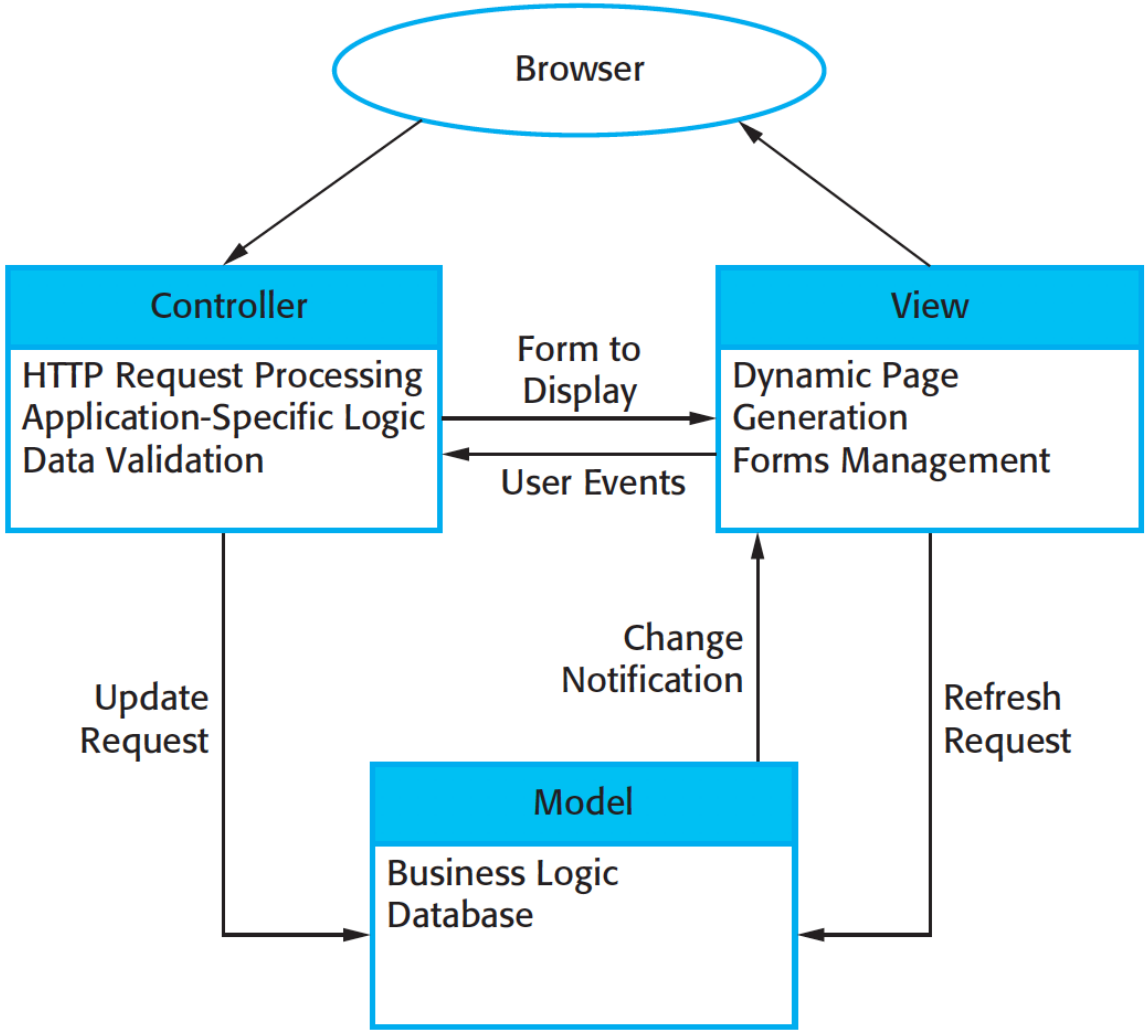
Architectural Design

Model-View-Controller and Layered Architecture Pattern

Indranil Saha

Department of Computer Science and Engineering
Indian Institute of Technology Kanpur

