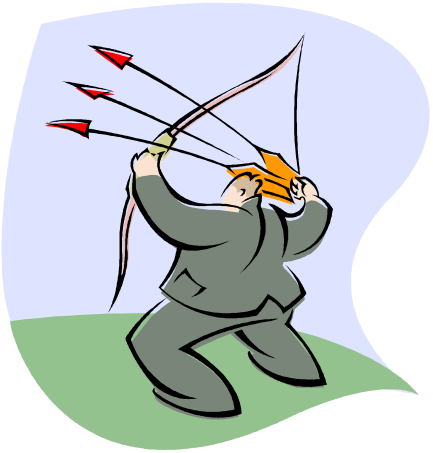
Software Engineering
Tools and Practices

Course Outcomes – why you are taking this class

- Become familiar with the **practices** used by Software Engineers for creating software applications
- Become familiar with the various modern **tools** used by Software Engineers for creating applications

- So that you can get a job and work in a team

What are Software *practices?*

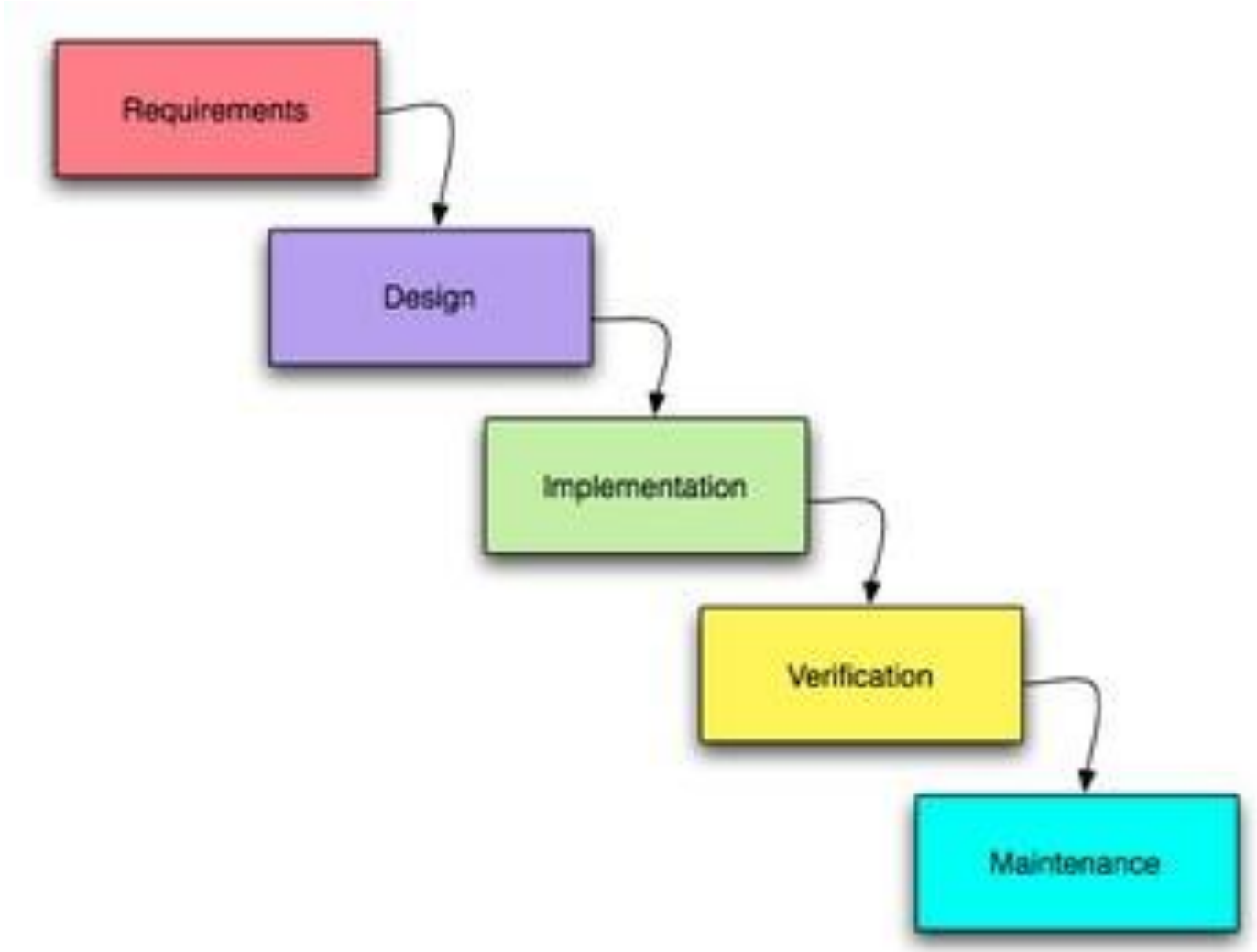


A software *process* defines the steps you take develop (good) software

- A software *process* typically defines **phases** (or stages) and steps you take within each phase to develop (good) software

- What phases can you think of?

