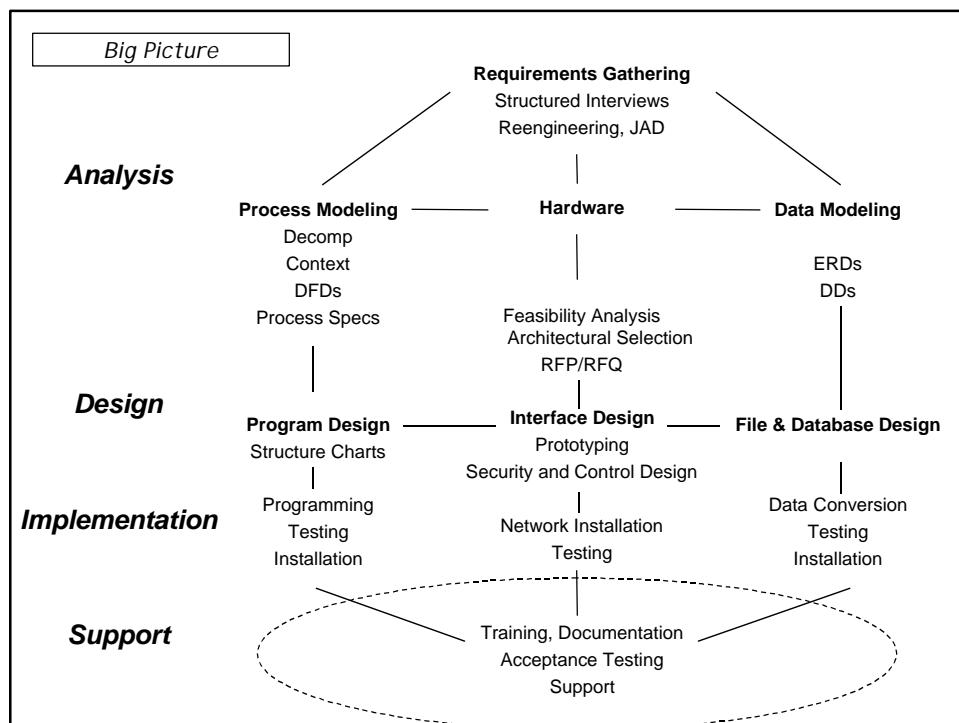


## Beyond Design

Documentation  
Installation  
Support



## Wrap-up Documentation

- for client/users
  - reference manual
  - task guide
  - training guide
- for operations staff
  - operations manual
- for technical staff
  - technical manual
  - program manual

## Client/user Documentation

- reference manual
  - organized by system function or command
  - for experienced users
- task guide
  - “how to” orientation
  - organized by business function
- training guide
  - overview, tutorial
- build into application as much as possible

## Operations Documentation

- operations manual
  - startup/shutdown
  - backup procedures
  - focus on restart and recovery

## Technical Documentation

- technical manual
  - general design spec
  - system architecture
  - interface specs (with other systems)
- program documentation
  - design specs
  - source code
  - test data
  - change log

## System Installation

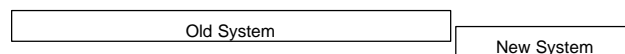
- Involves “cutting over” to new application
- Some people call this “implementation”
- Must be extremely well prepared
- Can make or break a project

## Installation (cutover) Approaches

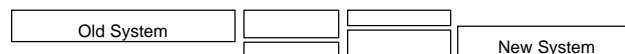
### Parallel



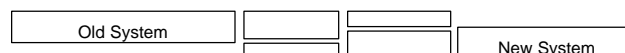
### Plunge



### Phased



### Pilot



## System Installation (4Ps)

	Risk	Cost	Duration
Plunge	High	Low*	Short*
Parallel	Low	High	Long
Pilot	Med	Med	Variable
Phased	Med	Med	Variable

Notes: ‘\*’ assumes installation is successful :-)  
always consider combining approaches

