



## د. لمياء أبوزيد

# 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

# Reengineering Approaches

---

❑ Five basic approaches to reengineering software systems exist:

1. Big Bang approach.
2. Incremental approach.
3. Partial approach.
4. Iterative approach.
5. Evolutionary approach:

❑ The five approaches are different in two aspects:

1. the extent of reengineering performed
2. the rate of substitution of the operational system with the new one.

# Reengineering Approaches- Big Bang Approach

---

- ❑ The **Big Bang** approach replaces the whole system at once.
- ❑ Once a reengineering effort is initiated, it is continued until all the objectives of the project are achieved and the target system is constructed.
- ❑ This approach is generally used if **reengineering cannot be done in parts**.
  - For example, if there is a need to move to a different system architecture, then all components affected by such a move must be changed at once.
- ❑ The disadvantage of Big Bang is that the **reengineering project becomes a monolithic** task, which may not be desirable in all situations.
- ❑ The Big Bang approach **consumes too much resources** at once for large systems, and takes a long stretch of time before the new system is visible.

# Reengineering Approaches - Incremental Approach

---

- ❑ In the **Incremental Approach** approach a system is reengineered **gradually**, one step closer to the target system at a time. For a large system, several new **interim versions** are produced and released.
- ❑ Successive interim versions satisfy increasingly more project goals than their preceding versions.
- ❑ The advantages of this approach are as follows:
  - locating errors becomes easier, because one can clearly identify the newly added components.
  - It becomes easy for the customer to notice progress, because of interim version release.
- ❑ The disadvantages of the incremental approach are as follows:
  - with multiple interim versions and their version controls, reengineering takes much longer time to complete
  - even if there is a need, the entire architecture of the system cannot be changed.

# Reengineering Approaches- Partial Approach

---

- ❑ In the **partial approach** only a part of the system is reengineered and then it is integrated with the non-engineered portion of the system.
- ❑ The following three steps are followed in the partial approach:
  1. The existing system is **partitioned** into two parts: one part is identified to be reengineered and the remaining part to be not reengineered.
  2. **Reengineering work is performed** using either the “Big Bang” or the “Incremental” approach.
  3. The two parts the not-to-be-reengineered part and the reengineered part of the system, are **integrated** to make up the new system.
- ❑ The advantage is reducing the scope of reengineering that is: less time and costs less.
- ❑ A disadvantage of the partial approach is that modifications are not performed to the interface between the portion modified and the portion not modified.

# Reengineering Approaches- Iterative Approach

---

- ❑ The **iterative** reengineering process is applied on the source code of a few procedures at a time, with each reengineering operation lasting for a short time. This process is repeatedly executed.
- ❑ During the execution of the process four types of components can coexist:
  1. Old components not reengineered.
  2. Components currently being reengineered.
  3. Components already reengineered.
  4. New components added to the system.
- ❑ There are two advantages of the iterative reengineering process:
  - It guarantees the continued operation of the system during the execution of the reengineering process
  - The maintainers' and the users' familiarities with the system are preserved.

# Reengineering Approaches - Evolutionary Approach

---

- ❑ In the **Evolutionary** approach components of the original system are substituted with re-engineered components.
- ❑ Software engineers focus their reengineering efforts on identifying **functional objects** irrespective of the locations of those components within the current system.
- ❑ There are two advantages of the Evolutionary approach:
  - The resulting design is more cohesive (built with **functionally cohesive components**).
  - The scope of individual components is reduced.
- ❑ A major disadvantage:
  - All the functions with much similarities must be first identified throughout the operational system.
  - Next, those functions are refined as one unit in the new system.



# Data Reverse Engineering (DRE)

---

□ Data Reverse Engineering (DRE) is defined as “the use of structured techniques to reconstitute the data assets of an existing system” .

□ The purpose of DRE is as follows:

1. Knowledge acquisition.
2. Tentative requirements.
3. Documentation.
4. Integration.
5. Data administration.
6. Data conversion.
7. Software assessment.
8. Quality assessment.
9. Component reuse.

