Design Patterns

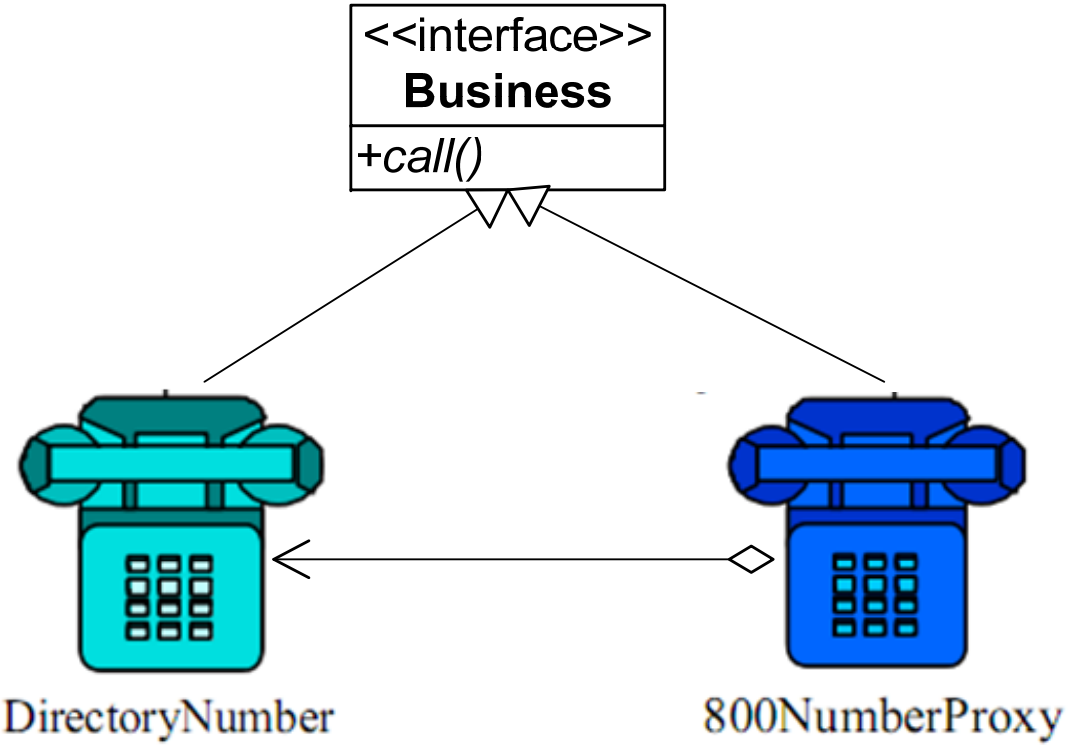
宋 杰
Song Jie
东北大学 软件学院
Software College, Northeastern University

