

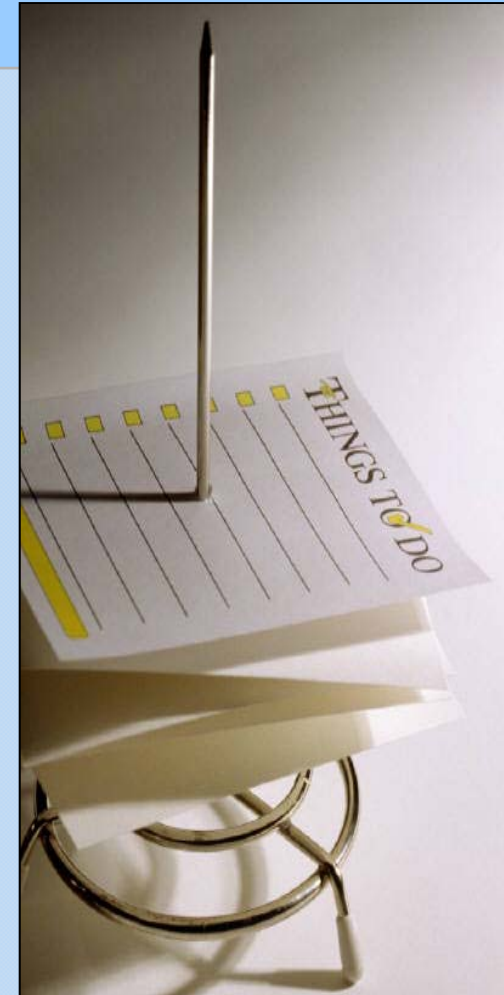
Introduction to Pattern Oriented Analysis and Design (POAD)

Instructor: Dr. Hany H. Ammar

Dept. of Computer Science and
Electrical Engineering, WVU

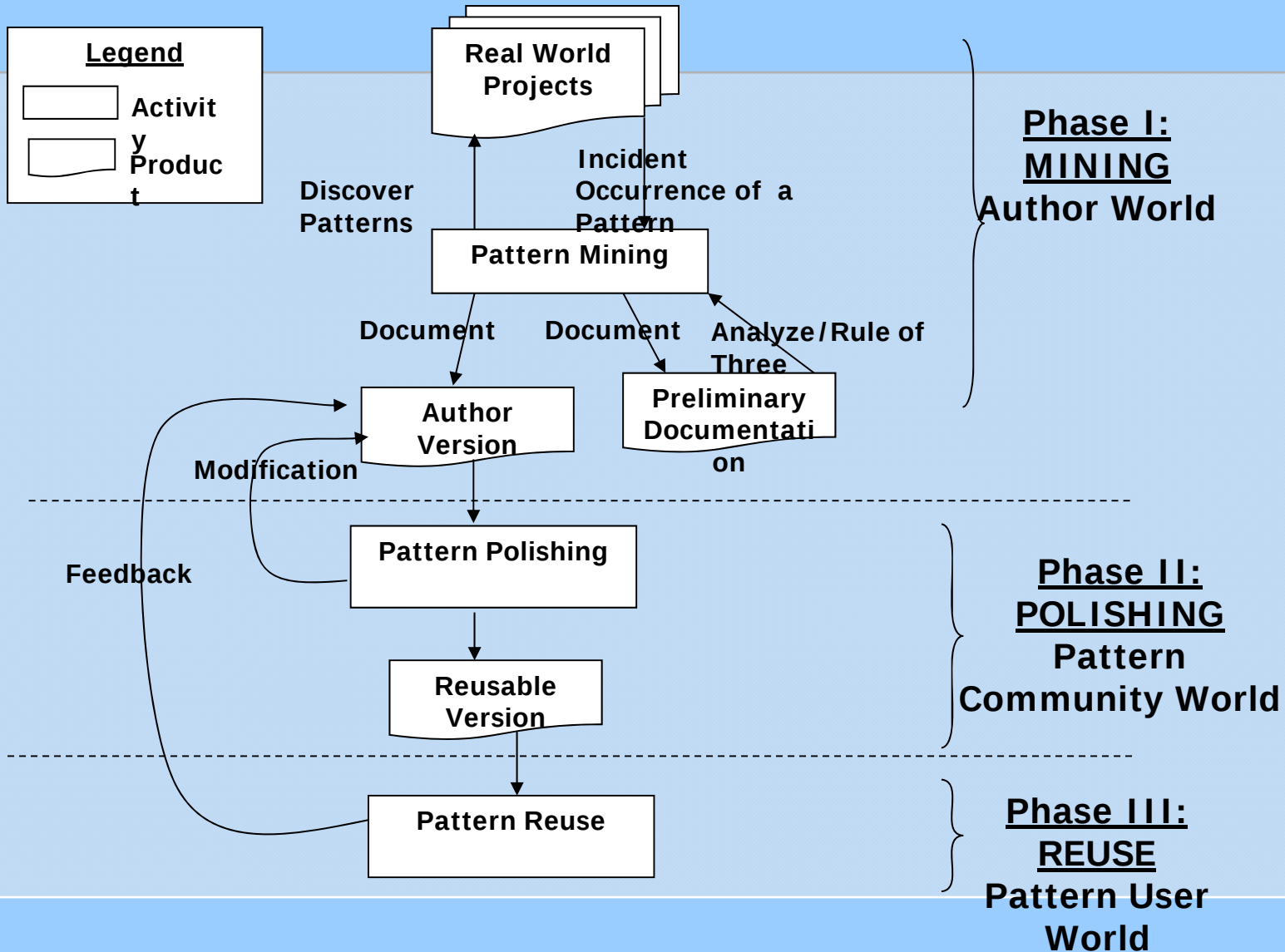
Outline

- Review of Design Patterns
 - The Lifecycle of a Pattern
 - Examples of Design Patterns
 - The Command Pattern
 - The Observer Pattern
 - The Strategy Pattern
- Pattern Oriented Development
 - The Analysis phase
 - The Design phase
 - The design refinement phase
- The Feedback Control Example



The Lifecycle of Patterns

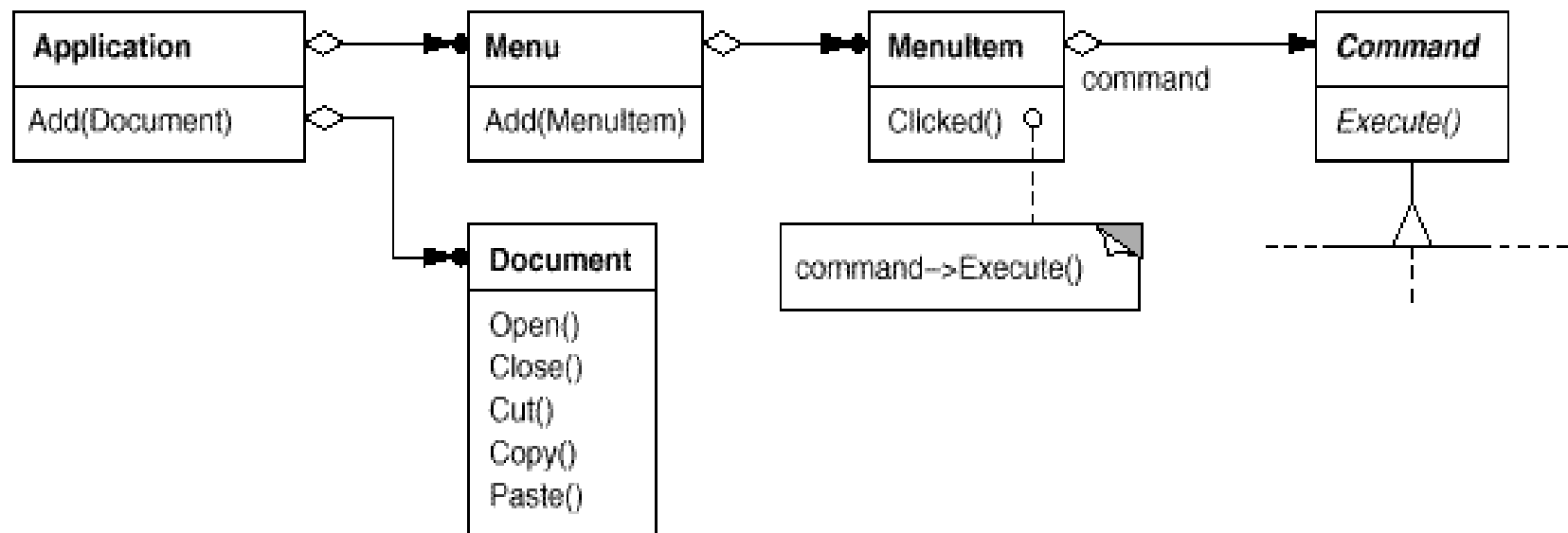
(From the Reference "Pattern-Oriented Analysis and Design",
Sherif M Yacoub and Hany H. Ammar Addison-Wesley Inc., 2004)



