

Spring ve Proxy Örüntüsü

Proxy Örüntüsü Nereelerde Kullanılır?

- Bazı durumlarda **nesnelerin hemen yaratılması** uygun olmayabilir
 - Örneğin request veya session scope bean'ler
- Ya da asıl nesneye erişim gerçekleşmeden önce veya erişimden sonra **ilave bir takım işlemlerin yapılması** söz konusu olabilir
 - Transaction başlatılması/sonlandırılması,
 - Güvenlik kontrolü,
 - Validasyon, caching gibi

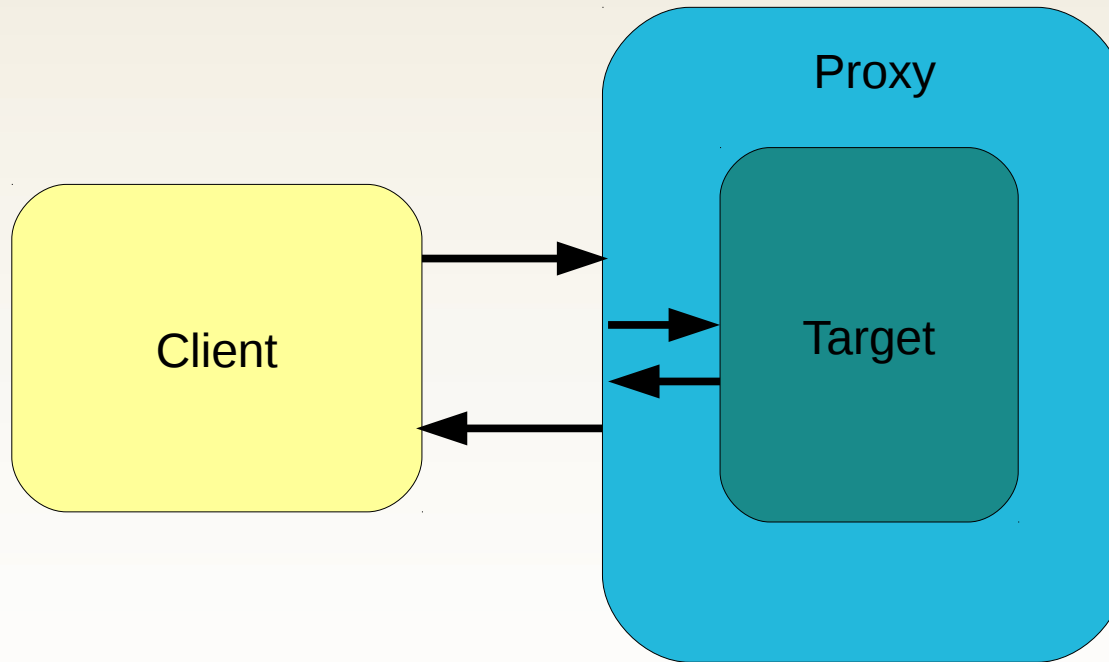
Proxy Örüntüsünün Özellikleri

- **Asıl nesnenin yerine geçen**, yaratılmasını ihtiyaç anına veya uygun ortam oluşana kadar erteleyen, onun yerine ilave kabiliyetleri gerçekleştiren **başka bir nesne** yaratılır
- Bunun adı **proxy nesnedir**
- **Asıl nesneye erişim** proxy üzerinden dolaylı biçimde gerçekleşir
- Proxy ile target nesne **aynı türdendir**
- Dolayısı ile **istemciler** proxy'nin varlığından **habersizdir**

Proxy Örüntüsünün Yapısı

Proxy, target nesne ile aynı tipte olup, client ile target nesnenin arasına girer

