

Diplomarbeiten zum Thema „Application Mining“ js/11.11.02

1 Hintergrund

Der Begriff „Application Mining“ taucht seit Bearbeitung des Jahr-2000- sowie des Euro-Problems zunehmend in Literatur und Presse auf. Damals wurden mit entsprechenden Werkzeugen riesige Code-Bestände z.B. nach Jahr-2000-Fehlern durchsucht, die weitergehende Bedeutung dieser Tools aber oftmals nicht erkannt.

Heute wird „Application Mining“ für zwei unterschiedliche jedoch zusammenhängende Szenarien verwendet:

1.1 Steigerung der Entwicklungseffizienz

Hier ergibt sich die Motivation zum Einsatz solcher Werkzeuge aus dem Umstand, dass Legacy-Anwendungen zwar einerseits eine lebenswichtige Säule der Geschäftstätigkeit großer Firmen darstellen, andererseits sind sie in der Regel sehr alt und dabei meistens schlecht dokumentiert. Die heute tätigen Entwickler können sich daher die Zusammenhänge der Software nur mit erheblichem Aufwand erarbeiten. Die Wartung wird damit immer kostspieliger bei gleichzeitig steigendem Risiko.

Application Mining Werkzeuge ermöglichen die Analyse der Anwendungen und sind in der Lage, auf dieser Basis Anwendungsstrukturen und Komponentenbeziehungen transparent zu machen, Abläufe in der Software zu simulieren, Auswirkungen von Änderungen aufzuzeigen sowie eine aktuelle und konsistente Dokumentation des Codes zu gewährleisten.

Mittlerweile gibt es solche Werkzeuge nicht nur für das Mainframe-Umfeld, sondern auch für die Client-Server- und Web-Entwicklung, da hier ähnliches Potential zur Effizienzsteigerung erkannt wurde.

1.2 Komponentenbildung aus vorhandenen Geschäftsregeln

Moderne Application Mining Werkzeuge gehen über die bisher genannten Funktionen einen deutlichen Schritt hinaus und unterstützen die Entwickler bei der Bildung von Software-Komponenten aus den vorhandenen Geschäftsprozessen.

Die Notwendigkeit einer solchen Komponentenbildung ist vor allem in Zusammenhang mit der Integration von Legacy-Anwendungen in moderne E-Business-Umgebungen erkannt worden. Denn eine solche Integration ist in aller Regel mit folgenden Schwierigkeiten verbunden:

- Bei der Entwicklung von Legacy-Systemen wurde deren Integrationsfähigkeit in andere Technologien nicht berücksichtigt und ist daher so ohne weiteres gar nicht realisierbar.
- Eine Neuentwicklung auf Basis aktueller Technologie ist mit erheblichem Aufwand und entsprechend großem Risiko verbunden. Die dabei entstehenden Kosten stehen in keinem wirtschaftlichen Verhältnis zum Nutzen.
- Die Anpassung an eine bestimmte Technologie impliziert zumeist, dass ein erneuter Technologiewechsel auch eine erneute Anpassung erfordert. Ein solcher Wechsel findet oft bereits während einer noch laufenden Änderungsphase statt.

Einen Weg aus diesem Dilemma legt die Erkenntnis nahe, dass lediglich ca. 10% des vorhandenen Codes die eigentliche Geschäftslogik in Form von Business Rules beschreiben. Der restliche Code besteht aus Technologie, enthält also z.B. Anweisungen zur Darstellung von Information, zum Zugriff auf Datenbanken etc.

Gelänge es nun, die Geschäftslogik im bestehenden Code zu finden, zu extrahieren, zu kapseln und mit Wrappern an die jeweilige Technologie anzubinden, wäre hinsichtlich der eigentlichen Logik ein hohes Maß an Technologie-Unabhängigkeit erreicht. Der restliche Code muss ohnehin den neuen Technologien angepasst bzw. neu entwickelt werden. Dies ist in der Regel mit modernen Werkzeugen vergleichsweise einfach möglich.

Application Mining Werkzeuge erheben den Anspruch, diesen Vorgang des Auffindens, Extrahierens, Nachdokumentierens und Kapselns der Business Rules in weiten Teilen zu automatisieren. Die Einbindung in Wrapper-Objekte (z.B. für Corba, COM, EJB) erfolgt praktisch auf Knopfdruck.