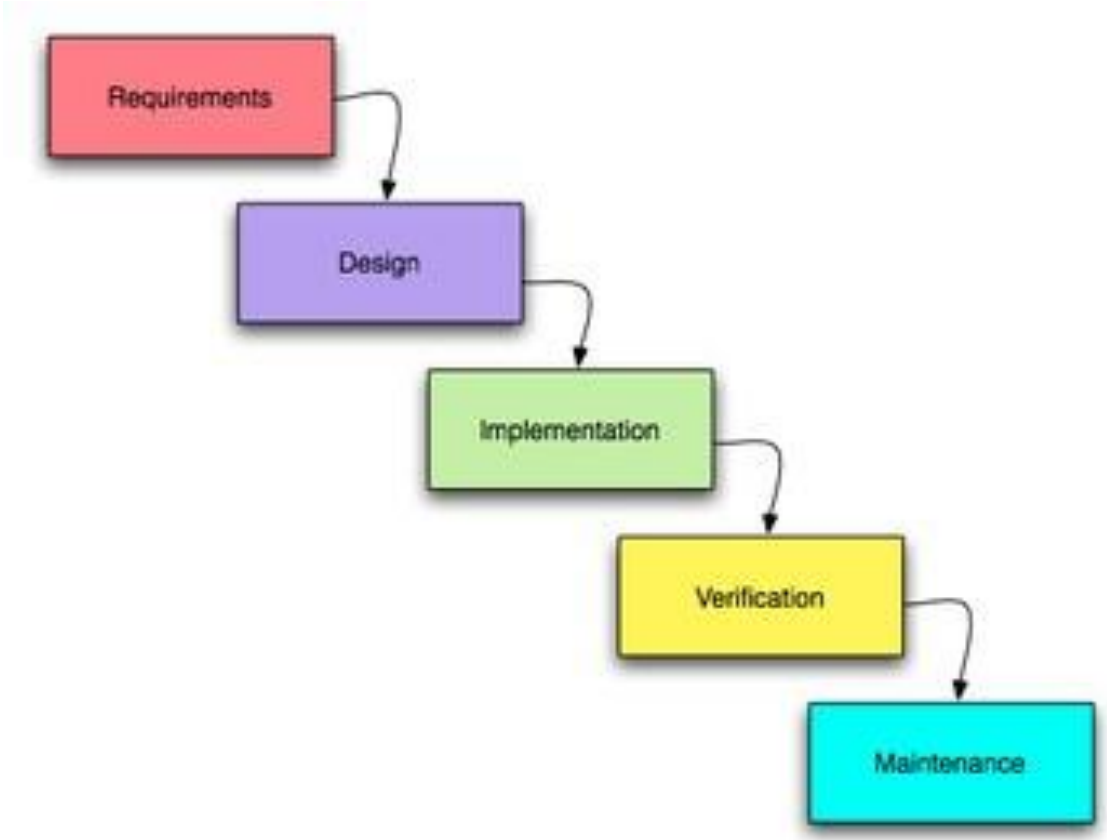
A Software Life Cycle

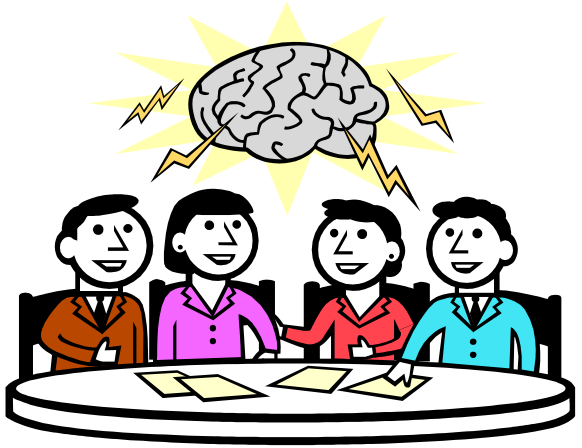


Software *practices* are specific things you do as part of the software development *process*

- That is, **practices** are activities that implement the **process**

Software development *practices* that might take place in each phase below





What are Software *tools?*

Introduction

- Software Engineering Methods are intended to make software engineering more systematic .
- Software development environment are the computer based tools that are intended to assist the software life cycle processes.
- Tools are often designed to support particular software engineering methods.

Software Engineering Tools and Methods

- Software Engineering tools represent
- a set of management and technical tools to support software development

- usually integrated in a coherent framework
- Methods usually provide a notation and vocabulary, procedures for performing
- identifiable tasks, and guidelines for checking both the process and the product.

