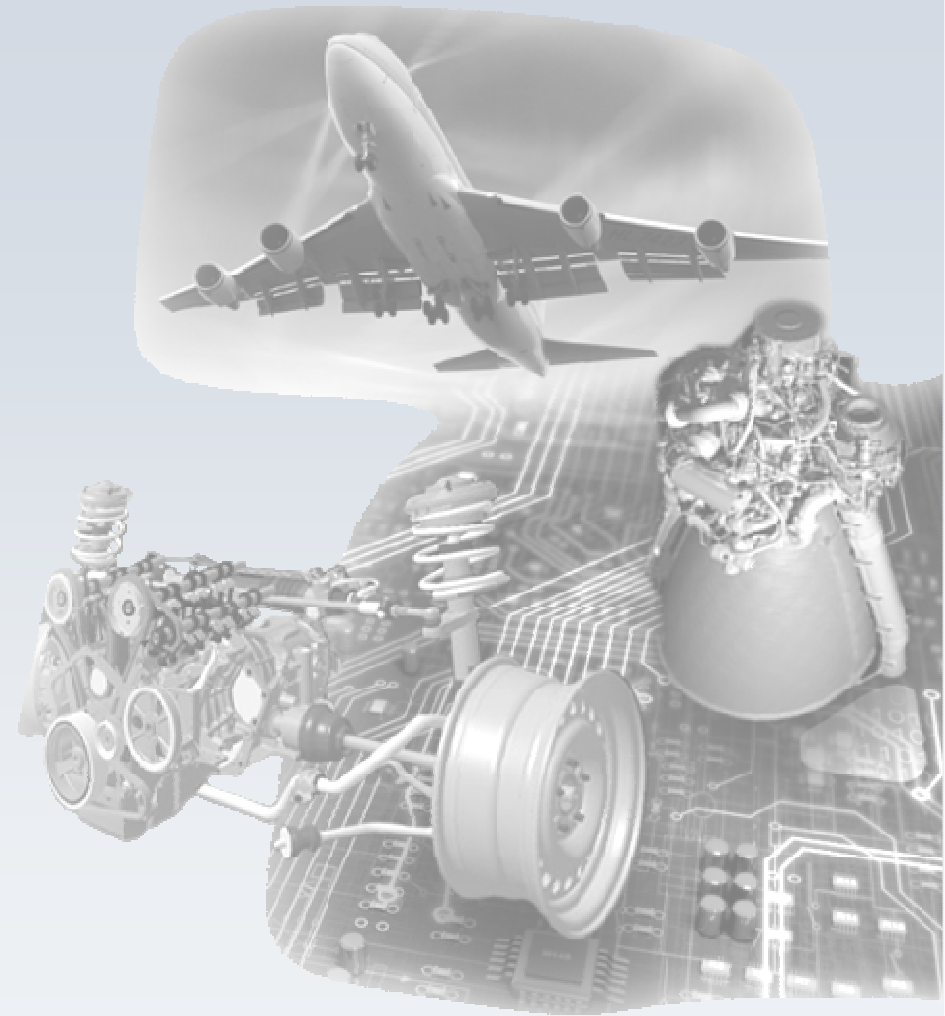
Best-in-class product companies are those that build a strong competency in software delivery

Best-in-class produce results:

- **19%** more likely to meet revenue targets than the industry average

- **4.4x** more embedded software than competitors

- **50%** fewer defects in embedded software

- **25%** decrease in product development time

Contributing practices:

- **Manage Requirements:** 2x more likely to manage requirements throughout entire lifecycle

- **Manage Change:** 2x more likely to notify other disciplines of change

- **Able to Predict Systems Behavior:** 7x more likely to digitally validate systems behavior prior with simulation across Software, Electrical & Mechanical components

Source: *"Embedded Systems Development", Aberdeen Group, March 2009*

Requirements Engineering

Model Driven Engineering

Requirements Engineering

Best Practices

Software Delivery
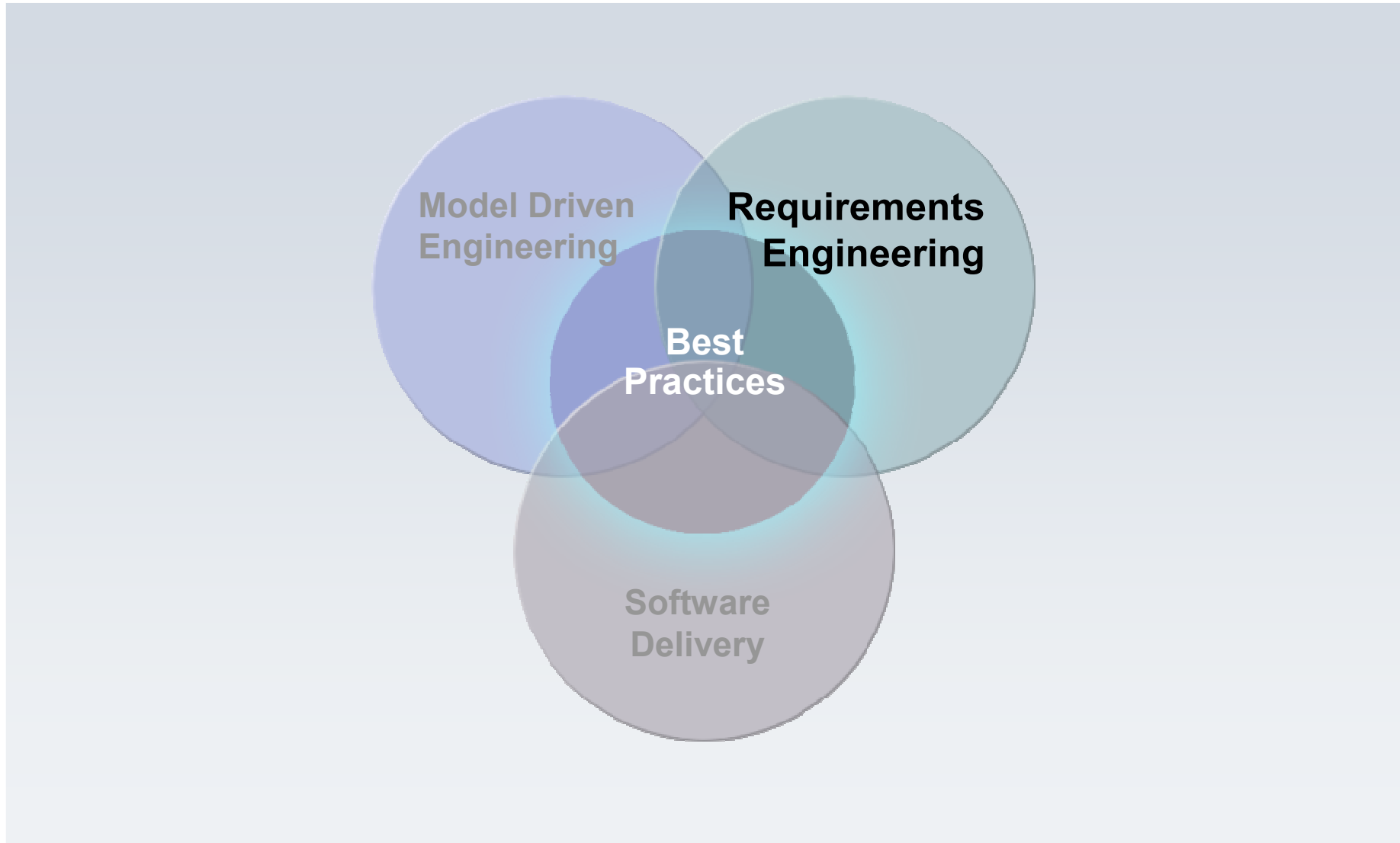
Requirements Engineering Best Practices

