

# Empfehlungen vom BigData Admin an den Oracle DBA

Alexander Hofstetter, Florian Feicht  
Trivadis GmbH  
München

## Schlüsselworte

Big Data, Hadoop, Oracle, DBA, Elastic, ELK

## Einleitung

Bis vor einigen Jahren war Big Data nur als Hype und Schlagwort in der IT bekannt. Mittlerweile hat sich das radikal verändert. Die von mancher befürchteten Ablösung der relationalen Datenbanken ist natürlich nicht eingetreten. Trotzdem gibt es einige Use Cases, welche nicht mehr ohne entsprechende Technologien auskommen. Es gibt aber einen wesentlichen Unterschied bei der Einführung eines Big Data Systems in die jeweiligen Unternehmen im Gegensatz zu klassischen IT-Projekte. Die Initiative liegt bei den Fachbereichen. Diese brauchten eine schnellstmögliche Umsetzung für ihre Anforderungen aus dem Business. Dabei war eine langwierige Umsetzung durch die IT Abteilung eher hinderlich. In letzter Zeit ist aber deutlich zu beobachten das die jeweiligen Systeme wieder zurück in den Betrieb der IT gegeben werden. Je größer und umfangreicher diese Systeme desto wichtiger und deutlicher kommen die klassischen Stärken eines IT Betriebs zum Tragen.

## Unterschiedliche Anforderungen und Aufgaben

Das klassischen Oracle DBA-Geschäft gibt es bereits seit langer Zeit. Die entsprechenden Vorgehensweisen, Methoden und Tools haben deshalb eine lange Historie. Das führt dazu das die Herangehensweise an etwaige Probleme bereits sehr professionell ist. Auf der anderen Seite wird man allerdings auch blind für neue Lösungsansätze. Hinzu kommt das es hier in der Regel nur sehr wenige Produkte eingesetzt werden. Dazu zählt natürlich auf der einen Seite die Oracle Datenbank, auf der anderen Seite gibt aber auch bestimmte Monitoring Tools die immer wieder eingesetzt werden. Als wichtigste KPI's zählen Performance, Verfügbarkeit, Stabilität aber natürlich auch eine langfristige Planbarkeit. Ebenfalls typisch für die Oracle DBA Welt ist ein strikter Fokus auf die Infrastruktur. Sofern nicht als Applikations-DBA tätig ist eine Zusammenarbeit oftmals eine gerne vernachlässigte Tätigkeit.

Ganz im Gegensatz dazu ist das Big Data Umfeld zu sehen. Die Produkte sind erst seit wenigen Jahren im professionellen IT-Betrieb zu finden. Eine lange Erfahrung im Operationsbereich ist deshalb Mangelware. Außerdem wird hier eine Vielzahl von Komponenten eingesetzt, eine Spezialisierung auf bestimmte Hersteller ist nicht möglich. Der wohl größte Unterschied ist aber die pure Anzahl an zu verwaltenden Systemen. Dies gilt sowohl auf Hardwarebasis als auch an Softwarekomponenten. Umgebungen mit 100 oder mehr Knoten sind hier keine Seltenheit.

■ Komponenten



*Abb. 1: Auszug Big Data Komponenten*

Um hier im Administrationsumfeld handlungsfähig bleiben zu können ist der Einsatz neuer Tools zwingend notwendig. Dazu zählt auf der einen Seite die komplette Automatisierung aller anfallenden Aufgabe. Doch auch ein konsequentes Configuration Management ist unerlässlich. Auf der anderen Seite muss ein Schwerpunkt auf das Monitoring gelegt werden. Dabei ergeben sich aus der Erfahrung zwei Themenblöcke. Das ist zum einen die Aggregation aller anfallenden Logfiles. Zum anderen müssen die entsprechenden Performance KPI's entsprechend gesammelt werden. Auf dieser Datenbasis muss eine zentrale automatisierte Auswertung erfolgen. Doch auch im Zusammenarbeitsmodell ergeben sich Änderungen. Eine sehr enge Zusammenarbeit mit Entwicklern ist einer der Erfolgsfaktoren im Big Data Umfeld. Es gibt keine Trennung zwischen Applikations-Admins und Infrastruktur-Admins.

### **Automatisierung**

Bei der Administration von Big Data Cluster ist die Einführung eines Configuration Management unausweichlich. Der Cluster umfasst eine hohe Zahl an Servern, die je nach Auslastung und Kapazität erweitert wird. Dadurch muss ein neuer Knoten schnell und ohne großen Aufwand hinzugefügt werden.

