SOFTWARE MAINTENANCE AND CONFIGURATION MANAGEMENT

SOFTWARE MAINTENANCE:

- In a software lifetime, type of maintenance may vary based on its nature.
- It may be just a **routine maintenance** tasks as some bug discovered by some user or it may be a **large event** in itself based on maintenance size or nature.

TYPES OF SOFTWARE MAINTENANCE:

- **Corrective Maintenance**
  - It includes **modifications and updation** done in order to correct or fix problems, which are either **discovered by user** or concluded by user **error reports**.
  - It deals with the **repair of faults or defects** found in system **functions**.
  - A **defect** can result due to **errors** in **software design**, **logic** and **coding**. **Design errors** occur
    - when changes made to the software are incorrect, incomplete, wrongly communicated or the change request is misunderstood.
  - **Bugs** observed while the system is in use.

CONT…

- **Adaptive Maintenance**
  - It includes **modifications and updations** applied to **keep the software product up-to-date**.
  - It is the **implementation of changes** in a part of the system, which has been **affected by a change** that occurred in **some other part** of the system.
  - It need when the **customers need the product to run on new platform**.

CONT…

- **Perfective Maintenance**
  - It includes **modifications and updates** done in order to **keep the software usable over long period of time**.
  - It includes **new features**, **new user requirements for refining the software** and improve its **reliability** and **performance**.
  - It deals with **implementing new or changed user requirements**.
  - It involves making **functional enhancements** to the system in addition to the activities **to increase the system's performance** even when the changes have **not been suggested by faults**.

CONT…

- **Preventive Maintenance**
  - It includes **modifications and updations to prevent future problems** of the software.
  - It aims to **attend problems**, which are **not significant** at this moment but may cause **serious issues in future**.
  - It involves **performing activities** to **prevent the occurrence of errors**.
  - It tends to **reduce the software complexity** thereby improving program understandability and increasing software maintainability.
  - It includes **documentation updating**, **code optimization** and **code restructuring**.

RE-ENGINEERING:

- **“When we need to update the software to keep it to the current market, without impacting its functionality, it is called software re-engineering.”**
- It is a process where the design of software is changed and programs are re-written.
- Re-Engineering Process
  - **Decide** what to re-engineer. Is it whole software or a part of it?
  - **Perform** Reverse Engineering, in order to obtain specifications of existing software.
  - **Restructure** Program if required. For example, changing function-oriented programs into object-oriented programs.
  - **Apply** Forward engineering concepts in order to get re-engineered software.

