

Softwareprojekt:

„Agile Softwareentwicklung in einem
Open-Source-Projekt“

Sommersemester 2016


Franz Zieris

AG Software Engineering

Freie Universität Berlin

27.04.2016

Das Projekt: Worum geht es?

- Teilnahme an der Entwicklung eines wachsenden, lebendigen, Open-Source-Projekt
- Das Projekt?  Saros
 - Ein IDE-Plug-In, entwickelt u.a. hier am Fachbereich
- Ziele:
 - **Praxis** – Phasen der Softwareentwicklung durchlaufen, in einem komplexen Umfeld *Softwareprojekt*
 - **Selbstorganisation** – Teamarbeit koordinieren, Probleme erkennen und Reibungsverluste minimieren *Agil*
 - **Kommunikation** – einen Open-Source-Prozess kennenlernen *Open-Source*

Lernziele (aus der alten Studienordnung)

§ 5

Lehr- und Lernformen

4. Projekt: Projekte sind Lehrveranstaltungen, in denen ein größeres, meist anwendungsorientiertes Problem theoretisch und praktisch in einer Weise gelöst werden soll, die einer realen Situation soweit wie möglich entspricht. Das schließt die formale Problemspezifikation, die Zerlegung in Teilprobleme, die Festlegung von Schnittstellen sowie den Einsatz von Projektmanagementmethoden ein. Neben dem Erwerb von Fähigkeiten zur selbständigen Anwendung von Problemlösungsmethoden der Informatik auf eine konkrete Aufgabe dient ein Projekt auch der Vertiefung von kooperativen Arbeitstechniken. Gut dokumentierte, lauffähige Programme und ein zusammenfassender Projektbericht, aus dem die eigenen Leistungen hervorgehen, sind zum Abschluss des Projekts vorzulegen.

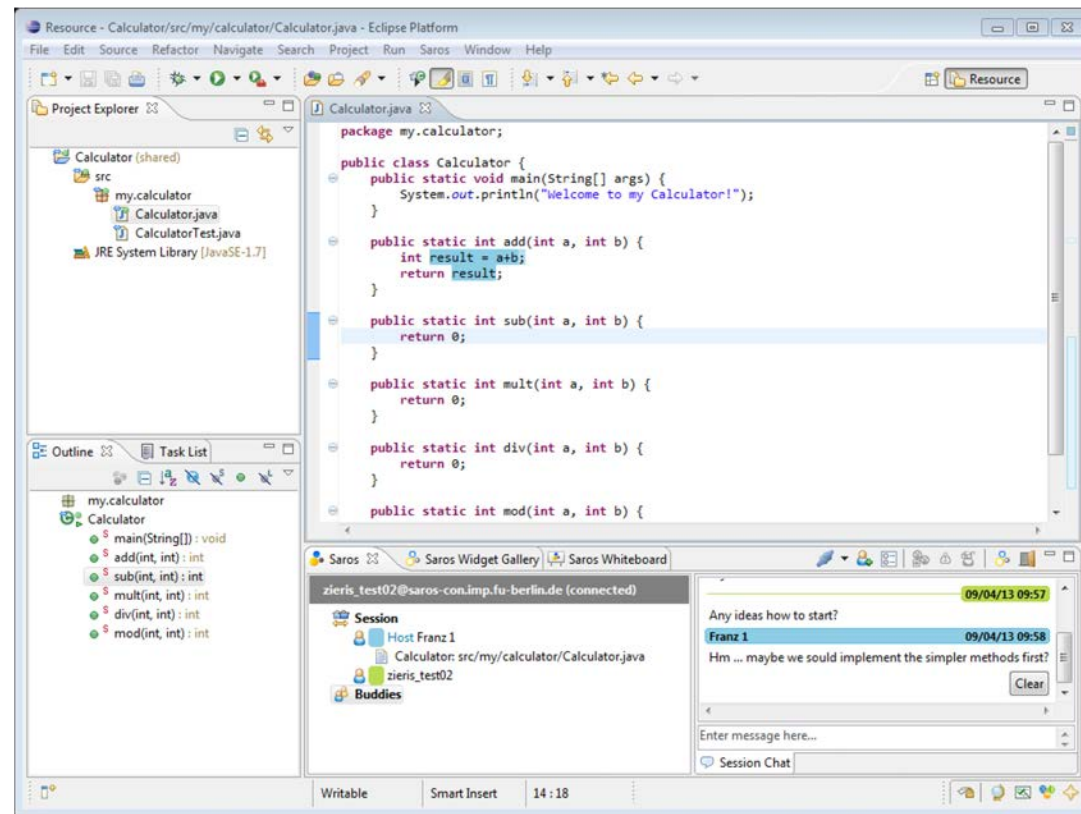
- wirklich echtes Projekt
- eigenverantwortliches Handeln
- Herausforderungen meistern

