Rapid Application Development (RAD) Strategies

Barry Boehm, USC

Intel Presentation

June 4, 1998

email: boehm@sunset.usc.edu

http://sunset.usc.edu

Outline

- USC Center for Software Engineering
- RAD Context
- RAD Opportunity Tree and Strategies
- RAD Issues
  - Biggest Opportunity Areas
- Some Successful RAD Experiences

USC-CSE Objectives and Context

- **Create software engineering technology addressing key future needs**
  - **Understand current needs**
  - **Infer future needs**
- **Transition technology into practice**
  - **Affiliates program, research contracts**
- **Grow future software engineering leaders**
  - **MSCS-SW Engr., Ph.D. programs**

USC-CSE Principals

- **Bob Balzer, ISI** — **Architecture, environments, processes**
- **Barry Boehm, CS** — **Processes, environments, architectures, economics**
- **Ellis Horowitz, CS** — **Environments, languages, open systems, object bases**
- **Lewis Johnson, ISI** — **Software understanding, hypermedia, KBSE**
- **Dan Port, CS** — **OO architecting and development, formal methods**
- **Eb Rechtin, EE** — **Systems architecting principles and processes**
- **Walt Scacchi, BA** — **KBSE, processes, environments, management**
- **Bert Steece, BA** — **Statistics, COCOMO II**
- **Dave Wile, ISI** — **Transformations, architectures, environments**

– Research Associates

- **Dr. Anne Curran, CS** — **Environments, operating systems**
- **Dr. Ray Madachy, Litton** — **Cost and process modeling**

USC-CSE Affiliates (26)

- **Commercial Industry (10)**
  - **Bellcore, EDS, Hughes, Lucent, Motorola, Network Programs, Rational, Sun, TI, Xerox**
- **Aerospace Industry (9)**
  - **Allied Signal, Boeing/Rockwell, GDE Systems, Litton, Lockheed Martin, Northrop Grumman, Raytheon/E-Systems, SAIC, TRW**
- **Government (4)**
  - **FAA, USAF Rome Lab, US Army Research Labs**
- **FFRDC's and Consortia (4)**
  - **IDA, MCC, SEI, SPC**

Membership Benefits

USC Visits to Affiliates

- Annual one-day lecture/visit by USC professor
- Long-term visit arrangements
- Student resumes for summer or full-time positions

Affiliate Visits to USC

- Annual Research Review
- Corporate campus visits
- Long-term visit arrangements

Workshops and Conferences at USC

- Focused Workshops and Knowledge Summaries
- Executive Software Seminars
- Annual Software Engineering Conference (joint with UC Irvine)

Tool and Knowledge Transfer

- Prototype tools for experimentation
- Technical reports
- Web site -- http://sunset.usc.edu
- Proceedings from conferences, workshops, and reviews

Influence on Research and Curriculum Directions

- Annual Research Reviews
- Technical Steering Committee

Opportunities to Pursue Joint Projects

- COCOMO 2.0 Project (also involving UC Irvine)

Ph.D. Students

WinWin

Ming June Lee (Ph.D., 1996) - Formal Modeling of the Win Win System
Yimin Bao (Ph.D., 1996) - Tool Integration Language and Support
Hoh In - Automated Conflict Detection and Option Generation
June Sup Lee - Architectures for Groupware Systems

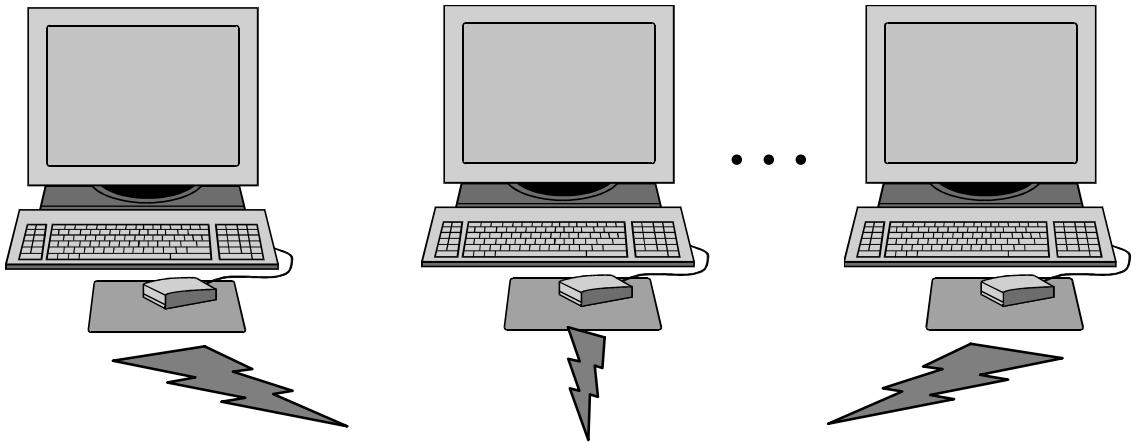
Architecture

Ahmed Abd-Allah (Ph.D., 1996) - Heterogeneous Architecture Style Composition
Cristina Gacek - Architecture Style Composition Framework
Alex Egyed - Integration of Multiple Architecture Views
Simei Du - Architecture Attribute Analysis Tool Integration

Cost and Process Modeling

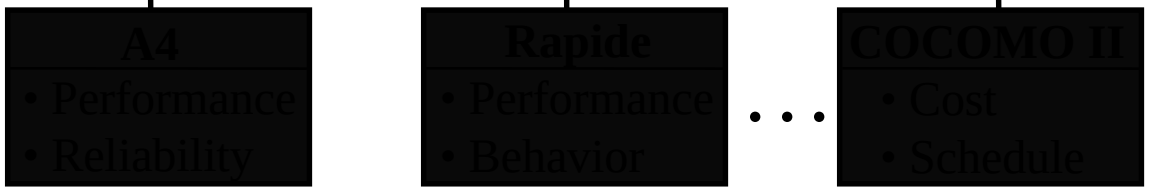
Ray Madachy (Ph.D., 1995 - ISE) - System Dynamics Model of Inspection Effects
Brad Clark (Ph.D., 1997) - Effect of Process Maturity on Software Cost
Sunita Devnani-Chulani - Software Cost-Schedule-Quality Tradeoff Model
Chris Abts - COTS Integration Cost Model
Allen Nikora - Software Reliability Modeling

