



www.MemText.de

- Daniel Quathamer
danielq@memtext.de
- Meikel Bisping
mbisping@memtext.de

Supportadresse
support@superx-projekt.de

<http://www.superx-projekt.de>

Version 1.2
Stand 12.03.2014

[Lehrfilm zur Installation des COB-Moduls](#)

Sun, Sun Microsystems, Solaris, Java, JavaServer Web Development Kit, JDBC und JavaServer Pages sind eingetragene Warenzeichen von Sun Microsystems, Inc. UNIX ist ein eingetragenes Warenzeichen von X/Open Company, Ltd. Windows, WindowsNT, Win32, VBScript und Office 2000 sind eingetragene Warenzeichen von Microsoft Corp. Linux ist eingetragenes Warenzeichen von Linus Torvalds. Alle weiteren Produktnamen sind Warenzeichen der jeweiligen Hersteller.

Dieses Produkt beinhaltet Software, die von der Apache Software Foundation (<http://www.apache.org/>) entwickelt wurde.

SuperX wird unter der deutschen Variante der GPL-Lizenz von dem Land Nordrhein-Westfalen, vertreten durch die FernUniversität Hagen, diese wiederum vertreten durch die Geschäftsstelle der Initiative CampusSource bei der FernUniversität Hagen, Feithstraße 142, D-58084 Hagen vertrieben (www.campussource.de). Details zu den Lizenzbedingungen finden Sie im COB-Modul-Archiv (/lizenz.txt) oder unter <http://www.campussource.de/lizenz/>. Ergänzende Hinweise finden Sie auf der Projekthomepage unter <http://www.superx-projekt.de>.

Inhaltsverzeichnis

1 Einführung.....	5
1.1 Hintergründe.....	5
1.2 Kurzüberblick.....	7
2 Installation des COB-Moduls.....	8
2.1 Kurzanleitung	9
2.2 Ordnerstruktur und Umgebung des COB-Moduls.....	11
2.3 Entladen der COB-Daten.....	12
2.3.1 Allgemeine Parameter zum Entladen.....	12
2.3.2 Entladen mit oder ohne Gültigkeitszeiträumen	12
2.3.3 Entladen mit Shellscripten.....	13
2.3.3.1 DBMS-spezifische Konfiguration	14
2.3.3.1.1 Entladen aus Informix/ UNIX	14
2.3.3.1.2 Entladen aus Postgres/ Unix	15
2.3.3.1.3 Entladen aus Informix/ Windows	16
2.3.4 Nur beim PUSH-Verfahren: Kopieren der Rohdaten.....	16
2.4 Neuinstallation: Erzeugung der Prozeduren, Schlüssel und Datentabellen.....	16
2.4.1 Konstanten.....	17
2.5 Upgrade eines vorhandenen COB-Moduls.....	19
2.6 Laden der Daten	19
2.6.1 Allgemeines.....	19
2.6.2 Laden mit Shellscripten.....	19
2.6.3 Hochschulspezifische Transformationen.....	20
2.6.4 Separate Übernahme des Organigramms	21
2.6.5 Geldgeber zu Drittmitteln zuordnen.....	22
2.6.5.1 Geldgeber KLR (Kurz).....	23
2.6.5.2 Mittelherkunft KLR.....	24
2.6.6 Validierung KLR-Daten im Vorsystem.....	26
2.6.6.1 Validierung in COB-GX.....	26
2.6.6.1.1 Validierung von Primärbuchungen in COB-GX.....	26
2.6.6.1.2 Validierung von Sekundärbuchungen in COB-GX.....	27
2.7 Vergeben von Rechten für eingeschränkte User.....	28
2.7.1 a. Institutionsrechte.....	28
2.7.2 b. Sichtrechte.....	28
2.7.2.1 Kostenstellen.....	28
2.7.2.2 Kostenträger.....	29
2.7.2.3 Kostenarten.....	30
2.8 Planbuchungen.....	30
2.9 Anpassungen der Berichte	31
2.9.1 Berichtsblatt Kennzahlen aus der Kostenrechnung (nur relevant für NRW).....	31
2.9.2 Auslastung (Lehrangebot / Lehrnachfrage).....	32
2.10 Entfernen des COB-Moduls.....	32
3 Bestandteile des COB-Moduls.....	32
3.1 Datentabellen.....	33

