

Vorlesung:
Betriebssysteme
Hochverfügbarkeit (Einführung)

4. Quartal 2017

Fallbeispiel

Im Folgenden sollen anhand eines Fallbeispiels einige wichtige Themen erläutert werden.

Wie bereits besprochen, handelt es sich um einen Infrastruktur-Lösungsansatz, der sowohl einzelne Computersysteme wie z.B. PCs oder Server als auch Vernetzung berücksichtigt.

Welche aktuellen Anforderungen bestehen:

- Erreichbarkeit
- Uptime 100%
- Ausfallsicherheit
- 24/7
- Datensicherheit
- Redundanzen
- Automatismen

Fallbeispiel

Wir berücksichtigen dabei zunächst folgende grundlegenden Aspekte:

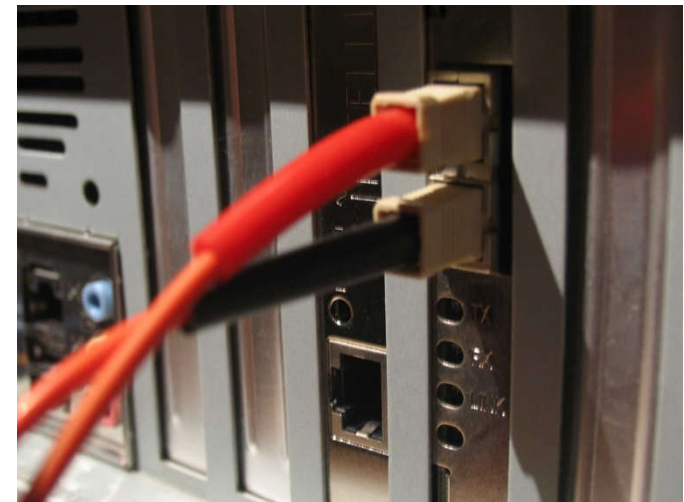
- Storage (DAS, SAN, NAS)
 - Ausführung (CPU, RAM)
 - Netzwerk
 - Dienste
 - Services
 - Cluster
- All dieses soll ausfallsicher gemacht werden.

Hochverfügbarkeit

- ◆ Einleitung und Motivation
- ◆ Begrifflichkeiten
- ◆ HA-Cluster
- ◆ Praktisches Beispiel
- ◆ Ausblick

Hochverfügbarkeit

- ◆ Einleitung und Motivation
- ◆ In diesem Quartal werden wir am Beispiel aktueller Anforderungen an IT zunächst Netzwerkgrundlagen erarbeiten, dann einen Exkurs in die Virtualisierung vornehmen, um auf dieser Basis die Grundzüge von Hochverfügbarkeitslösungen zu erörtern.



Hochverfügbarkeit

- ◆ Einleitung und Motivation
- ◆ Die beste Content-Management-Software und das beste Webdesign sind nutzlos, wenn der Webserver nicht erreichbar ist. Durch die zunehmende Globalisierung sind Unternehmen vermehrt auf Onlineapplikationen angewiesen, die 24 Stunden am Tag verfügbar sind. Dank der hier vorgestellten reinen Software-Lösung ist diese hohe Verfügbarkeit selbst für kleine und mittelständische Unternehmen erschwinglich.

Hochverfügbarkeit

- ◆ Begrifflichkeiten
- ◆ Der Begriff **Hochverfügbarkeit** bezeichnet die Fähigkeit eines Systems, bei Ausfall einer seiner Komponenten einen uneingeschränkten Betrieb zu gewährleisten.

Hochverfügbarkeit

- Ein System wird als verfügbar bezeichnet, wenn es in der Lage ist, die Aufgaben zu erfüllen, für die es vorgesehen ist. Als Verfügbarkeit wird die Wahrscheinlichkeit, dass ein System innerhalb eines spezifizierten Zeitraums funktionstüchtig (verfügbar) ist, bezeichnet. Die Verfügbarkeit wird als Verhältnis aus Downtime und Uptime eines Systems bemessen:

$$\text{Verfügbarkeit} = \frac{\text{Uptime}}{\text{Downtime} + \text{Uptime}}$$

Hochverfügbarkeit

- ◆ "Ein System gilt als **hochverfügbar**, wenn eine Anwendung auch im Fehlerfall weiterhin verfügbar ist und ohne unmittelbaren menschlichen Eingriff weiter genutzt werden kann. In der Konsequenz heißt dies, dass der Anwender keine oder nur eine kurze Unterbrechung wahrnimmt. Hochverfügbarkeit (abgekürzt auch HA, abgeleitet von engl. High Availability) bezeichnet also die Fähigkeit eines Systems, bei Ausfall einer seiner Komponenten einen uneingeschränkten Betrieb zu gewährleisten".[\[1\]](#)
- ◆ Die Definition des Begriffes **High Availability** (engl. Hochverfügbarkeit) durch das Institute of Electrical and Electronics Engineers ([IEEE](#)) lautet:
 - ◆ *"Availability of resources in a computer system, in the wake of component failures in the system."*

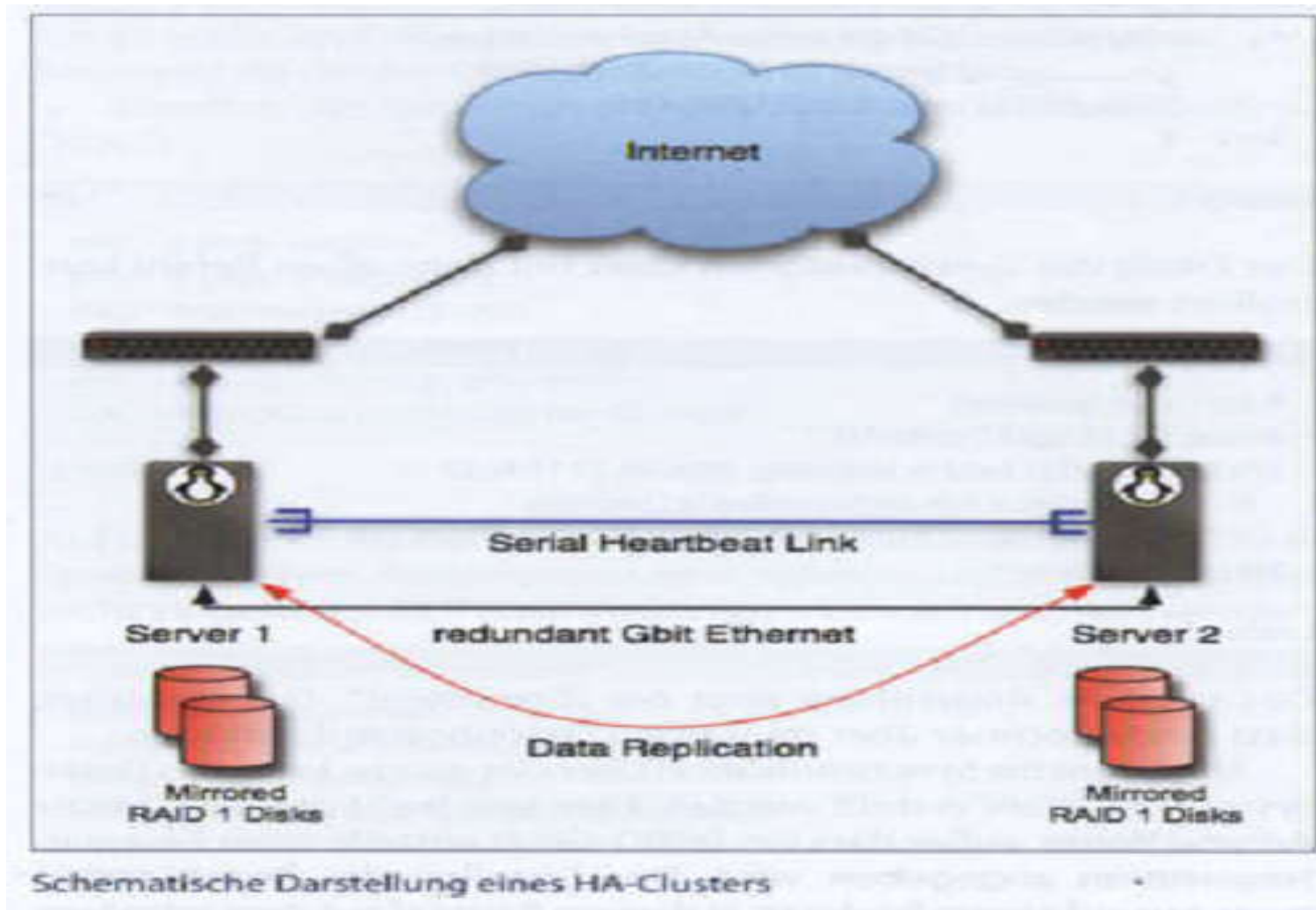
Hochverfügbarkeit

- ◆ Begrifflichkeiten
- ◆ Was ist ein HA-Cluster?

