Session 9

Software Implementation and CASE

Review

- Maintenance activities involve making enhancements to the software products developed in the earlier stages of the life cycle, adapting products to new environments and correcting problems.
- Maintenance activities account for a large portion of the system life cycle costs, even as much as 60-70 percent of the total costs set aside for the project.
- Corrective maintenance involves testing and diagnosis, then designing and making changes to get the system to do what it was expected to do.
- Perfective maintenance involves designing and enhancing the system to perform better than it was originally expected to do.
- Adaptive maintenance is a subset of perfective maintenance and comes into picture when we wish to take advantage of the latest technological advancements.

Review Contd…

- Preventive maintenance is also called reengineering.
- Issues related to software maintenance include:
  - Lack of documentation
  - Dynamic personnel
  - Motivation or morale
  - Patchy code
  - Outdated technology
  - Round-the-clock operations
- Maintenance metrics can be calculated based on the following factors:
  - System availability
  - Maintenance turnaround
  - Productivity

Objectives

- *Analyze why Implementation is an important phase of SDLC*
- *Discuss the various activities of Implementation Phase*
- *Discuss the various Implementation Strategies*
- *Discuss the need for Contingency Planning and Post Implementation Maintenance*
- *Learn the meaning of CASE and its evolution*
- *Discuss the classification of CASE*
- *Correlate the use of CASE to software engineering concept*
- *Discuss CASE Toolkits, CASE Methodology companion, CASE Workbench, Repository features*

Activities

- Creating an Installation Plan
- Implementing physical procedure
- Preparing and Converting data
- Conducting User Training
- Running the system

Installation Plan

- Purchase of hardware
- Preparation of site
- Purchase of environmental software
- Installation of developed software
- Training

Physical Procedure

- An organization will follow simultaneously with the automated system
- Spells the transition from old to the new system
- Requires handling of user training

Data Conversion

- List all existing files to be converted
- List new files to be created and data required for it
- List all new documents and procedures that go into use during conversion
- Identify controls for the conversion process
- Prepare conversion schedule
- Assign responsibility for each task

User training

- Two levels of training
  - Training for systems operations
  - Training for users of the system

Training for system operators

- How the equipment has to be used
- Minor troubleshooting
- Maintenance activities and their frequency

Training for end-users

- Using the equipment
- Using the application
- Using the data
- Adding, deleting and editing of records
- Retrieving information from the system
- Utilizing the system generated information
- Elementary troubleshooting

Parallel Run

- The running of old and new systems together

- Factors to be considered
    - No major faults in the package
    - Important figures tally
    - All modules are used and found satisfactory
    - Fair amount of confidence amongst users in being able to use the software