Configuration Management ist die Beschreibung, wie Serversysteme aussehen sollen. In Tools wie zum Beispiel Puppet, definiert man Rollen und wie diese installiert und konfiguriert sind. Jetzt gewährleisten die Puppet Agents, dass alle Server, die einer Rolle zugewiesen sind, den gleichen Stand besitzen.

Durchgeführt werden kann dies durch zwei verschiedene Arten von Tools.

Zum einen Pull basierende Tools, wie zum Beispiel Puppet. Dort wird auf jedem Zielsystem ein Agent installiert, welcher den Zielzustand sich von einem Repository zieht.

Auf der anderen Seite gibt es Push basierende Tools, wie zum Beispiel Ansible.

Diese werden von einem Client aus gestartet. Anschließend wird auf die Konfiguration auf einen oder mehreren Zielsystemen ausgerollt.

Des Weiteren muss in solch großen und agilen Umgebungen Use Cases rund um die Installation und Konfiguration der Software automatisiert werden. Die Erfahrung hat gezeigt, dass man bei neuen Clusterknoten schnell und unproblematisch die Rollen der einzelnen Server tauschen will. Das bedeutet, dass die neue Hardware ggf. mehr geeignet ist diverse Services zu betreiben, als die vorherige Hardware, und diese eigentlich besser für andere Aufgaben innerhalb des Clusters genutzt werden könnte.

### **Monitoring – Logfiles**

Durch die große Anzahl von Servern und Dienste, die auf diesen laufen, ist es sehr schwierig den Überblick zu behalten und im Fehlerfall auf alle zugehörigen Logfiles zuzugreifen.

Bei diesem Problem kann eine Kombination aus Tools helfen. Bei der MapR Hadoop Distribution wird eine Kombination aus fluentd, elasticsearch und Kibana verwendet.

fluentd ist ein OpenSource Datensammler. Geschrieben ist das Tool ins Ruby und Performance relevante Teile sind in C geschrieben. In einer Konfiguration wird festgeschrieben, welche Daten gesammelt werden und wohin die Daten im JSON Format geschickt werden.

In diesem Fall an elasticsearch.

Elasticsearch ist eine REST basierte Suchmaschine auf Java Basis. Es speichert die Suchergebnisse in JSON ab. In diesem Fall bekommt elasticsearch die Daten von fluentd und legt einen Index über diese Daten. Dadurch sind Suchanfragen sehr schnell. Elasticsearch nutzt Apache Lucene für die Indexierung und Suche. Zur Visualisierung der Suchanfragen wird das Tool Kibana verwendet. Kibana wird wie elasticsearch von der Firma elastic betrieben. Kibana bietet verschiedene Funktionen. Es fängt an mit einer einfachen Suche (Discover), über Erstellung von Charts (Visualize) oder Dashboards (Dashboard).

In der von MapR eingesetzten Version sind die Suchabfragen mit einem Zeitfilter belegt und die Ergebnisse werden mit Zeitstempel und dem Inhalt im JSON Format angezeigt. In der Visualize Funktion lassen sich verschiedene Abfragen als Grafik anzeigen (unter anderem: Linien-, Balken- oder Kuchendiagramme)

Die Dashboards sind eine Zusammenstellung von mehreren Grafiken aus dem Visualisierungsbereich. Die Dashboards werden mit einem Zeitraumfilter dargestellt, zum Beispiel die letzten 30 Minuten. Zusätzlich können weitere Filter durch Eingabe oder Klick auf die Diagramme gesetzt werden.

Diese Kombination lässt sich auch gut in der Oracle Datenbanklandschaft verwenden. In der Oracle Datenbank werden seit 11g die Logfiles standardmäßig im Automatic Diagnostic Repository (ADR) gespeichert. Dies ist unter der ORACLE\_BASE/diag zu finden.

Jetzt könnte man für alle Datenbanken und Middleware auf allen Servern die dort gespeicherten Logs und Tracefiles sammeln. Wenn jetzt zum Beispiel ein Fehler in einer Oracle Datenbank auftritt könnte man alle Komponenten auf den gleichen Server / Cluster in dem Zeitraum prüfen, ob andere Datenbanken auch betroffen sind.

Oder ein schönes Kibana Dashboards bauen, die pro Server oder Datenbank die Fehler in den letzten 30 Tagen anzeigen (nach ORA-NR sortiert).