Review of Design Patterns

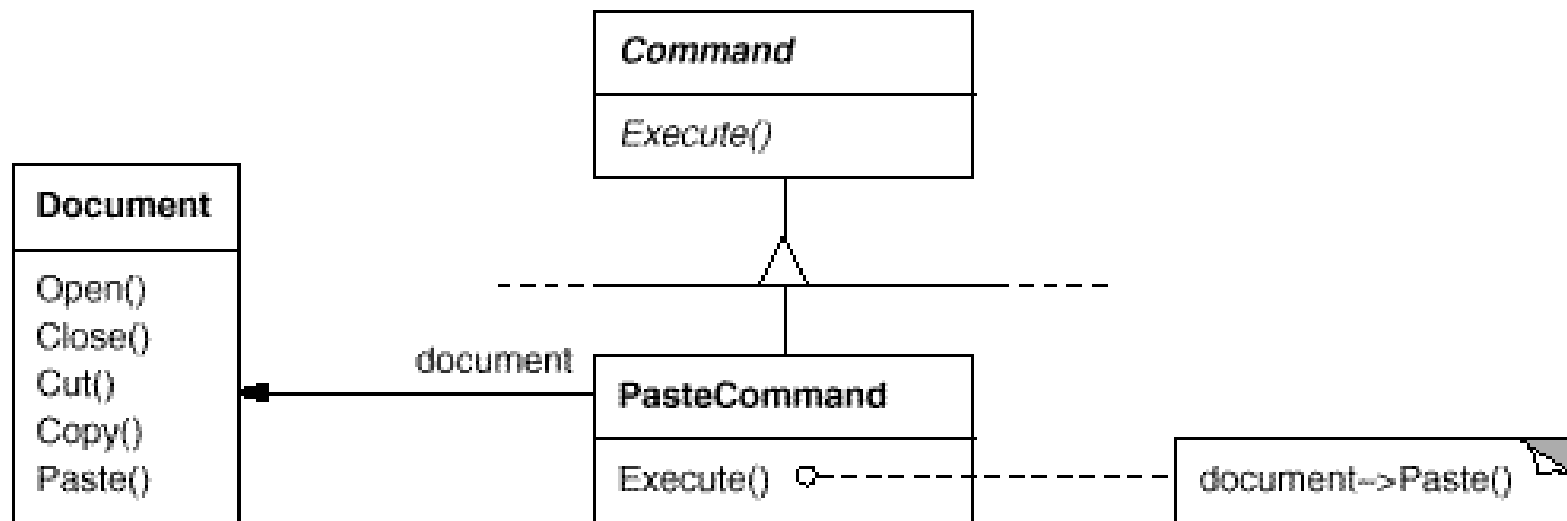
- Examples of Design Patterns (From the Design Patterns CD by Gamma et al, Addison-Wesley Inc., 1998)

The Command Pattern: Encapsulate a request as an object



Review of Design Patterns

The Command Pattern



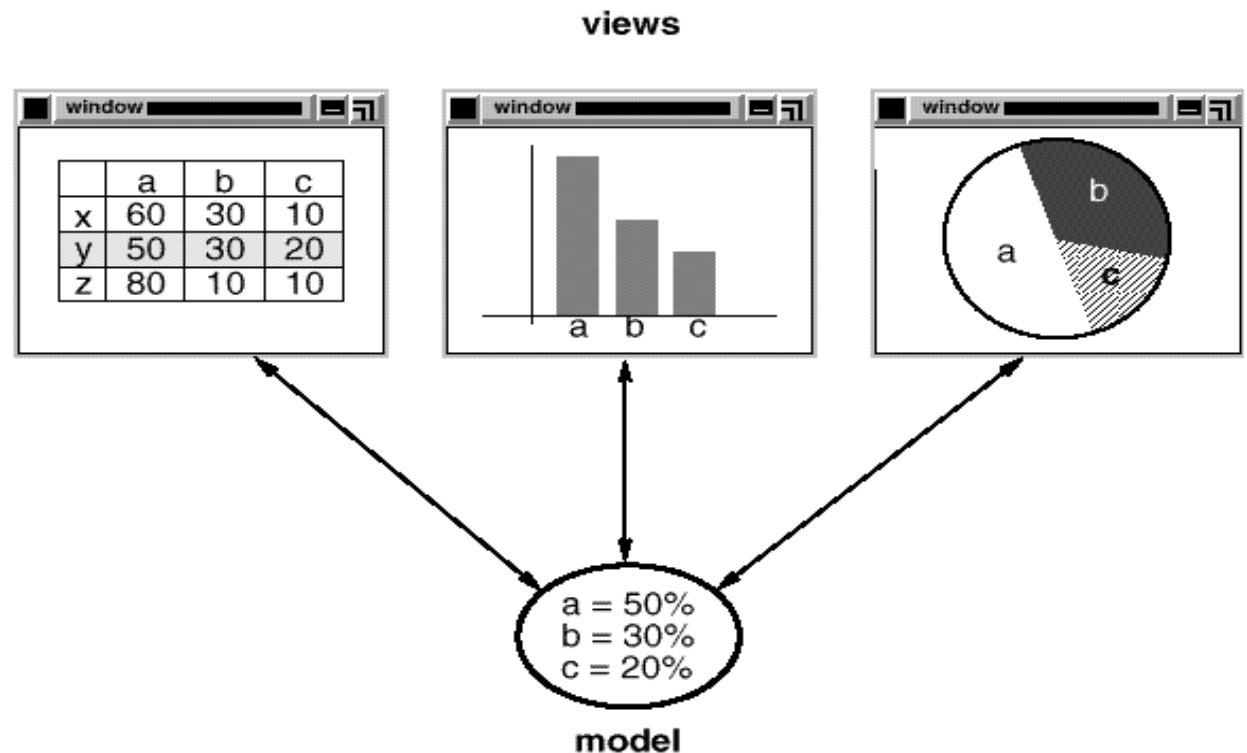
From the Design Patterns CD by Gamma et al, Addison-Wesley Inc., 1998

Review of Design Patterns

Examples of Design Patterns

The Observer Pattern: when one object changes state, all its dependents are notified and updated automatically

