

Vom Wandel profitieren – der Entwicklungsauftrag in der Praxis

Im folgenden soll der strukturierte Entwicklungsprozeß beim Durchlauf eines Entwicklungsauftrags (EA) in **devcontrol** demonstriert werden:

Vorausgegangen ist die Analyse über die von einer Entwicklungsleistung betroffenen Objekte der Applikation. Es wird vorausgesetzt, dass für den Entwickler ein eignes Datenbank Schema der Applikation zur Verfügung steht. In **devcontrol** besteht für einen Entwickler immer ein zu bearbeitender Entwicklungsauftrag.

Check-Out

Die für die Entwicklung relevanten Objekte können in **devcontrol** für einen EA reserviert werden. Wird für das entsprechende Objekt der Source Code im Source Control System versioniert, wird die aktuelle Version ins Filesystem ausgecheckt. Beide Vorgänge sind nicht zwingend, das Metadaten Objekt muss nicht der Versionskontrolle unterworfen sein oder kann auch in einem anderen EA reserviert sein.

Entwicklung

devcontrol ist kein Werkzeug zum Editieren von Source Codes. Für die Datenbank stehen Werkzeuge von Oracle und anderen Anbietern zur Verfügung. Das gilt um so mehr für Hersteller gebundene Source Codes wie Oracle Forms, Borland Delphi, Oracle JDeveloper oder Microsoft Visual Studio. Eine Ausnahme gilt für Datenbank basierende Objekte: Diese können auch in **devcontrol** verändert werden und mit dem Objekt-Generator die entsprechende Skripts erzeugt werden. Änderungen des Entwicklers im Datenbank Schema werden von **devcontrol** automatisch protokolliert und einem EA zugeordnet.



Vom Wandel profitieren – der Entwicklungsauftrag in der Praxis

Analyse

Wenn die Entwicklungsleistungen in einem EA beendet sind, werden die Filesystem basierenden Source Codes dem EA zugeordnet und mit dem jeweiligen Objekt-Parser die Metadaten abgeglichen. Für Datenbank basierende Objekte werden auf Grund der protokollierten Schema Änderungen die Metadaten mit dem Objekt-Parser abgeglichen. Bis zu diesem Zeitpunkt muß keine Reservierung von Objekten in devcontrol erfolgt sein. Alle Änderungen können nur im Schema des Entwicklers oder im Filesystem vorhanden sein. Nach dem Metadaten Abgleich können noch Änderungen gelöscht werden, weil sie nur zufällig im Schema entstanden sind, aber keinen Bezug zur Entwicklungsleistung haben.

ORACLE PARTNER

Check-In

Mit Abschluß eines Entwicklungsauftrages werden die Änderungen für Datenbank Objekte und Source Code Objekte in das Metadaten Repository übernommen. Die Source Codes werden im Source Control System versioniert. Vorgeschaltete Prüfroutinen können den Abschluß eines EAs verhindern. Das kann manuell zu bearbeitende Eigenschaften eines Objekts (Beschreibung einer Spalte) oder auch manuell zu pflegende Relationen (Zuordnung einer Tabelle zu einer Datenbank-Rolle) betreffen.

Historie

Beim Check-In eines Entwicklungsauftrags werden die Änderungen an den Metadaten des Application Repositories protokolliert. So kann der Zustand der Applikation vor oder nach einem beliebigen EA wiederhergestellt werden.

Dokumentation

Die Auswertung über die Veränderungen des Metadaten Repositories durch einen Entwicklungsauftrag ergibt eine Übersicht der Entwicklungsleistungen.

mining

Vom Wandel profitieren – der Entwicklungsauftrag in der Praxis

Versioning

Entwicklungsaufträge bilden Entwicklungsleistungen ab. Häufig ist ein EA nicht ausreichend, um die gesamten Anforderungen abzubilden. Durch Tests können sich notwendige Änderungen und Erweiterungen ergeben, die in weiteren EAs abgebildet werden. Um EAs zusammenzufassen, können Versionen gebildet werden.

Rollout

Der Abgleich des Metadaten Repositorys mit der Applikation durch die jeweiligen Objekt-Generatoren erzeugt Skripte für den Rollout eines Entwicklungsauftrages. Die Skripte dienen der Änderung der produktiven Applikation, die auch im Batch erfolgen können.



Nach dem Check-In stehen alle im Metadaten Repository verwalteten Objekte wieder für weitere EAs zur Verfügung. Es besteht keine Koppelung zwischen dem Check-In eines Entwicklungsauftrags in einer Entwicklungsumgebung und dem Rollout des Entwicklungsauftrages in eine produktive Umgebung.

mining