3.1.1	Tabelle cob_busa: Primärbuchungen.....	34
3.1.2	Tabelle cob_vtbu: Verteilbuchungen.....	34
3.1.3	Tabelle cob_umks: Verzeichnis der Verrechnungssätze/Festpreise.....	34
3.1.4	Tabelle cob_su_imp_stud: Studierendendaten in COB.....	34
3.1.5	Tabelle cob_imp_flaeche: Flächen in der KLR.....	34
3.2	Schlüsseltabellen.....	34
3.2.1	Die Tabelle cifx.....	35
3.2.2	Die Tabellen cob_cif / cob_cifx.....	35
3.2.3	Die Tabelle cob_inst (Verzeichnis der Kostenstellen).....	35
3.2.4	Die Schlüsseltabelle cob_fikr (Verzeichnis der Kosten-/Erlösarten).....	36
3.2.5	Die Tabelle cob_proj (Verzeichnis der Kostenträger/Projekte).....	36
3.2.6	Die Tabelle cob_exkotr (Verzeichnis der externen Kostenträger).....	37
3.2.7	Die Schlüsseltabelle cob_kpum (Verzeichnis der Leistungsarten).....	37
3.2.8	Die Schlüsseltabelle cob_proj_to_inst.....	37
3.2.9	Die Schlüsseltabelle aggregierung.....	37
3.2.10	Die Schlüsseltabelle cob_drittmittel (obsolet).....	37
3.2.11	Die Schlüsseltabelle cob_stug (Verzeichnis der Studiengänge).....	38
3.2.12	Die Schlüsseltabelle cob_geldgeber.....	38
3.2.13	cob_alt_hier, cob_trees, cob_tree_cfg, cob_alt_keys.....	38
3.3	Hilfstabellen.....	38
3.3.1	cob_busa_aggr.....	39
3.3.2	cob_vtbu_aggr.....	40
3.3.3	cob_busa_vtbu.....	40
3.4	Alternative Hierarchien / Sichten.....	40
3.4.1	Prozedur sp_cob_alt_hier.....	41
3.4.2	Prozedur sp_cob_inst_hier.....	41
3.4.3	Prozedur sp_cob_ktr_hier.....	41
3.4.4	Prozedur sp_cob_fikr_hier.....	41
4	Anhang.....	43
4.1	Geldgebersichten und Drittmittel.....	43

1 Einführung

Das Berichtssystem SuperX ist ein sog. Data Warehouse, d.h. beliebig viele Datenquellen werden unter einer einheitlichen Auswertungsschnittstelle zur Verfügung gestellt. Da jede Hochschule unterschiedliche Datenquellen besitzt und nach SuperX übernehmen will, bereiten wir für jede Datenquelle ein Modul vor, z.B. ein COB-Modul oder ein SOS-Modul.

Das SuperX-COB-Modul entlädt Daten aus HISCOB (Version 5.x-11.x) und lädt sie in SuperX, um dort Abfragen im Bereich Kostenrechnung zu ermöglichen.

Das COB-Modul wurde entsprechend der Anforderungen aus dem SuperX-Projekt in Baden-Württemberg erweitert. Erste Tests des Moduls an Pilothochschulen haben gezeigt, dass der erste Aufbau von Abfragemasken bei Benutzung umfangreicher alternativer Hierarchien recht lange dauert. Außerdem wurde der Wunsch geäußert, dass ein eingestelltes Standarddatum von einer Abfrage zur nächsten beibehalten wird.

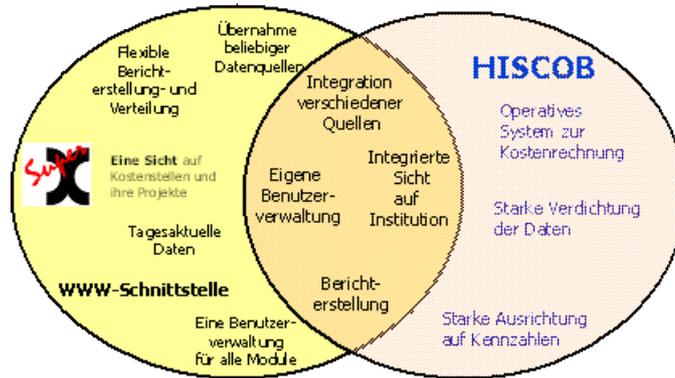
Wir werden versuchen, in den Folge-releases Verbesserungen in diesen Punkten zu erreichen.

