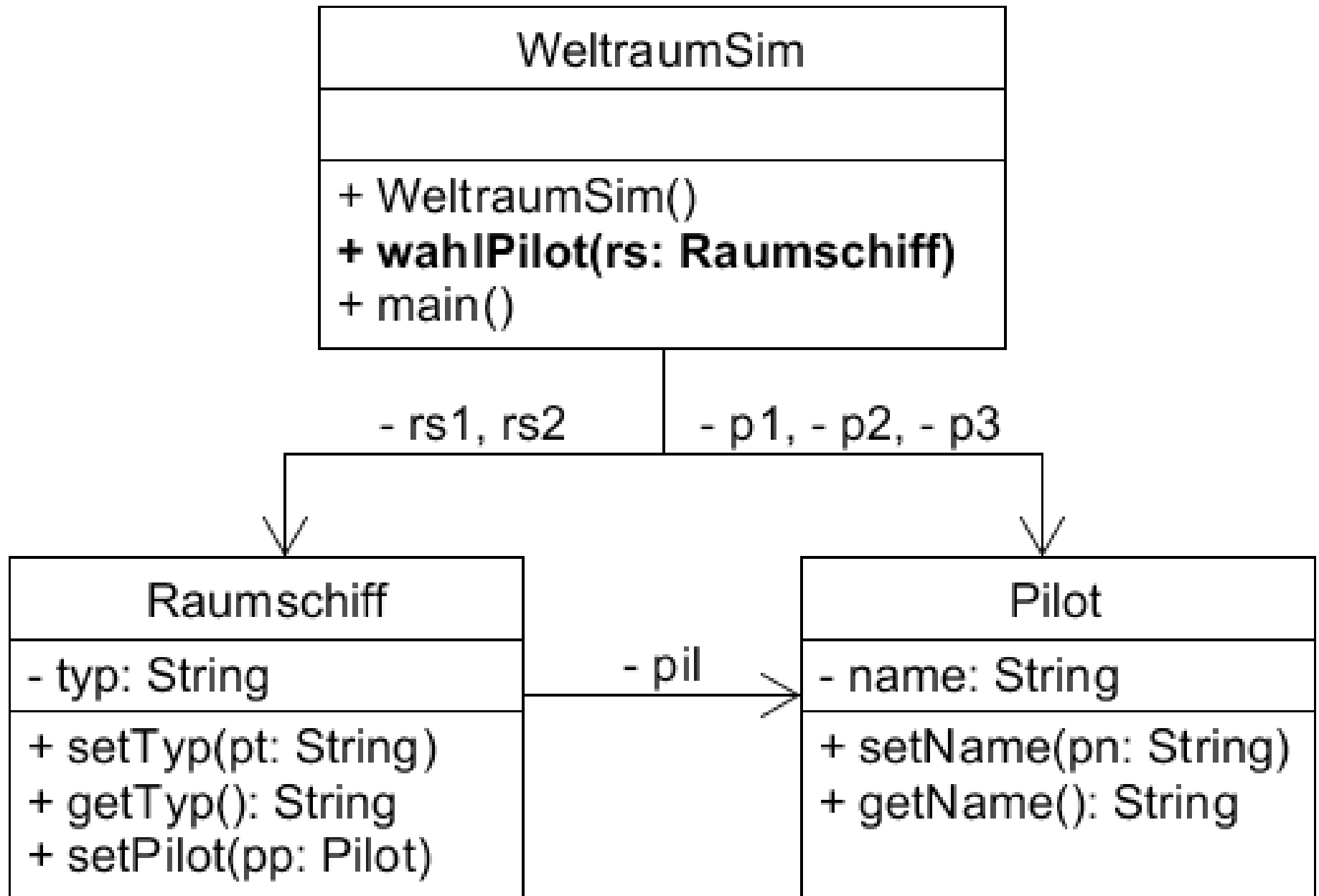
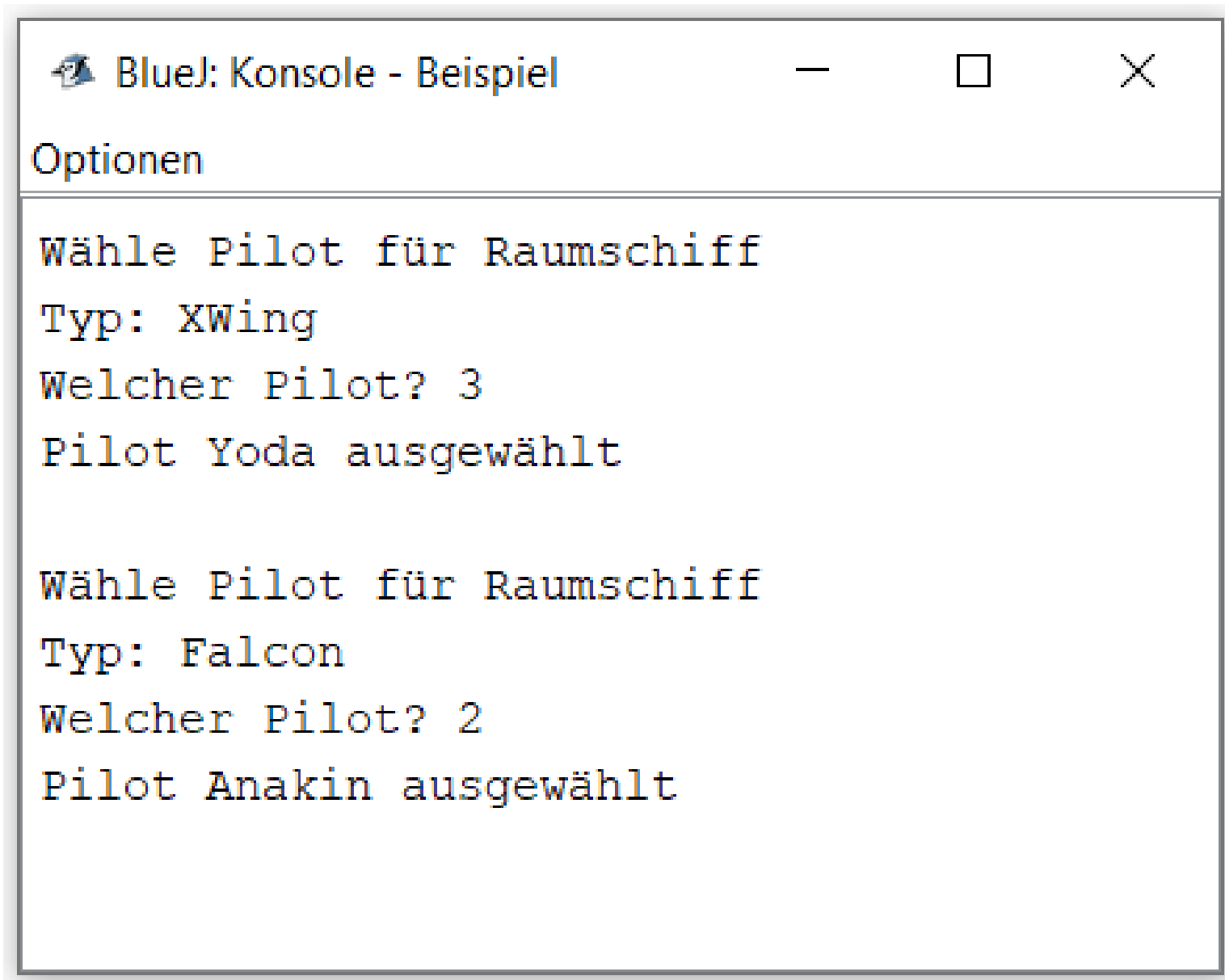


