



System and Software Construction, Transition and Support Planning

**CS 577B
Spring 2002**

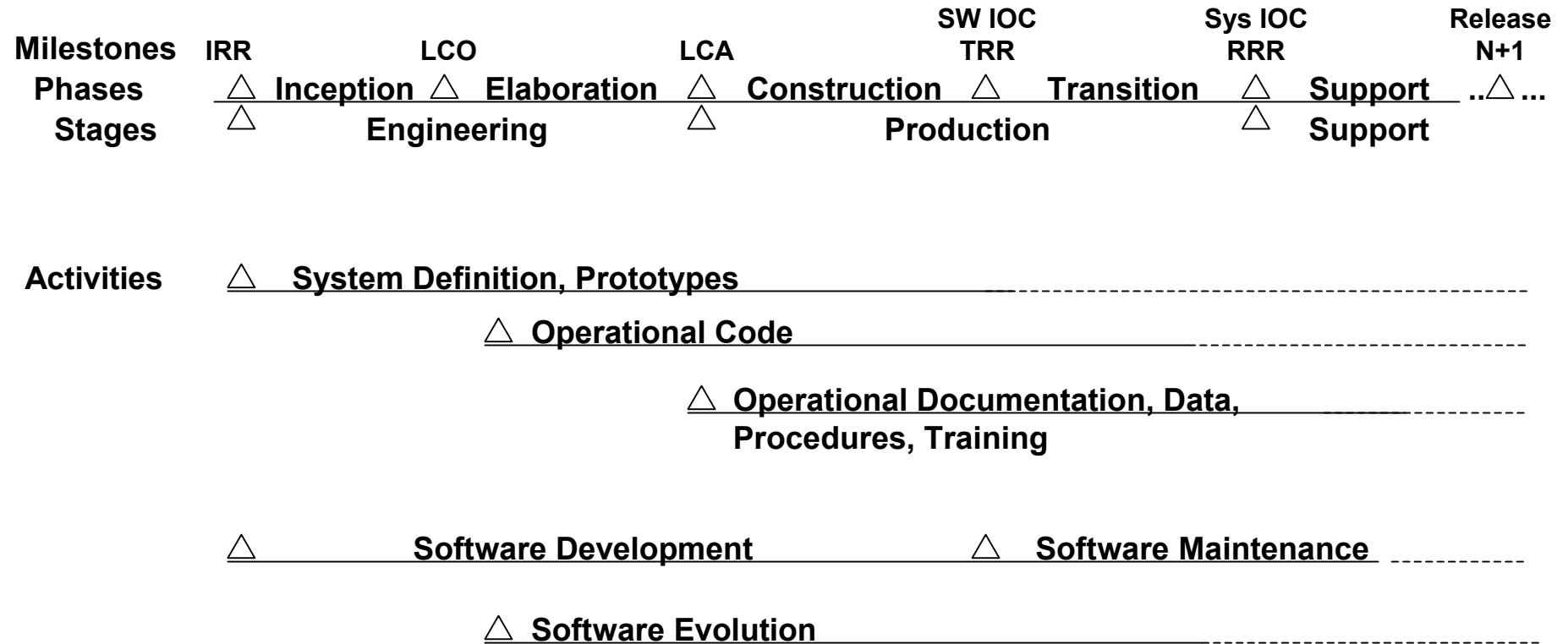


Outline

- **Overview**
 - **Software Maintenance and System Support**
 - **CS 577b Challenges**
- **The System and Software Transition Plan (SSTP)**
- **The System and Software Support Plan (SSSP)**
- **Construction Planning**



Life Cycle Definitions





Transition and Support Planning

- **Objective: to make winners of the operational stakeholders**
 - **Users**
 - **Support stakeholders**
 - **Software maintainers, system administrators, database administrators, operators, user supporters**
- **IOC (Initial Operational Capability) Milestone**
 - **Like having your first child**
 - **Operational system health preservation**
 - **Corrective and adaptive maintenance**
 - **Quality of service**
 - **Dependability: safety, security, availability, ...**
 - **Changes need to consider**
 - **Continuous operation**
 - **No surprises or major inconveniences**
 - **Preservation of existing services**



Software Maintenance Phenomenology

- **The Iron Law of Software Maintenance**
- **Software Maintenance production function**
- **The software maintenance manager's whiteboard**
- **Lehman-Belady “Laws” of evolution dynamics**
- **Effects of COTS on software maintenance**



