

## Airbags für Oracle

### Hochverfügbarkeit von Oracle-Datenbanken unter Windows 2000

Wenn es um relationale Datenbanken geht, steht Oracle mit seinen Produkten bei vielen Unternehmen ganz oben auf der Hitliste der bevorzugten Plattformen. Es müssen jedoch nicht immer High-End-UNIX-Systeme sein, die den hohen Anforderungen der Oracle-Gemeinde entsprechen. Mit Windows 2000 schickt auch Microsoft eine stabile und skalierbare Basis ins Rennen um die Gunst der verwöhnten Oracle-Anwender.

#### DIRK PELZER

Die Hochverfügbarkeit einer Datenbank ist für viele Firmen unternehmenskritisch geworden. Nicht nur im E-Commerce kosten Systemausfälle leicht zehntausende von Mark pro Stunde. Umso wichtiger wird es, in Hochverfügbarkeitslösungen zu investieren, um blamable Serverausfälle zu vermeiden, die einige Unternehmen im vergangenen Jahr nicht nur schlechte Kritiken eingebracht, sondern teilweise gar deren Börsenwert nachhaltig beeinflusst haben. Abhilfe schaffen Cluster-Systeme, die in vielfältiger Art und Weise dafür sorgen, dass der Ausfall eines Servers nicht zum Blackout einer ganzen Anwendung führt. Mit Oracle-Datenbanken auf der einen Seite und dem Windows 2000 auf der anderen treffen zwei Welten aufeinander, die in der Vergangenheit nicht gerade als Traumpaar gehandelt wurden. Denn Ausfallsicherheit und Skalierbarkeit großer Oracle-datenbanken galten als die Domäne proprietärer UNIX-Systeme. Doch Windows 2000 ist erwachsen geworden und so existieren mittlerweile eine ganze Reihe unterschiedlicher Alternativen, um Oracle unter Windows 2000 hochverfügbar und skalierbar auszulegen. Eine Lösung, die den Spagat in beide Richtungen gleichermaßen versucht, ist der Oracle Parallel Server (OPS). Mit diesem sind Unternehmen in der Lage, hochskalierbare Lösungen aufzubauen, die zudem auch ausfallsicher sind. Wenn die Skalierbarkeit eine eher untergeordnete Rolle spielt und stattdessen die Verfügbarkeit der Datenbank im Vordergrund steht, eignen sich Failover-Lösungen, wie sie beispielsweise mit dem Microsoft Cluster Service (MSCS) oder anderen Failover-Produkten realisierbar sind. Eine weitere Alternative stellen Replikationsmechanismen dar, die auf der Transaktionsebene aufsetzen und Datensätze von einem System aus auf eines oder mehrere weitere räumlich verteilte replizieren.

#### Hochverfügbar und skalierbar: Oracle Parallel Server

Skalierbarkeit und Hochverfügbarkeit sind zwei Attribute, die in der Windows-Welt nicht recht zusammen passen wollten. Von UNIX-Anwendern meist müde belächelt wurden deshalb auch die »kleinen Kisten«, auf denen Oracle unter Windows NT früher häufig anzutreffen war. Um das ändern, bietet Oracle bereits seit 1997 seinen Parallel Server für Windows NT und mittlerweile auch für Windows 2000 an. Er setzt die Enterprise Edition von Oracle 8i voraus und hat die Zielsetzung, sowohl die Skalierbarkeit, als auch die Verfügbarkeit der Oracle-Datenbank zu verbessern. Dazu arbeitet er nach dem Shared-Disk-Prinzip. Dies hat zur Folge, dass alle Clusterknoten gleichzeitig Zugriff auf die Datenbankdateien haben. Damit können Lesezugriffe parallelisiert und schneller ausgeführt werden, falls die zugrundeliegende Anwendung den Parallelzugriff unterstützt. Schreibzugriffe hingegen müssen serialisiert werden, um der Inkonsistenz des Datenbestandes vorzubeugen. Der Distributed-Lock-Manager (DLM) übernimmt diese Aufgabe. Das Problem bei dieser Vorgehensweise besteht allerdings darin, dass der DLM bei schreibintensiven Anwendungen und sehr vielen Knoten sehr leicht zum Flaschenhals wird. Die Performance in OPS-Clustern mit zwei Knoten kann mit einer nicht für den OPS-Einsatz optimierten Anwendung sogar schlechter sein, als bei einem einzelnen Datenbank-Server.