- Hilfe zur Selbsthilfe
- Prozess wichtiger als Produkt

- durch Praktiken und Werkzeuge

über Saros

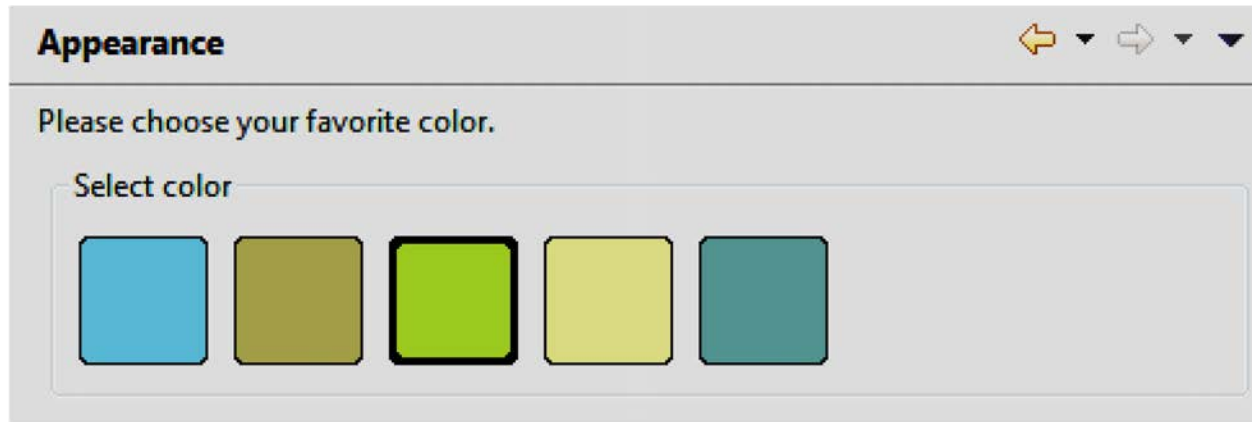
- Verteilte Softwareentwicklung in Echtzeit
- Plugin für Eclipse und IntelliJ IDEA
 - Java
 - HTML, JavaScript



Rückblick: frühere Runden

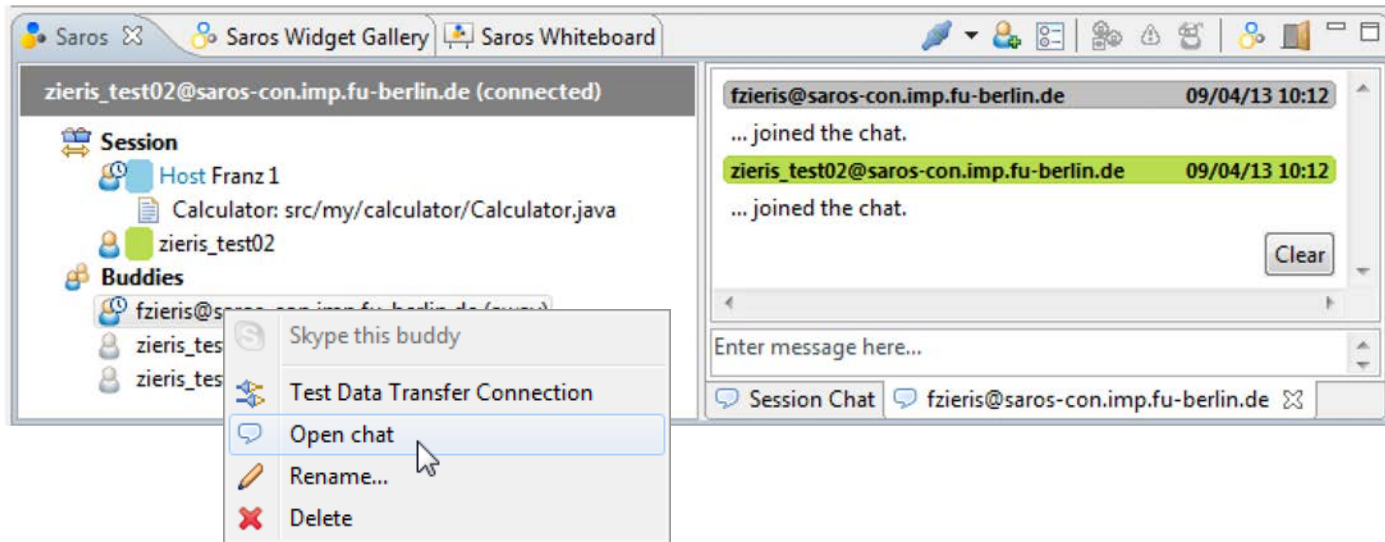
- Was wurde in vorherigen Durchführungen geleistet?
 - Eine Reihe von Bugfixes
 - Größere Features
 - neues Farbmanagement
 - 1-zu-1 Chat
 - Besserer Remote Cursor
 - Bessere Selection Annotations