Hochverfügbarkeit

- ◆ Die Harvard Research Group (HRG) teilt Hochverfügbarkeit in ihrer Availability Environment Classification (kurz: AEC 0-5) in 6 Klassen ein [HRG_2002].
- ◆ AEC-0 99% 3,7 Tage
- ◆ AEC-1 99,9% 8,8 Stunden
- ◆ AEC-2 99,99% 52,2 Minuten
- ◆ AEC-3 99,999% 5,3 Minuten
- ◆ AEC-4 99,9999% 32 Sekunden
- ◆ AEC-5 99,99999 3 Sekunden

Hochverfügbarkeit



Annahmen

In den Konfigurationsbeispielen werden folgende IP-Adressen, Rechnernamen und Dateisysteme symbolisch verwendet:

- „links“ ist der Rechnername des primären Rechners
- „rechts“ ist der Rechnername des sekundären Rechners
- Als „öffentliches“ Netzwerk wird 10.0.0.0 angenommen
- Als „privates“ Netzwerk wird 192.168.0.0 angenommen
- Das DRBD-Blockgerät wird in /data eingehängt
- Das DRBD-Blockgerät ist /dev/drbd0
- Die Ressource heißt r0

Hochverfügbarkeit

```
resource r0 {
  protocol C;
  incon-degr-cmd "echo '!DRBD! pri on incon-degr' | wall ; sleep 60 ; halt -f";
  startup {
    wfc-timeout 60;
    degr-wfc-timeout 120; # 2 minutes.
  }
  disk {
    on-io-error detach;
  }
  net {
    # timeout 60; # 6 seconds (unit = 0.1 seconds)
    on-disconnect reconnect;
  }
  syncer {
    rate 30M;
    group 1;
  }
  on links {
    device /dev/drbd0;
    disk /dev/sda3;
    address 192.168.0.1:7788;
    meta-disk internal;
  }
  on rechte {
    device /dev/drbd0;
    disk /dev/sda3;
    address 192.168.0.2:7788;
    meta-disk internal;
  }
}
```

Hochverfügbarkeit

DRBD-Konfigurationsdatei

Die wichtigsten Parameter in /etc/drbd.conf	
resource	Der Name des entsprechenden DRBD-Dienstes.
protocol	Standardprotokoll ist „C“. „A“ und „B“ sind nützlich für asynchrone Spiegelung und werden hier nicht weiter besprochen.
incon-degr-cmd	Aktion im Fall der Feststellung einer „split-brain“-Situation, also wenn beide Rechner primary sind. Standard ist, den entsprechenden Rechner im Notfall zu stoppen.
on-io-error	Was passieren soll, wenn eine Festplatte einen Ein/Ausgabebefehl verursacht. Standard ist, das DRBD-Gerät zu dekonfigurieren.
on-disconnect	Definiert die Aktion, die unternommen wird, falls der sekundäre Rechner über das Netz nicht mehr erreichbar ist. Standard ist hier der Versuch, die Verbindung wiederherzustellen.
rate	Die Geschwindigkeit bei der Synchronisierung. Achtung: Die Angabe erfolgt in Byte pro Sekunde.
group	Falls mehrere Ressourcen definiert werden, kann mit diesem Parameter die Priorität festgelegt werden, welche Ressource nach einem Ausfall als erstes wieder synchronisiert werden soll.
meta-disk	Definiert, wo die Metadaten gespeichert werden. Auf kleinen Systemen ist dies üblicherweise „internal“, also auf derselben Partition. Alternativ kann eine eigene Partition pro Ressource angegeben werden. Vor allem bei größeren Systemen mit vielen physikalischen Festplatten kann dies geschwindigkeitssteigernd wirken.