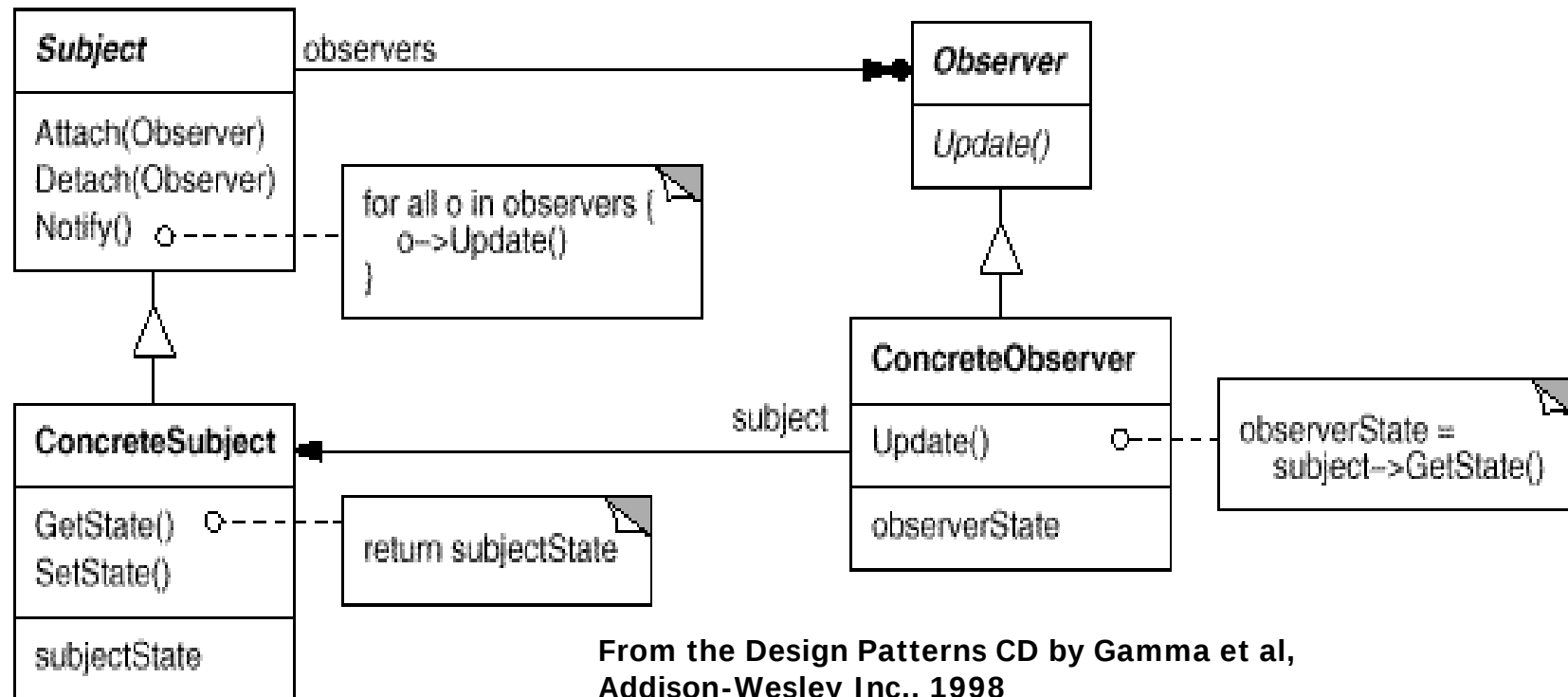
Model View
Controller
example



Review of Design Patterns

Examples of Design Patterns

The Observer Pattern class diagram

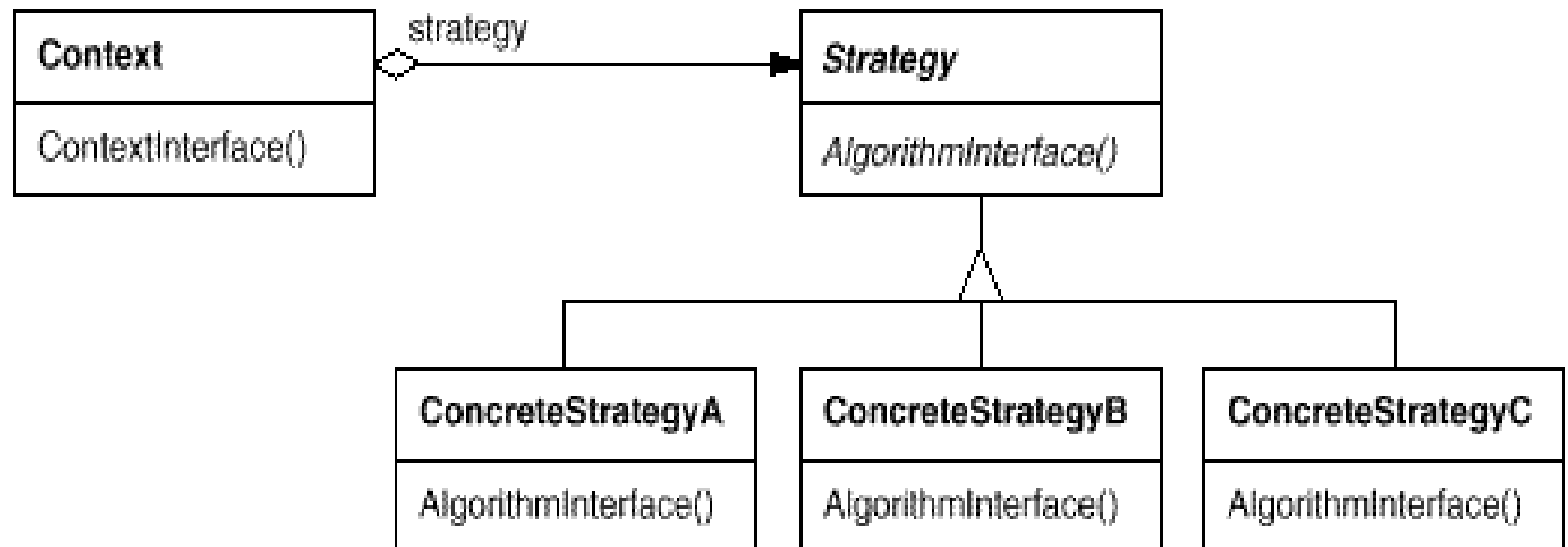


From the Design Patterns CD by Gamma et al,
Addison-Wesley Inc., 1998

Review of Design Patterns

Examples of Design Patterns

The Strategy Pattern: lets the algorithm vary independently from clients that use it





Pattern Oriented Development

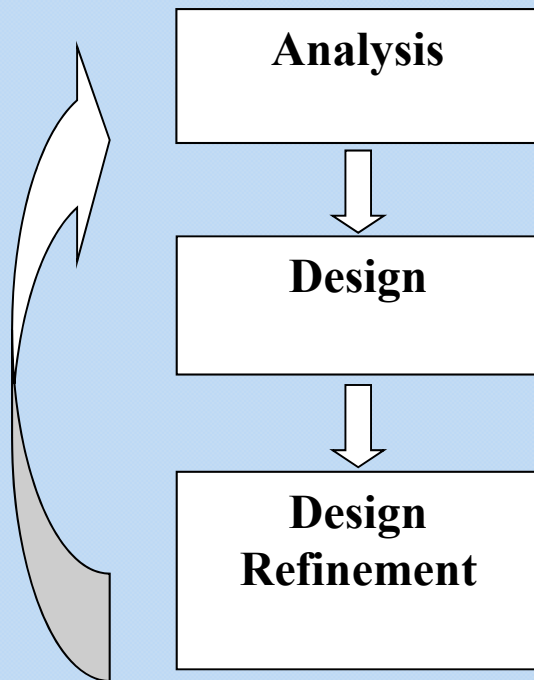
- 
- Design Patterns are used in an ad-hoc strategy for design refinement
 - They are also used to address a set of design problems without any guidance of how these patterns can be glued or interface together
 - Is there a way to use design patterns as building blocks or as components in the design of systems ?