Der OPS stellt in Verbindung mit der Oracle 8i Enterprise Edition sehr hohe Anforderungen an die zur Verfügung stehende Hardware. Er skaliert dabei aufgrund seiner Architektur besser, wenn wenige, aber dafür mit mehr Speicher und schnelleren CPUs ausgestattete Maschinen zum Einsatz kommen. So ist die Performance bei einem Zwei-Knoten-OPS-Cluster bestehend aus Vier-Prozessor-Maschinen

erwartungsgemäß besser, als bei einem Vier-Knoten-Cluster mit Zwei-Prozessor-Maschinen. Einen nicht unwesentlichen Anteil daran hat zudem der Cluster-Interconnect, über den die gesamte Kommunikation zwischen den Cluster-Knoten abgewickelt wird und der folglich sehr hohen Ansprüchen genügen muss. Über den Interconnect regeln die beteiligten Clusterknoten den Schreibzugriff auf die Datenbank. Ein Knoten der beispielsweise schreibend auf eine Tabelle zugreifen möchte, meldet dies beim Distributed Lock Manager an und erhält von diesem die Mitteilung, ob die Datenstruktur für den Schreibzugriff freigegeben ist. Sofern das der Fall ist, setzt er selbst einen »Lock«, um exklusiven Zugriff zu erhalten. Nach Beendigung des Schreibzugriffs muss der Knoten den Lock wieder über den DLM freigeben, damit die anderen Knoten wieder Schreibzugriff erhalten können. Für die Koordination der Cluster-Knoten sind in einer Shared-Disk-Architektur auf diese Weise etwa einhundert Mal mehr Nachrichten erforderlich, als dies bei einem Shared-Nothing-Cluster der Fall ist. Ein Cluster-Interconnect basierend auf einer Netzverbindung mit 100 Mbit/s Ethernet kann dafür bereits zu langsam sein. Statt dessen empfiehlt sich an dieser Stelle der Einsatz von Hochgeschwindigkeits-Interconnects wie etwa Compaq ServerNet VI oder Gigabit Ethernet. In naher Zukunft wird auch das kurz vor der kommerziellen Verfügbarkeit stehende Infiniband in diesem Bereich eine tragende Rolle übernehmen.

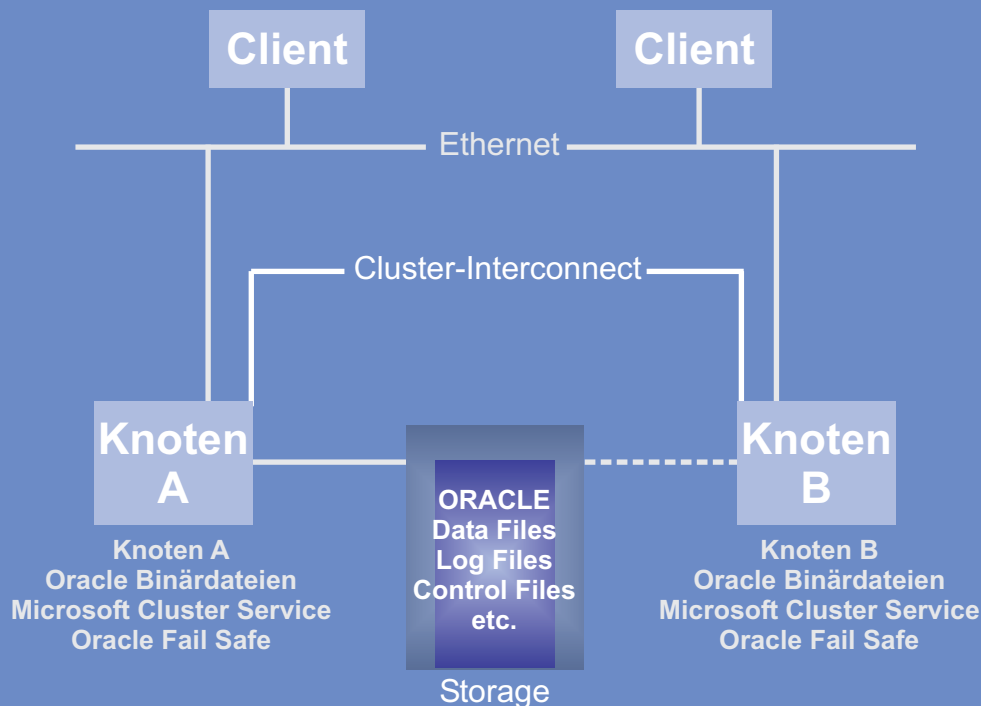
Der Failover bei OPS ist aufgrund der Tatsache, dass alle Clusterknoten auf denselben Datenbestand zugreifen, relativ leicht zu bewerkstelligen. Allerdings spielen Aspekte, wie die Cache-Kohärenz und das Aufräumen von Locks nach einem Failover eine große Rolle für die Failover-Geschwindigkeit. Der DLM muss nämlich insbesondere dafür sorgen, dass alle Locks, die ein ausgefallener Knoten in der Datenbank gesetzt hatte, beseitigt werden, damit die anderen Knoten weiter arbeiten können. Somit ergibt sich bei schreibintensiven Anwendungen, in denen ständig viele Locks gesetzt werden eine verlängerte Failover-Zeit. Aufgrund seiner Architektur empfiehlt sich der Einsatz des OPS nur für sehr spezielle abgegrenzte Anwendungen, die zudem die Parallelverarbeitung unterstützen und weniger für den generellen Einsatz mit Failover- und Skalier-

