

# Der DBA als Werkzeugmacher?

## Einblicke in die Toolbox eines DBA Teams

**Martin Berger**  
**Hutchison Drei Austria GmbH**  
**Wien**

### Schlüsselworte

Tools, APEX, sql, shell

### Einleitung

In jedem DBA Team (und wohl auch in anderen technischen Teams) sammeln sich über die Jahre verschiedenste kleinere und größere Werkzeuge an, um widerkehrende, oder auch nur sporadisch auftretende Arbeit zu erleichtern, verbessern oder zu beschleunigen. Hier werden einige solche Tools eines konkreten Teams und deren Hintergründe und Werdegang vorgestellt. Es geht nicht um „Best Practice“ sondern vielmehr soll gezeigt werden, dass es Sinn macht und wertvoll ist, auf spezifische Bedürfnisse individuell zu reagieren.

### Von Excel zu APEX

Wie wohl in vielen anderen Firmen begann die Verwaltung von Datenbanken (und auch vielen anderen Systemen anderer Teams) in einfachen excel-Sheets. Die Teams waren klein und die Anzahl der zu verwaltenden Systeme überschaubar. Natürlich wurden nur die wichtigen Informationen im Excel-Sheet gespeichert. Und doch kam es nach einiger Zeit zu etwas Wildwuchs, wie Abb1. Zeigt.

OracleServerOverview											
	A	B	J	K	L	M	N	O	P	Q	R
	Name	Name	Dev Prod Test etc.	Oracle Owner	Listener Owner	Listener name	SLA	TABLESPACES Monitored ?	Retention Policies (Browse/Retention)	IP	Port
5	oneracx1p (isprac1) /dev/md/rmiddle/rdsk/d126	MDDL1	Prod.	oracle	oracle	LISTENER_MDDL1		YES	3/3	10.238.251.34	3112
6		APPL1	Prod.	oracle	oracle	LISTENER_APPL1	ABB	YES	3/3	10.238.251.34	3054
7		RMAN1	Prod.	oracle	oracle	LISTENER_RMAN1		YES	3/3	10.238.251.34	3080
8						LISTENER_DWHCONT					
9	oneracx2p (isprac2) /dev/md/rmiddle/rdsk/d126	MDDL2	Prod.	oracle	oracle	LISTENER_MDDL2		YES	3/3	10.238.251.35	3112
10		APPL2	Prod.	oracle	oracle	LISTENER_APPL2	ABB	YES	3/3	10.238.251.35	3054
11		RMAN2	Prod.	oracle	oracle	LISTENER_RMAN2		YES	3/3	10.238.251.35	3060
12						LISTENER_DWHCONT					
13	oneicx1p	ICTPRD	Prod.	oracle9	oracle9	LISTENER	BBB	YES	3/84	10.238.2.3	1521
14	elpfundb1	ORP	R/T	e40gr2	e40gr2	LISTENER_ORP		YES	3/3	10.238.2.95	1521
15		NETX	R/T	e40gr2	e40gr2	LISTENER	---	YES	3/3	10.238.2.95	1521
16	dipianrac1	GENDB011	Prod.	oracle	grid	billclu-scan:1521		yes	3/84	billclu-scan 10.238.2.17[4-6]	1521
17		PROV/PROD1	Prod.	oracle			BBB	yes	3/84	billclu-scan 10.238.2.17[4-6]	1521
18		MEDSIDP1	Prod.	oracle			yes	3/84	billclu-scan 10.238.2.17[4-6]	1521	

Abb. 1: Excel

Auch wenn das Excel-Sheet für den Menschen brauchbare Informationen zu enthalten scheint, ist jede Art von darauf aufbauender Automatisierung, ja sogar einfachste Konsistenzchecks, nahezu unmöglich.

Hier einige Beispiele:

- Die Spalten A und B haben beide den Titel „Name“. Spalte A ist ursprünglich der hostname, Spalte B der Instanzname.
- Spalte A beinhaltet manchmal nicht nur den hostnamen sondern auch einen virtuellen Namen sowie spezifische Details für die Cluster Konfiguration.
- Die Spalte „Listener name“ beinhaltet manchmal den Listener, über den die jeweilige Instanz erreichbar ist – oder einen SCAN-DNS Eintrag samt Port.
- Die Spalte „Tablespaces Monitored?“ beinhaltet „Yes“ in verschiedenen Schreibweisen.
- Die Spalte „Retention Policies (Browse/Retention)“ hat in einer Spalte unterschiedliche Werte, durch einen Separator getrennt
- In der Spalte „IP“ sind IPs, DNS Namen und IP-Bereiche gemischt

Diese Beispiele zeigen, dass ein Excel-Sheet, das für Menschen noch wunderbar zu verwenden ist, für die maschinelle Bearbeitung oder auch nur Verarbeitung höchst aufwändig sein kann.

