

# Eclipse Plugin zum Visualisieren von Graphen.

Bachelorpraktikum - Pflichtenheft Review 1

14.12.2006

Gruppe: G<sup>222</sup>

# Gliederung

- 1 Gruppenorganisation
- 2 Pflichtenheft
- 3 Use-Cases
- 4 Ausblick
- 5 Qualitätssicherung
- 6 Zeitplanung und -erfassung

# Gliederung

- 1 Gruppenorganisation
- 2 Pflichtenheft
- 3 Use-Cases
- 4 Ausblick
- 5 Qualitätssicherung
- 6 Zeitplanung und -erfassung

## verwendete Software

Infrastruktur - Nicht das Ziel, aber macht vieles Leichter

- SVN zur Versionsverwaltung
  - Benachrichtigung an Gruppenverteiler
- $\text{\LaTeX}$  zum Schreiben der Dokumente
- Projekt-Homepage mit automatischem Upload
- Mailverteiler
- ICQ zur schnellen Kommunikation
- Mantis - Bug Tracking System

## verwendete Software

Infrastruktur - Nicht das Ziel, aber macht vieles Leichter

- SVN zur Versionsverwaltung
  - Benachrichtigung an Gruppenverteiler
- $\text{\LaTeX}$  zum Schreiben der Dokumente
- Projekt-Homepage mit automatischem Upload
- Mailverteiler
- ICQ zur schnellen Kommunikation
- Mantis - Bug Tracking System

## verwendete Software

Infrastruktur - Nicht das Ziel, aber macht vieles Leichter

- SVN zur Versionsverwaltung
  - Benachrichtigung an Gruppenverteiler
- L<sup>A</sup>T<sub>E</sub>X zum Schreiben der Dokumente
- Projekt-Homepage mit automatischem Upload
- Mailverteiler
- ICQ zur schnellen Kommunikation
- Mantis - Bug Tracking System

## verwendete Software

Infrastruktur - Nicht das Ziel, aber macht vieles Leichter

- SVN zur Versionsverwaltung
  - Benachrichtigung an Gruppenverteiler
- L<sup>A</sup>T<sub>E</sub>X zum Schreiben der Dokumente
- Projekt-Homepage mit automatischem Upload
- Mailverteiler
- ICQ zur schnellen Kommunikation
- Mantis - Bug Tracking System

## verwendete Software

Infrastruktur - Nicht das Ziel, aber macht vieles Leichter

- SVN zur Versionsverwaltung
  - Benachrichtigung an Gruppenverteiler
- L<sup>A</sup>T<sub>E</sub>X zum Schreiben der Dokumente
- Projekt-Homepage mit automatischem Upload
- Mailverteiler
- ICQ zur schnellen Kommunikation
- Mantis - Bug Tracking System



## verwendete Software

Infrastruktur - Nicht das Ziel, aber macht vieles Leichter

- SVN zur Versionsverwaltung
  - Benachrichtigung an Gruppenverteiler
- L<sup>A</sup>T<sub>E</sub>X zum Schreiben der Dokumente
- Projekt-Homepage mit automatischem Upload
- Mailverteiler
- ICQ zur schnellen Kommunikation
- Mantis - Bug Tracking System

## verwendete Software

Infrastruktur - Nicht das Ziel, aber macht vieles Leichter

- SVN zur Versionsverwaltung
  - Benachrichtigung an Gruppenverteiler
- L<sup>A</sup>T<sub>E</sub>X zum Schreiben der Dokumente
- Projekt-Homepage mit automatischem Upload
- Mailverteiler
- ICQ zur schnellen Kommunikation
- Mantis - Bug Tracking System

# Einarbeitung

Komponenten die unser Programm enthalten soll

- Eclipse und Pluginkonzept
- Eclipse Plugin Entwicklung
- XML, GraphML
- $\LaTeX$  und spezielle Pakete
- UML-Diagramme
- Draw2D
- SWT

# Einarbeitung

Komponenten die unser Programm enthalten soll

- Eclipse und Pluginkonzept
- Eclipse Plugin Entwicklung
- XML, GraphML
- $\LaTeX$  und spezielle Pakete
- UML-Diagramme
- Draw2D
- SWT