Rapid Requirements Negotiation



Distributed Stakeholders

Rapid Requirements Negotiation Tool



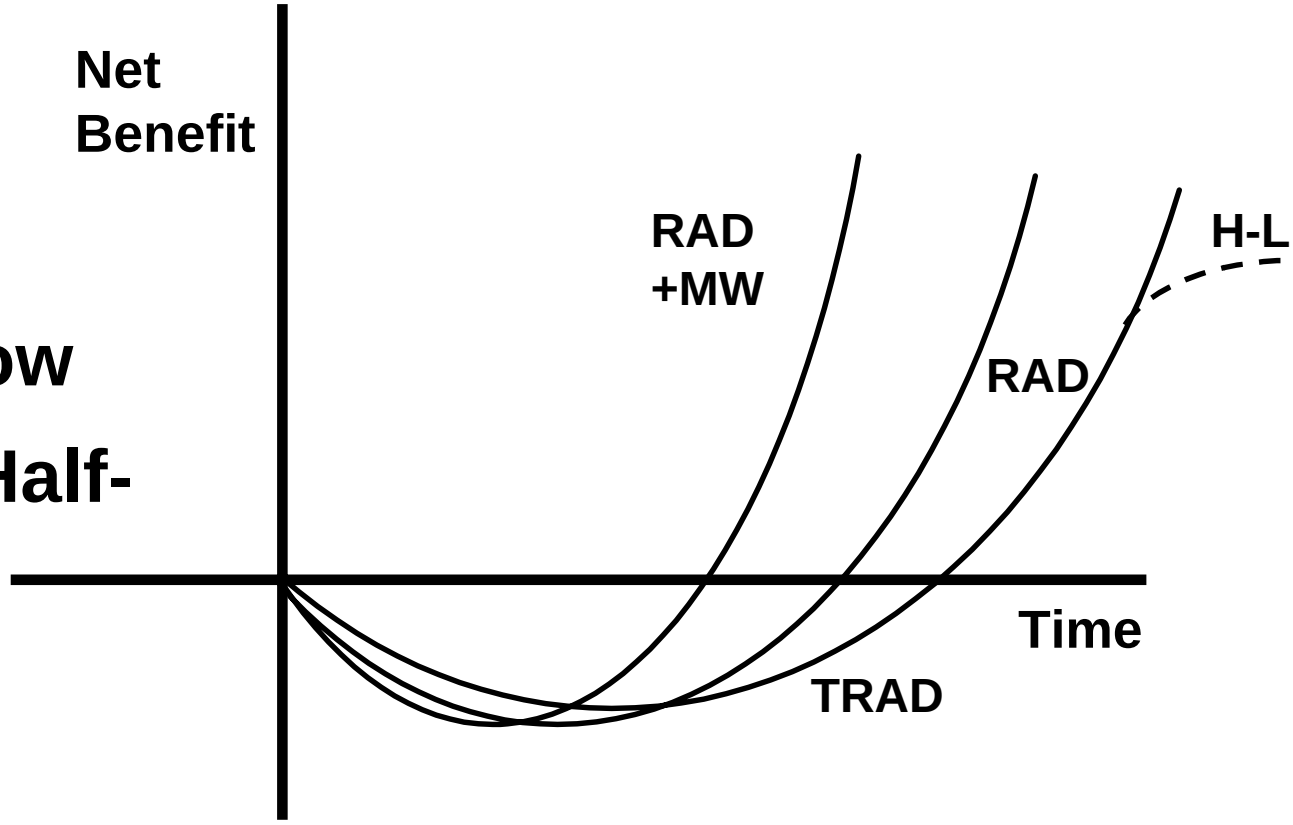
Tradeoff Analysis Tools

Outline

- USC Center for Software Engineering
- RAD Context
- RAD Opportunity Tree and Strategies
- RAD Issues
  - Biggest Opportunity Areas
- Some Successful RAD Experiences

