
Websummary von Elsener David[[Summary bearbeiten](#)]

NeTS Event Monitor

Visualisierung des Event-Systems in einer Eclipse RCP Applikation

Elsener David, Netcetera AG

Umfeld

Die IPA wurde im Rahmen des Projektes NeTS absolviert. NeTS steht für "Netzweites Trassenplanungssystem" und ist das Werkzeug, das die SBB für das Erstellen des schweizweiten Fahrplanes seit Mitte Dezember 2009 einsetzt. NeTS basiert auf einer Client/Server Architektur, wobei die Client-Seite als Eclipse RCP Applikation realisiert ist. Das Arbeitsgebiet für die IPA beschränkte sich dabei auf die Client-Seite.

Das Aktualisieren und Synchronisieren der unterschiedlichen Komponenten des User Interfaces in NeTS wird über ein hauseigenes, zentrales Event-System gesteuert. Über das Event-System werden unterschiedliche Arten von Ereignissen verarbeitet, welche die Darstellung von Daten beeinflussen und auf welche die unterschiedlichen UI-Komponenten entsprechend reagieren müssen.

Aufgrund der über die Zeit angewachsenen Komplexität des Event-Systems, ist es für das Entwickler-Team oft schwierig, Fehlverhalten aufzuspüren und zu analysieren, da das Event-System nur eine sehr beschränkte Einsicht in die Verarbeitung von Events ermöglicht.

Aufgabenstellung

Im Rahmen der IPA sollte ein Tool zur Visualisierung des Event-Systems entwickelt werden. Da es als Unterstützung des Entwickler-Teams gedacht ist, sollen mit diesem Tool alle relevanten Informationen über das Event-System übersichtlich dargestellt werden. Folgende Funktionalitäten muss das neue Tool, das in Form eines Eclipse RCP Plugins entwickelt werden sollte, enthalten:

- Es soll möglich sein, verschiedene Arten von Events darzustellen.
- Für jeden Event sollen die Quelle, das Ziel, die Art und die Argumente des Events ersichtlich sein.
- Der Benutzer hat die Möglichkeit, für ihn nicht relevante Events zu filtern.
- Die dargestellten Events können beliebig sortiert werden.
- Die Grundfunktionalität des Tools soll als separat installierbares Eclipse Plugin implementiert werden.

Vorgehensweise

Um das Projekt möglichst genau planen und effizient realisieren zu können, habe ich das Projekt in folgende Projektphasen gegliedert:

- Projektinitialisierung: Beinhaltet in erster Linie den Projektantrag und die Projektplanung.
- Voranalyse: Dient zur Abgrenzung des Systems und der Analyse des Umfelds.
- Konzept: Ausarbeitung und Konzipierung der Lösungsvarianten. Definition der nicht-funktionalen Testfälle.
- Realisierung: Umsetzung der in der Konzeptphase geplanten Lösungsvarianten.
- Testing: Nicht-funktionale Tests durchführen (GUI-Testing) und Logik mittels JUnit testen.
- Abschlussarbeiten

Lösung

Die Realisierung der Logik für das Abfangen von Events habe ich mit AspectJ, einer Erweiterung für aspektorientiertes Programmieren mit Java, umgesetzt. Das Feuern von Events geschieht über einen hauseigenen, zentralen Manager. Dieser stellt die nötige Funktionalität zur Verfügung, um sich für Events zu registrieren und um sie zu feuern. Wird nun ein Event geworfen, benachrichtigt der Manager alle für diesen Event registrierten Listeners. AspectJ erlaubte es mir nun, mich mittels Aspekten in gewisse Codestellen zu

schleusen, ohne dass ich die betroffenen Stellen verändern musste. Damit konnte ich mich in die Methode, welche für das Feuern von Events zuständig ist, schleusen und meine Logik einbauen.

Eine weitere Schwierigkeit war die korrekte Handhabung und Visualisierung von verschachtelten Events. Es kann sein, dass aufgrund eines Events weitere Events angestossen werden, was zur Bildung einer Hierarchie führt. Das Aufbauen der Hierarchie löste ich mittels einer Rekursion. Als Grundlage dafür dient das Event-Model, das unter anderem die Start- und Endzeit eines Events beinhaltet. Für jeden bereits vorhandenen Event wird überprüft, ob dieser eine Endzeit hat und damit bereits fertig auf ihn reagiert wurde. Wenn dies nicht der Fall ist, weiss ich, dass der neue Event während der Abarbeitung des aktuellen Events geworfen wurde.

Das GUI wurde mittels SWT und JFace Komponenten umgesetzt. Die Informationen über die abgefangenen Events werden auf eine Übersichts- und Detailansicht aufgeteilt. Hierbei hört die Detailansicht auf die Selektion in der Übersichtsliste und stellt detaillierte Informationen über den selektierten Event dar.

The screenshot shows two windows from the EventMonitor application. The top window, titled 'EventMonitor', displays a table of events. The bottom window, titled 'EventDetails', shows the details for the selected event.

Timestamp (ms)	EventType	Source	ObjectClass	Number of ...	Number...
119781	EMPTY SELECTION	TrasseListView		2	0
121735	SELECTION	TrasseListView	ListTrasseModel	5	1
124047	SELECTION	TedView	TedTrasseModel	1	1
126422	CHANGED	TrassenOperatio...	Trasseable	6	1
128531	SELECTION	TedView	TedTrasseModel	1	1
139125	SELECTION	TedView	TedTrasseModel	1	1
148422	CHANGED	TrassenOperatio...	Trasseable	6	1

The 'EventDetails' window shows the following information for the selected event:

- Event**
 - EventType: CHANGED
 - Source: ch.sbb.nets.rcp.core.service.TrassenOperationenServiceWrapper
 - Source: 148422-151156 ms
- Arguments**
 - ObjectClass: ch.sbb.nets.rcp.core.model.Trasseable
 - Number of Arguments: 1
 - Arguments: ServiceHelper.1
- Handlers**

Timestamp (ms)	Class	Duration (ms)
149531	TedView.3	1625
148735	TrasseListView	796

Fazit

Ein wichtiges Kriterium für jedes IT-Projekt ist sicherlich eine gute Vorgehensweise und damit eine gute Projektplanung. Ich habe mir deshalb genug Zeit genommen, um meine Projektgliederung zu durchdenken und dann die Zeitplanung entsprechend vorzunehmen. Das Resultat und auch der Vergleich der geplanten und der schlussendlich verwendeten Zeit zeigen, dass mir das grundsätzlich gelungen ist. Eine Ausnahme war jedoch die Realisierung der Filterfunktionalität für die Übersichtsansicht. Ich habe mich zu sehr auf meine Erfahrung im Projekt verlassen und die Problematik zu wenig durchdacht, was bei der Realisierung zu Problemen führte und mehr Zeit in Anspruch nahm, als geplant.

Etwas Wichtiges, das ich aus diesem Projekt mitnehme, ist, dass es in der Konzeptphase notwendig ist, alles, und damit auch klare und für einfach gehaltene Sachverhalte zu analysieren und dementsprechend zu planen. Denn damit werden unerwartete Probleme in der Realisierung vorgebeugt.

Ansonsten bin ich sehr zufrieden mit meiner Arbeit. Die Projektgliederung in die verschiedenen Phasen war erfolgreich und hat mich gut zum Ziel geführt.

pkorg - ein Produkt der Ivaris AG
Copyright 2009 Ivaris AG. Alle Rechte vorbehalten.
Design 2009 by dezember und juli