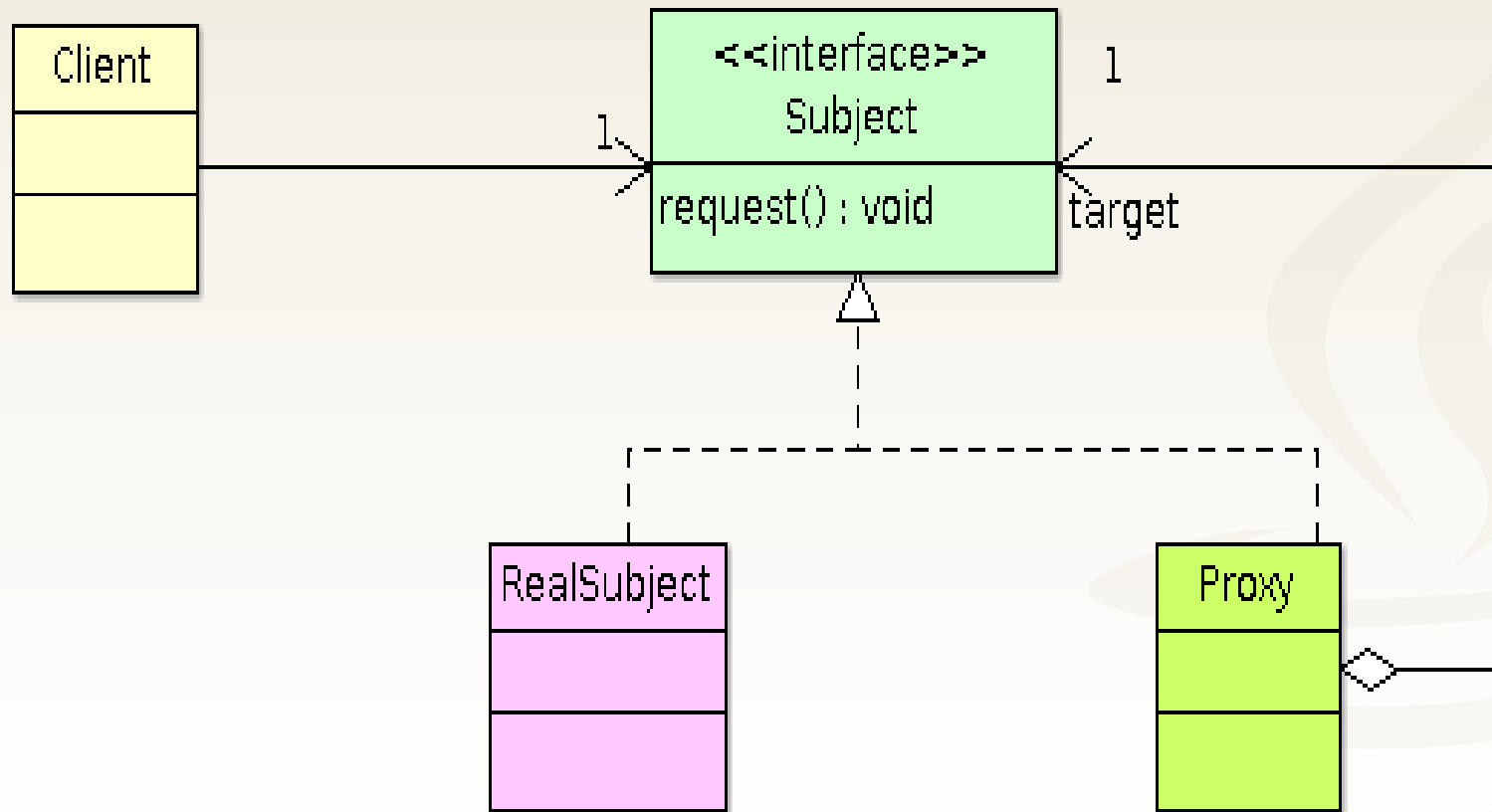
Client proxy nesne ile konuştuğunun farkında değildir



