

Entwicklung eines zentralen Steuerungsprogramms für einen Forschungsflugsimulator



TECHNISCHE
UNIVERSITÄT
DARMSTADT

Vortragender: Frederik Bark, Heiko Carrasco

Team: Frederik Bark, Heiko Carrasco, Jonas Meurer und Leonardo Zaninelli



Der Simulator

Aktuelle Situation



- ▶ Flugsimulator am Fachbereich Flugsysteme und Regelungstechnik

Der Simulator

Aktuelle Situation



- ▶ Flugsimulator am Fachbereich Flugsysteme und Regelungstechnik
- ▶ Testen von Thesisthemen

Der Simulator

Aktuelle Situation



TECHNISCHE
UNIVERSITÄT
DARMSTADT

- ▶ Flugsimulator am Fachbereich Flugsysteme und Regelungstechnik
- ▶ Testen von Thesisthemen
- ▶ Komplexes System bestehend aus mehreren Computern

Der Simulator

Aufbau des Systems



TECHNISCHE
UNIVERSITÄT
DARMSTADT

- ▶ 9 Computer mit drei verschiedenen Betriebssystemen

Der Simulator

Aufbau des Systems



TECHNISCHE
UNIVERSITÄT
DARMSTADT

- ▶ 9 Computer mit drei verschiedenen Betriebssystemen
- ▶ Diverse selbstgeschriebene Programme

Der Simulator

Aufbau des Systems

- ▶ 9 Computer mit drei verschiedenen Betriebssystemen
- ▶ Diverse selbstgeschriebene Programme
 - ▶ Abhängigkeiten über Computer hinweg

Der Simulator

Aufbau des Systems

- ▶ 9 Computer mit drei verschiedenen Betriebssystemen
- ▶ Diverse selbstgeschriebene Programme
 - ▶ Abhängigkeiten über Computer hinweg
 - ▶ Verschiedene Programmiersprachen