Software Engineering Tools

- Some particular Software Engineering tools
  - Software Design Tools
  - Software Construction Tools
  - Software Requirement Tools
  - Software Testing Tools
  - Software Maintenance Tools
  - Software Configuration Management Tools

- Software Engineering Management Tools
- Software Engineering Process Tools
- Software Quality Tools
- Infrastructure Support Tools
- Miscellaneous Tools Issues

Software Requirements Tools

- These Tools are classified into two categories
  1. Requirements modeling tools 2. Requirement traceability tools
  - Requirements modeling tools
  - These tools are used for analyzing, specifying, and validating software requirements

- Requirement traceability tools
  - Since they are also relevant in other life cycle processes, they are presented separately from the requirements modeling tools.

Software Design Tools

- Creating Software Designs
- Checking Software Designs.

Software Construction Tools

- These tools are concerned with
- Production of the program representation
- Translation of the program representation
- These tools are
- Program Editors
- Creation and Modification Of Programs
- Compiler and Code generators
- Non Interactive translators of source code
- Interpreters
- Provide Software execution through emulation
- Debugger
- Support Construction Process

Software Testing Tools

These tools are categorized as

- Test generators
- assist in the development of test cases.
- Test execution frameworks
- enable the execution of test cases in a controlled environment where the behavior of the object under test is observed.
- Test evaluation tools
- assessment of the results of test execution,
- helping to determine whether or not the observed behavior conforms to the expected behavior .
- Test management tools
- Managing software testing process.

- Performance analysis tools
- measuring and analyzing software performances

Software Maintenance Tools

These tools are categorized as

- Comprehension tools
- assist in the human comprehension of programs. Ex: Animators
- Reengineering tools
- examination and alteration of the subject software to reconstitute it in a new form
- Reverse engineering tools
- Assist the process by working backwards from an existing product

Software Configuration Management Tools

These tools are categorized as

- Tracking
  - used in connection with the problem-tracking issues associated with a particular software product
  - Version management
  - involved in the management of multiple versions of a product
  - Release tools
  - used to manage the tasks of software release and build

Software Engineering Management Tools

These are categorized as

- Project Planning and Tracking
- used in software project effort measurement and cost estimation, as well as project scheduling
- Risk Management
- used in identifying, estimating, and monitoring risks.
- Measurement
- assist in performing the activities related to the software measurement program

Software Engineering Process Tools

These are categories as

- Process modeling tools
- These tools are used to model and investigate software engineering processes.
- Process management tools
- provide support for software engineering management.
- Process-centered software engineering environments
- incorporate information on the software life cycle processes and guide and monitor the user according to the defined process.
- Integrated CASE environments
- Integrated computer-aided software engineering tools or environments covering multiple phases of the software engineering life cycle belong

Software Quality Tools

These are categories as

- Review and Audit tools
- used to support reviews and audits
- Static Analysis tools
- used to analyze software artifacts, such as syntactic and semantic analyzers, as well as data, control flow, and dependency analyzers

Miscellaneous Tool Issues

This are categories as

- Tool Integration Techniques
- Used to make individual tools cooperate
- Integration techniques are applied
- Meta-tools
- Meta-tools generate other tools; compiler-compilers are the classic example.
- Tool Evaluation

- Evolve new tool

Software Engineering Methods

- Methods usually provide a notation and vocabulary, procedures for performing identifiable tasks, and guidelines for checking both the process and the product
- These are categorized as
- Heuristic methods
- dealing with informal approaches.
- Formal methods
- dealing with mathematically based approaches.
- Prototyping methods

- dealing with software engineering approaches based on various forms of prototyping

Heuristic Methods

- These are categories as
  - ➢ Structured methods
    - ✓ The system is built from a functional viewpoint, starting with a high-level view and progressively refining this into a more detailed design.
  - ➢ Data-oriented methods
    - ✓ the starting points are the data structures that a program manipulates rather than the function it performs
  - ➢ Object-oriented methods
    - ✓ The system is viewed as a collection of objects rather than functions
  - ➢ Domain-specific methods
    - ✓ includes specialized methods for developing systems which involve real-time, safety, or security aspects

