

# SAP: Datenkrake und Administratorschreck

Jochen Hein  
Auf der Fittell 18  
53347 Alfter  
Deutschland  
<jochen@jochen>

## 1 Einleitung

SAP Software ist in vielen Unternehmen und Rechenzentren im Einsatz. Dabei ist SAP immer mehr zur zentralen Datendrehscheibe in Unternehmen geworden - mit den entsprechenden Anforderungen an die Administratoren.

Im Rahmen der Arbeitsteilung in vielen Rechenzentren ist es auch für Unix- oder Datenbank-Administratoren sinnvoll, Grundverständnis der Architektur eines SAP Systems zu haben. Im Vortrag möchte ich verschiedene Begriffe aus dem SAP-Umfeld genauer erläutern und Zusammenhänge aufzeigen:

- SAP-System: Identifikation durch SAP-System-ID (kurz: SID)
- SAP-Systemlandschaft (oder Transportlandschaft)
- ABAP- und Java-Systeme: Einsatzszenarien
- Zentral- und Dialoginstanzen, Lastverteilung im/für SAP
- Prozesse eines SAP-Systems: Monitoring eines SAP-Systems
- HTTP- und SMTP-Zugriffe auf bzw. von SAP Systemen
- DB-Schnittstelle eines SAP-Systems
- Spezielle Anforderungen an Unix-Systeme (z.B. Dateirechte)
- Tipps und Tricks für Administratoren

Damit soll es einem Unix-Administrator möglich sein, sinnvoll und erfolgreich mit SAP-Administratoren zusammen zu arbeiten. Im Rahmen des Vortrages möchte ich auch auf Konfliktfelder im Rahmen der SAP-Administration eingehen, z.B. Fragen wie:

- Warum braucht der SAP-Admin root-Rechte auf dem System?
- Kann man nicht SAP-Software paketieren?
- Welche Rechte braucht der SAP-Admin in der Datenbank?
- Warum hält man sich besser an die Vorgaben der SAP?

## 2 Begriffe

### 2.1 SAP-System-ID

Ein SAP-System wird mit der System-ID identifiziert. Dies ist ein dreistelliges Kürzel, das mit einigen wenigen Einschränkungen frei wählbar ist. Die Einschränkungen sind im Installationshandbuch dokumentiert, so darf z.B. SAP nicht verwendet werden. Wenn man abstrakt über SAP-Systeme spricht, dann sagt man auch <SID> oder <sid> (für die SID in Kleinbuchstaben).

In einer Transportlandschaft muss die SID eindeutig sein – bei großen Outsourcern kann es aber durchaus vorkommen, dass mehrerer Kunden ein System “PRO” haben.

Die <SID> taucht an vielen Stellen im SAP-System auf – u.a. in den Namen der Betriebssystem-Benutzer <sid>adm für das SAP-System und db2<sid> oder ora<sid> für die Datenbank.

## 2.2 SAP-Transportlandschaft

Die Transportlandschaft besteht meist aus einem Entwicklungs-, einen Integrations- und dem Produktionssystem. Mit den SAP-Transporten können Programme und Daten zwischen den Systemen transportiert werden, I.d.R. Vom Entwicklungssystem über die Integration in die Produktion. Damit werden die Systeme nachvollziehbar mit Anpassungen versorgt. Im SAP heißt der Komplex dann Korrektur- und Transportwesen (CTS). Technisch ist die Landschaft häufig durch ein gemeinsam verwendetes Verzeichnis /usr/sap/trans realisiert.

## 2.3 SAP-Transportlandschaft

Die Transportlandschaft besteht meist aus einem Entwicklungs-, einen Integrations- und dem Produktionssystem. Mit den SAP-Transporten können Programme und Daten zwischen den Systemen transportiert werden, I.d.R. Vom Entwicklungssystem über die Integration in die Produktion. Damit werden die Systeme nachvollziehbar mit Anpassungen versorgt. Im SAP heißt der Komplex dann Korrektur- und Transportwesen (CTS). Technisch ist die Landschaft häufig durch ein gemeinsam verwendetes Verzeichnis /usr/sap/trans realisiert.

