

Geschrieben von:

Freitag, den 11. Januar 2008 um 20:38 Uhr



Mit dem neuen Release des Open-Source Application Frameworks Spring.Net können Entwickler nun auch .NET Anwendungen mit Konzepten bauen, die bisher nur unter Java zur Verfügung standen.

"Spring soll es einfach machen zwischen Implementierungen hin- und herzuspringen", sagte der Spring.Net Projektleiter Mark Pollack in einem Interview. "Wir sind nicht an bestimmte Sprachen gebunden, aber an Programmierideen. Spring bringt Ideen, die in Java entwickelt wurden nun nach .NET."

SpringSource, einst bekannt unter dem Namen Interface21, veröffentlichte zuvor das Spring Framework Programmiermodell für Java.

Die zwei wesentlichen neuen Merkmale des Spring Framework für .NET sind [Dependency Injection](#) und ein aspektorientiertes Programmier-Framework, das das objektorientierte (OOP) Modell, welches in .NET verwendet wird ergänzt.

Dependency Injection ist ein Entwurfsmuster und dient in einem objektorientierten System dazu, die Abhängigkeiten zwischen Komponenten oder Objekten zu minimieren. In der traditionellen OOP, wo ein Objekt A nach einem Objekt B fragt, ist die Relation und Abhängigkeit zwischen diesen beiden Objekten fest kodiert. Falls das Objekt B entfernt wird, muss der gesamte Code modifiziert werden.

Dependency Injection, auf der anderen Seite, hängt nicht davon ab ob Objekt A nach Objekt B fragt.

"Normalerweise sind die Dinge fest programmiert, was Tests erschwert, während dieser Ansatz, bei dem einem Objekt nur das gegeben wird was es braucht, statt das nach was es fragt, die Austauschbarkeit erhöht", sagte Pollack.

SpringSource's Update für .NET implementiert Java Schlüsseltechnologien

Geschrieben von:

Freitag, den 11. Januar 2008 um 20:38 Uhr

Die AOP ist ein Programmierkonzept, welches das Problem der so genannten Cross-Cutting Concern behandelt. Das aspektorientierte Programmier-Framework erlaubt es Funktionen abzuarbeiten, wenn eine Methode aufgerufen wird. Falls die Methode aufgerufen wird, wird eine fest definiertes Verhalten oder eine Aktion initialisiert.

Auf diese Weise wird ein zweiter Schritt oder ein zweiter Prozess aufgerufen, wenn ein Programm eine Aufgabe ausführt, und kann dieses dann kontrollieren, unterstützen oder mit dem ersten Prozess zusammenarbeiten.

Das erlaubt es Features, wie die Ausnahmebehandlung oder das Caching effektiver zu implementieren, statt wie bisher, wo Prozesse gestartet und gestoppt werden und redundanter Code Operationen mehrfach ausführen muss, sagte Pollack. Die aspektorientierte Programmierung erlaubt es nun, auch Aufgaben wie das Tracing zu modularisieren.

Java besitzt dieses Feature in Enterprise JavaBeans. Nun macht es SpringSource's Spring.Net auch für .NET Programmierer verfügbar.

Andere Schlüsseltechnologien aus Spring.Net 1.1 sind die Integration von ASP.NET AJAX (Asynchronous JavaScript und XML) und eine portable Service Abstraktion, um reine .NET Objekte mit .NET Remoting exportieren zu können.

Die Version beinhaltet auch eine "Aspect Bibliothek" mit vordefinierten, leicht verwendbaren Aspekten für die Transaktionsverwaltung, Protokollierung, Leistungsüberwachung, Caching und Ausnahmebehandlung. Zusätzlich stellt die Version eine deklarative Transaktionsverwaltung über XML Konfiguration bereit, sowie das ADO.NET Data Access Framework, welches die Nutzung von ADO.NET vereinfacht.

Das aktuelle Release kann von der offiziellen Seite