TEC.1735 –MARS Pilotsystem - Dokumentation

Letztes Update - 28.09.2012 - joerg.mann@1und1.de

Dokumentation aus Basis der Versionen:

Mars – WIP 3.2

Marsadm – WIP 3.2

Mars-Status – 0.068j

Vom Leser dieser Dokumentation wird erwartet, dass er grundlegendes Wissen im Umgang mit Filesystemen, hier verwendeten Administrationstools hat und vor allem das er sicher im Umgang und dem Wissen um DRBD ist.

Mars ist derzeit nur in einer Beta-Version für den Pilotbetrieb der „statistik.schlund.de“ verfügbar. Dies impliziert, dass einerseits keine vollständige Implementierung der Kommandos und daraus folgend andererseits auch kein vollständiger Funktionsumfang im Vergleich zu ähnlichen Softwareprodukten vorhanden ist.

Im Rahmen des aktuellen Projektes TECITO.1735 wird der Rollout auf zunächst 10 Systeme im Bereich SHL geplant.

Dokument History

24.02.2012 – erste Version

27.02.2012 – Syntax, Schreibfehler, Anmerkungen Holger

29.02.2012 – Notizen von Daniel

10.04.2012 – Überarbeitung mit aktuellen Informationen, MARS-STATUS hinzugefügt

28.09.2012 – Anpassungen für aktuelle Version von Mars-Modul, MARSADM und MARS-STATUS

Inhaltsverzeichnis

0. Warum MARS ?

Die Entwicklung von MARS wurde begonnen da aktuell im Unternehmen eingesetzte Softwarepakete zur Replikation zahlreiche Schwachpunkte und Unzulänglichkeiten enthalten. Hauptsächlich trifft diese auf DRBD zu. Die wichtigsten Kritikpunkte sind hier:

- Möglichkeiten zur asynchronen Replikation
- Replikation über längere Distanzen (als BS – BAP)
- Replikation von Datenträgern über 4 TB
- I/O Verhalten bei größeren Lasten unzulänglich
- maximal 2 Partner möglich

MARS soll diese Problematik zusammen mit weiteren wichtigen Funktionseigenschaften beheben.

0.1 MARS Meilensteine

Für das Projekt MARS sind folgende Meilensteine vorhanden:

- 19.09.2011 - erster Code mars-Kernel, mars-Modul und MARSADM zum Testen verfügbar
- 21.10.2011 - erster Code MARS-STATUS verfügbar
- 01.09.2011 - Start Projekt TEC.1603 - Mars Light 1.0
- 07.02.2012 - Release Beta1
- 13.02.2012 - Inbetriebnahme Pilotsystem statistik.schlund.de
- 27.07.2012 - Start Projekt TECITO.1735 - Mars Light 2.0

0.2 Grundfunktionen MARS-Cluster

Der Aufbau von MARS ist dabei im Vergleich zu DRBD grundlegend anders. Während DRBD vollkommen synchron betrieben wird, arbeitet MARS asynchron. Synchron heißt dabei, dass ein „OK“ an die Applikation erst zurück gegeben wird, wenn die Daten auf beiden Seiten geschrieben wurden. Bei MARS (asynchron) werden die zu schreibenden Daten auf dem aktiven Node (Primary-System) in ein Transaktionslogfile geschrieben. Danach wird das „OK“ zur Applikation gemeldet. Im Folgenden wird auf dem aktiven Node das Transaktionslogfile abgespielt. Weiterhin wird das Transaktionslogfile (bzw. dessen Änderungen) an die inaktiven Nodes (Partner) übertragen und dort ebenfalls abgespielt. Der hier entstehende Zeitverzug hat jedoch keinen Einfluss auf die Applikationen.

MARS nutzt (im Vergleich zu DRBD) verschiedene Kommunikationswege um Daten zwischen den Nodes eines Clusters auszutauschen. Diese Wege können entsprechend Betriebsmodus getrennt verwendet und gesteuert werden. Im Einzelnen sind dies:

Cluster Statusinformationen

- Diese Daten werden fortlaufend zwischen allen Nodes im Cluster ausgetauscht, auch unabhängig davon ob Ressourcen zwischen den Nodes repliziert werden oder nicht
- Der Austausch von Statusinformationen im Cluster kann während des Betriebes von MARS nicht unterbrochen werden

→ Siehe dazu auch 2.1 Systemdaten /mars

Sync-Daten

- Sync-Daten werden zum Abgleich der Ressourcen zwischen dem aktiven und dem (den) inaktiven Nodes Blockweise übertragen
- Die Übertragung der Sync-Daten kann – getrennt für alle Nodes - pausiert und wieder gestartet werden

  → Siehe dazu auch 3.3 Ressource pause-sync / resume-sync

Logtransfer

- Transaktionslogfiles enthalten die fortlaufenden Schreibzugriffe auf die Ressource
- Getrennt für jeden Node kann die Übertragung der Transaktionslogfiles pausiert und auch wieder gestartet werden

  → Siehe dazu auch 3.7 Ressource log-rotate / log-delete / log-delete-all

1. Störungen im Betrieb - WTW ?

Eine Beschreibung bekannter Probleme und Störungen ist im Wiki unter

http://wiki.intranet.1and1.com/bin/view/PO/ProjektMars

hinterlegt. Ein Kurzhilfe ist den Man-Pages der Scripte bzw. mit der Option „*--help*“ verfügbar.