1.1 Hintergründe

Das COB-Modul dient als WWW-basiertes Frontend für HISCOB, das Controlling-System der HIS GmbH. Es liefert Daten aus HISCOB über eine einheitliche Benutzerschnittstelle aus und kann z.B. für dezentrale User via WWW zur Verfügung gestellt werden. Die Berichte in SuperX bilden teilweise vorhandene Berichte in HISCOB nach (z.B. das Schnellinfo), teilweise werden andere Akzentuierungen vorgenommen. Die vorhandenen Abfragen dienen als Beispiele und können von den Anwendern beliebig an örtliche Gegebenheiten angepasst werden.

Sowohl SuperX als auch COB sind Systeme, in denen Daten aus anderen Systemen zusammengestellt und verdichtet werden. Zunächst stellt sich daher die Frage, welche zusätzlichen Funktionen SuperX bietet. Die folgende Abbildung stellt beide Systeme zusammen:

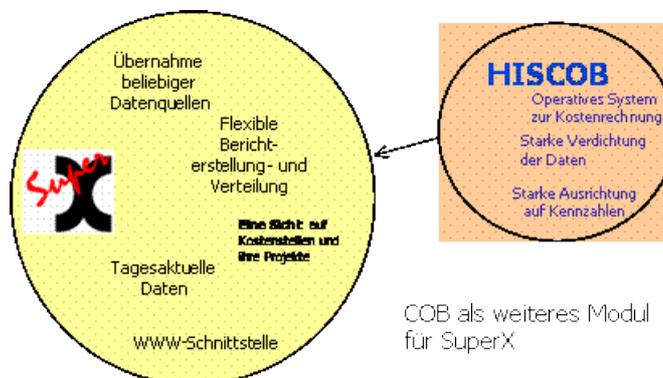
HISCOB besitzt als Controlling-Instrument eine starke Ausrichtung auf Kennzahlen, während SuperX eher deskriptiv ausgerichtet ist. Die Datenquellen für COB sind festgelegt, in SuperX können beliebige Datenquellen aufgenommen werden. Beide Systeme bieten die Möglichkeit zur Berichterstellung, SuperX auch via WWW.



Anders als in HISCOB können die Statistiken in SuperX je nach Modul stichtagsbezogen oder tagesaktuell sein, während sie in HISCOB eher stichtagsbezogen sind.

Die folgende Abbildung veranschaulicht die Integrationsmöglichkeit der beiden Systeme. Dabei wird HISCOB als zusätzliche Datenquelle in ein umfassenderes System SuperX aufgenommen.

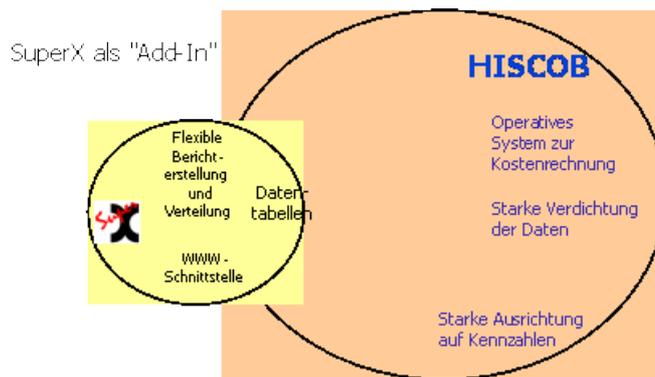
Wenn ein umfassendes Data Warehouse mit mehreren Modulen (z.B. SOS, SVA) existiert, kann COB als weiteres Modul aufgenommen werden. Ggf. müssten dann allerdings gemeinsame Schlüssel angepasst werden.



Außerdem muss ein Entladerhythmus aus COB und ein Übernehmerhythmus nach SuperX eingestellt werden.