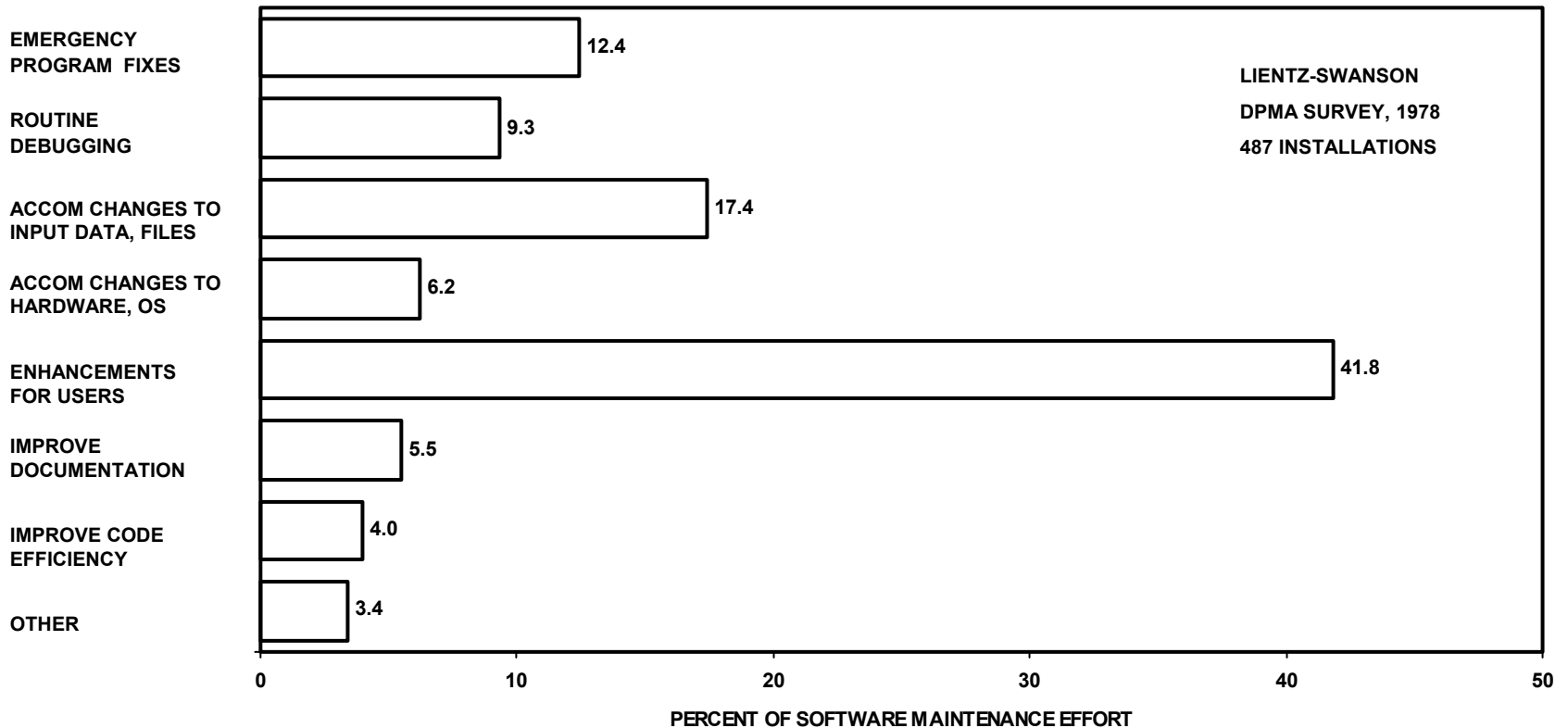
The Iron Law of Software Maintenance

- **Useful software systems will spend twice as much on software maintenance as they did for development**
 - **For CS 577 clients:**