Aufgrund der wachsenden Aufgaben und dem Drang zu Automatisierung wurde im Team entschlossen, die Daten in eine Oracle Datenbank zu speichern. Als Frontend wurde APEX ausgesucht, da das Team mehrere Vorteile darin sah: APEX ist ein Standardprodukt von Oracle, was den Support und zukünftige Upgrades erleichtern sollte. Es entstehen keine zusätzlichen Lizenzkosten (die Lizenzen für die DB waren schon anderwärtig abgedeckt, und der Ressourcenaufwand minimal). Die Installation erschien relativ einfach. Und auch der Know-How Gewinn bei diesem internen Projekt erschien von Vorteil für das Team und die Firma.

Das alles hat über einige Iterationen und Anpassungen zu einer kleinen APEX Applikation geführt:

Welcome: BERGERMA Logout

Db Name	Version	Life Cycle Status	System Usage	Tag	Cluster Name	Servers	Ora Instances
APM0001	11.2.0.4.8	InProduction	Production	h3genv,b000c,bs-ok	cr001	av0001p,av0002	APM0001,APM0002
APM0002	11.2.0.4.8	InProduction	Production	h3genv,prp000c	cr002	av0003p,av0004	APM0003,APM0004
APM0003	11.2.0.4.8	InProduction	Test/Development	h3genv,prp000c	cr003	av0005p,av0006	APM0005,APM0006
APM0004	11.2.0.4.8	InProduction	Test/Development	h3genv,prp000c,erppw	cr004	av0007p,av0008	APM0007,APM0008
APM0005	11.2.0.4.8	InProduction	Test/Development	h3genv,m000c,erppw,bs-ok	cr005	av0009p,av0010	APM0009,APM0010
APM0006	11.2.0.4.8	InProduction	Test/Development	h3genv,b000c,bs-ok	cr006	av0011p,av0012	APM0011,APM0012
APM0007	11.2.0.4.8	InProduction	Test/Development	h3genv,prp000c,erppw,bs-ok	cr007	av0013p,av0014	APM0013,APM0014
APM0008	11.2.0.4.8	InProduction	Test/Development	h3genv,b000c,erppw	cr008	av0015p,av0016	APM0015,APM0016
APM0009	11.2.0.4.8	InProduction	Test/Development	h3genv,prp000c,erppw	cr009	av0017p,av0018	APM0017,APM0018
APM0010	11.2.0.4.8	InProduction	Test/Development	h3genv,prp000c,erppw,bs-ok	cr010	av0019p,av0020	APM0019,APM0020
APM0011	11.2.0.4.8	InProduction	Test/Development	h3genv,prp000c,erppw,bs-ok	cr011	av0021p,av0022	APM0021,APM0022
APM0012	11.2.0.4.8	InProduction	Test/Development	h3genv,prp000c,erppw,bs-ok	cr012	av0023p,av0024	APM0023,APM0024
APM0013	11.2.0.4.8	InProduction	Test/Development	h3genv,prp000c,erppw,bs-ok	cr013	av0025p,av0026	APM0025,APM0026
APM0014	11.2.0.4.8	InProduction	Test/Development	h3genv,prp000c,erppw	cr014	av0027p,av0028	APM0027,APM0028
APM0015	11.2.0.4.8	InProduction	Test/Development	h3genv,prp000c,bs-ok,erppw	cr015	av0029p,av0030	APM0029,APM0030

Abb. 2: APEX - Überblick

Hier sind die einzelnen Spalten und Zeilen – wie in einer relationalen Datenbank gewohnt) klar getrennt und haben eindeutige Inhalte. Es sollte klar sein, dass auch eine einfache Verwaltung von Datenbanken mehr als eine Tabelle erfordert, und so die verschiedenen Anzeigen / Reports immer nur eine gewisse Sichtweise darstellen. Dafür ist für jeden der das Schema versteht und etwas SQL beherrscht jede beliebige Auswertung, auch wenn sie im APEX nicht als Report zur Verfügung steht, möglich.