# Einarbeitung

Komponenten die unser Programm enthalten soll

- Eclipse und Pluginkonzept
- Eclipse Plugin Entwicklung
- XML, GraphML
- $\text{\LaTeX}$  und spezielle Pakete
- UML-Diagramme
- Draw2D
- SWT

# Einarbeitung

Komponenten die unser Programm enthalten soll

- Eclipse und Pluginkonzept
- Eclipse Plugin Entwicklung
- XML, GraphML
- $\text{\LaTeX}$  und spezielle Pakete
- UML-Diagramme
- Draw2D
- SWT

# Einarbeitung

Komponenten die unser Programm enthalten soll

- Eclipse und Pluginkonzept
- Eclipse Plugin Entwicklung
- XML, GraphML
- $\text{\LaTeX}$  und spezielle Pakete
- UML-Diagramme
- Draw2D
- SWT

# Einarbeitung

Komponenten die unser Programm enthalten soll

- Eclipse und Pluginkonzept
- Eclipse Plugin Entwicklung
- XML, GraphML
- $\text{\LaTeX}$  und spezielle Pakete
- UML-Diagramme
- Draw2D
- SWT



# Einarbeitung

Komponenten die unser Programm enthalten soll

- Eclipse und Pluginkonzept
- Eclipse Plugin Entwicklung
- XML, GraphML
- $\text{\LaTeX}$  und spezielle Pakete
- UML-Diagramme
- Draw2D
- SWT

# Aufgabenverteilung

- XML-Evaluation
  - Marco Möller
- Draw2D/SWT-Einarbeitung
  - Andreas Franke
  - Peter Schauss
- Eclipse-Plugin-Entwicklung
  - Martin Konrad

# Aufgabenverteilung

- XML-Evaluation
  - Marco Möller
- Draw2D/SWT-Einarbeitung
  - Andreas Franke
  - Peter Schauss
- Eclipse-Plugin-Entwicklung
  - Martin Konrad

# Aufgabenverteilung

- XML-Evaluation
  - Marco Möller
- Draw2D/SWT-Einarbeitung
  - Andreas Franke
  - Peter Schauss
- Eclipse-Plugin-Entwicklung
  - Martin Konrad

# Aufgabenverteilung

- XML-Evaluation
  - Marco Möller
- Draw2D/SWT-Einarbeitung
  - Andreas Franke
  - Peter Schauss
- Eclipse-Plugin-Entwicklung
  - Martin Konrad

# Aufgabenverteilung

- XML-Evaluation
  - Marco Möller
- Draw2D/SWT-Einarbeitung
  - Andreas Franke
  - Peter Schauss
- Eclipse-Plugin-Entwicklung
  - Martin Konrad

# Aufgabenverteilung

- XML-Evaluation
  - Marco Möller
- Draw2D/SWT-Einarbeitung
  - Andreas Franke
  - Peter Schauss
- Eclipse-Plugin-Entwicklung
  - Martin Konrad

# Aufgabenverteilung

- XML-Evaluation
  - Marco Möller
- Draw2D/SWT-Einarbeitung
  - Andreas Franke
  - Peter Schauss
- Eclipse-Plugin-Entwicklung
  - Martin Konrad



# Gliederung

- 1 Gruppenorganisation
- 2 Pflichtenheft**
- 3 Use-Cases
- 4 Ausblick
- 5 Qualitätssicherung
- 6 Zeitplanung und -erfassung

# Verwendete Tools

Hiermit haben wir das Pflichtenheft erstellt

- $\text{L}^{\text{A}}\text{T}_{\text{E}}\text{X}$
- $\text{L}^{\text{A}}\text{T}_{\text{E}}\text{X}$ -Pakete
- Makefile
- automatisches On-Server-Make-On-Commit
- UMLet
- ...

# Verwendete Tools

Hiermit haben wir das Pflichtenheft erstellt

- $\text{\LaTeX}$
- $\text{\LaTeX}$ -Pakete
- Makefile
- automatisches On-Server-Make-On-Commit
- UMLet
- ...