RAD Motivation

- Earlier ROI
- Market Window
- Technology Half-Life

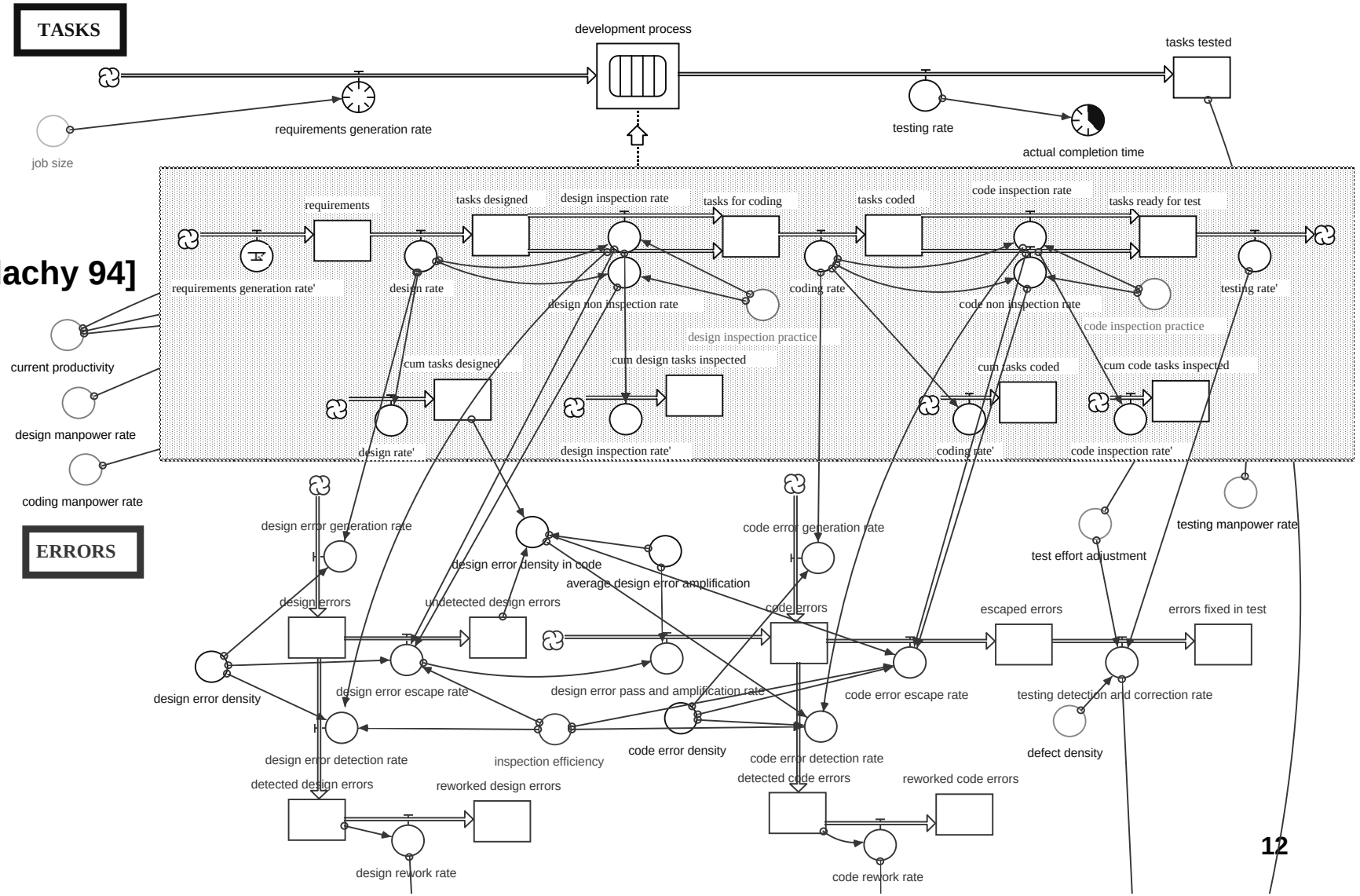


Complexity of RAD Improvements

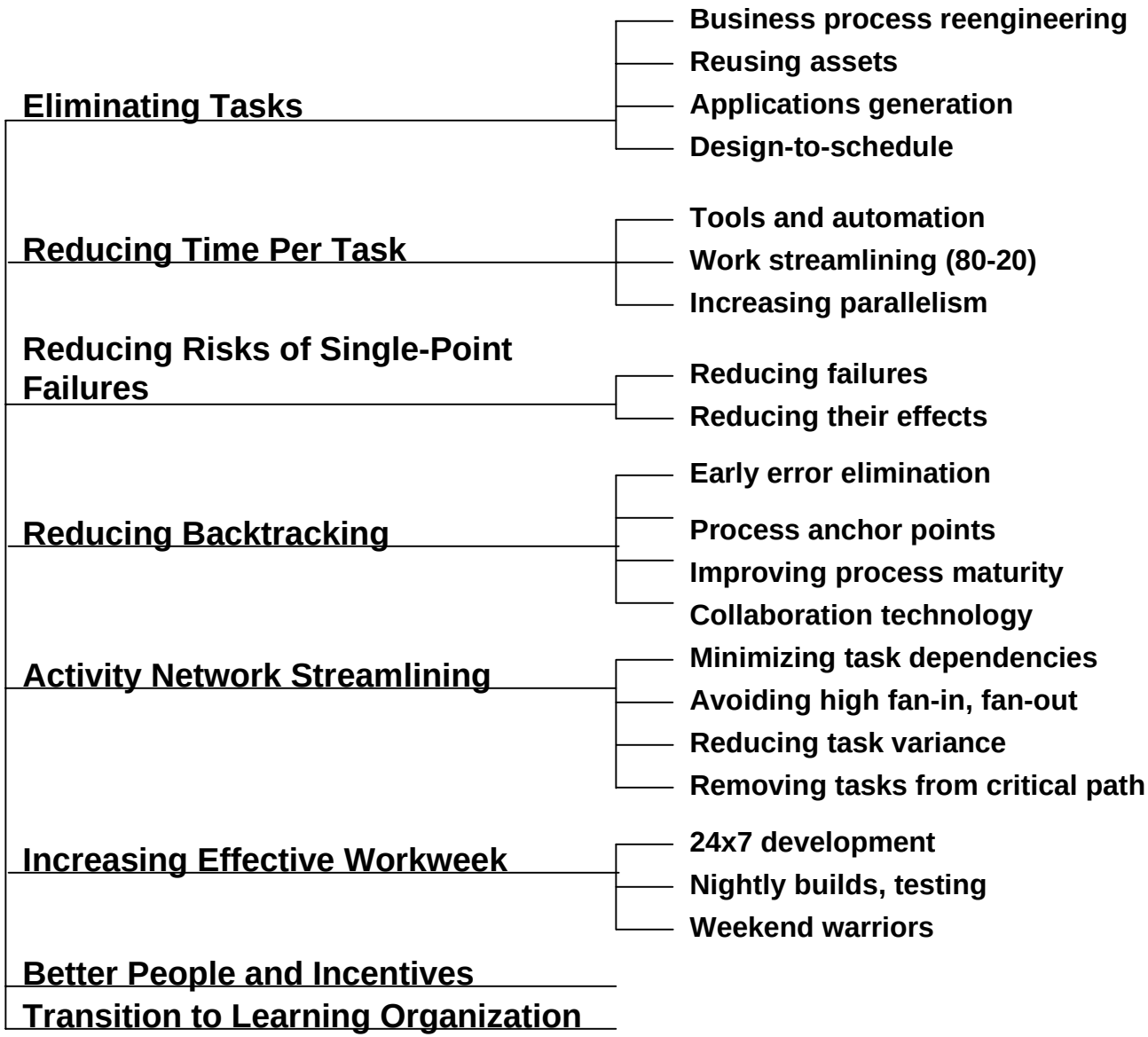
- **Cost: task savings basically map 1:1 into project savings**
- **Schedule: task savings map 1:1 into project savings only while task is on critical path**
  - **Complicating factors: scale, dynamism, interdependent tasks**
  - **System dynamics an attractive analysis approach**

Example System Dynamics Analysis (Madachy)

from [Madachy 94]