Referenzen und Methodenaufrufe

Wahl des Piloten durch Benutzer



Wahl des Piloten durch Benutzer



Wahl des Piloten durch Benutzer

Methode für Pilotenwahl:

- Typ des Raumschiffs ausgeben
- Benutzer den Pilot wählen lassen
- Pilot für das Raumschiff setzen

Für jedes Raumschiff eine eigene Methode?

Wahl des Piloten durch Benutzer

Methode für Pilotenwahl:

- Typ des Raumschiffs ausgeben
- Benutzer den Pilot wählen lassen
- Pilot für das Raumschiff setzen

Für jedes Raumschiff eine eigene Methode?

Nein, besser eine Methode, und das **Raumschiff als Parameter** übergeben

```
public class WeltraumSim
{
    private Raumschiff rs1, rs2;
    private Pilot p1, p2, p3;
    ...
```

```
public void main()
{
    wahlPilot(rs1);
    wahlPilot(rs2);
    ...
```

*je nachdem, ob
rs1 oder rs2
als Parameter
gesetzt wird,
arbeitet wahlPilot()
unterschiedlich*

```
public class WeltraumSim  
{
```

```
...
```

```
public void wahlPilot(Rauschiff rs)
```

```
{
```

```
    Console.println("Typ: " + rs.getTyp());
```

```
    Console.println("Welcher Pilot?");
```

```
    int w = Console.readInt();
```

```
    if (w == 1)
```

```
    {
```

```
        rs.setPilot(p1);
```

```
    }
```

```
...
```

```
public class WeltraumSim
{
```

```
...
```

```
public void wahlPilot(Rauschiff rs)
```

```
{
```

```
    Console.println("Typ: " + rs.getTyp());
```

```
    Console.println("Welcher Pilot?");
```

```
    int w = Console.readInt();
```

```
    if (w == 1)
```