- Users approve implementation when they are fully satisfied

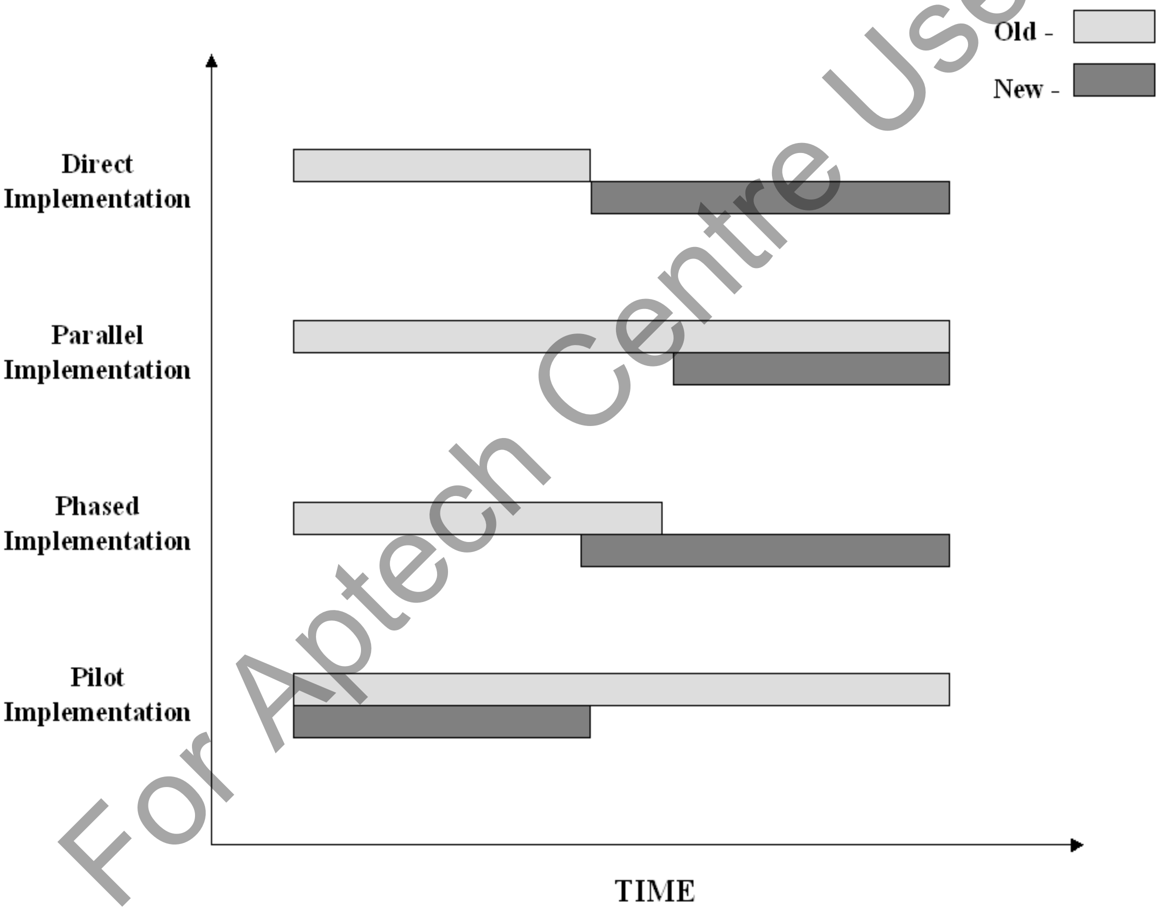
Post Implementation Review

- Has the information system reduced/ increased the operating cost of application
- Does the system provide accurate and up-to-date information in desirable formats?
- Has the new system improved the way systems were operated earlier?
- Does the new system need more or less number of users/ operators than before?
- Has the new system affected the procedures of the organization?
- Has the new system improved productivity ?
- Has the new system improved service to customers?

Implementation Strategies

- Parallel run
- Direct Cutover
- Phased Approach
- Pilot Approach

Implementation Strategies Contd…



Implementation Strategy – Choosing Factors

- Nature and size of application
- Type of organization
- Existence of an internal EDP environment
- Number of personnel involved in usage of software
- Data volumes
- Location spread of the application
- Availability of certain tools
- Manpower availability
- Criticality of the application

Data Conversion

- When data is available as part of some existing application
- When application is being automated for the first time
- When volumes of data are very high

Contingency Planning

- Another backup machine
- Leased time
- Service from another location
- Manual System

Post Implementation Maintenance

- Troubleshoot problems
- Modify the system when and if bugs are discovered
- Modify the system for changed requirements of the user