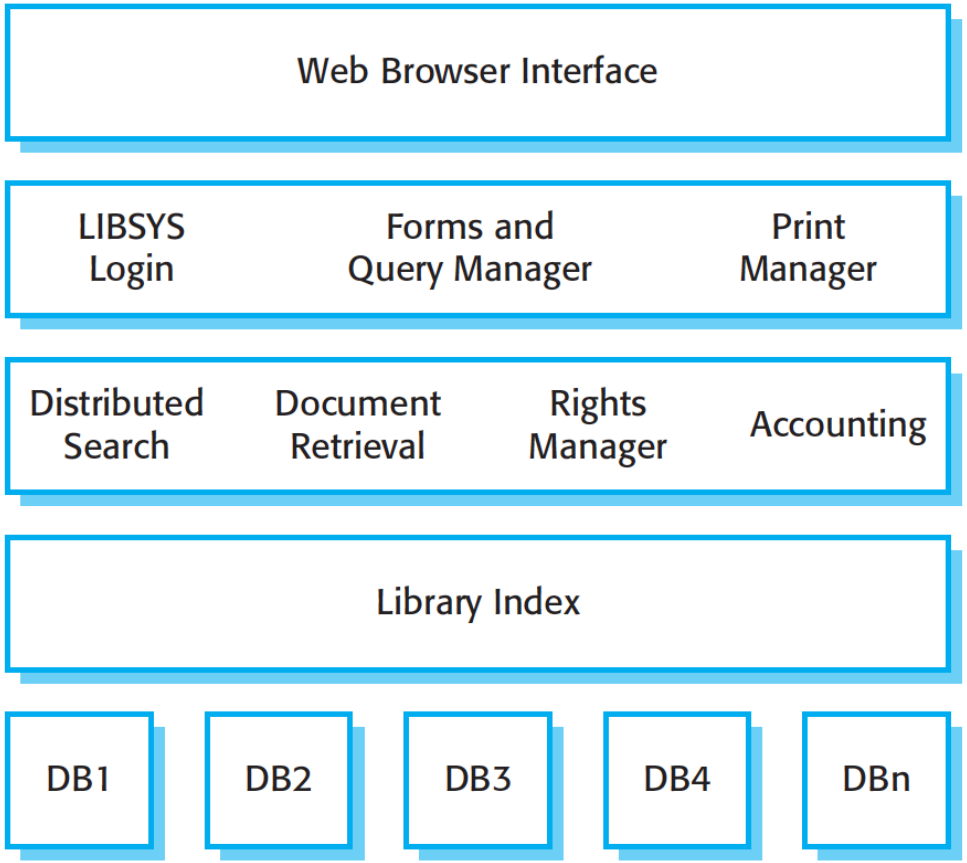
Model-View-Controller Pattern



Model-View-Controller Pattern - Contd.

- **When Used:**
  - When there are multiple ways to view and interact with data
  - When the future requirements for interaction and presentation of data are unknown
- **Advantages:**
  - Allows the data to change independently of its representation and vice versa
  - Supports presentation of the same data in different ways with changes made in one representation shown in all of them
- **Disadvantages:**
  - Can involve additional code and code complexity when the data model and interactions are simple

Layered Architecture Pattern



Layered Architecture Pattern - Contd.

- **When Used:**
  - When building new facilities on top of existing systems
  - When the development is spread across several teams with each team responsibility for a layer of functionality
  - When there is a requirement for multi-level security
- **Advantages:**
  - Allows replacement of entire layers as long as the interface is maintained
  - Redundant facilities (e.g., authentication) can be provided in each layer to increase the dependability of the system
- **Disadvantages:**
  - Providing a clean separation between layers is often difficult - a high-level layer may have to interact directly with lower-level layers rather than through the layer immediately below it
  - Performance degradation due to multiple levels of interpretation of a service request as it is processed at each layer

Architectural Design

Model-View-Controller and Layered Architecture Pattern

Indranil Saha

Department of Computer Science and Engineering
Indian Institute of Technology Kanpur