# Projektantrag

Das Ziel des Projektes ist es Aufzuzeigen wie Eclipse RCP 3 Applikationen nach Eclipse RCP 4 migriert werden können

Das vorliegende Dokument ist der Projektantrag der die beteiligten

Parteien über den Projektrahmen, die Projektziele und den Projektumfang

informieren soll.

|  |  |
| --- | --- |
| Autor: | Mike Rothenbühler |
| Version: | 0.1 |
| Status: | In Arbeit |
| Ablage: | <https://github.com/MikeR13/MAS/blob/master/Deliverables/Projektantrag.docx> |
| Institution: | Hochschule für Technik und Informatik Bern |
| Verteiler: | Brawand Ueli, Hoffmann Marc, Rothenbühler Mike |

**Versionkontrolle**

|  |  |  |  |
| --- | --- | --- | --- |
| **Datum** | **Version** | **Autor** | **Bemerkungen** |
| 09.05.2013 | 0.1 | MIRO | Erster Wurf |
| 12.05.2013 | 0.2 | MIRO |  |
|  |  |  |  |

Inhaltsverzeichnis

[Projektantrag 1](#_Toc356145344)

[Einleitung 4](#_Toc356145345)

[1. Zweck des Dokumentes 4](#_Toc356145346)

[Ausgangslage 4](#_Toc356145347)

[2. Anlass und Begründung des Projektes 4](#_Toc356145348)

[3. Problemstellung 5](#_Toc356145349)

[4. Randbedingungen 5](#_Toc356145350)

[5. Situationsanalyse 5](#_Toc356145351)

[6. Erbrachte Vorleistung 5](#_Toc356145352)

[Abgrenzung 6](#_Toc356145353)

[Ziele und Lösungen 6](#_Toc356145354)

[7. Zielvorstellungen 6](#_Toc356145355)

[8. Lösungen 6](#_Toc356145356)

[9. Sicherheits- und Datenschutzaspekte 6](#_Toc356145357)

[Chancen und Risiken 6](#_Toc356145358)

[10. Risikoidentifizierung 6](#_Toc356145359)

[11. Risikoursachenanalyse 7](#_Toc356145360)

[12. Risikobewertung und Risikoquantifizierung, 7](#_Toc356145361)

[Mittelbedarf 7](#_Toc356145362)

[13. Hardware 7](#_Toc356145363)

[14. Software 7](#_Toc356145364)

[15. Personal 7](#_Toc356145365)

[16. Sonstiges 7](#_Toc356145366)

[Planung und Organisation 7](#_Toc356145367)

[17. Projektorganisation 7](#_Toc356145368)

[18. Termine 7](#_Toc356145369)

[19. Prioritäten 8](#_Toc356145370)

[20. Projektmethode 8](#_Toc356145371)

[Wirtschaftlichkeit 9](#_Toc356145372)

[21. Kosten 9](#_Toc356145373)

[22. Nutzen 9](#_Toc356145374)

[Konsequenzen 10](#_Toc356145375)

[23. Bei Realisierung 10](#_Toc356145376)

[24. Bei verspäteter Realisierung 10](#_Toc356145377)

[25. Bei Nichtrealisierung 10](#_Toc356145378)

[Antrag 10](#_Toc356145379)

[26. Bisherige Entscheide 10](#_Toc356145380)

[27. Projektantrag 10](#_Toc356145381)

# Einleitung

## Zweck des Dokumentes

Dieser Projektantrag informiert die an „Migration von Eclipse 3.x nach Eclipse 4“ beteiligten Parteien über den Projektrahmen, die Projektziele und den Projektumfang. Das Dokument soll in erster Linie einen Überblick über das Projekt ermöglichen.

# Ausgangslage

## Anlass und Begründung des Projektes

Die SBB IT haben in mehreren Projekten Eclipse 3.x Rich Client Platform (im Folgenden RCP genannt) im Einsatz. Die neue Version Eclipse 4 RCP wurde bereits in mehreren neueren Projekten eingesetzt und hat sich bewährt.

