

## Шаблон Proxy

//на бг. този шаблон се нарича Пълномощно, но тук ще се придържаме към англ. вариант.

В шаблона прокси се поддържат обекти които контролират създаването и достъпа до други обекти. Обикновено ролята на прокси играе малък (public) обект, който се намира в по-сложни (private) обекти и се активира веднъж при определени обстоятелства.

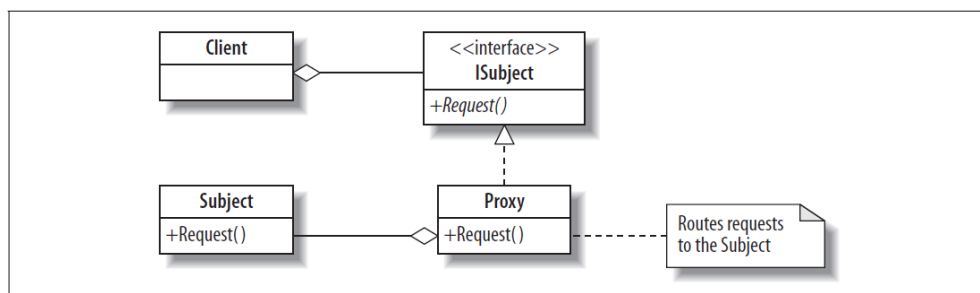
*Пример:*

Съществуват различни системи които изискват права за достъп до определени ресурси и всички те могат да се разглеждат като примери за използването на прокси. Например Facebook – много хора се регистрират само за да видят съдържанието което другите са публикували, но практически те не участват. При добра политика всеки новорегистриран започва с обикновена страница и не му се предоставя възможност до всичко преди да е добавил приятели или да публикува.

Друга отличителна черта на тези системи е, че първо трябва да се регистрираме към системата и след това да влезнем (log on) в някоя сесия. След като влезнем имаме достъп до страниците на нашите приятели. Всички действия които можем да изпълняваме (poke, write on walls, send gifts, и т.н.) са интерпретирани от браузъра и се изпращат по мрежата до най-близкия Facebook сървър. Обектите са страниците на Facebook, а прокситата са механизмите които позволяват регистрацията и входа.

### Структура

Можем да видим участниците в този шаблон на следната диаграма:



Това са:

**ISubject** – Общ интерфейс за потребители и проксита, който позволява те да бъдат взаимнозаменяеми

**Subject** – Класът които се представлява от проксита

**Proxy** – Класът който създава, контролира, завишава сигурността за удостоверен (authenticate) достъп до Subject

**Request** – Операцията към Subject, която се използва през прокси

Вижда се че централния клас Proxy имплементира интерфейса ISubject. A Client може да използва ISubject за абстракция за да се скриват прокситата. Въпреки това, понякога може да се освободим от този интерфейс. Всеки обект на Proxy поддържа референция към Subject, където е същината на цялото действие. Проксита изпълнява работа като изглед (frontend). Неговата важност може да се повиши като се направи Subject да е private клас, така че Client да няма достъп до Subject освен чрез Proxy.

**Има няколко варианта на прокси които се използват:**

- Виртуални (Virtual proxies) – Позволява създаването на един обект преди друг (полезно е когато създаването на обекта е бавно или може да се окаже ненужно)
- За удостоверяване (Authentication proxies) - Проверява дали разрешенията за достъп са коректни;
- Отдалечени (Remote proxies) – Закодира заявките и ги изпраща по мрежата;
- Интелигентни (Smart proxies) – Добавя или променя нещо към заявките преди да ги изпрати.

Ако трябва да илюстрираме със Facebook:

- Забавяне на зареждането на някои елементи (virtual proxy)
- Влизане в системата (authentication proxy)
- Изпращане на заявки през мрежата (remote proxy)
- Изпълняване на действия към приятелите (smart proxy)