Hochverfügbarkeit

```
SHRLL
```

```
links:~# drbdsetup /dev/drbd0 primary --do-what-I-say
```

Listing 3

Unmittelbar nach Eingabe dieses Befehls beginnt die Synchronisierung. Der Fortschritt kann mit folgendem Befehl beobachtet werden:

```
SHRLL
```

```
links:~# cat /proc/drbd
version: 0.7.17 (api:77/proto:74)
SYN Revision: 2125 build by lmb@chip, 2006-03-27 17:40:22
 0: cs:SyncTarget st:Primary/Secondary ld:Inconsistent
    ns:0 nr:3628 dw:3628 dr:0 al:0 bm:48 lo:0 pe:0 ua:0 ap:0
    [>.....] sync'ed: 1.1% (389428/393056)K
    finish: 0:01:37 speed: 3,628 (3,628) K/sec
links:~ #
```

Listing 4

Hochverfügbarkeit

Der Erfolg der Synchronisation kann mit demselben Befehl kontrolliert werden:

```
SVN
```

```
links: -- # cat /proc/drbd
```

```
version: 0.7.17 (api:77/proto:74)
```

```
SVN Revision: 2125 build by lmb@chip, 2006-03-27 17:40:22
```

```
0: cs:Connected st:Primary/Secondary ld:Consistent
```

```
ns:0 nr:393056 dw:393056 dr:0 al:0 bm:72 lo:0 pe:0 ua:0 ap:0
```

```
links: -- #
```

Hochverfügbarkeit

```
links:~#
```

```
links:~# mkfs.ext3 /dev/drbd0
```

```
links:~# tune2fs -i0 -c0 /dev/drbd0
```

Listing 6

Der zweite Befehl verhindert, dass das Dateisystem beim Einhängen unnötigerweise geprüft wird. Nach dem Anlegen des Dateisystems und des entsprechenden „Mountpoints“ kann das Gerät eingehängt werden:

```
links:~#
```

```
links:~# mkdir /data
```

Listing 7

```
links:~#
```

```
links:~# mount /dev/drbd0 /data
```

Listing 8

Erstellen der Datenverzeichnisse und Symlinks

In diesem Tutorial wird davon ausgegangen, dass das DRBD-Gerät in `/data` eingehängt wird. Dienste wie Apache oder MySQL speichern ihre Daten standardmäßig, aber an anderen Orten. Es gibt zwei Möglichkeiten, den Diensten den Standort ihrer Daten mitzuteilen: Erstens das Erstellen von Symlinks und zweitens die Anpassung in den Konfigurationsdateien. Ersteres ist zu bevorzugen, da dadurch die Originalkonfigurationen der Dienste nicht angepasst werden müssen.

Bei vielen neueren Linux-Distributionen wie Opensuse sind die Daten von Apache in `/srv/www` abgelegt, in anderen, wie Debian Sarge oder CentOS unter `/var/www`. Bei MySQL ist meistens `/var/lib/mysql` als Ort der Datenspeicherung vorgesehen.

Erstellen der Verzeichnisse und Symlinks auf dem Rechner "links":

Hochverfügbarkeit

Auf dem Rechner „rechts“ sind nur die Symlinks anzulegen. Diese Symlinks zeigen allerdings ins Leere, solange das DRBD-Gerät dort nicht eingehängt ist. Fehlermeldungen können ignoriert werden:

SHELL

```
rechts:~# rm -r /srv/www
rechts:~# ln -s /data/srv/www /srv/www
rechts:~# rm -r /var/lib/mysql
rechts:~# ln -s /data/var/lib/mysql /var/lib/mysql
```

Listing 10

Bevor nun Heartbeat konfiguriert und ein erstes Mal gestartet werden kann, muss das Dateisystem wieder getrennt und das DRBD-Gerät in den Status „secondary“ gebracht werden:

SHELL

```
links:~# umount /data
links:~# drbdadm secondary r0
```

Hochverfügbarkeit

Die Konfigurationsdateien sind in der Regel alle im eigenen Verzeichnis `/etc/ha.d/` zu finden. Heartbeat benötigt drei Konfigurationsdateien für den Betrieb:

- `/etc/ha.d/authkeys`
- `/etc/ha.d/ha.cf`
- `/etc/ha.d/haresources`

Diese Konfigurationsdateien müssen bis auf die Ausnahme der Datei „ha.cf“ auf beiden Servern identisch sein. Der Unterschied in der „ha.cf“ beschränkt sich auf die IP-Adresse des Partnerrechners.

