



## **Diplomarbeit: Design und Implementierung einer Deployment-, Test- und Monitoring-Umgebung für verteilte Anwendungen**

Die Entwicklung von verteilten Anwendungen und Middleware Systemen, die solche Anwendungen unterstützen ist ein mühsamer Prozess. Um ein verteiltes Programm starten zu können, ist es, im Gegensatz zu Programmen, die auf nur einer Maschine laufen, meistens nötig den Code auf mehrer Systeme zu verteilen um anschließend auf jedem System ein Programm zu starten. Dieser *Bootstrap*-Vorgang muss im Allgemeinen in einer bestimmten Reihenfolge erfolgen und macht es nicht selten auch noch nötig zusätzliche Dienste, wie zum Beispiel einen Namensdienst, zu instanziiieren. Besonders wenn es um das Testen und Debuggen verteilter Anwendungen geht, entfällt auf das *Deployment*, d.h. die Verteilung des Codes und die Instaziierung der nötigen Prozesse, ein nicht unerheblicher Teil des gesamten Zeit.

Im Zuge der Diplomarbeit soll deshalb eine Deployment-Plattform entstehen, die es erlaubt Java-Code automatisch auf verschiedene Maschinen zu verteilen und dort auch auszuführen. Wichtig dabei ist vor allem eine kontrollierte Ausführung. Darunter fällt zum Einen bedeutet, dass die Entscheidung was auf welchem Rechner zur Ausführung kommt, der Kontrolle des Benutzers unterliegt, falls es von diesem gewünscht wird und zum Anderen, dass dem Benutzer Informationen über den Zustand eines gestarteten Prozesses, aber auch dessen Ausgaben an zentraler Stelle zugänglich gemacht werden.

Während der Arbeit sollten zunächst die Konzpte vorhandener Deployment-Systeme wie *ICEGrid* oder *Weevil* evaluiert werden. Darauf aufbauend soll ein eingenes Konzept erstellt werden, wie ein flexibles und konfigurierbares, aber leichtgewichtiges Deployment erreicht werden kann. Dieses Konzept kann einige der folgenden Punkte enthalten: die Integration von Sicherheitsmechanismen, eine Dezentralisierung der Infrastruktur, die Entwicklung einer graphischen Benutzeroberfläche (GUI), die Unterstützung von Benutzereingaben, sowie die dynamische Auswertung von Parametern zur Laufzeit. Zu letztgenanntem zählen zum Beispiel die Anzahl der Aufrufe, die an einen Dienst abgesetzt wurden,