
Cassey - Common Answer Set Evaluation sYstem

Jean Gressmann
Benjamin Kaufmann
Robert Lenk

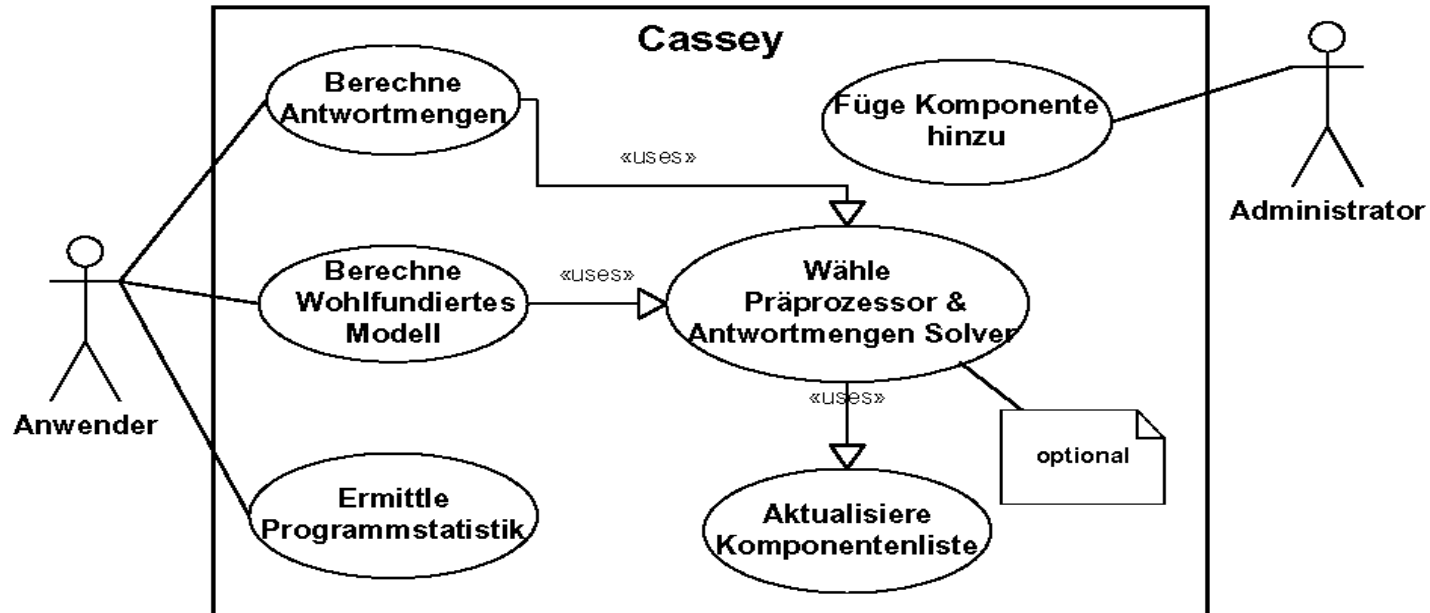
Roadmap des Vortrags

- Projektübersicht
- Gewählte Technologien
- Architektur & Implementation
- Test & Portabilität
- Vergleich mit anderen Technologien
- Ausblick

Cassey...

- Steht für Common Answer Set Evaluation sYstem
- Ist ein verteiltes System zur Berechnung von Antwortmengen
- Kann über dynamisch ladbare Komponenten erweitert werden

Entwurfsziele...



- Dynamisches Hinzufügen neuer Antwortmengen Solver
- Plattformunabhängiger Server
- Plattformunabhängiger Client mit GUI

...und deren Realisierung

- Server (C++)
 - Stellt Basisfunktionalität bereit
 - Delegiert Arbeit an Komponenten
- Komponenten (C++)
 - Basieren auf **CORBA Component Model**
- Client (Java)
 - SWT GUI Framework
- CORBA 2.x für die Client-Server Kommunikation

Was sind Antwortmengen?

- Kurz, es geht um Konsequenzen...

Programm:

vogel(tweety). vogel(tux).

penguin(tux).

