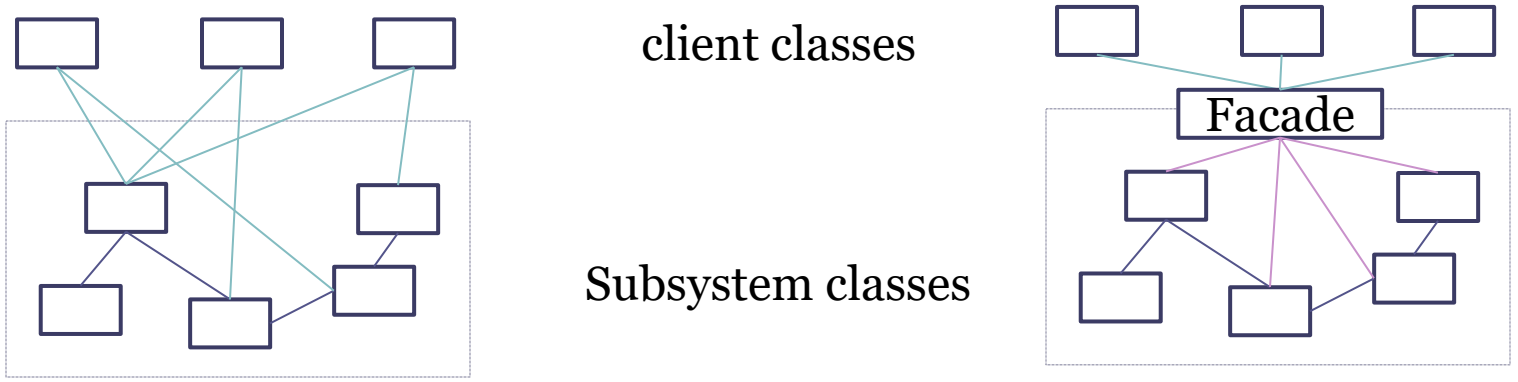
Design Patterns

Façade

Dr. Chad Williams
Central Connecticut State University

Design pattern: Façade

- **Category**: Structural design pattern
- **Intent**:
  - Define a unified interface to a set of interfaces in a subsystem to make the subsystem easier to use.
- **Motivation**
  - Structuring a system into subsystems helps reduce complexity, adding Façade can help reduce dependencies between subsystems

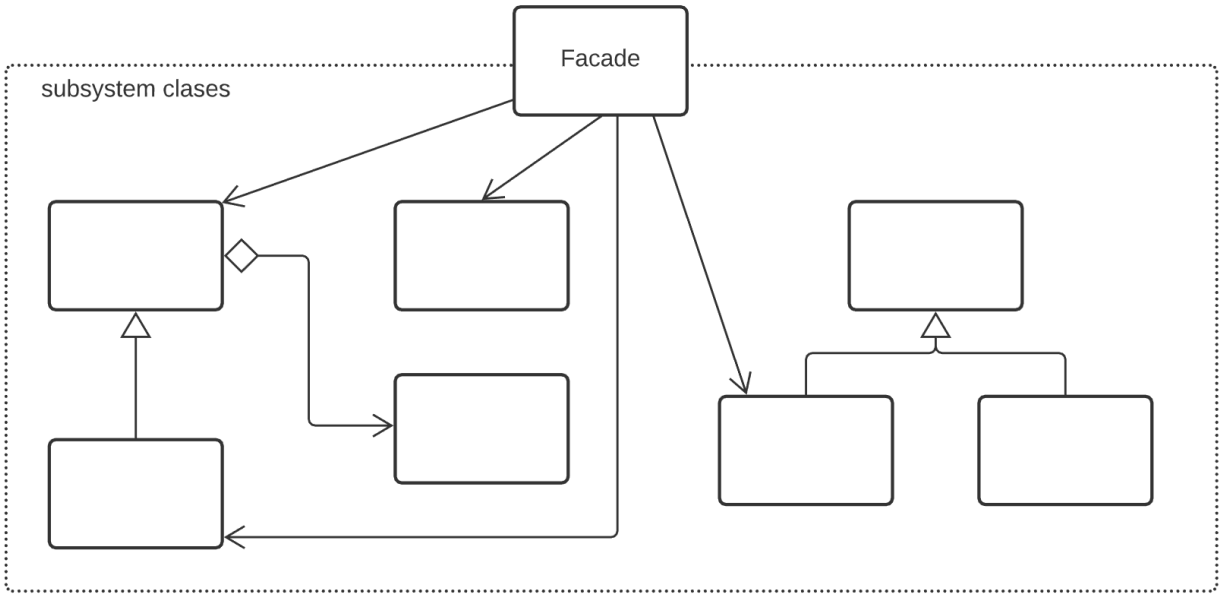


Applicability

- Use the Façade pattern when:
  - Provide a simple interface to a complex subsystem
    - Often times as subsystems evolve can increase number of components internally to improve modularity, but can also make their use more complex, goal is to simplify use for most clients. If clients need more customization can still interact directly outside of Façade
    - Many dependencies between client classes and subsystem abstractions. Façade can help decouple and promote more independence
    - Good for layered abstractions, providing single interface for interactions between layers