Best in class companies…

- Engineer requirements:
  - From the beginning of the product and system lifecycle
  - Through every phase of development
  - Across all disciplines of mechanical, electronic and softw

- Ensure traceability across all levels of requirements

- Mature from an isolated to a collaborative environment

- Invest the same focus and rigor on engineering requirements as in managing mechanical Bill of Materials

- Integrate Requirements Engineering closely with Change, Product and Portfolio Management, and Quality Assurance

IBM Requirements Engineering Solution

Capture • Trade-off Analysis • Validation • Change Management • Traceability • Impact Analysis • Reporting & Metrics • Monitoring

Business Analysis	Systems/Product Analysis & Implementation
Product Management	

Ideas	Analysis	Implementation	Test & Maintenance

Requirements Definition   *Requirements Management*
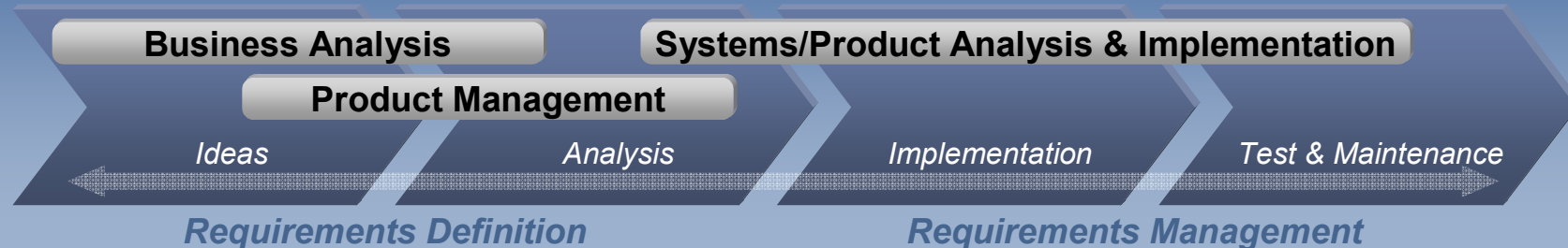
Requirements begin with Product Management
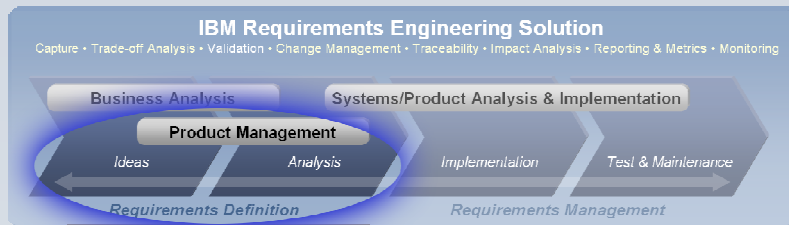
- **Automated idea capture**
  - ▶ Capture ideas and product requirements from internal stakeholders and customers

- **Value-based feature selection**
  - ▶ Understand customer value when making decisions about products

- **Market-based analysis**
  - ▶ Analyze product decisions in relation to your goals, objectives, and competitive position
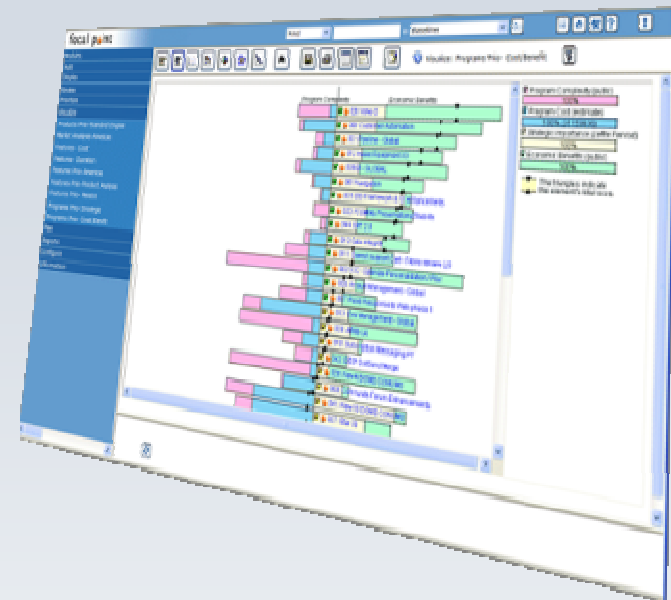
- *Are we doing the **RIGHT** products?*
- *Are we spending in the **RIGHT** areas?*
- *Do we have the **RIGHT** resources?*

Manage Requirements Across the Lifecycle and Across Disciplines
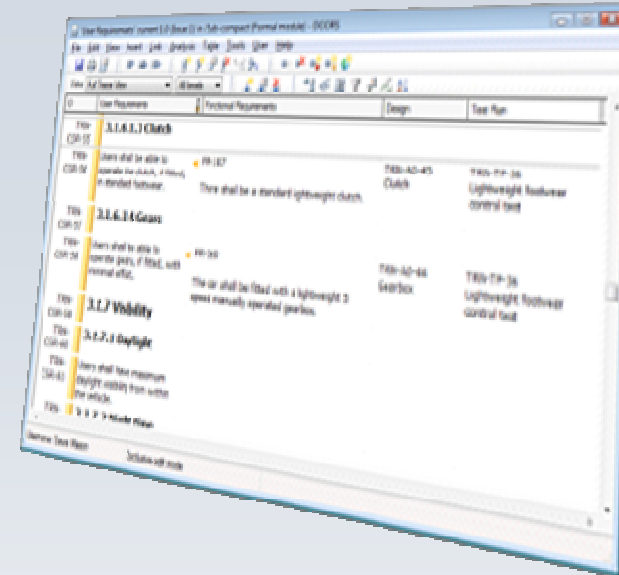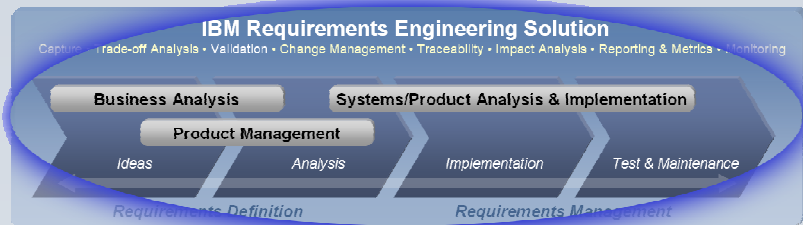
- **Provides end-to-end visibility of requirements**

