

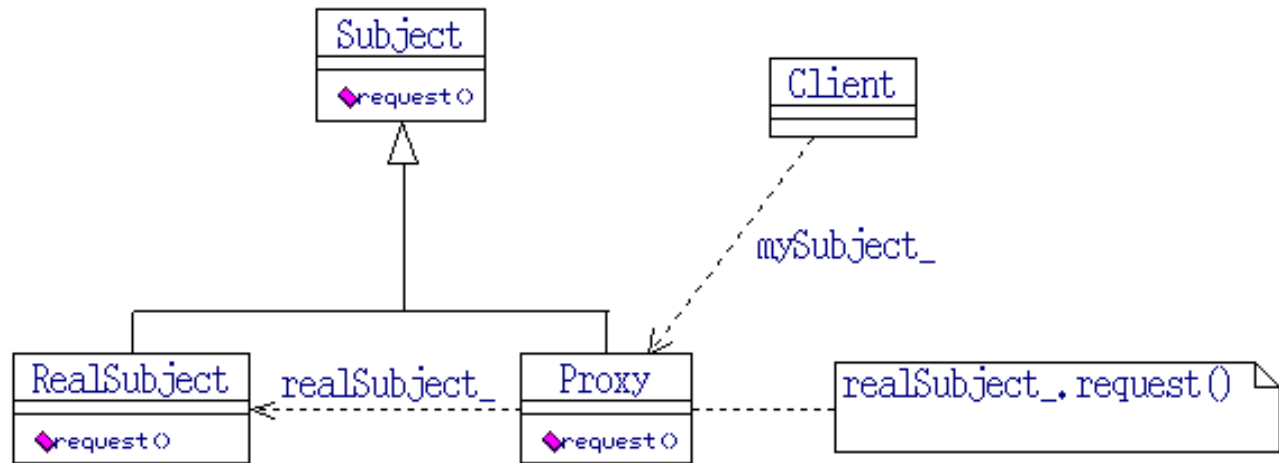
## ■ Design Patterns

- Holder (Proxy)
- Adapter
- Factory

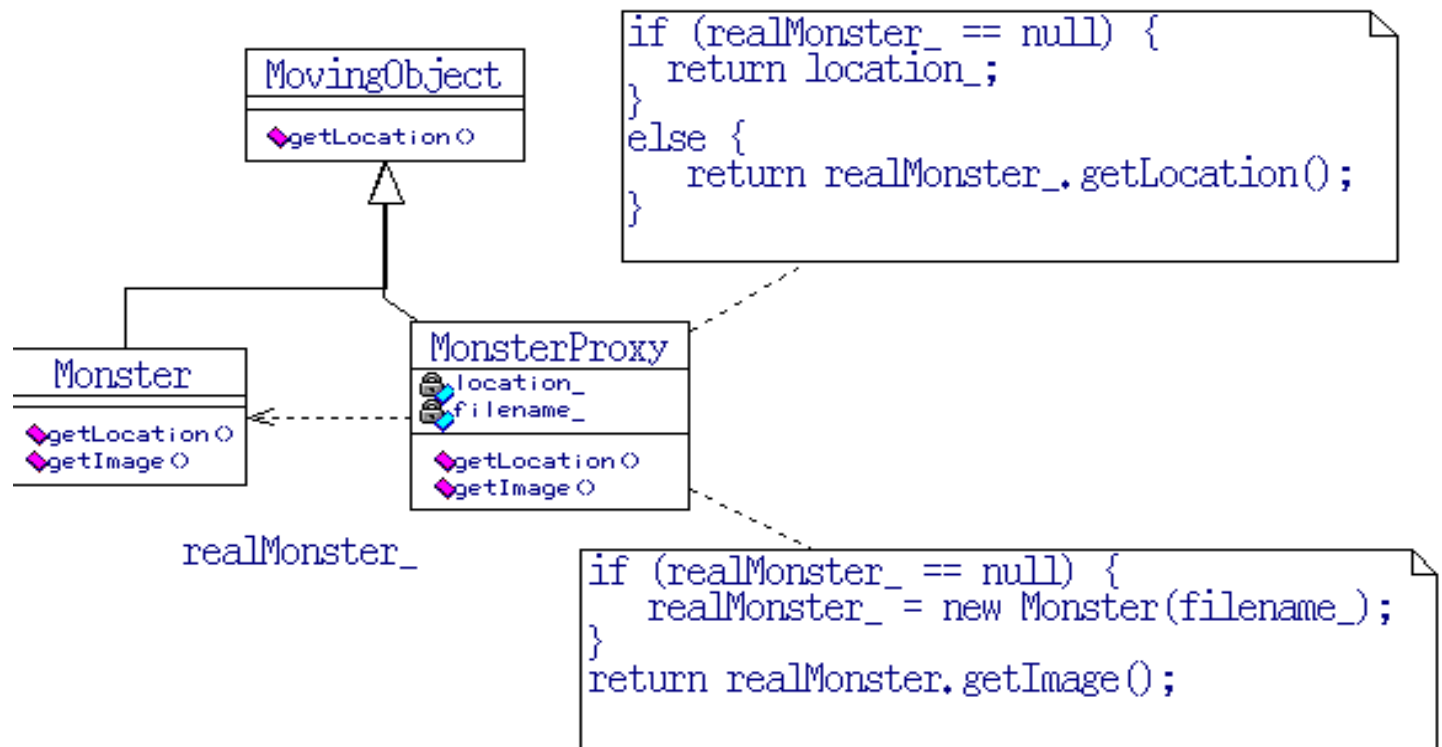
## ■ Proxy

- Placeholder for another subject
- Control access (Protection Proxy)
  - check a requesting object's credentials before returning data object
  - eg. check user and group vs. file protection codes before returning file data
