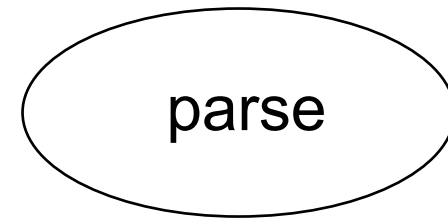


Function Modeling With Data Flow Diagrams

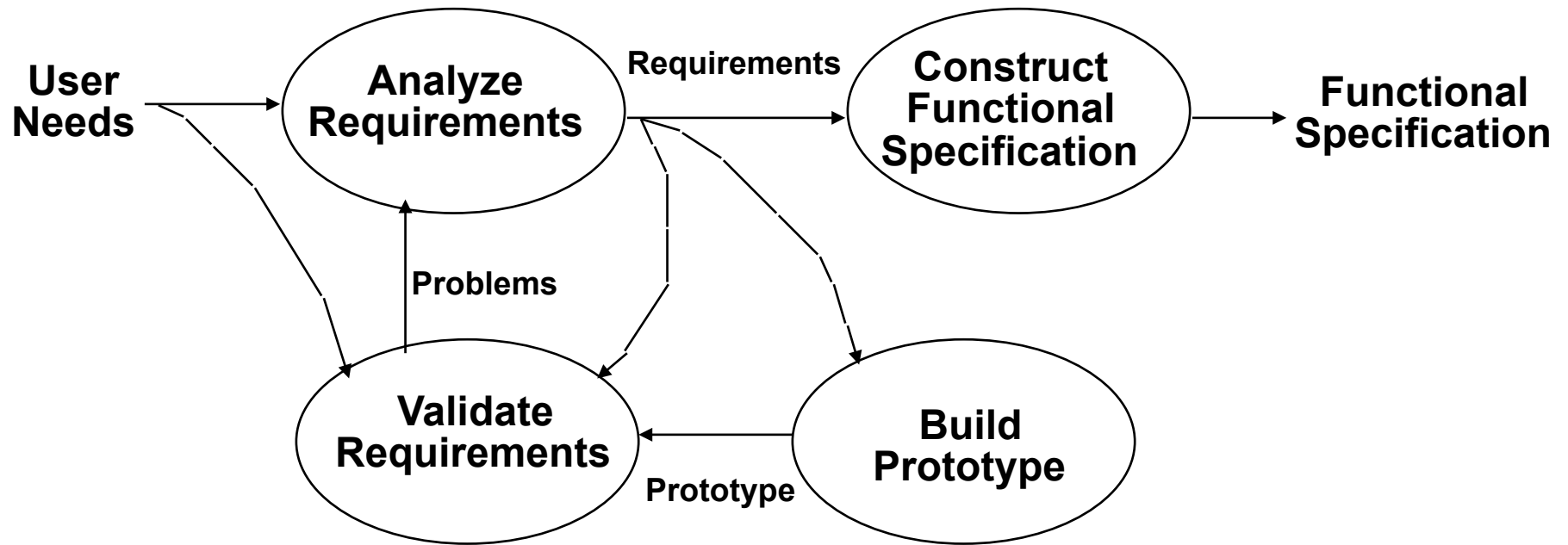
- Analysis technique based on activities (verbs)
- Lower level (detail) diagrams may refine high level activities
- May be integrated with a data dictionary
- May be extended with control information to model real-time systems

Data Flow Diagram Icons

- Activities are denoted by circles or ovals. The label should be a verb or verb phrase
- Arcs connecting circles denote data flows. Each arc is labeled with a different noun phrase
- Two horizontal parallel lines denote a repository (persistent storage for data)
- Rectangles denote external actors or interfaces