Neues Farbmanagement



- nicht mehr RGB-Farben, sondern festes Schema
- Lieblingsfarbe wird zugewiesen, wann immer es möglich ist
- Farbzusordnungen sind unabhängig vom Sitzungsbeitritt
- **Technisch**: komplette Neuentwicklung, Ersatz

1-zu-1 Chat



- *session-unabhängiger* Chat
- **Technisch**: Reintegration und umfangreiches Refactoring eines experimentellen Features

Besserer Remote Cursor

```
23         return new FileInputStream(filename);
24     }
25     /**
26     *
27     * @param response
28     */
29     public void prepareResponse(Response resp
30
31 }
```

- Labels für den Cursor der anderen Sitzungsteilnehmer
- **Technisch**: komplette Neuentwicklung

Bessere Selection Annotation

```
/**
 * New awesome class
 */
public class Baff {

    /**
     * Says "Hello"
     *
     * @return a String representing a friendly greeting
     */
    public static String sayHello() {
        return "Hello";
    }
}
```

Local

```
/**
 * New awesome class
 */
public class Baff {

    /**
     * Says "Hello"
     *
     * @return a String representing a friendly greeting
     */
    public static String sayHello() {
        return "Hello";
    }
}
```

Local

```
/**
 * New awesome class
 */
public class Baff {

    /**
     * Says "Hello"
     *
     * @return a String representing a friendly greeting
     */
    public static String sayHello() {
        return "Hello";
    }
}
```

Remote

```
/**
 * New awesome class
 */
public class Baff {

    /**
     * Says "Hello"
     *
     * @return a String representing a friendly greeting
     */
    public static String sayHello() {
        return "Hello";
    }
}
```

Remote

- “Blockige” Hervorhebung für Selektionen
- **Technisch**: komplette Neuentwicklung



Technisches

Über Saros, Historie & Status

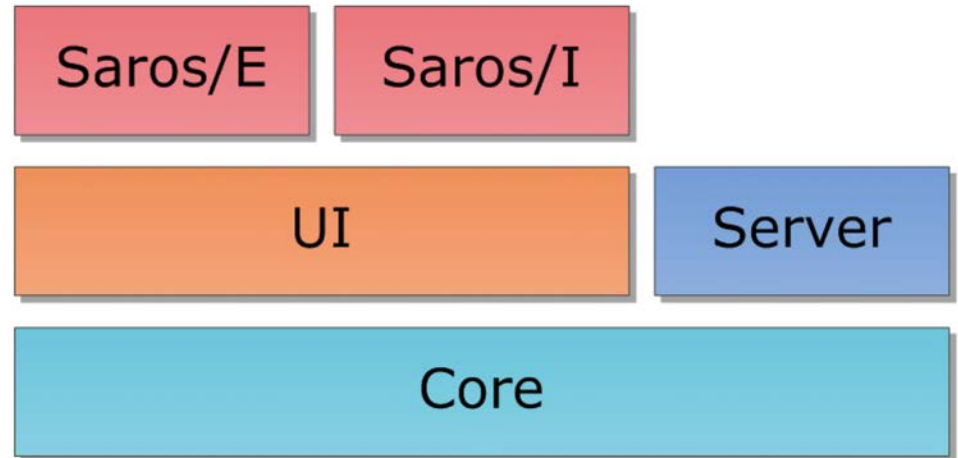
Saros-Historie

- **2006:**
 - Start der Entwicklung als Eclipse-Plugin
 - Zunächst nur zwei Nutzer mit exklusiven Schreibrechten
- **2014:**
 - Start der Entwicklung als IntelliJ-Plugin
 - Start der Entwicklung als Stand-Alone-Server
- **2015:**
 - Start der Entwicklung einer IDE-unabhängigen Nutzeroberfläche
 - Teilnahme am Google Summer Code
- **Stand heute:**
 - Bisher 75+ Entwickler/innen
 - 1500-2000 Downloads/Monat
 - In den populärsten 0,4% aller GitHub-Projekte (Stars & Forks)