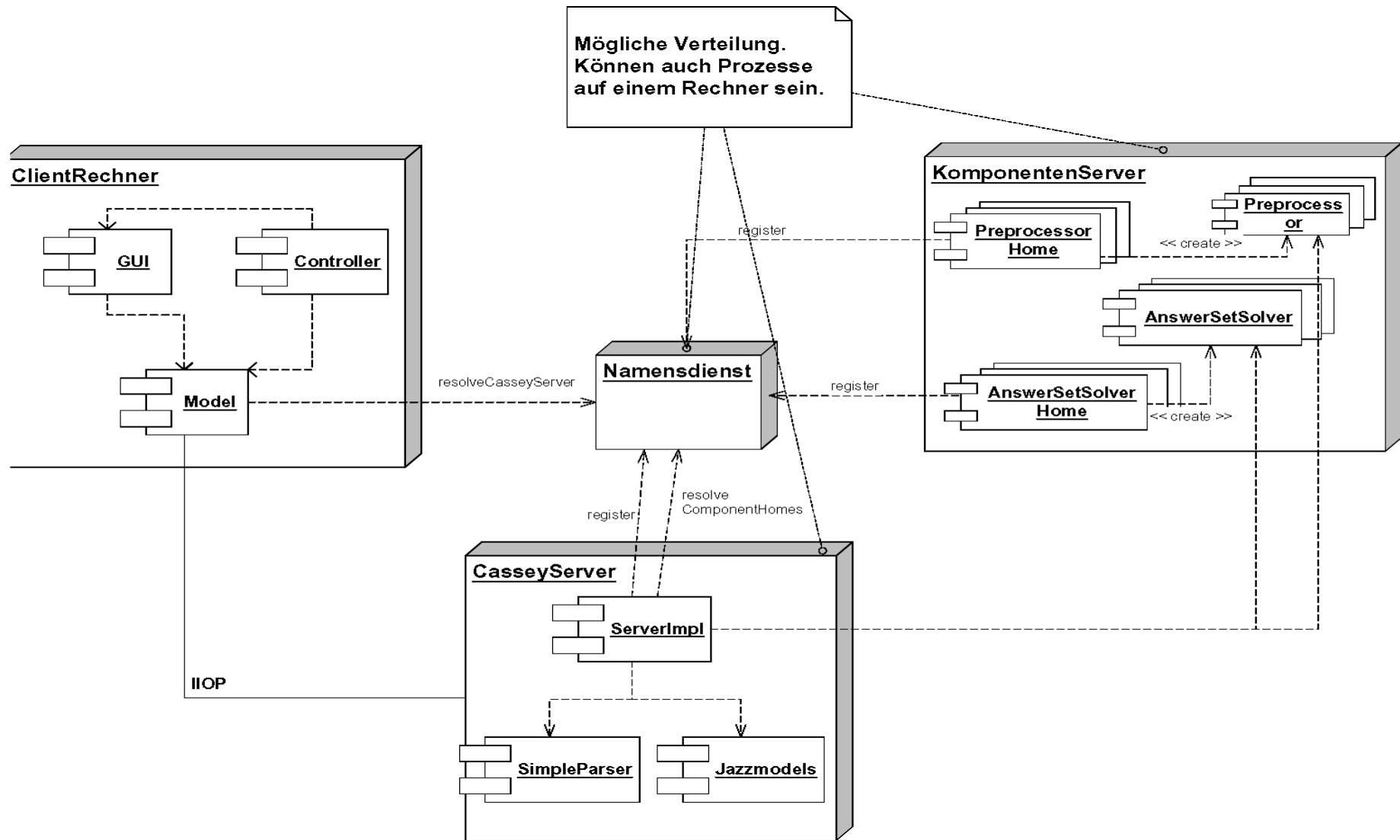
fliegt(X) :- vogel(X), not penguin(X).

Antwortmenge:

fliegt(tweety) penguin(tux) vogel(tux)