Documentation

- Installation manual
- Operations manual
- User manual

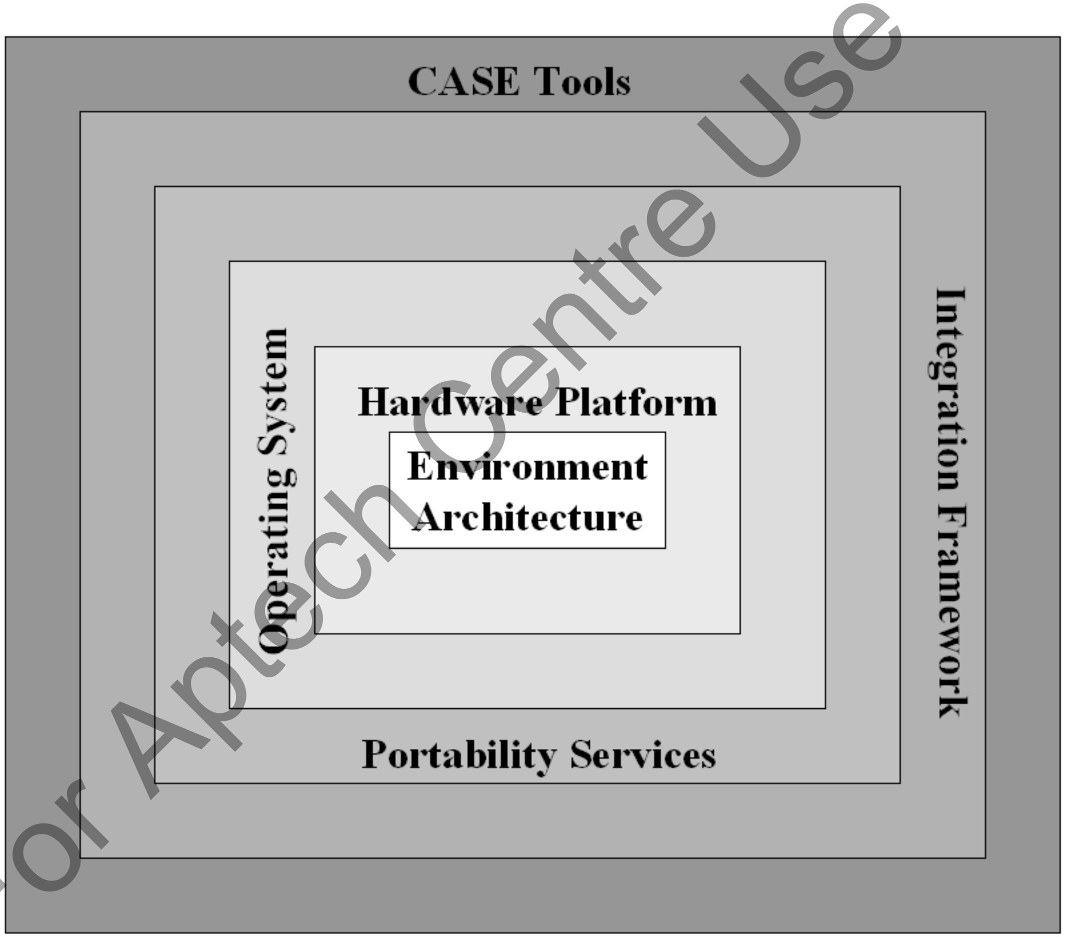
CASE

- Computer Aided Software Engineering
- Enables people working on a software project to store data about a project
  - Plan and schedule
  - Track the progress
  - Analyze and store data about user requirements
  - Store the design of a system
- Aids a software engineer to maintain and develop software
- Automates the principles of software engineering

CASE Tools

- Production-process support technology
- Process Management Technology
- Meta-case Technology