barkeitsanforderungen.

#### Microsoft Cluster Server und Oracle Fail Safe

Einen anderen Ansatz als Oracle mit dem OPS verfolgt Microsoft mit seinem Cluster Service (MSCS) für Windows 2000. Dieser steht unter Windows 2000 Advanced Server als Zwei-Knoten-Lösung zur Verfügung und mit Windows 2000 Data Center in einer Variante mit vier Knoten. Der MSCS gehört zur Kategorie der sogenannten Shared-Nothing- oder Failover-Cluster, was soviel heißt, dass jeder Cluster-Knoten zu jedem Zeitpunkt exklusiven Zugriff auf die ihm zugewiesenen Ressourcen hat. Das gilt insbesondere für die Festplattenlaufwerke, auf denen sich die Datenbankdateien befinden. Das unterscheidet diesen Cluster-Typ signifikant vom OPS, bei dem ja alle Cluster-Knoten gleichzeitig auf die Plattenlaufwerke zugreifen können. Der Vorteil dieser Lösung besteht darin, dass ein aufwendiger Kommunikationsmechanismus, wie ihn der Distributed Lock Manager darstellt, entfallen kann. Die Kommunikation über den Cluster-Interconnect beschränkt sich demzufolge im Wesentlichen auf den Austausch von Heartbeatsignalen, anhand derer die Cluster-Knoten feststellen, ob noch alle Systeme am Leben sind. Daraus resultierend ergibt sich andererseits jedoch eine eingeschränkte Skalierbarkeit der bereitgestellten Datenbank, denn diese kann zu jedem Zeitpunkt nur von genau einem Knoten ausgeführt werden. Allerdings sind mit Oracle 8i sogenannte Active/Active-Cluster-Konfigurationen möglich, die eine einfache Form der Lastverteilung bieten. In einer solchen Konstellation stellt jeder Knoten des Clusters eine oder mehrere eigene Datenbanken bereit. Im Bedarfsfall ist er aber in der Lage, zusätzlich die Datenbank(en) eines ausgefallenen anderen Knoten zu übernehmen. Die einzelnen Cluster-Knoten müssen allerdings so ausgestattet sein, dass sie im Notfall auch in der Lage sind, die zusätzliche Belastung zu verkraften. Das heißt im Umkehrschluss, dass in einer Active-/Active-Konfiguration beispielsweise die CPU-Auslastung jedes Knoten durchschnittlich 45 Prozent nicht überschreiten darf, um ausreichende Reserven für den Fall des Falles bereit stellen zu können. Gleiches gilt für den Speicherausbau.

