SPECIALIZED SOFTWARE MODELS

PRESENTED BY
Dr. S. Peerbasha

SPECIALIZED SOFTWARE MODELS

1. COMPONENT BASED DEVELOPMENT
2. THE FORMAL METHOS MODEL
3. THE ASPECT ORIENTED SOFTWARE DEVELOPMENT

COMPONENT BASED DEVELOPMENT

✓The component based development model incorporates many of the characteristics of the spiral model.

✓It is evolutionary in nature, Specialized process model demanding an iterative approach to the creation of software.

✓However, the component based development model constructs applications from prepackaged software components.

COMPONENT BASED DEVELOPMENT

- ✓ Modeling and construction activities begin with the identification of candidate components.
- ✓ These components can be designed as either conventional software modules or object oriented classes or packages of classes.

COMPONENT BASED DEVELOPMENT

1. The component based development specialized process model incorporates the following steps:-
2. Available component based products are researched and evaluated for the application domain in question.
3. Component integration issues are considered.
4. A software architecture is designed to accommodate the components.
5. Components are integrated into the architecture.
6. Comprehensive testing is conducted to ensure proper functionality.

THE FORMAL METHODS MODEL

- The formal methods model encompasses a set of activities that leads to formal mathematical specification of computer software.
- Formal methods enable you to specify, develop, and verify a computer based system by applying a rigorous, mathematical notation.
- A variation on this approach, called clean room software engineering.

THE FORMAL METHODS MODEL

- Ambiguity, incompleteness, and inconsistency can be discovered and corrected more easily, but through the application of mathematical analysis.
- When formal methods are used during design, they serve as a basis for program verification and therefore enable you to discover and correct errors that might otherwise go undetected.

THE FORMAL METHODS MODEL

Draw Backs:

- The development of formal models is currently quite time consuming and expensive.
- Because few software developers have the necessary background to apply formal methods, extensive training is required.
- It is difficult to use the models as a communication mechanism for Technically unsophisticated customers.

ASPECT ORIENTED SOFTWARE DEVELOPMENT

- AOSD defines “aspects” that express customer concerns that cut across multiple system functions, features, and information.

•*Often referred to as aspect oriented programming (AOP), is a relatively new software engineering paradigm that provides a process and methodological approach for* ***defining, specifying, designing, and constructing aspects.”***

•*Grundy provides further discussion of aspects in the context of what he calls aspect oriented component engineering (AOCE):*