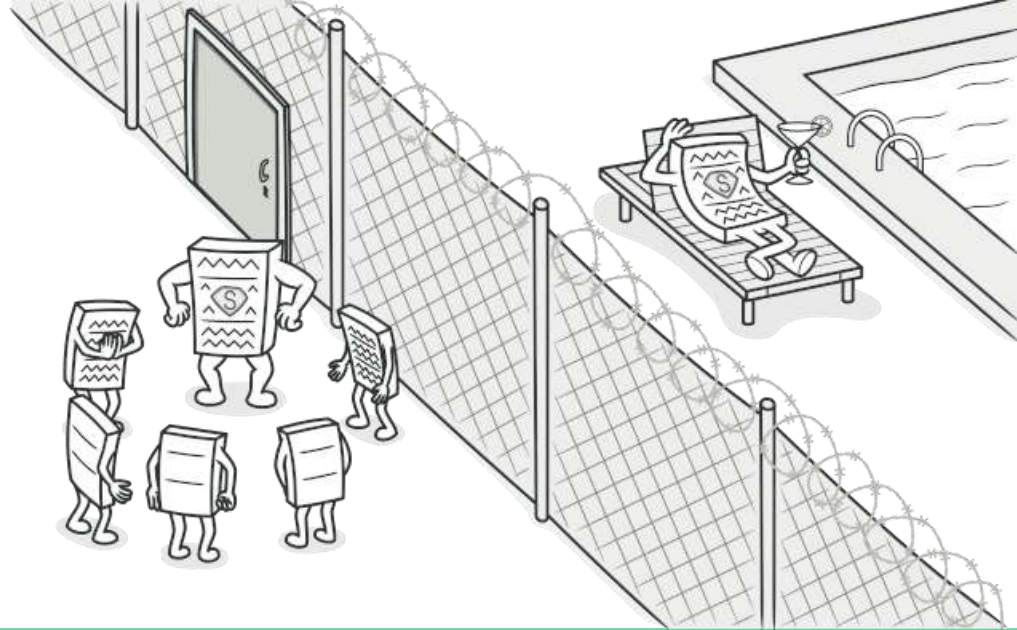


# Proxy

Čeněk Řehoř



# Idea



Co má pan prezident ke  
štědrovečerní večeři?



Pan prezident o  
Vánocích večeří  
rybí polévku,  
bramborový salát  
a kapra.



Prezidentská kancelář  
(Proxy)

Hej Miloši, co máš na  
Vánoce k večeři?

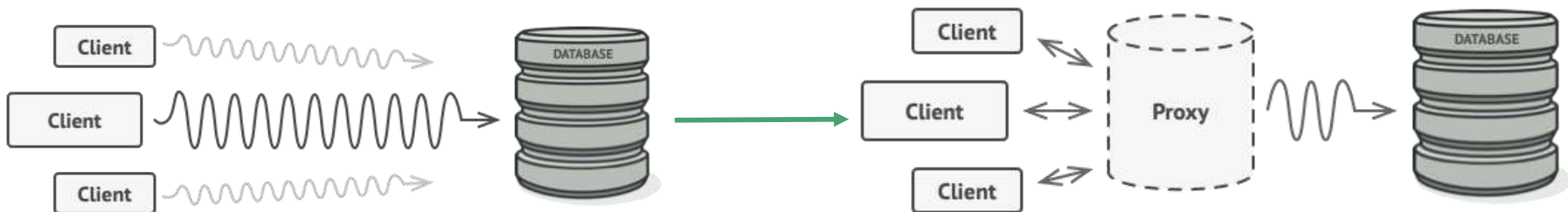


Rybí polévku a  
kapra se salátem.