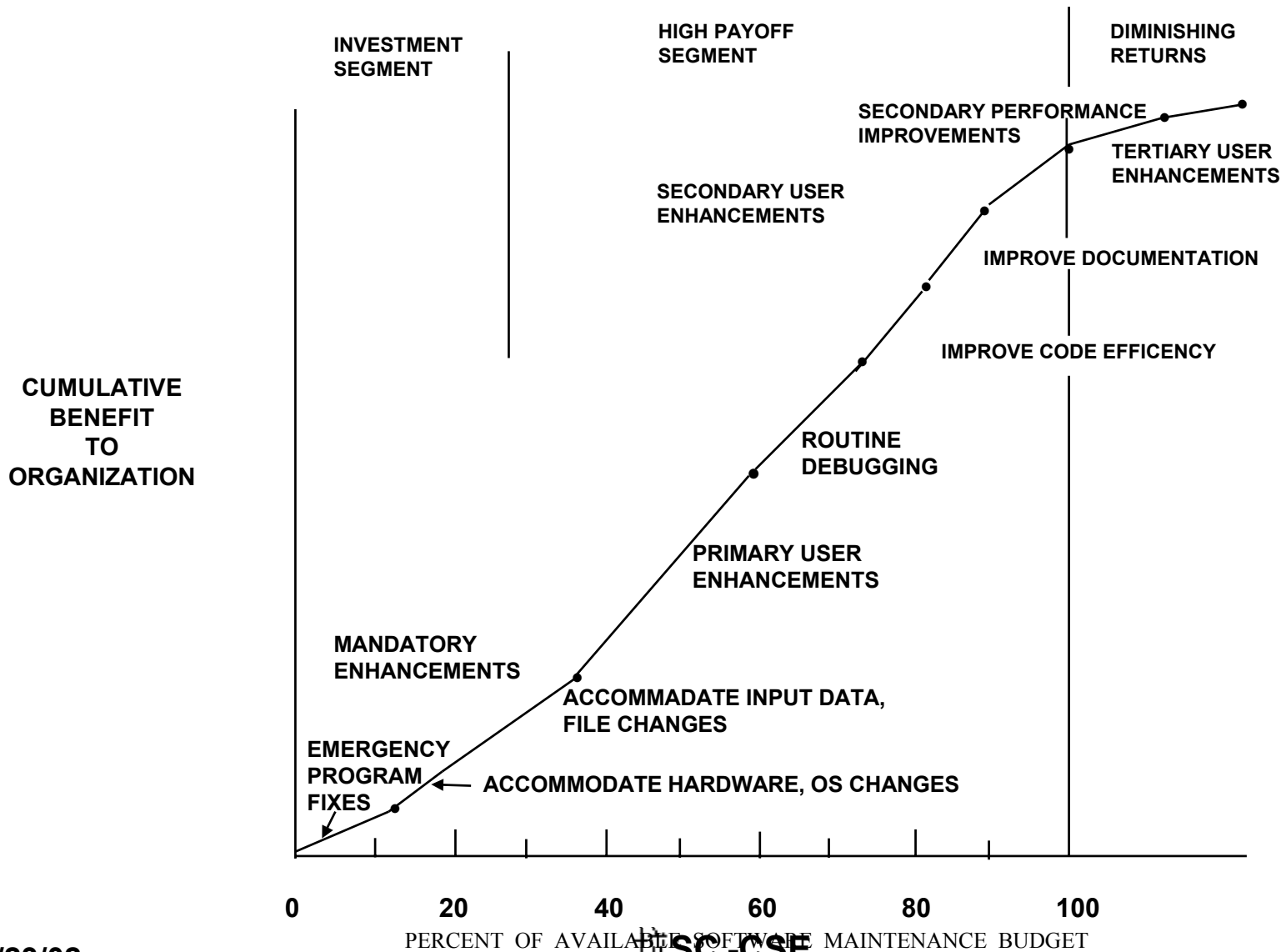
Development = (24 weeks)(12 hr/week)(5 persons) = 1440 person-hr

Maintenance = (2)(1440 PH)(\$50/PH) = \$144,000

Distribution of Software Maintenance Effort



The Software Maintenance Production Function





Lehman-Belady “Laws” of Evolution Dynamics

1. Continuing change
2. Increasing complexity
3. Cyclic self-regulation
4. Invariant work rate
5. Conservation of familiarity

Effects of COTS on Software Maintenance

- You have no control over COTS evolution
- Maintenance headaches scale exponentially with # COTS
- You need a pro-active COTS-based-system evolution strategy
 - Go for dominant open standards
 - Factor evolution requirements into COTS selection
 - Evaluate COTS vendors' evolution track records
 - Use flexible architectures
 - CORBA, COM, software bus, layering, event/message-based
 - Encapsulation : use COTS wrappers
 - Technology watch investments
 - Synchronized COTS upgrade/release strategy



CS 577b Challenges

- **No developers available for software/system support**
- **Short time period to develop and transition system**
 - **And for clients to prepare for transition system**
- **Period of change in ISD support**
 - **Focus on pure campus-wide infrastructure : computing, communications, networking, data**
 - **Applications organizations manage their own specialized infrastructure**

Outline

- **Overview**
 - **Software Maintenance and System Support**
 - **CS 577b Challenges**
- • **The System and Software Transition Plan (SSTP)**
- **The System and Software Support Plan (SSSP)**
- **Construction Planning**

Transition Plan Overview

- Purpose: ensure successful transition into operations
- Timing: draft plan 2 weeks after LCA ARB reviews; final plan at Transition Readiness ARB Review
- Completion Criteria: stakeholder concurrence; feasibility of execution; assurance of successful transition
- Intended Audience: transition stakeholders: user, customer, developer, maintainer, operator, supplier
- Dependencies: OCD 4 transition aspects; LCP 2 deliverables; LCP 3 life cycle support responsibilities

1. Transition Strategy

- Transition objectives
 - Extent: full operation; limited pilot operation
 - Sites: one/many; homogeneous/heterogeneous
 - Developer continuity: none/full/intermediate
 - Degree of operational test and evaluation
 - Product transitioned: COTS/new system/legacy upgrade
- Transition process strategy
 - Cutover: instant/incremental/parallel operation
 - Phasing of multiple increments to multiple sites
 - Role of alpha testing, beta testing, formal operational testing

2. Preparing for Transition

- Hardware preparation: purchases, installation
- Software preparation: licenses, rehosting, data
- Site preparation: facilities, equipment, communications
- Staff preparation: training, teambuilding, change readiness
- Operational test: criteria, procedures, personnel, instrumentation, analysis

Other Transition Plan Sections

3. Stakeholder Roles and Responsibilities

- Coordinated with stakeholder organizations

4. Milestone Plan

- Best to work backward from completed transition
- Coordinated with stakeholder organizations