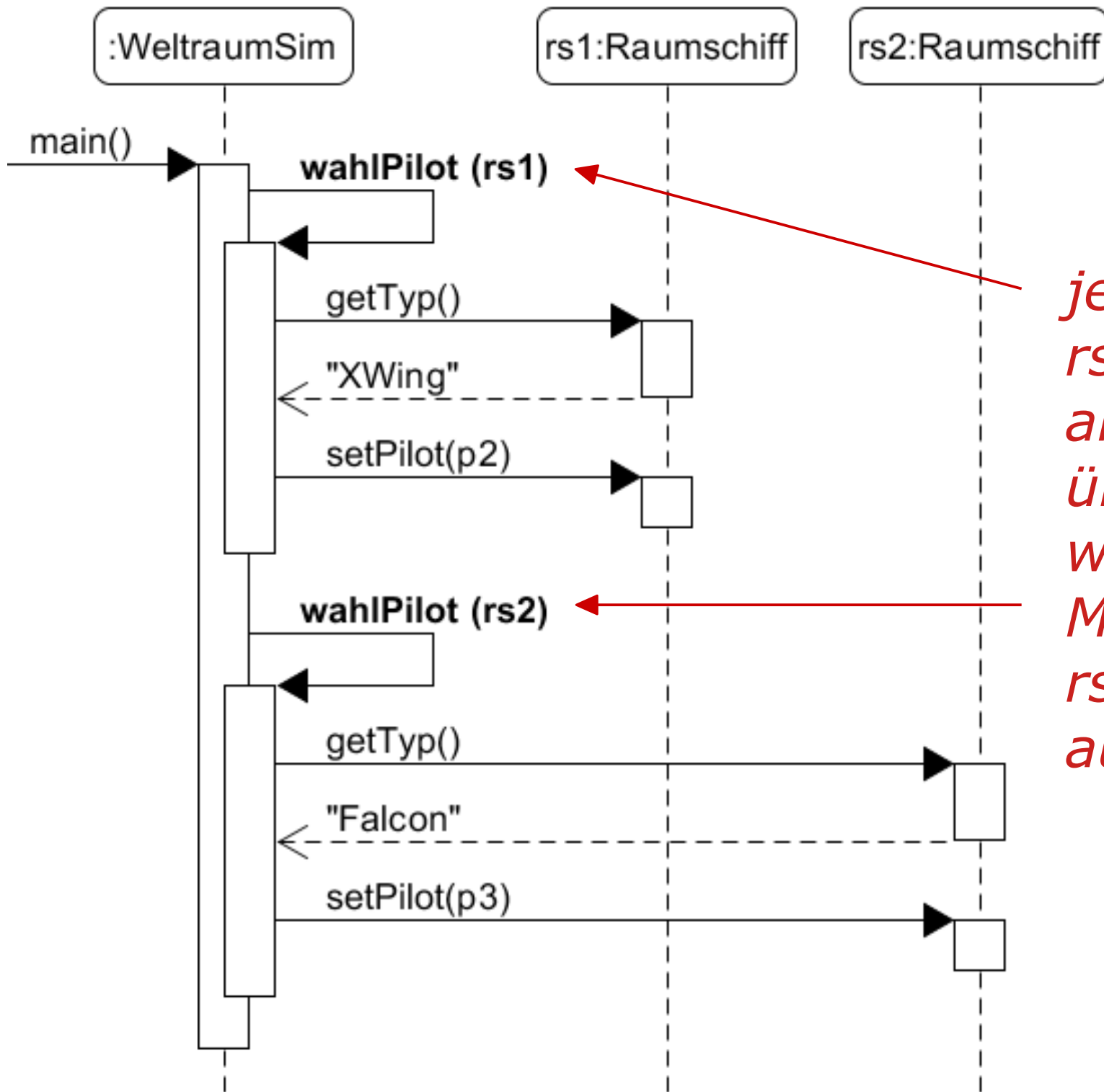
```
    {
```

```
        rs.setPilot(p1);
```

```
    }
```

```
...
```

*für Parameter **rs**
können Methoden
der Klasse **Raumschiff**
aufgerufen werden*



je nachdem ob rs1 oder rs2 als Parameter übergeben wird werden diese Methoden für rs1 oder rs2 aufgerufen

Vorsicht !

```
public void wahlPilot(Rauschiff rs)
```

```
{
```

```
    Console.println(rs.getTyp());
```

```
    ...
```

```
    if (w == 1)
```

```
    {
```

```
        rs.setPilot(p1);
```

```
    }
```

```
    ...
```

*Referenz „rs“
darf nicht
null sein!*

*Sonst stürzt das
Program ab.*

Vorsicht !

```
public void wahlPilot(Rauschiff rs)  
{  
    if (rs != null)  
    {  
        Console.println(rs.getTyp());  
        ...  
        if (w == 1)  
        {  
            rs.setPilot(p1);  
        }  
        ...  
    }  
}
```

*Daher: vor
Methodenaufruf
prüfen, ob die
Referenz nicht
null ist!*

Autor / Quellen

Autor:

- Christian Pothmann (cpothmann.de)
Freigegeben unter CC BY-NC-SA 4.0, Mai 2021