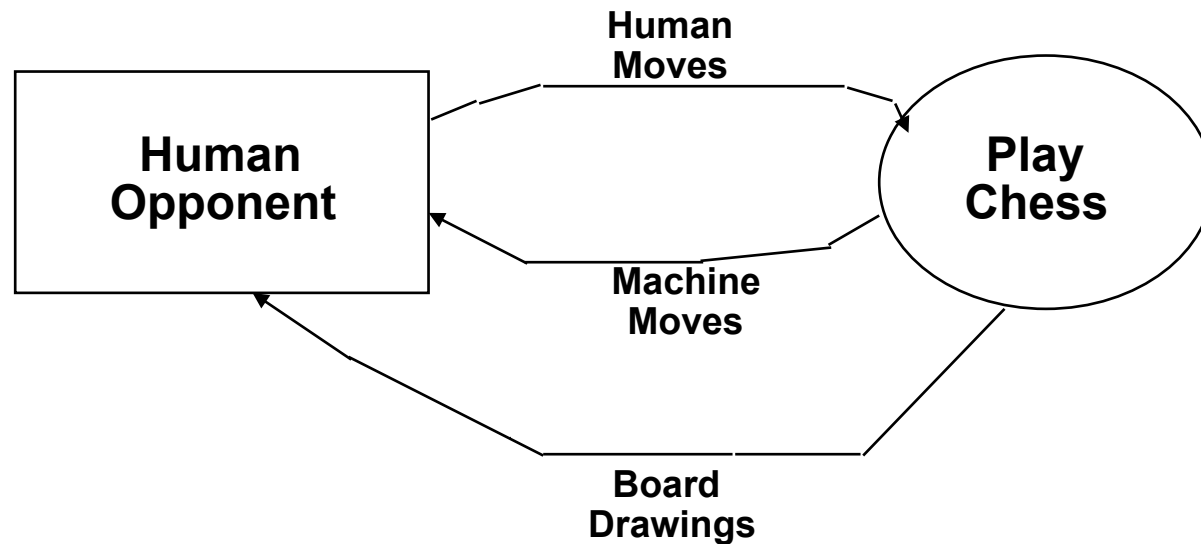


Example Dataflow Diagram



Context Diagram

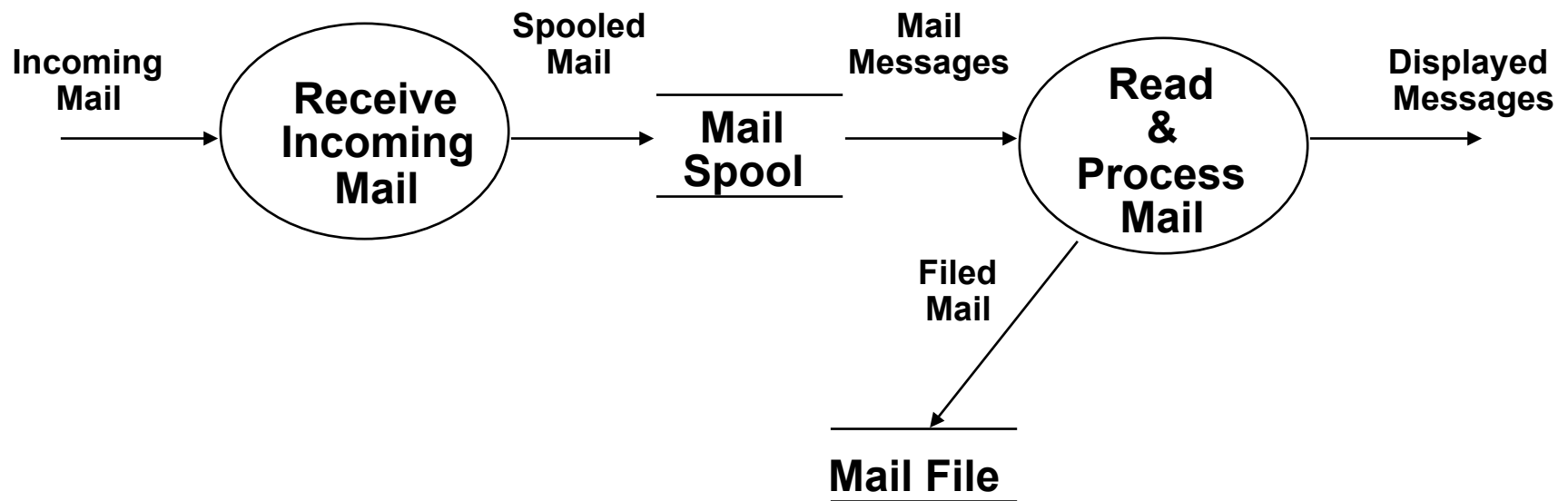
- The top level data flow diagram is called the context diagram
- It contains exactly one activity node denoting the overall function of the system
- It can help define the boundaries of the system to be built



Data Dictionary

- Every arc in a data flow diagram (DFD) indicates a flow of data
- A *data dictionary* allows more precise descriptions of the data to be made
 - For example, textual definitions and data type can be specified
- Sometimes it is desirable to indicate that a data flow needs to be captured and saved
- Typically all labeled data in a DFD should appear in the data dictionary