# Verwendete Tools

Hiermit haben wir das Pflichtenheft erstellt

- $\text{\LaTeX}$
- $\text{\LaTeX}$ -Pakete
- Makefile
- automatisches On-Server-Make-On-Commit
- UMLet
- ...

# Verwendete Tools

Hiermit haben wir das Pflichtenheft erstellt

- $\text{\LaTeX}$
- $\text{\LaTeX}$ -Pakete
- Makefile
- automatisches On-Server-Make-On-Commit
- UMLet
- ...

# Verwendete Tools

Hiermit haben wir das Pflichtenheft erstellt

- $\text{\LaTeX}$
- $\text{\LaTeX}$ -Pakete
- Makefile
- automatisches On-Server-Make-On-Commit
- UMLet
- ...

# Verwendete Tools

Hiermit haben wir das Pflichtenheft erstellt

- $\text{\LaTeX}$
- $\text{\LaTeX}$ -Pakete
- Makefile
- automatisches On-Server-Make-On-Commit
- UMLet
- ...

# Inhalte des Pflichtenhefts

- **Einleitung**
- Vision
- Ist/Soll
- Featurelist
- Use-Case-Felderklärung
- Use-Cases
- Glossar
- Literatur/Links



# Inhalte des Pflichtenhefts

- **Einleitung**
- **Vision**
- Ist/Soll
- Featurelist
- Use-Case-Felderklärung
- Use-Cases
- Glossar
- Literatur/Links

# Inhalte des Pflichtenhefts

- Einleitung
- Vision
- Ist/Soll
- Featurelist
- Use-Case-Felderklärung
- Use-Cases
- Glossar
- Literatur/Links

# Inhalte des Pflichtenhefts

- **Einleitung**
- **Vision**
- **Ist/Soll**
- **Featurelist**
- Use-Case-Felderklärung
- Use-Cases
- Glossar
- Literatur/Links

# Inhalte des Pflichtenhefts

- Einleitung
- Vision
- Ist/Soll
- Featurelist
- Use-Case-Felderklärung
- Use-Cases
- Glossar
- Literatur/Links

# Inhalte des Pflichtenhefts

- Einleitung
- Vision
- Ist/Soll
- Featurelist
- Use-Case-Felderklärung
- Use-Cases
- Glossar
- Literatur/Links

# Inhalte des Pflichtenhefts

- Einleitung
- Vision
- Ist/Soll
- Featurelist
- Use-Case-Felderklärung
- Use-Cases
- Glossar
- Literatur/Links

# Inhalte des Pflichtenhefts

- Einleitung
- Vision
- Ist/Soll
- Featurelist
- Use-Case-Felderklärung
- Use-Cases
- Glossar
- Literatur/Links

# Gliederung

- 1 Gruppenorganisation
- 2 Pflichtenheft
- 3 Use-Cases**
- 4 Ausblick
- 5 Qualitätssicherung
- 6 Zeitplanung und -erfassung



# Ein Beispiel Use-Case

## Exportieren von Graphen nach SVG

<b>ID</b>	UC-GUI-Export	
<b>Status</b>	In Planung	
<b>Priorität</b>	Hoch	
<b>Akteur</b>	Benutzer	
<b>Kurzbeschreibung</b>	Das angezeigte Graphenlayout wird in einer SVG Datei gespeichert.	
<b>Vorbedingung</b>	Ein Graph wird angezeigt.	
<b>Nachbedingung</b>	Es existiert eine SVG Datei die dem Graphenlayout entspricht.	
	<b>Aktion</b>	<b>Reaktion</b>
<b>Normaler Ablauf</b>	1 Klicken auf den Export-Button. 3 Eingabe des Dateinamens und bestätigen.	2 Ein Dateiauswahldialog wird geöffnet. 4 Es wird <b>UC-API-Export</b> aufgerufen.
<b>Alternativer Ablauf</b>		

# Use-Case Kategorien

- Installation
- graphische Benutzeroberfläche
- Java-API

# Use-Case Kategorien

- Installation
- graphische Benutzeroberfläche
- Java-API

# Use-Case Kategorien

- Installation
- graphische Benutzeroberfläche
- Java-API

# Installations-Use-Cases

- Installation des Plugins
- Deinstallation des Plugins

# API-Use-Cases 1

## Erzeugen von Objekten

- Erzeugen des Graphen
- Erzeugen eines Knotens
- Erzeugen einer Kante
- Erzeugen einer Skriptumgebung
- Erzeugen eines Skripts