RAD Opportunity Tree

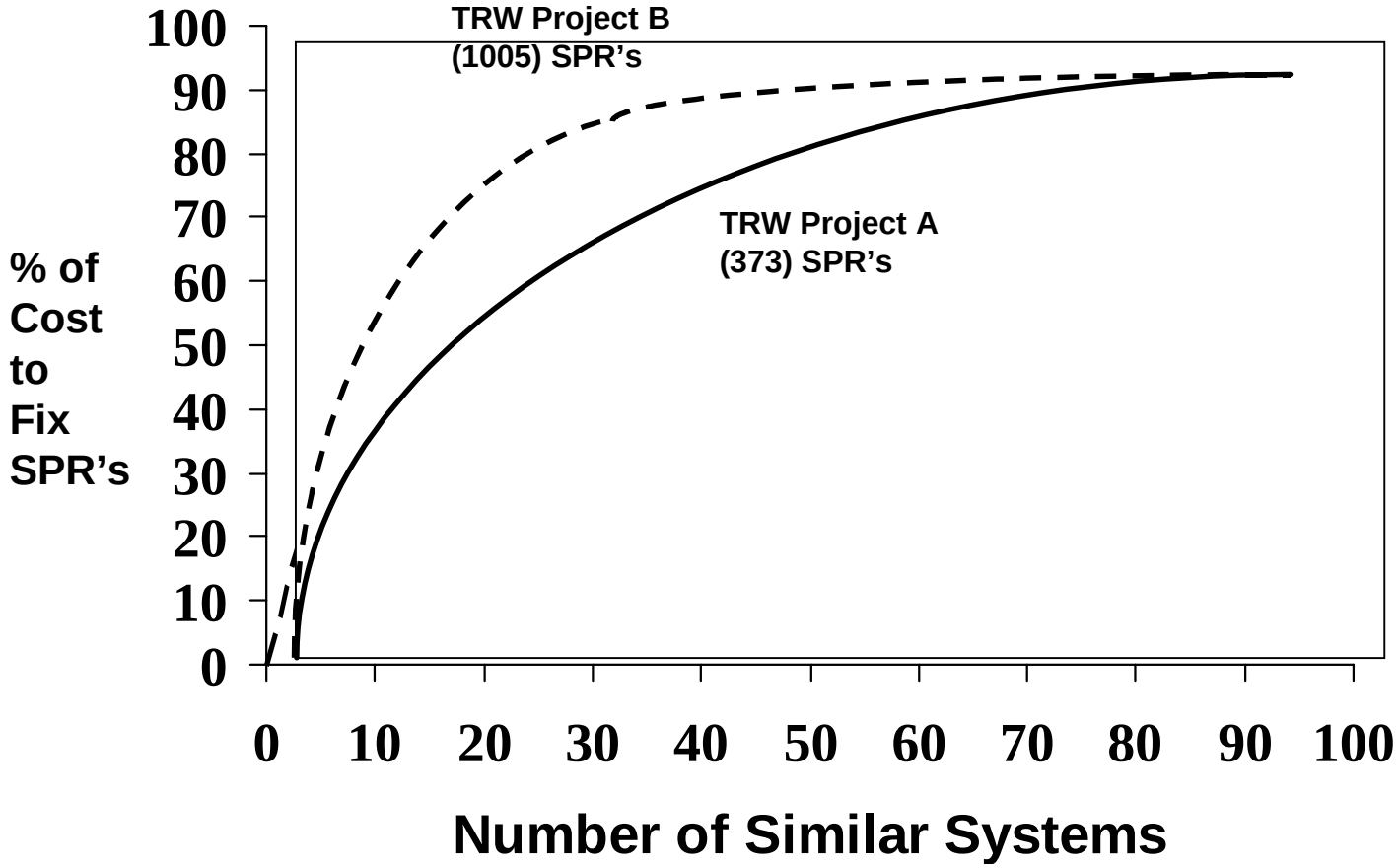


Design-to-Schedule

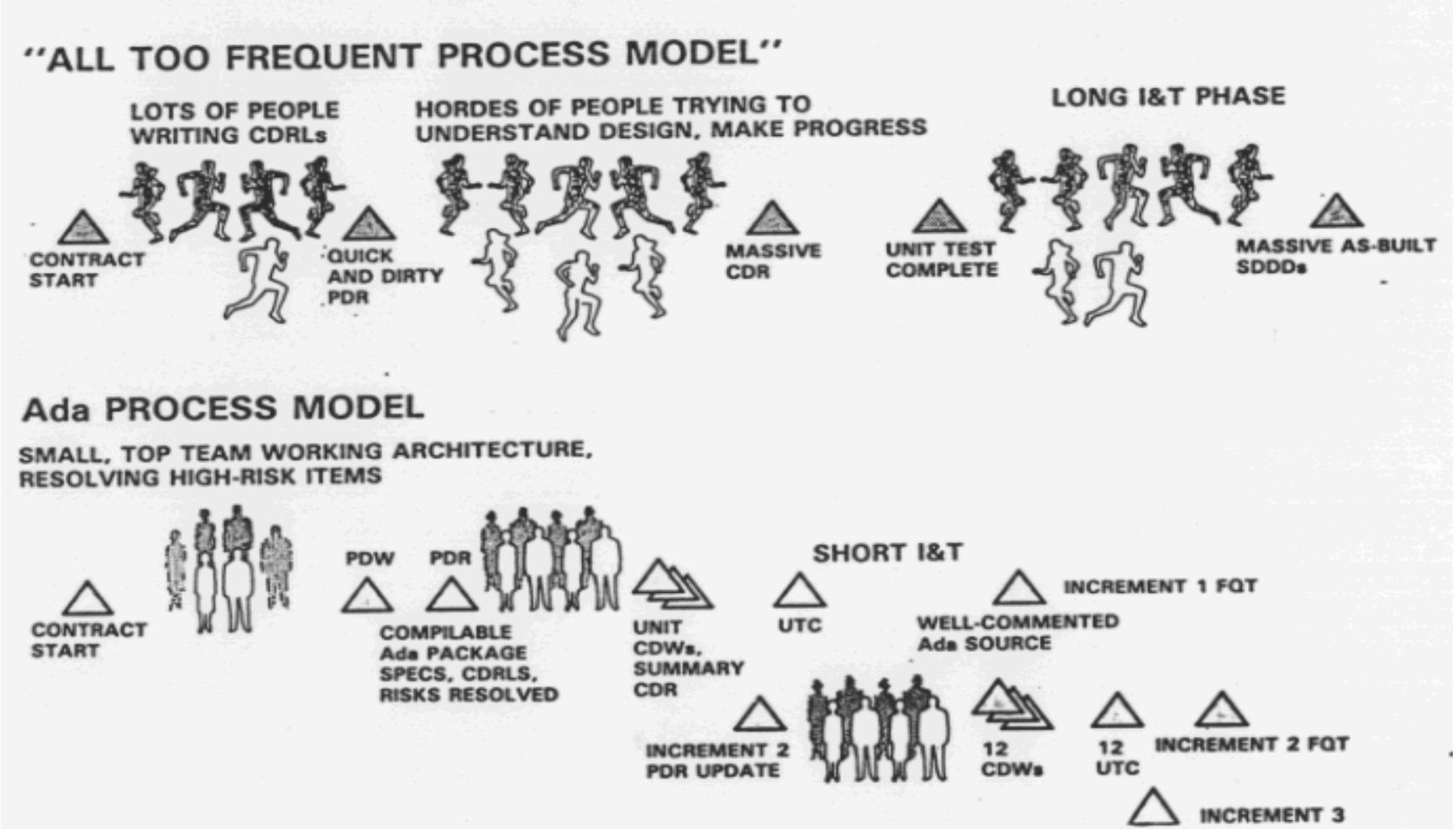
"If schedule is your independent variable, then just modulate your functionality to meet schedule."

- **Critical success factors:**
  - **Prioritized requirements**
  - **A reasonable "ballpark" schedule estimate**
  - **Software design for ease of contraction**
  - **Schedule tracking for midcourse corrections**

Pareto Analysis of Rework Costs



Ada Process Model for Large-System RAD



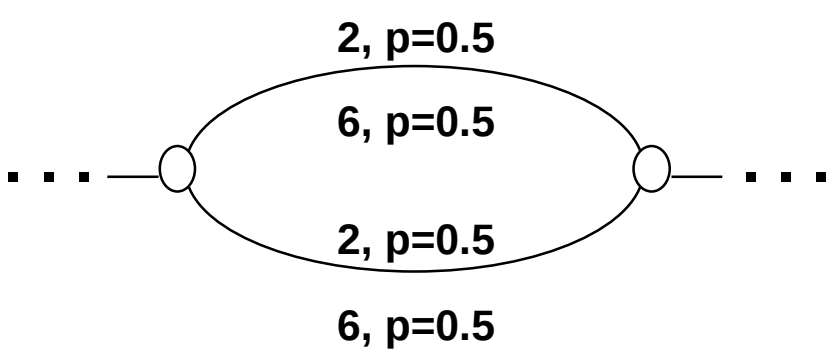
Collaboration and Negotiation Techniques

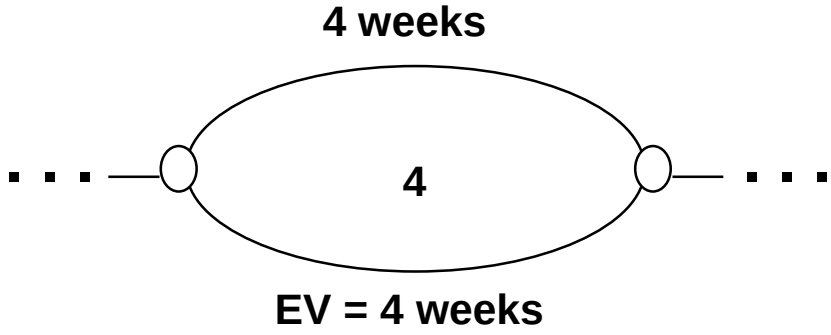
- **Highlighted in 3 of the last 6 ICSE keynote addresses**
  - **Tom DeMarco, "How the requirements were negotiated is far more important than how the requirements were specified."**
  - **Ed Yourdon, "Negotiation is the best way to avoid Death March projects."**
  - **Mark Weiser, "Problems with reaching agreement were more critical to my projects' success than such factors as tools, process maturity, and design methods."**

Reducing Task Variance

- **Or, "Where did that week go?"**

-- Are these two networks equivalent?





