

Apex Hochverfügbar

**Ernst Leber
MT AG
Ratings**

Schlüsselworte

APEX, Development, Verfügbarkeit, Datenbank

Einleitung

Die Einsatzgebiete von APEX werden wegen der enormen Produktivität immer größer und APEX wird zunehmend auch in sensitiven Bereichen eingesetzt. Die Frage nach der Hochverfügbarkeit von APEX gewinnt damit immer mehr und mehr an Bedeutung.

Bevor wir die Möglichkeiten betrachten, wie APEX hochverfügbar werden kann, müssen die Rahmenbedingungen geklärt werden. Dafür muss der Begriff Hochverfügbarkeit definiert und die Anforderungen geklärt werden.

Hochverfügbarkeit

Das sagt Wikipedia:

Hochverfügbarkeit (englisch high availability, HA) bezeichnet die Fähigkeit eines Systems, trotz Ausfalls einer seiner Komponenten mit einer hohen Wahrscheinlichkeit (oft 99,99 % oder besser) den Betrieb zu gewährleisten. In Abgrenzung zur Fehlertoleranz kann es bei dem Betrieb im Fehlerfall zu einer Unterbrechung kommen.

Das bedeutet doch, dass ein System redundant ausgelegt sein muss, damit es hochverfügbar ist. Als logische Konsequenz muss das redundante System an einem anderen Ort stehen, damit auch bei größeren Schadensereignissen ein reibungsloser Betrieb gewährleistet ist. Hier kommt dann sofort der Begriff des Rechenzentrums ins Spiel.

Bei kleinen und mittelständigen Unternehmen ist das Rechenzentrum schon mal ein dehnbarer Begriff. Aber die Hauptsache ist, dass die redundanten Systeme an unterschiedlichen Orten stehen.

Zurück zur Definition der Hochverfügbarkeit, da ist von einem System die Rede. Im Falle von APEX hängt die Verfügbarkeit nicht nur von APEX in der Datenbank, sondern auch vom Tomcat ab. Je nach dem wo Tomcat installiert ist, haben Sie unter Umständen zwei Server, die betrachtet werden müssen. Das macht die Betrachtung der Verfügbarkeit schwieriger.

Was soll umgeschaltet werden

Wollen Sie, dass bei einem Ausfall der Datenbank, der zugehörige Tomcat Server weiterlaufen soll und das nur die Datenbank umgeschaltet wird? Soll bei einem Ausfall von Tomcat die Datenbank den anderen Tomcat nutzen oder soll gleich alles geschwenkt werden? Aus diesen Fragen wird ersichtlich, dass, egal welche Komponente ausfällt, in jedem Fall eine Umschaltung auf einen Teil oder auf das komplette redundante System erforderlich ist. Je nach Art der Umschaltung kann dieser Vorgang beliebig komplex werden.

In diesem Vortrag wird daher der komplette Strang, Datenbank einschließlich Tomcat, als ein System betrachtet, das in Fehlerfall umgeschaltet wird.

Wenn wir jetzt noch die Forderung nach einer günstigen bis kostenlosen Lösung stellen, kommt eine Enterprise Edition wegen der Lizenzkosten nicht mehr in Frage. Also bleiben Oracle XE und die

Standard Edition 2 SE2 übrig. Wie kann man diese Versionen inklusive Tomcat hochverfügbar machen?

Um es vorweg zu nehmen, es gibt keine Patentlösung oder ein Patentrezept, aber es gibt mehrere Möglichkeiten Apex (hoch)verfügbar zu machen.

Lösungsansätze

Um das Szenario in einem vertretbaren Rahmen zu halten, gehe ich von einer Anwendung aus, die eine Downtime von 15-30 Minuten zulässt. Das Umschalten sollte aber möglichst automatisch erfolgen. Es werden Lösungswege aufgezeigt, wie die Verfügbarkeit mit Oracle-Mitteln, aber ohne Enterprise Features zu nutzen, gesteigert werden kann.

In einem weiteren Kapitel werden die Möglichkeiten und Wege, die ein Linux Cluster im Zusammenhang mit einer Oracle SE2 oder XE Datenbank bietet, diskutiert. Es werden unterschiedliche Ansätze hinsichtlich der Architektur einer solchen Lösung sowie die Vor- und Nachteile vorgestellt. Im Rahmen dieses Manuskriptes werde ich die Lösungswege und die Vor- und Nachteile nur grob skizzieren.

Wenn man bei der Oracle SE2 bleibt, sind grundsätzlich 2 Instanzen erforderlich. Hier könnte als Hochverfügbarkeitslösung eine Pluggable Database genutzt werden, die über einen automatischen Refresh auf der redundanten Seite mit der aktiven Datenbank synchron gehalten wird. Das wäre eine Primary Standby DB Lösung. Im Fehlerfall wird auf der Standby Seite der Refresh ausgeschaltet, die Datenbank geöffnet und der Tomcat dort aktiviert.

Als weitere Lösung käme ein Cluster basierend auf der SE2 in Frage. Beide Nodes laufen im Normalfall parallel. Bei Ausfall eines Nodes übernimmt der verbleibende Node die komplette Last und Tomcat wird über eine VIP umgeschaltet.

Bei Einsatz eines Linux Clusters werden zwei identische Systeme aufgebaut, die aktiv passiv betrieben werden. Auf jedem Server sind Tomcat und die Datenbank installiert. Die Anbindung an die Außenwelt erfolgt über eine Application VIP die im Fehlerfall automatisch umgeschaltet wird. Alle Files der Datenbank, das kann in diesem Fall auch eine XE Datenbank sein, liegen in einem Clusterfilesystem. Die Datenbank auf der passiven Seite ist gestoppt und wird im Fehlerfall aktiviert.

Kontaktadresse:

Ernst Leber
MT AG
Balcke-Dürr-Allee 9
D-40882 Ratingen

Telefon: +49 2102 30961-0
Fax: +49 2102 30961-101

E-Mail: ernst.leber@mt-ag.com
Internet: www.mt-ag.com