Status von Saros (Produkt)

- Drei Varianten von Saros:

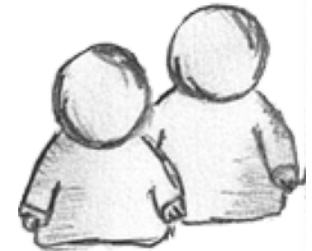
- Saros/Eclipse
- Saros/IntelliJ
- Saros/Server



- nutzen alle den gemeinsamen Saros/Core
- IDE-Varianten nutzen gemeinsame GUI
 - Oberfläche in HTML und JavaScript

Status von Saros (Prozess)

- Open-Source-Projekt
 - öffentliche Mailinglisten
(Geschäftssprache: Englisch)
 - öffentlicher Code-Review-Prozess
- Continuous Integration
- Automatische Tests
 - Unit-Tests
 - System-Tests
- Statische Code-Analyse
 - inkl. Architecture Compliance Checking



Organisatorisches (I)

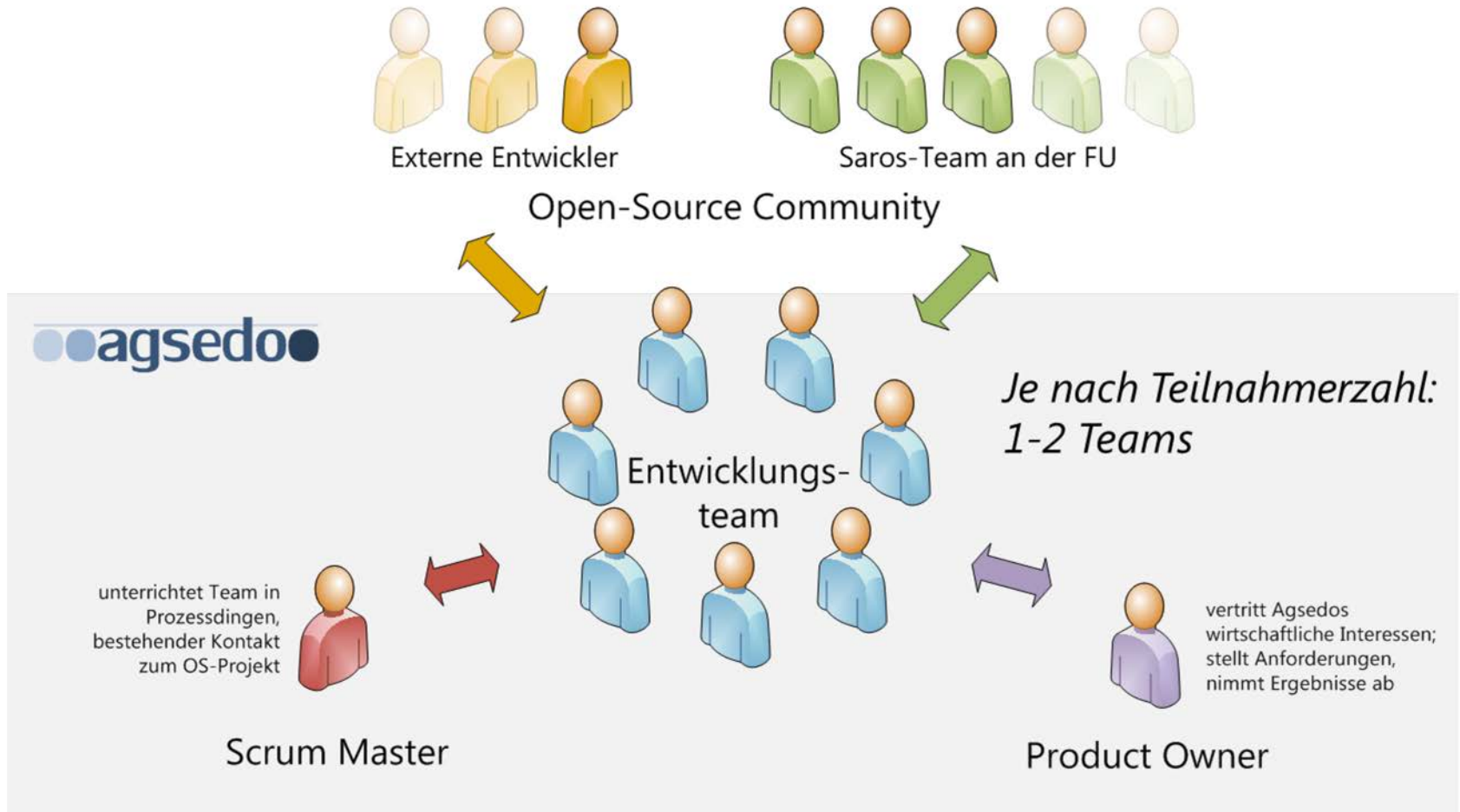
Kontext, Teamstruktur & Zeitplan

Kontext



- Fiktives Mittelstandsunternehmen agsedo
 - entwickelt Individualsoftware für eine Reihe namhafter Firmen
 - Zwei Standorte in Deutschland
 - Nutzen seit einem Jahr Saros – wollen nun etwas zurückgeben und haben auch Featurewünsche
 - Budget für ein bis zwei Teams (**ihr**) für die Dauer von sechs Wochen ist bewilligt

Personelle Strukturierung des Projekts



Zeitliche Strukturierung des Projekts

6 Wochen: Einführung + 5× einwöchige Iterationen

KW 33	KW 34	KW 35	KW 36	KW 37	KW 38
Einführungs- woche	Sprint 1	Sprint 2	Sprint 3	Sprint 4	Sprint 5

Iteration: 5 Tage

Montag	Dienstag	Mittwoch	Donnerstag	Freitag
Planung der Iteration	Teamarbeit	Teamarbeit	Teamarbeit	Teamarbeit
Teamarbeit				Auswertung

Organisatorisches (II)

Bewertungsmaßstab & Modulkriterien



Modulkriterien



- wochenweise **Gruppen- und Einzelnoten**
- Jede/r Einzelne/e ist nicht nur für sich selbst, sondern auch für das Gesamtprojekt verantwortlich

Modulkriterien – Gruppennote



- Im Kern:
Produktivität und **Qualität der Zusammenarbeit**
- Im Einzelnen:
 1. Größe der vorgenommenen Arbeit in einer Iteration