EV = 4 weeks

4 Equally likely outcomes

2
2 → 2

2
6 → 6

6
2 → 6

6
6 → 6

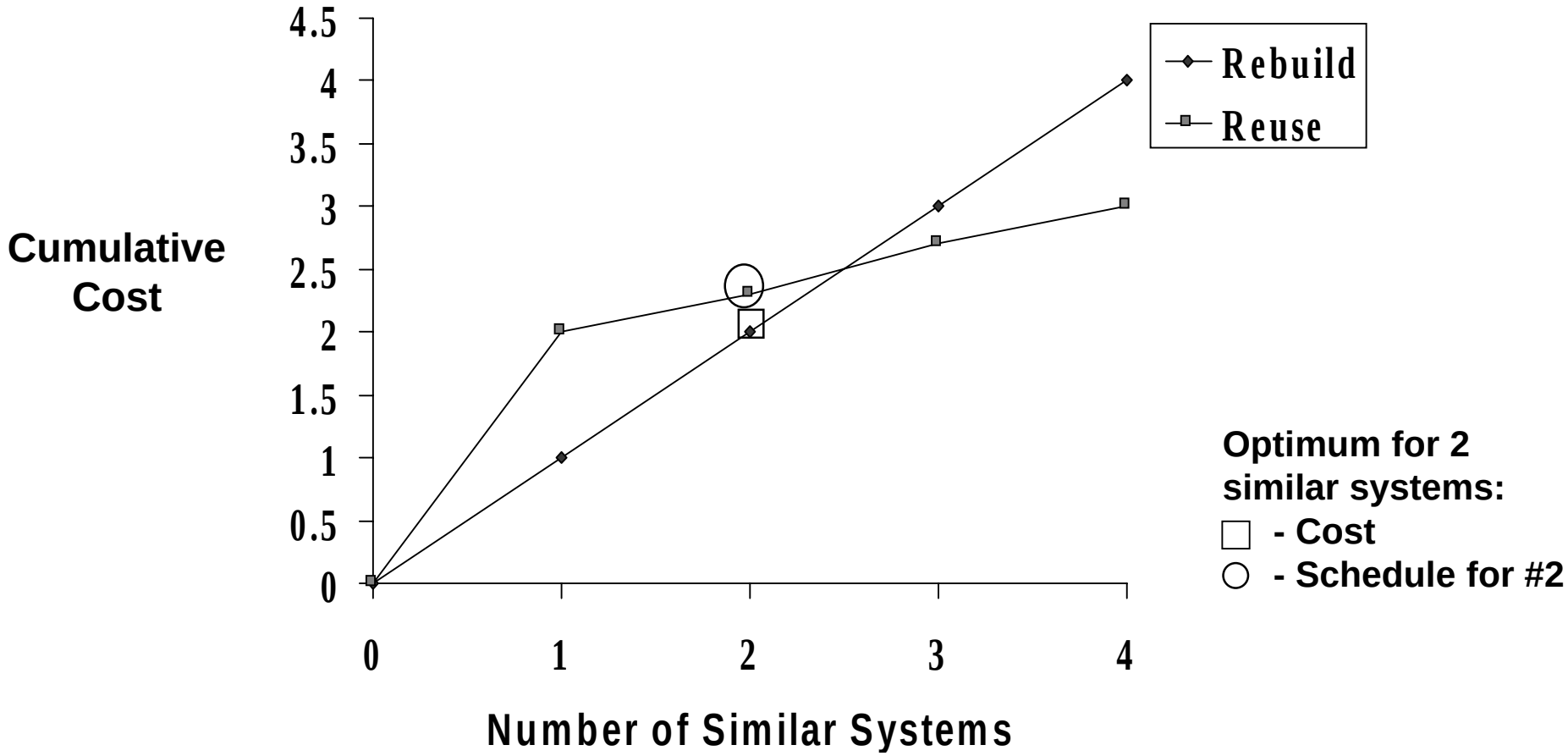
20

EV = 20/4 = 5 weeks

Getting Tasks Off the Critical Path

- **Decomposition and parallelization**
  - **Replace Critical Design Review by unit inspections**
  - **Pre-integrate subsystems**
  - **Parallelize off-nominal testing**
  - **Massive beta testing**
  - **Pre-work unit level acceptance tests**
- **Pre-positioning facilities, components, tools, experts, data**
  - **"Overinvesting" in reusable components**

"Overinvesting" in Reusable Components



People and RAD

- Better and fewer people
  - Bright, quick, versatile, adaptable, creative, experienced, focused
- Clear RAD priorities and incentives
- Teambuilding and shared vision
  - All of the stakeholders
- Co-location
- Capitalization
- Learning, metrics, continuous improvement

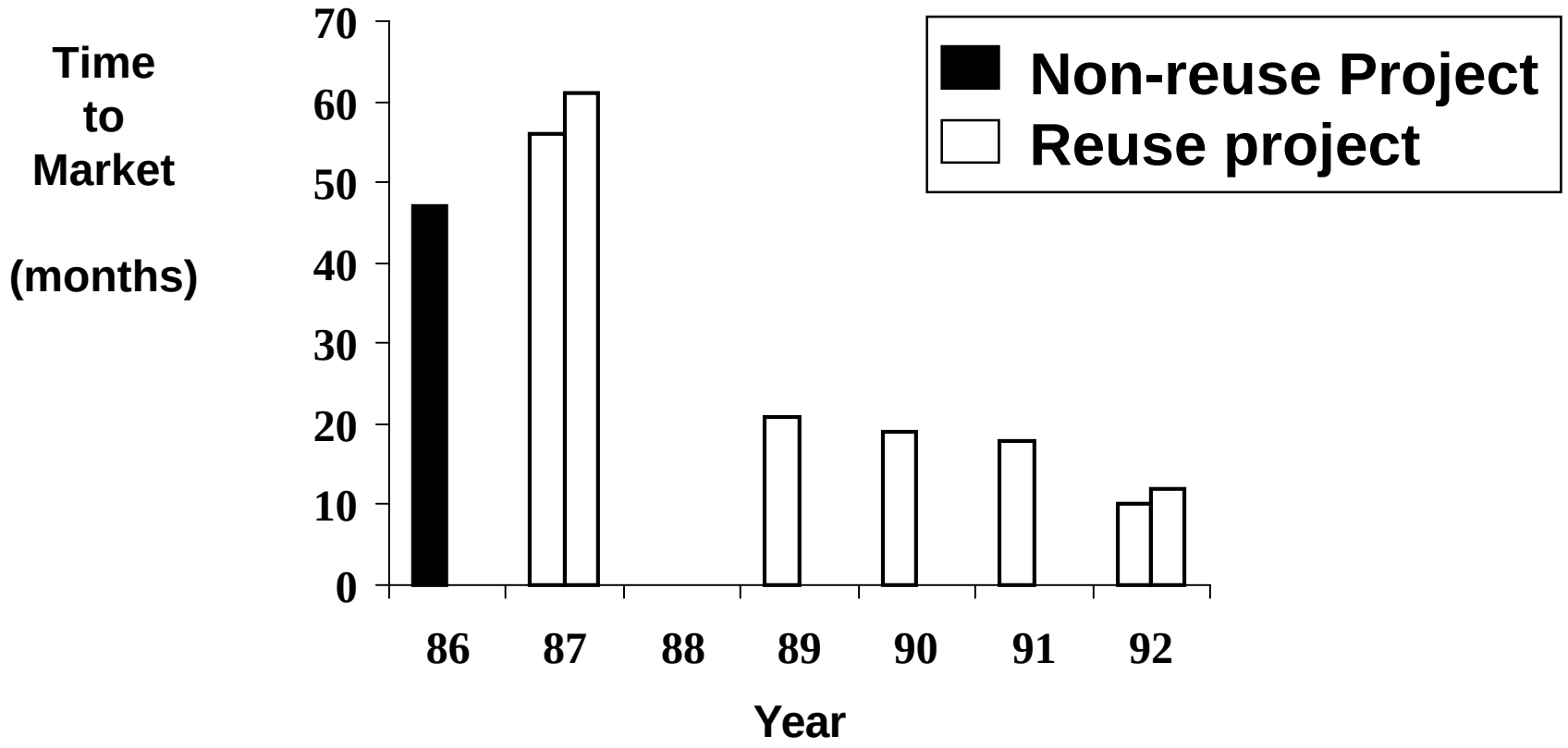
Outline

- RAD Context
- RAD Opportunity Tree and Strategies
- RAD Issues
  - Biggest Opportunity Areas
- Some Successful RAD Experiences

