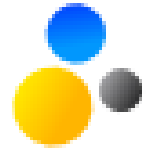


Konzeptvortrag
zur Diplomarbeit:

„Integration des
Eclipse-Communication-Frameworks
in Saros“

von Björn Gustavs

- Was ist „Saros“
 - Plug-In für Eclipse zur verteilten Paar Programmierung (*Distributed Pair Programming*)
 - entstand als Diplomarbeit 2006 an der FU-Berlin
 - stetige Weiterentwicklung in der Arbeitsgruppe „Software Engineering“



- verteilte Paar-/Partyprogrammierung?
 - mehrere Entwickler arbeiten an verschiedenen Orten am gleichen Projekt
 - Hauptziele:
 - Teilnehmer arbeiten an identischem Stand eines Projektes
 - Teilnehmer sehen wo andere im Code arbeiten
 - Mittel: Änderungen und Steuerinformationen eines Entwicklers werden an alle anderen übertragen und reproduziert/angewendet

Saros - Netzwerkschicht

- Saros nutzt XMPP (Jabber) zur Kommunikation zwischen Teilnehmern einer Sitzung
- nutzt XMPP Bibliothek „Smack“ (2 Jahre inaktiv)
- alle Teilnehmer sind am XMPP-Netzwerk angemeldet
- Änderungsinformationen (Aktivitäten) und weitere Statusinformationen werden als Nachrichten an andere Teilnehmer gesendet
- Dateiübertragung via In-Band-Bytestream, Jingle, Nachrichten, Socks5 Bytestream

Eclipse Communication Framework

- Eclipse Communication Framework (ECF), was ist das?
 - stellt APIs für Interprozesskommunikation und auch Benutzerkommunikation bereit
 - zielt darauf ab, Kommunikationsfunktionalitäten für Entwickler von Eclipse Anwendungen bereitzustellen
 - Messaging, Presence, File Transfer, ...
 - Basiert auf
 - erweiterbarer „Kommunikationscontainer“
 - zentraler Zugriff auf Kommunikation über ECF
 - erweiterbar mittels Adapter
 - Provider: Kommunikationsprotokolle werden gekapselt und dem Container zur Verfügung gestellt (XMPP, MSN, IRC, Yahoo, Bittorrent,...)

Ziel dieser Diplomarbeit

- Ziel dieser Diplomarbeit?
 - möglichst viele Kommunikationselemente von Saros durch ECF-Komponenten ersetzen
 - Entfernen von Netzwerkcode aus Saros
 - evtl. Code von Saros in das ECF Projekt einbringen

Vorteile / Gefahren

- Vorteile
 - Auslagern (von Teilen) der Netzwerkkomponente
 - Nutzen der Erfahrung von Experten bzgl. Netzwerkkommunikation
 - Wiederverwendung
 - Reduzieren eigener Netzwerkcodepflege

- Gefahren
 - Gefahr des Verlustes vorheriger Verbesserungen des Netzwerkcodes
 - Verbesserungen von Smack müssen erhalten bleiben
 - oder wenn möglich in ECF integrieren
 - nicht alles nach ECF auslagerbar → eigener Provider

ECF Integration

- ECF Integration, wie?
 - Netzwerk-Code von Saros auf ECF Gegenstücke prüfen
 - de.fu_berlin.inf.dpp.net.*
 - de.fu_berlin.inf.dpp.communication.muc.*
 - de.fu_berlin.inf.dpp.ui.chat.*
 - de.fu_berlin.inf.dpp.ui.RosterView
 - Schwierigkeit:
 - Saros arbeitet teils sehr netzwerknah, ECF ist auf eine höhere Ebene ausgelegt

ECF Integration (2)

1. XMPPConnection (Smack) auf XMPPContainer (ECF) umstellen

- XMPPContainer bietet Zugriff auf XMPPConnection

2. bisherigen Netzwerkzugriff refaktorisieren

- Netzwerkzugriffe nur aus Netzwerkschicht (nicht zB aus GUI heraus)
- bisherige Netzwerkzugriffe über diesen Container führen

ECF Integration (3)

3. sukzessiv Netzwerkbereiche auf ECF „portieren“

- Roster (Management, GUI)
- Nachrichtenversand / -handling
- Presence
- Chat / Chatraum
- Dateiübertragung (?)
- je Portierungsschritt
 - Testcases an neue Technik anpassen
 - Pre- und Post-Tests

ECF Integration (4)

4. Dateiübertragung

- XMPP Provider von ECF
 - bietet nach außen nicht die Komplexität und Nähe zur Netzwerkschicht (zu abstrahiert)
 - kommt mit „normaler“ Smack-Bibliothek
- Möglichkeit:
 - neuer ECF XMPP-Provider
 - Transferimplementierungen in diesem Provider
 - » zeitlich nicht im Rahmen dieser Diplomarbeit
 - Provider bringt verbesserte Smack-Bibliothek mit (H.Staib)

Zeit- / Milestoneplan *(Monat entspricht Milestone)*

- 1. Monat:
 - Einarbeitung in Saros und ECF
 - ECF-Test-Plug-In
 - Durchführungsplanung
- 2. Monat:
 - Refaktorisierung, Netzwerkschicht trennen (Netzwerkcode aus Plugin/GUI raus) (2W)
 - Saros auf ECF+Provider umstellen (2,5 W)
- 3. Monat:
 - erweiterten XMPP-Provider implementieren und notwendige Logik
- 4. Monat:
 - Rosterfunktionalität auf ECF portieren (2W)
 - Messaging und Presence auf ECF portieren (2W)
- 5. Monat:
 - MUC auf ECF portieren (2W)
 - ECF Analyse, Pro/Contra ausarbeiten, mit ECF Test-Plugin (2W)
 - Update-Site um ECF-Abhängigkeit erweitern
- 6. Monat: „schreiben“





Danke !
Fragen ?