## API-Use-Cases 2

### Verketten und modifizieren von Objekten

- Hinzufügen eines Knoten
- Hinzufügen einer Kante
- Binden eines Skripts an Knoten oder Kante
- Setzen eines Filters
- Zurücksetzen eines Filters

## API-Use-Cases 3

### Operationen auf und mit einem Graphen

- Berechnung des Graphenlayouts
- Abbruch der Berechnung des Graphenlayouts
- Zeichnen des Graphen in GUI
- Exportieren von eines Graphen nach SVG
- Ausführen eines Skripts
- Speichern eines Graphen
- Laden eines Graphen



# GUI-Use-Cases 1

Analog zu Api-Use-Cases

- Setzen eines Filters
- Rücksetzen eines Filters
- Berechnung des Graphenlayouts
- Abbruch der Berechnung des Graphenlayouts
- Exportieren von Graphen nach SVG
- Ausführen eines Skripts
- Speichern eines Graphen
- Laden eines Graphen

# GUI-Use-Cases 2

reine GUI Funktionen

- Zoomen der Ansicht
- Verschieben der Ansicht (“Pan”)
- Reset Zoom & “Pan”
- Anzeige eines Graphen-Overview
- Ausfalten eines Knotens
- Zusammenfallen eines Knotens
- Anzeige zweier Graphen nebeneinander

# Gliederung

- 1 Gruppenorganisation
- 2 Pflichtenheft
- 3 Use-Cases
- 4 Ausblick**
- 5 Qualitätssicherung
- 6 Zeitplanung und -erfassung

# Proof-Of-Concept

- installierbares Minimalplugin
- SWT-Miniprogramm
- Draw2D-Beispiel
- ...

# Proof-Of-Concept

- installierbares Minimalplugin
- SWT-Miniprogramm
- Draw2D-Beispiel
- ...

# Proof-Of-Concept

- installierbares Minimalplugin
- SWT-Miniprogramm
- Draw2D-Beispiel
- ...

# Proof-Of-Concept

- installierbares Minimalplugin
- SWT-Miniprogramm
- Draw2D-Beispiel
- ...

# Design

- **API-Konzipierung**
- Festlegung einer Datenstruktur
- Festlegung eines Dateiformats
- Testprogramm zur SVG-Erzeugung
- ...



# Design

- API-Konzipierung
- Festlegung einer Datenstruktur
- Festlegung eines Dateiformats
- Testprogramm zur SVG-Erzeugung
- ...

# Design

- API-Konzipierung
- Festlegung einer Datenstruktur
- Festlegung eines Dateiformats
- Testprogramm zur SVG-Erzeugung
- ...

# Design

- API-Konzipierung
- Festlegung einer Datenstruktur
- Festlegung eines Dateiformats
- Testprogramm zur SVG-Erzeugung
- ...

# Design

- API-Konzipierung
- Festlegung einer Datenstruktur
- Festlegung eines Dateiformats
- Testprogramm zur SVG-Erzeugung
- ...

# Gliederung

- 1 Gruppenorganisation
- 2 Pflichtenheft
- 3 Use-Cases
- 4 Ausblick
- 5 Qualitätssicherung**
- 6 Zeitplanung und -erfassung

## Ziele Qualitätssicherung (ISO/IEC 9126)

- **Funktionalität**
- Zuverlässigkeit
- Benutzbarkeit
- Effizienz
- Änderbarkeit
- Übertragbarkeit

## Ziele Qualitätssicherung (ISO/IEC 9126)

- Funktionalität
- Zuverlässigkeit
- Benutzbarkeit
- Effizienz
- Änderbarkeit
- Übertragbarkeit

## Ziele Qualitätssicherung (ISO/IEC 9126)

- Funktinalität
- Zuverlässigkeit
- Benutzbarkeit
- Effizienz
- Änderbarkeit
- Übertragbarkeit