## 2.4 ABAP und/oder Java?

Ursprünglich war das SAP R/2-System in S/360-Assembler implementiert. Für Reports wurde dann eine Reporting-Sprache entwickelt, der ABAP (Allgemeiner Berichts-Aufbereitungs Prozessor, oder Y-ABAP, da die Assembler-Makros dafür mit Y begannen).

Daraus wurde der ABAP/III entwickelt, eine Hochsprache, die dann in den Y-ABAP übersetzt wurde, die aber immer noch nur zur Listenerstellung taugte.

Mit dem R/2-Release 4.3 wurde der ABAP/4 dann zur Advanced Business Application Programming Language, mit der auch komplexe Transaktionen implementiert werden konnten. Mit der Einführung von R/3 wurden alle SAP-Funktionen neu in ABAP entwickelt, teilweise wurden Programme auch für R/2 und R/3 gemeinsam verwendet.

In vielen Firmen wurde Know-How zur ABAP-Entwicklung aufgebaut oder externe Berater beschäftigt. Die Nachfrage nach ABAP-Programmierern war hoch, und damit auch die Kosten. Mit dem Java- und Internet-Hype war der Eindruck, dass diese Technologie preiswerter und flexibler sein könnte. Heute bestehen viele SAP-Module aus einer Kombination von ABAP- und Java-Programmen, was aus meiner Sicht im wesentlichen die Komplexität im Betrieb vergrößert.

## 2.5 SAP-Instanzen

Ein SAP System besteht aus einer Datenbank, die in der Regel die <SID> als Datenbank-ID verwendet. Die Applikationslogik steckt in der Zentral- bzw. Dialoginstanz. Die Zentralinstanz stellt dabei Dienste bereit, die im SAP-System nur einmal vorhanden sind, insbesondere den Message- und den Enqueue-Server. Dialoginstanzen dienen dazu die Systemlast auf mehrere Rechner verteilen zu können.

Die SAP-Prozesse laufen unter dem Unix-Benutzer <sid>adm bzw. Windows-Benutzer SAPService<SID>. Für die Datenbank wird z.B. ora<sid> für Oracle und db2<sid> bei DB2 UDB verwendet. Ein SAP Basis Administrator benötigt Zugriff auf beide Benutzer.

Bei der Installation mittels SAPINST werden die notwendigen Benutzer angelegt, sofern diese noch nicht vorhanden sind. Daher muss SAPINST als root gestartet werden bzw. Unter Windows als lokaler Administrator.

Zu einer Instanz gehören zwei Programme, die häufig setuid-root sind: saposcol zur Abfrage von OS-Performance-Daten und icmbnd zum Binden an Ports < 1024. Daher benötigt der SAP-Admin auch root-Rechte, wenn der SAP-Kernel aktualisiert wird – und den gibt es praktisch alle paar Wochen neu..

## 2.6 Prozesse im SAP-System

In einem ABAP-System gibt es zwei zentrale Prozesse, die nur einmal vorhanden sind: Der Message-Server und der Enqueue-Server. Der Message-Server hat eine Übersicht über die zum SAP-System gehörenden Instanzen und kann die Last zwischen diesen Instanzen verteilen. Der Enqueue-Server verwaltet die SAP-interne Sperrtabelle. Früher zeichneten diese beiden Prozesse die Zentralinstanz des Systems aus. Heute kann man diese Prozesse “Standalone” betreiben – das ist insbesondere in HA-Umgebungen interessant. Der Restart einer kompletten Instanz kann durchaus längere Zeit benötigen und durch das verwendete Shared-Memory auch mal fehlschlagen. Message- und Enqueue-Server sind recht kleine Programme, die sehr schnell neu gestartet werden können – und die Dialoginstanzen laufen durch.

### 2.6.1 Prozesse in einer ABAP-Instanz

Innerhalb einer Instanz werden die Work-Prozesse durch das Programm disp+work verwaltet. Dieses Programm hat zwei Funktionen:

- Der Dispatcher verteilt die Benutzer-Requests auf die Work-Prozesse.
- Die Work-Prozesse arbeiten die Requests ab.