 - nicht zu wenig
 - nicht zu viel
 2. Grad des Erfolgs
 - nicht nur funktionaler Umfang, sondern auch Qualitätssicherung und Dokumentation
 3. Prozess/Prozesssteuerung
 - Durchführungsqualität der Praktiken
 - Umgang mit Problemen, Innovation
 - Art der Zusammenarbeit

Modulkriterien – Einzelnote



- *„Wie kann ich dafür sorgen, dass meine individuelle Leistung auch gewürdigt wird?“*
 - jeden Freitag bis 12:30 Uhr per Mail max. ½ DIN A4-Seite
 - Was waren meine drei Hauptleistungen diese Woche?
 - Was waren die größten Probleme, die ich diese Woche hatte?
 - Auf welche Weise habe ich versucht diese Probleme zu lösen? Erfolgreich?
 - Wer von meinen Kolleg/inn/en hat mich am besten unterstützt? In welcher Weise?

Wie geht es weiter?

- **Ab sofort:**
 - Lesen der Literatur (siehe Webseite)
 - <http://git-scm.com/book> (Kapitel 1-3, evtl. 5)
 - <http://www.scrumguides.org> (komplett)
- **In der nächsten Woche:**
 - (ca.) 15-minütige Einzelgespräche
 - Überprüfung des Literaturverständnisses
- **Fragen:**
 - Im KVV-Forum stellen

Mailinglisten des Saros-Projekts

- Auf Sourceforge
 - **dpp-devel@** die Entwicklerliste
 - **dpp-announce@** Release-Ankündigungen
 - **dpp-robot@** automatische Nachrichten
(Bugtracker, Commits, CI-Server)
 - <https://sourceforge.net/p/dpp/mailman/>

**Danke für die
Aufmerksamkeit.**

Bildquellen



<http://www.iconarchive.com/show/oxygen-icons-by-oxygen-icons.org/Status-mail-task-icon.html>

http://www.iconfinder.com/icondetails/40736/128/certificate_icon



http://www.iconfinder.com/icondetails/8810/128/calendar_date_time_icon

<http://www.iconarchive.com/show/crystal-clear-icons-by-everaldo/Mimetype-schedule-icon.html>

http://www.iconfinder.com/icondetails/34297/128/friends_group_guy_msn_people_users_icon