## Ziele Qualitätssicherung (ISO/IEC 9126)

- Funktinalität
- Zuverlässigkeit
- Benutzbarkeit
- Effizienz
- Änderbarkeit
- Übertragbarkeit

## Ziele Qualitätssicherung (ISO/IEC 9126)

- Funktinalität
- Zuverlässigkeit
- Benutzbarkeit
- Effizienz
- Änderbarkeit
- Übertragbarkeit

## Ziele Qualitätssicherung (ISO/IEC 9126)

- Funktinalität
- Zuverlässigkeit
- Benutzbarkeit
- Effizienz
- Änderbarkeit
- Übertragbarkeit

# Hilfsmittel zur Qualitätssicherung

- RUP-Modell
- Pair-Programming
- SVN zur Versionsverwaltung
- Code-Conventions
- JUnit-Tests
- Systemtests
- Bug-Tracking System

# Hilfsmittel zur Qualitätssicherung

- RUP-Modell
- Pair-Programming
- SVN zur Versionsverwaltung
- Code-Conventions
- JUnit-Tests
- Systemtests
- Bug-Tracking System

# Hilfsmittel zur Qualitätssicherung

- RUP-Modell
- Pair-Programming
- SVN zur Versionsverwaltung
- Code-Conventions
- JUnit-Tests
- Systemtests
- Bug-Tracking System

# Hilfsmittel zur Qualitätssicherung

- RUP-Modell
- Pair-Programming
- SVN zur Versionsverwaltung
- Code-Conventions
- JUnit-Tests
- Systemtests
- Bug-Tracking System

# Hilfsmittel zur Qualitätssicherung

- RUP-Modell
- Pair-Programming
- SVN zur Versionsverwaltung
- Code-Conventions
- JUnit-Tests
- Systemtests
- Bug-Tracking System



# Hilfsmittel zur Qualitätssicherung

- RUP-Modell
- Pair-Programming
- SVN zur Versionsverwaltung
- Code-Conventions
- JUnit-Tests
- Systemtests
- Bug-Tracking System

# Hilfsmittel zur Qualitätssicherung

- RUP-Modell
- Pair-Programming
- SVN zur Versionsverwaltung
- Code-Conventions
- JUnit-Tests
- Systemtests
- Bug-Tracking System

# Gliederung

- 1 Gruppenorganisation
- 2 Pflichtenheft
- 3 Use-Cases
- 4 Ausblick
- 5 Qualitätssicherung
- 6 Zeitplanung und -erfassung**

# Zeitplanung und -erfassung

- 1000 h Zielmarke
- 150 h Bisher
  - 5% Qualitätssicherung
  - 20% Treffen für Designfragen
  - 35% Ausarbeitung des Designs
  - 40% Toolsuche und Installation
- Stundenerfassung in Datei im SVN

# Zeitplanung und -erfassung

- 1000 h Zielmarke
- 150 h Bisher
  - 5% Qualitätssicherung
  - 20% Treffen für Designfragen
  - 35% Ausarbeitung des Designs
  - 40% Toolsuche und Installation
- Stundenerfassung in Datei im SVN

# Zeitplanung und -erfassung

- 1000 h Zielmarke
- 150 h Bisher
  - 5% Qualitätssicherung
  - 20% Treffen für Designfragen
  - 35% Ausarbeitung des Designs
  - 40% Toolsuche und Installation
- Stundenerfassung in Datei im SVN

# Zeitplanung und -erfassung

- 1000 h Zielmarke
- 150 h Bisher
  - 5% Qualitätssicherung
  - 20% Treffen für Designfragen
  - 35% Ausarbeitung des Designs
  - 40% Toolsuche und Installation
- Stundenerfassung in Datei im SVN

# Zeitplanung und -erfassung

- 1000 h Zielmarke
- 150 h Bisher
  - 5% Qualitätssicherung
  - 20% Treffen für Designfragen
  - 35% Ausarbeitung des Designs
  - 40% Toolsuche und Installation
- Stundenerfassung in Datei im SVN