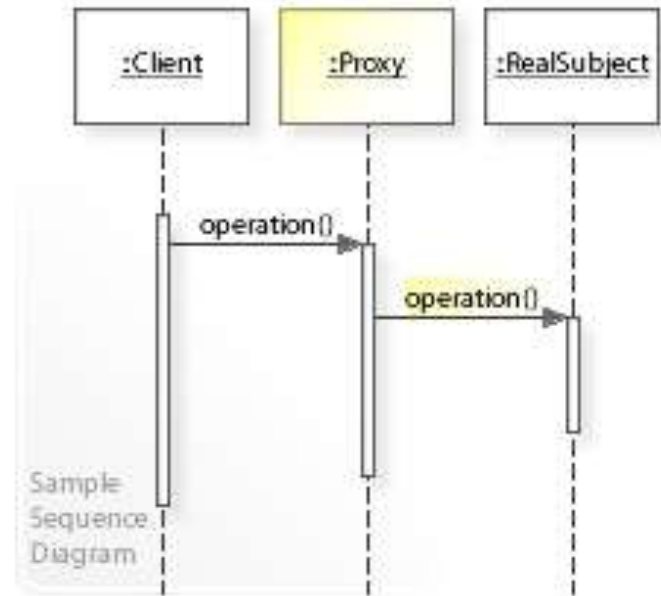
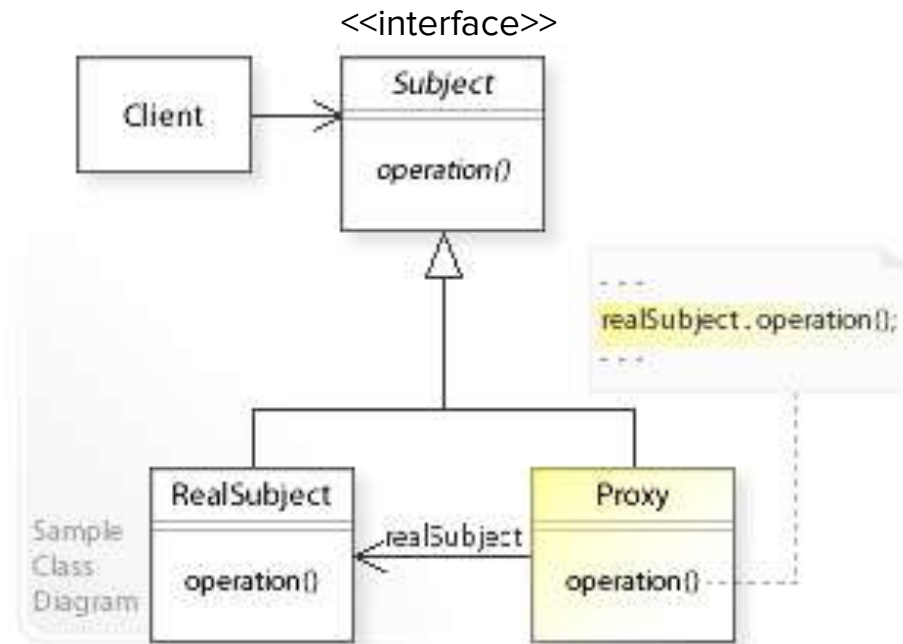


# Podrobnosti

- Strukturální návrhový vzor
- Alternativní název Surrogate (náhradník)
- Prostředník operací nad objektem, který výrazně nemění zvenku pozorovatelné chování
  - Zachovává původní rozhraní
- Příklady:
  - Zastupování vzdálených objektů
  - Efektivnější přístup k objektu (cache, lazy-loading)
  - Kontrola práv přístupu k objektu
  - Logování přístupu k objektu



# Princip



# Podobné návrhové vzory

- Adapter
  - Adapter implementuje jiné rozhraní, proxy implementuje stejné
- Decorator
  - Můžou mít podobnou implementaci
  - Rozdílný účel - decorator přidává objektu vlastnosti, proxy zprostředkovává již existující
- Facade
  - Oba vzory přidávají vrstvu navíc mezi klienta a objekt/subsystém
  - Facade poskytuje zjednodušené rozhraní, proxy to samé

# Typy

- Protection proxy
  - Kontrola či omezování přístupu ke skutečnému objektu

# Protection proxy



Pije pan prezident alkohol?



To je zajímavá otázka, vy ano?



# Protection proxy

```
interface Page
{
    public function visit(AppContext $context);
}

class AdminPage implements Page
{
    private Page $page;

    public function __construct(Page $page) { ... }

    public function visit(AppContext $context)
    {
        if ($context->user()->role != 'admin') {
            return $context->errorResponse(403);
        }

        return $this->page->visit($context);
    }
}
```

Přepsaná funkce z rozhraní

Konec s chybou, pokud uživatel nemá  
dostatečná oprávnění

Volání funkce na původním, nechráněném objektu



# Typy

- Protection proxy
  - Kontrola či omezování přístupu ke skutečnému objektu
- Remote proxy
  - Lokální zástupce vzdáleného objektu (à la RPC, COM, CORBA, Java RMI)
  - Bankomat může mít proxy objekty pro přístup k bankovním datům na vzdáleném serveru

# Remote proxy

RealSubject

```
public interface PrezidentInterface extends Remote {  
  
    public String zeptejSe(String otazka) throws RemoteException;  
  
}
```

Subject

```
public class PrezidentMilos extends UnicastRemoteObject  
    implements PrezidentInterface {  
  
    public PrezidentMilos() throws RemoteException {...}  
  
    @Override  
    public String zeptejSe(String otazka) {...}  
  
}  
  
public class MilosRMI {  
  
    public static void main(String[] args) throws Exception {  
        LocateRegistry.createRegistry(1099);  
  
        Naming.rebind(  
            "//localhost/PrezidentskaKancelar",  
            new PrezidentMilos()  
        );  
    }  
  
}
```

```
public class Redakce {
```