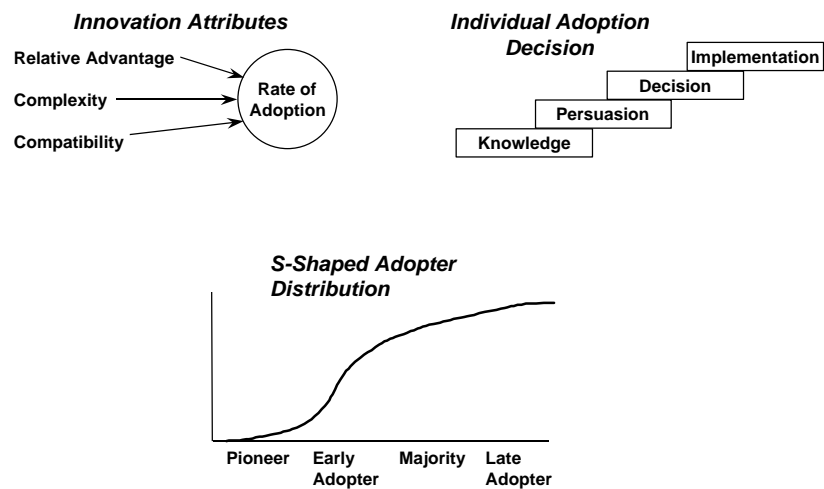
## Preparing Users for Change

- Behavioral issues will always be important
  - system should be owned by user
  - establish need for change
  - establish need for system
- Structured analysis and design tends to increase user participation in SDLC
  - structured interviews
  - JAD
  - prototyping

## User Involvement During Implementation Phase

- designing and procuring forms
- preparing test data
- planning for physical equipment in user areas
- conducting user training
- converting files and databases
- conducting systems and acceptance tests

## Innovation Diffusion Theory



## Support Phase

“Systems support is the ongoing maintenance and upkeep of a system after it is placed into day-to-day operation.”

## Support Phase Terminology

- maintenance vs. enhancement (textbook)
- corrective vs. perfective (research)
- bugs vs. features (practice)

### Support Phase: Maintain System

- Define & Validate Problems
- Benchmark Problem Components
- Understand Problem Components
- Edit & Test Problem Components
- Update Documentation

### The Problem

- “Most mature IS organizations devote approximately 3/4 of their analyst and programmer resources to support.”
- support cost = 2x to 5x development cost
- don't shortcut SDLC to save development cost



## Software Maintenance

- There is no such thing as a “free” change to a software application
  - all changes must be done with extreme care and regard for good software engineering practice
  - all changes to software must lead to review of overall structure of resulting system
  - all changes must be fully documented at ALL relevant levels of documentation

» Macro, 1990

## Current Problem

- 70 billion lines of COBOL code in use in US
- This code is depreciating because of new and changing business conditions
- Hardware is constantly improving, but this makes software maintenance more difficult

### Why Maintenance Costs are High

- Obsolete programming techniques were used
- Documentation is obsolete or absent
- Many programmers made modifications
- Old versions of languages were used

### Why Maintenance Costs are High (cont'd)

- Languages were mixed within the program
- Unskilled programmers made enhancements
- Architecture changes are required
- Maintenance is cumulative

### Change or Rewrite?

- “Even a strong original program structure slides progressively into a spaghetti affair unless definite effort is expended.”
- substantive changes (to modify function)
- consequential changes (to preserve structure)
- rewrite as  $s + c$  tends toward 100%

### Support Phase: Recover System

- determine cause of problem
- some options include
  - simple reboot of PC
  - cancel/reinitialize online session by sysops
  - recover database by data administration
  - recover network by network administration
  - repair/replace hardware by vendor service staff

### Support Phase: Assist Client/User

- Level of support depends on nature of app
- Some options include
  - help line (phone in)
  - help desk (drop in)
  - bug busters (on site)
  - 7x24 service

### Support Phase: Enhance & Re-engineer

- Analyze change request
  - route to analysis, design, or implementation phase
- Write simple, new programs
  - e.g. user developed reports
- Restructure files & databases
- Analyze program support costs
- Reengineer & test programs

## Re-engineer and Test Programs

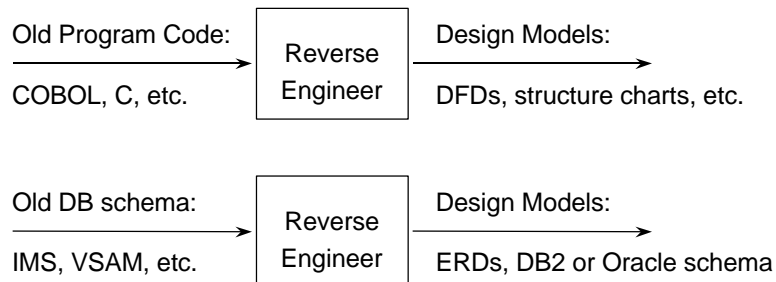
- code reorganization
  - group or separate to improve coupling & cohesion
- code conversion
  - convert language from one version/dialect to another
- code slicing
  - factor out reusable code into library modules
- all to do what should have been done 1st time!

## Versions, Releases, and Fixes

- versions
  - involve major functional or technical changes
  - must test as initial implementation (ver 2.0)
- releases
  - involve minor modifications at regular intervals
  - test with original test data sets (ver 2.1)
- fixes
  - involve bugs which cannot wait for next release
  - test system affected components (ver 2.1a)

## Reverse Engineering

- Moving backwards through SDLC



- Compare with “forward engineering”

## *Model of Organizational Change*

