Faculty of Computers and Information
BSc. Software Engineering Program

Dr. Lamia Abo Zaid

د. لمياء أبوزيد

Based on slides by © Tripathy & Naik

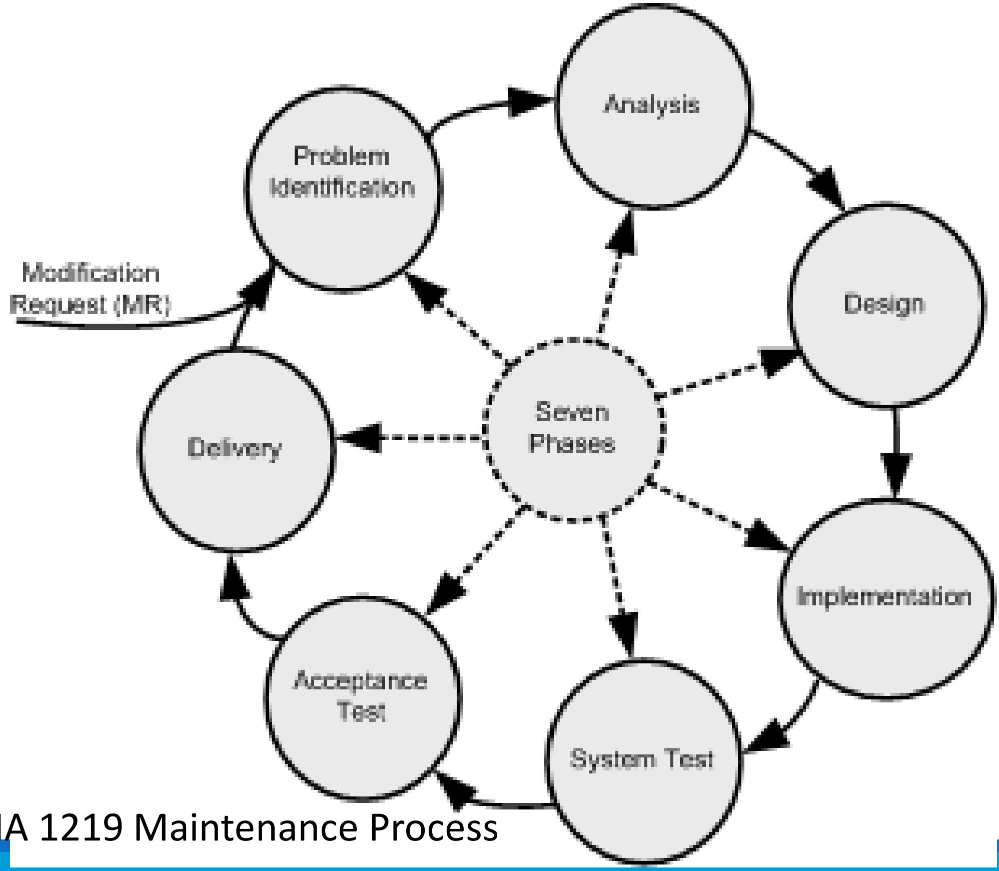
Software Evolution : TOC

1. Introduction to Software Evolution
2. Taxonomy of Software Maintenance and Evolution
3. Evolution and Maintenance Models
4. Reuse and Domain Engineering
5. Program Comprehension
6. Impact Analysis
7. Refactoring
8. Reengineering
9. Legacy Information Systems