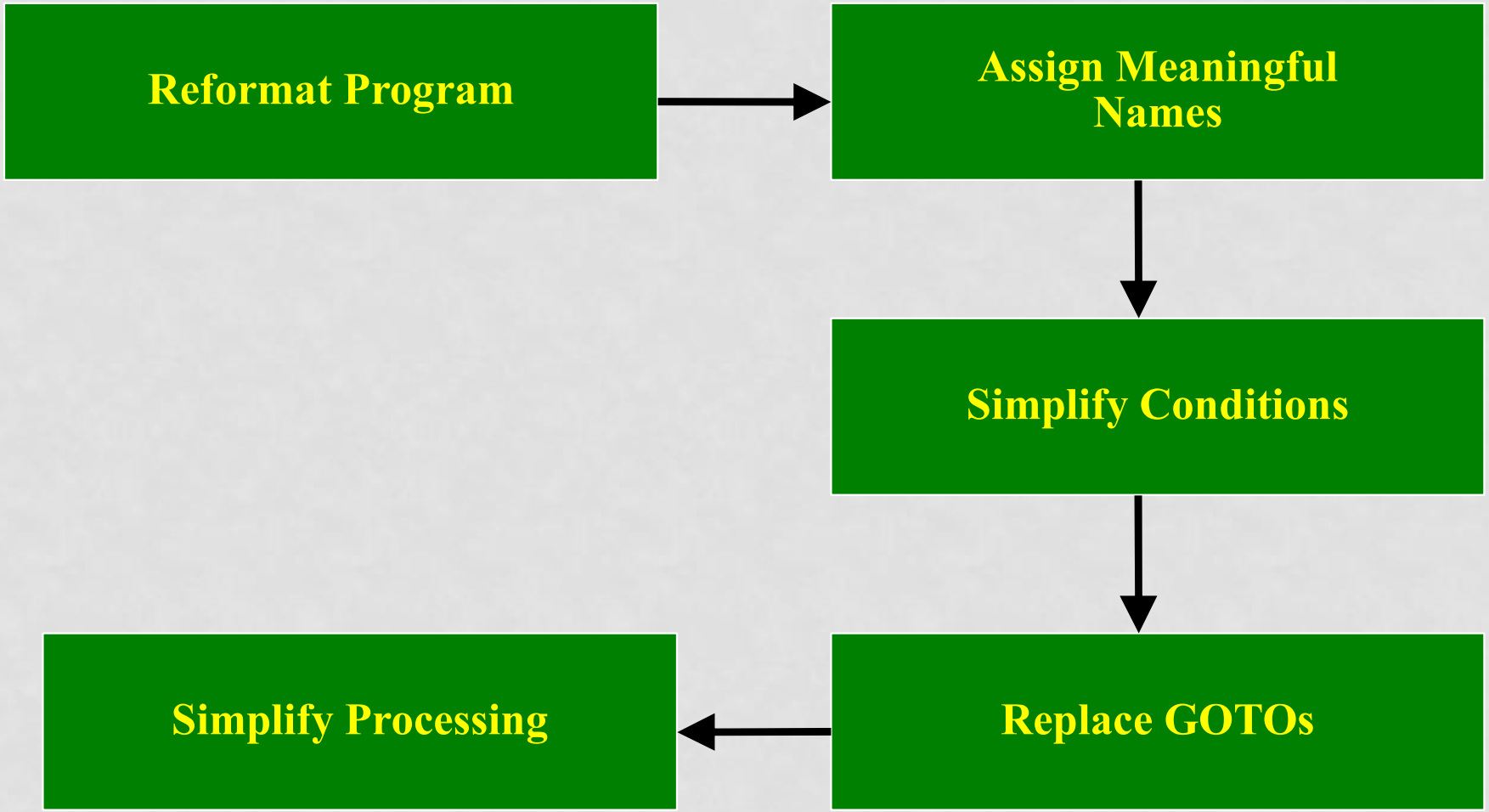
REVERSE ENGINEERING

- **It is the process of recovering the design and the requirements specification of a product from an analysis of a program code.**
- Reverse engineering is an important maintenance technique:
  - several existing software products are unstructured
  - were not developed using software engineering principles
- First carry out cosmetic changes to the code to improve:
  - readability
  - structure
  - understandability
- Without changing any functionalities.

CONT…

- During the reverse engineering
  - The old code is analyzed to extract the module specifications.
  - The module specifications are analyzed to produce the design.
  - The design is analyzed (abstracted) to produce the original requirements specification.
- The change requests are then applied to the requirements specification
- Arrive at the new requirements specification.
- Forward engineering is carried out to produce the new code.

COSMETIC CHANGES



CONT…

- After the cosmetic changes, the process of extracting the code, design by:
  - fully understand the code.
- Automatic tools can be used to help derive:
  - data flow and control flow diagrams from the code.
- Extract structure chart:
  - module invocation sequence and data interchange among modules.
- Extract requirements specification (SRS):
  - after thoroughly understanding the code.
  - design has been extracted.

CONT…

- Reformat the program:
- Give more meaningful names to:
  - Variables, data structures, and functions.
- Replace complex and nested conditional expressions:
  - simpler conditional statements

FORWARD ENGINEERING

- Forward engineering is a process of obtaining desired software from the specifications in hand which were brought down by means of reverse engineering.
- It assumes that there was some software engineering already done in the past.
- It is same as software engineering process with **only one difference** it is carried out always after reverse engineering.
- It applies software engineering principles, concepts and methods to **re-create an existing application**.
- The redeveloped program extends the capabilities of the older application.