- **Comprehensive support for recording, structuring, managing, and analyzing requirements and their traceability**

- **Requirements are persistent at all levels of decomposition**

- *Can manage requirements across multiple engineering disciplines - Software, Electronic & Mechanical*

- *Scalable for large projects with many users*

Link Testing to Requirements as well as to Design.

IBM Requirements Engineering Solution

Capture • Trade-off Analysis • Validation • Change Management • Traceability • Impact Analysis • Reporting & Metrics • Monitoring

Business Analysis

Systems/Product Analysis & Implementation

Product Management

Ideas   *Analysis*   *Implementation*   *Test & Maintenance*

Requirements Definition   *Requirements Management*
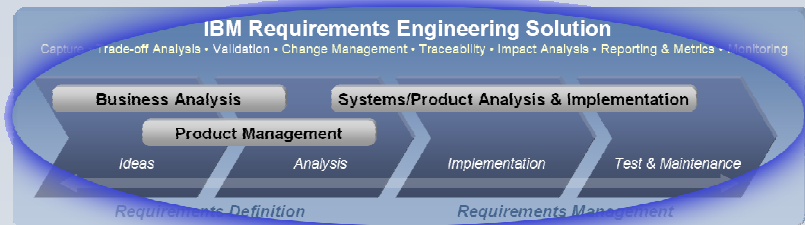
▶ Incorporates requirements in test- and quality plans

▶ Relates test cases, test suites, test execution and test result with requirements

▶ Provides traceability between and within hierarchies of requirements and tests

Quality Management

▪ *Shows which requirements are tested and which not*

Requirements Engineering Process & Tool Enforcement ROI – quantifiable savings

- **Advanced Weapon Control System – incremental development lifecycle**

Improvements	Before	After
Requirements Volatility - @ Preliminary Design Review @ Final Design Review	72% 33%	48% 17%
Requirements Changes Implemented - Changes accepted Changes rejected	98% 2%	16% 84%
Testing Time - Integration System User Acceptance	9 weeks 13 weeks 22 weeks	4 weeks 6 weeks 10 weeks
Defects found after production	728	165
Software Requirements Specification Production Time	10 days	2 days

Model Driven Systems Engineering

Model Driven Engineering

Requirements Engineering

Best Practices

Software Delivery
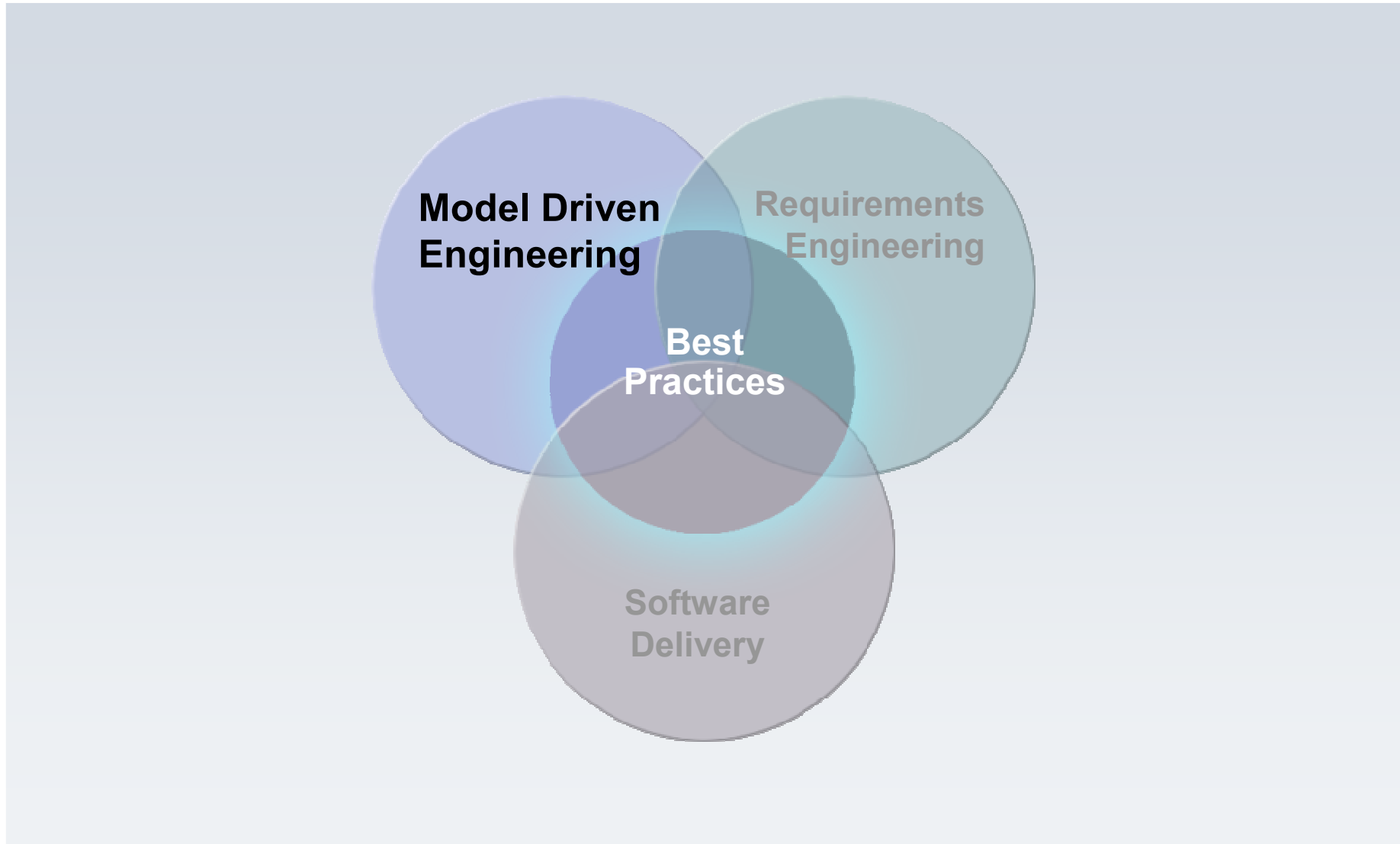
Development time should be spent developing
80% of development costs are spent identifying and fixing defects

