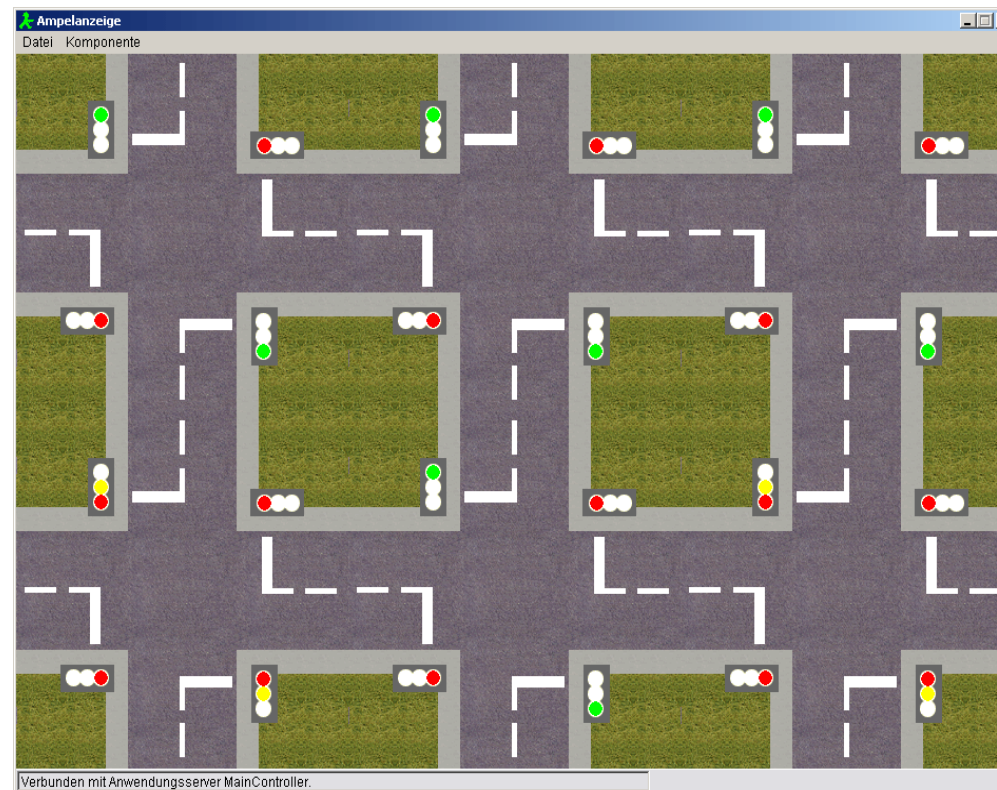


🕒 Ampelsteuerung in Java, C via CORBA

Middleware und Komponentenprogrammierung im Sommersemester 2004





🕒 Agenda Ampelsteuerung in Java, C via CORBA

1. Vorbetrachtungen
2. Entwurfsentscheidungen
3. Design Patterns
4. Anwendungsbeispiel
5. Komponentenbeispiel
6. Modultest
7. Ausblick