Grundsätzlich besteht noch eine weitere, „abgeschwächte“ Integrationsmöglichkeit, bei der SuperX (d.h. Kernmodul + COB-Modul) ein zusätzliches Frontend für die COB-Datenbank wird.

SuperX als Add-In: Es liefert eigene Berichte direkt aus COB, und enthält keine anderen Module. Das SuperX-Kernmodul wird zusammen mit dem COB-Modul auf dem COB-Server (in einer getrennten Datenbank) installiert, und die Datentabellen werden direkt (d.h. ohne Entladen) aus COB übernommen.



Diese Variante wird momentan nicht durch automatisierte Skripte unterstützt.

1.2 Kurzüberblick

Die Module enthalten die wichtigsten Prozeduren, Tabellen und Abfragen für die jeweilige Datenquelle. Folgende Tabellen sind generell zu unterscheiden:

- *Datentabellen* enthalten die entladenen Basisdaten aus HISCOB
- *Hilftabellen* enthalten aggregierte Tabellen und werden von den Abfragen genutzt. Durch Hilftabellen wird die Performance der Abfragen besser, außerdem stehen bei möglichen Ladefehlern die relevanten Tabellen für die Abfragen noch zur Verfügung.
- *Schlüsseltabellen* enthalten Schlüssel und Metadaten, z.B. Kostenarten, Semester, Abschluss etc.

Das COB-Modul besteht im Endzustand aus Tabellen, Prozeduren und Abfragen. Da die COB-Datenbank der HIS-GmbH an sehr vielen Hochschulen eingesetzt wird, ist das COB-System gut geeignet für die Übernahme nach SuperX. Das COB-Modul ist weniger komplex als z.B. das SOS-Modul.

Falls es bei der Implementation des COB-Moduls zu Problemen kommt, können Sie sich unter www.superx-projekt.de informieren.

Supportadresse allgemein:

support@superx-projekt.de

Supportadresse für Baden-Württemberg:

support-bw@superx-projekt.de

Sven Gutow gutow@his.de	Daniel Quathamer danielq@memtext.de
Oliver Seegers seegers@his.de	Meikel Bisping

2 Installation des COB-Moduls

Um das COB-Modul zu installieren, muss man zunächst das Kernmodul Version 3.0(rcx) von SuperX installieren. Dies lässt sich auf einem separaten Rechner installieren, aber auch auf dem COB-Rechner, je nach Installationsvariante.

Zunächst gehen Sie in das Verzeichnis \$SUPERX_DIR (normalerweise /home/superx) und entpacken dort das Archiv `cob<<Versionsnr>>.tar.gz`.

Bei der Installation des COB-Moduls werden zentrale Schlüsseltabellen erzeugt und Installationscripte gestartet. Zunächst müssen die notwendigen Tabellen erzeugt werden, danach können Daten aus COB übernommen werden. Im Anschluss daran werden Hilfstabellen erzeugt, und die Abfragen eingespielt. Die Leserechte müssen dann manuell gepflegt werden. Damit zukünftig COB-Daten in die SuperX-Datenbank eingelesen und mitgelieferte Abfragen getestet werden können, muss die SuperX-Datenbank zunächst einmalig um die COB-Bestandteile erweitert werden. Dazu gehören Tabellen (Basisdaten, Schlüsseltabellen), Prozeduren, Abfragen und Masken.

Die Scripte des COB-Moduls laufen unter LINUX (Suse, RedHat - ggfs. nach gewissen Anpassungen auch unter AIX und SOLARIS) und unter Windows mit Cygwin.

2.1 Kurzanleitung

Folgende Arbeitsschritte sind notwendig:

- 1) Entpacken des COB-Moduls in `$$SUPERX_DIR` und Einrichten der Umgebung (Prüfen ob in Ihrer `$$SUPERX_DIR/db/bin/SQL_ENV` alle Einträge aus `SQL_ENV.cob.sam` vorhanden sind, ggfs. rüberkopieren). Aktivieren Sie die Umgebung mit
`.$SUPERX_DIR/db/bin/SQL_ENV`
- 2) Wenn der COB-Rechner ein Windows-Rechner ist, müssen Sie zunächst den SuperX-Client installieren (<http://download.superx-projekt.de>); dies ist im Paket des SuperX-Client dokumentiert.
- 3) Kopieren der Entladescripte unter `$$SUPERX_DIR/db/module/cob/rohdaten` zum COB-Rechner
- 4) Anpassen der **Umgebungsvariablen** für das Entladescript (COB-Version, Startjahr) in der Datei `$$SUPERX_DIR/db/module/cob/rohdaten/COB_ENV` (für UNIX) bzw. `cob_env.bat` (Windows). Für beide Dateien sind Muster mit jeweils der Endung `.sam` vorhanden.
- 5) Nur unter Windows: Entscheiden Sie sich für eine Entladevariante (RMI-ODBC oder dateibasiertes Entladen) und gehen Sie entsprechend der Anleitung vor.
- 6) Ausführen des Entladescripts `cob_unload.x` (UNIX) bzw. `cob_unload.bat` (Windows). Prüfen Sie die Logdatei `cob_unload.err`
- 7) Kopieren des Verzeichnisinhalts von `rohdaten` vom COB-Rechner nach
`$$SUPERX_DIR/db/module/cob/rohdaten`
- 8) Erzeugen des **COB-Moduls** in der SuperX-Datenbank:
`$$SUPERX_DIR/db/module/cob/cob_modul_erzeugen.x`
Es werden Tabellen, Views etc. erzeugt, und die Masken werden eingespielt. Nach der Installation haben die User der Gruppen *Administration* oder *superx* Leserecht auf die Abfragen. Falls ein Fehler auftritt, versuchen Sie die Ursache zu beheben, starten Sie dann
`$$SUPERX_DIR/db/module/cob/cob_modul_entfernen.x` (etwaige Fehler ignorieren)
und anschließend erneut
`$$SUPERX_DIR/db/module/cob/cob_modul_erzeugen.x`
- 9) Nun bereiten Sie die erste Übernahme vor. Je nach Datenumfang und Rechner bietet es sich an, am Anfang nur wenige Rohdaten zum Testen zu nehmen, sonst dauert der erste Update je nach Datenmenge 1-6 h. Dazu gibt es ein Shellscript `cob_shrink.x`, das die Rohdaten auf max. 100 Sätze kürzt. Beim Echtbetrieb später können Sie mit `cob_unshrink.x` dies wieder rückgängig machen.
- 10) Um Probleme der Vergleichbarkeit mit HISCOB aufgrund von Gültigkeitszeiträumen zu vermeiden, sollte die Gültigkeit von Kostenstellen,-arten und -trägern überschrieben und die Standbuttons für die Sichten abgeschaltet werden. Erstellen Sie dazu eine Kopie der Datei `finalize.sql.sam` und nennen Sie sie `finalize.sql`
- 11) Übernahme der entladenen **COB-Daten** nach SuperX:

`$SUPERX_DIR/db/module/cob/cob_update.x --mit-organigramm`

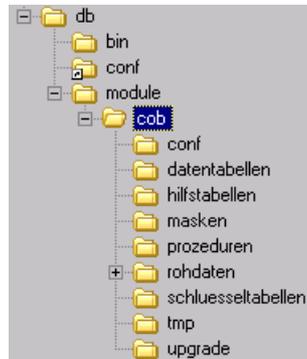
(falls Sie das Organigramm anwederweitig pflegen, lassen Sie den Parameter `--mit-organigramm` weg)

Die Logdatei heißt `cob_update.err`.

- 12) Bei Fehlern oder Warnungen im Update wird eine Meldung in der Logdatei ausgegeben. Ansonsten melden Sie sich im Applet neu an, und testen Sie nun die Abfragen.
(Im XML-Frontend müssen Sie zunächst unter `http://rechnername:8080/superx/servlet/SuperXManager` den Server-Cache aktualisieren und sich dann neu anmelden, oder Sie starten Tomcat einfach neu)
- 13) Für eingeschränkte User müssen Sie Rechte vergeben (vergl. entsprechender Abschnitt)
- 14) Schritt 10 wird bei jedem SuperX-Update wiederholt. Nun muss der Entladerhythmus geplant werden, und die Cronjobs werden eingerichtet. Eine Vorlage befindet sich in `$COB_PFAD/cob_update_cron.x.sam`

2.2 Ordnerstruktur und Umgebung des COB-Moduls

Die folgende Abbildung zeigt die Ordnerstruktur des COB-Moduls.