F & A:

Frage: Was für Kommandos können ohne geladenes Mars-Modul ausgeführt werden?
Antwort: create-cluster und join-cluster
→ Siehe auch unter 2.2 Ressource / Cluster einrichten

Frage: Kommunizieren die Nodes über SSH?! Wozu -A? No go.
Antwort: In der aktuellen Betaversion / Pilotbetrieb ja, später wird dies (sicher) überarbeitet

Frage: Kommando verify ?
Antwort: Diese Kommando ist in MARS nicht vorhanden. Im normalen Betriebsfall sind die Ressourcen nicht identisch, da MARS ja asynchron arbeitet und eine Verzögerung beim abspielen der Logfiles vorhanden ist. Als „Abhilfe“ kann hier der normale „Sync“ verwendet werden. Dieser arbeitet sozusagen als „Verify mit Repair“.
→ Siehe dazu auch unter 3.3 Ressource pause-sync / resume-sync

Frage: Kann ein fake-sync rückgängig gemacht werden?
Antwort: Ja, die Ressource auf invalidate schalten.
→ Siehe auch unter 3.4 Ressource fake-sync

Frage: Ist es eine gute Idee einen Port über 1024 zu nehmen?
Antwort: Default ist Port 7777, ab der aktuellen Version aber über zur Compiletime einstellbar.

Frage: Kann der Zugriff auf die LV (Disk-Device) gesperrt werden?
Antwort: Aktuell nicht vorhanden, ist als Todo aufgenommen.

Frage: Gibt es ein Rollback der Transaktionslogfiles?
Antwort: In der aktuell geplanten Light-MARS-Version nicht, erst in einer der späteren Versionen.

2. Einrichtung

Die nachfolgende Dokumentation geht davon aus, dass die Software MARS als Kernelmodul in der aktuellen Version auf den Systemen installiert ist. Dies gilt ebenso für die verwendeten Hilfsprogramme von MARS.

In den aktuell verwendeten Beta-Version von MARS werden einige Parameter für die Betriebsumgebung in den Konfig-Files des Mars-Modules verwaltet. Diese Parameter sind für die derzeit verwendeten Systeme angepasst und hier auch allgemein gültig.

2.1 Systemdaten /mars

Zur Einrichtung von MARS ist auf den System eine Partition /mars (vorzugsweise – siehe 2.0) einzurichten. Auf dieser werden interne Files und Systemlinks (als Sym-Link-Tree) zu Steuerung von MARS, sowie die Transaktionslogfiles abgespeichert. Diese Partition sollte nach Möglichkeit nicht auf dem gleichen Volumes/Platten/Enclosures liegen, wo auch die Ressourcen liegen. Zu empfehlen sind hier getrennt Disks, mit einem entsprechenden Raidset. Die Größe der Partition sollte ausreichend sein um eine möglichst große Anzahl von Transaktionslogfiles zu speichern. Zu beachten ist, dass mit hoher der Anzahl der Transaktionslogfiles mehr Möglichkeiten zur eventuellen Fehlerbehebung im „Fall der Fälle“ zur Verfügung stehen.

Die vom MARS angelegten Systeminformationen werden zwischen allen im Cluster bekannten Nodes ausgetauscht. Dieser Austausch läuft parallel auf allen bekannten Nodes die lokalen und kann nicht angehalten werden. Zu den Informationen gehören Daten zu den verwendeten Ressourcen, deren Größe und Zustand. Ebenfalls werden hier Informationen zum Zustand des Primary/Secondary, des Sync- und Replay-Zustandes übertragen. Durch den Austausch der Statusinformationen sind auf allen im Cluster bekannten Nodes alle Informationen zu allen beteiligten Nodes und Ressourcen vorhanden.

→ Erfahrungen (der statistik.schlund.de) zeigen, dass 15GB pro Stunden geschrieben werden können.

→ In der aktuellen Version sollte für die Systemdaten und das Disk-Device ein unterschiedlicher Filesystemtyp verwendet werden.

2.2 Ressource / Cluster einrichten

Die Einrichtung der Ressourcen (Datenvolume) erfolgt nach den Vorgaben des Systems. Es sind entsprechende PV's / LV's bzw. Disk's bereit zu stellen. In Beispiel dieser Dokumentation legen wir in der Volume Group „vg-test“ ein Logical Volume „Mars-ResA“ mit 100GB an.

lvcreate vg-test -L 100G -n LV-ResA

Das Volume ist auf dem aktiven System (Primary) und den inaktiven Systemen (Secondary) identisch anzulegen. Als nächstes wird das aktive System in Betrieb genommen. Dazu wird in Abfolge der Cluster und die Ressource innerhalb von MARS angelegt, sowie das Modul geladen.

MARSADM create-cluster

modporbe mars

MARSADM create-resource Mars-ResA /dev/vg-test/LV-ResA

Das Mars-Device der angelegten Ressource steht jetzt unter „/dev/mars/Mars-ResA“ zur Verfügung und kann verwendet werden. Im nächsten Schritt werden die entsprechenden inaktiven Nodes in Betrieb genommen. Dazu werden die Nodes dem Cluster hinzugefügt. Somit wird zunächst sicher gestellt, dass alle

Statusinformationen des Cluster verfügbar sind. Nachdem der inaktive Node mit dem Cluster verbunden ist, wird die Ressource hinzugefügt.