Induced Errors

Costs of Design Change

Time

Requirements

Design

Implementation

Test

$80/defect

$240/defect

$960/defect

$7,600/defect

During the requirements phase

During the design & construction phase

During the QA/Testing phase

Once released as a product

Source: 2008 GBS Industry standard study
Defect cost derived in assuming it takes 8 hrs to find, fix and repair a defect when found in code and unit test.
Defect FFR cost for other phases calculated by using the multiplier on a blended rate of $80/hr

The MDD enablers for developing systems and software
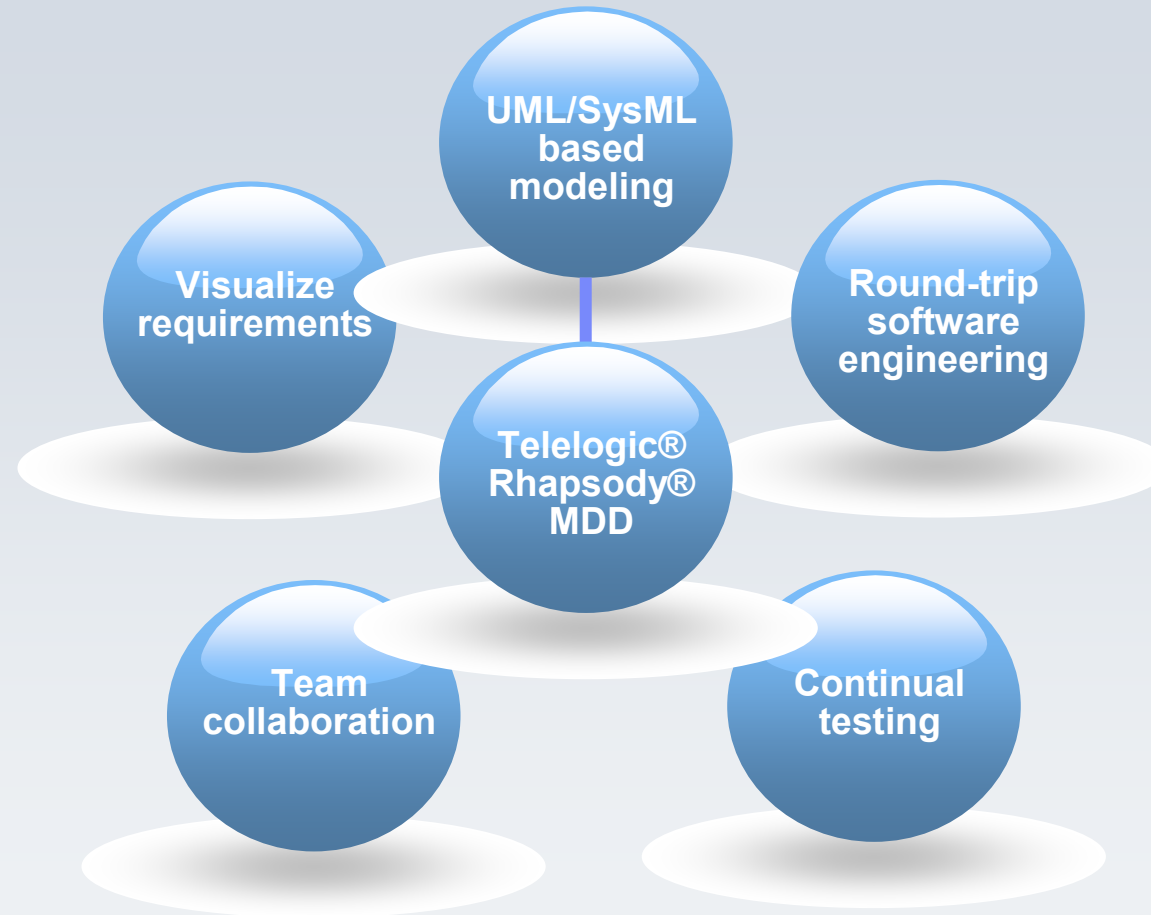
Visual requirements capture

Visualize
requirements

- Extend requirements engineering to development
  - ▸ Iterate requirements into executable systems model for analysis and validation

- Confirm project requirements
  - ▸ Animate and simulate use case functionality and behavior
  - ▸ Ensure that the design fulfills the requirements; ensure regulatory compliance
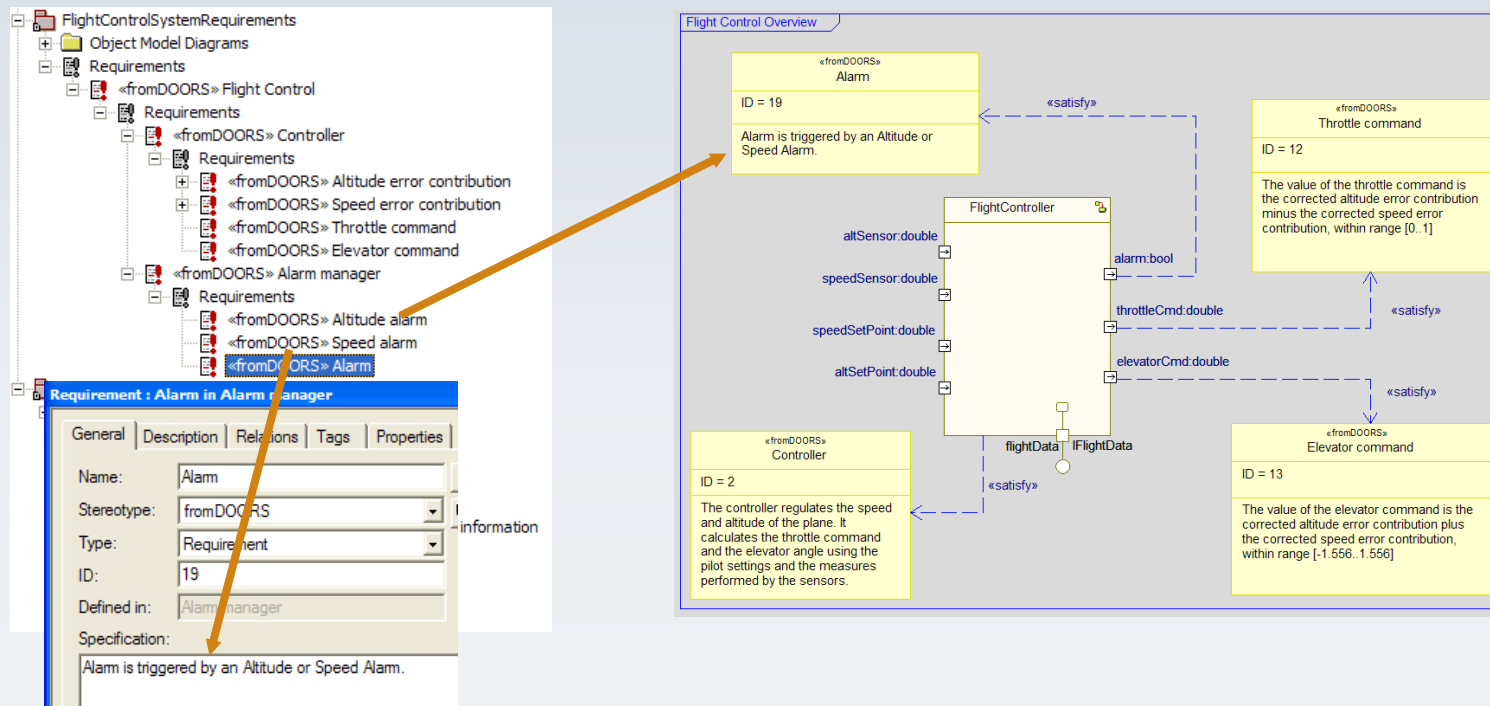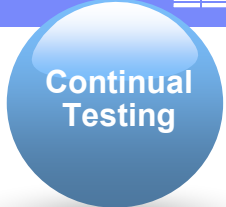
Simulation, execution & animation