Der Teil des Schemas, der hinter dem Report in Abb. 2 steht sieht vereinfacht so aus:

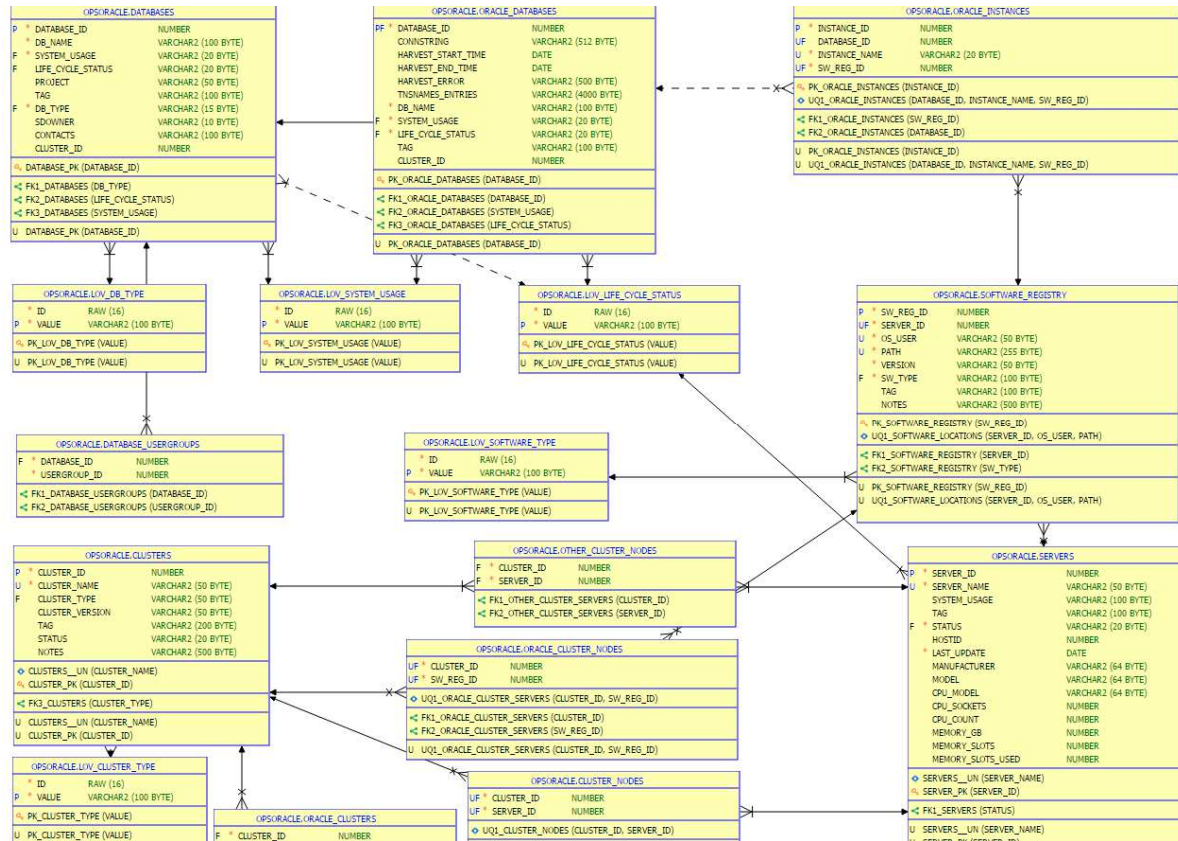


Abb. 3: Schema - vereinfacht

Dieses Schema ist über die Zeit weit weniger statisch, als es erscheinen mag. Neue Tabellen und Relationen kommen genauso regelmäßig dazu, wie Tabellen erweitert werden und Objekte auch wieder verschwinden. Heute würden dafür Schlagwörter wie „agil“ verwendet werden.

Andere Views im APEX zeigen zum Beispiel, wo welche Datenbank Software installiert ist:

Databases Servers Clusters **SoftwareRegistry** ChangeLog

Q Go 1. Primary Report Actions Create

Server Name like 'av[redacted]p'