2 Aufgabenstellung Diplomarbeit

Während erstgenanntes Szenario des Application Mining öfters anzutreffen, der Umgang mit den Werkzeugen und Methoden hinreichend bekannt ist, trifft dies auf den Vorgang der Extraktion von Geschäftsregeln und deren Kapselung als Komponenten nicht zu. Laut Aussage des NOVIBEL-Partners Micro Focus, der weltweit zu den führenden Herstellern solcher Werkzeuge zählt, wurde in Deutschland noch kein Referenzprojekt durchgeführt (in USA und Großbritannien wurden schon Projekte realisiert).

Das Anliegen an die Diplomarbeit ist daher, ein solches Referenzprojekt (in adäquater Größe) durchzuführen unter Einbeziehung vorhandener Erkenntnisse aus Literatur und Wissenschaft. Aus einer bestehenden Cobol-Anwendung sind dabei die vorhandenen Geschäftsregeln zu isolieren, zu dokumentieren und in Komponenten zu überführen.

In diesem Rahmen sind Grundlagen hinsichtlich folgender Aspekte und Fragestellungen zu erarbeiten:

- Vorgehen, Methodik bei Suche, Definition, Extraktion, Dokumentation, Kapselung und Wiederverwendung von Geschäftsregeln
- Auswirkungen verschiedener Parametereinstellungen auf Identifikation von Regeln, Klonen und Redundanzen etc.
- Welche manuellen Arbeitsschritte fallen an bis zu einer vollständigen Regeldefinition? In welcher Weise sind sie durchzuführen? Wer ist daran beteiligt? Wie groß ist der Klärungsbedarf mit Fachexperten? (möglichst eine Systematisierung erarbeiten)
- Regeln oder Empfehlungen zur Komponentenbildung (Grenzen, Größe, Schnittstellen etc.)
- Sofern im Rahmen der Arbeit möglich, prototypischer Einsatz der Komponenten in einer J2EE- oder .NET-Umgebung durch Generierung entsprechender Wrapper und Entwicklung von Oberfläche und Datenzugriff.
- Auch wenn hier keine Projektvergleiche durchgeführt werden, sollte es möglich sein Schlussfolgerungen zu ziehen, unter welchen Randbedingungen Application Mining sinnvoll ist und wann nicht.

3 Administration

Grobe Abschätzung für zeitlichen Ablauf:

- 2 Monate für Einarbeitung und Literaturrecherche
- 2 Monate für Analyse
- 2 Monate für Schreiben der Diplomarbeit.

Zur Durchführung der Arbeit ist eine entsprechende Anwendung bereit zu stellen. Wir gehen derzeit davon aus, dass ein NOVIBEL-Kunde (Versicherung) eine Komponente zur Verfügung stellt. Notfalls ist zu prüfen, ob es für die Uni Leipzig Möglichkeiten gibt, entsprechende Software beizusteuern.

Der Kunde muss darüber hinaus einen Ansprechpartner benennen, der mit den IT- und Geschäftsprozessen vertraut ist. Für diesen wird während der Analysephase ein Aufwand von ca. 20 – 50 Prozent der Arbeitszeit entstehen, je nach Qualität der vorhandenen Codedokumentation.

Die erforderlichen Software-Werkzeuge werden von unserem Partner Micro Focus gestellt.

Ort der Durchführung:

- zum Teil bei NOVIBEL (z.B. wichtige Abstimmungen, Konzeption, ggf. Rücksprache mit Fachexperten des Kunden)
- Realisierung oder Formulierung der Diplomarbeit kann bei NOVIBEL und in Leipzig erfolgen (Rückfragen und Absprachen müssen einfach und zuverlässig abgewickelt werden können).

Noch zu klären:

- Abstimmung hinsichtlich Verteilung Betreuung zwischen Uni und NOVIBEL.
- Arbeit sollte wegen Verwendung von Kundensoftware und Einsatz moderner Technologie vertraulich bleiben. Wissenschaftliche Artikel setzen wahrscheinlich Zustimmung des Endkunden voraus.
Anzustreben ist, dass der Kunde außer einer Vertraulichkeitserklärung (falls gewünscht), den generierten Komponenten sowie einem Exemplar der Diplomarbeit keine Rechte bekommt.
- Zeitraum der Arbeit