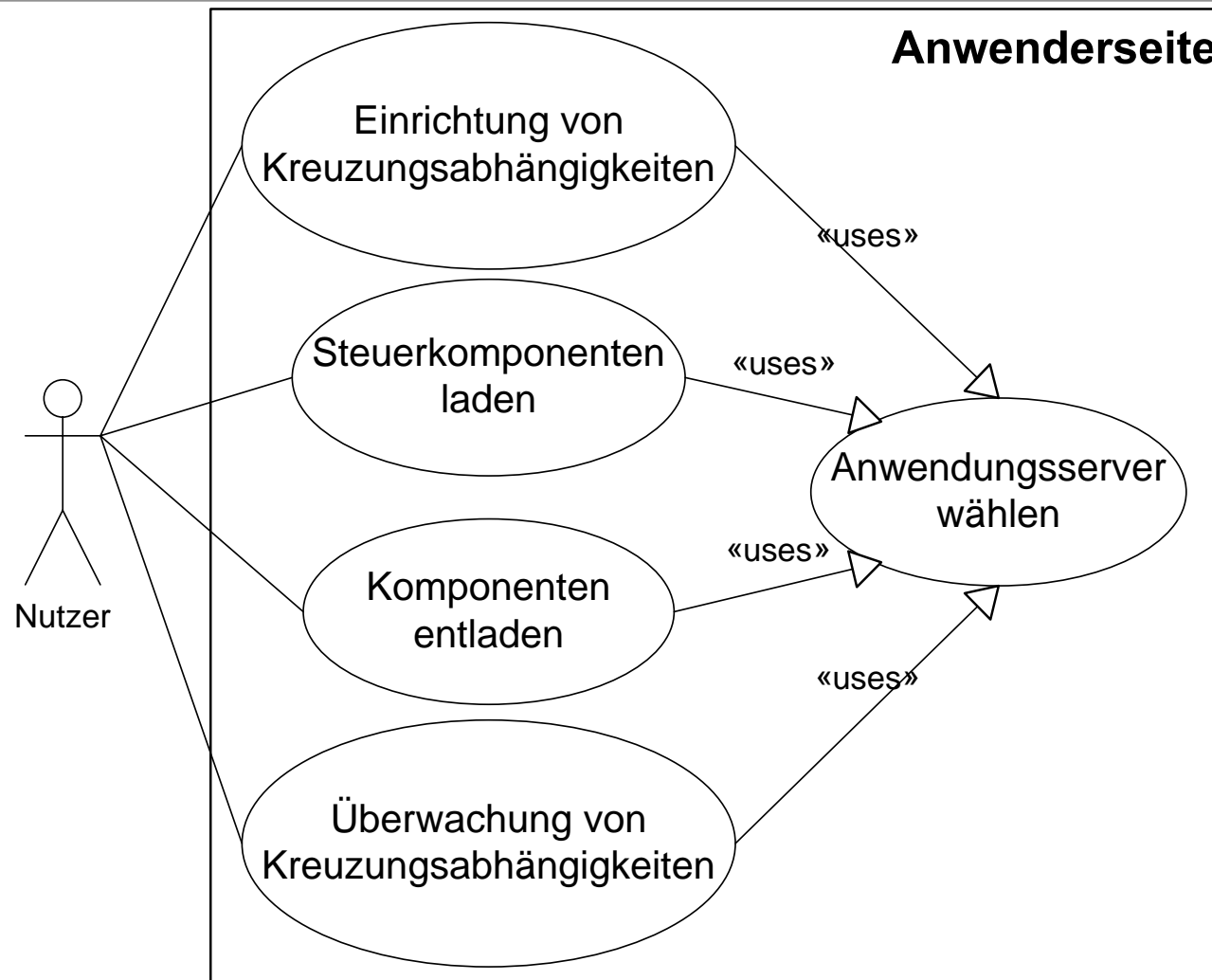
🕒 1. Vorbetrachtungen (Anforderungen)

- Mitarbeiter in der Verkehrsleitzentrale
- Kreuzungsüberwachung
- Steuern von Kreuzungen durch hinzuladbare Programmkomponenten
 - Grüne Welle
 - Notfallschaltung
- Lokale Verteilung aus der Realität nachbilden

🕒 1. Vorbetrachtungen

- Ampeln
 - stehen nur als Blackbox zur Verfügung
 - Nicht direkt steuerbar
 - Weiterhin autonom
- Realisierungskriterien
 - Einfache GUI
 - Einfache Kreuzungen