Continual Testing

- **Promote agile development by validating requirements early**

- **Perform design level debugging**

- **Identify and address errors early when they are less costly to fix**

- **Visually communicate intended behavior to customer**
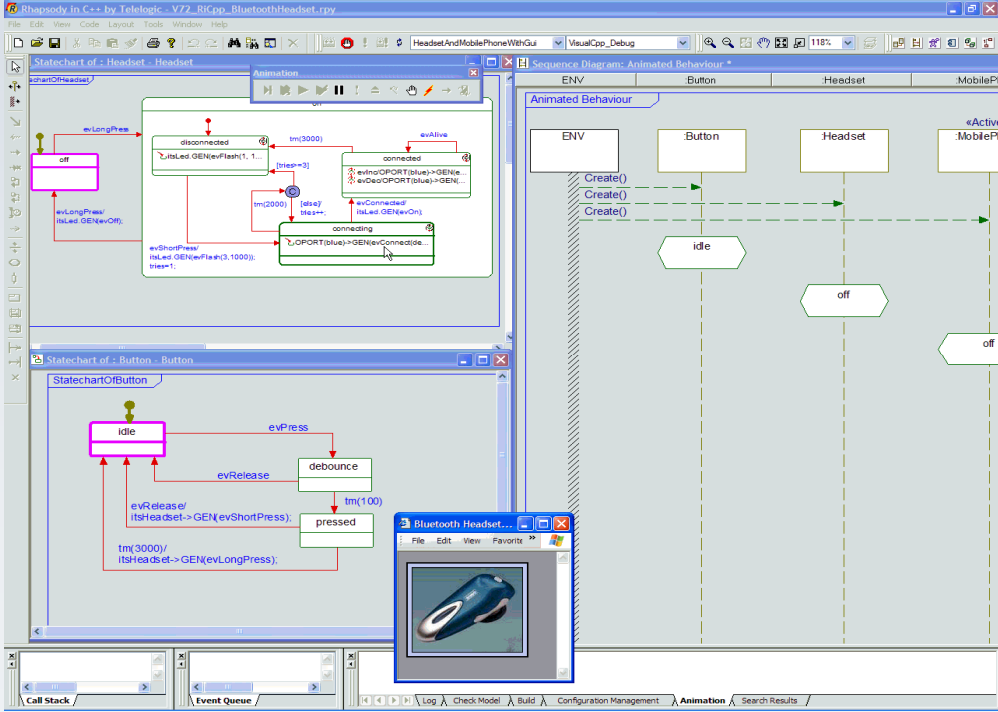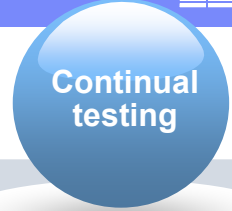
Model-driven testing

Continual testing

- Bring the benefits of abstraction and automation to testing

- Reduce defects early in the process when they are less costly to fix

- Deliver products meeting customer expectations
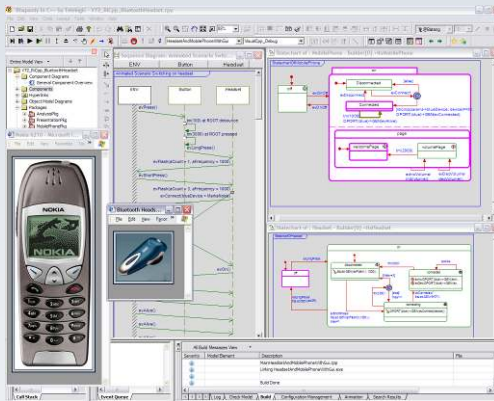
Simulation   **Requirements-based testing**   **Automated unit testing**

Sequence Diagrams
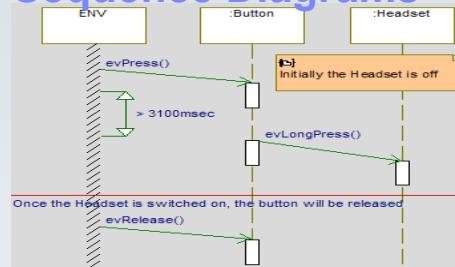
Finding & Correcting Errors

Host based   Target based

Automation, Traceability, Time....