5. Required Resources

- Budgets, key stakeholder time, facilities

Outline

- **Overview**
 - **Software Maintenance and System Support**
 - **CS 577b Challenges**
- **The System and Software Transition Plan (SSTP)**
- • **The System and Software Support Plan (SSSP)**
- **Construction Planning**

System and Software Support Plan

- **Purpose:** to guide support stakeholders in successfully operating and maintaining the system
- **Timing:** Builds on OCD, LCP. Developed during Construction Phase. ARB review at TRR. Updated during Transition Phase.
- **Intended Audience:** Support stakeholders, plus developers (target for Transition Plan), users (operational information)
- **Responsibilities:** joint between support stakeholders (normally the leaders) and developers (leaders in 577b)
- **Completion Criterion:** stakeholder concurrence, feasibility of execution, resource realism

Support Plan Contents

- 1. Support Objectives and Assumptions (why, whereas)**
- 2. Support Strategy and Environment (what, when)**
- 3. Support Responsibilities (who, where)**
- 4. Support Approach (how)**
- 5. Support Resources (how much)**

Example Support Objectives and Assumptions

- **Key driving objectives for support activities**
 - **Ensure system safety; top customer satisfaction; competitive speed**
 - **Replace inefficient old systems quickly**
 - **Provide more promising support career paths**
- **Assumptions required for workability of support plan**
 - **Continuity of funding, staffing, executive support**
 - **Controllable/negotiable interfaces with other systems**
 - **Stability of next-release requirements and schedules**

Support Strategy

- **Estimated support lifetime**
 - Under 1 year; over 5 years; until XYZ phased out
- **Release strategy**
 - Continuous small fixes; annual release; COTS refresh cycle
 - Early next-release content (evolution reqts.)
 - Release transition strategy (testing; multisite sequencing)
- **Release requirements determination**
 - Primary drivers (budget, schedule, staffing, marketplace)
 - Process (stakeholder win-win, bidding, sub-allocations)
- **Release process**
 - Normally inception/elaboration/construction/transition

Approach and Resources

- **Approach: identify differences in the support approach and the development approach in LCP Section 4**
 - E.g., differences in methods, tools, & facilities
- **Support budgets: Work breakdown structure; initial budgets consistent with FRD Section 2.**

Outline

- **Overview**
 - **Software Maintenance and System Support**
 - **CS 577b Challenges**
- **The System and Software Transition Plan (SSTP)**
- **The System and Software Support Plan (SSSP)**
- • **Construction Planning**



Initial Operational Capability (IOC) Deliverables

Construction Phase

Construction Deliverables

- Construction Specialty Plans
 - In Construction Transition Support (CTS) MBASE guidelines
- Tracking and Control Deliverables
 - Metrics
- Working Deliverables
 - Code base
 - COTS packages (integrated, configuration)
 - Revised Documentation



Specialty Plans

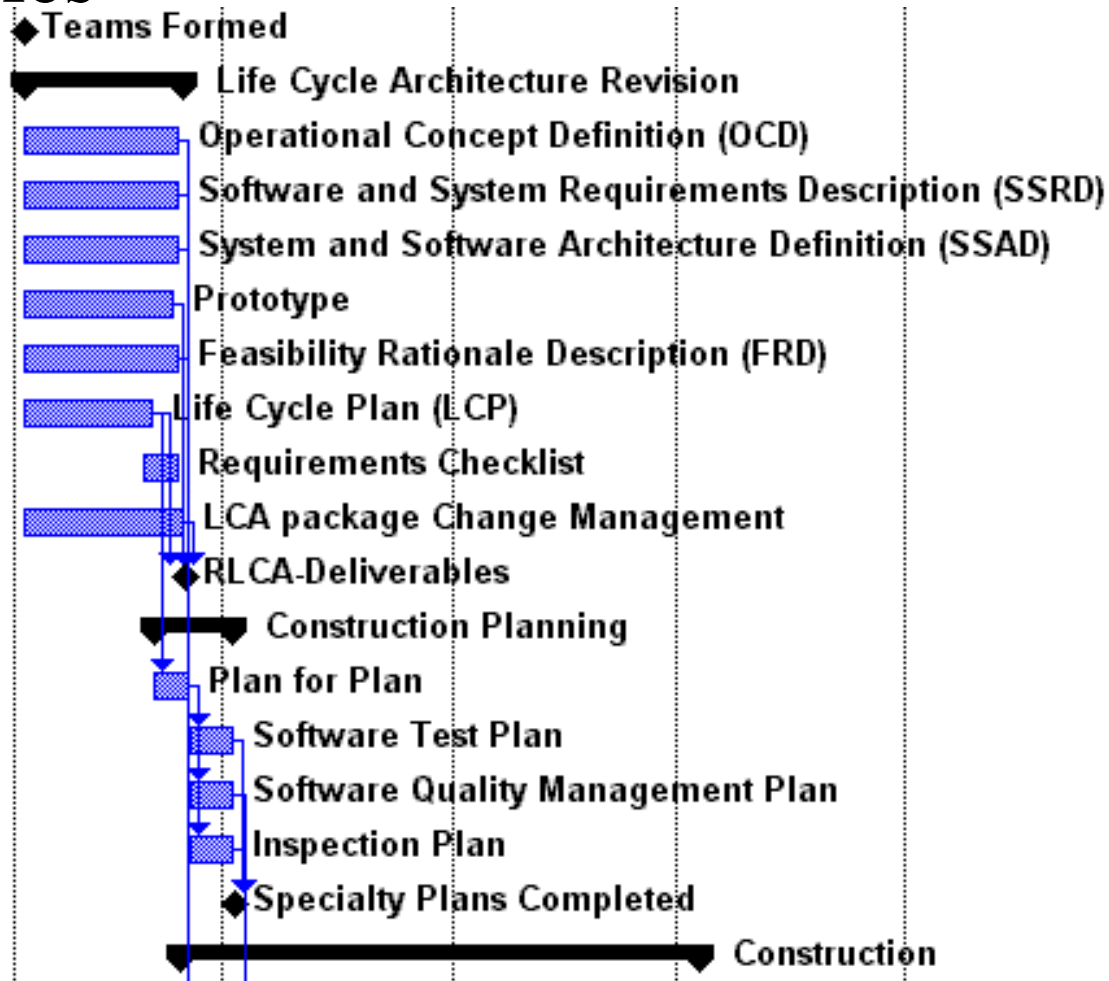
- Iteration Plan
- Quality Management Plan
- Test Plan
- Inspection Plan
- Security Plan (optional)
- Safety Plan (optional)

All Plans and Major Activities Should be Explicitly Planned

- Allocate effort and people in LCP to
 - Write plans
 - Execute plan activities
 - Prepare for RLCA and TRR reviews, core capability demos
- Anticipate and account for risks
 - Allocate extra time for risky items
 - Explicitly schedule critical contingency plans
- Be consistent with the class schedule

Construction Planning Activities, Milestones and Deliverables

Teams Formed	
<input type="checkbox"/> Life Cycle Architecture Revision	
Operational Concept Definition (OCD)	
Software and System Requirements Desc	
System and Software Architecture Definitio	
Prototype	
Feasibility Rationale Description (FRD)	
Life Cycle Plan (LCP)	
Requirements Checklist	
LCA package Change Management	
RLCA-Deliverables	
<input type="checkbox"/> Construction Planning	
Plan for Plan	
Software Test Plan	
Software Quality Management Plan	
Inspection Plan	
Specialty Plans Completed	
<input type="checkbox"/> Construction	



Project Schedule

- Jan. 14** - **Re-form teams**
- Feb. 4** - **Draft LCA-Rebaseline on Web**
- Feb. 14-15** - **LCA-Rebaseline ARB reviews**
- Mar. 18-27** - **Core Capability Demos**
- Apr. 1** - **CTS Working Set #1 due**
- Apr. 8** - **Draft Transition Package**
- Apr. 15-16** - **Transition Readiness Reviews**
- May 1** - **Product Delivery**
- May 3** - **Individual Critiques, Client Evaluations**

Summary of Client Activities

Jan 14 - Feb 14: Work with teams:

- Rebaseline prototype, prioritized requirements
- Consider use of Easy Win Win
- Plan for CS 577b specifics, including transition strategy, key risk items

Feb 14-15: Participate in ARB review of rebaselined Life Cycle Architecture Package

Feb18- Apr 12: Nominal Weekly Meetings with Teams to:

- Discuss status and plans
- Provide access to key transition people for strategy and readiness discussions