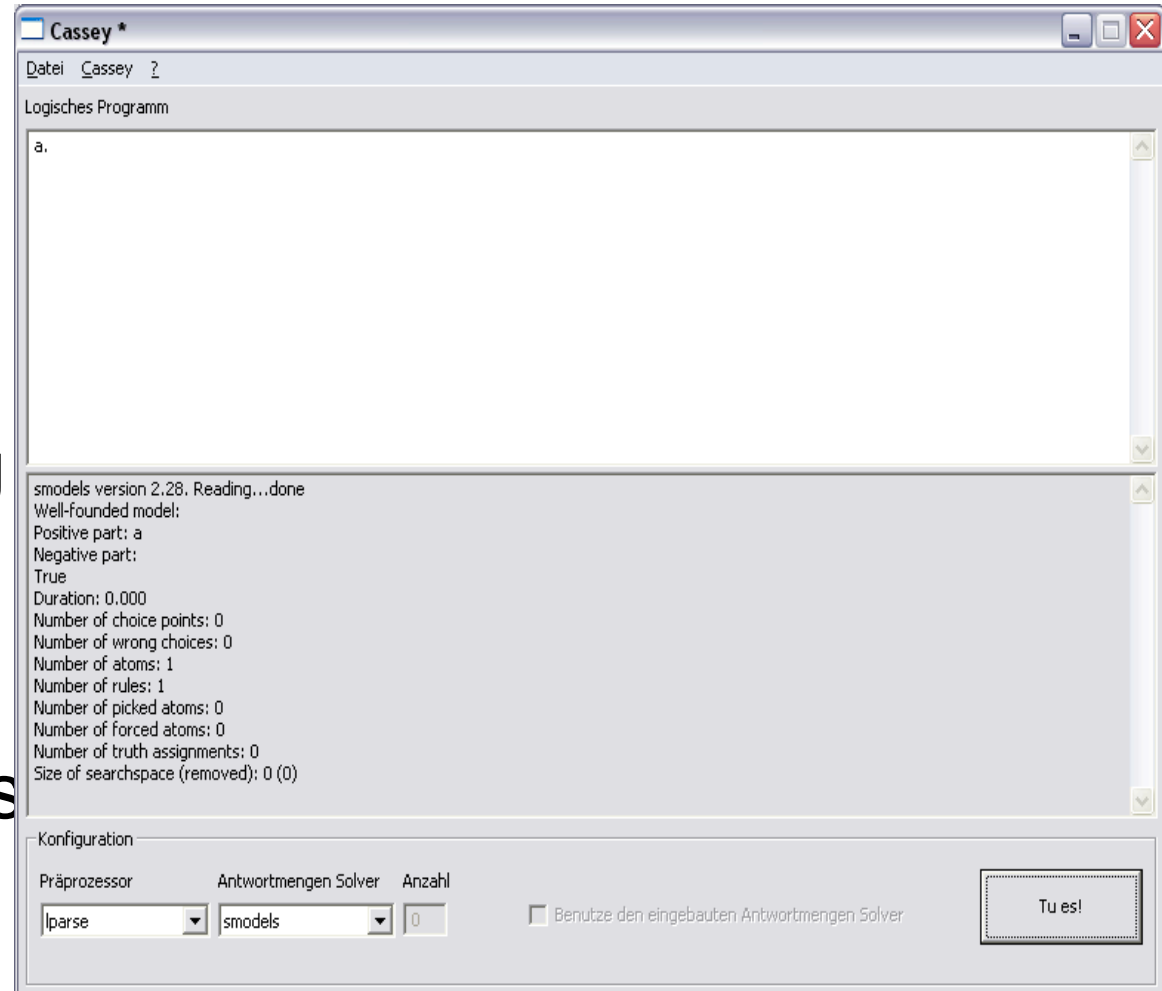
vogel(tweety)