→ Das „cretate“ Kommando darf immer nur einmal auf einem Node ausgeführt werden, da sonst die Struktur des Systems zerstört wird.

-> Zu beachten ist, dass eine entsprechende Freischaltung und ein Zugriff (ssh -A xxx) aktiviert ist.

MARSADM join-cluster $hostname-primary

modprobe mars

MARSADM join-resource Mars-ResA /dev/vg-test/LV-ResA

Entsprechend Anzahl der inaktiven Nodes und Ressourcen sind die Schritte zu wiederholen. Die Installation bzw. Vorbereitung des Clusters ist jetzt abgeschlossen, die inaktiven Nodes sind jetzt betriebsbereit.

Alle im Cluster vorhandenen Nodes tauschen untereinander fortlaufend selbstständig Systemdaten aus. Dies er Mechanismus ist unabhängig von den verbunden Ressourcen.

→ Siehe dazu auch 2.1 Systemdaten /mars

3. MARSADM

Die Steuerung eines Mars-Clusters erfolgt über das Hilfsprogramm „MARSADM“. Die Anzeige der verschiedenen Zustände des Clusters, der Nodes und der Transaktionslogfiles erfolgt über das Hilfsprogramm „MARS-STATUS“. Eingriffe von „Hand“ sollten generell vermieden werden. Dies auch aus dem Grund, dass verschiedene Funktionen aus technischen Gründen und bedingt durch den fortlaufenden Entwicklungsprozess der Software ständig verändert und angepasst werden. Bei der Entwicklung von MARS wurde darauf geachtet, dass hinsichtlich der implementierten Kommandos, die gleichen Funktionen angewandt und hinterlegt wurden.

→ Im Gegensatz zu den vom DRBD bekannten Kommandos, ist bei den nachfolgend beschriebenen Mars-Kommandos mehrfach der Zusatz „local“ vorhanden. In der Regel wirken sich Kommandos auf alle angeschlossenen Systeme aus und werden hier ausgeführt. Mit dem Zusatz „local“ wirken diese Kommandos jedoch nur auf dem System, auf dem sie eingegeben wurden. So wird beispielsweise bei dem Kommando „***MARSADM disconnect Mars-ResA***“ ein disconnected auf allen Systemen des Mars-ResA durchgeführt, während bei dem Kommando „***MARSADM disconnect-local Mars-ResA***“ das Mars-ResA nur auf dem lokalen System disconnected wird.

3.1 Ressource connect / disconnect

Nach der Einrichtung der Nodes sind diese in der Regel „connected“. In „disconneted“ Zustand werden keine Transaktionslogfiles und keine Sync-Daten, sondern ausschließlich Statusinformationen des Clusters übertragen. Mit den folgenden Kommandos kann diese Zustände verändert werden:

MARSADM connect Mars-ResA	→ die Ressource wird auf allen Nodes connected
MARSADM disconnect Mars-ResA	→ die Ressource wird auf allen Nodes disconnected
MARSADM connect-local Mars-ResA	→ die Ressource wird nur local connected
MARSADM disconnect-local Mars-ResA	→ die Ressource wird nur local disconnected

→ Anstelle des Ressourcennamens kann auch der Zusatz „all“ verwendet – der sich dann auf alle Ressourcen im Cluster auswirkt.

→ Die Kommandos „Connect / Disconnect“ haben keine Auswirkung auf einem Primary-Node, sondern nur auf den Secondary Nodes. Gleiches gilt für die *-local Kommandos.

→ Der Status des „Connect-Schalter“ ist über MARS-STATUS einsehbar.

→ Siehe dazu auch unter 2.1 Systemdaten /mars

3.2 Ressource attach / detach

Mit den Kommandos „detach“ kann eine Ressource vollständig vom Betrieb abgetrennt werden. Damit werden alle Schreib- und Leseoperation auf die Ressource unterbunden. Über das Kommando „attach“ kann dieser Zustand wieder aufgehoben werden.

MARSADM detach Mars-ResA	→ Zugriffe auf die Ressource Mars-ResA werden unterbunden
MARSADM attach Mars-ResA	→ Zugriffe auf die Ressource Mars-ResA werden wieder erlaubt

→ Der Status des „Attach-Schalter“ ist über MARS-STATUS einsehbar.

→ Siehe dazu auch unter 2.1 Systemdaten /mars

Während des „detach“ Betriebes werden nur die Statusinformationen des Clusters übertragen. Eine Wiedergabe von möglichen Transaktionslogfiles oder ein laufender Sync-Prozesse wird angehalten.

3.3 Ressource pause-sync / resume-sync

Analog dem vom DRBD bekannten Verhalten müssen die Ressourcen bei (bzw.) vor der ersten Inbetriebnahme oder für den Fall einer vollständigen Wiederherstellung gesynct (abgeglichen) werden. Dabei wird die betreffende Ressource blockweise vom aktiven Node auf den zu syncenden Node übertragen. Während des laufenden Sync-Prozesses arbeiteten alle anderen Nodes uneingeschränkt weiter. Auf den Node, auf dem der Sync-Prozess läuft, werden zu dieser Zeit keine Transaktionslogfiles abgespielt.