Requirements

Modeling

Traceability

Documentation

System Behavior

Dynamic System Modeling

MATLAB / Simulink

Auto Generated Code for Embedded Systems
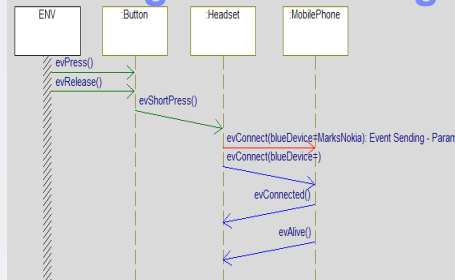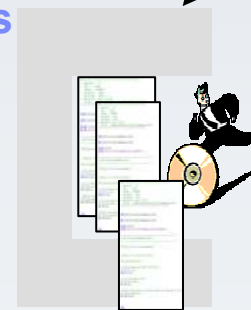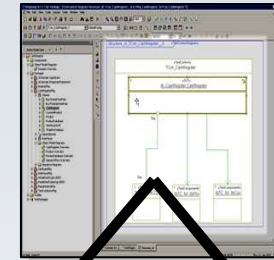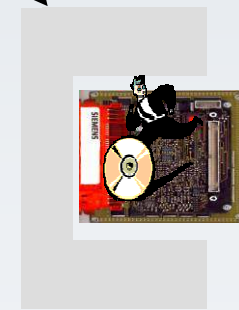
MDD automation enables real savings in simulation and test

- Comprehensive test process model
  - *7:1 return on system engineering investments*

- Integrated end-to-end technology
  - *Reduced manual test effort of 90+%*

- Industry-based test solutions
  - *10 - 20% savings for test case and script reuse*

- Advanced defect analysis and quality management processes
  - *Defect reduction up to 69%*

Standardized Quality Offerings

Client Results

- Quality improvements ranging between 30 - 70%

- Cycle time improvements of 20 - 50%

- Cost reductions of 25 - 60%

Test Centers of Excellence

Test consulting

Life cycle testing

Test automation

Reporting

Specialized testing

Management

Infrastructure testing
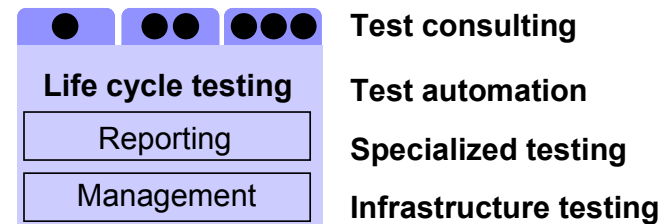
Source: GBS Test Practices Investor Board Briefing

Software Development for Systems

Systems Engineering

Requirements Engineering

Best Practices

Software Delivery
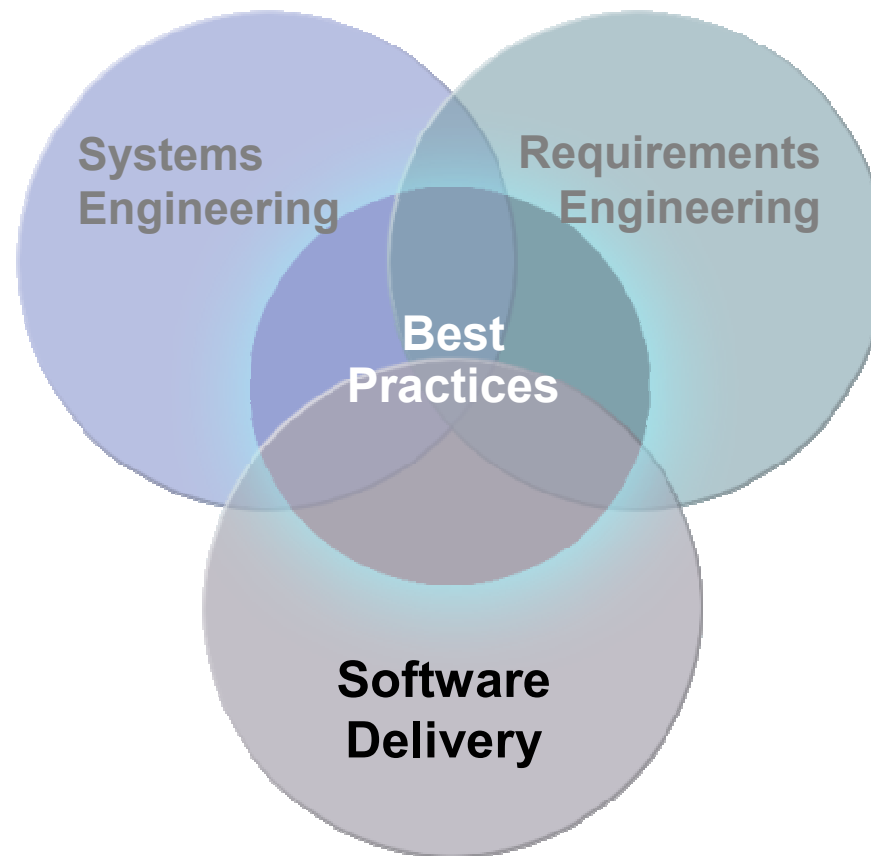
Agility @ scale with Rational Team Concert

1,362 total resources worldwide

Canada – 299 (22%)

Poland – 3 (<1%)

China – 78 (6%)

France – 30 (2%)

Israel – 29 (2%)

Japan – 4 (<1%)

United States – 639 (47%)
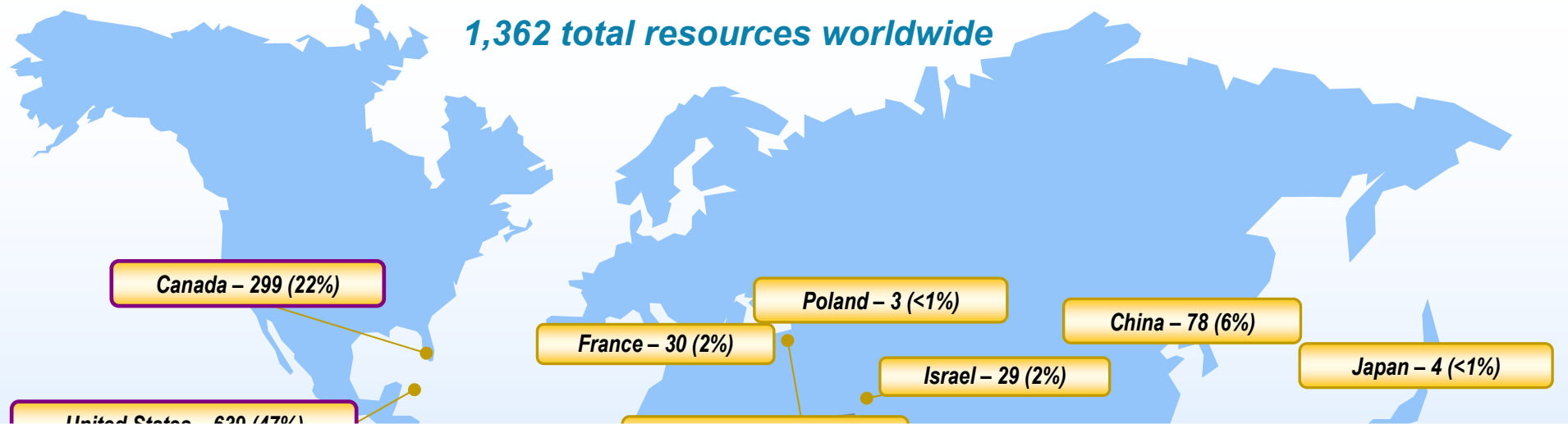
"We've really enjoyed the way Jazz deals with collaboration, keeping track of tasks and linking changes to actual tasks. This is light-years ahead of what we had before with CVS."