Cassey – Architektur

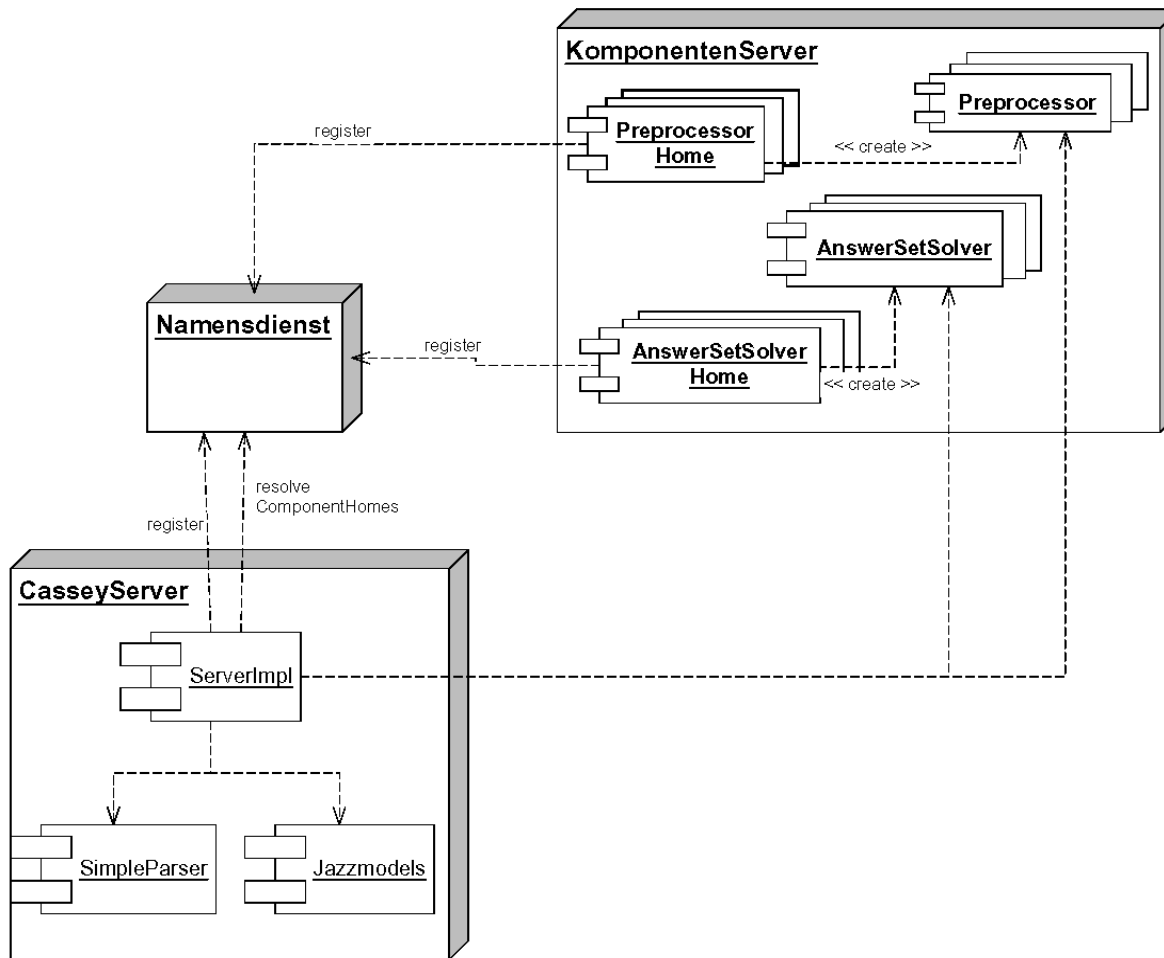


Casseys Client

- Delegiert Funktionalität an den Server
- Plattform-unabhängige GUI dank Java und SWT
- Komponenten als Strings