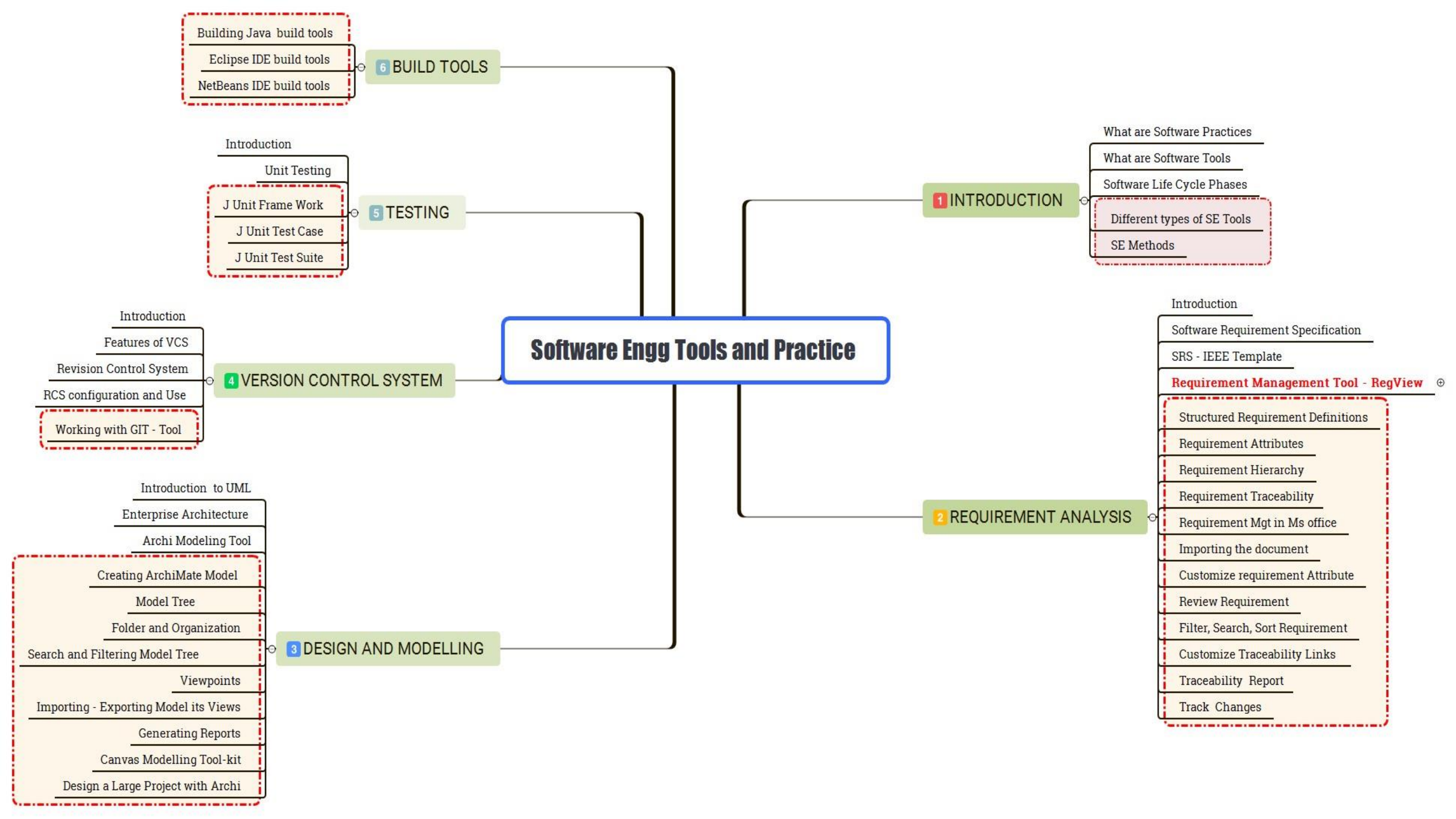
Formal Methods

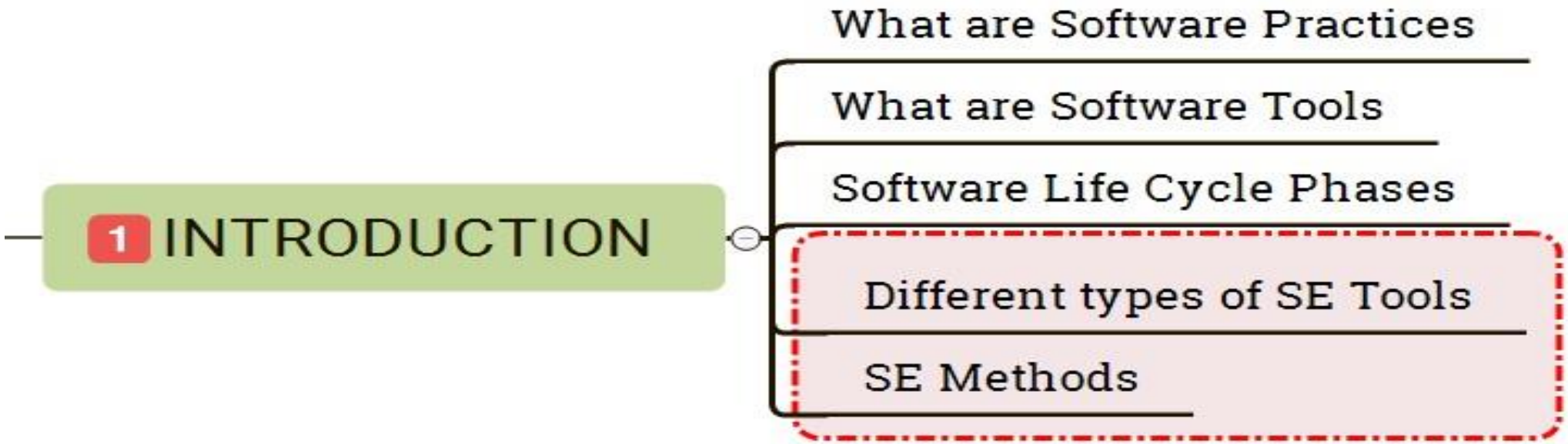
These are categorized as

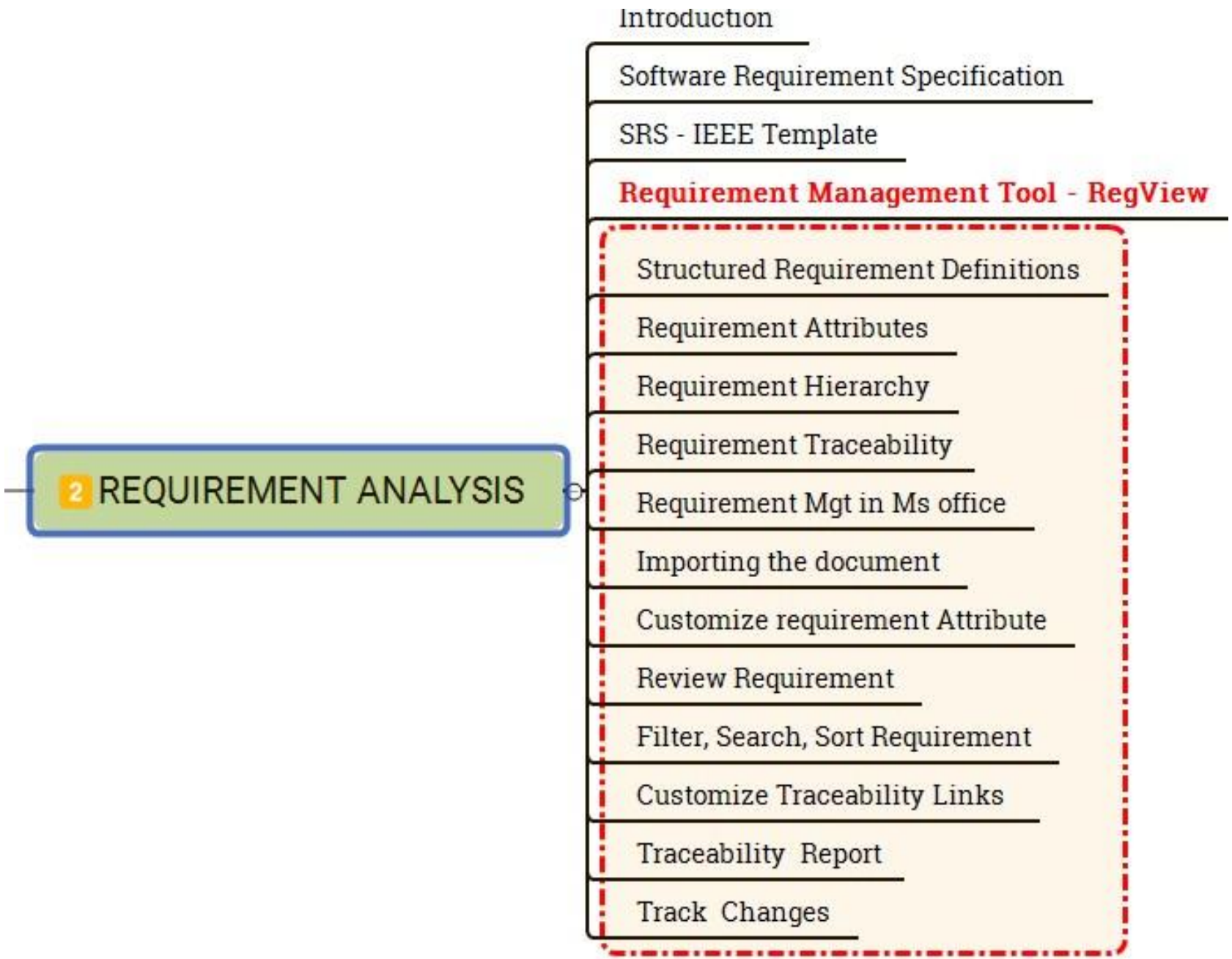
- Specification languages and notations
- This topic concerns the specification notation or language used. Specification languages can be classified as model-oriented, property-oriented, or behavior-oriented
- Refinement
- This topic deals with how the method refines (or transforms) the specification into a form which is closer to the desired final form of an executable program.
- Verification/Proving properties:
- This topic covers the verification properties that are specific to the formal approach, including both theorem proving and model checking