# Database Reverse Engineering (DBRE)

---

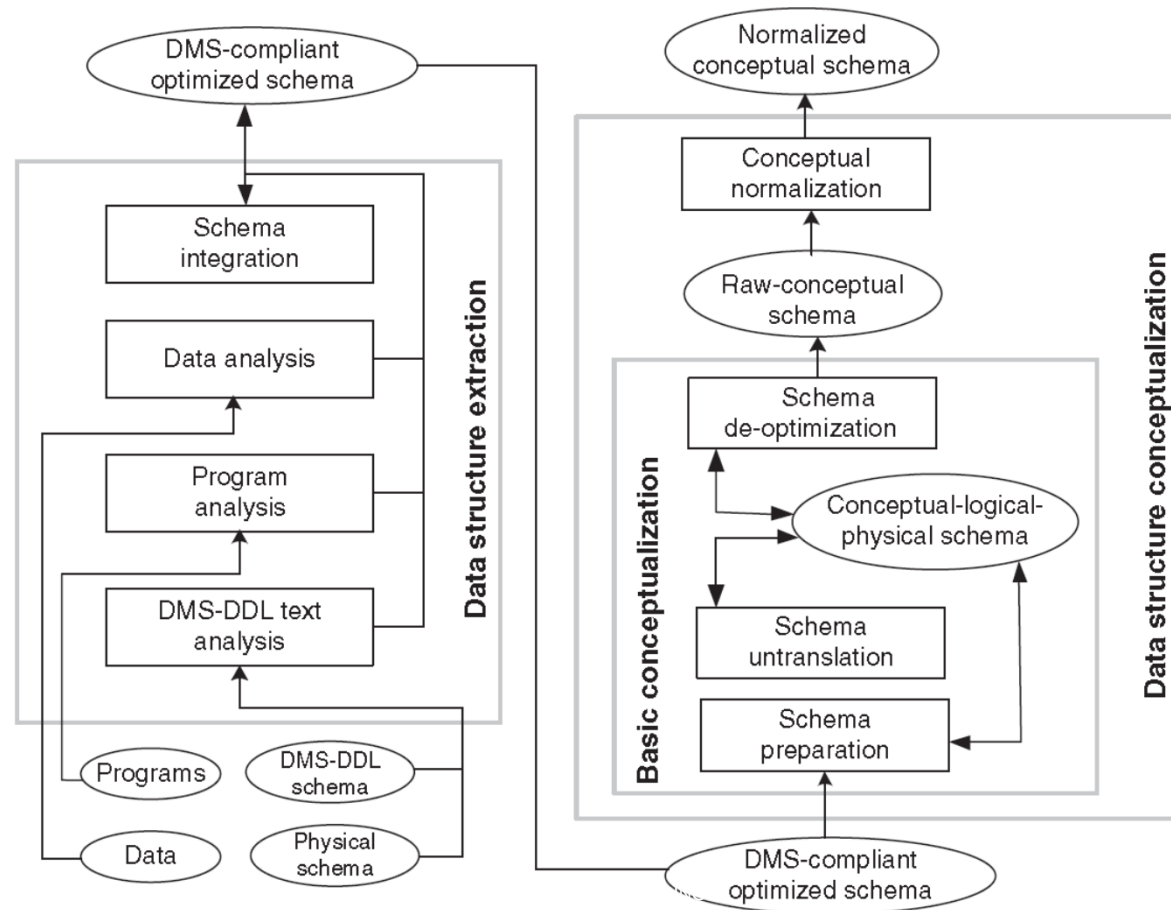
- ❑ Database reverse engineering (DBRE) refers to recovering the specifications (i.e. the conceptual schema) in database of such applications
  - The conceptual schema is an abstract, implementation independent description of the stored data.
- ❑ A DBRE process facilitates understanding and redocumenting an application's database and files.
- ❑ By means of a DBRE process, one can recreate the complete logical and conceptual schemas of a database physical schema.
  - A logical schema describes the data structures in concrete forms as those are implemented by the data manager.
  - The physical schema of a database implements the logical schema by describing the physical constructs.

# Database Reverse Engineering (DBRE)

---

- ❑ The forward design process of a database comprises three basic phases as follows:
  1. Conceptual phase: The user requirements are gathered, studied, and formalized into a conceptual schema.
  2. Logical phase: the conceptual schema is expressed as a simple model, which is suitable for optimization reasoning (ER or ORM).
  3. Physical phase: the logical schema is described in the data description language (DDL) of the data management system and the host programming language.
- ❑ A DBRE process is based on backward execution of the logical phase and the physical phase.
- ❑ The process is divided into two main phases:
  1. data structure extraction.
  2. data structure conceptualization.

# Database Reverse Engineering (DBRE)



# Questions

---

?