Mit den folgenden Kommandos kann der Sync-Prozess angehalten bzw. wieder gestartet werden.

MARSADM pause-sync Mars-ResA	→ der Sync-Prozess für die Ressource wird auf allen Nodes angehalten
MARSADM pause-sync-local Mars-ResA	→ der Sync-Prozess für die Ressource wird auf dem lokalen Node angehalten
MARSADM resume-sync Mars-ResA	→ der Sync-Prozess für die Ressource wird auf allen Nodes wieder gestartet
MARSADM resume-sync-local Mars-ResA	→ der Sync-Prozess für die Ressource wird auf dem lokalen Node wieder gestartet

→ Die Kommandos „Sync / Replay“ hat keine Auswirkung auf dem Primary-Node.

Zu beachten ist weiterhin, dass der Sync-Prozess als Fast-Full-Sync im MARS implementiert ist. Dieser Modus impliziert einerseits das gleichzeitige syncen der Ressourcen auf der einen Seite und das gleichzeitige abspielen von Transaktionslogfiles. Im Ergebnis findet kein wirklich voller Sync, es werden nur veraltete Daten neu geschrieben.

→ Wenn der Sync-Prozesses auf einem Node gestartet wird, dass sich bereits im Betrieb gefunden hat (nachträgliches Sync als Verify oder Kennzeichnung als inkonsistent) bewirkt zusätzlich das nicht mehr benötigte Transaktionslogfiles und andere Statusinformationen gelöscht werden.

3.4 Ressource fake-sync

Unter bestimmten Umständen – wie z.B. zu Testzwecken oder bei einer Erstinbetriebnahme von leeren Ressourcen – kann der Sync-Prozess gefakt (simuliert) werden. Damit erklärt MARS den Datenbestand als valide (gültig). Zur Steuerung wird das nachfolgende Kommando verwendet:

MARSADM fake-sync Mars-ResA	→ die Ressource wird als gesynct betrachtet

→ Anstelle des Ressourcennamens kann auch der Zusatz „all“ verwendet – der sich dann auf alle Ressourcen im Cluster auswirkt.

→ Der Status des „Sync-Schalter“ ist über MARS-STATUS einsehbar.

3.5 Ressource invalidate-remote

Entsprechend Betriebsmodus ist es notwendig, dass Daten einer Ressource für ungültig erklärt werden müssen, um so z.B. einen erneuten Fast-Full-Sync zu starten. Die Steuerung erfolgt mit den folgenden Kommandos:

MARSADM invalidate-remote Mars-ResA → die Ressource wird entfernten Node ungültig

→ Das Kommando „*MARSADM invalidate Ressourcename*“ ist aus Gründen der Kompatibilität vorhanden, jedoch in MARS ohne Funktion.

→ Anstelle des Ressourcennamens kann auch der Zusatz „all“ verwendet – der sich dann auf alle Ressourcen im Cluster auswirkt.

3.6 Ressource pause-replay / resume-replay

Ebenso wie der Sync-Prozess kann auch der Replay-Prozess – also das Abspielen der Transaktionslogfiles – pausiert und wieder neu gestartet werden.

MARSADM pause-replay Mars-ResA	→ für die Ressource wird auf allen Nodes keine Logfiles mehr abspielen
MARSADM pause-replay-local Mars-ResA	→ für die Ressource wird nur local kein Logfile mehr abspielen
MARSADM resume-replay Mars-ResA	→ Logfile für die Ressource auf allen Nodes wieder abspielen
MARSADM resume-replay-local Mars-ResA	→ Logfile für die Ressource wird nur local wieder abspielen

→ Anstelle des Ressourcennamens kann auch der Zusatz „all“ verwendet – der sich dann auf alle Ressourcen im Cluster auswirkt.

→ Der Status des „Replay-Schalter“ ist über MARS-STATUS einsehbar.

3.7 Ressource log-rotate / log-delete / log-delete-all

Im Betrieb werden alle Änderungen der Ressource (Schreibzugriffe auf das Datenvolumen) in Transaktionslogfiles abgespeichert. Diese werden auf alle Nodes im Cluster übertragen, die mit der Ressource verbunden sind (join-ressource). Für den Betrieb ist eine Verwaltung der Transaktionslogfiles zwingend notwendig. Die Verwaltung ist entsprechend den Betriebsparametern anzupassen. Gelöscht werden können im Betrieb nur nicht mehr benötigte und abgespielte Transaktionslogfiles. Nicht mehr benötigt heißt in diesen Fall, dass eine Reihe von Bedingungen erfüllt sein müssen:

- das Transaktionslogfile muss auf alle aktiven und inaktiven Nodes übertragen sein
- die Transaktionslogfiles müssen auf den beteiligtem Node abgespielt sein.
- das Transaktionslogfile muss auf allen Systemen den gleichen Zustand (siehe TODO) haben

Die Transaktionslogfiles werden (wie auch andere interne Files) mit einer fortlaufenden Versionsnummer

versehen. Durch eine Rotation der Transaktionslogfiles, wird eine neue Version des Transaktionslogfiles erzeugt.