Server Name	Sw Type	Version	Path
av[redacted]	Oracle RDBMS EE 64bit	11.2.0.4.5	/[redacted]product/rdbms_112045Ja
av[redacted]p	Oracle GI Cluster 64bit	12.1.0.2.5	/[redacted]grid_12102
av[redacted]p	Oracle Client 64bit	11.2.0.3.4	[redacted]product/client_11203
a[redacted]p	Oracle RDBMS EE 64bit	11.2.0.4.8	/[redacted]product/rdbms_112048Ja
av[redacted]	Oracle RDBMS EE 64bit	11.2.0.4.8	/[redacted]product/rdbms_112048Ja
[redacted]p	Oracle RDBMS EE 64bit	12.1.0.2.161018	/[redacted]product/rdbms_12102_Oct16Ja
av[redacted]p	Oracle RDBMS EE 64bit	12.1.0.2.161018	/[redacted]product/rdbms_12102_Oct16Ja

Abb. 4: SoftwareRegistry

Und auch Informationen zu Clustern sind eigens dargestellt:

Databases Servers **Clusters** SoftwareRegistry ChangeLog

Q Go Actions Create

Tag like '%OEL7%'

Cluster Name	Cluster Type	Cluster Version	Servers	Tag	More Info
[redacted]	Oracle Clusterware	12.1.0.2.170117 JAN2017 BP	[redacted]	h3genv,OEL7.3,b52-dc,emagent	SCAN name: c[redacted] SCAN IP Range: [redacted]/29 Nodes: oracle@[redacted] oracle@[redacted]
[redacted]	Oracle Clusterware	12.1.0.2.170117 JAN2017 BP	a[redacted], a[redacted]	h3genv,OEL7.3,mgc-dc,emagent	SCAN name: c[redacted] SCAN IP Range: [redacted]/29 Nodes: oracle@a[redacted] oracle@a[redacted]
[redacted]	Oracle Clusterware	12.1.0.2.161018 OCT2016 PSU	a[redacted], a[redacted]	h3genv,OEL7.2,mgc-dc,emagent	SCAN name: c[redacted] SCAN IP Range: [redacted]/29 Nodes: oracle@a[redacted] oracle@a[redacted]
[redacted]	Oracle Clusterware	12.1.0.2.170418 APR2017 BP	a[redacted]	h3genv,OEL7.3,mgc-dc,emagent	SCAN name: c[redacted] internal SCAN IP Range: [redacted]/29 Nodes: oracle@a[redacted] oracle@a[redacted]

Abb. 5: Cluster

Neben all den schönen Prinzipien zu Tabellen und Relationen zeigt sich hier ein pragmatischer Ansatz sehr gut: In der Spalte „Servers“ sind alle Server, aus denen der Cluster besteht, zusammengefasst. Die Spalte „Tag“ ist vom Typ VARCHAR2 und hat unterschiedlichste Schlüsselwörter, durch Beistrich getrennt. Die Spalte „More Info“ enthält sogar einen mehrzeiligen, strukturierten Text.

Dies alles ist erlaubt und gültig, obwohl wir uns in einer relationalen Welt bewegen: Der Report selbst ist nur eine View und dient der Veranschaulichung. Mehrere Servernamen in einer Spalte und sogar strukturierter Text helfen den DBAs, alle notwendigen Informationen schnell zu erfassen.

Die Spalte „Tag“ ist eine Chimäre für alle Informationen, die strukturiert notwendig sein könnten (meistens als Filter) die aber nur von begrenzter Lebensdauer sind oder wo sich schlicht der Aufwand zusätzlicher Tabellen derzeit nicht auszahlt. Falls es jemals als notwendig erachtet werden sollte, wird das Schema entsprechend adaptiert.

Ein spezielles Feature in dieser APEX Applikation ist das Logbook:

LogBookEntry

When 2017-05-15 10:31:10

Object edwp06

\*Who EDWPO6 oracle db on...  
EDWPC... oracle db on...

\*What

resources - storage, memory, processes, limits...

errors - database errors (600, 7445, 4031, ...), bugs

objects - create, drop db objects...

configuration - change in db/instance parameters, orahome change, listener

security - permissions, users, grants

**Flags**

knowhow - database specific docu, policies exceptions, things good to know about

SR/WO - action is result of SR or WO

INC - action is result of INC

AGR - action is result of AGREement of involved parties (verbal or email)

DBA - action is result of brain activity of DBA

Abb. 6:Logbook

Es ermöglicht es allen Teammitgliedern, einfach alle Aktivitäten auf beliebigen Objekten zu dokumentieren. Die Uhrzeit wird automatisch eingefügt, Das Objekt wird bei der Eingabe aus der Liste der Datenbanken, Server und Cluster dynamisch vorgeschlagen. Der Fließtext Ist einfach die Beschreibung der Aktivität. Zusätzliche Flags helfen, ein wenig Struktur zu schaffen.