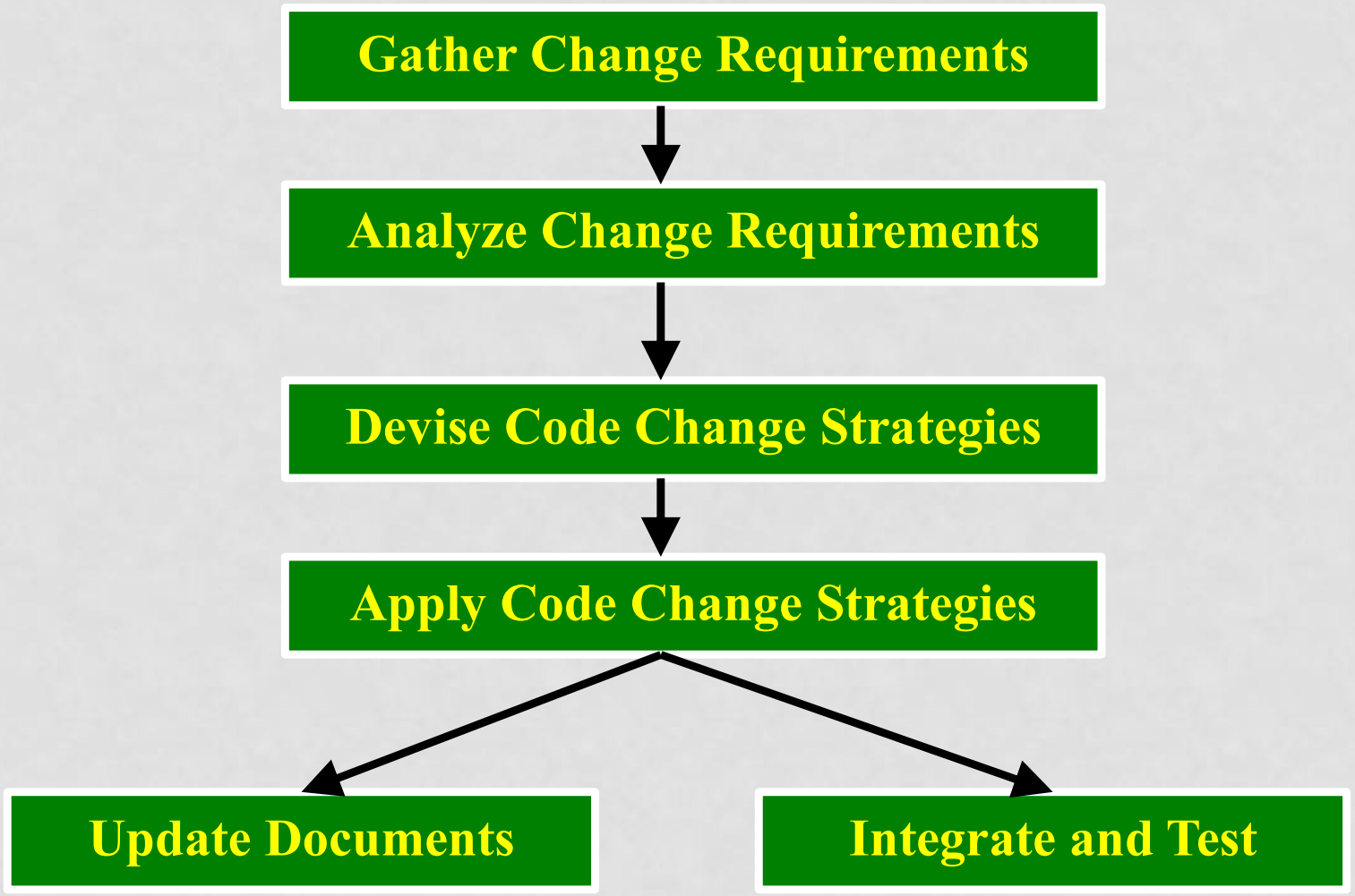
SOFTWARE MAINTENANCE PROCESS MODELS

- Maintenance activities are not unique:
  - depend on the extent of modifications required
  - depend on condition of the product:
    - how structured it is
    - how well documented it is

MODEL - 1

- **When the required changes are small and simple:**
  - The code can be directly modified
  - changes reflected in all relevant documents
- **When required changes are not small.**
  - More elaborate activities are required
  - Start by gathering change requirements
  - Analyze change requirements
  - Formulate strategies for code change