MARSADM log-rotate Mars-ResA	→ für die Ressource das Logfile rotieren
MARSADM log-delete Mars-ResA	→ für die Ressource nächstes inaktive Logfile löschen
MARSADM log-delete-all Mars-ResA	→ für die Ressource alle nicht mehr benötigten Logfiles löschen

→ Anstelle des Ressourcennamens kann auch der Zusatz „all“ verwendet – der sich dann auf alle Ressourcen im Cluster auswirkt.

3.8 Ressource Primary / Secondary

Mit den Kommandos „*primary*“ und „*secondary*“ kann ein Node in den aktiven bzw. passiven Modus umgeschaltet werden. Im Cluster kann es für jede Ressource nur einen aktiven Node (Primary) geben. Die Anzahl der inaktiven Nodes (Secondarys) ist nicht beschränkt. Die Umschaltung des Modus wirkt sich auf alle Systeme die mit dieser Ressource verbunden sind aus. Dies kann (zB. wegen einem gemouteten Ressource auch erst zu einem späterem Zeitpunkt erfolgen).

MARSADM primary Mars-ResA	→ für die Ressource wird in den aktiven Modus geschaltet
MARSADM secondary Mars-ResA	→ für die Ressource wird in den passiven Modus geschaltet

→ Zu beachten ist, dass die Umschaltung des Modus sich auch auf alle anderen Nodes im Cluster auswirkt.

→ Zu beachten ist weiterhin, dass bei einem laufenden Sync-Prozess das Kommando „secondary“ auf dem Primary-Node nicht ausgeführt werden sollte. Dies würde dazu führen das der Secondary-Node keinen aktiven Primary-Node mehr hat von dem er Daten syncen kann. In Folge entstehen inkonsistente Daten.

3.9 Ressource up / down

Die Kommandos „up“ und „down“ sind nur aus Gründen der Kompatibilität zu DRBD in MARS eingefügt wurden. Im normalen Betrieb sind diese Kommandos eigentlich nicht notwendig. Entgegen dem Verhalten von DRBD ändert sich der Status der Ressourcen mit dem Laden/Starten des Moduls bei MARS nicht.

Beim Aufruf von MARS mit dem Kommando „down“ werden die folgenden Aktionen ausgeführt:

- pause-replay -> Abspielen des Transaktionslogfiles anhalten
- pause-sync -> laufenden Sync-Prozess anhalten
- disconnect -> keine Logfiles/Sync-Daten mehr übertragen
- detach -> Ressource vom System trennen

→ Anstelle des Ressourcennamens kann auch der Zusatz „all“ verwendet – der sich dann auf alle Ressourcen im Cluster auswirkt.

Das Kommando „up“ führt die entsprechenden Kommandos in umgekehrter Reihenfolge (attach, connect, resume-sync, resume-replay) aus.

4. MARS-STATUS

Während des Betriebes von MARS werden zwischen allen Nodes im Cluster Statusinformationen ausgetauscht, die von den beteiligten Nodes ausgewertet und verarbeitet werden. Weiterhin werden verschiedene Parameter aus der Betriebsumgebung ausgelesen, auswertet und ebenfalls verarbeitet. Im Ergebnis erfolgt ein speziell auf den Node angepasster Betrieb des MARS Modules.

→ Das Programm MARS-STATUS befindet sich aktuell in der Entwicklung. Demnach kann eine fehlerfreie Erkennung der verschiedenen Zustände innerhalb von MARS – insbesondere bei der Erkennung des Status – nicht garantiert werden.

4.1 Monitor Funktion

Für den Betrieb im Monitor-Modus wird „MARS-STATUS“ mit dem Parameter „*–monitor“* aufgerufen. Vom Programm wird (sofern vorhanden) ein entsprechendes Exit-Code und eine Fehlermeldung ausgegeben. Das Format ist identisch dem Format, dass von Nagios-Checks verwendet wird.
[Funktion ist aktuell noch nicht implementiert]

4.2 Optionen

Das MARS-STATUS Programm kann entsprechend Anforderungen mit verschiedenen Optionen aufgerufen werden:

-- ressource x

Mit dieser Option wird die Ausgabe der Statusinformation auf eine bestimmte Ressource beschränkt.

-- interval x

Die Anzeige rotiert selbstständig in x Sekunden.

-- history

Neben den Statusinformationen werden Angaben zu den Transaktionslogfiles, deren Versionen und der Status ihrer Verarbeitung angezeigt.

-- system

Die Ausgabe wird um die einstellbaren Systemoptionen erweitert.

-- monitor

Die Option „monitor“ zeigt die Systemzustände (entsprechend ausgewählten Optionen) zusammengefasst an.

-- cstate

Option gibt als Wert Connect oder Disconnet zurück. Diese Option benötigt als weitere Option den Namen einer bestimmten Ressource. Grundlage für das Connect / Disconnet ist der Zustand der „Switches“ für die betreffende Ressource.

→ Option ist nur zur Kompatibilität zu DRBD vorhanden. Sie gibt keine reale Auskunft über den Zustand der Ressource.