Biggest Opportunity Areas

- People
- Prepositioning
  - Domain engineering, architecting
  - Reusable everything: plans, specs, class libraries, middleware, tests, manuals
  - People and teambuilding
  - Tools and facilities
  - Preworking asset evolution

Reuse at HP's Queensferry Telecommunication Division



Prepositioning Example: TRW CCPDS-R

- Large IR&D investment
  - Domain engineering and architecting
  - Infrastructure: UNAS precursor
  - Tools: Rational Ada tools; metrics; documents templates
  - Ada process model
- People, teambuilding, and incentives
  - Flowdown of award fee to performers
- Results
  - Spectacular 3-week SW Engineering Exercise
  - 1 MLOC within ambitious budget and schedule

RAD with the WinWin Spiral Model

- 11 weeks to architect 15 Library Multimedia archive applications
  - Films/videos, images, manuscripts, urban plans, business reports, etc.
  - Using LCO, LCA anchor point deliverables
  - Using WinWin negotiation system
  - All delivered on schedule; clients highly satisfied
- 11 weeks to develop 6-8 of architected applications
  - Full product-oriented deliverables
  - All delivered on schedule
  - All highly satisfactory to clients except one
    - 3-in-1 merge of image archives

Problem Statement #4:

Medieval Manuscripts

Ruth Wallach, Reference Center, Doheny Memorial Library

I am interested in the problem of scanning medieval manuscripts in such a way that a researcher would be able to both read the content, but also study the scribe's hand, special markings, etc. A related issue is that of transmitting such images over the network.

Digital Manuscript Archive Home Page - Netscape

File Edit View Go Communicator Help

Back Forward Reload Home Search Guide Print Security Stop

Bookmarks Location: http://sunset.usc.edu:80/cs577_96/dma/home/

Internet New and Cool Look Up

Digital Manuscript Home Pag

el Southern in on-line of your in

hi Collection

Digital Manuscript Archive Manuscript Index Page - Netscape

File Edit View Go Communicator Help

Back Forward Reload Home Search Guide Print Security Stop

Bookmarks Location: http://sunset.usc.edu:80/cs577_96/dma/scripts/detail

Internet New and Cool Look Up

Antiphonarium

Title	Antiphonarium
Author	Catholic Church
Date	15th Century
Type	Liturgical & Ritual
Style	Not Available
Physical Characteristics	On vellum; red staves with black Gregoria capitals and rubrication. Dimension of leaves is 57cm x 41cm.

Document: Done

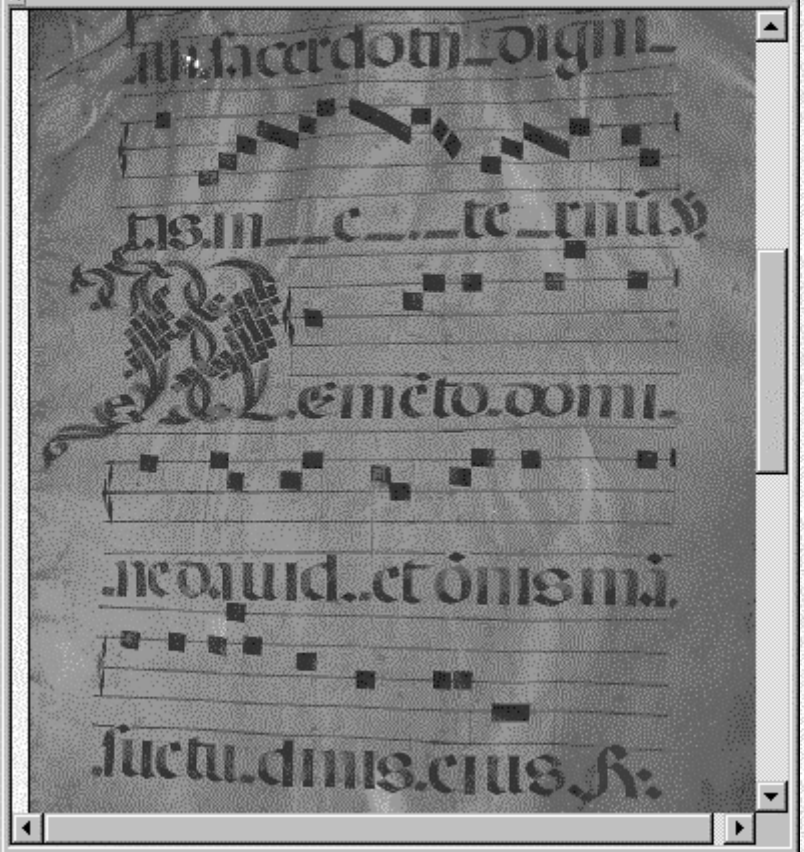
Digital Manuscript Archive Display Leaf Image Page -...