"Less emails. Less lost info. Most accurate vision of features and tasks. Very light weight for everyone. Supports sprints / iterations very nicely."

\- IBM internal feedback

Improve Quality by leveraging standardized Methods.

Method Library (2 Process) for Software & System Development
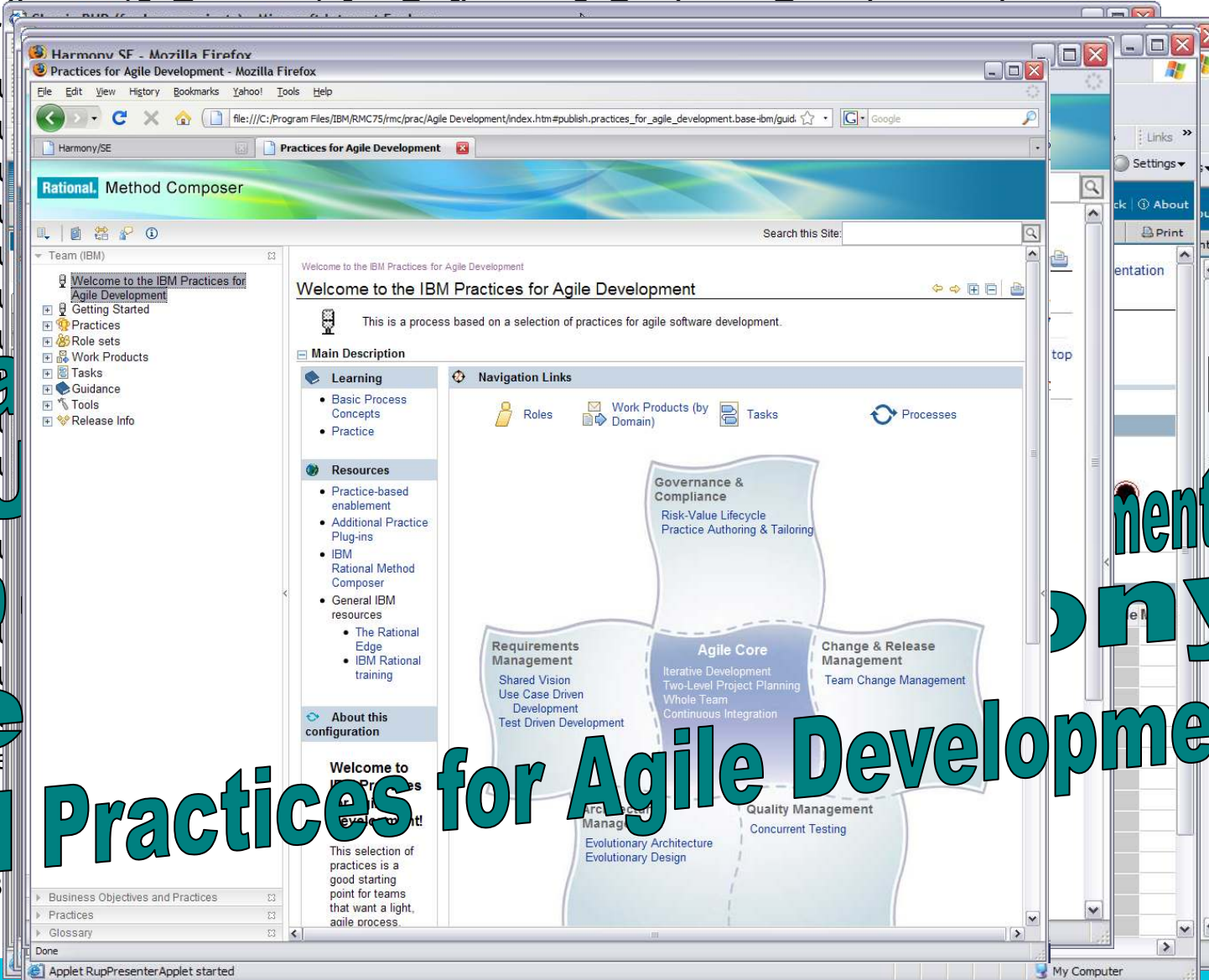
Link your Methods to a deployment platform: move from static to dynamic processes
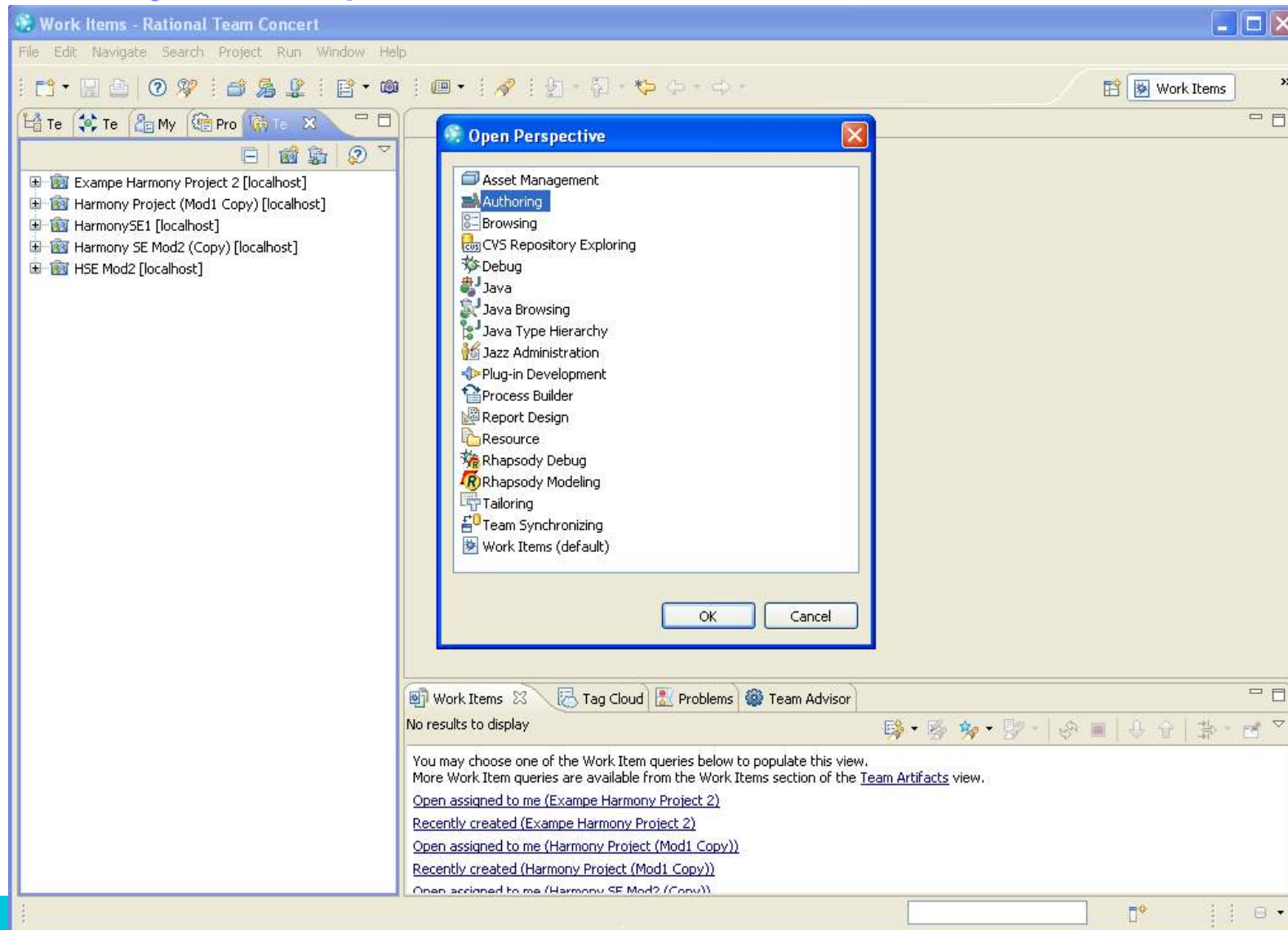
Once in RTC, Assign Work and track Team Execution