-- dstate

Option gibt als Wert UpToDate, UpDateIng, OutDate, InvaliDate, SwitchOff oder Failed/unknow zurück. Diese Option benötigt als weitere Option den Namen einer bestimmten Ressource. Grundlage für die verschiedenen Zustände sind:

- UpToDate : kein Replay aktiv | kein Sync aktiv | Delay 0 | Logfiles ok | keine Todos | System ok
- UpDateIng : wie vor, jedoch Replay aktiv | Delay < 1 min
- OutDate : wie vor, jedoch Delay > 1 min
- InvaliDate : Sync aktiv oder notwendig | Konsistenz Logfiles ungenau
- SwitchOff : nicht alle Switches sind „on“
- Failed : Fehler im System (Disk voll, Netzwerkfehler, not joined)

→ Option ist nur zur Kompatibilität zu DRBD vorhanden. Sie gibt keine reale Auskunft über den Zustand der Ressource.

-- role

Option gibt als Wert „Primary“ oder „Secondary“ zurück. Diese Option benötigt als weitere Option den Namen einer bestimmten Ressource.

→ Option ist nur zur Kompatibilität zu DRBD vorhanden. Sie gibt keine reale Auskunft über den Zustand der Ressource.

-- debug

Es werden zusätzlich interne Ausgaben des MARS-Modules aus dem Kernel ausgelesen.

Eine Kombination der verschiedenen Parameter ist möglich und gewollt.

4.3 Anzeige

In den Ausgaben von MARS-STATUS werden an verschiedenen Stellen Angaben hinzugefügt, die einen Hinweis auf mögliche notwendige Arbeiten anzeigen. Unterschieden werden hier:

- WORK: System arbeitet (zB. Sync / Replay läuft)
- TODO: System ist aktuell mit einem Fehler versehen (zB. Logfiles sind nicht identisch)
- HINT: System benötigt Eingriff (zB. Ältere Logfiles können gelöscht werden)

(*roter Text Kommentar zu den Ausgaben*)

a) MARS-STATUS ohne Optionen :

```
MARS Status - istore-test-bap7, 0.069
MARS Admin  - /usr/lib/mars/0.1348749967-f021125/MARSADM 267700a8a877ad79bc9b86f17891c9d7bf6fa766
MARS Module - 0.1348749967-f021125 (  2012-09-27 14:54:10)
MARS Kernel - 3.2.26-20120925
--------------------------------------------------------------------------------
(Angaben zu den eingesetzten Versionen)

-> check resource DeviceA, with 0.010TB, Primary Node is istore-test-bs7
(Anzeige der Zusammenfassung des Devices der Resource)

  -> local Node (istore-test-bap7) as Secondary, System alive
        Devices : Disk-Device /dev/vg-test-bap7/DeviceA, used as Mars-Device /dev/mars/DeviceA, not resized
                  ---> HINT: unable to mount, Device is Secondary or mars is starting
        Sync    : 10737418240 bytes (0.010TB) synced = 100.00%
        Logfile : 94097408 bytes (0.088GB) in log-000007562 active , Logfiles received with -592.73 mb/s
        Replayed: 0 bytes (0.000GB) now replayed, Todo 0 (0.000GB) = 0.00%
                  ---> HINT: Replay not started, Logfile inactive = (Size: 94097408)
        Actual  : Status=Secondary, Syncstatus=off, Logfileupdate=on
        Switches: Attach=on Connect=on Sync=on AllowReplay=on
        Status  : OutDate = (joined)(alive)(synced)(replay stopped)(attached)(connected)(synced)(replayed)
        (Anzeige des lokalen Zustandes der Resource)

  -> remote Node (istore-test-bs7) as Primary, System alive
        Devices : Disk-Device /dev/vg-test-bs7/DeviceA, used as Mars-Device /dev/mars/DeviceA, not resized
        Sync    : 10737418240 bytes (0.010TB) synced = 100.00%
        Logfile : 94097408 bytes (0.088GB) in log-000007562 active
        Replayed: 10754456 bytes (0.010GB) now replayed, Todo 8360 (0.000GB) = 11.43%
                  ---> WORK: Replay in progress = (11.43% < 100.00%)
        Actual  : Status=Primary, used Device=on
        Switches: Attach=on [masked: Connect=on Sync=on AllowReplay=on]
        Status  : UpDateIng = (joined)(alive)(synced)(replay running1)(attached)(connected)(synced)(replayed)
        (Anzeige des remote Zustandes der Resource)

  -> modus for Device-BS7 is clustered (2 nodes)
(Anzeige Zusammenfassung Resource)
```

Details:

- *Device: physikalisches Device für die Resource, Resourcenname und Check nach Resize*
- *Sync: Größe des Device*
  s und der Zustand der gesyncten Daten
- *Logfile: Größe und Name des Logfiles*
- *Replayed: Größe vom abgespielten Logfile*
- *Actual: Secondary/Primary, Sync aktiv/inaktiv, Logfileupdate aktiv/inaktiv*
- *Switches: Aattch, Connect, Sync, Allowreplay*
- *Status: Summary aus allen Angaben*

b) MARS-STATUS mit Optionen „--system“:

-> AVG-Limit: is unset, used full speed
-> Network-Traffic-Limit: is unset, used full speed, current 5 kb/s
-> Memory-Limit: is unset, used full speed
-> Server-IO-Limit: is unset, used full speed
-> Free-Space-Limit on /mars: is set to 8 mb, current 33195.25 mb
-> Free-Space-Limit for Auto-Log-Delete: is set to 8 gb
-> Free-Space-Limit for Auto-Log-Rotate: is set to 32 gb
-> Mars-Transaktion running normaly
-> Diskspace on Cluster: ok
(Anzeige des aktuell eingestellte Systemparameter, soweit verfügbar (vom MARS-Modul bereit gestellt) werden die SOLL und IST Werte mit angezeigt)

c) MARS-STATUS mit Optionen „--debug“:

-> MARS WARNINGS:
Fri Sep 28 12:57:00 2012: mars_peer0[5] ernel/mars_net.c 520 mars_recv_raw(): #7068 got EOF from ……
Fri Sep 28 12:57:00 2012: mars_peer0[5] ernel/mars_net.c 695 _mars_recv_struct(): #7068 called from line 70 status = -32
Fri Sep 28 12:57:00 2012: mars_peer0[5] ernel/mars_net.c 810 _mars_recv_struct(): #7068 called from line 70 status = -32
Fri Sep 28 12:57:00 2012: mars_peer0[5] old/mars_light.c 1116 peer_thread(): communication error on receive, status = -32
-> MARS ERRORS:
Fri Sep 28 10:28:15 2012: mars_light[5] old/sy_generic.c 357 get_inode(): cannot stat '/mars/r.....
(Anzeige /proc/sys/mars/)*

d) MARS-STATUS mit Optionen „--history“:

-> History Replay/Status
Logfile Version: 000007568 - Size: 278588680
Source: istore-test-bap7, Check: ad7165b08c7d31812e20be89af3c7c6a, ReplayPosition: 278588680, Todo: 0 blocks
Source: istore-test-bs7, Check: ad7165b08c7d31812e20be89af3c7c6a, ReplayPosition: 278588680, Todo: 0 blocks
---> WORK: Logfiles has all equal Sizes and Checksums, can be deleted ...
Logfile Version: 000007569 - Size: 0
Source: istore-test-bap7, Check: 2bd0947f1d708b7bd18aed7a0f56acdc, ReplayPosition: 0, Todo: 0 blocks
Source: istore-test-bs7, Check: 2bd0947f1d708b7bd18aed7a0f56acdc, ReplayPosition: 0, Todo: 0 blocks
---> WORK: logfiles are actual and unused.

Details:

- *Logfile Version: Nummer und Größe des Logfiles*
- *Source: Ursprung des Logfileeintrages (auf dem es erstellt wurde)*
- *Check: Prüfsumme des Logfiles*
- *ReplayPosition: Aktuell abgepsielter Zustand*
- *Todo: was noch abzuspielen ist*

e) MARS-STATUS mit Optionen „–monitor“:

Secondary [resource-DeviceA on istore-test-bap7]
(Anzeige wie Option --role)
UpToDate [(joined)(alive)(synced)(replay wait)(attached)(connected)(synced)(replayed)]
(Anzeige wie Option --dstate)
Connect [resource-DeviceA on istore-test-bap7]
(Anzeige wie Option --cstate)

→ Die Option „–monitor“ benötigt nicht zwingend einen Ressourcen-Namen.

f) MARS-STATUS mit Optionen „–cstate –ressource=RessourcenNamen“:

Connect [resource-DeviceA on istore-test-bap7]
(Anzeige Statuswert [Name des Ressource Name des Hostes]

→ Die Option „–cstate“ benötigt zwingend einen Ressourcen-Namen.

g) MARS-STATUS mit Optionen „–dstate –ressource=RessourcenNamen“:

UpToDate [resource-DeviceA on istore-test-bap7]
(Anzeige Statuswert [Name des Ressource Name des Hostes]

→ Die Option „–dstate“ benötigt zwingend einen Ressourcen-Namen.

h) MARS-STATUS mit Optionen „–role –ressource=RessourcenNamen“:

Primary [resource-DeviceA on istore-test-bap7]
(Anzeige Statuswert [Name des Ressource Name des Hostes]

→ Die Option „–role“ benötigt zwingend einen Ressourcen-Namen.

5. Betriebsparameter

Während des Betriebes von MARS werden zwischen allen Nodes im Cluster automatisch Statusinformationen ausgetauscht, die von den beteiligten Nodes ausgewertet und verarbeitet werden. Weiterhin werden verschiedene Parameter aus der Betriebsumgebung ausgelesen, auswertet und ebenfalls verarbeitet. Im Ergebnis erfolgt ein speziell auf den Node angepasster Betrieb des MARS Moduls. Die Änderung dieser Werte gehen nach einem Neuladen des Moduls verloren (proc Filesystem!).