Eclipse RCP kurz erklärt (aus <http://de.wikipedia.org/wiki/Eclipse_(IDE))>:

„*Eclipse bietet die Rich Client Platform, welche es Anwendungsentwicklern ermöglicht, basierend auf dem Eclipse Framework, von der Eclipse-IDE unabhängige Anwendungen zu schreiben. Eine Übersicht befindet sich auf der Eclipse-Homepage.[<http://eclipse.org/community/rcp.php>] Beispielsweise basiert IBM Workplace Clients auf Eclipse RCP.*

*Die folgenden Komponenten (Plug-ins) werden mindestens für ein Eclipse-Rich-Client-Platform-Programm benötigt:*

* *Eclipse Core Platform – steuert den Lebenszyklus einer Eclipse-Application*
* *Standard Widget Toolkit (SWT)*
* *JFace*

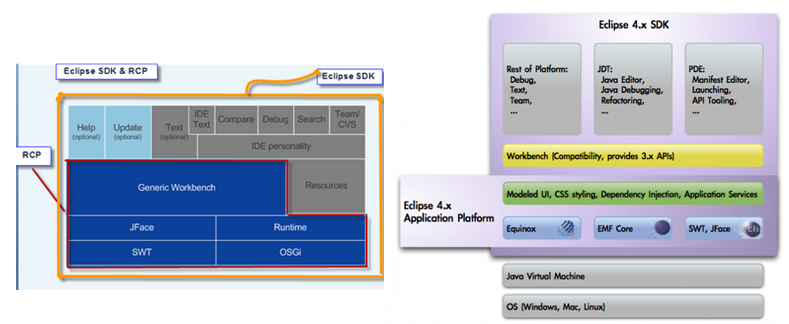
*Weitere Eclipse-Komponenten, wie das Hilfesystem oder das automatische Aktualisierungssystem, können auch eingesetzt werden. Üblicherweise wird das Eclipse-OSGi-Framework Equinox zum Kombinieren („bundling“) der Komponenten eingesetzt.*“

Eclipse RCP 4 bietet ein sehr viel einfacheres Programmiermodell an als sein Vorgänger. Die vielfältigen APIs aus RCP 3.x wurden deutlich reduziert und vereinheitlicht. Die über das gesamte API verteilten Singletons wurden entfernt.

Neben der Vereinfachung wurden moderne Konzepte wie Dependency Injection und Declarative Styling eingeführt.

Die Implementation von Rich Client Applikation in Eclipse RCP wird mit der Version 4 flexibler und deutlich vereinfacht. Die Produktivität der Programmierer steigt, die Testbarkeit und die Wartung der Applikationen wird erleichtert.

Die Migration auf die neue Version ist somit auch für bestehende Eclipse RCP Applikationen durchaus ein Thema. Eclipse 3.x wird aktuell noch unterstützt, es könnte jedoch schon bald nicht mehr der Fall sein.



## Problemstellung

Da sich mit der Version 4 einiges an Eclipse RCP geändert hat ist eine Migration nicht einfach so zu bewerkstelligen. Es gibt aus der Community (noch) nicht viele Berichte zu gelungenen Migrationen, geschweige denn wie eine solche Migration erfolgreich durchgeführt werden kann.

Es sollen Erkenntnisse gewonnen werden, wie eine erfolgreiche Migration durchgeführt werden kann, ohne dass die bestehende Applikation in den Punkten

* Funktionalität
* Performance
* Stabilität
* Usability
* Look and Feel

negativ beeinflusst wird. Die Arbeit an der bestehenden Applikation soll auch während der Migrationszeit möglich sein. Hierfür müssen Lösungen erarbeitet werden.

Mit den Erfahrungen und Ergebnissen aus der Master Thesis soll eine Migration auch für grosse Projekte relativ einfach möglich sein.

## Randbedingungen

Das Projekt wird im Rahmen der Semesterarbeit und der Diplomarbeit durchgeführt. Dafür gelten die an der HTI üblichen Bedingungen.

## Situationsanalyse

Um die erarbeiteten Migrationsvorschläge durchzuführen und nachzuweisen, dass diese auch funktionieren wird das SBB-Projekt RCS (Rail Control System) als Versuchsobjekt genommen (**TODO** besserer Satz..)