Client

```
    public static void main(String[] args) throws Exception {  
        PrezidentInterface prezident = (PrezidentInterface)  
            Naming.lookup("//localhost/PrezidentskaKancelar");  
        System.out.println(prezident.getClass());  
        System.out.println(prezident.zeptejSe("Je pan prezident vzhůru?"));  
    }  
}
```

Output:

```
--- exec:3.1.0:exec (default-cli) @ MilosRMI ---  
class jdk.proxy1.$Proxy0  
Zzzz...
```

Rozhraní implementuje zástupná třída

# Typy

- Protection proxy
  - Kontrola či omezování přístupu ke skutečnému objektu
- Remote proxy
  - Lokální zástupce vzdáleného objektu (RPC, COM, CORBA, Java RMI)
  - Bankomat může mít proxy objekty pro přístup k bankovním datům na vzdáleném serveru.
- Virtual proxy
  - Zastupuje objekt jednodušším objektem, dokud není potřeba vyrobit skutečný.
  - Nahrávání obrázků on demand (lazy-loading)

# Virtual proxy

```
public static class EditorWindow extends JFrame {  
  
    private final RenderableImage openFileIcon;  
    private final RenderableImage saveFileIcon;  
  
    private boolean menuBarVisible = false;  
  
    public EditorWindow() {  
        openFileIcon = openImage("icons/open.png");  
        saveFileIcon = openImage("icons/save.png");  
    }  
  
    public void openMenuBar() {  
        menuBarVisible = true;  
    }  
  
    @Override  
    public void paint(Graphics g) {  
        super.paint(g);  
  
        if (menuBarVisible) {  
            openFileIcon.render(g, 0, 0);  
            saveFileIcon.render(g, 10, 0);  
        }  
    }  
}
```

Client

Subject

```
private static interface RenderableImage {  
    public void render(Graphics g, int x, int y);  
}  
  
private static class RealImage  
    implements RenderableImage {  
  
    private Image imageData;  
  
    public RealImage(String path) {  
        imageData = ImageIO.read(new File(path));  
    }  
  
    @Override  
    public void render(Graphics g, int x, int y) {  
        if (imageData != null) {  
            g.drawImage(imageData, x, y, null);  
        }  
    }  
}
```

RealSubject

```
private static class DeferredImage  
    implements RenderableImage {  
  
    private final String path;  
    private RealImage realImage;  
  
    public DeferredImage(String path) {  
        this.path = path;  
    }  
  
    @Override  
    public void render(Graphics g, int x, int y) {  
        if (realImage == null) {  
            realImage = new RealImage(path);  
        }  
        realImage.render(g, x, y);  
    }  
}
```

Proxy

# Typy

- Protection proxy
  - Kontrola či omezování přístupu ke skutečnému objektu
- Remote proxy
  - Lokální zástupce vzdáleného objektu (RPC, COM, CORBA, Java RMI)
  - Bankomat může mít proxy objekty pro přístup k bankovním datům na vzdáleném serveru.
- Virtual proxy
  - Zastupuje objekt jednodušším objektem, dokud není potřeba vyrobit skutečný.
  - Nahrávání obrázků on demand (lazy-loading)
- Cache proxy
  - Cachuje výsledky metod původního objektu

# Cache proxy



Jakým autem jezdí pan prezident?



Jirkova paměť (cache)

Pan prezident jezdí  
vozem Škoda Superb.

Čím drandíš, Mildó?



No Superbem,  
ne?



# Typy

- Protection proxy
  - Kontrola či omezování přístupu ke skutečnému objektu
- Remote proxy
  - Lokální zástupce vzdáleného objektu (RPC, COM, CORBA, Java RMI)
  - Bankomat může mít proxy objekty pro přístup k bankovním datům na vzdáleném serveru.
- Virtual proxy
  - Zastupuje objekt jednodušším objektem, dokud není potřeba vyrobit skutečný.
  - Nahrávání obrázků on demand (lazy-loading)
- Cache proxy
  - Cachuje výsledky metod původního objektu
- A další...

# Výhody a nevýhody

## Výhody

- Výkon - virtual proxy, optimalizace strategie přístupu k objektu
- Dělení kódu - administrativní kód oddělen od funkcionality
- Bezpečnost
- Kontrola objektu bez vědomí klienta

## Nevýhody

- Vrstva navíc - snižuje výkon a přidává komplexitu do kódu
  - ...ale to by se stejně stalo i bez použití proxy



# Shrnutí

- kontrola přístupu k datům
- práce se vzdálenými objekty
- lazy-loading, caching
- chytré ukazatele
- a to vše s původním rozhraním!



Víte, co je to „proksy“,  
pane moderátore?

Děkuji za  
pozornost