Pattern Oriented Development

- Pattern Oriented Analysis and Design (POAD)
 - The process aspects of POAD explains the phases and steps to develop an application design using patterns
 - the POAD process has three phases:
 - Analysis
 - Design
 - Design Refinement

Pattern Oriented Development



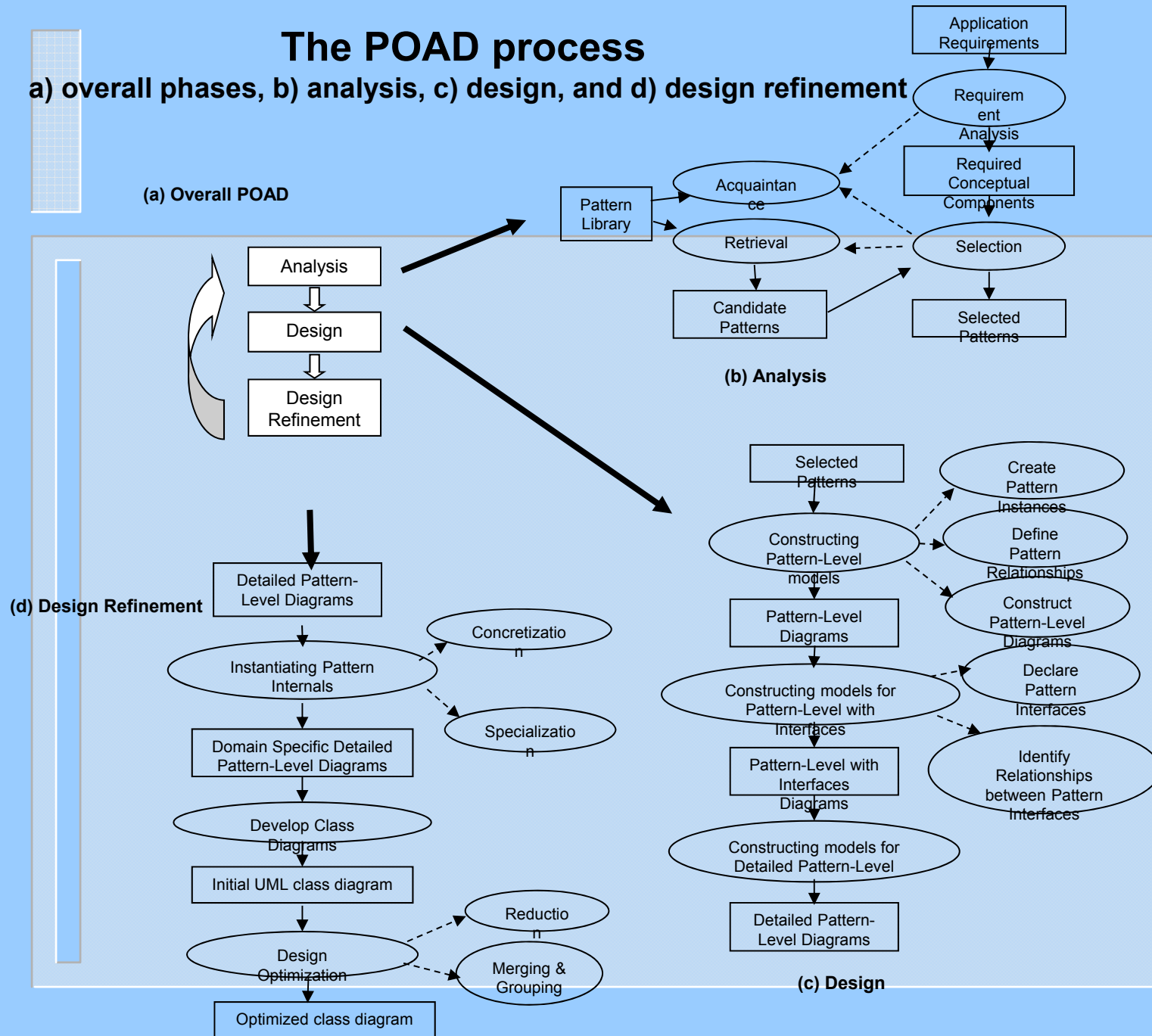
a logical model is developed and patterns are selected

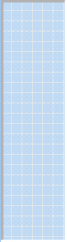
where patterns are glued together to produce a detailed pattern-level diagram

An initial class diagram, and a more dense and profound class diagram, and sequence diagrams are developed

The POAD process

a) overall phases, b) analysis, c) design, and d) design refinement





POAD Analysis Phase

- Develop use case diagrams to identify the problems to be solved and the possible breakdown of the application as a set of logical components.
- Acquaintance with relevant pattern databases to get the analyst familiar with existing solutions.
- Retrieval of patterns from the domain specific databases to select a set of candidate patterns in an automated fashion.
- Selection of patterns from a set of candidate patterns for possible inclusion in the design process.



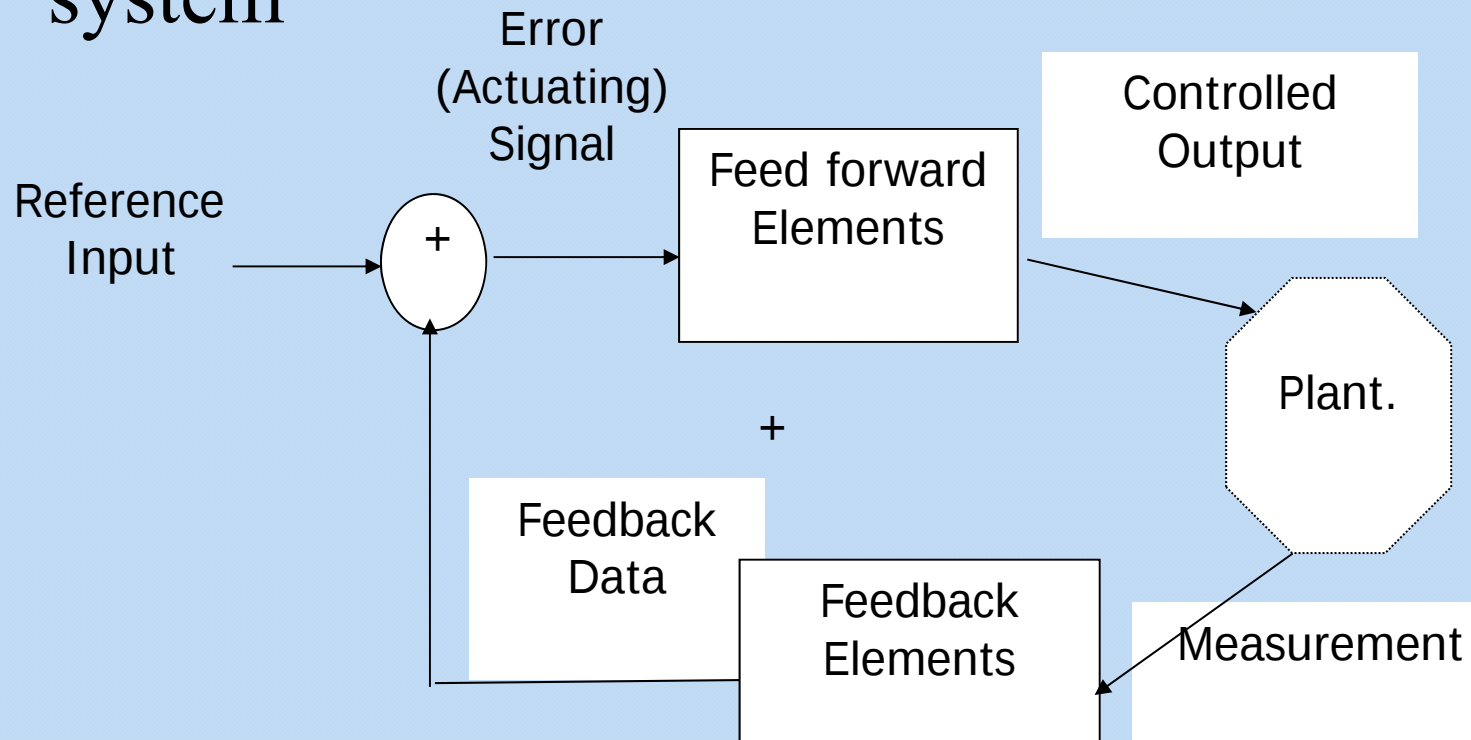
Construct Pattern-Level Models

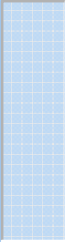
- 
- Create an instance for each selected pattern by describing the patterns and their constituents in an application specific context
 - Define how these instances are related to each other
 - The semantic of a dependency relationship used between patterns has a "*uses*" meaning