- Perform extra functions (Smart Reference)
  - eg. perform locking of data before allowing it to be accessed or modified
- Allow lazy creation (Virtual Proxy)
  - eg. load an image from a file only when it is needed
- Provide a persistent reference when data is changing

# The Proxy Pattern



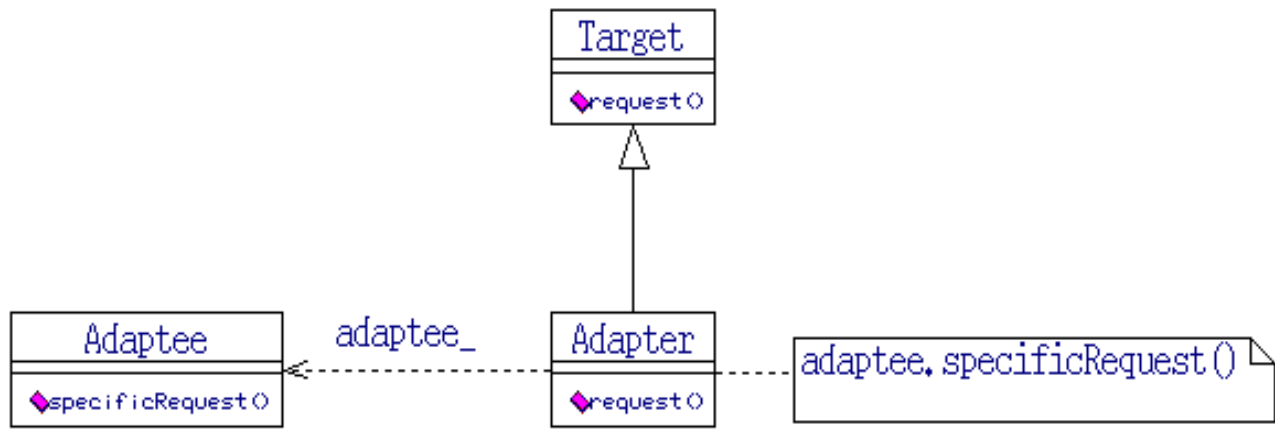
- Example:



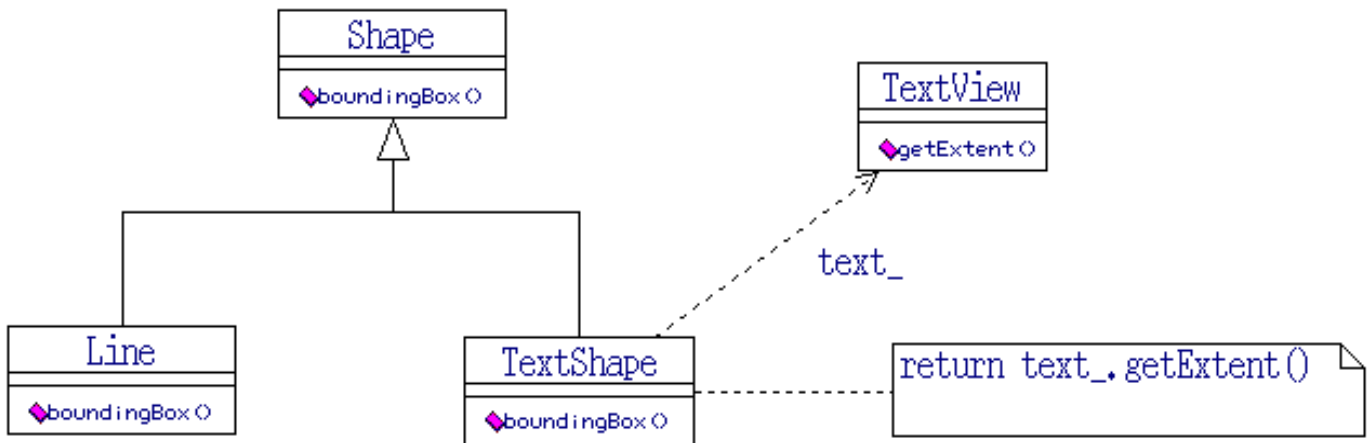
## ■ Adapter

- Convert an interface to look like another interface
- Use existing classes in a new way (under a new interface definition)

# The Adapter Pattern



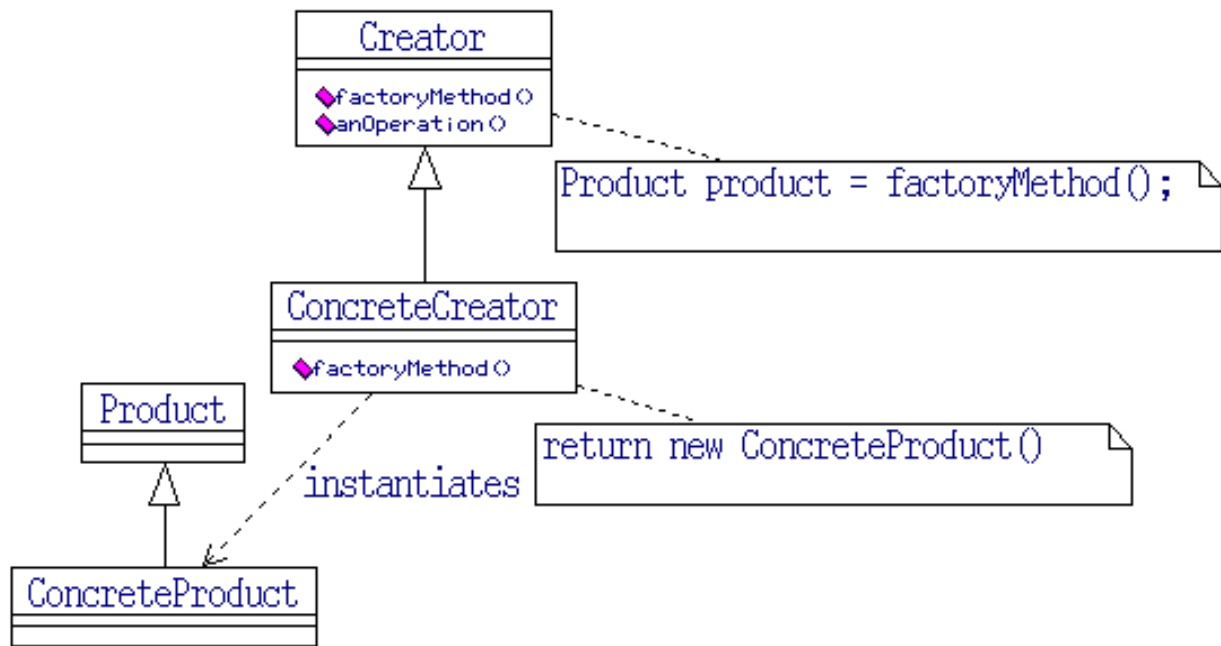
- Example:



## ■ Factory

- Provide an interface for creating a type of object when actual objects instantiated may be of different types
- A factory may also return a different product at different times in response to the same request

# The Factory Pattern



- Example:

