Chapter 1

Software Process

Copyright © 2000 by Addison Wesley
Version 1.0

Topics

- The Nature of Software Development
- System Planning
  - Software Lifecycle Phases
  - Software Development Approaches

The nature of software (Brooks)

- The software essence
  - Complexity
  - Conformity
  - Changeability
  - Invisibility
- The software accidents
  - Stakeholders
  - Process
  - Modeling language and tools
Software development invariant

- Software production is an art
  - Software is developed, not manufactured
  - … but
    - OT & re-use
    - COTS
    - ERP
  - … but what about core business?
  - Component technology
    - CORBA
    - DCOM
    - EJB

Stakeholders

- Two groups
  - Customers
    - Users
    - System owners
  - Developers
    - Analysts
    - Designers
    - Programmers, etc.

- Main causes of software failures
  - “Great designs come from great designers”

Process

- Process for:
  - Order of activities
  - Product delivery (what, when)
  - Assignment to developers
  - Monitoring → measuring → planning

- Cannot be codified or standardized
- Process and project size
- Iterative and incremental
### CMM

- **Level 1**: Initial
  - Unpredictable and undisciplined process that depends on the current staff.

- **Level 2**: Repeatable
  - Repeatable project management; consistent time and effort predictions for similar projects.

- **Level 3**: Defined
  - Both management and engineering processes are codified and followed.

- **Level 4**: Managed
  - Metrics used to control the process.

- **Level 5**: Optimizing
  - Continuous process improvement is in place.

### ISO 9000

- **Quality management**
- **Process**
- **ISO standards are about**
  - What must be accomplished
  - Not about how
- **Certification**
  - Company must document and record its activities
  - On-site audit by an ISO registrar

### Modeling Language and Tools

- **Language**
  - Visual
  - Declarative semantics
- **Tool**
  - CASE
  - Repository
  - Collaboration
  - Versions
  - Consistency and integrity of models
  - Report and code generation
UML

- Rational Software Corporation
- OMG
- Rational Unified Process
- OO
- Implementation independent
- Models
  - State
  - Behavior
  - State change
- CASE and process improvement

System Planning

- Business strategy
  - Small organizations
  - Large organizations
- Approaches
  - SWOT
  - VCM
  - BPR
  - ISA
- Effectiveness vs. efficiency

SWOT

- Mission statement
- Internal strengths and weaknesses
- External opportunities and threats
- Objectives
- Goals
- Strategies
- Policies
**Value chain** – from raw materials to final products sold and shipped to customers

- Primary activities
- Support activities
  - Incl. IS development
- IT can transform organization’s value chain

**Organizations structured as vertical units**

- Who is responsible for a business process
- Processes cut horizontally across the business and end at points of contact with customers
- Process redesign
- Workflow analysis
- BPI
- IT support

**Neutral architectural framework**

- Does not include a system planning methodology
- Table of thirty cells
  - Five rows (perspectives)
    - Planner, owner, designer, builder, subcontractor
  - Six columns (descriptions, architectural models)
    - What, how, where, who, when, why
## Systems and management levels

<table>
<thead>
<tr>
<th>Level of decision making</th>
<th>Focus of decision making</th>
<th>Typical IT applications</th>
<th>Typical IT initiatives</th>
</tr>
</thead>
<tbody>
<tr>
<td>Strategic</td>
<td>Strategies in support of organizational long-term objectives</td>
<td>Market and sales analysis, Product planning, Performance evaluation</td>
<td>Data mining, Knowledge management</td>
</tr>
<tr>
<td>Tactical</td>
<td>Policies in support of operational goals and resource allocation</td>
<td>Budget analysis, Salary forecasting, Inventory control, Customer service</td>
<td>Data warehousing, Accounting, Processing, Supplies</td>
</tr>
<tr>
<td>Operational</td>
<td>Day-to-day staff activities and production support</td>
<td>Supply, Forecasting, Purchasing, Accounting</td>
<td>Database, Transaction processing, Application generation</td>
</tr>
</tbody>
</table>

## Software lifecycle phases

- **Coarse granularity**
  - Analysis
  - Design
  - Implementation

- **Refined granularity**
  - Requirements determination
  - Requirements specification
  - Architectural design
  - Detailed design
  - Implementation
  - Integration
  - Testing

## Requirements phase

- **Requirement** – statement of a system service or constraint

- **Service**
  - Business rule
  - Computation

- **Constraint**

- **Information gathering**

- **Requirements document**
Specification phase

- Requirements document → specification document
- Visual modeling
  - Class diagrams
  - Use case models
- Implementation independent

Architectural design

- Solution strategy
  - Client
  - Server
  - Application logic layer
- Modules (use cases)
- UML:
  - Packages
  - Components
  - Deployment

Detailed design

- User interface (client)
- Database (server)
- Application logic
- UML
  - Class diagrams
  - Use cases
  - Activity diagrams
  - Sequence diagrams
  - Collaboration diagrams
  - Statecharts
Implementation

- Installation
- Coding
- Loading test and production databases
- Testing
  - Performance tuning
- DBA
- User training

Integration

- Incremental integration
- Dependencies between modules (coupling)
  - Stubs
  - Drivers
- Uniform distribution of intelligence in modern OO systems
- Designing OO systems for integration

Maintenance

- Housekeeping
- Adaptive maintenance
- Perfective maintenance
- Software phasing-out
  - Perfective maintenance cannot help
  - Unpredictable effects of changes
  - Lack of documentation
  - Platform to be replaced
Project planning in lifecycle

- “Fixed” constraints
  - Time
  - Money
- Moving target
- Project feasibility
  - Operational
  - Economic
  - Technical
  - Schedule
- Project plan

Metrics in lifecycle

- Part of project and process management
- **Metrics** = measurements
- Measuring *software products* (quality and complexity)
- Measuring *development products*
  (process metrics)

Testing in lifecycle

- Spans the lifecycle
- Test plans and test cases
- Traceability to use cases
- SQA
- Test types:
  - Formal reviews (walkthroughs, inspections)
  - Execution-based
  - Incremental (regression) testing
  - Capture-playback tools
Software development approaches

The past
- Procedural programs
- Deterministic execution
- Program in control

The present
- Interactive program
- Event-driven execution
- Objects
- Structured vs. Object-Oriented

Structured approach

Modeling techniques
- DFD
- ERD

Problems
- Sequential and transformational
- Inflexible solutions
- No reuse

Object-Oriented approach

Data-centric
- Event-driven
- Addresses emerging applications
- Addresses application backlog
- Follows iterative and incremental process

Problems
- Semantic gap in case of relational database implementation
- Project management
- Solution complexity
Summary

- **Nature of software development – craft or even art**
- **The triangle for success** – stakeholders, process, modeling language and tools
- **System planning** – SWOT, VCM, BPR, ISA
- **The software development lifecycle**
- **Structured development approach**
- **Object-oriented development approach**