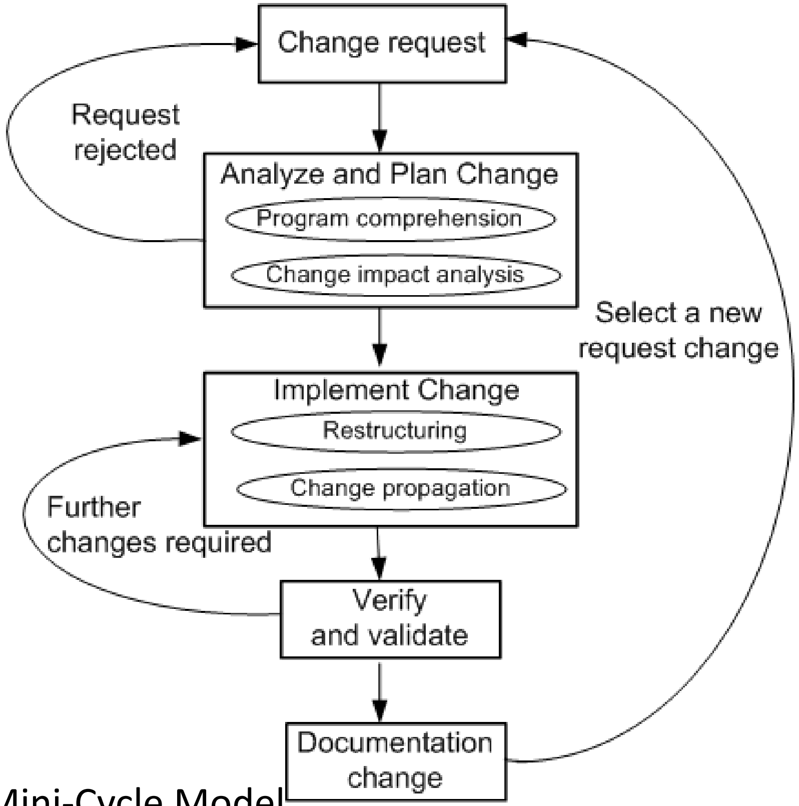
Impact Analysis

- ❑ Impact analysis is a tool for controlling change, and thus for avoiding deterioration
- ❑ The maintenance process is started by performing impact analysis. Impact analysis enables understanding impact of change via identifying the components that are impacted by the Change Request (CR).
- ❑ Impact of the changes are analyzed for the following reasons:
  - to estimate the cost of executing the change request.
  - to determine whether some critical portions of the system are going to be impacted due to the requested change.
  - to record the history of change related information for future evaluation of changes.
  - to understand how items of change are related to the structure of the software.
  - to determine the portions of the software that need to be subjected to regression testing after a change.

Impact Analysis - MR cycle



IEEE/EIA 1219 Maintenance Process



Change Mini-Cycle Model

Traceability

- ❑traceability as the ability to trace between software artifacts generated and modified during the software product life cycle
- ❑Traceability of artifacts between different models is known as external traceability,
- ❑Tracing dependent artifacts within the same model is known as internal traceability.
  - Internal traceability primarily focuses on source code artifacts.
  - classical impact analysis techniques, based on program dependency
    - call-graph-based analysis,
    - static program slicing,
    - dynamic program slicing.

Ripple Effect Analysis

❑A topic related to impact analysis is ripple effect analysis.

❑Ripple effect means that a modification to a single variable may require several parts of the software system to be modified.

❑Analysis of ripple effect reveals what and where changes are occurring.

❑Measurement of ripple effects provides the following information about an evolving software systems:

- between successive versions of the same system, measurement of ripple effect will tell us how the software's complexity has changed.
- when a new module is added to the system, measurement of ripple effect on the system will tell us how the software's complexity has changed because of the addition of the new module.