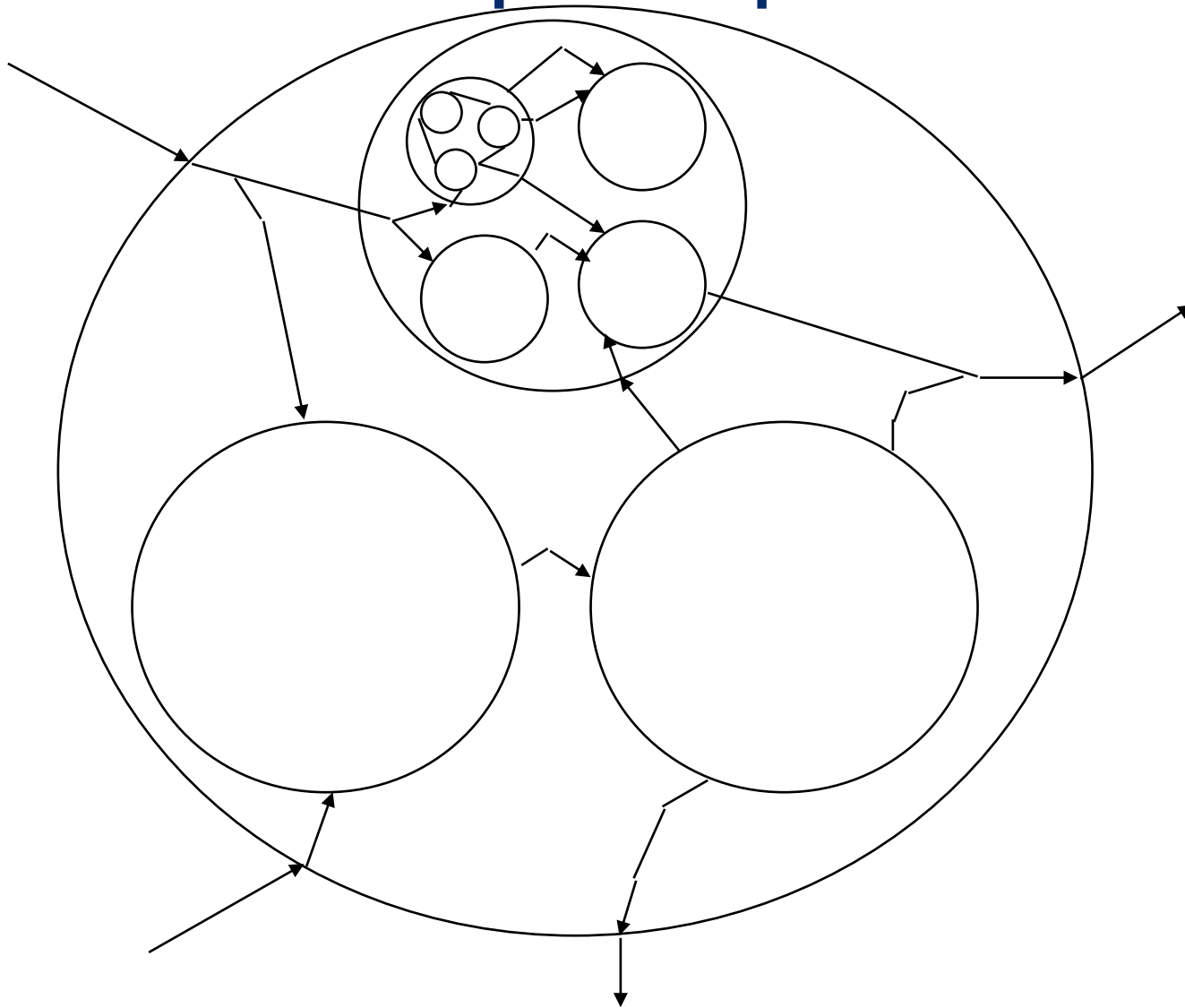
Example DFD using Repositories



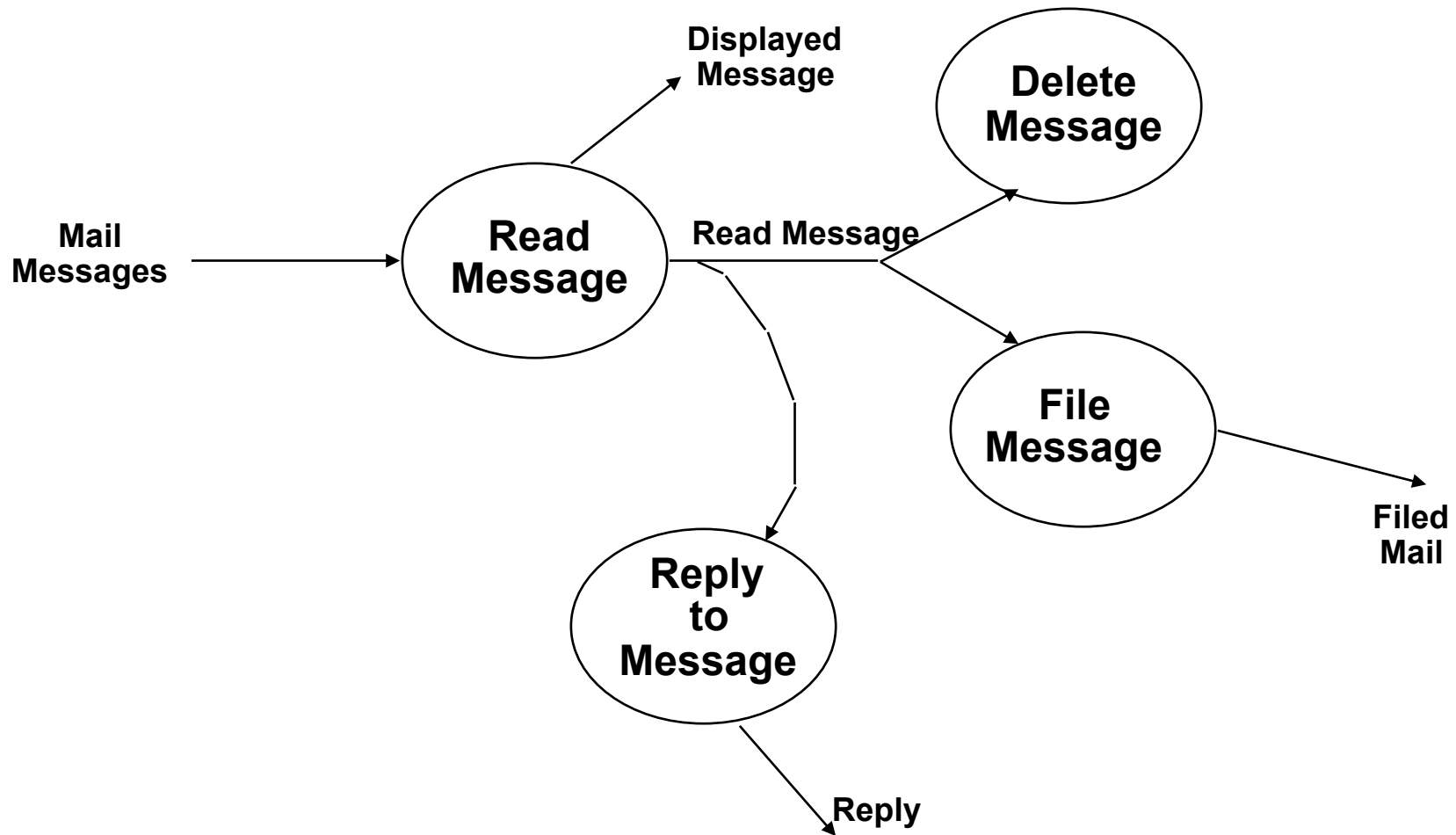
Refining DFDs

- One important part of system modeling is separating important concepts from details
- Data flow modeling supports this by allowing diagrams to be nested
- An activity node in a high level diagram can be expanded to a full diagram at a lower level
- Lines coming into and leaving the activity node must be present in the low level diagram (*balancing*)
- This consistency rule can be checked automatically by a CASE tool

Four Levels of DFDs Superimposed



Example : Refined DFD



Oops - The possibility of a Reply was not mentioned in the parent diagram

Activity Refinement

- Data flow modeling begins with a Context Diagram. Its activity node is expanded into a refined data flow diagram
- A node can continue to be expanded until it becomes clear how the function of the node will be accomplished
- The final description of functionality is expressed in a textual specification using a notation such as pseudo code
- Data flow diagrams do not include a specification of the order in which the activity nodes are executed

Data Refinement

- It is also possible to refine the data flows and repositories in a DFD. There are several ways to do this:
 - Data dictionary: allows you to specify data type, structure, and descriptive text
 - Data modeling diagram: allows you to use techniques such as Entity-Relationship models or class diagrams to specify your data