Der Client von RCS weist folgende Kennzahlen auf:

* 70 Plugins
* über 10 Hauptfenster
* Dutzende von Dialogen
* 270‘000 Zeilen Code

**TODO RCS kurz fachlich und auch technisch erklären?**

## Erbrachte Vorleistung

Das Know How über Eclipse RCP 4 wurde bereits vor Projektstart aufgebaut.

**SWOT Analyse??**

# Abgrenzung

Aus Zeitgründen wird es nicht möglich sein während der Projektdauer das gesamte RCS auf Eclipse RCP 4 zu migrieren. Es werden ausschliesslich die ausgewählten Aspekte migriert.

# Ziele und Lösungen

## Zielvorstellungen

Das Ziel des Projektes ist es mindestens 5 Aspekte von Eclipse RCP 3 auf Eclipse 4 RCP zu migrieren. Als Aspekt wird zum Beispiel das plugin.xml, die Selektion oder auch Dependency Injection betrachtet.

Ein Aspekt gilt als erfolgreich abgehandelt wenn

* der Handbucheintrag - wie die Migration erfolgen kann - gemacht ist
* der praktische Beweis im Code erbracht worden ist, dass es funktioniert
* Risiko (Evaluation, Beurteilung, Behandlung etc..)
* Vergleich zu anderen möglichen Lösungen?
* Tests/Qualität
* **TODO** Was noch????

Die Erfahrungen im Projekt sollen in einem Bericht geschildert werden.

## Lösungen

Die Lösung eines Aspektes wird jeweils während einer Iteration erarbeitet. Nach dem heutigen Wissenstand gibt es noch keine möglichen Lösungsvarianten.

**TODO** hier irgendwo Anforderungen an Migration reinbringen: <http://de.wikipedia.org/wiki/Migrationsstrategien_(Informationstechnik)>

## Sicherheits- und Datenschutzaspekte

Der Code der SBB darf nicht an Dritte gelangen. Um dies zu verhindern wird der Code ausschliesslich im SBB Netz bearbeitet und dann höchstens Ausschnitte daraus publiziert. Der Datenschutzaspekt muss noch mit den SBB geklärt werden.

# Chancen und Risiken

**TODO**

## Risikoidentifizierung

Migration wegen unbekannten technischen Problemen nicht möglich

Verlust von Funktionalität

Performanceeinbussen

Verlust von Stabilität

Verlust von Usability

Wartbarkeit nimmt ab

Look and Feel wird von Anwendern nicht toleriert

## Risikoursachenanalyse

**TODO**

## Risikobewertung und Risikoquantifizierung,

**TODO**

# Mittelbedarf

## Hardware

PC bzw. Laptop

## Software

Plattform: Java 7, Eclipse RCP

Dokumentation: Microsoft Office

## Personal

Projektrealisator: Mike Rothenbühler  
Projektbetreuer: Marc Hoffmann  
Experte: Ueli Brawand

## Sonstiges

Code Verwaltung: SVN/Repository bei SBB

Buildinfrastruktur: Hudson

# Planung und Organisation

## Projektorganisation

|  |  |  |
| --- | --- | --- |
| **Rolle** | **Name** | **Email** |
| Projektumsetzung | Mike Rothenbühler | [michael.rothenbuehler@sbb.ch](mailto:michael.rothenbuehler@sbb.ch) |
| Projektbetreuer | Marc Hoffmann | [marc.hoffmann@sbb.ch](mailto:marc.hoffmann@sbb.ch) |
| Experte | Ueli Brawand | [ueli.brawand@besonet.ch](mailto:ueli.brawand@besonet.ch) |