The Feedback Control Example

From Pattern-Oriented Analysis and Design,
Sherif M Yacoub and Hany H. Ammar Addison-Wesley Inc., 2004

■ Block diagram for a feedback control system





The Feedback Control Example

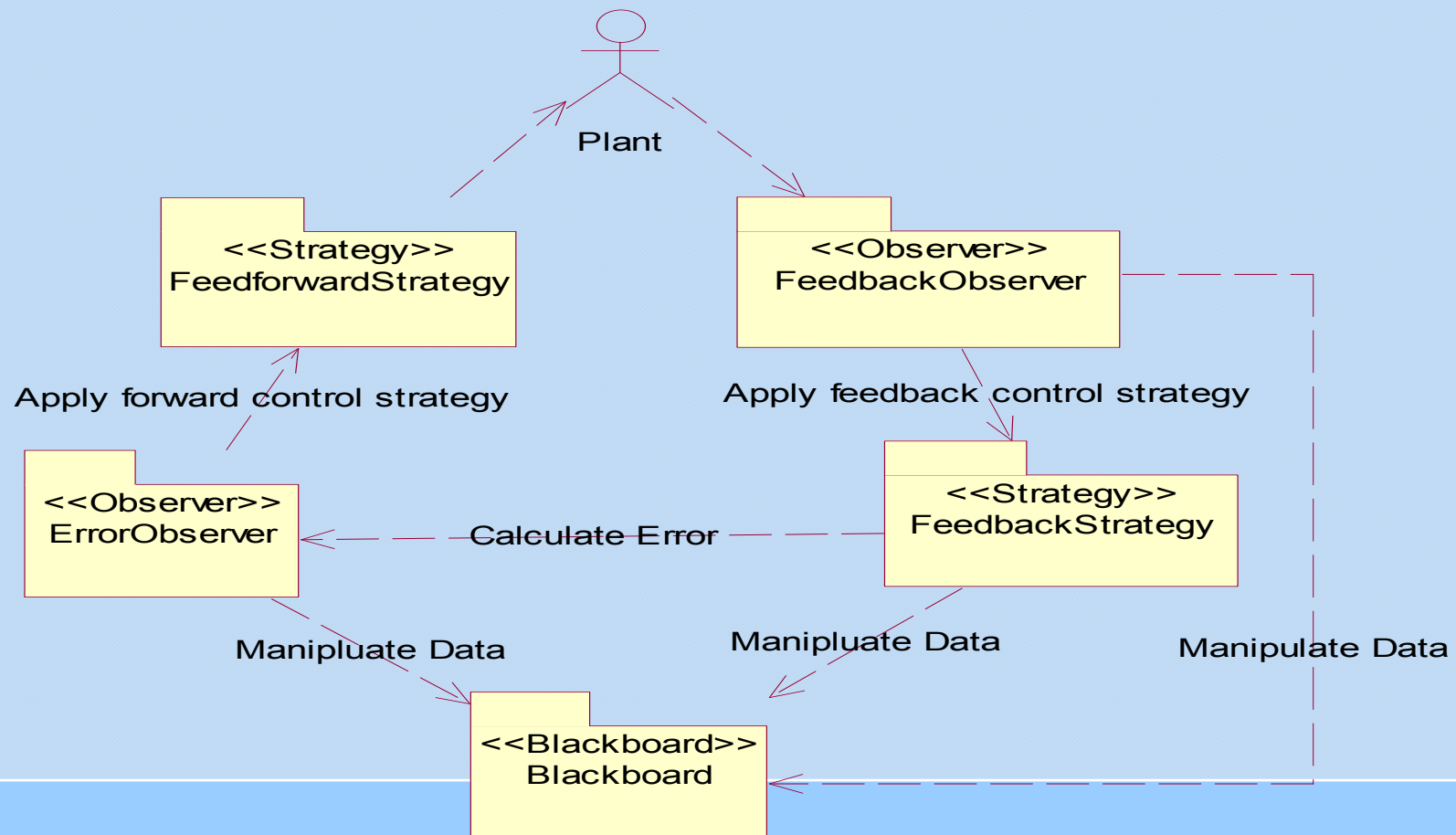
■ POAD Analysis Phase (Pattern Selection)

- The *feedforward* component implements some sort of a control strategy (instance of Strategy pattern)
- The *feedback* component receives measurements and applies a feedback control strategy
- In the *error calculation* component, the feedback controller notifies the error calculation unit with the feedback data (instances of the Observer pattern)
- Data of different types need to be exchanged between the framework components (Measurement, Feedback Data, input data , and error data)

The Feedback Control Example

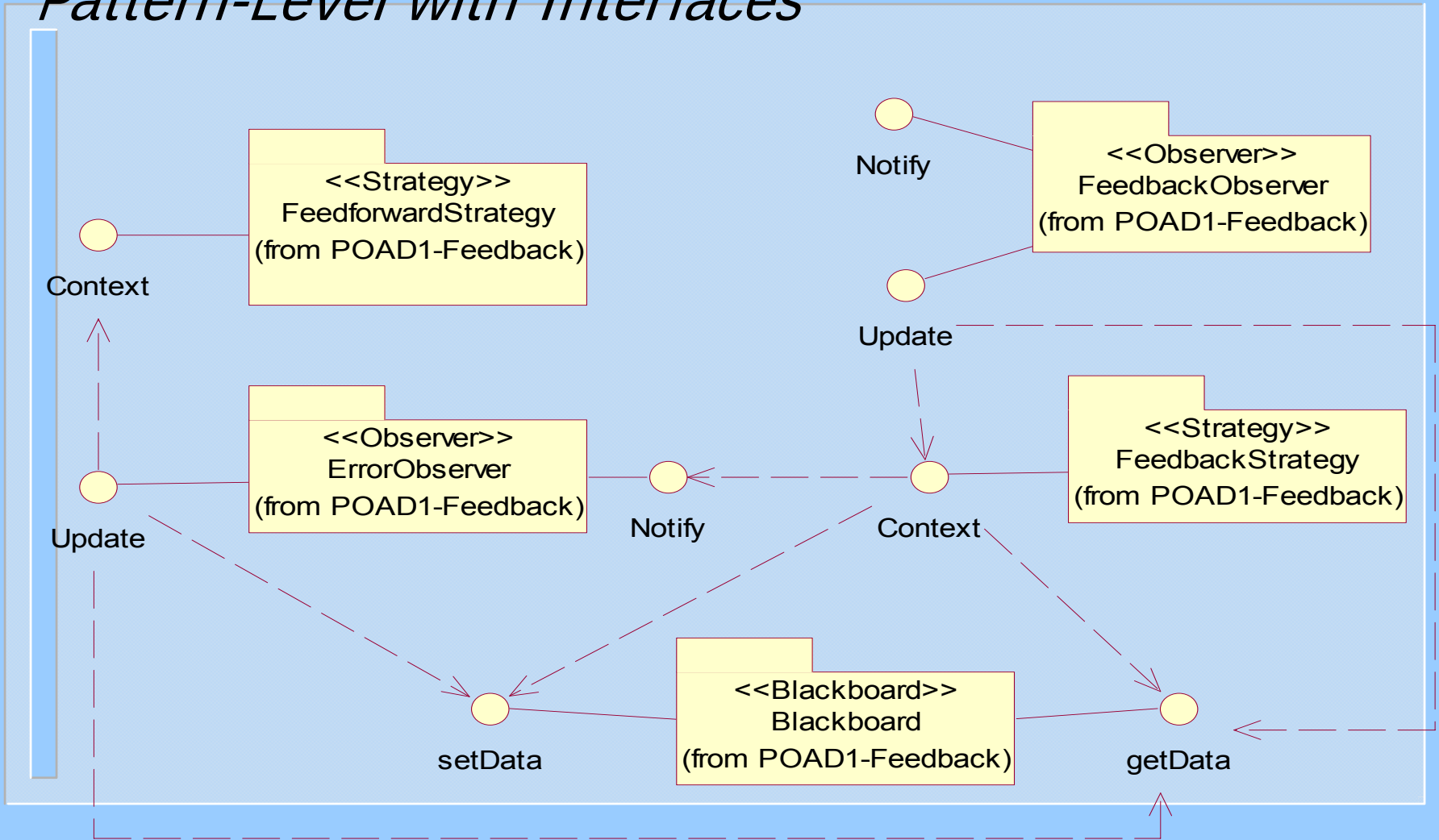
Pattern-Level diagram for feedback control system