Casseys Server



- Komponenten werden dynamisch und auf Anfrage des Clients lokalisiert

Casseys Component Model

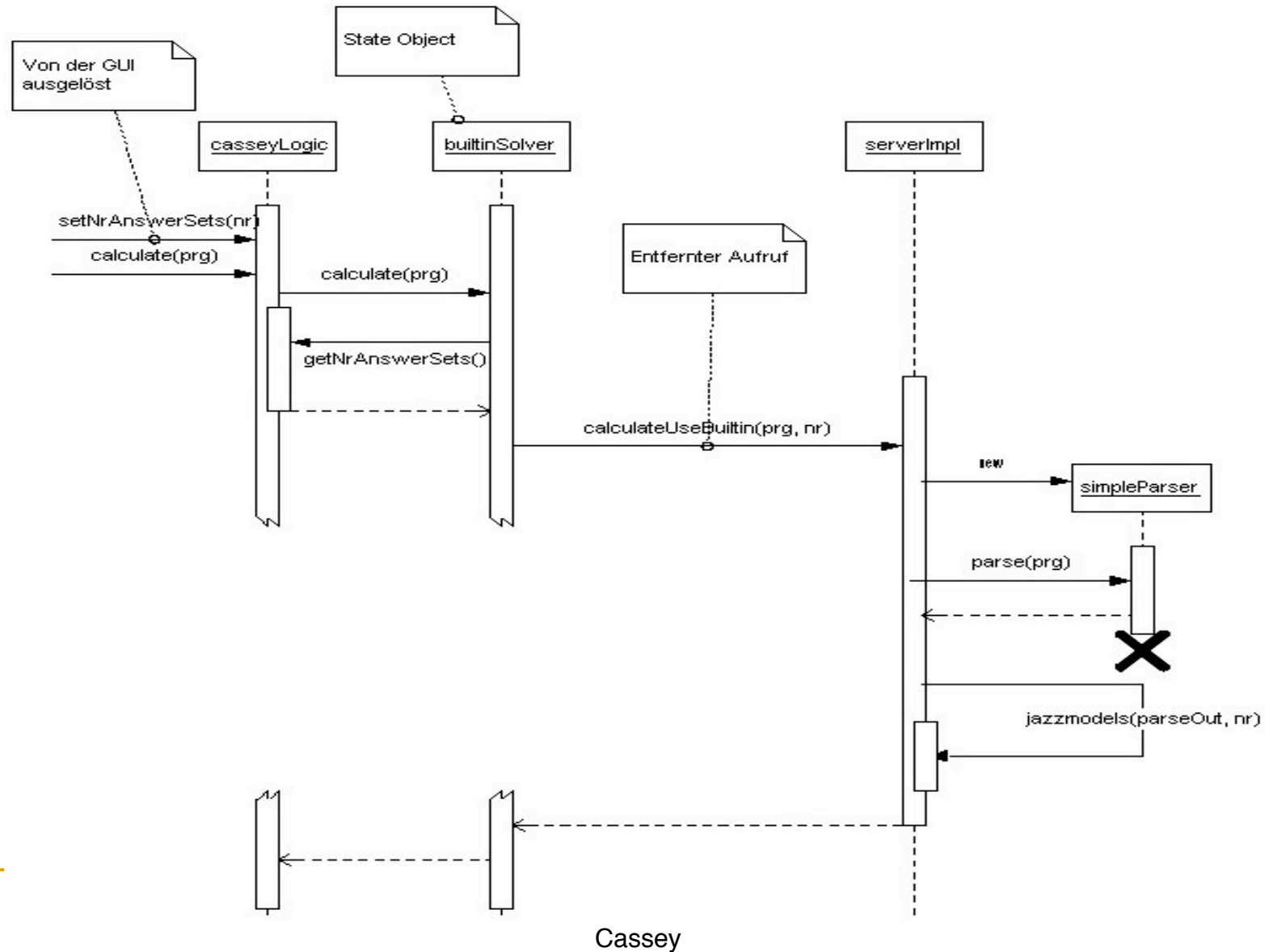
- Zwei Komponententypen
 - Präprozessoren
 - Antwortmengen Solver
- Mit Cassey kommen...
 - der lparse Präprozessor
 - der smodels Antwortmengen Solver