# Zeitplanung und -erfassung

- 1000 h Zielmarke
- 150 h Bisher
  - 5% Qualitätssicherung
  - 20% Treffen für Designfragen
  - 35% Ausarbeitung des Designs
  - 40% Toolsuche und Installation
- Stundenerfassung in Datei im SVN

# Zeitplanung und -erfassung

- 1000 h Zielmarke
- 150 h Bisher
  - 5% Qualitätssicherung
  - 20% Treffen für Designfragen
  - 35% Ausarbeitung des Designs
  - 40% Toolsuche und Installation
- Stundenerfassung in Datei im SVN

# Überblick

- **bisher**
  - Gruppenorganisation
  - Pflichtenheft
  - Use-Cases
  - Qualitätssicherung
  - Zeitplanung
- **in Zukunft**
  - Tool Evolution Abschließen
  - Demo Programme schreiben
  - Design Dokument erstellen
  - Qualitätssicherung weiter planen

# Überblick

- bisher
  - Gruppenorganisation
    - Pflichtenheft
    - Use-Cases
    - Qualitätssicherung
    - Zeitplanung
- in Zukunft
  - Tool Evolution Abschließen
  - Demo Programme schreiben
  - Design Dokument erstellen
  - Qualitätssicherung weiter planen

# Überblick

- bisher
  - Gruppenorganisation
  - Pflichtenheft
  - Use-Cases
  - Qualitätssicherung
  - Zeitplanung
- in Zukunft
  - Tool Evolution Abschließen
  - Demo Programme schreiben
  - Design Dokument erstellen
  - Qualitätssicherung weiter planen

# Überblick

- bisher
  - Gruppenorganisation
  - Pflichtenheft
  - Use-Cases
  - Qualitätssicherung
  - Zeitplanung
- in Zukunft
  - Tool Evaluation abschließen
  - Demo Programme schreiben
  - Design Dokument erstellen
  - Qualitätssicherung weiter planen

# Überblick

- bisher
  - Gruppenorganisation
  - Pflichtenheft
  - Use-Cases
  - Qualitätssicherung
  - Zeitplanung
- in Zukunft
  - Tool Evaluation Abschließen
  - Demo Programme schreiben
  - Design Dokument erstellen
  - Qualitätssicherung weiter planen

# Überblick

- bisher
  - Gruppenorganisation
  - Pflichtenheft
  - Use-Cases
  - Qualitätssicherung
  - Zeitplanung
- in Zukunft
  - Tool Evaluation Abschließen
  - Demo Programme schreiben
  - Design Dokument erstellen
  - Qualitätssicherung weiter planen



# Überblick

- bisher
  - Gruppenorganisation
  - Pflichtenheft
  - Use-Cases
  - Qualitätssicherung
  - Zeitplanung
- in Zukunft
  - Tool Evaluation Abschließen
  - Demo Programme schreiben
  - Design Dokument erstellen
  - Qualitätssicherung weiter planen

# Überblick

- bisher
  - Gruppenorganisation
  - Pflichtenheft
  - Use-Cases
  - Qualitätssicherung
  - Zeitplanung
- in Zukunft
  - Tool Evaluation Abschließen
  - Demo Programme schreiben
  - Design Dokument erstellen
  - Qualitätssicherung weiter planen

# Überblick

- bisher
  - Gruppenorganisation
  - Pflichtenheft
  - Use-Cases
  - Qualitätssicherung
  - Zeitplanung
- in Zukunft
  - Tool Evaluation Abschließen
  - Demo Programme schreiben
  - Design Dokument erstellen
  - Qualitätssicherung weiter planen

# Überblick

- bisher
  - Gruppenorganisation
  - Pflichtenheft
  - Use-Cases
  - Qualitätssicherung
  - Zeitplanung
- in Zukunft
  - Tool Evaluation Abschließen
  - Demo Programme schreiben
  - Design Dokument erstellen
  - Qualitätssicherung weiter planen

# Überblick

- bisher
  - Gruppenorganisation
  - Pflichtenheft
  - Use-Cases
  - Qualitätssicherung
  - Zeitplanung
- in Zukunft
  - Tool Evaluation Abschließen
  - Demo Programme schreiben
  - Design Dokument erstellen
  - Qualitätssicherung weiter planen

# Fragen?