This is an architectural pattern based on the data flow architectural style



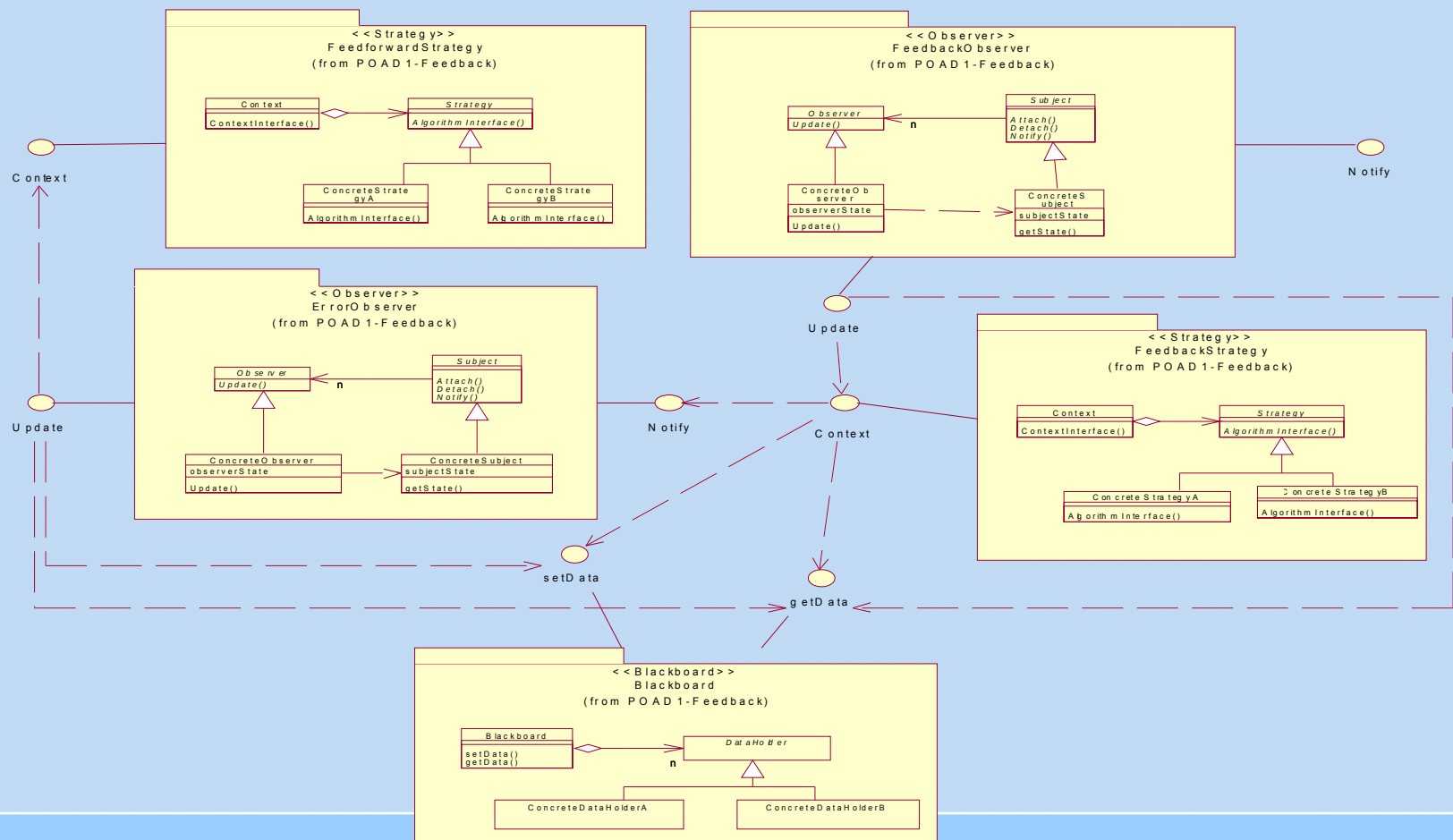
The Feedback Control Example

Pattern-Level with Interfaces



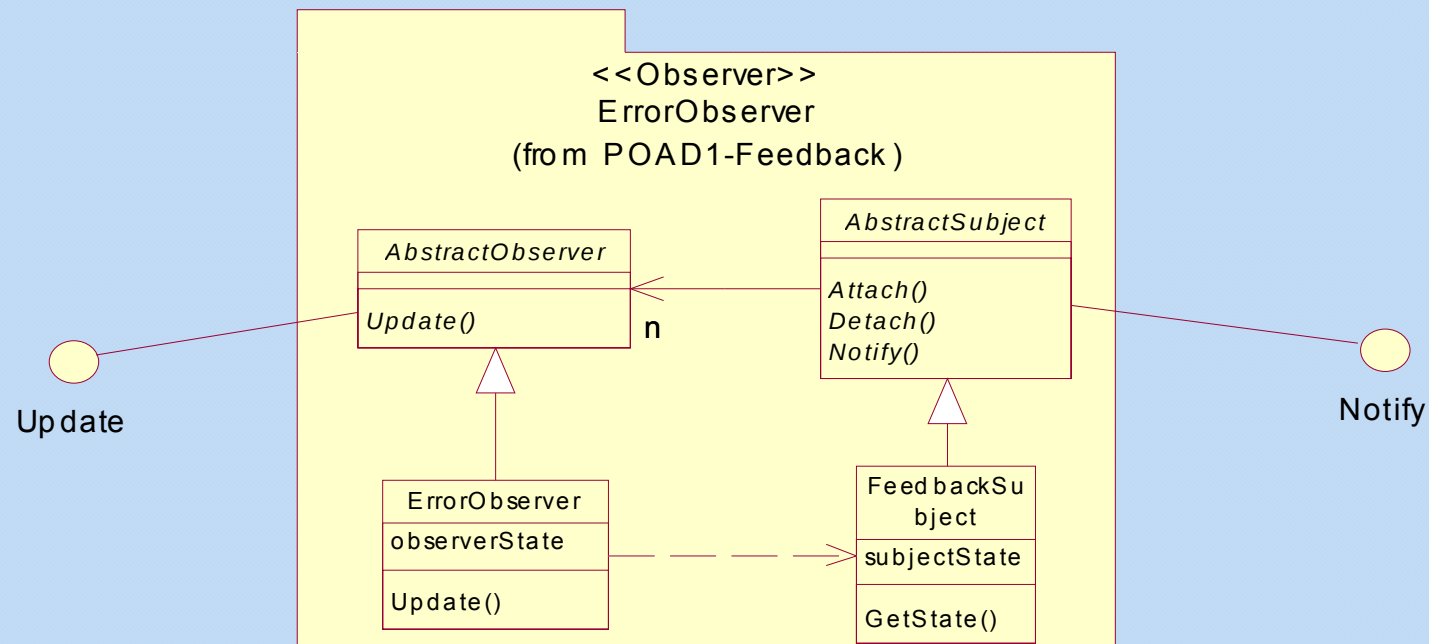
The Feedback Control Example

Detailed Pattern-Level diagram



