



LaunchEclipse

Platz schaffen auf dem Windows-Desktop

Wenn man sehr viele umfangreiche Java-Projekte entwickelt, legt man sich dafür gerne eigene Eclipse-Workspaces an, um innerhalb von Eclipse nicht den Überblick zu verlieren. Eine Verknüpfung auf dem Windows-Desktop startet anschließend Eclipse mit dem entsprechenden Workspace. Das hier vorgestellte LaunchEclipse [1] kann einem die einzelnen Verknüpfungen auf dem Windows-Desktop ersparen und erleichtert den Start von Eclipse mit unterschiedlichen Start-Konfigurationen. Auch wenn es sich bei LaunchEclipse um kein Eclipse-Plug-in im eigentlichen Sinne handelt, so ist es doch ein interessantes Tool, welches in unserer Plug-in-Parade erwähnt werden sollte.

Das Problem

Die Eclipse-Plattform lässt sich sehr stark konfigurieren. Dies geschieht entweder über Kommandozeilenparameter oder über Java-Systemeigenschaften. In der Regel ist ein Kommandozeilenparameter einfach eine Abkürzung, um nicht die entsprechende Systemeigenschaft setzen zu müssen. Es gibt wesentlich mehr Systemeigenschaften als Kommandozeilenparameter, um Eclipse an die eigenen Bedürfnisse anzupassen. Die

wichtigsten Parameter sind in Tabelle 1 aufgeführt.

Die sicherlich am häufigsten benutzten Parameter sind die zum Setzen eines anderen Workspace und die zum Konfigurieren der Java Virtual Machine. Wenn man viele unterschiedliche Konfigurationen und Workspaces nutzt, kann es sehr leicht passieren, dass man auf diese Art und Weise entsprechend viele Verknüpfungen auf dem Desktop hat, um Eclipse zu starten.

Die Lösung

Genau diesem Problem hat sich LaunchEclipse angenommen. Es bietet auf einfache Art und Weise die Möglichkeit, verschiedene Konfigurationen mit unterschiedlichen Kommandozeilenparametern anzulegen und zu starten. Die Installation ist denkbar einfach, denn man muss lediglich die mitgelieferte *.exe*-Datei in einen beliebigen Ordner kopieren. Anschließend wird das Programm mit einem Doppelklick (oder einer Verknüpfung im Autostart-Ordner) gestartet und nistet sich dann in der Windows-Taskbar ein (Abb. 1).

Ein Rechtsklick auf das Icon genügt, um die Liste aller angelegten Konfigura-

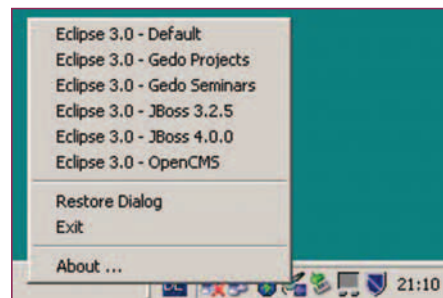


Abb. 1: Das Schnellstart-Menü von LaunchEclipse (hier mit sechs angelegten Konfigurationen)

tionen angezeigt zu bekommen. Nach Auswahl der gewünschten Konfiguration genügt ein Klick, um Eclipse mit den hinterlegten Parametern zu starten. Um Konfigurationen anzulegen oder zu ändern, muss man den Dialog-Modus (Abb. 2) aufrufen. Dies geschieht entweder über einen Doppelklick oder über das Kontextmenü, welches man über den Rechtsklick angezeigt bekommt.