File Edit View Go Communicator Help

Back Forward Reload Home Search Guide Print Se

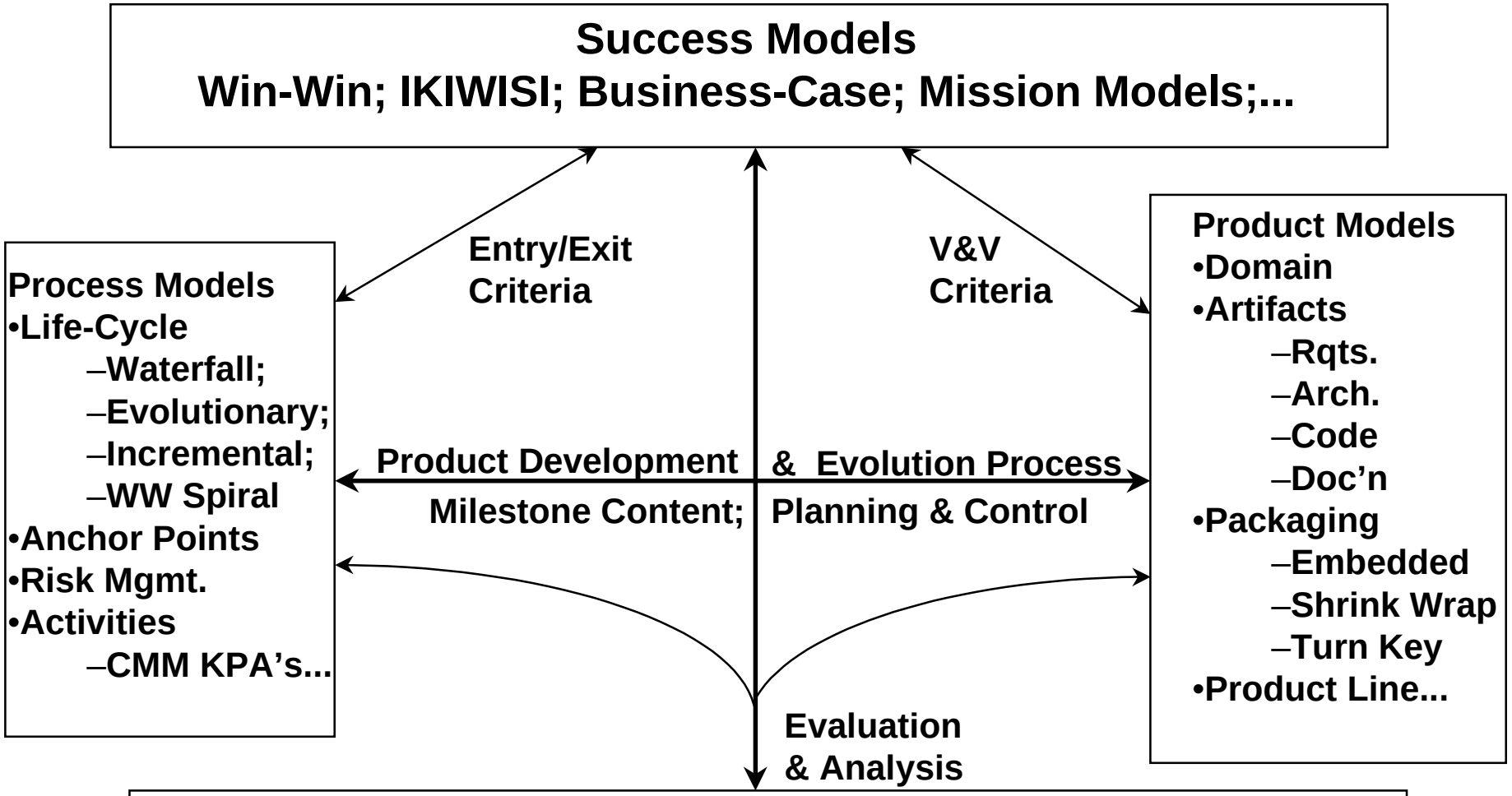
Bookmarks Location: tiphonarium&pageno=10&m=1&p=0

Internet New and Cool Look Up



Document: Done

MBASE* Integration Framework



*MBASE: Model-Based Architecting and Software Engineering

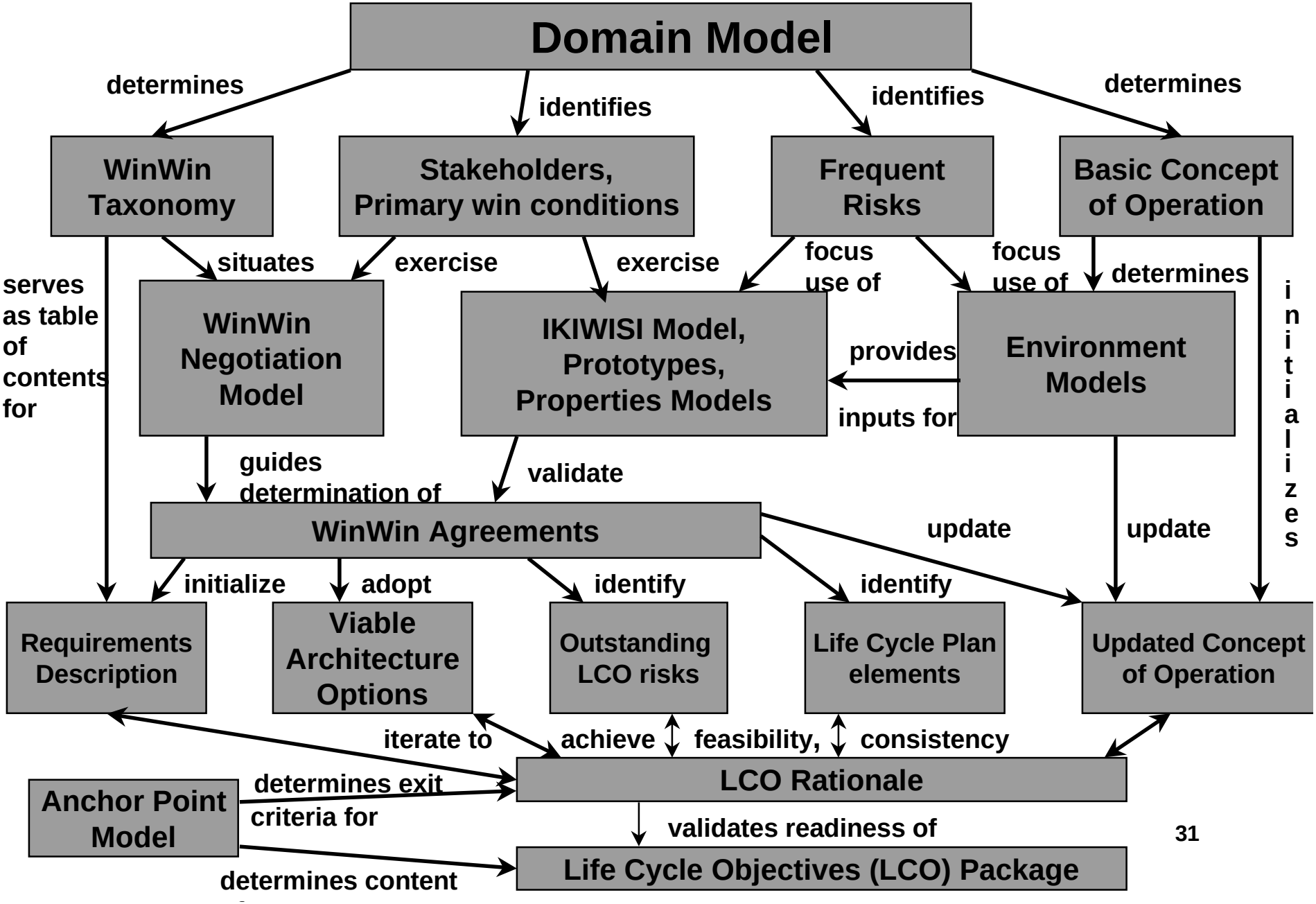
Elements of Critical Front End Milestones

(Risk-driven level of detail for each element)