Einträge selbst können so aussehen:



This query returns more than 1,000 rows, please filter your data to ensure complete results.



	When	Object Name	Who	Flags	
	2017-05-12 18:08:02	E [redacted] oracle db on at [redacted] [redacted] av [redacted]	bergerma	DBA	BEGIN DBMS_WORKLOAD_REPOSITORY.modify_snapshot_settings( retention =>288000, -- 200 days interval => 15); -- Minutes. END; /
	2017-05-11 11:08:38	[redacted] oracle db on [redacted] [redacted]	bergerma	SRWO	-- WO0000000032908 -- username in quotes as it begins with a number. -- this leads to case-sensitive username/profile (!) create profile [redacted] limit sessions_per_user 50;  CREATE USER "[redacted] [redacted] app" IDENTIFIED BY "[redacted] [redacted]" DEFAULT TABLESPACE [redacted] TEMPORARY TABLESPACE [redacted] PROFILE "[redacted] [redacted] app" PASSWORD EXPIRE ACCOUNT UNLOCK;  GRANT CONNECT TO [redacted] ALTER USER [redacted] DEFAULT ROLE CONNECT;  GRANT CREATE TABLE TO [redacted];

Abb. 7: Logbook - Details

Hier zeigt die simple Volltextsuche von APEX ihre Stärke. Zu jedem Schlagwort, oder auch einer Kombination wie Datenbankname und JAHR bzw. YYYY-MM lassen sich Einträge finden. Natürlich ist die Qualität und Vollständigkeit der Einträge entscheidend. Besonders, da alle Mitglieder des Teams den Nutzen direkt erleben gab es hier nie Probleme.

Diese APEX Applikation existiert schon recht lange. Obwohl sie kein offizielles Tool ist, hat sie schon eine CMDB überlebt und ist automatisierte Datenquelle für deren Nachfolgeprodukt.

## James!

In jedem Team gibt es wiederkehrende Aufgaben und Tätigkeiten. Für viele Menschen erscheinen sie recht schnell langweilig und uninteressant. Das kann dazu führen, dass sie nicht mit der entsprechenden Sorgfalt und Konzentration ausgeführt werden, wodurch sich Fehler einschleichen können, die spätere Betriebsabläufe empfindlich stören.

Um diese Aktivitäten automatisieren zu können, und wiederkehrende Schritte im Automaten (sic!) zu vereinheitlichen, wurde james entwickelt.

James' Zweck ist es, Scripte auszuführen. Dabei kümmert sich James um Standardfunktionalität wie logging, die Wiederaufnahme der Aktivitäten im Fehlerfall und stellt Funktionen zur Verfügung, die öfters verwendet werden können.

Ein Beispiel für logging ist hier dargestellt:

```
variable assignment: dgname=SVENCRYPTFS_ACFS  
variable assignment: grid_version=12c  
variable assignment: pattern=SVENCRYPTFS_000  
loading workflow specific variables.
```

```
Full Oracle Cluster detected. Cluster name is crsxxx with cluster nodes  
"avxxxxxt avyyyyyt", remote nodes "avyyyyyt"
```

```

====> executing workflow diskgroup_create from category database from
/zzz/james/workflows_database/diskgroup_create <====
=====> executing workflow step
/zzz/james/workflows_database/diskgroup_create/10.create_diskgroup.sh
SQL*Plus: Release 12.1.0.2.0 Production on Wed May 10 13:11:15 2017
Copyright (c) 1982, 2014, Oracle. All rights reserved.
...
...
With the Real Application Clusters and Automatic Storage Management options
=====> executing workflow step
/install/oracle/wok/james/workflows_database/diskgroup_create/20.mount_disk
group.remote.sh

```

Zu Beginn wird der spezifische Workflowname und die Kategorie zu der er gehört sowie das Verzeichnis ausgegeben. Danach werden die jeweiligen Schritte genannt. Welche Informationen ein jeder Schritt ausgibt, ist vollkommen frei.