Der Simulator

Aufbau des Systems



TECHNISCHE
UNIVERSITÄT
DARMSTADT





Der Simulator

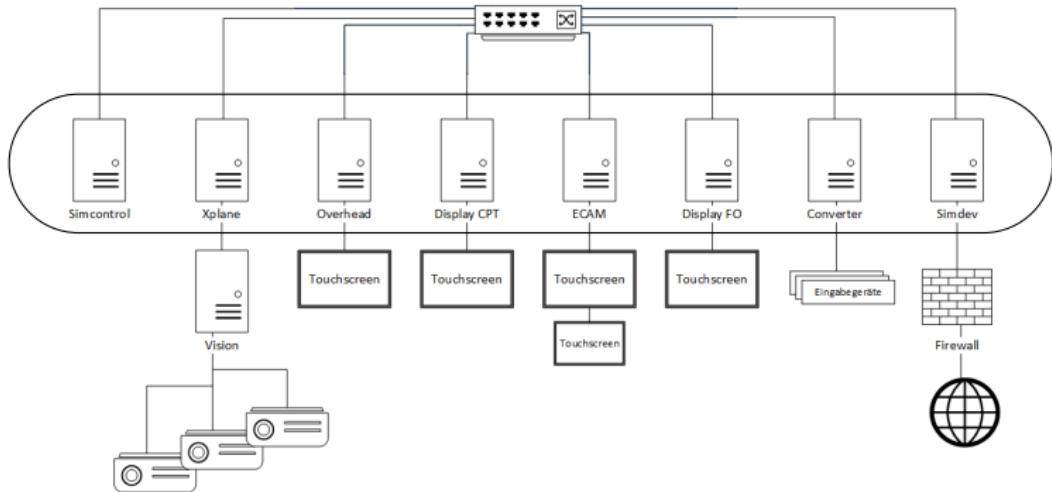
Aufbau des Systems



TECHNISCHE
UNIVERSITÄT
DARMSTADT

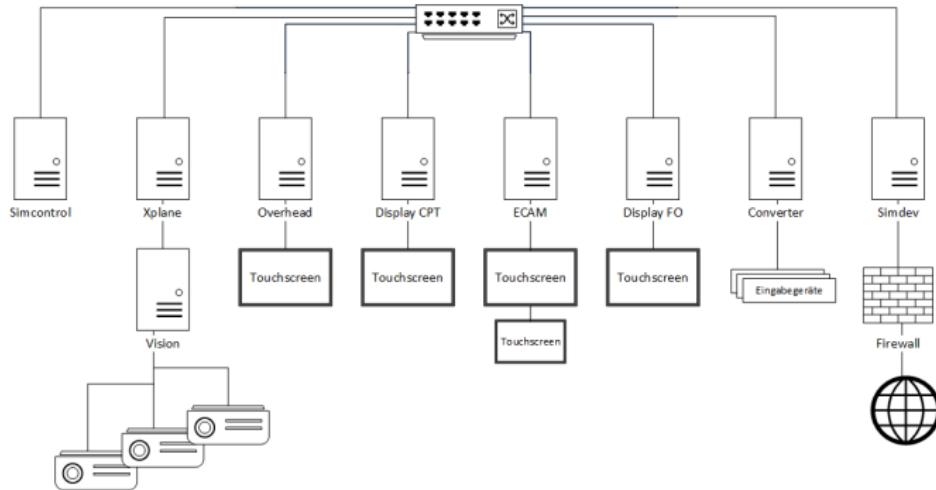
Der Simulator

Aufbau des Systems



Der Simulator

Aufbau des Systems



Features:

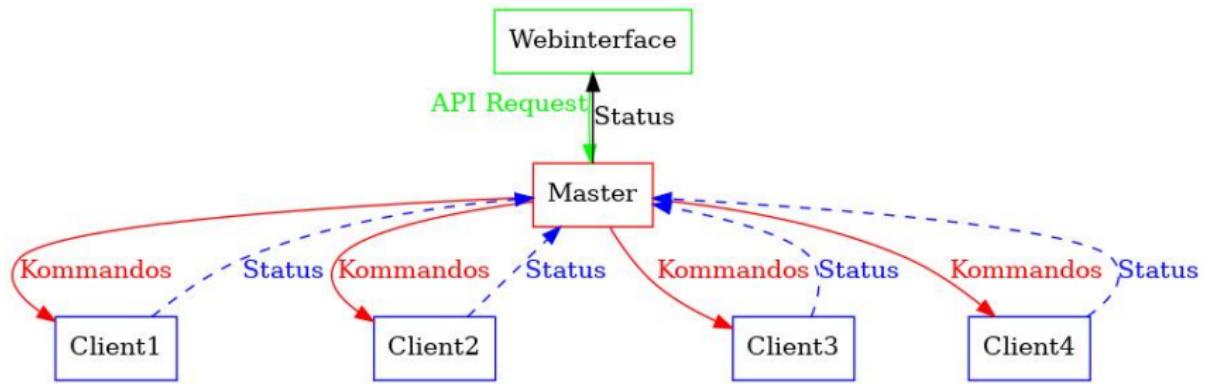
- ▶ starten/steuern eines einzelnen Clients über eine Benutzeroberfläche
- ▶ erstellen und ausführen von Scripts zur Automatisierung des Startvorgangs

Sprache:



Software

Aufbau



Software Fortschritt



TECHNISCHE
UNIVERSITÄT
DARMSTADT

Flugsimulator

Clients:

client1

IP: 0.0.0.0

MAC: 00:00:00:00:00:00

Programs:

Hello World

Program +

client2

IP: 0.0.0.1

MAC: 00:00:00:00:00:02

Programs:

Info! No programs for this client in system.

Program +

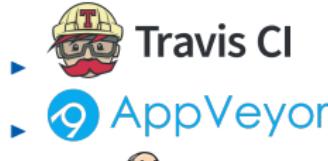
Client +

Die Software für die Clients muss auf vielen verschiedenen Systemen laufen:

1. Hohe Testabdeckung (mindestens 90%):



2. Viele Testumgebungen:



- ▶ **Travis CI** (Linux)
- ▶ **AppVeyor** (modernes Windows)
- ▶ **Jenkins** (Windows XP)

Qualitätssicherung

Portabilität



TECHNISCHE
UNIVERSITÄT
DARMSTADT

Zusammengefasst in Pullrequests von Github für jede Userstory:

Add more commits by pushing to the `20_programme_hinzufuegen` branch on `bp-flugsimulator/server`.