In einem SAP-ABAP-System gibt es verschiedene Typen von Work-Prozessen:

- Dialog: Hier werden die Requests der Online-Benutzer bearbeitet. Dabei wird der Kontext eines Anwenders bei “Enter” in den entsprechenden Work-Prozess “hineingerollt” und nach Anzeige des nächsten Bildschirmes wieder herausgerollt. Danach steht der Workprozess den anderen Anwendern zur Verfügung und beim nächsten “Enter” kann ein anderer Work-Prozess verwendet werden.
- Background (Batch): Hier werden SAP-Jobs abgearbeitet. Diese werden in der Transaktion SM37 verwaltet und können ABAPs oder Betriebssystem-Kommandos (als <sid>adm) aufrufen.
- Verbucher: Der Verbucher nimmt die durch die Anwender veranlaßten Änderungen und schreibt diese in die Datenbank. Dieses Verfahren kommt noch aus der Mainframe-Zeit: Hier dauerte die Datenänderung so lange, dass diese in den Hintergrund ausgelagert wurde. Den Status des Verbuchers sieht man in der Transaktion SM13.
- Spool: Hier werden Druckaufträge für die Ausgabegeräte aufbereitet.

Zu einer Dialoginstanz gehört außerdem noch ein Gateway, das für RFC-Aufrufe (Remote Function Call) verwendet wird. SAP verwaltet ein internet “Systemlog”, dafür gibt es je Instanz einen Sendeprozess (rslgsend, per Symlink als se.sap<SID> gestartet) an das zentrale Systemlog und in der Zentralinstanz einen entsprechenden Empfangsprozess (rslgcoll, per Symlink als co.sap<SID> gestartet).

Für System-Monitoring und Zugriff auf OS-Performance-Daten verwendet SAP den saposcol (bei dem neuesten Releases abgelöst durch hostcontrol).

Von außen kann man auch ein wenig in die Instanz schauen, indem man sich als <sid>adm anmeldet, in das Profil-Verzeichnis wechselt und dann das Programm dpmon aufruft:

```
cdpro
dpmon pf=<SID>_<Instanz-Name>_<hostname>
```

### 2.6.2 Eine Java-Instanz

Eine Java-Instanz besteht aus einem Dispatcher und Server-Knoten. Der Dispatcher nimmt wieder die Requests entgegen und verteilt diese auf den Server-Knoten.

Welchen Status die einzelnen Prozesse in der Java-Instanz haben kann man in älteren Versionen mittels jcmom sehen. Als <sid>adm:

```
cdpro
jcmom pf=<SID>_<Instanz-Name>_<hostname>
```

### 2.6.3 SAP Management-Konsole

Zu jeder Instanz gehört ein sog. SAPstart-Service, der auch dann läuft, wenn die Instanz gestoppt ist. An diesen kann man sich mittels Web-Browser anmelden und den Status von Instanzen ansehen und ändern. Als URL verwendet man `http://<host>:5<Instanz-Nr>13`. Bei der Instanz 00 also den Port 500013. In der Java-Anwendung können dann auch andere SAP-Systeme und -Instanzen eingebunden werden.

## 2.7 SAP im Dateisystem

Zu einem SAP-System gehören eine Reihe von Verzeichnissen und Dateien. Hier eine Übersicht über die wichtigsten Verzeichnisse und der Aliase des Benutzers `<sid>adm`, mit denen man schnell in das Verzeichnis wechseln kann:

- `/usr/sap/trans`: Das Transport-Verzeichnis
- `/sapmnt/<SID>/profile`: Die Profile des SAP-Systems (Anzahl Workprozesse, Größe der Hauptspeicherbereiche, verschiedene Parameter), Alias: `cdpro`
- `/sapmnt/<SID>/global`: Dieser Speicherbereich ist auf allen Instanzen (z.B. Via NFS) verfügbar. Hier werden Job-Logs u.ä. Abgelegt., Alias: `cdglo`
- `/sapmnt/<SID>/exe`: Die Betriebssystem-Programme des SAP-Systems, Alias: `cdexe`
- `/usr/sap/<SID>` Hier liegen die Daten zum SAP-System
- `/usr/sap/<SID>/DVEBMGS<##>`: Die Daten zur Zentralsinstanz ABAP, Alias: `cdD`
- `/usr/sap/<SID>/D<##>`: Die Daten zur Dialoginstanz ABAP
- `/usr/sap/<SID>/D<##>/work`: Das Work-Verzeichnis.
- `/usr/sap/<SID>/SCS<##>`: System-Central-Services – Zentralinstanz Java,
- `/usr/sap/<SID>/JC<##>`: “Zentralinstanz Java”, Alias: `cdJC`
- `/usr/sap/<SID>/J<##>`: Dialoginstanz Java, Alias: `cdJ`