## Failover-Cluster mit MSCS und Oracle Fail Safe



Zwei-Knoten-Cluster basierend auf Windows 2000 Advanced Server mit Microsoft Cluster Service und Oracle Fail Safe. Im Normalbetrieb ist jeweils ein Knoten der aktive und beantwortet Client-Anfragen. Im Fehlerfall übernimmt der überlebende Knoten das Datenlaufwerk und stellt die Datenbank wieder zur Verfügung.

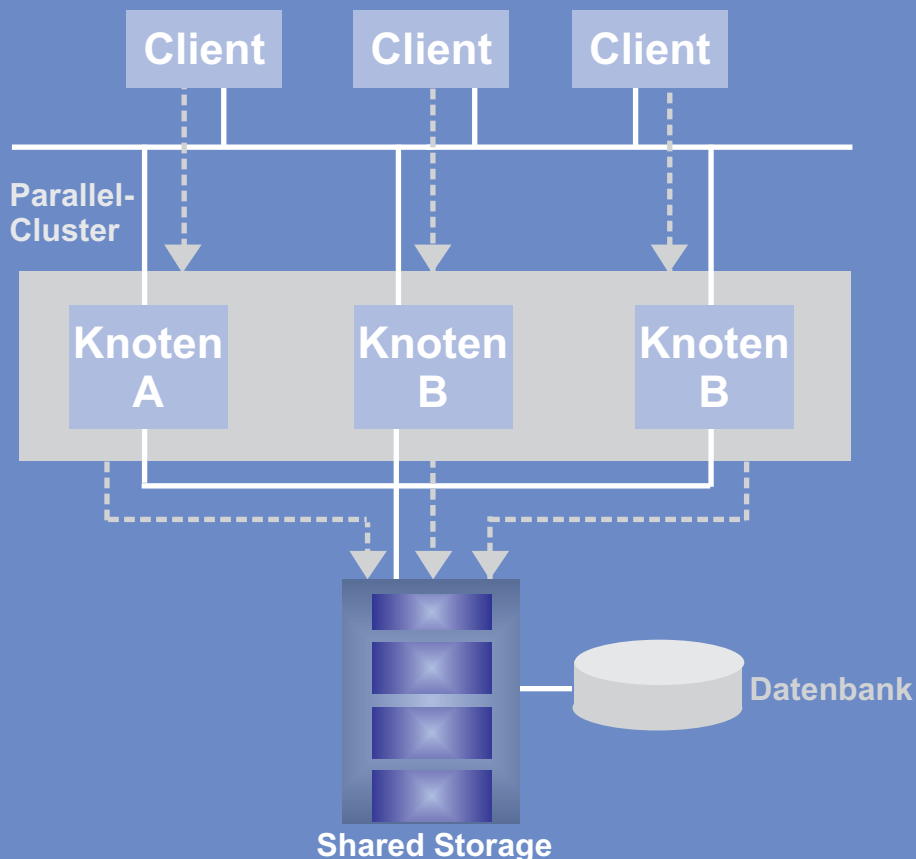
Da der MSCS selbst keine native Unterstützung für Oracle-Datenbanken bietet, hat Oracle eine Lösung namens Oracle Fail Safe (OFS) entwickelt, die auf den MSCS aufbaut und diesen entsprechend ergänzt. So liefert der Datenbank-Spezialist eigens geschriebene DLLs, mit denen sich kurze Failover-Zeiten realisieren lassen. OFS sorgt zudem dafür, dass nicht nur die Datenbank selbst ausfallsicher wird, sondern auch weitere Oracle-Komponenten, wie Forms und Reports. Die Oracle-Entwickler haben zudem eine erweiterte Skriptschnittstelle implementiert, über die der Systemverwalter beispielsweise bestimmte Lastverteilungsszenarien, Offline-Backups oder eine präventive Verifizierung der Cluster-Gruppen vornehmen kann. Da für die Konfiguration einer Oracle-Datenbank innerhalb

eines Failover-Clusters zahlreiche Einstellungen modifiziert und angepasst werden müssen, hat Oracle zudem eine eigene Benutzeroberfläche geschrieben, die zahlreiche Assistenten enthält. Mit deren Hilfe wird der Systemverwalter in die Lage versetzt, für das Clustering notwendige Änderungen nicht fehleranfällig in den Oracle-Konfigurationsdateien durchführen zu müssen, sondern komfortabel per Mausklick. Die Lösung ist für die Standard- und Enterprise-Edition von Oracle 8i für Windows 2000 freigegeben (siehe [technet.oracle.com/tech/nt/failsafe/htdocs/sw\\_comp\\_311.html](http://technet.oracle.com/tech/nt/failsafe/htdocs/sw_comp_311.html)), vergleichsweise einfach zu implementieren und stellt auch geringere Anforderungen an die verwendete Hardware.