Mar 18-27: Core Capability Demos

Apr 15-Apr 16: Project Transition Readiness Reviews

Apr 18: Begin Installation and Transition

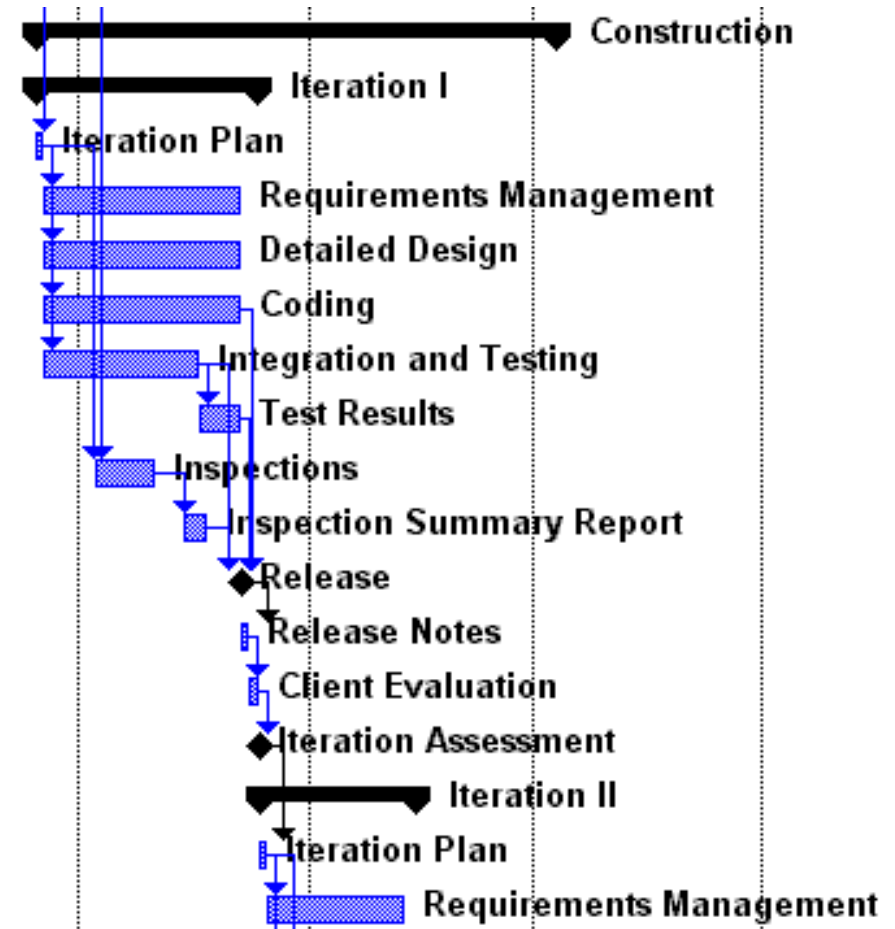
- Install Product
- Execute Transition Plan

May 1: Release Readiness Review for Product Release (client acceptance)

May 3: Client Evaluations

Iteration Activities, Milestones and Deliverables

[-] Construction
[-] Iteration I
Iteration Plan
Requirements Management
Detailed Design
Coding
Integration and Testing
Test Results
Inspections
Inspection Summary Report
Release
Release Notes
Client Evaluation
Iteration Assessment
[-] Iteration II
Iteration Plan
Requirements Management



Working Deliverables

(per iteration)

- Iteration Plan
- Test Reports
- Inspection Reports
- Release Notes
- Iteration Assessment Report
- Source Code (under CM)
- *As-built* OCD, SSRD, SSAD, FRD, LCP

Iteration Plan

- Plan for the upcoming iteration is planned in the current iteration
- One iteration plan is input to the next iteration plan.
 - Iteration Content
 - Use Cases/Requirements Addressed
 - Objectives
 - Schedule of Activities
 - Team Responsibilities
 - ...

Iteration Assessment Report

- Each iteration is concluded by an iteration assessment
 - Objectives Reached
 - Adherence to Plan
 - Use Cases/Requirements Implemented
 - Results Relative to Evaluation Criteria
 - External Changes Occurred
 - Rework Required
 - ...

Release Notes

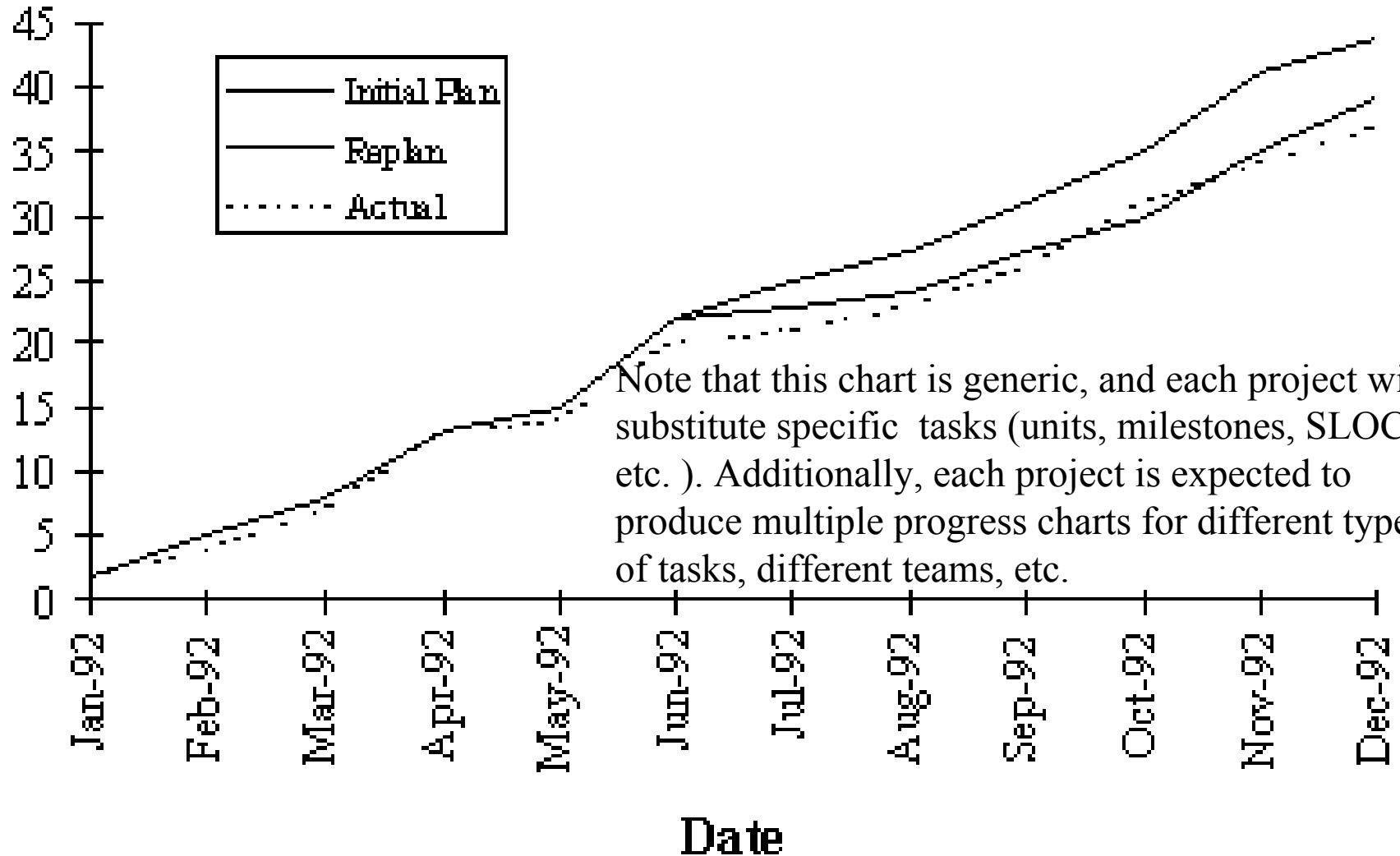
- The purpose of the Release Notes is to describe the release
 - New Features and Important Changes since the previous release
 - Upcoming Changes that will be incorporated in future releases
 - Known Bugs and Limitations
 - ...