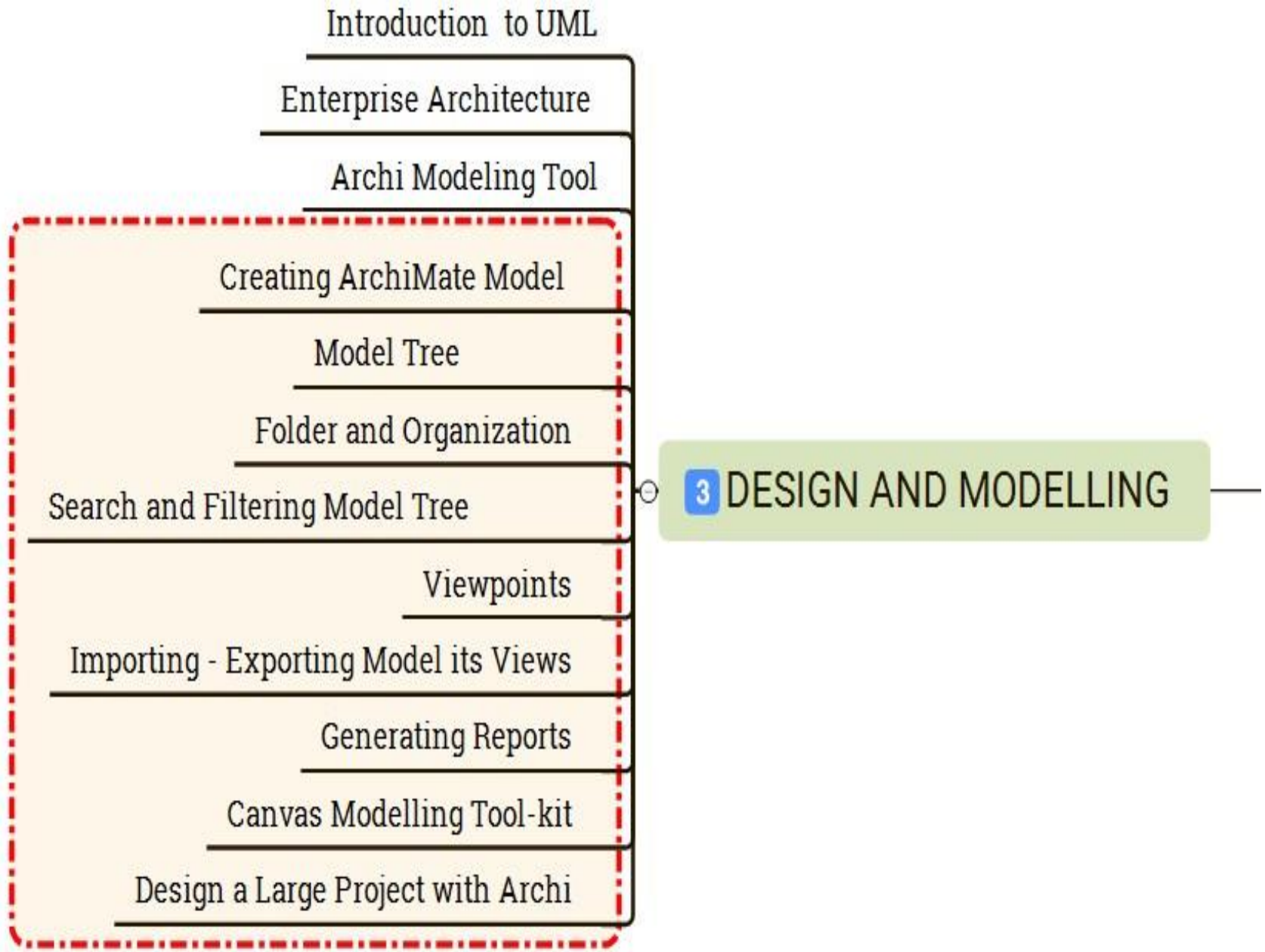
Prototyping Methods

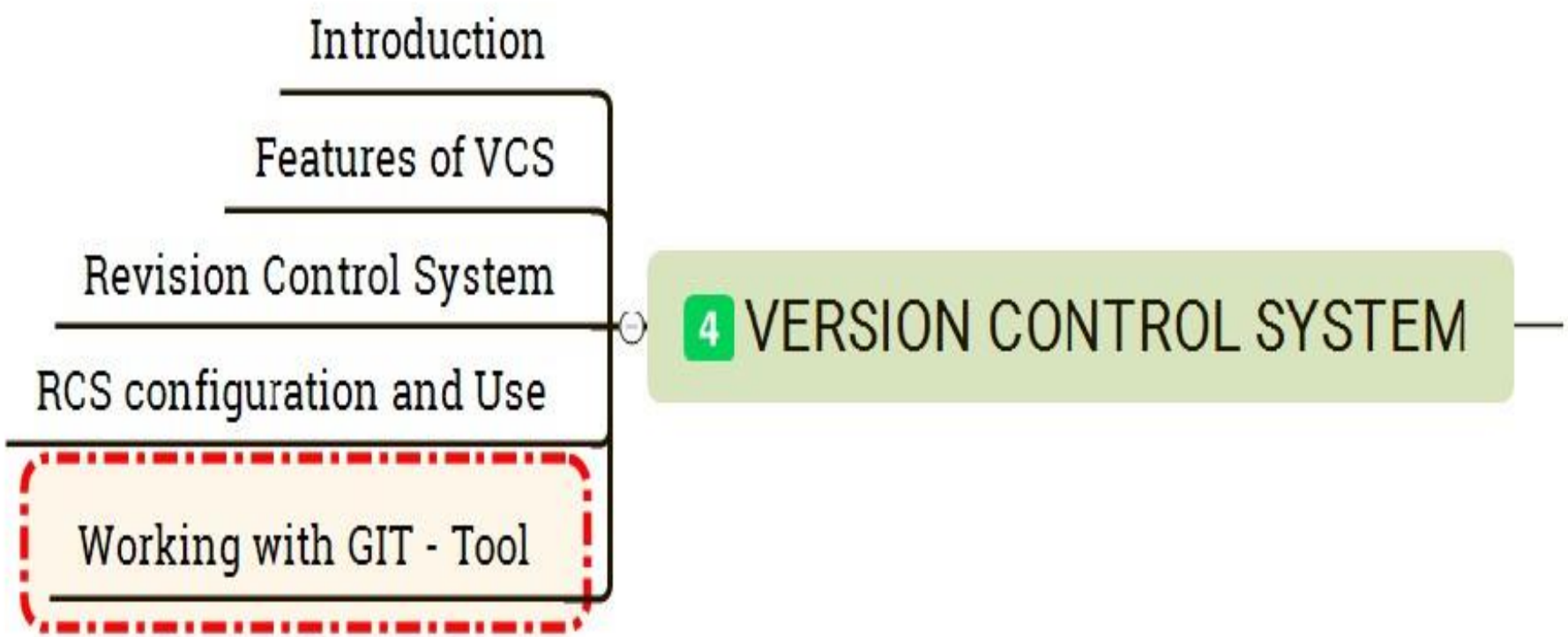
- These are categorized as
- Prototyping styles
- The prototyping styles topic identifies the various approaches: throwaway, evolutionary, and executable specification
- Prototyping targets
- Examples of the targets of a prototyping method may be requirements, architectural design, or the user interface
- Prototyping evaluation techniques
- This topic covers the ways in which the results of a prototype exercise are used.s