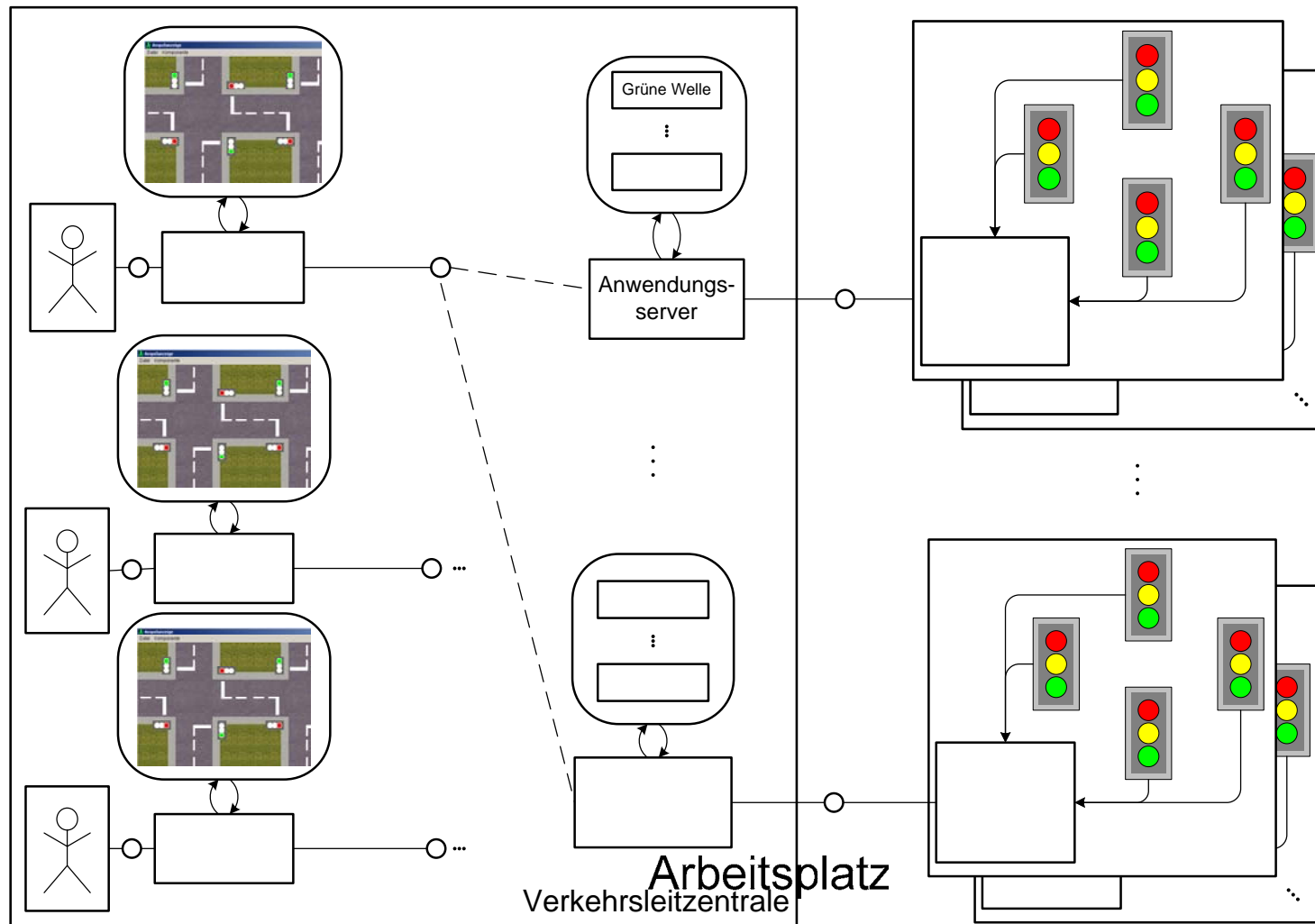
🕒 2. Entwurfsentscheidungen (Use-Case)



🕒 2. Entwurfsentscheidungen

- Klassisches Client-Server-Szenario
- Multi-User-Umgebung
- Graphische Repräsentation bleibt entkoppelt
- Nach Initialisierung autarke Funktionsweise

🕒 2. Entwurfsentscheidungen (Aufbauplan)