 All checks have passed Hide all checks

7 successful checks

		continuous-integration/appveyor/branch — AppVeyor build succeeded	Details
		continuous-integration/appveyor/pr — AppVeyor build succeeded	Details
		continuous-integration/jenkins/branch — This commit looks good	Details
		continuous-integration/travis-ci/pr — The Travis CI build passed	Details
		continuous-integration/travis-ci/push — The Travis CI build passed	Details

 This branch has conflicts that must be resolved [Resolve conflicts](#)

Use the [web editor](#) or the [command line](#) to resolve conflicts.

Qualitätssicherung

Portabilität



TECHNISCHE
UNIVERSITÄT
DARMSTADT

Vor dem mergen einer Userstory wird eine Liste abgearbeitet:

TheMangalex commented 4 days ago • edited

Owner +

Informationen

Reviewer	Datum	Zeit	Userstory
Frederik	29.11.17	0.5h	19

Liste

- Test Coverage (auf unseren Dateien) mindestens 90%
- Jede Klasse und Funktion ist grob Dokumentiert
- Schwierige Codestellen sind dokumentiert
- Alle Test laufen fehlerfrei durch
- Der Code ist korrekt formatiert
- Code erfüllt das Akzeptanzkriterium der Userstory (und dabei spezifisch nur das Akzeptanzkriterium dieser Userstory)
- Die Userstory ist ausgefüllt (Datum, Zeit)

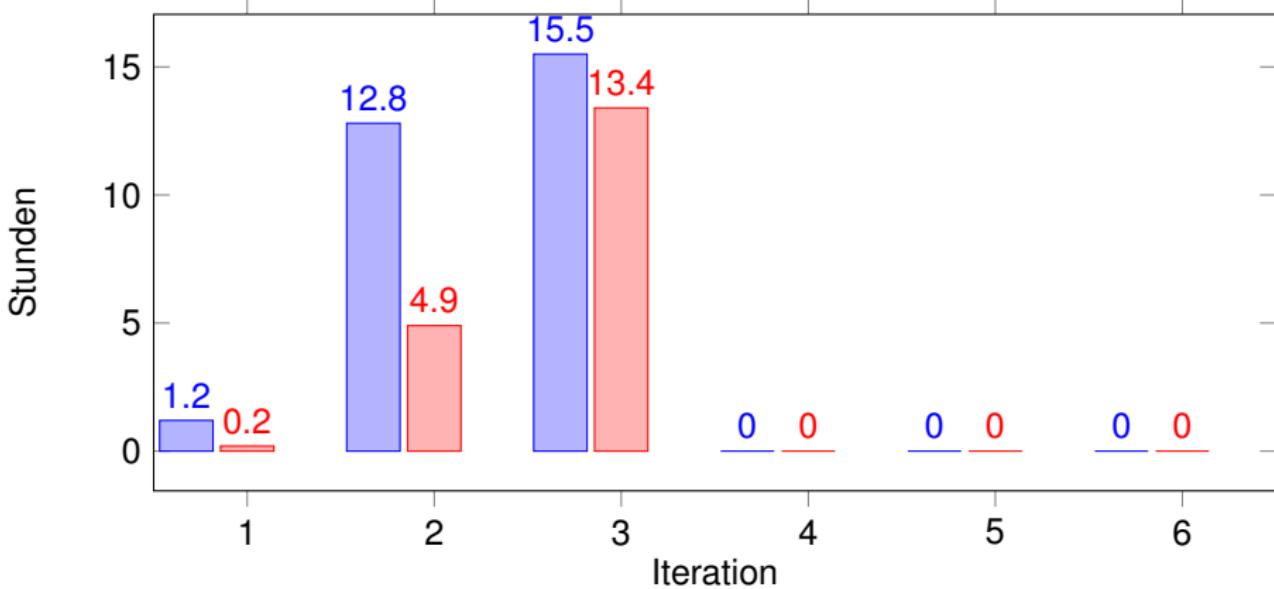
TheMangalex approved these changes 4 days ago

[View changes](#)

Auch andere Personen am Institut müssen das Interface verwenden können.

1. Nutzerstudie
2. Benutzeranleitung

Zeiterfassung



Fragen?