### Plattenspeicher als Single-Point-of-Failure

Ein Aspekt der bislang noch nicht betrachtet wurde, ist der Einfluss des Plattenspeichers auf die Verfügbarkeit des Clusters. Häufig werden die Daten eines Clusters auf einem Plattensubsystem abgelegt, das zwar mit allerlei redundanten Netzteilen, RAID-Platten und dergleichen aufzuwarten vermag, aber dennoch einen Single-Point-of-Failure darstellt. Der Grund dafür ist, dass etwa ein Kurzschluss das gesamte Plattensubsystem in Bruchteilen von Sekunden hinwegzuraffen vermag, ohne dass die integrierten redundanten Komponenten etwas dagegen ausrichten könnten. Das gilt im übrigen gleichermaßen für den Oracle Parallel Server, wie auch für den Microsoft Cluster Service. Systemadministratoren, die derartige Ausfälle vermeiden möchten, setzen daher auf Replikationstools, die einfach den Platteninhalt eines Subsystems auf ein anderes spiegelt. Hierfür bieten sich zum einen controllerbasierte Lösungen an oder Softwarelösungen, die zudem den Vorteil haben, preiswerter zu sein. Bei controllerbasierten Lösungen sorgt die Firmware des im Plattensub-

system integrierten RAID-Controllers dafür, dass die Daten beispielsweise per Fibre-Channel auf zwei unterschiedliche Subsysteme geschrieben werden. Fällt eines aus, findet ein für den Anwender transparenter Failover auf das überlebende System statt. Dieser Lösungsansatz setzt in der Regel identische Plattensubsysteme eines Herstellers voraus und ist meist extrem kostspielig. Eine vergleichsweise preiswerte softwarebasierende Lösung bietet die Firma Veritas mit ihrem Volume Manager für Windows 2000 an. Dieser ist unter anderem in der Lage, den Speicherplatz zweier unterschiedlicher Subsysteme per RAID-1-Konfiguration zu einem logischen Volume zusammenzufassen. Die zugrunde liegenden Plattensubsysteme müssen weder vom selben Hersteller stammen noch am selben Standort sein. Es wäre im Extremfall sogar möglich, einen Plattenbereich auf einer Symmetrix von EMC zu verwenden und für den anderen eine oder mehrere IDE-Platten. Damit lassen sich auf relativ einfache Art und Weise Lösungen aufbauen, die katastrophenfalltauglich sind.



### Oracle Parallel Server

Beim Oracle Parallel Server beantworten alle vorhandenen Clusterknoten gleichzeitig eingehende Client-Anfragen und greifen dabei alle auf denselben Datenbestand (Shared-Disk) zu. Ein Distributed Lock Manager sorgt für Datenkonsistenz beim Schreibzugriffen.

### Ausfallsicherheit durch Replikation

Unternehmen, die auf eine Failoverfunktion verzichten können, aber dennoch Ausfallsicherheit benötigen, liebäugeln häufig mit einer reinen Replikationslösung. Die Daten werden hier auf zwei oder mehr Systemen redundant vorgehalten. Die Replikation kann dabei auf zwei Arten erfolgen, nämlich synchron oder asynchron. Bei synchroner Replikation gilt eine Transaktion erst dann als abgeschlossen, wenn sie von allen an der Replikation beteiligten Systemen als abgeschlossen bestätigt wurde. Ist das nicht der Fall, findet ein Rollback der Transaktion statt. Dies zieht eine mehr oder weniger geringe Beeinträchtigungen der Performance nach sich, speziell dann, wenn die an der der Replikation beteiligten Server über eine größere Distanz miteinander verbunden sind. Bei der asynchronen Replikation muss nur das primäre System die Transaktion bestätigen. Alle anderen Systeme werden zu einem späteren Zeitpunkt aktualisiert. Das hat zur Folge, dass bei der asynchronen Replikation unter Umständen Transaktionen auf dem Weg zu den Zielsystemen verloren gehen können und zeitverzögert nachgezogen werden müssen. Während dieses Zeitraumes sind die Datenbestände zwischen den beteiligten Rechnern inkonsistent, was je nach Anwendung zu unerfreulichen Nebenwirkungen führen kann. Wenn höchste Datensicherheit und Konsistenz gewährleistet sein soll, ist somit die synchrone Replikation erforderlich.