5.1 Load AVG

- Wert in der Datei unter „*/proc/sys/mars/loadavg_limit*“
- Parameter mit dem maximalen Load-AVG des lokalen Node
- bei Erreichen dieses Wertes wird ein disconnect der Ressource durchgeführt

5.2 Network-Traffic-Limit

- Anzeige für das Limit für den ausgehenden Netzwerk-Traffic insgesamt

5.3 Server-IO-Limit

- Anzeige für das Limit für das eingehenden Netzwerk-Traffic insgesamt

5.3 Memory-Limit

- Anzeige für das Limit des zur Verfügung stehenden Speichers

5.4 Free-Space-Limit

- Parameter mit dem maximalen Füllstand des /mars Devices

5.5 Free-Space-Limit Auto-Log-Delete

- Wie vor, jedoch Speicherplatz bis ein automatisches Log-Delete durchgeführt wird

5.6 Free-Space-Limit Auto-Log-Rotate

- Maximale Größe die ein Transaktionslogfile haben darf, bis ein automatisches Log-Rotate durchgeführt wird

6. Debug

In der aktuellen Mars-Version sind folgende Debug-Optionen nutzbar:

MARS-STATUS –debug	= Auswertung der Dateien *„/proc/sys/errors“* und *„…/warnings*“
/mars/log.txt	= Diese Datei muss angelegt werden (touch /mars/log.txt) und enthält Debug-Informationen des Mars-Modules. Die Datei kann sehr groß werden!

7. Betriebsfälle

7.1 Mars-Systemdirectory voll

Im Betrieb kann es vorkommen das das Mars-Systemdirectory (/mars) voll wird. Dies wird in der Praxis beispielsweise durch das Debug-Logfile (/mars/log.txt) oder häufiger durch die Transaktionslogfiles vorkommen.

Für das kontinuierliche rotieren und löschen der Transaktionslogfiles sind entsprechende Cron-Jobs auf dem Node einzurichten. Sollte das Mars-Systemdirectory trotzdem „überlaufen“, schaltet MARS in den Notbetrieb. Auf dem Primary Node werden dabei folgende Aktionen ausgeführt:

1. es wird versucht Logfiles zu rotieren und zu löschen um wieder Speicherplatz zur Verfügung zu haben
2. sofern 1. immer noch nicht ausreicht, schaltet der Primary auf „Durchgang“. Es werden keine Transaktionslogfiles mehr geschrieben, die Daten werden direkt auf das Disk-Device geschrieben. Weiterhin werden die Secondary Nodes als „invalidate“ gekennzeichnet.

Um diesen Fall wieder aufzulösen, ist im Schritt der Primary Node wieder in einen Zustand zu bringen, der wieder Transaktionslogfiles schreibt. Dazu ist als erstes das System wieder mit ausreichend Platz zu versehen. Danach ist der Primary Node kurzzeitig in den Secondary Modus zu schalten.

Im nächsten Schritt sind die Secondary Nodes wieder in Betrieb zu setzen. Hierfür ist ein Sync-Prozess zu starten.

7.2 Transaktionslogfiles

Im normalen Betrieb werden die Transaktionslogfiles vom Primary Node auf den Secondary Node übertragen. Sollten hier – oder auf anderen Wegen – Fehler entstehen, haben die Transaktionslogfiles unterschiedliche Prüfsummen.

Weiterhin kann es bei Problemfällen vorkommen, dass nicht alle benötigten Transaktionslogfiles in den entsprechenden Versionen auf den Secondary Nodes verfügbar sind. Dies kann zum Beispiel eintreten wenn der Primary Node ausgefallen und wieder in betrieb genommen wurde, wenn auf dem Primary Node das Mars-Systemdirectory zugelaufen ist.

In den genannten Fällen kann ein geregelter Betrieb durch ein Kommando „invalidate-remote“ und einen nachfolgenden Sync wieder hergestellt werden. Entsprechend Ausfallzeit und Datenbestand ist die benötigte Zeit nicht sehr hoch (relativ), da ja ein Fast-Full-Sync verwendet wird.

7.3 Split-Brain

Unter bestimmten – ungewollten – Umständen kann es vorkommen das ein Split-Brain Zustand entsteht. Im Fall von MARS bedeutet dies, dass unterschiedliche Prüfsummen eines Transaktionslogfiles vorhanden sind. In der Praxis kann dies beispielsweise (siehe oben) vorkommen wenn Fehler auf dem Primary Node auftreten.

Abhilfe kann hier nur geschaffen werden, in dem von „Hand“ der richtige Node auf Primary gesetzt wird. Alle anderen (Secondary) Nodes müssen nachfolgend neu gesynct werden.

8. Übersetzungen

Die folgenden Übersetzungen beziehen sich auf den Zusammenhang mit Mars.

Cluster	= Verbund aller Systeme, unabhängig ob Ressourcen verbunden oder gejoint werden.
Node	= einzelnes System das mit dem Cluster verbunden ist
Primary	= aktiver Node
Ressource	= Datenvolumen auf die zu replizierenden Daten liegen
Sync-Daten	= Block-Daten die zum Sync einer Ressource notwendig sind
Secondary	= passiver Node
Transaktionslogfile	= Logfile, in dem die einzelnen Änderungen der Ressource hinterlegt sind
Version	= Nummer der fortlaufenden Transaktionslogfiles
Mars-Systemdirectory	= Datenvolumen auf dem Mars-Status-Informationen und Logfiles abgelegt werden typischer Weise „/mars“
Mars-Device	= Datenvolumen das als Ressource von MARS bereit gestellt wird. typischer Weise „/dev/mars/resourcenname“
Disk-Device	= Datenvolumen das „unter“ dem Mars-devices liegt Typischer Weise „/dev/volumengroup/lvm-name“