## Termine

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **Datum** | **Termin** | **Beteiligte Personen** | **Rolle** | **Status ?? TODO** |
| 02.05.2013 | Kickoff Meeting | Brawand Ueli | Experte | Erledigt, 02.05.2013 |
| Hoffmann Marc | Betreuer |
| Rothenbühler Mike | Projektumsetzer |
| 21.05.2013 | Abstract | Hoffmann Marc | Reviewer | In Arbeit |
| Rothenbühler Mike | Realisator |
| 21.05.2013 | Projektantrag | Hoffmann Marc | Reviewer | In Arbeit |
| Rothenbühler Mike | Realisator |
| 16.06.2013 | Statusbericht | Brawand Ueli |  | Wie oft jetzt? |
| Hoffmann Marc |  |
| Rothenbühler Mike |  |
| 30.06.2013 | Statusbericht | Brawand Ueli |  | Wie oft jetzt? |
| Hoffmann Marc |  |
| Rothenbühler Mike |  |
| TBD | Review | Brawand Ueli |  |  |
| Hoffmann Marc |  |
| Rothenbühler Mike |  |
| 14.07.2013 | Statusbericht | Brawand Ueli |  | Wie oft jetzt? |
| Hoffmann Marc |  |
| Rothenbühler Mike |  |
| 28.07.2013 | Statusbericht | Brawand Ueli |  | Wie oft jetzt? |
| Hoffmann Marc |  |
| Rothenbühler Mike |  |
| 04.08.2013 | Statusbericht | Brawand Ueli |  | Wie oft jetzt? |
| Hoffmann Marc |  |
| Rothenbühler Mike |  |
| **TBD** | Schlussreview |  |  |  |
| 12.09.2013 | Eingabe Präsentationsbedürfnisse | Hoffmann Marc |  |  |
| Rothenbühler Mike |  |
| 12.09.2013 | Abgabe Dokumentation/Anhänge | Hoffmann Marc |  |  |
| Rothenbühler Mike |  |
| 12.09.2013 | Upload Poster | Hoffmann Marc |  |  |
| Rothenbühler Mike |  |  |

Evtl. noch Rolle der „Teilnehmer“

**TODO Mehr Termine??**

## Prioritäten

1. Ein Aspekt nach dem anderen migrieren
2. TODO

## Projektmethode

Das Projekt wird mit Kanban durchgeführt. Der Betreuer bestimmt das aktuelle Backlog und priorisiert die Backlog Items. Das Backlog kann jederzeit vom Betreuer angepasst und neu priorisiert werden.

Es sollen maximal 2 Items in Bearbeitung sein (WIP, Work in progress).

**TODO** Erwähnen Board? Soll die Iteration timeboxed, oder bis ein Aspekt fertig ist sein? ETC..

# Wirtschaftlichkeit

## Kosten

Kosten entstehen in diesem Sinne keine, es wird mit einem Zeitaufwand von 360 Stunden im Zeitraum vom 21.04.2013 bis 15.09.2013 gerechnet. Dazu kommen noch der Zeitaufwand des Betreuers sowie des Experten (**TODO** wie kann dieses Kapitel besser gestaltet werden?)

## Nutzen

Eclipse RCP 4 bietet ein sehr viel einfacheres Programmiermodell an als sein Vorgänger. Die vielfältigen APIs aus RCP 3.x wurden deutlich reduziert und vereinheitlicht. Die über das gesamte API verteilten Singletons wurden entfernt.

Neben der Vereinfachung wurden moderne Konzepte wie Dependency Injection und Declarative Styling eingeführt.

Die Implementation von Rich Client Applikation in Eclipse RCP wird mit der Version 4 flexibler und deutlich vereinfacht. Die Produktivität der Programmierer steigt, die Testbarkeit und die Wartung der Applikationen wird erleichtert.

Die Migration auf die neue Version ist somit auch für bestehende Eclipse RCP Applikationen durchaus ein Thema. Eclipse 3.x wird aktuell noch unterstützt, es könnte jedoch schon bald nicht mehr der Fall sein.

Der wirkliche Nutzen soll am Ende des Projektes in reflexiver Weise beurteilt werden.

# Konsequenzen

## Bei Realisierung

Eclipse RCP Applikationen können erfolgreich migriert werden und nach der Migration von den Vereinfachungen und auch Verbesserungen profitieren.

## Bei verspäteter Realisierung

Eine verspätete Realisierung ist nicht möglich, da die Termine fix sind.

## Bei Nichtrealisierung

Wird erkannt, dass das Projekt aus zeitlichen oder technischen Gründen nicht realisierbar ist, so wird es nicht als Diplomarbeit weitergeführt.

# Antrag

## Bisherige Entscheide

## Projektantrag

**TODO**

Qualitative und quantitative Kriterien

Wo Projektvorgehen (Kanban)

Zeitbudget