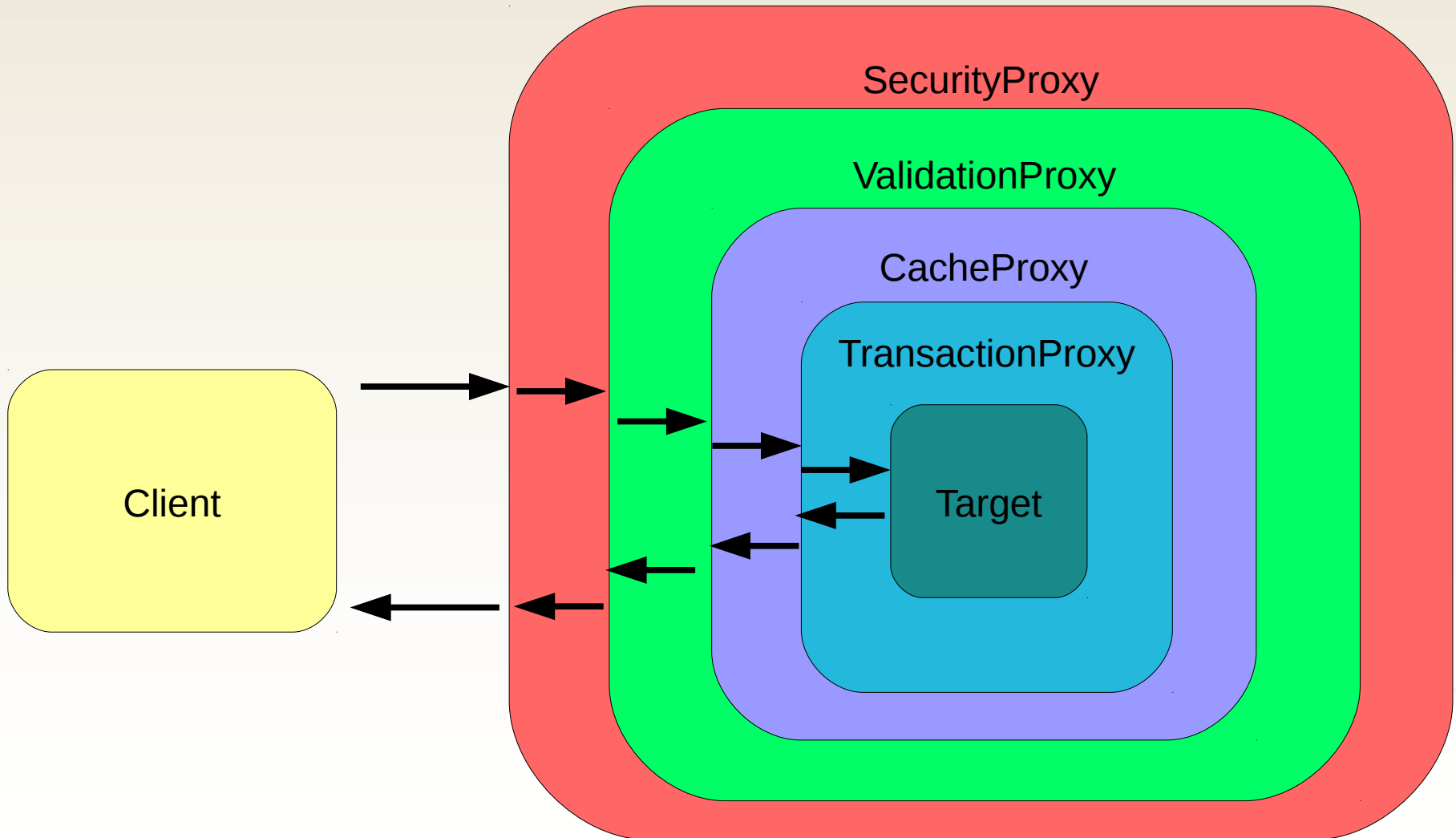
Client'in target nesne üzerindeki metot çağrıları öncelikle proxy nesneye erişir

Proxy metot çağrısından önce veya sonra bir takım işlemler gerçekleştirebilir

Proxy Örüntüsü Sınıf Diagramı



İç İçe Proxy Nesne Zinciri



Spring İçerisinde Proxy Örüntüsünün Kullanımı

- Spring Application Framework'ün **pek çok kabiliyeti** proxy örüntüsü üzerine kuruludur
 - Transaction yönetimi
 - Bean scope kabiliyeti (request ve session scope bean'ler)
 - Aspect oriented programlama altyapısı (Spring AOP)
 - Metot düzeyinde validasyon ve caching
 - Remoting
 - Spring security'de metot düzeyinde yetkilendirme

Spring İçerisinde Proxy Örüntüsünün Kullanımı

- Spring bu kabiliyetleri hayata geçirmek için genellikle uygulama geliştiricilerden habersiz **otomatik olarak proxy oluşturma** işini gerçekleştirir
- Diğer bean'lere de **bağımlılık olarak proxy nesne enjekte** edilir
- Diğer bean'ler proxy ile çalıştıklarının farkında **değillerdir**

Proxy Oluşturma Yöntemleri

- **Interface Proxy**

- Asıl nesnenin sahip olduğu arayüzlerden birisi kullanılır
- JDK proxy olarak da bilinir

- **Class Proxy**

- Asıl nesnenin ait olduğu sınıf extend edilerek gerçekleştirilir
- CGLIB/Javassist proxy olarak da bilinir

İletişim

- **Harezmi** Bilişim Çözümleri
- Kurumsal Java Eğitimleri
- <http://www.java-egitimleri.com>
- info@java-egitimleri.com