Das Masken-Verzeichnis im COB-Modul ist nicht zu verwechseln mit dem des Kernmoduls: Im Masken-Verzeichnis des COB-Moduls werden die bei Installation mitgelieferten COB-Abfragen gespeichert: das Masken-Verzeichnis des Kernmoduls dient als Arbeitsbereich für eigene Anpassungen. Diese Trennung ist wichtig, falls Sie Updates oder neue Abfragen zum COB-Modul installieren wollen.

Die jeweiligen Pfade zum COB-Modul werden in der Datei `$SUPERX_DIR/db/bin/SQL_ENV` festgelegt. Übertragen Sie ggf. die Angaben von `db/bin/SQL_ENV_cob.sam` nach

`$SUPERX_DIR/db/bin/SQL_ENV`, und rufen Sie das Script einmal auf mit

`. $SUPERX_DIR/db/bin/SQL_ENV`

Die folgende Tabelle zeigt einen Auszug aus der `SQL_ENV`

```
SUPERX_MODULE=$SUPERX_DIR/db/module; export SUPERX_MODULE
SUPERX_ROHDATEN=rohdaten; export SUPERX_ROHDATEN

COB_PFAD=$SUPERX_MODULE/cob; export COB_PFAD
COB_ERRORDAT=$COB_PFAD/cob_update.err; export COB_ERRORDAT
COB_LOAD_PFAD=$COB_PFAD/$SUPERX_ROHDATEN; export COB_LOAD_PFAD
COB_ERRORMAIL=$ERRORMAIL #Standard-Mailadr.
export COB_ERRORMAIL
COB_LOGMAIL=$LOGMAIL #Standard-Mailadr.
export COB_LOGMAIL
COB_BACKUP="false"
export COB_BACKUP
```

Sie müssen diese Pfade ergänzen, damit die Scripte und cron-Jobs laufen.

In den jeweiligen Scripten wird diese für das Setzen der Umgebungsvariablen genutzt, so dass in der `.profile` des Users SuperX keine Änderungen notwendig sind. Hinweis für Datenbankserver unter AIX oder anderen Linux / Unix-Derivaten: Beachten Sie, dass die Scripte

nur dann lauffähig sind, wenn auf dem Datenbankserver unter /bin die Datei (oder ein Link) `sh` und `bash` liegt. Die Scripte von SuperX erwarten die `bash`-Shell Linux-konform im Verzeichnis /bin; wenn dies nicht der Fall ist, sollten `sh` und `bash` z.B. von /usr/bin nach /bin kopiert oder gelinkt werden.

2.3 Entladen der COB-Daten

Zunächst ist das Verfahren beim Entladen selbst zu konfigurieren. Einerseits können die Daten mittels der mitgelieferten Shellscripate entladen werden, oder (bei Einsatz in HISinOne) direkt durch den Applikationsserver. Letzteres ist im [HISWIKI](#) dokumentiert, die Shellscripate sind [unten](#) dokumentiert.

2.3.1 Allgemeine Parameter zum Entladen

JDBC_PARAM	Wenn Sie unter Postgres aus COB-GX 12 oder höher entladen, muss beim Unload ein spezielles JDBC-Kommando abgesetzt werden, das dem Client sozusagen den Weg zum COB-Schema zeigt. Dieses Kommando wird wie folgt aktiviert: JDBC_PARAM="set search_path to cob;"
VERSION	Versionsnummer des Quellsystems(Ganzzahl), möglich sind bei COB 6,7,8,9,10,11,12,13 , bei HISinOne können Sie das Feld leer lassen
SOURCESYSTEM	Beim Quellsystem COB wählen Sie hier "cob", bei oder HISinOne wählen Sie "hisinone"
START_COB	Startjahr zum Entladen: Ab welchem Haushaltsjahr soll entladen werden? Default ist der Wert "2003".
TRANSACTION_OFF	Nur für Informix: Transaktionen abschalten? Wenn Sie tagsüber entladen, Transaktionen eingeschaltet sind und die COB -Tabellen groß sind, dann sollte dieser Wert gesetzt sein: SET ISOLATION TO DIRTY READ;