In allen Fällen kann, wie bereits ausgeführt, entweder eine firmware- oder treiberbasierte Lösung im Betriebssystem selbst zum Einsatz kommen. Der Vorteil der firmwarebasierten Lösung besteht dabei primär darin, dass sie nicht auf Problematiken wie offene Dateien Rücksicht nehmen muss, da sie auf Blocklevel-Ebene agiert und gewissermaßen ein Duplexing aller Dateioperationen auf zwei getrennten Storage-Systemen durchführt. Zudem können die Daten mit Fibre-Channel- oder SCSI-Bus-Geschwindigkeit auf alle beteiligten Subsysteme transferiert werden, was gegenüber einer auf Ethernet basierten Lösung einen erheblichen Geschwindigkeitsvorteil mit sich bringt. Dafür gestaltet sich bei der firmwarebasierten Lösung eine Rück-synchronisation nach dem Ausfall des primären

Subsystems nicht ganz unkompliziert und muss sehr sorgfältig mit Skripten implementiert und getestet werden.

Eine weitere zuverlässige Variante stellt die Replikation auf Transaktionsebene dar, so wie sie beispielsweise in Form der Oracle Advanced Replication realisiert ist. Diese arbeitet im Gegensatz zu den anderen betrachteten Replikationsmechanismen nicht auf der Datei- oder Blockebene, sondern auf Transaktionsebene. Oracle Advanced Replication unterstützt sowohl synchrone als auch asynchrone Replikation, aber auch eine Hybridform, die Teile der Datenbank synchron und andere asynchron repliziert. Bei der synchronen Replikation müssen die Transaktionen auf allen Systemen, die an der Replikation beteiligt sind, erfolgreich abgeschlossen sein. Andernfalls findet ein Rollback der Transaktion statt. Oracle unterstützt zusätzlich insgesamt vier Modi. Diese sind:

- **Primary Site Ownership:** Die Daten gehören einer Site und können nur in dieser geändert werden.
- **Dynamic Ownership:** Der Besitz über die Daten kann wandern. Es ist aber sichergestellt, dass Updates nur in der Site stattfinden, die den Besitz über die Daten hat.
- **Shared Ownership:** Die Daten können auf allen Sites geändert werden. Dies führt zu temporären Inkonsistenzen und Replikationskonflikten.
- **Fail-Over:** Daten werden von einem System auf ein zweites repliziert. Wenn das erste ausfällt, kann mit den Daten des Spiegelsystems weitergearbeitet werden. Es findet kein IP-Failover statt. Anwendungen müssen gegebenenfalls umkonfiguriert werden.

Die Replikationsvariante eignet sich vor allem für Konfigurationen, bei denen die Daten nur auf einem System modifiziert und dann an weitere Systeme übertragen werden müssen. Ein mögliches Einsatzgebiet wäre beispielsweise ein Bauteilekatalog, der in der Unternehmenszentrale gepflegt und dann per Replikation an Unterstandorte, Vertriebspartner oder Händler repliziert wird. Die Verfügbarkeit einer auf diese Weise ausfallsicher