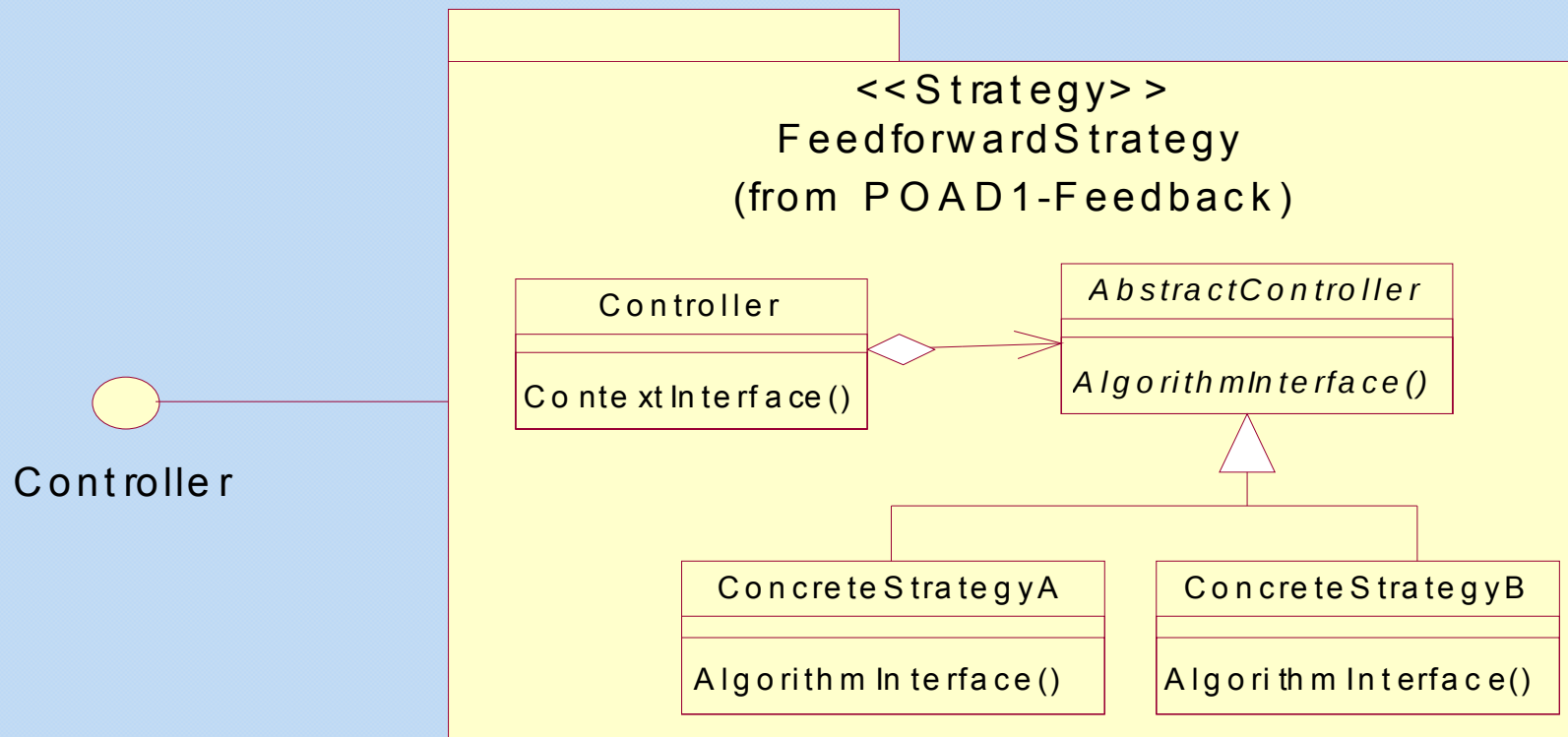
The Feedback Control Example

Instantiating the *ErrorObserver* pattern



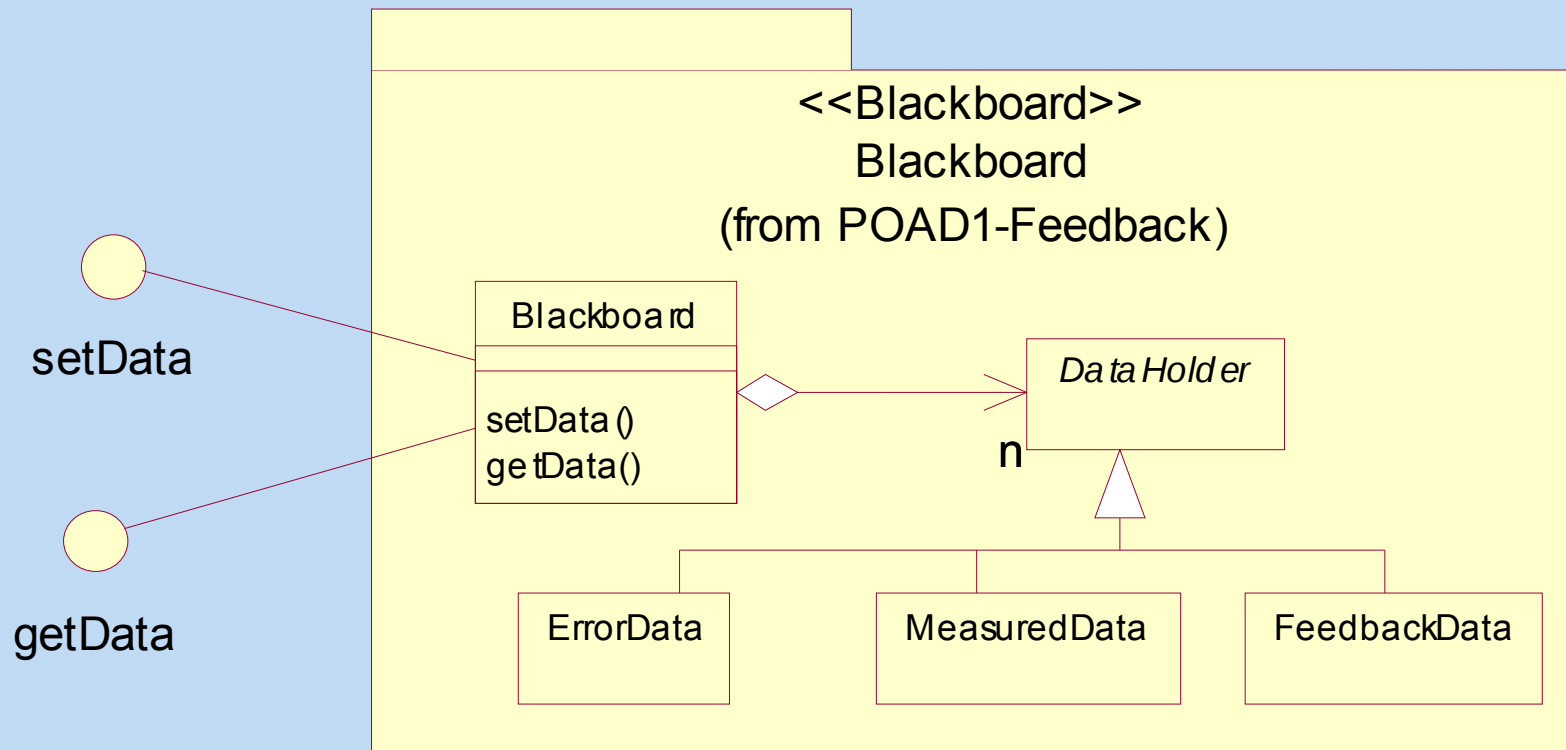
The Feedback Control Example

Instantiating the *FeedforwardStrategy* pattern



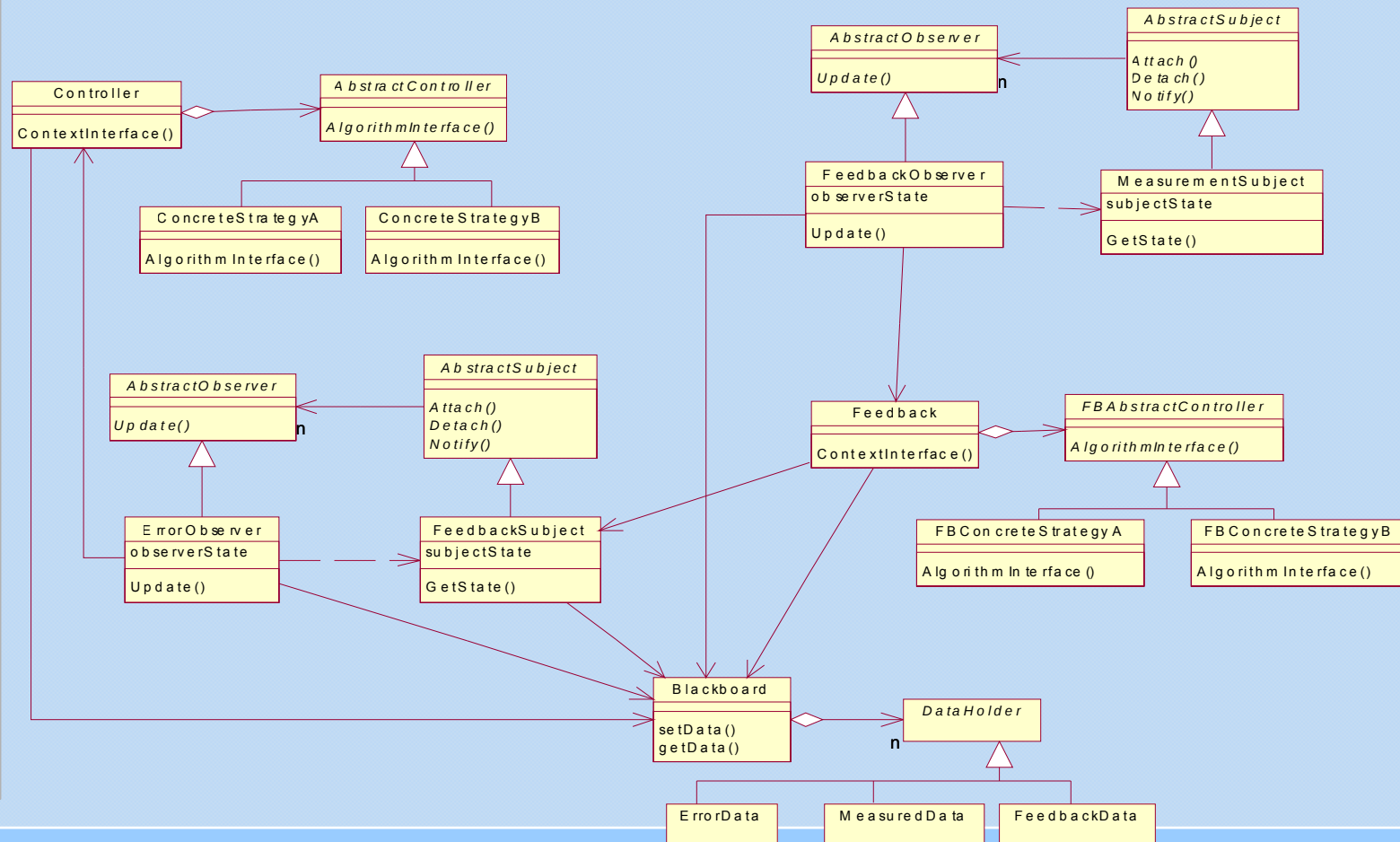
The Feedback Control Example