2.3.2 Entladen mit oder ohne Gültigkeitszeiträumen

Die Entladeroutinen erlauben es, **Gültigkeitszeiträume** von integrierten Schlüssel Tabellen (bei COB-GX sind dies `inst`, `fikr`, `proj`, `gege`, bei HISinOne sind dies `cost_center`, `financial_account`, `project` und `budget_source`) zu ignorieren. Dies kann in folgendem Fall nützlich sein:

Das Vorsystem ignoriert ggf. die **Gültigkeit von Schlüsseln**, das DWH aber nicht. Wenn Sie also Gültigkeitszeiträume begrenzt haben, kann es vorkommen dass Berichtsergebnisse anders aussehen als im Vorsystem. Um dies zu vermeiden können Sie die Gültigkeiten beim Entladen auf "Unendlich" setzen. Dies ist bei der Auslieferung des Moduls für COB-GX sogar der Standard.

Um die Gültigkeitszeiträume zu ignorieren (oder nicht) gehen Sie bitte wie folgt vor: Setzen Sie die entsprechende Variable auf 0, um die Gültigkeiten zu beachten, und 1, um sie zu ignorieren. Hier eine Übersicht der Variablen:

Merkmal	Variablenname	Tabelle COB-GX	Tabelle HISinOne
Kostenstelle	COB_VONBIS_INST	inst	cost_center
Kostenträger	COB_VONBIS_PROJ	proj	project
Kostenart	COB_VONBIS_FIKR	fikr	financial_account
Geldgeber	COB_VONBIS_GEGE	gege	budget_source

Besonderheit bei Kostenträgern: meist ist es eher sinnvoll, die alten Kostenträger auszublenden, d.h. die Gültigkeit auszuwerten. Andererseits gibt es häufig bei Kostenträgern Buchungen, die nach offiziellem Ende des Projektes stattfinden. Daher haben wir einen Automatismus eingebaut, daß die (evtl. eingeschränkte) Gültigkeit eines Kostenträgers auf die Jahre ausgedehnt wird, zu denen es Primärbuchungen gibt.

2.3.3 Entladen mit Shellscripten

Das Entladen aus COB gestaltet sich je nach Datenbank- und Betriebssystem unterschiedlich. Zunächst wird der häufigste und einfachste Fall beschrieben, das Entladen unter UNIX.

Kopieren Sie die Datei

```
$SUPERX_DIR/db/module/cob/rohdaten/COB_ENV.sam
```

nach

```
$SUPERX_DIR/db/module/cob/rohdaten/COB_ENV
```

und passen Sie sie entsprechend der weiteren Anleitung an.

COB_ENV Die Umgebung für das Entladescript aus COB

```
##Pfad für Entladedaten:
COB_PFAD=. ; export COB_PFAD
##hier muss Unterverzeichnis unl existieren!
DBASE=cob
export DBASE
#Start des Ladens:
start_cob=2002; export start_cob #Startzeit-
punkt
VERSION=11
export VERSION
#SOURCESYSTEM auswählen. Möglich ist cob für
COB-GX oder hisinone für das laden aus HISinO-
ne
SOURCESYSTEM=cob
export SOURCESYSTEM
```

Im Kopf der Datei `cob_unload.x` wird diese Datei aufgerufen; zunächst muss das Ausgabeverzeichnis `COB_PFAD` und das Startjahr, ab dem COB-Daten vorliegen bzw. entladen werden sollen, angegeben werden. Außerdem müssen der COB-Datenbankname (`DBASE`) und die Versionsnummer eingegeben werden (Ganzzahl).

Die Variable `SOURCESYSTEM` beschreibt das Quellsystem. Dies kann entweder `COB-GX` oder `HISinOne` sein. Achten Sie bitte darauf, dass der Wert `cob` oder `hisinone` klein geschrieben ist.

Zum Entladen via "Push" oder "Pull" siehe das SuperX-Adminhandbuch Kernmodul.

Zunächst werden DBMS-spezifische Parameter beschrieben. Darüberhinaus können Sie auch steuern, ob die [Gültigkeiten](#) zentraler Schlüssel Tabellen ignoriert werden sollen oder nicht.