## Airbags für Oracle

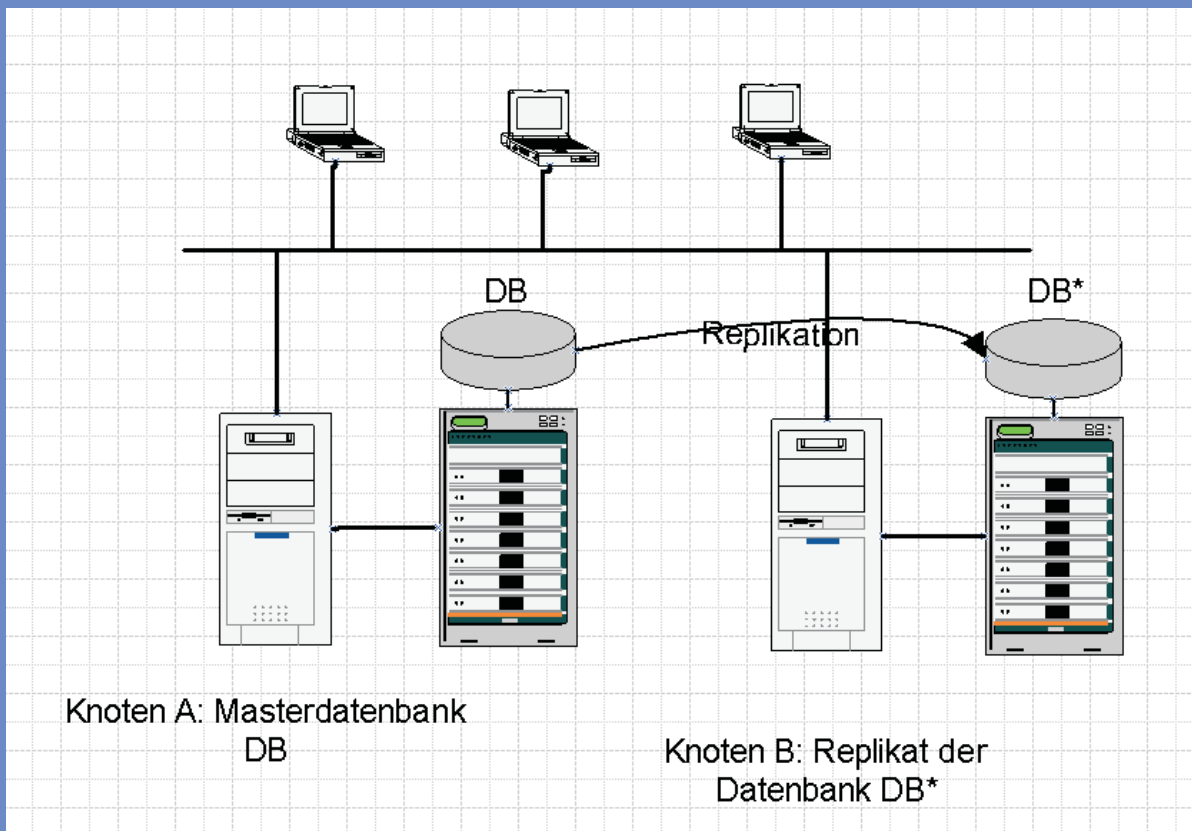
ausgelegten verteilten Datenbank kann natürlich bei Bedarf durch entsprechende Failover-Mechanismen weiter gesteigert werden.

### Fazit

Die Möglichkeiten, Oracle unter Windows 2000 ausfallsicher auszulegen sind ebenso vielfältig wie zahlreich. Auch Mischformen zwischen den einzel-

nen Technologien sind möglich und so dürfte für viele Unternehmen, die auf UNIX verzichten und auf Windows 2000 standardisieren wollen, eine passende Lösung dabei sein, die ihren Anforderungen entspricht. Jedoch gilt auch für Windows 2000, dass die Lösung vorher vernünftig konzipiert und getestet werden muss, bevor sie in den Produktivbetrieb geht. Schnellschüsse führen hingegen regelmäßig zu Frustration und unerwünschten Ergebnissen.

### Ausfallsicherheit durch Replikation



Bei der Replikation werden Teile der Datenbank auf Transaktionsebene von einem Knoten auf einen weiteren übertragen und dort in eine identische Datenbank eingetragen. Die Replikation kann synchron oder asynchron erfolgen.

### Zur Person

DIPL. ING. DIRK PELZER arbeitet als freier Consultant und Journalist in München. Er betreibt ein Storage Labor für verschiedene namhafte Fachzeitschriften. Zudem beschäftigt er sich mit Speichernetzen und Hochverfügbarkeit.