Innerhalb der Übersichtsliste lässt sich mit einem Rechtsklick in die freie Fläche eine neue Konfiguration erzeugen. Dazu öffnet sich der BEARBEITEN-Dialog, über den die einzelnen Parameter der anzulegenden Konfiguration eingegeben werden können. Man kann der Konfiguration einen Namen geben, der dann in der Liste angezeigt wird, und weitere Eclipse-Kommandozeilenparameter eintragen. Der Pfad zu der Eclipse-*.exe*-Datei lässt sich über einen DATEIAUSWAHL-Dialog einstellen

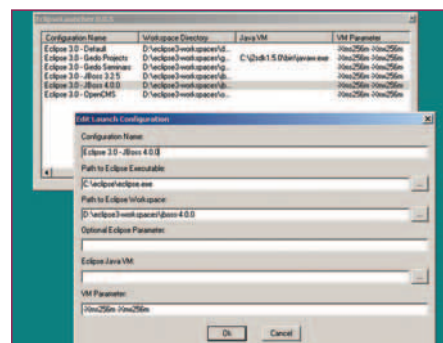


Abb. 2: Die Übersichtsliste der eingegebenen Konfigurationen und der BEARBEITEN-Dialog

Parameter	Beschreibung
-application <i>applicationId</i>	Gibt die Anwendung an, die gestartet werden soll. Wenn nichts angegeben ist, wird die Eclipse-Workbench gestartet.
-data <i>workspacePath</i>	Der Pfad zu dem Workspace, auf dem die Eclipse-Plattform gestartet wird. Relative Pfade sind relativ zu dem Verzeichnis, aus dem Eclipse gestartet wird.
-vm <i>vmPath</i>	Der Platz, an dem sich die Java-Runtime-Umgebung (JRE) befindet, die zum Starten von Eclipse benutzt werden soll. Wenn keine JRE angegeben wird, benutzt Eclipse das erste, sich im Pfad befindliche Java-Executable.
-vmargs <i>args</i>	Dieser Parameter dient zur Konfiguration der benutzten JVM. Alle Argumente werden an die JVM weitergegeben. Er muss an letzter Stelle der Kommandozeile stehen.
-consolelog	Gibt alle Ausgaben des Eclipse-Error-Logfiles zusätzlich auf der Konsole aus, über die Eclipse gestartet wurde. Sehr hilfreich im Zusammenspiel mit dem <i>-debug</i> -Parameter.
-debug [<i>optionsFile</i>]	Startet die Eclipse-Plattform im Debug-Modus. Die Debug-Optionen werden aus der angegebenen Datei geladen.

Tabelle 1: Die wichtigsten Eclipse-Kommandozeilenparameter

ebenso wie der Pfad zu dem zu verwendenden Eclipse-Workspace. Zusätzliche Parameter können in einem Textfeld eingegeben werden. Außerdem lässt sich die Java VM einstellen, die zum Starten von Eclipse benutzt werden soll und auch ihre Parameter (z.B. `-Xms256m -Xmx256m` zum Setzen des Speichers) werden in einem Textfeld eingegeben.

Über die Liste der Konfigurationen lassen sich auch vorhandene Einträge editieren oder löschen. Außerdem besteht die Möglichkeit, einen Eintrag zu duplizieren, wenn man einen neuen auf der Basis eines alten Eintrages anlegen möchte. Alle eingegebenen Informationen legt LaunchEclipse in der Windows Registry unter dem Schlüssel EclipseLauncher ab. Wenn das

Programm gelöscht werden soll, dann sollte man neben dem Entfernen der `.exe`-Datei auch daran denken, die entsprechenden Registry-Einträge zu löschen.

Marcus Redeker

■ Links & Literatur

[1] launchclipse.sourceforge.net



SAP Web Dynpro

Toolset für die Generierung webbasierter Anwendungen

SAP unterstützt mit der Integrations- und Anwendungsplattform Netweaver und deren technologischer Basis, dem Web Application Server, seit einiger Zeit auch Java und J2EE. Für das Design von webbasierten Anwendungen wird mit Web Dynpro

für Java ein eigenes Programmiermodell angeboten, dessen Entwicklungsumgebung das auf Eclipse basierende SAP Netweaver Developer Studio ist. Die Features von Web Dynpro und den Web Dynpro-Toolsets werden hier kurz vorgestellt.

Web Dynpro

Web Dynpro ist begründet auf dem Model View Controller (MVC) Pattern. Dies ist mit Sicherheit nichts Besonderes, auch andere Programmiermodelle für die Erstellung von User Interfaces verwenden MVC

Anzeige