Metrics

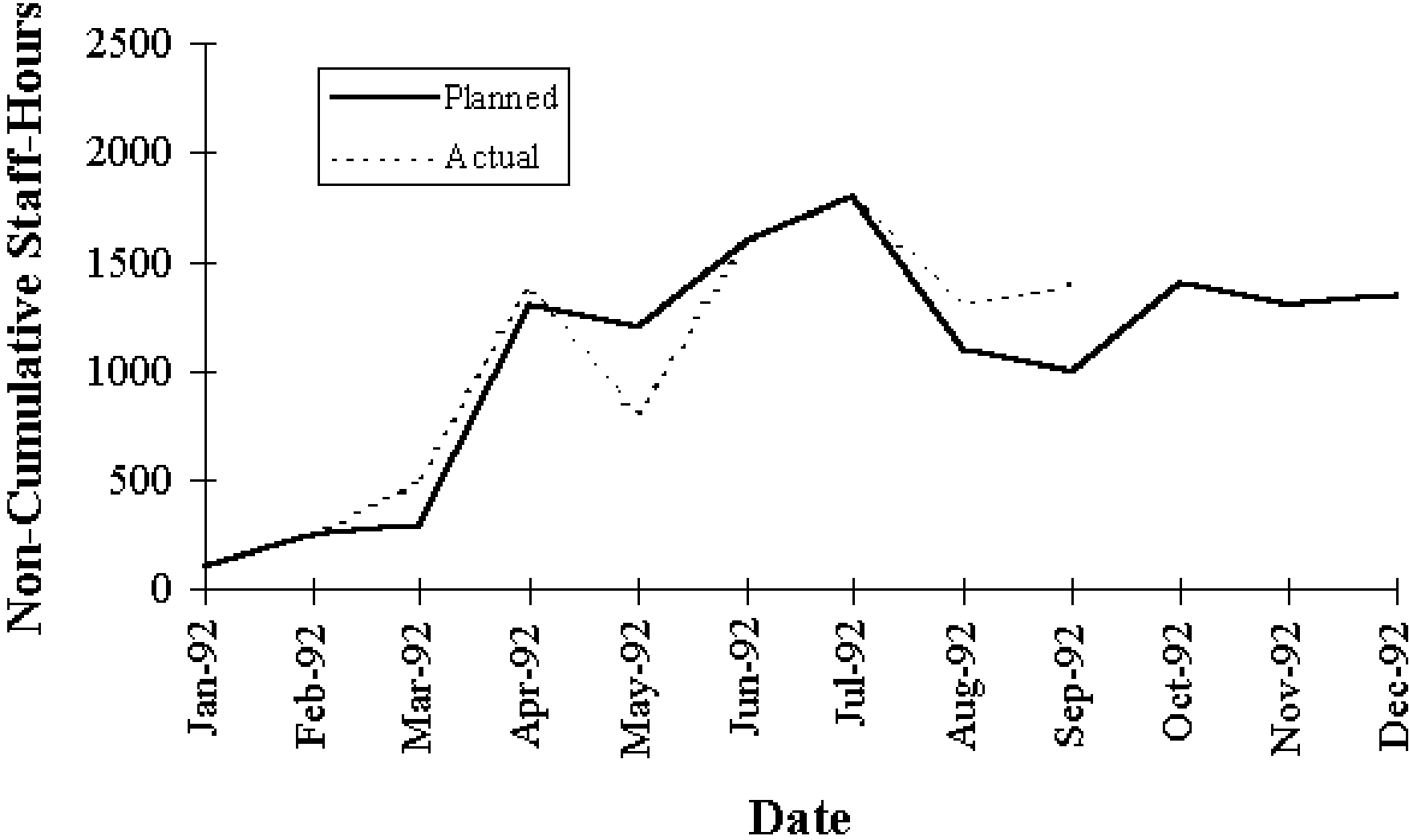
- A metric quantifies a characteristic of a process or product.
 - Metrics can be directly observable quantities or can be derived from one or more directly observable quantities.
 - Examples of raw metrics include:
 - number of source lines of code,
 - number of documentation pages,
 - number of staff-hours,
 - number of tests,
 - number of requirements
 - Examples of derived metrics include:
 - source lines of code per staff-hour,
 - defects per thousand lines of code,
 - cost performance index.