## Airbags für Oracle

### Cluster-Technologien

#### Performance-Cluster

Performance-Cluster sind überwiegend im technisch-wissenschaftlichen Umfeld zu finden. Dort werden sie beispielsweise zur Berechnung von Wetterdaten oder für Simulationen in der Automobilindustrie eingesetzt. Der Hauptfokus von Performance-Clustern liegt in der Skalierbarkeit, bei der durch Hinzufügen zusätzlicher Systeme und deren CPUs die Rechenleistung verbessert werden soll. Die Verfügbarkeit eines Dienstes, wie beispielsweise einer Datenbank spielt hingegen nur eine untergeordnete Rolle.

#### Parallel-Cluster

Ähnlich wie bei Parallel-Clustern besteht die Aufgabe von Performance-Clustern darin, die Skalierbarkeit einer Anwendung zu erhöhen. Durch Hinzufügen zusätzlicher Knoten sollen Antwortzeiten verkürzt, Zugriffe beschleunigt und somit die Gesamtperformance einer Anwendung insgesamt verbessert werden. Bei einem Ausfall eines Clusterknotens übernehmen aber die verbleibenden Systeme die zusätzliche Last und stellen somit die Verfügbarkeit der Anwendung wenn auch mit verschlechterter Performance sicher.

#### Failover-Cluster

Failover-Cluster erfüllen primär die Aufgabe, Dienste, wie etwa einen Web-Server oder eine Datenbank ausfallsicher auszulegen. Der typische Failover-Cluster besteht aus zwei Knoten, wobei einer aktiv ist und eingehende Anfragen bearbeitet, während der zweite den aktiven überwacht und bei einem Ausfall die Funktionen des ersten übernimmt. Failover-Cluster bieten je nach Anwendung auch eine eingeschränkte Skalierbarkeit. Dabei stellen beide Knoten eines Failover-Clusters jeweils voneinander unabhängige Dienste zur Verfügung. Bei einem Serverausfall übernimmt der Überlebende wiederum die Dienste seines Partners und stellt diese wieder zur Verfügung. So wäre beispielsweise eine Konfiguration denkbar, bei der ein Clusterknoten eine Datenbank bereitstellt und der andere einen Web-Server (Active/Active-Betrieb). Beim Ausfall eines Knotens übernimmt der zweite die Dienste des anderen und stellt sowohl die Datenbank, als auch die Datenbank bereit.

## Airbags für Oracle

### Begriffe

#### Cluster-Interconnect

Private Netzwerkverbindung, über die die einzelnen Cluster-Knoten Statusmeldungen austauschen und feststellen, ob die am Verbund beteiligten Knoten noch aktiv sind.

#### Shared-Disk-Cluster

Cluster-System, bei dem alle beteiligten Knoten gleichzeitig Zugriff auf einen gemeinsamen Plattenbereich haben (Shared-Disk). Dieser Plattenbereich enthält die Daten, auf die alle Cluster-Knoten zugreifen, um Client-Anfragen zu bearbeiten. Um die Daten konsistent zu halten, ist eine Distributed Lock Manager notwendig.

#### Shared-Nothing-Cluster

Cluster-System, bei dem jeder Knoten zu jedem Zeitpunkt exklusiven Zugriff auf die Ressourcen hat, die zur Bereitstellung eines Dienstes benötigt werden. Dies gilt insbesondere für Plattenlaufwerke. Diese haben zwar in der Regel eine physikalische Verbindung zu den am Cluster beteiligten Knoten, jedoch wird durch Treiber dafür gesorgt, dass nur jeweils ein Knoten tatsächlich Zugriff auf die Plattenbereiche erhält.

#### Distributed Lock Manager

Da bei Parallel-Clustern alle Knoten eines Systems gleichzeitig auf dieselben Ressourcen und hierbei insbesondere den Plattenspeicher (Shared-Storage) zugreifen dürfen, ist bei Parallel-Clustern eine Instanz erforderlich, die Schreibzugriffe auf die gemeinsamen Ressourcen verwaltet und serialisiert. Nur so lassen sich die Daten gegen etwaige Inkonsistenzen schützen, die entstehen können, wenn derselbe Datensatz von mehr als einem Knoten gleichzeitig verändert wird. Diese als Distributed-Lock-Manager (DLM) bezeichnete Instanz sorgt dafür, dass solche Inkonsistenzen ausgeschlossen werden. Die Notwendigkeit für den Einsatz eines DLM schränkt allerdings die Skalierbarkeit eines Parallel-Clusters ein, da der für die Serialisierung und