Die Datei „authkeys“ enthält die Hashmethode und einen Schlüssel, mit dem die Kommunikation zwischen den beiden Servern rudimentär verschlüsselt wird.

```
root@node1:~# cat /etc/ha.d/authkeys
auth 1
1 sha1 supergeheimnis
```

Hochverfügbarkeit

In der zweiten Datei, `/etc/ha.d/ha.cf` (an dieser Stelle auf das Wesentliche gekürzt), werden die Netzwerkverbindungen und das Erkennen und Verhalten bei Ausfällen definiert.

```
logfile /var/log/ha-log
keepalive 2
deadtime 30
initdead 120
warntime 5
serial /dev/ttyS0
auto_failback off
ucast eth0 10.0.0.1
ucast eth1 192.168.0.2
node links
node rechts
```

Hochverfügbarkeit

Die wichtigsten Konfigurationsparameter in `/etc/ha.d/ha.cf` sind:

<code>keepalive</code>	Der zeitliche Abstand zwischen zwei Pings
<code>deadtime</code>	Definiert die Zeit, nach welcher der zweite Server als tot deklariert wird.
<code>initdead</code>	Beim Start von Heartbeat wird etwas länger gewartet, bevor der zweite Server als tot deklariert wird.
<code>warntime</code>	Nach der Zeit, nach der kein Ping zurückkam, wird eine Warnung in die Logdatei geschrieben (late heartbeat)
<code>auto_failback</code>	Definiert, ob nach der Wiederverfügbarkeit des primären Servers die Dienste automatisch auf den primären Server zurückschalten. Dies ist nützlich, wenn der sekundäre Server schlechter ausgestattet ist als der primäre.
<code>ucast</code>	Definiert die Netzwerkschnittstelle und die IP-Adresse des jeweils anderen Servers. Standardmäßig wird mcast (Multicast) verwendet, was allerdings nicht empfohlen wird. Die Anzahl der Linien mit diesem Parameter korrespondiert mit der Anzahl verwendeter Ethernet-Schnittstellen.
<code>node</code>	Bei den beiden Zeilen beginnend mit „node“ werden die Namen der beiden Server angegeben. Achtung Falle: Diese Namen müssen mit der Ausgabe des Befehls „uname -n“ unbedingt übereinstimmen.

Hochverfügbarkeit

Die dritte und wichtigste Datei `/etc/ha.d/haresources` enthält die Definitionen der Ressourcen, wie Service-IP-Adressen und die zu startenden Dienste.

```
configure (/etc/ha.d/haresources)
links 10.0.0.3 192.168.0.3 \
  drbdisk:r0 \
  Filesystem::/dev/drbd0::/data:ext3 \
  mysql \
  apache2
```

Listing 14

Konfiguration der Dienste

Wie bereits erwähnt, verwaltet Heartbeat den Start und Stopp der Services. Normalerweise werden Dienste wie MySQL oder Apache beim Systemstart automatisch gestartet. Das muss durch entsprechende Konfiguration verhindert werden.

Mit auf RPM basierenden Distributionen, wie Opensuse oder Redhat, kann wie folgt vorgegangen werden:

```
5/7/11  
chkconfig mysql off  
chkconfig apache2 off
```

Hochverfügbarkeit

Vorlesung: 27 Betriebssysteme

2017 Prof. Dr. G. Hellberg

Hochverfügbarkeit

Vorlesung: 28 Betriebssysteme

2017 Prof. Dr. G. Hellberg

Hochverfügbarkeit

Vorlesung: 29 Betriebssysteme

2017 Prof. Dr. G. Hellberg

Hochverfügbarkeit

Vorlesung: 30 Betriebssysteme

2017 Prof. Dr. G. Hellberg

Hochverfügbarkeit

Vorlesung: 31 Betriebssysteme

2017 Prof. Dr. G. Hellberg

ENDE

Fragen?



Gliederung

- Tannenbaum, Andrew, Moderne Betriebssysteme
- M. Weber, Foliensatz Universität Ulm
- Microsoft Whitepapers