Ripple Effect Analysis

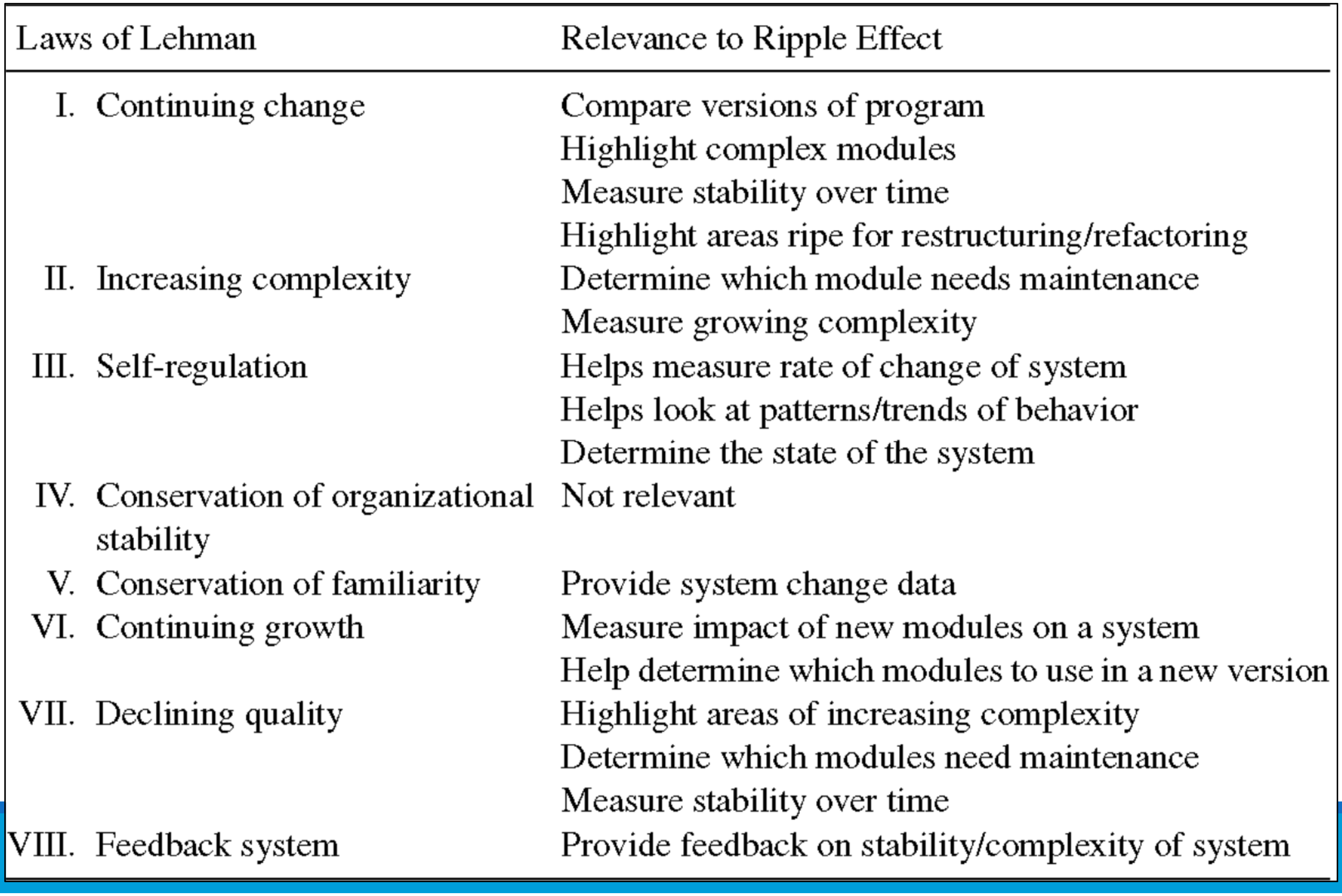
❑ Ripple effect is computed by means of error flow analysis.

- In error flow analysis, definitions of program variables involved in a change are considered to be potential sources of errors. Inconsistency can propagate from those sources to other variables in the program. The other sources of errors are successively identified until error propagation is no more possible

Ripple Effect Analysis

- ❑Stability reflects the resistance to the potential ripple effect which a program would have when it is changed.
- ❑Measurement of ripple effects provides the following information about an evolving software systems:
  - between successive versions of the same system, measurement of ripple effect will tell us how the software's complexity has changed.
  - when a new module is added to the system, measurement of ripple effect on the system will tell us how the software's complexity has changed because of the addition of the new module.

