

# Monitoring von WebLogic 12C (12.2) Domains mit RESTful Interface.

Pascal Brand

dbi services

Schweiz

## Schlüsselworte

Middleware, WebLogic, Monitoring, RESTful, Metriken

## Einleitung

Dieser Vortrag erläutert die Möglichkeiten WebLogic Server mit dem RESTful Interface zu überwachen. Dazu werden wir das Konzept und die Verwendung des RESTful Interfaces mit WLS vorstellen. Wir werden erklären welche Metriken für die Monitoring „Tools“ verfügbar sind und wie man sie nachverfolgen kann.

## Konzept

Hier werden wir das Konzept von RESTful im Allgemeinen vorstellen und vor allem wie es mit WebLogic zusammenarbeitet.

## Generell

REST – „Representational state transfer“ - ist ein Architekturstil der für die Web-Entwicklung verwendet wird. Systeme und Websites, die diesen Entwicklungsstil einsetzen, sind auf hohe Leistung, Zuverlässigkeit und Skalierbarkeit ausgerichtet.

Es ist eine auf Webstandards basierende Architektur die ein HTTP-Protokoll für die Datenkommunikation verwendet. Für REST ist jede Komponente eine Ressource auf die man durch eine Standardschnittstelle mit HTTP Methoden Zugriff hat.

## RESTful und WebLogic

Oracle WebLogic Server 12.2.1 bietet eine neue REST Management Schnittstelle mit voller Unterstützung für alle WebLogic Ressourcen.

WebLogic generiert dynamisch, inkrementell und nach Bedarf während der Laufzeit REST Ressourcen durch den Einsatz von Bean Bäumen und Bean Informationen. Diese REST Ressourcen bieten eine Alternative für die Konfigurierung, Überwachung, Bereitstellung und Verwaltung von WebLogic Server. Zusätzliche Ressourcen wie WLST Scripts oder JMX Java Applikationen sind nicht erforderlich. RESTful Modules sind auf jeder WebLogic Domain Server installiert.

Diese Ressourcen URLs sind in drei Kategorien aufgeteilt:

- Konfiguration prüfen
- Konfiguration ändern
- Laufzeitdaten abrufen.

Folgende HTTP Methoden des WebLogic Server REST Interfaces ermöglichen die Konfigurationsüberprüfung oder Änderung und die Objektüberwachung:

GET: Daten abrufen für ein Objekt im Baum  
PUT: Ein neues Objekt in dem Baum erstellen oder ein bestehendes Objekt aktualisieren.  
DELETE: Ein Objekt aus dem Baum löschen.

## Monitoring WebLogic

WebLogic REST ist sehr einfach, leicht und schnell. REST-Anfragen erreichen das Ziel sehr schnell, aber die verlangte Aktion kann viel mehr Zeit in Anspruch nehmen. Überwachungsdatenabfragen sind Aktionen die nicht viel Aufwand erfordern, da diese Laufzeitdaten in den WebLogic Server Runtime-Beans schon verfügbar sind.

In diesem Vortragskapitel werden wir erklären welche WebLogic Server Laufzeitdaten uns besonders interessieren und wie wir sie sammeln können.

Obwohl WLST kompakt und leicht zu schreiben ist, muss man mit einem Leistungsaufwand rechnen. Jedes Mal wenn ein WLST Skript ausgeführt wird muss eine Java Virtual Maschine auf der Kundenseite gestartet werden. Diese Situation kann sehr aufwändig werden, wenn diese WLST Operationen in regelmäßigen Abständen durchführt werden, wie bei einem Monitoring Tool. Auf der anderen Seite werden alle WebLogic Laufzeitdaten mit REST über eine einfache URL von einem Webbrowser, einem UNIX Kommando wie „curl“ oder einem Monitoring Tool abgerufen. Man kann zum Beispiel Laufzeitdaten von jedem WebLogic Server mit einer schnellen HTTP-Anfrage statt einem JVM Verfahren für WLST abzurufen.

Wir werden in den folgenden Bereichen die WLST Skripte und die REST-URLs vergleichen.

Allgemeine Serverzustand Metriken

Java Virtual Maschine Metriken

WebLogic Threads Metriken

JDBC (Java Database Connections) Metriken

Application Key Metriken

## Monitoring Tool Implementation

Im diesem Kapitel beschreiben wir die Implementierung von WebLogic REST Interface in zwei Monitoring-Tools, Nagios und DM (dbi Monitoring).

Nagios bietet umfassende Überwachungsansichten und Schwellenwertüberwachungen für Warnmeldungen. Diese beruhen auf Skripten die ausgeführt werden um die erforderlichen Überwachungsmetriken einzuholen. Diese Skripte können leicht durch WebLogic RESTful-Anfragen ersetzt werden.

Das aktuelle Dbi-Monitoring-Tool bietet eine schwellenwertbasierte Überwachung für Warnmeldungen. Es beruht auf Plugins für jedes unterstützte Ziel. Eines dieser Plugins nutzt das RESTful Interface um die WebLogic Server Metriken zu sammeln.

## Demo

Hier zeigen wir wie WebLogic REST konfiguriert und wie ein Monitoring-Tool eingesetzt werden kann um Middleware Targets zu überwachen. Die oben beschriebenen Aspekte werden hier auch vorgeführt.

## Schlussfolgerung

Dieser Vortrag stellt die REST Architektur und das Potential des WebLogic REST Interface vor. Er legt dar wie leicht es für die Überwachung einer WebLogic Domain eingesetzt werden kann und zeigt die Geschwindigkeit dieses Interface auf.

## Kontaktadresse:

dbi services  
Pascal Brand  
Rue de la Jeunesse 2  
CH-2800 Delémont  
Telefon: +41 32 422 96 00  
Fax: +41 32 422 96 15  
E-Mail [pascal.brand@dbi-services.com](mailto:pascal.brand@dbi-services.com)  
Internet: [www.dbi-services.com](http://www.dbi-services.com)