Progress

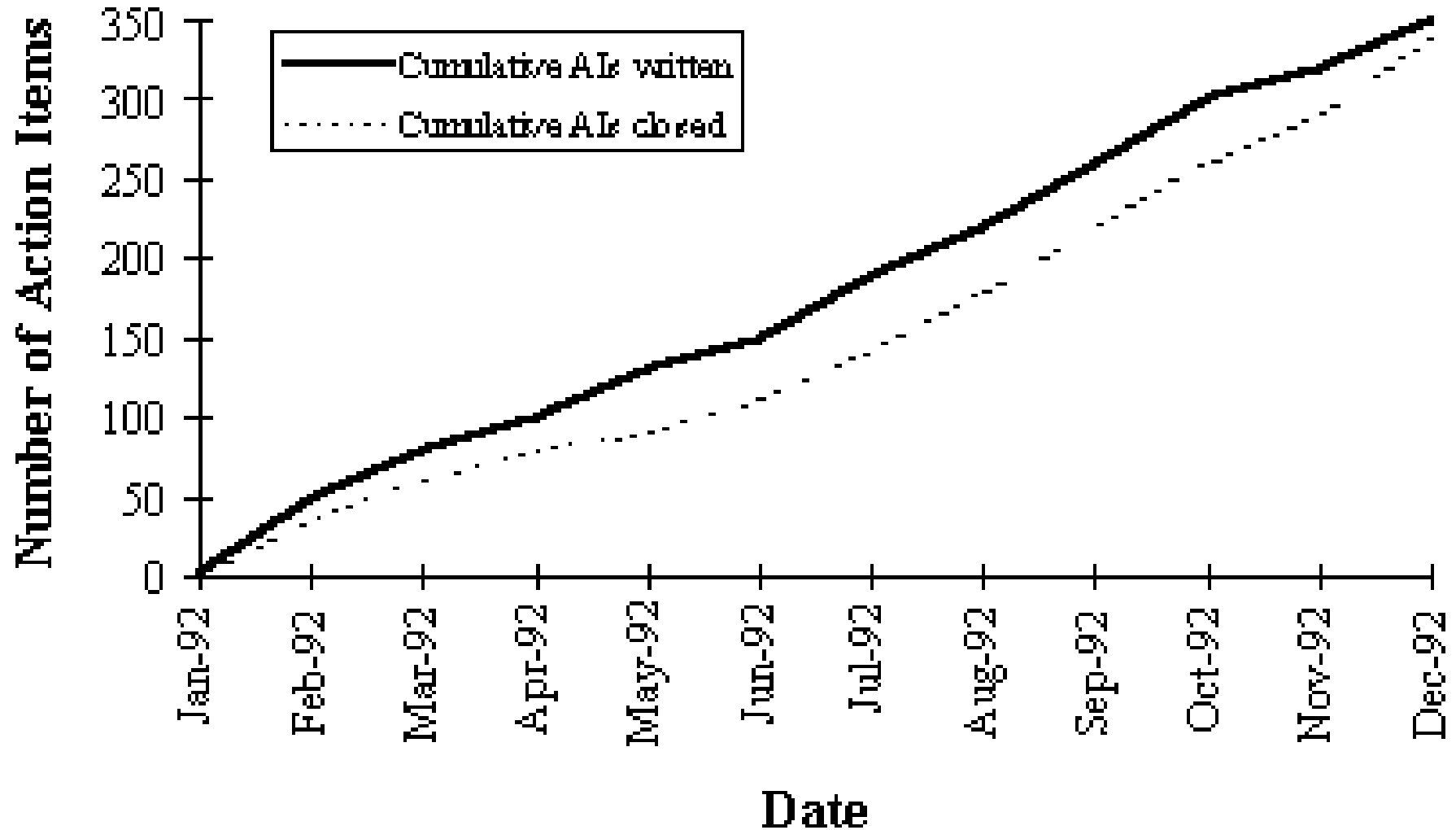
Cumulative Tasks Completed



Effort



Review Results



Trouble Reports