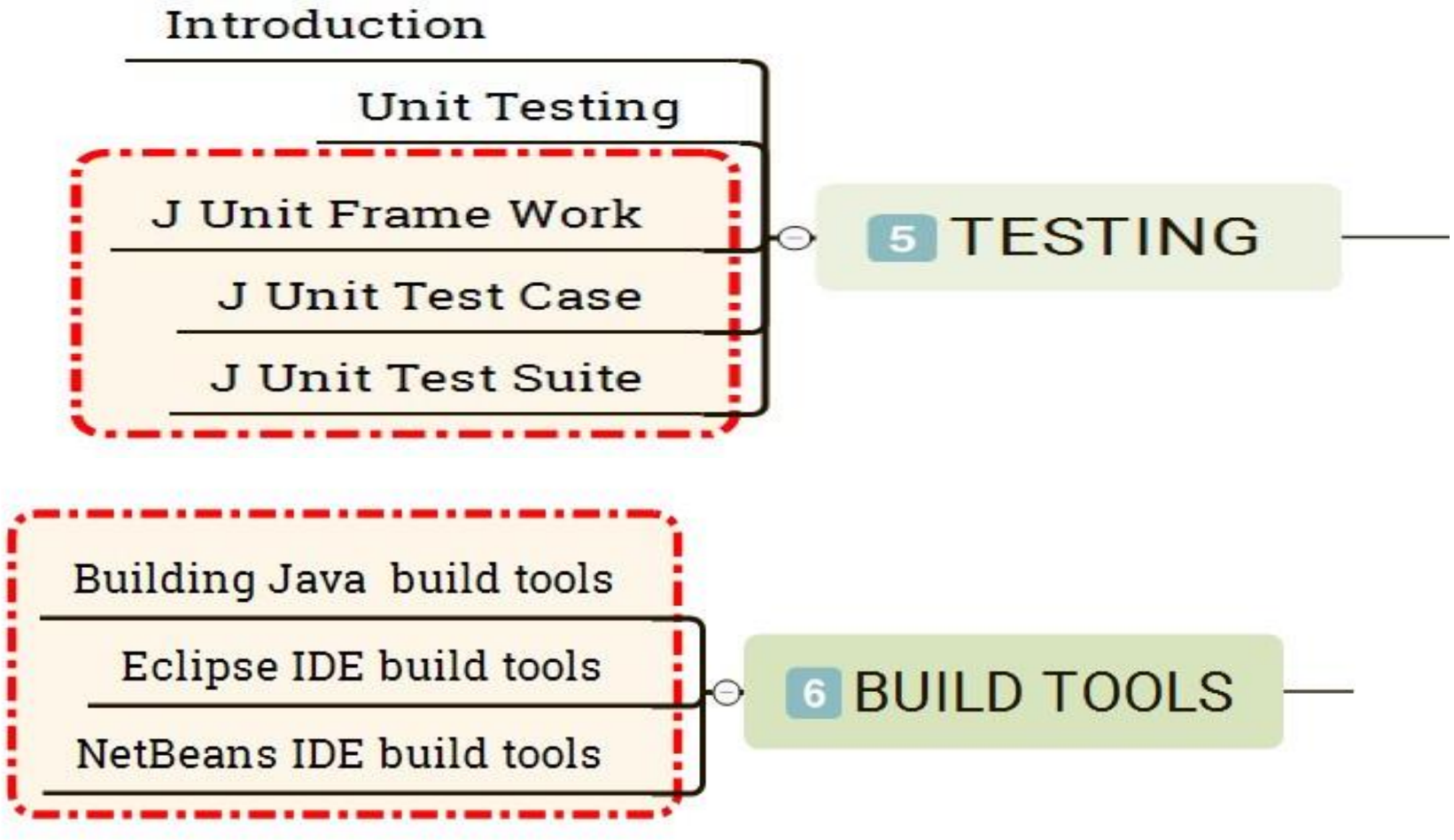

Software Engg Tools and Practice
1 INTRODUCTION
What are Software Practices
What are Software Tools
Software Life Cycle Phases
Different types of SE Tools
SE Methods
2 REQUIREMENT ANALYSIS
Introduction
Software Requirement Specification
SRS - IEEE Template
Requirement Management Tool - RegView
Structured Requirement Definitions
Requirement Attributes
Requirement Hierarchy
Requirement Traceability
Requirement Mgt in Ms office
Importing the document
Customize requirement Attribute
Review Requirement
Filter, Search, Sort Requirement
Customize Traceability Links
Traceability Report
Track Changes
3 DESIGN AND MODELLING
Introduction to UML
Enterprise Architecture
Archi Modeling Tool
Creating ArchiMate Model
Model Tree
Folder and Organization
Search and Filtering Model Tree
Viewpoints
Importing - Exporting Model its Views
Generating Reports
Canvas Modelling Tool-kit
Design a Large Project with Archi
4 VERSION CONTROL SYSTEM
Introduction
Features of VCS
Revision Control System
RCS configuration and Use
Working with GIT - Tool
5 TESTING
Introduction
Unit Testing
J Unit Frame Work
J Unit Test Case
J Unit Test Suite
6 BUILD TOOLS
Building Java build tools
Eclipse IDE build tools
NetBeans IDE build tools


1 INTRODUCTION
What are Software Practices
What are Software Tools
Software Life Cycle Phases
Different types of SE Tools
SE Methods


2 REQUIREMENT ANALYSIS
Introduction
Software Requirement Specification
SRS - IEEE Template
Requirement Management Tool - RegView
Structured Requirement Definitions
Requirement Attributes
Requirement Hierarchy
Requirement Traceability
Requirement Mgt in Ms office
Importing the document
Customize requirement Attribute
Review Requirement
Filter, Search, Sort Requirement
Customize Traceability Links
Traceability Report
Track Changes


3 DESIGN AND MODELLING
Introduction to UML
Enterprise Architecture
Archi Modeling Tool
Creating ArchiMate Model
Model Tree
Folder and Organization
Search and Filtering Model Tree
Viewpoints
Importing - Exporting Model its Views
Generating Reports
Canvas Modelling Tool-kit
Design a Large Project with Archi


4 VERSION CONTROL SYSTEM
Introduction
Features of VCS
Revision Control System
RCS configuration and Use
Working with GIT - Tool


Introduction
Unit Testing
J Unit Frame Work
J Unit Test Case
J Unit Test Suite
5 TESTING
Building Java build tools
Eclipse IDE build tools
NetBeans IDE build tools
6 BUILD TOOLS