Casseys

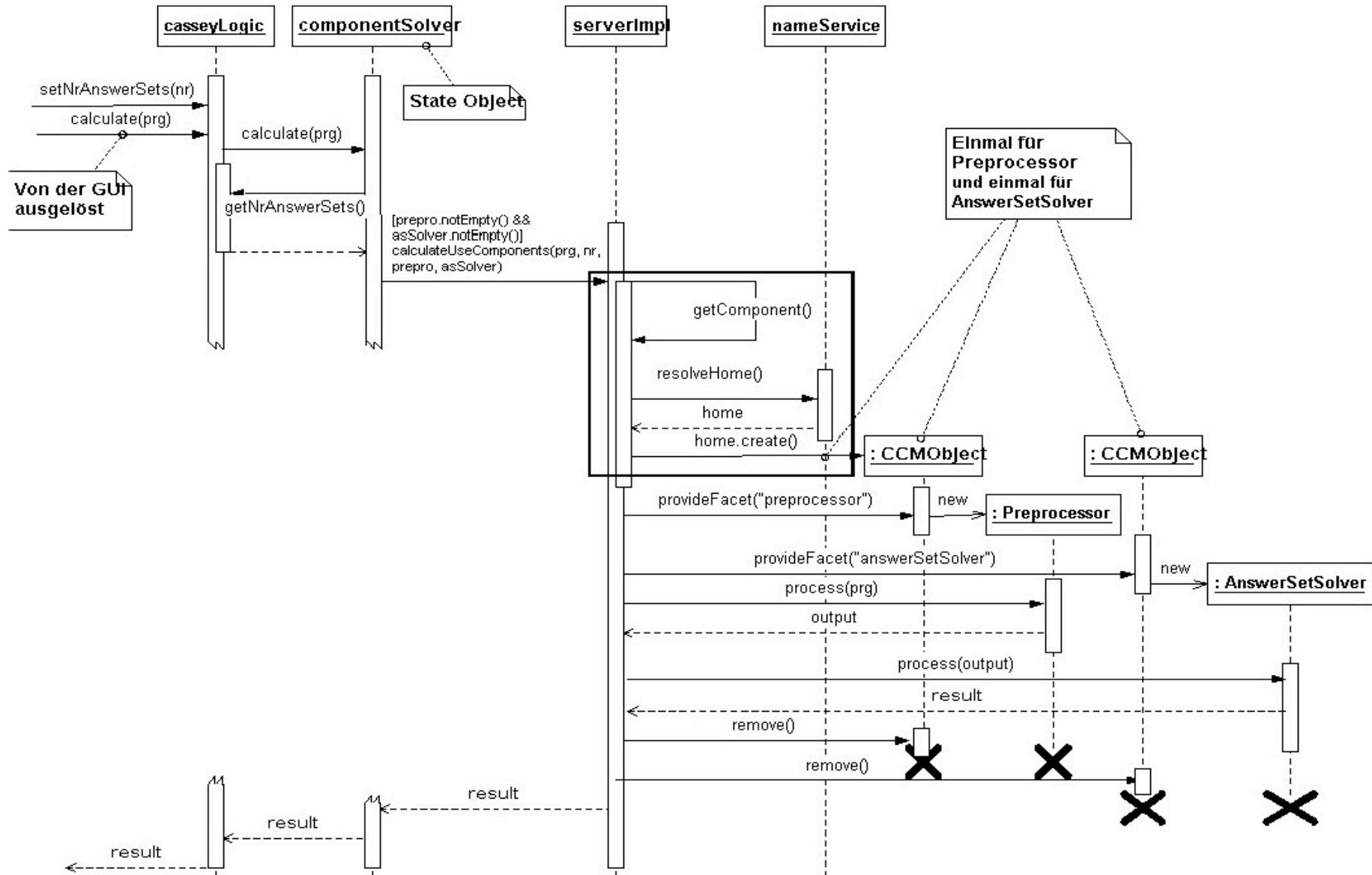
Komponentenschnittstellen

```
interface Preprocessor {  
    string process(string program);  
};  
  
interface AnswerSetSolver {  
    string calculateAnswerSets(string program,  
        unsigned nrOfAnswerSets);  
    string calculateWellfoundedModel(string  
        program);  
};
```