Automated Reporting and Metrics

Configuration Mgt allows Enterprise collaboration
Enable multiple teams to access methods and Systems Engineering artifacts.
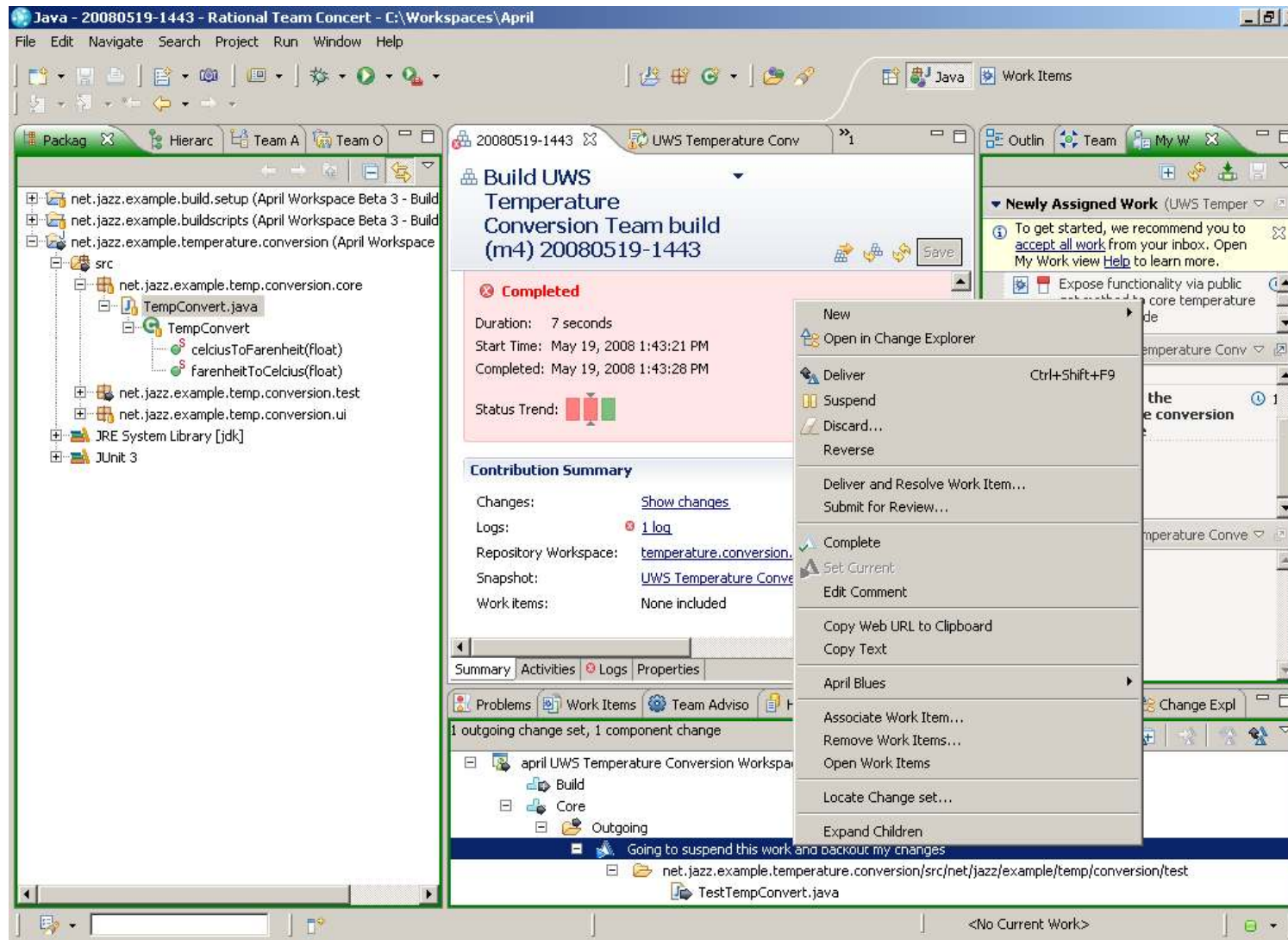
In-context collaboration

Collaborate in Context

Track processes, teams, projects, portfolios

Trending by Project or by Individual Team

Team Member Details

Current Milestone status

Customers are seeing real results

Volkswagen AG	▪ **20% productivity increase**
TTI Telecom	▪ **Improved time-to-market by 50%** ▪ **Estimated savings of $2 million per year**
Unisys	▪ **Accelerated client success:** ▪ **25-40% improvement in cycle time** ▪ **75-100% improvement in productivity** ▪ **25-60% cost savings**
Ericsson	▪ **80% fewer bugs; 100% productivity increase**
Lockheed Martin Canada	▪ **$409K Net Quantifiable Benefit; 222% ROI**
Credence Systems Corporation	▪ **1200% increase in developer productivity** ▪ **90% reduction in bug backlog**
Covarity	▪ **Reduced development lifecycle by 25%**
Information Builders, Inc.	▪ **96% productivity increase**