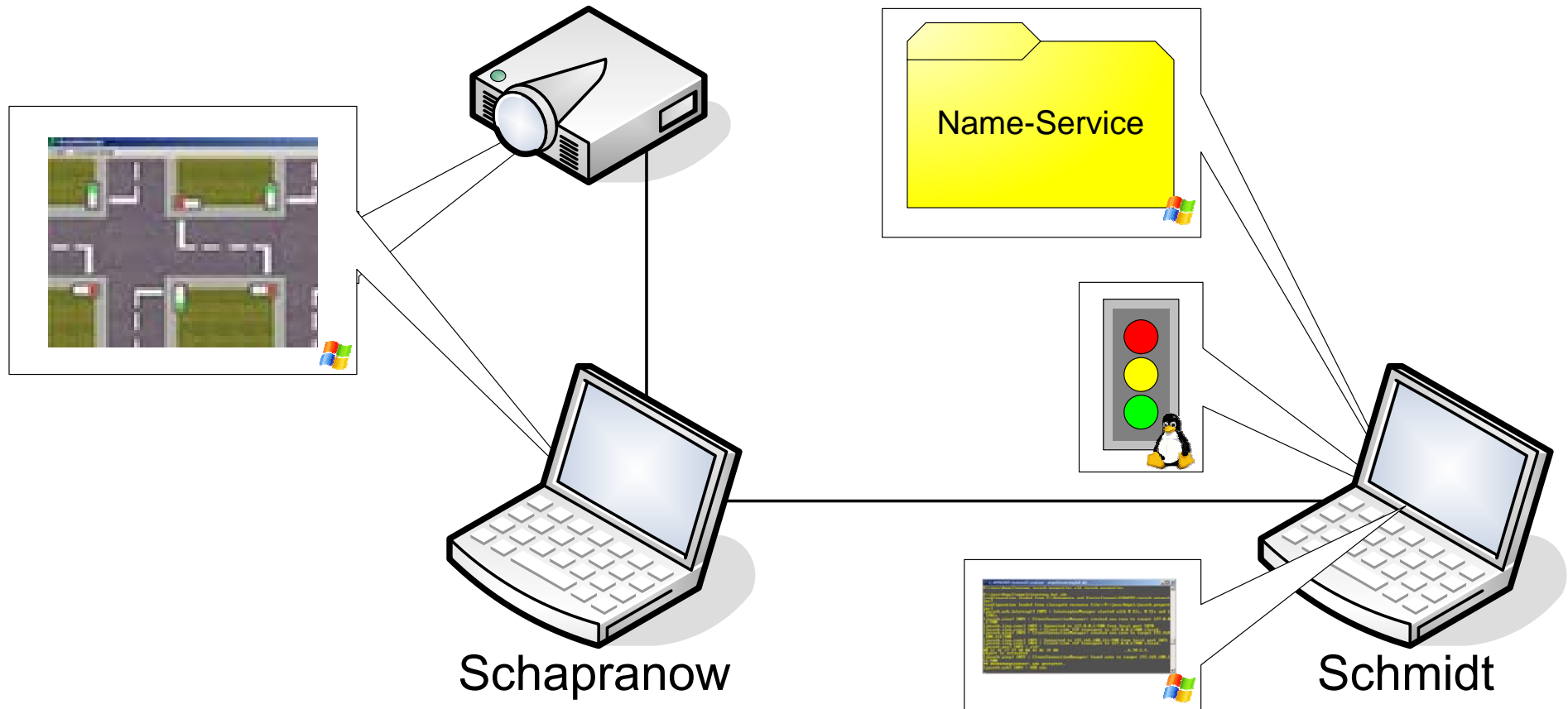
🕒 3. Design Patterns

- Bridge (Viewer → MCORBClient → ORBClient → ORBInit)
- Facade, Interface (GUI, Components)
- Singleton (ComponentManager)
- Proxy, Decorator (ORBClient)
- „Iterator“ (ComponentList, GUIList, ...)
- Observer (Ampelkreuzung → GUI)