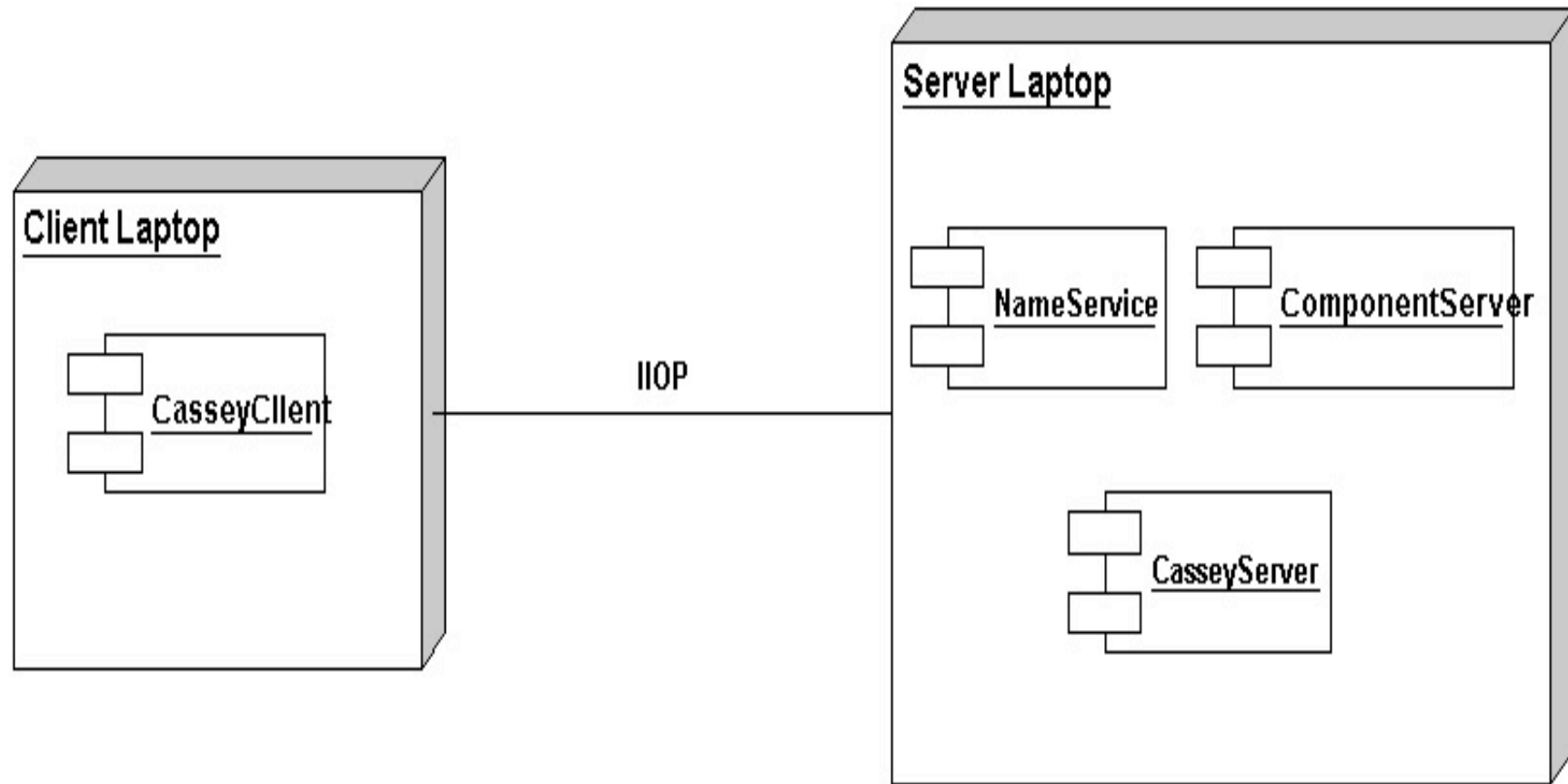
Ablauf: Eingebauter Solver



Ablauf: Solver als Komponente



Präsentation der Anwendung



Identifizierte Entwurfsmuster

- Client

- MVC => Observer
- State
- Proxy, Abstract Factory, Broker, ...

- Server

- Template Method
- Facade

... code a little, test a little ...

- Quellcodeverwaltung über CVS
- CPPUNIT (Server)
 - Servant Implementation
 - jazzmodels
 - simpleparser
- JUnit
 - Model

Cassey goes .NET ...

- Problem: Komponenten müssen der Runtime bekannt gemacht werden, aber...
- .NET stellt keinen Namensdienst zur Verfügung
- Für Cassey bedeutet dies die Serverarchitektur zu ändern

... durch Servermodifikation

- Server und Komponenten laufen auf der selben Maschine
- Komponenten werden auf Anfrage aus einem wohldefinierten Verzeichnis geladen
- Aufruf der Komponentenfunktionalität über Reflection (invoke)
- Server bietet dem Client die Komponentenfunktionalität wie gehabt an

Ausblick...

- Benchmarking und Jobverwaltung im Server
- Sicherheitskonzept
- Besseres Packaging und Deployment
- Entkopplung von Präprozessor und Antwortmengen Solver

Quellen

- P. Simons (1997). *Towards constraint satisfaction through logic programs and the stable model semantics.* Helsinki: Helsinki University of Technology.
- T. Schaub (2003). *Skript zur Vorlesung Antwortmengenprogrammierung.* Potsdam: Universität Potsdam.
- E. Gamma, R. Helm, R. Johnson & J. Vlissides (1997). *Design Patterns.* Boston: Addison-Wesley.
- M. Henning & S. Vinoski (1999). *Advanced CORBA programming with C++.* Boston: Addison-Wesley.
- Object Management Group (2002). *CORBA Components.*
- Frank Pilhofer (2002). *Writing and Using CORBA Components.*
- Martin Fowler (1999). *UML Distilled: A Brief Guide To The Standard Object Modeling Language.* Boston: Addison-Wesley.
- Microsoft MSDN Library – January 2004.