Instantiating the *Blackboard* pattern



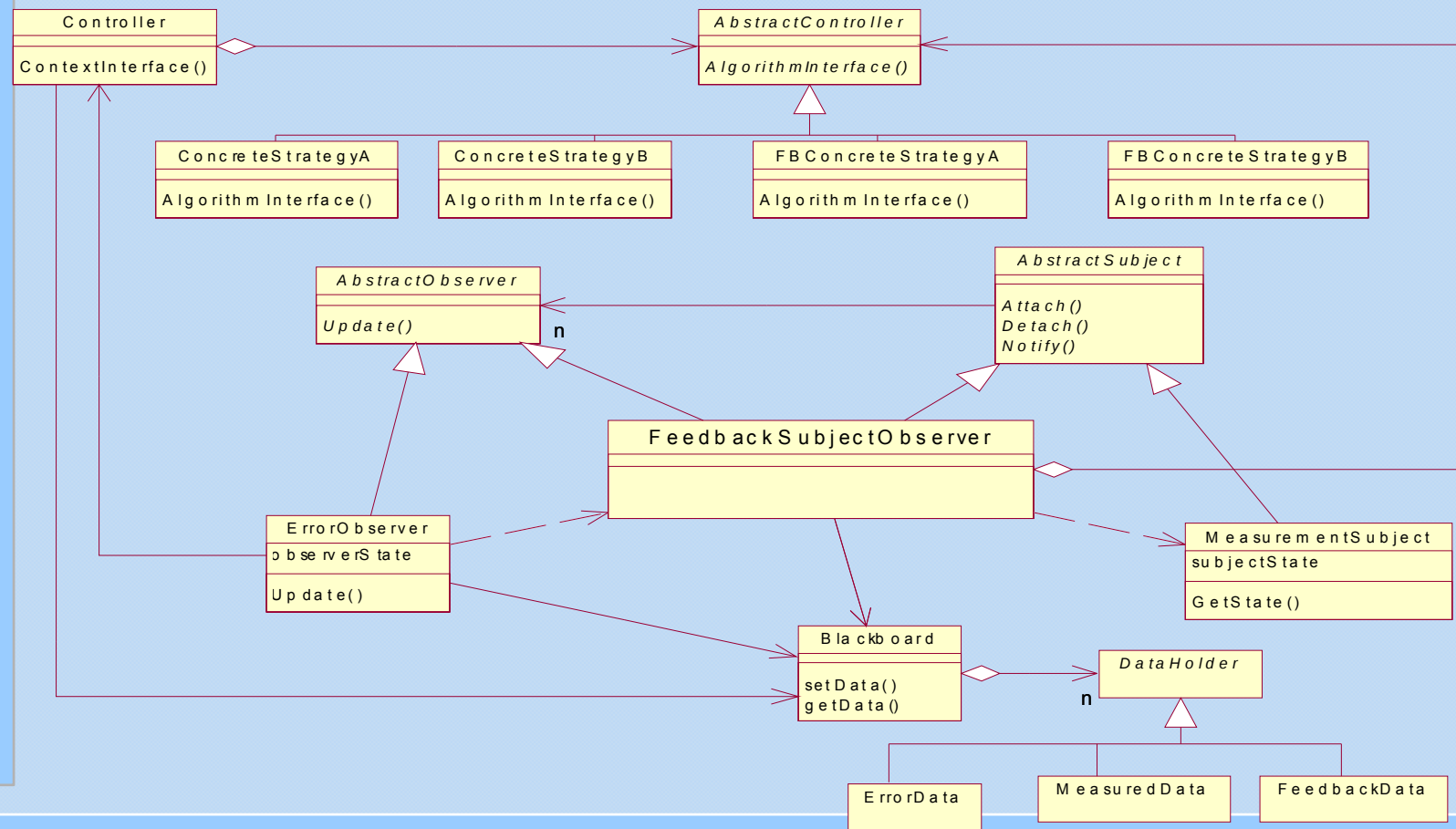
The Feedback Control Example

The initial class diagram



The Feedback Control Example

The refined class diagram ready for code generation



The Feedback Control Example

■ Object Collaboration Diagram