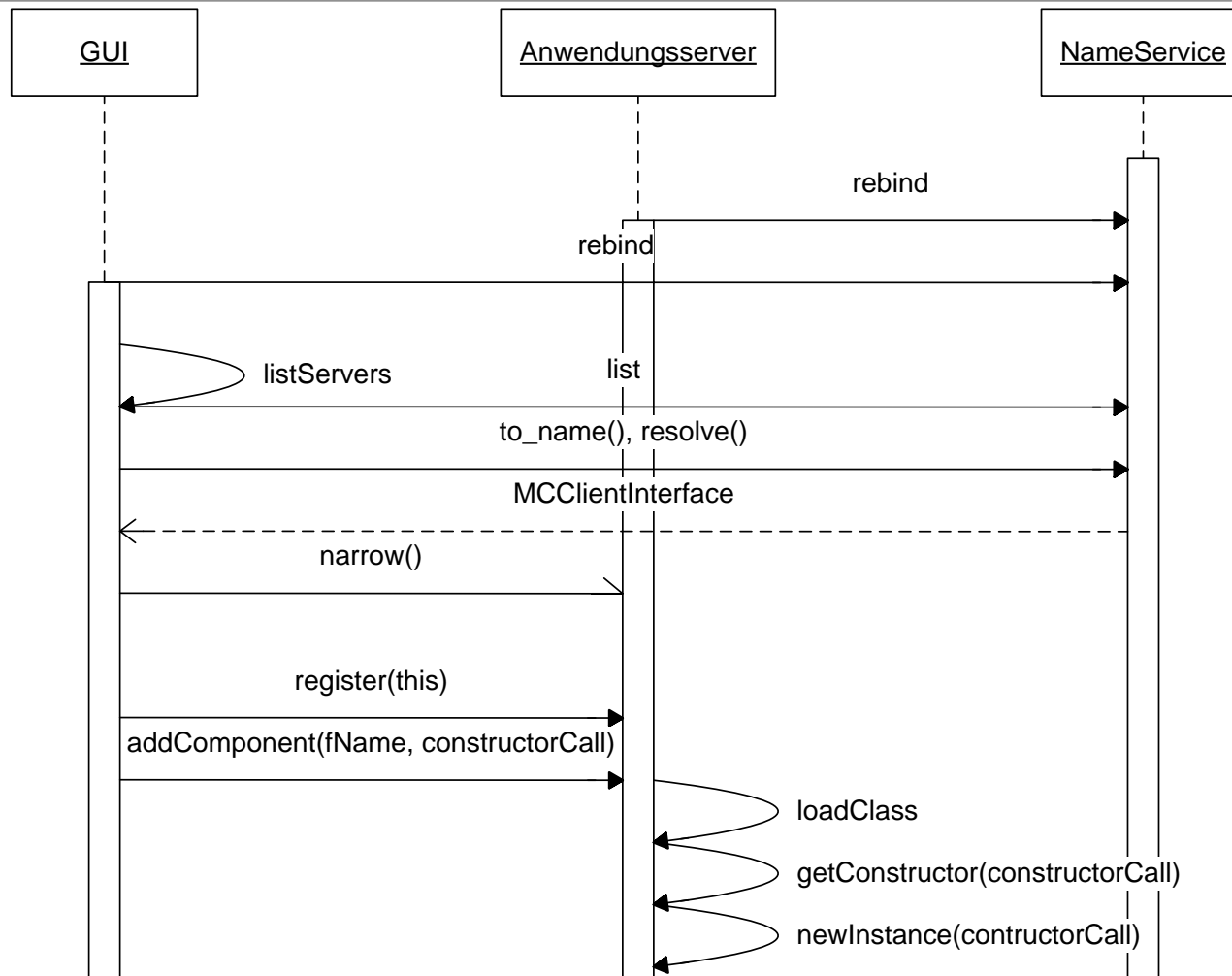
🕒 4. Anwendungsbeispiel

- Verbinden zu einem Anwendungsserver
- Konstruktion einer Kreuzungskombination
- Anmelden von Komponenten
- Abmelden der GUI
- Erneutes Verbinden einer GUI / Multi-User

🕒 4. Anwendungsbeispiel



🕒 4. Sequence-Diagramm



🕒 5. Komponentenbeispiel

- “Program to an interface, not an implementation”

```

public interface Components {
    public void startComponent();
    public void finalize();
    public void setComponentList(ComponentManager cManager);
    public String getComponentName();
    public String getCreationTime();
}

```



🕒 5. Komponentenbeispiel

```
package de.potsdam.uni.hpi.ss2004.corba;
```

...

```
orb = ORB.init(args, props);  
poa = POAHelper.narrow(orb.resolve_initial_references("RootPOA"));  
poa.the_POAManager().activate();  
obj = orb.resolve_initial_references("NameService");  
nc = NamingContextExtHelper.narrow(obj);  
obj = nc.resolve_str("Ampelsteuerung");  
path = new NameComponent[] {  
    new NameComponent("Ampelsteuerung", "")  
};  
obj = nc.bind_new_context(path);  
nc = NamingContextExtHelper.narrow(obj);
```

🕒 5. Vergleich von CORBA & DCOM anhand der IDL

```

..module main {
    interface MCClientInterface {
        void setStatusText(in string msg);
    }
}...

[ uuid(DA62D7F6-7517-4e3f-9CE7-E181159B1D4B), version(1.0) ]
..library main {

    [ uuid(4AAFD280-8B4E-4c1e-BA77-8738D45166CA), dual ]
    interface MCClientInterface : IDispatch {
        HRESULT setStatusText([in] BSTR msg);
    }

    [ uuid(4AAFD280-8B4E-4c1e-BA77-8738D45166CA), helpstring("GUI") ]
    coclass main {
        importlib("stdole32.tlb");
        [default] interface MCClientInterface;
    };
};...

```

🕒 6. Modultests

- JUnit
- Schnittstellenverhalten
- Sichere Überprüfung (Konstruktoren, Standardmethoden)
- Assertions

```

public void assertReq(Components c) {
    ...
    assertNotNull(c.getComponentName());
    System.out.println("\t" + c.getComponentName());
    assertNotNull(c.getCreationTime());
    System.out.println("\t" + c.getCreationTime());
    c.startComponent();
    ...
}

```

🕒 7. Ausblick

- Verbesserung der Umwelt
 - “Intelligente” Ampeln
 - Sensoren
- Weitere Kreuzungstypen
 - Reine Fußgängerampeln
 - T-Kreuzungen
- Mächtigerer Viewer
 - Variables Straßenlayout
- Sicherheit, Multi-User-Stabilität



Quellennachweise

- Java Programming With Corba - Advanced Techniques For Building Distributed Applications, 3Rd Edition (Wiley)
- <http://www.jacorb.org/releases/2.1/ProgrammingGuide.pdf.gz>, Mai 2004
- <http://gsraj.tripod.com/misc/compare.html>, Mai 2004
- <http://www.gnome.org/projects/ORBit2/orbit-docs/orbit/book1.html>, Juni 2004