11. Proxy Pattern

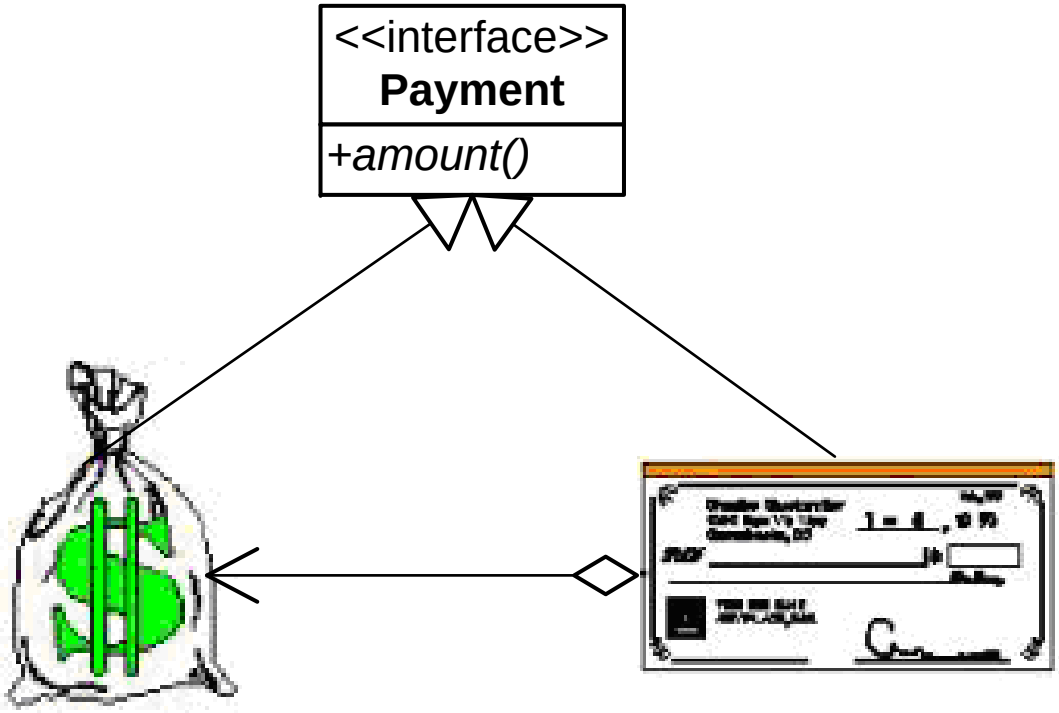
Intent

- Provide a surrogate or placeholder for another object to control access to it.
- 代理模式给某一个对象提供一个代理对象，并由代理对象控制对原对象的引用。