Participants

- Façade
  - Knows which subsystems are responsible for request
  - Delegates client requests to appropriate components of the subsystem
- Subsystem classes
  - Implement subsystem functionality
  - Have no knowledge of the Façade, keep no references to it



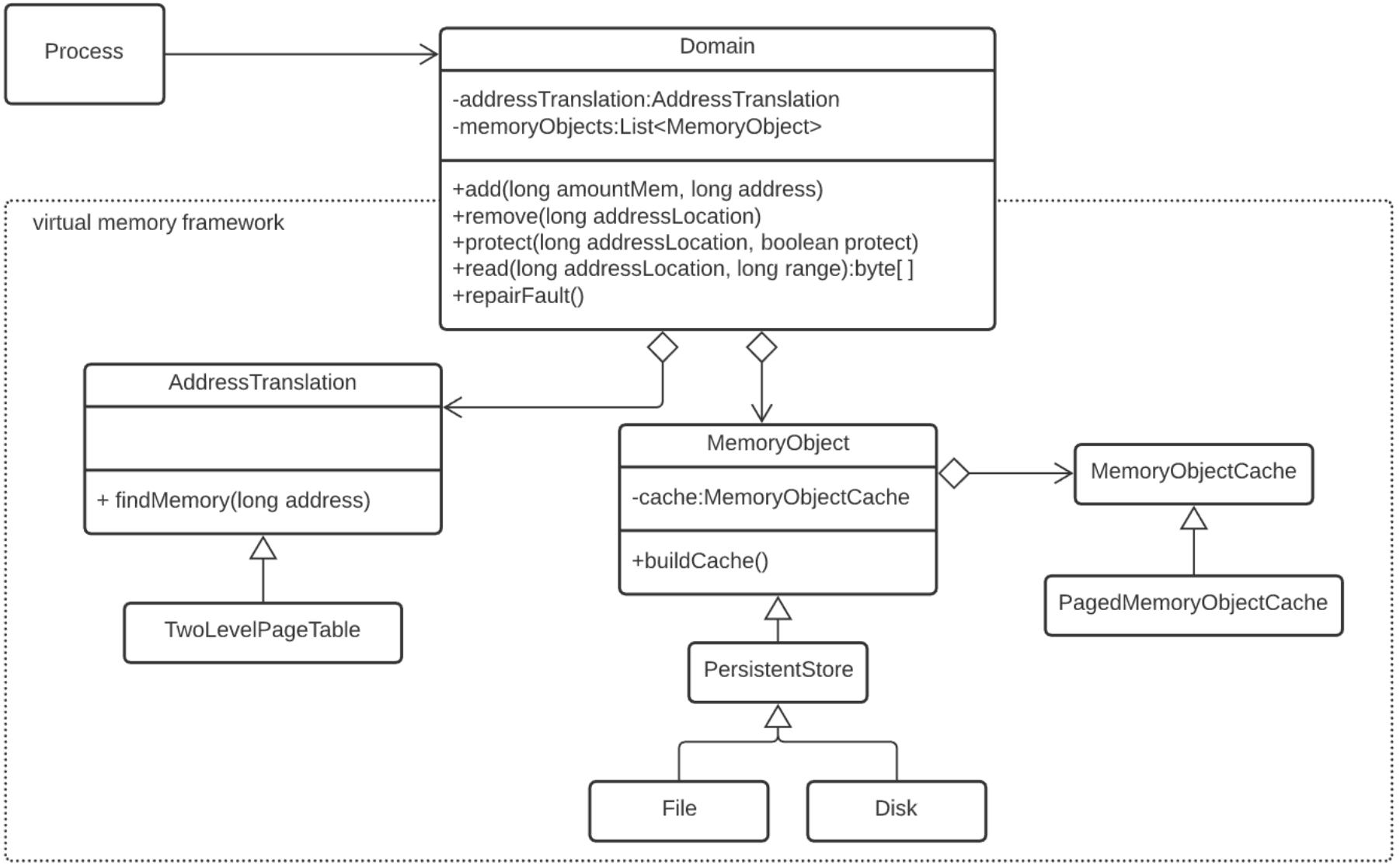
Collaborations

- Client communicates with subsystem by sending requests to Façade
- Façade forwards to appropriate component of subsystem to do the work
  - Façade doesn't do the work, but may need to translate from simple interface request to which subsystems are interacted with
  - Clients that use the Façade do not need to interact with any subsystem objects directly

Consequences

- Benefits
  - Simplifies client interaction
  - Makes subsystem easier to use
  - Promotes weak coupling of clients and subsystem allowing easier modification of subsystem without impacting clients
  - Does not limit clients ability to use full subsystem, but gives choice of Façade for ease of use or access subsystem for generality

Sample Façade UML



More Facade examples

- Video conversion
  - Codec for original, audio file, video file, codec for output, translation
- Network layered architecture
  - Layer provides service subsystem coordinates details for how accomplished for standard calls
- Quick reorder
  - Add items to cart, collect shipping data, process payment, place order