Im Fehlerfall bricht James nicht einfach ab, sondern wartet auf eine Eingabe:

```

loading workflow specific variables.

Full Oracle Cluster detected. Cluster name is crsxxx with cluster nodes
"avxxxxxt avyyyyyt", remote nodes "avyyyyyt"

====> executing workflow rdbms_12102_Apr17BPJa from category install from
/zzz/james/workflows_install/rdbms_12102_Apr17BPJa <====

resume directory /hhh/james/rdbms_12102_Apr17BPJa/resume exists. following
step have been already executed:

20.prechecks.rdbms.12102.remote.sh

... < some more steps> ...

62.opatch.install.sh

hit Ctrl+C here and delete some files from
/hhh/james/rdbms_12102_Apr17BPJa/resume if you want to reexecute some
steps.

resume/continue the previous run? [Y(es)/(N)o]: <N>

type (N)o again to confirm that you DO NOT want to continue in the
workflow. workflow will restart from the beginning. [Y(es)/(N)o]: <N>

```

Dadurch kann der Fehler, falls möglich in einem anderen Terminal behoben werden und der jeweils fehlgeschlagene Schritt wird wiederholt. Dadurch können kleine Fehler manuell behoben werden und der gesamte Workflow muss nicht nochmals gestartet werden, was oft einen erheblichen Cleanu-Aufwand erfordern würde.

Die Standardfunktionen, welche James zur Verfügung stellt sind zum Beispiel unzählige Checks für Verzeichnisse, Kernelparameter, DNS Einträge, laufende Prozesse und vieles mehr.

Zusätzlich werden aber auch Funktionen zum adaptieren von Templates – zum Beispiel für Responsefiles – oder auch das Bearbeiten des oratab files bereitgestellt.

Mit diesem Rüstzeug haben sich über die Jahre Workflows der Kategorien Installation (binaries), Database (create, upgrade, patch, timezone-patch), clone (generic oder EBS in verschiedenen Varianten), password (müssen regelmäßig und unique geändert werden) und grid (install, patch, addnode) angesammelt.

Die Installation eines ORACLE\_HOME durchläuft ca 30 Schritte, bei denen auch alle Anpassungen an die spezifische Umgebung wie z.B. Backup oder Filesystem Berechtigungen beinhaltet sind.

Als fleißiges Mitglied im DBA Team macht James auch Einträge in das Logbook und dokumentiert seine Arbeit.

Object Name	Who	What
oracle db EBSSID06 on av31047t av31050t	james	database EBSSID06 upgraded to 12c ORACLE_HOME=/appl/oracle/product/ebs_rdbms_12102_Apr17a_1. WO0000000037816
oracle db APXSID06 on av31047t av31050t	james	patched timezone to current. after upgrade patch
oracle db APXSID06 on av31047t av31050t	james	database APXSID06 upgraded to 12c ORACLE_HOME=/appl/oracle/product/rdbms_12102_Apr17BPJa. WO0000000037818

Abb. 8: Logbook - James

## Hydra

Hydra ist ein Werkzeug zum parallelen Starten von beliebigen Jobs mit einem klaren Fokus auf Parallelität und der gleichzeitigen Vermeidung Lastspitzen.

Das Konzept ist erstaunlich simpel: In ein Input-Verzeichnis werden für jeden Job ein file erzeugt. Der filename hat das Format `job.ziel.beschreibung`. Zu jeder Klasse von JOBS muss es eine config file mit einer maximalen parallelität geben. Zusätzlich können weitere Limits wie die maximale Laufzeit (danach wird der Job beendet) die maximale Parallelität innerhalb eines Ziels oder sogar spezifische Werte für ein konkretes Ziel angegeben werden.

Hydra versucht mittels round-robin möglichst viele Jobs abzuarbeiten, aber die Last auf jedem Ziel so gering wie möglich zu halten. Aktive Jobs werden in ein runqueue Verzeichnis verschoben und die ProzessID sowie Beginnzeit und maximale Laufzeit werden dem Filenamen hinzugefügt.

Das Ergebnis jedes Jobs wird in einem Logfile gespeichert (und die Logfiles nach einigen Tagen auch gleich komprimiert und archiviert).

Hydra wird unter Anderem zur Steuerung aller Datenbank Backup Jobs verwendet. Ein Script befüllt anhand eines Regelwerks alle Jobs für Full- Incremental- und Archivelog Backups. Hydra arbeitet sie anhand der Parallelitäts-limits ab. Dadurch werden alle Backups schnellstmöglich erledigt, ohne den einzelnen Server oder aber die Backup-Infrastruktur zu überlasten.

## Kontaktadresse:

Martin Berger

Hutchison Drei Austria GmbH



Brünnner Str., 52

A-1210 Wien

Telefon: +3 660 660 88300

E-Mail [martin.berger@drei.com](mailto:martin.berger@drei.com)

Internet: <https://www.drei.at/>