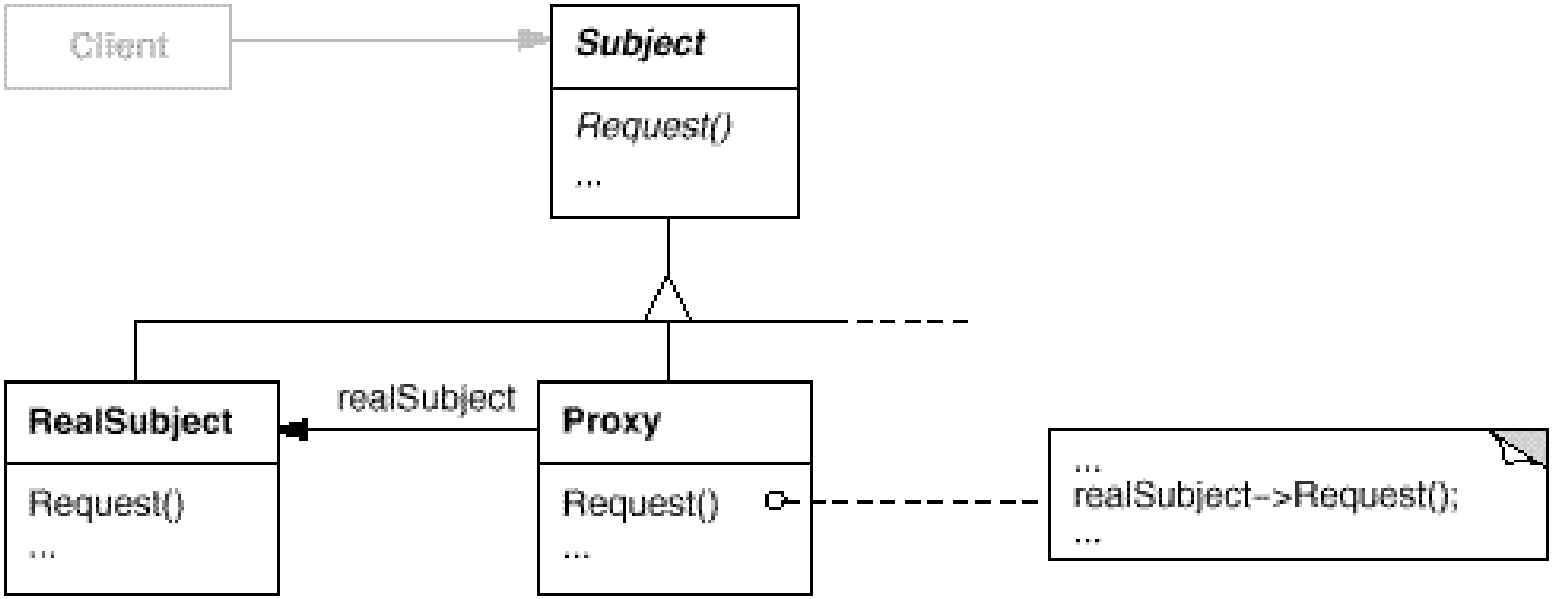
Example


<<interface>>
Business
+call()
DirectoryNumber
800NumberProxy

Example



Structure


Client
Subject
Request()
...
RealSubject
Request()
...
realSubject
Proxy
Request()
...
...
realSubject->Request();
...

Participants

- Proxy
  - Maintains a reference that lets the proxy access the real subject. Proxy may refer to a Subject if the RealSubject and Subject interfaces are the same.
  - Provides an interface identical to Subject's so that a proxy can by substituted for the real subject.
  - Controls access to the real subject and may be responsible for creating and deleting it.
- Subject
  - Defines the common interface for RealSubject and Proxy so that a Proxy can be used anywhere a RealSubject is expected.
- RealSubject
  - Defines the real object that the proxy represents.

Collaborations

- aProxy contains the reference of aRealSubject, thus it can manipulate the aRealSubject in anytime;
- aProxy provides an interface which is in according with the interface of aRealSubject, thus it can replacement the aRealSubject in anytime;
- aProxy controls the reference of aRealSubject, that is, it create and destroy the aRealSubject;
- aProxy forwards requests to aRealSubject when appropriate, depending on the kind of proxy.
- aProxy always add some pre- or post- behaviors when the invocation of clients is delegated to aRealSubject.

Consequences

- The Proxy pattern introduces a level of indirection when accessing an object. The additional indirection has many uses, depending on the kind of proxy:

Proxy Cases (Applicability)

- Remote Proxy: A remote proxy can hide the fact that an object resides in a different address space.
- Virtual Proxy: A virtual proxy can perform optimizations such as creating an object on demand (maybe the creating the real subject is expensive).
- Smart Reference Proxy: It add some additional functionalities when the real subject is invoked, for examples:
  - Recording the consumed time the real object being invoked
  - Counting the number of references to the real object
  - Loading a persistent object into memory when it's first referenced.
  - Checking that the real object is locked before it's accessed to ensure that no other object can change it.
- Protection Proxy (Access Proxy): Protection proxy control the access of real subject (access controlling).
- Copy-on-Write: it is for clone object, it really clone the real subject for client on demand, that is, the status of object is going to be changed.
- Cache Proxy: it provides the cached space for the real subject, thus the clients can store and share more results.
- Firewall Proxy: it filter the hostile requests and hostile data.
- Synchronization Proxy: it provides the functionalities that multiple clients (threads) can access the real subject without conflicts.

Example: Achilles

- Achilles是一个用来测试网站的安全性能的工具软件。
- 当Achilles处于截取状态时，它会向客户端假装是服务器，同时向真正的服务器假装是浏览器，在两端商议SSL通信。
- Achilles可以破解加密的数据，给Achilles的用户显示已经解密的内容，并且允许用户更改处于通信过程中的数据。

Example: Shortcut of Windows

- Alias in Macintosh
- Link in UNIX
- Shortcut in Windows

Example: 替考

- 有一位同学（考生）请求另一位同学（枪手 ）代替他参加一个英文水准考试。
- Who is Proxy and Who is RealSubject?
  - 枪手 is RealSubject;
  - 考生 is Proxy;

Extension: Transparent proxy

- Proxy doesn't always have to know the type of real subject.
- If a Proxy class can deal with its subject solely through an abstract interface, then there's no need to make a Proxy class for each RealSubject class; the proxy can deal with all RealSubject classes uniformly.
- But if Proxy is going to instantiate, then they have to know the concrete class.

Let’s go to next…