Je nach verwendetem Datenbanksystem finden sich die Datenbank-Dateien in unterschiedlichen Verzeichnissen. Hier als Beispiel ein System auf DB2-Basis:

- `/db2/<SID>`: Hier findet man alles zur Datenbank
- `/db2/<SID>/sapdata*`: Die Datencontainer der Datenbank

Bei unseren Systemen haben wir alle Dateisysteme, die zu einem SAP-System gehören in einer Volume-Group zusammen gefasst. Damit sind wir in der Lage, mit dem gesamten SAP-System umzuziehen, ohne Backup-Restore oder aufwändige Import-/Exports der Volume-Groups zu machen. Was aber in der Standard-Installation bleibt ist, dass das System ja den physischen Hostnamen in den Profilen und verschiedenen Tabellen verwendet – das bedeutet einiges an Nacharbeiten.

Bei HACMP-Clustern oder bei modernen Installationen kann man jedoch eine zusätzliche IP-Adresse (z.B. Per IP-Alias) auf das System bringen und einen Hostnamen zum SAP-System verwenden (z.B. `<SID>.example.com`). Wenn dann die DNS-Auflösung vorwärts und rückwärts korrekt ist, dann kann man das SAP-System auf diesen virtuellen Hostnamen installieren, den man dann auch beim Umzug mitnimmt.

Wenn man dann noch die User-IDs zentral verwaltet, dann ist der Umzug von SAP-Systemen nicht mehr aufwändig – SAP hat sogar mit dem “Adaptive Computing Controller” eine Software, die das unterstützt.

## 2.8 DB-Schnittstelle

SAP-Systeme können auf verschiedenen Datenbanksystemen betrieben werden. Dazu hat SAP eine eigene Datenbankschnittstelle entwickelt. Solange ABAP-Programme nur über diese Schnittstelle auf die Datenbank zugreifen sind diese zwischen den verschiedenen DB-Systemen portabel. Der ABAP-Entwickler kann auch native-SQL verwenden – in der Regel ist das für Anwendungen aber nicht notwendig und praktikabel.

Der SAP-Administrator wird aber um die Verwendung der DB-Tools (z.B. `Sqlplus` oder `db2 Command Line`) nicht herumkommen. Man kann sich sogar viele Dinge stark vereinfachen, wenn man per SQL auf bestimmte Daten zugreifen kann. SAP hat den Standard-Benutzer `SAP*` - dessen Passwort wird bei Installationen direkt abgefragt und gesetzt. Wenn man aber in der Tabelle `T000` einen neuen Mandanten anlegt, so gibt es dort noch keine Benutzer. Dann wird intern `SAP*` mit dem Passwort `PASS` verwendet (wenn der Profil-Parameter `login/no_automatic_sapstar` auf 0 sitzt). Wenn man nun keinen Zugang zu

einem Mandanten mehr hat, so kann man den Profil-Parameter setzen und den SAP\* "schiessen" - löschen oder umbenennen.

## 2.9 HTTP- und SMTP-Zugriffe auf bzw. von SAP-Systemen

Die ersten Versionen von SAP R/3 verwendeten TCP/IP mit eigenen Protokollen DIAG für SAPGUI und RFC für Remote Function Calls. Mit dem Internet-Hype Ende der 90er Jahre sollten auch SAP-Systeme HTTP und SMTP sprechen lernen. Nach den "Umwegen" Internet-Transaction-Server (ITS) und SAPconnect können heute SAP-Systeme direkt Internet-Protokolle anbieten. Dazu gibt es im ABAP-System den Internet Connection Manager (Transaktion SMICM) und das Internet Connection Framework (Transaktion SMICF). In diesen Transaktionen wird festgelegt, an welche Ports sich das SAP-System bindet und wie eingehende Requests verarbeitet werden. Die Einstellungen dazu sind oft nicht ganz durchsichtig, aber alles in allem funktioniert das gut.

## 3 Spezielle Anforderungen an Unix-Systeme

Bei älteren SAP-Systemen gab es zum Installationshandbuch die "OS-Dependencies", in denen die notwendigen Anpassungen dokumentiert waren. Häufig betrifft das die folgenden Parameter:

- Anzahl der Prozesse für Benutzer
- maximale Dateigrößen oder Memory-Größen
- umask der Benutzer
- Größe des Shared-Memories

Hier haben wir für unsere Systeme ein Standard-Image erstellt, das auch notwendige Tools wie sudo, perl u.ä. enthält. Damit ist man recht schnell in der Lage ein System neu aufzubauen – mein persönlicher Rekord liegt inklusive SAP-Kopie samt Datenbank bei 30 Minuten.

## 4 Tipps und Tricks für Unix-Administratoren

Viele Unix-Administratoren betrachten ein SAP als "eine ganz normale Anwendung". In vielen Fällen ist das nicht ganz falsch, aber in etlichen Fällen stellt SAP besondere Anforderungen. Das kann bedeuten, dass bestimmte Patches im Betriebssystem oder in der Datenbank eingespielt werden müssen, oder bestimmte User- oder Datei-Rechte notwendig sind. Gerade in Unternehmen, die gerade beginnen eine Unix-Security-Policy zu implementieren, müssen für SAP-Systeme besondere Ausnahmen definiert werden.

Das betrifft z.B. das Home-Verzeichnis des <sid>adm, damit z.B. das Monitoring-Tool dort die Profile auslesen kann. Auch verwendet SAP sehr viele TCP-Ports, die man nicht so einfach schliessen kann.

Zur uns hat sich bewährt, dass wir alle Systeme der Landschaft versuchen einheitlich zu betreiben. Wenn uns also irgendwo ein Fehler auffällt, dann prüfen wir das stets auf der gesamten Landschaft und implementieren einen entsprechenden Monitor. Damit fallen uns viele Probleme direkt bei der Installation eines neuen Systems auf.

Um ein SAP-System zu überwachen kann man verschiedene Tools einsetzen:

- SAP CCMS: Das Computing Center Monitoring System ist im SAP implementiert und kommt mit jedem ABAP-System mit. Es kann insbesondere im SAP viele Parameter überwachen – ich persönlich finde es allerdings ein wenig unhandlich.
- Zu Nagios gibt es ein Modul, das Parameter aus dem CCMS auslesen kann und in eine zentrale Umgebung bringen kann.
- Spezielle Module für kommerzielle Systeme wie BMC-Patrol oder Tivoli. Ich habe in den letzten Jahren sehr gute Erfahrungen mit dem Tool Xandria der Fa. Syslink gemacht.

Die wichtigsten Erfahrungen der vergangenen Jahre:

- Möglichst nah am SAP-Standard bleiben
- Abweichungen möglichst einfach halten und gut dokumentieren
- In der Systemlandschaft einheitlich arbeiten
- Was nicht maschinell geprüft wird, das wird voneinander abweichen. Als: Monitoring als Überwachung der "Baseline"
- Systemverfügbarkeit kann durch Monitoring deutlich verbessert werden

## 5 Weitere Informationsquellen für den Unix-Admin

Eine ausgezeichnete Informationsquelle für alle SAP-Anwender sind die Installations-Anleitungen unter <http://service.sap.com/instguides> sowie die sogenannten SAP-Hinweise oder SAP-Notes zu den verschiedenen SAP-Komponenten unter <http://service.sap.com/notes>. Zum Zugang benötigen Sie eine Kennung im SAPNet, die man bei den SAP-Kollegen bekommen kann.

Auch wenn man keinen Zugriff auf das SAPnet hat, so kann man – nur mit einer e-Mail-Adresse – Zugang zum SAP-Developer-Network (SDN, <http://www.sdn.sap.com>) bekommen. Da gibt es Blogs, Whitepapers und Foren, die oft hilfreiche Tipps enthalten.

### Referenzen

[1] <http://www.sap.com> ist die Internet-Seite der SAP AG

[2] SAPnet: <http://service.sap.com>

[3] SAP Developer Network (SDN): <http://www.sdn.sap.com>

[4] <http://www.jochen.org/~jochen/sap-r3/download.html> is a great site to visit

[5] SAP PRESS: verschiedene Bücher