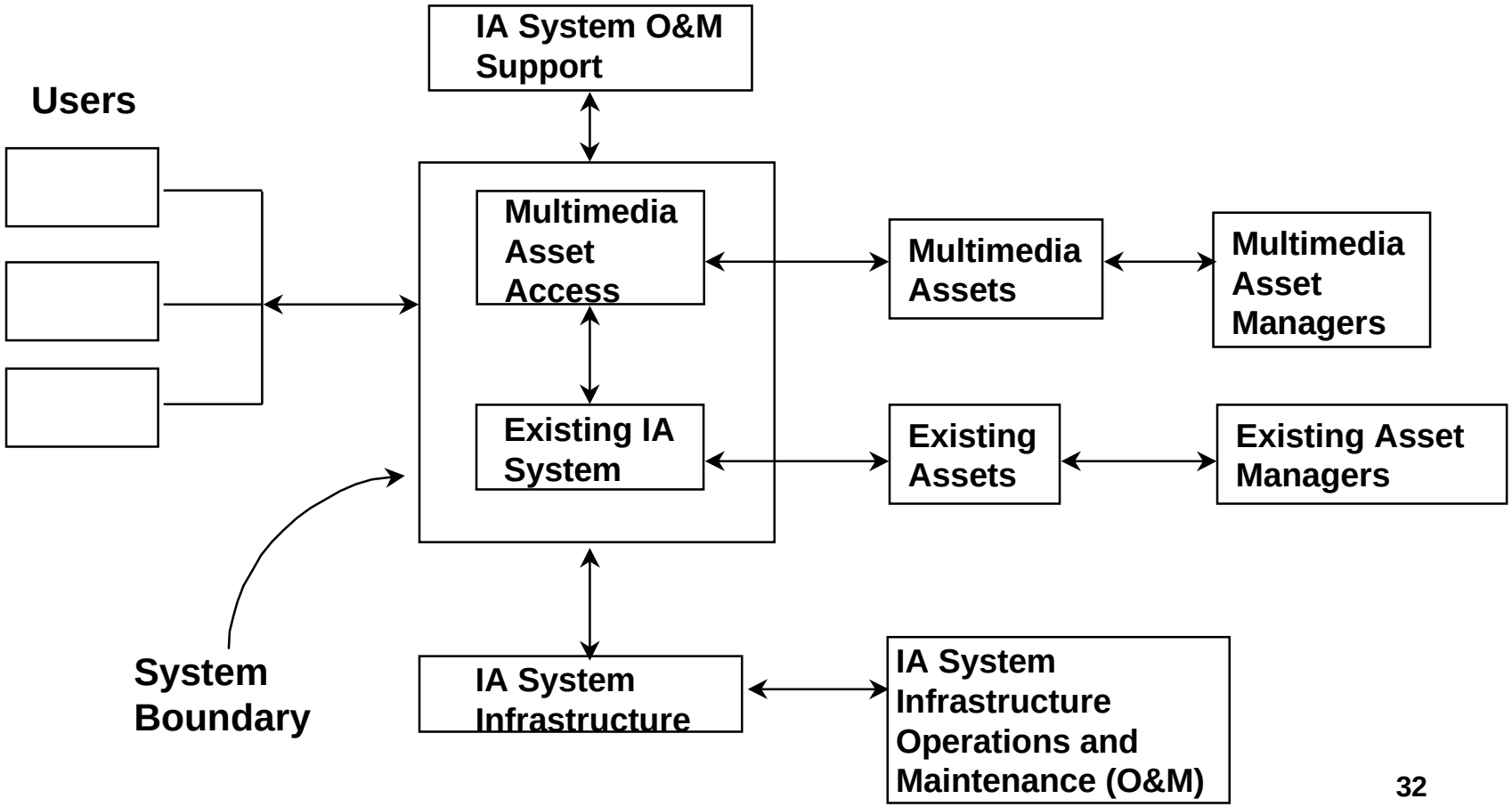
Milestone Element	Life Cycle Objectives (LCO)	Life Cycle Architecture (LCA)
Definition of Operational Concept	• Top-level system objectives and scope - System boundary - Environment parameters and assumptions - Evolution parameters • Operational concept - Operations and maintenance scenarios and parameters - Organizational life-cycle responsibilities (stakeholders)	• Elaboration of system objectives and scope of increment • Elaboration of operational concept by increment
System Prototype(s)	• Exercise key usage scenarios • Resolve critical risks	• Exercise range of usage scenarios • Resolve major outstanding risks
Definition of System Requirements	• Top-level functions, interfaces, quality attribute levels, including: - Growth vectors and priorities - Prototypes • Stakeholders' concurrence on essentials	• Elaboration of functions, interfaces, quality attributes, and prototypes by increment - Identification of TBD's((to-be-determined items) • Stakeholders' concurrence on their priority concerns
Definition of System and Software Architecture	• Top-level definition of at least one feasible architecture - Physical and logical elements and relationships - Choices of COTS and reusable software elements • Identification of infeasible architecture options	• Choice of architecture and elaboration by increment - Physical and logical components, connectors, configurations, constraints - COTS, reuse choices - Domain-architecture and architectural style choices • Architecture evolution parameters
Definition of Life-Cycle Plan	• Identification of life-cycle stakeholders - Users, customers, developers, maintainers, interoperators, general public, others • Identification of life-cycle process model - Top-level stages, increments • Top-level WWWWWHH* by stage	• Elaboration of WWWWWHH* for Initial Operational Capability (IOC) - Partial elaboration, identification of key TBD's for later increments
Feasibility Rationale	• Assurance of consistency among elements above - via analysis, measurement, prototyping, simulation, etc. - Business case analysis for requirements, feasible architectures	• Assurance of consistency among elements above • All major risks resolved or covered by risk management plan

*WWWWWHH: Why, What, When, Who, Where, How, How Much

MBASE Model Integration: LCO Stage



Domain Model: Block Diagram



IA: Information Archive

MBASE Laboratory

- 15 software engineering projects/year
  - 5-person USC Digital Library applications
- Rapidly developing successful applications
  - Multimedia, virtual assistants, data acquisition
- Integrating models and tools
  - DARPA-EDCS architecture and WinWin tools
  - Rational Rose, Unified Modeling Language
- Rapidly improving artifact integration
  - 1996 integrated specs, plans: 160 pages
  - 1997 integrated specs, plans: 103 pages
- Results transitioning to B-2, JSTARS, Satellite control, MCC SSEP, Rational
- Ultimate goal: Model-integrated SW Engr. agents

Skunk Works Rules As Applied To Software Engineering

1. Software Manager assigned to Projects reporting directly to Project Manager - oversees suppliers and in-house personnel.
3. Aggressive use of small software engineering teams
4. Careful application of configuration management and change control
5. Documentation restricted by “essential” criteria. Simplified formatting where appropriate
6. Regular metrics collection and analysis for managing progress against schedule and budget
7. Flowdown of basic rules to suppliers

Information contained in this slide was taken from a Lockheed Martin presentation given by Gary Hafen at the June 1997 Focused Workshop.

Skunk Works Rules As Applied To Software Engineering

8. Task Streamlining - “value-added” criteria applied - no duplication of effort

9. Software Development Organization tests and verifies software in lab or on vehicle prior to turning it over to formal independent test organization.

10. Use of “intent” of Mil-Stds rather than full conformance. Use of Mil-Std negotiated with customer up front.

12. End User encouraged to participate early during development and provide critique and feedback when decisions need to be made.

Information contained in this slide was taken from a Lockheed Martin presentation given by Gary Hafen at the June 1997 Focused Workshop.

How we measure our cycle time reduction efforts - X Factors

• Process Mapping	1.3
• Better Hardware	1.0
• Better Tools and Platforms	1.3
• Rigorous Software Inspections	1.2
• In-Process Software Metrics	1.1
• Sharing of Best Practices	1.4
• All Others	1.1
	3X
Benefit If Fully Deployed:	**4X**

Biggest Opportunity Areas

- People
- Prepositioning
  - Domain engineering, architecting
  - Reusable everything: plans, specs, class libraries, middleware, tests, manuals
  - People and teambuilding
  - Tools and facilities
  - Preworking asset evolution