Ripple Effect Related to Lehman’s Laws

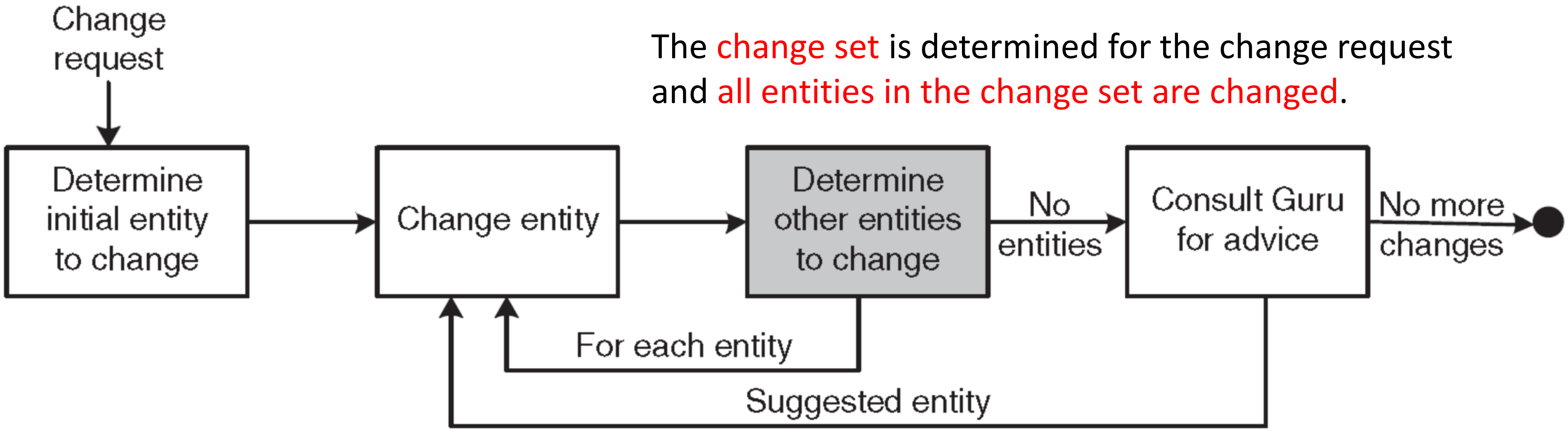


Laws of Lehman	Relevance to Ripple Effect
I. Continuing change	Compare versions of program Highlight complex modules Measure stability over time Highlight areas ripe for restructuring/refactoring
II. Increasing complexity	Determine which module needs maintenance Measure growing complexity
III. Self-regulation	Helps measure rate of change of system Helps look at patterns/trends of behavior Determine the state of the system
IV. Conservation of organizational stability	Not relevant
V. Conservation of familiarity	Provide system change data
VI. Continuing growth	Measure impact of new modules on a system Help determine which modules to use in a new version
VII. Declining quality	Highlight areas of increasing complexity Determine which modules need maintenance Measure stability over time
VIII. Feedback system	Provide feedback on stability/complexity of system