Complete CASE Environment



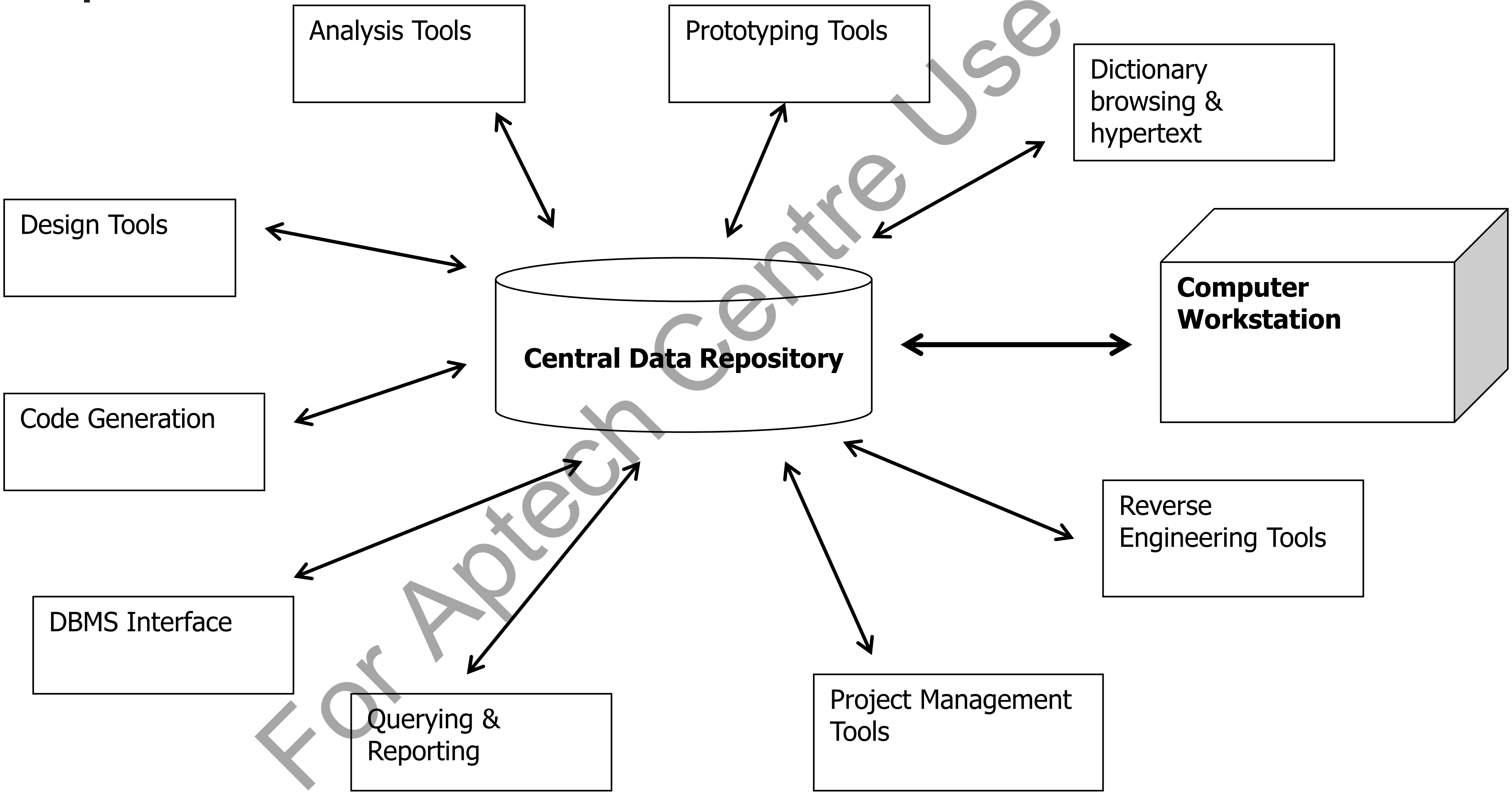
CASE Tools - Need

- Size of projects
- Requires systematization of development process
- User requirements
- Supports principles reusability, flexibility, maintainability of programs
- The Development process needs clear understanding and documentation
- ‘Zero defect’ software

Evolution of CASE

- Early 1980s
  - Had diagramming capabilities
- Late 1980s
  - Development of tools that could store diagram information
  - Perform cross-diagram checks through a repository
- The 1990s
  - Integration of project management and system lifecycle activities
  - Included automatic code generation
- Late 1990s
  - Integrated CASE framework where different CASE vendor products can be integrated

CASE Environment



CASE Terminologies

- CASE Technology
- CASE Tool
- CASE Toolkit
- CASE Workbench
- CASE Methodology
- CASE Methodology Companion
- CASE Hardware

CASE Tools - Classifications

- Tool
- Toolkit
- Methodology Companion
- Workbench

CASE Tools - Types

- Diagramming
- Prototyping
- Simulation
- Support
- Quality Assurance
- Software Configuration Management
- Re-engineering tools
- Reverse Engineering
- Documentation

CASE Toolkits

- Analysis
- Design
- Programming
- Project Management
- Maintenance

CASE Workbench

- Consists of tools that cover and integrate entire SDLC
- Offers a choice of tools and techniques
- Ensures that the output of one phase successfully becomes the input for another phase

CASE - Advantages

- Takes project from inception to implementation
- Assures smooth flow from one stage to another
- Error checking is easier
- A CASE workbench contains the 'information repository'

Summary

- Implementation is a very important phase of SDLC.
- There are no standard methodologies for implementation.
- The important activities of the implementation phase are:
  - Drawing up an Installation Plan
  - Implementing of Physical Procedures
  - Data Preparation and Conversion
  - Conducting User Training
  - Parallel run of the system
  - Seeking Implementation Approval
- The choice of an appropriate implementation strategy is critical to the successful implementation of any project.

Summary Contd…

- The implementation is usually followed by post-implementation maintenance phase.
- CASE automates the process of software development.
- The essential components of a CASE are tools, toolkits, methodology companions and workbenches.
- The important toolkits being commonly used today are:
    - Analysis toolkits
    - Design toolkits
    - Programming toolkits
    - Project Management toolkits
    - Maintenance toolkits
- CASE is in the process of becoming an integral part of every software organization.