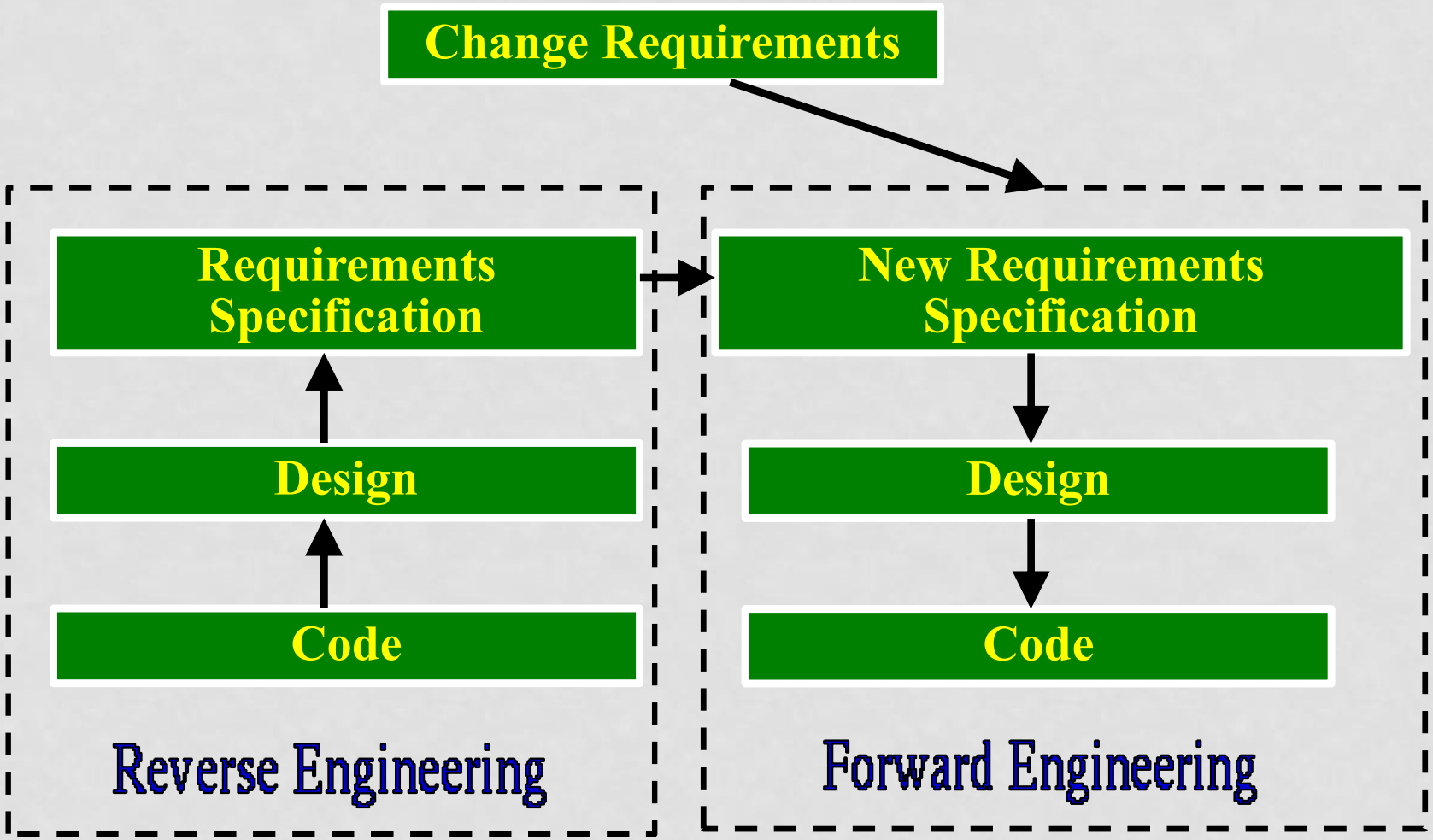
CONT…


Gather Change Requirements
Analyze Change Requirements
Devise Code Change Strategies
Apply Code Change Strategies
Update Documents
Integrate and Test

MODEL -2

- For complex maintenance projects, software reengineering needed:
  - A reverse engineering cycle followed by a forward engineering cycle.
- with as much reuse as possible from existing code and other documents.
- Preferable when:
  - amount of rework is important
  - software has poor structure

MODEL-2


Change Requirements
Requirements Specification
Design
Code
Reverse Engineering
New Requirements Specification
Design
Code
Forward Engineering

SCM (SOFTWARE CONFIGURATION MANAGEMENT) PROCESS

- The result of a large software development effort consist of a large number of objects i.e. source code, design document, SRS document, test document, etc.
- These objects are usually referred to and modified by a software developer throughout the SDLC.
- **The state of all these objects at any point of time is called the configuration of the software.**
- Software configuration management deals with **effectively tracking and controlling** the configuration of software during SDLC.

SCM (SOFTWARE CONFIGURATION MANAGEMENT) PROCESS

- The software configuration management process have four objectives:
  - To identify all items that collectively define the software configuration.
  - To manage changes to one or more of these items.
  - To facilitate the construction of different versions of an application.
  - To ensure that software quality is maintained as the configuration evolves over time.

CONT…

- Configuration Management Activities:
  - Configuration identification
    - Which part of system should be kept track of
  - Configuration control
    - Changes to a system happen smoothly