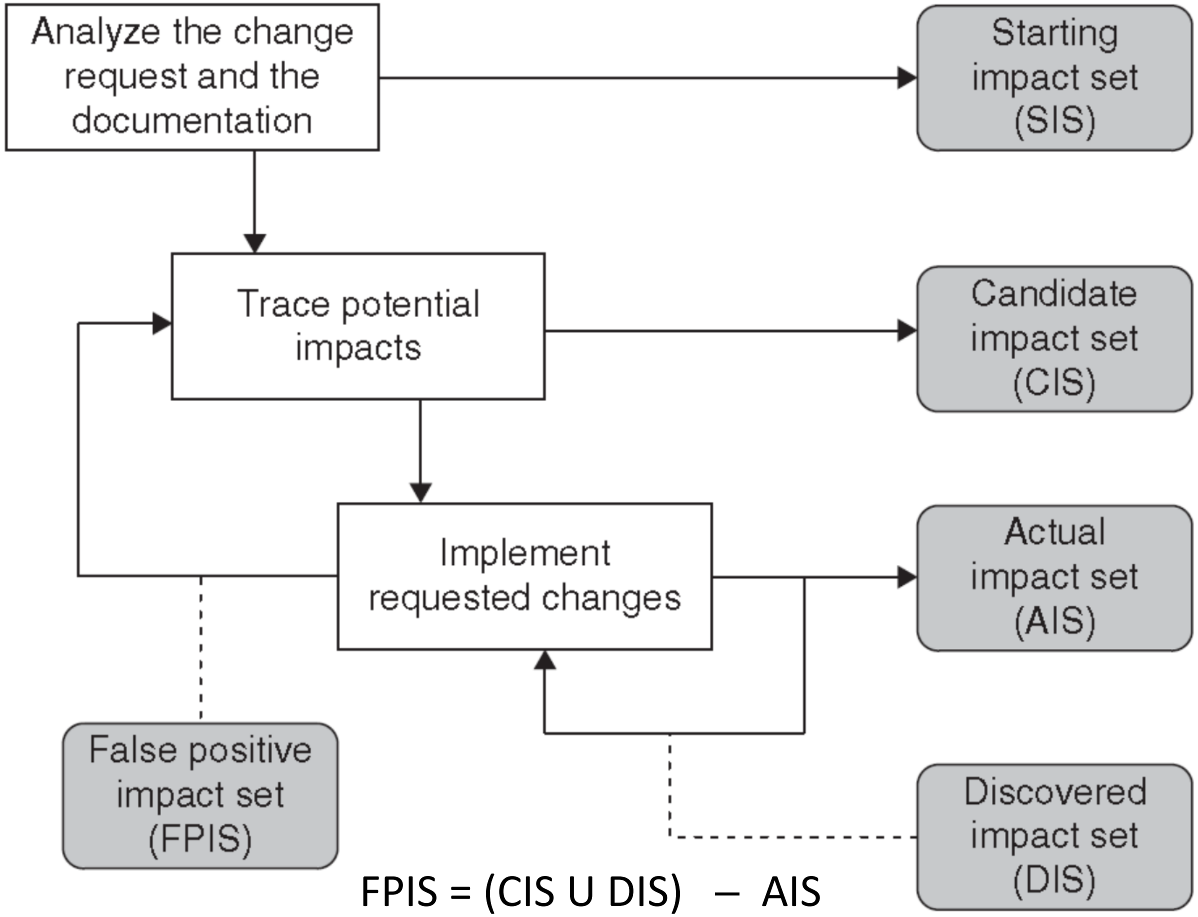
Change Propagation Model

- Change propagation means that if an entity (e.g. a function) is changed, then all related entities in the system are accordingly changed.
- Change propagation model is defined by Hassan and Holt (2006)

The change set is determined for the change request and all entities in the change set are changed.



Impact Analysis Process


Analyze the change request and the documentation
Trace potential impacts
Implement requested changes
False positive impact set (FPIS)
Starting impact set (SIS)
Candidate impact set (CIS)
Actual impact set (AIS)
Discovered impact set (DIS)

The initial set of objects (or components) presumed to be impacted by a software CR

The set of objects (or components) estimated to be impacted according to a certain impact analysis approach

The set of objects (or components) actually changed as a result of performing a CR

the set of new objects (or components), not contained in CIS, discovered to be impacted while implementing a CR.

FPIS = (CIS U DIS) – AIS

Impact Analysis Process

❑In the process of impact analysis it is important to minimize the differences between AIS and CIS, by eliminating false positives and identifying true impacts.

❑to evaluate the impact analysis process two traditional information retrieval metrics: recall and precision are used:

- **Recall:** represents the fraction of actual impacts contained in CIS, and it is computed as the ratio of |CIS ∩ AIS|/|AIS|.
  - □The value of recall is 1 when DIS is empty.
- **Precision:** represents the fraction of candidate impacts that are actually impacted, and it is computed as the ratio of |CIS ∩ AIS| / |CIS|.
  - □For an empty FPIS set, the value of precision is 1

Impact Analysis Process - Adequacy

- Adequacy of an impact analysis approach is the ability of the approach to identify all the affected elements to be modified. Ideally, AIS $\subseteq$ CIS.
- Adequacy is repressed in terms of a performance metric called inclusiveness, as follows:

$$Inclusiveness = \begin{cases} 1 & \text{if AIS } \subseteq \text{ CIS} \\ 0 & \text{otherwise} \end{cases}$$

- An inadequate approach is in fact useless, as it provides the maintenance engineer with incorrect information.

Questions

?