### **Monitoring – Performance**

Im Zusammenhang mit Oracle und Performance gibt es sehr viele Themen. Dazu zählt das sehr ausgereifte Wait Interface. Hier ist eine sehr detaillierte Auswertung möglich. Aber auch die Reporting Möglichkeiten wie Statspack, AWR und ASH sind sehr vielfältig und ausgereift. Allerdings ist eine zentrale Sammlung dieser Daten in nur sehr wenigen Fällen vorgesehen. Ganz im Gegensatz dazu wird im Big Data Umfeld vorgegangen. Schon alleine aufgrund der schier unendlichen Anzahl an Knoten und Clustern ist ein übergreifendes Performance Management das A und O. Dazu stehen verschiedene Tools zur Verfügung die über jeweilige Plugins zu Datensammlern umgebaut werden können. Im folgendem wird ein Beispiel aus der Hadoop Distribution MapR beschrieben. Dabei kommt folgende Toolkette zum Einsatz:

Die Sammlung der Daten erfolgt mit „collectd“. Dabei handelt es sich um ein Open Source Tool zur Sammlung von Systemstatistiken. Die Statistiken können anschließend über eine TCP-Schnittstelle bereitgestellt werden. Diese Sammlung kann beliebig erweitert werden. Anschließend werden die Daten in eine OpenTSDB gespeichert. TSDB steht für „The Scalable Time Series Database“. Hier gibt es zwei Konfigurationsmöglichkeiten. Entweder setzt OpenTSDB direkt auf Hadoop auf. Das heißt es wird keine wirkliche DB Struktur genutzt, sondern die Ablage über das Filesystem geregelt. Die andere, bei MapR auch im Einsatz befindliche, Möglichkeit sieht vor das als Backend eine HBase benutzt wird. Dabei handelt es sich um eine NoSQL Datenbank. Die schlussendliche Visualisierung der Metriken übernimmt Grafana. Hierbei wird als Datenquelle die entsprechende OpenTSDB hinterlegt. Grafana bietet aber auch die Möglichkeit verschiedene andere Systeme zu nutzen. Als Beispiel ist hier die InfluxDB zu nennen.

Wie können wir nun in der Oracle Welt profitieren? Auch hier werden mehr und mehr Umgebungen aufgebaut. Eine zentrale Darstellung der Performance Daten ist auch hier absolut notwendig. Sei es für

Forecast Planung oder Konsolidierungsvorschläge. Hier kann Grafana sehr sinnvoll eingesetzt werden. Es steht eine direkte Schnittstelle auf Oracle DB's zur Verfügung. Die Sammlung der Daten kann über eine Erweiterung von collectd erfolgen. Sinnvoller ist es aber hier nur Grafana als Visualisierungsschicht zu verwenden. Als Datensammler kann zum Beispiel das AWR Warehouse aus dem Enterprise Manager genutzt werden. Eine andere Alternative wäre hier TVD CapMan als zentrale Sammelinstanz.

### **Fazit**

In der Big Data Welt werden stehen andere Herausforderungen als in der klassischen Oracle DBA Welt an. Aufgrund der fehlenden langjährigen Erfahrungen wurden allerdings einige Herausforderungen neu betrachtet und gelöst. Dabei steht zum einen die konsequente Automatisierung im Fokus. Diese muss zukünftig auch in alle Oracle Projekten umgesetzt werden. Zum anderen eine zentrale Sammlung aller Logfiles. Hierbei ist es auch in der klassischen Oracle Welt eine erhebliche Erleichterung einen zentralen Überblick über alle anfallenden Fehler zu haben. Weiterhin ist eine zentrale Sammlung der Performancedaten bei immer größeren Umgebungen zwingend notwendig. Zusammenfassend lassen sich sehr viele Prinzipien und Tools übertragen und schaffen hier Erleichterung.

### **Kontaktadresse:**

Alexander Hofstetter, Florian Feicht  
Trivadis GmbH  
Lehrer-Wirth-Str 4  
D-81829 München

Telefon: +49 (0) 89 99 27 59 3  
Fax: +49 (0) 12-345 6788  
E-Mail: [alexander.hofstetter@trivadis.com](mailto:alexander.hofstetter@trivadis.com), [florian.feicht@trivadis.com](mailto:florian.feicht@trivadis.com)  
Internet: [www.trivadis.com](http://www.trivadis.com)