2.3.3.1 DBMS-spezifische Konfiguration

2.3.3.1.1 Entladen aus Informix/ UNIX

Einige Tabellen der COB -Datenbank werden durch ein Entladescript entladen. Das Script befindet sich hier:

```
$SUPERX_DIR/db/module/cob/rohdaten/cob_unload.x
```

Wenn Sie im Push-Verfahren entladen: Kopieren Sie den gesamten Verzeichnisinhalt ab `/rohdaten` auf den COB-Rechner, geben Sie dem Script `cob_unload.x` Ausführungsrechte, kopieren Sie die Beispieldatei `COB_ENV.sam` nach `COB_ENV` und passen Sie darin die Umgebungsvariablen an.

COB_ENV **Informix**

```
DATABASE=INFORMIX; export DATABASE #Datenbanksystem  
VERSION=11; export VERSION #HIS-Modulversion
```

Im Kopf der Datei `cob_unload.x` wird diese Datei aufgerufen; hier muss `INFORMIXSERVER` und `INFORMIXDIR` angegeben werden.

Testen Sie zunächst den Zugriff mit dem Befehl:

```
. COB_ENV  
dbaccess $DBASE
```

Wenn die Datenbankverbindung klappt, dann starten Sie das Script mit `cob_unload.x`

Die Rohdaten werden in das Unterverzeichnis `rohdaten/unl` kopiert. Das Entladedatum wird danach in der Textdatei `rohdaten/superx.datum` gespeichert; wenn das Script einen Fehler findet, dann wird das vorherige Datum (in der Datei `superx.datum.alt`) gesetzt.

Ausgaben des Scriptes werden in die Datei `cob_unload.err` geschrieben. Diese Datei wird auf Meldungen über Entladefehler geprüft.

Wenn Sie das Organigramm aus COB benutzen und öfter aktualisieren wollen als das gesamte COB-Modul, haben wir ein Entladescript nur für das Organigramm (und verwandte Tabellen) erzeugt. Dies liegt ebenfalls im Verzeichnis `$COB_LOAD_PFAD` und lautet `inst_unload.x`

Die Umgebungssteuerung und der Start des Scripts funktioniert analog zum normalen COB-Unload.

2.3.3.1.2 Entladen aus Postgres/ Unix

Wenn Sie HISCOB mit Postgres einsetzen müssen Sie wie im vorherigen Abschnitt beschrieben die `COB_ENV.sam` nach `COB_ENV` kopieren und anpassen. `SX_CLIENT` muss `psql` sein.

Wir empfehlen, den Entladevorgang vom SuperX-Rechner aus zu starten ("Pull"-Verfahren):

Zum Zugriff auf die HISCOB-Datenbank müssen Sie analog zur `db.properties` die `db-cob.properties` anpassen, nachdem Sie die Beispieldatei `db-cob.properties.sam` nach `db-cob.properties` kopiert haben. Verwenden Sie `propadmin.x /path/to/cob/db-cob.properties`. Wenn Sie mit COB-GX 12 oder höher arbeiten, wird als Datenbankname nicht mehr "cob" angegeben, sondern "hism". Eine Musterdatei `db-hism_pg.properties.sam` liegt im Archiv.

Zum Entladen verwenden Sie das Script `cob_unload.x`.

Wenn Sie den Entladevorgang auf dem COB-Rechner durchführen wollen ("Push-Verfahren"), müssen Sie dort Java 1.4.2 oder höher installieren. Außerdem werden Bibliotheken aus dem SuperX-Kernmodul benötigt.

Sie können das Kernmodul z.B. unter `home/cob/superx` entpacken. Erstellen Sie eine `SQL_ENV` in der dann als `$SUPERX_DIR=/home/cob/superx` eingetragen ist.

Vor dem Start des Entladevorgangs müssen immer die Umgebungsvariablen in der `SQL_ENV` geladen werden mit `. /home/superx/cob/db/bin/SQL_ENV`.

Im Weiteren verfahren Sie analog zur Vorgehensweise beim Entladen aus Informix/Unix, nur dass natürlich `INFORMIXSERVER` etc nicht angegeben werden braucht. `PGHOST` oder `PGPORT` wird ebenfalls im COB-Modul nicht benötigt, da der Unload intern mit `jdbc` läuft. Weitere Variablen in der Datei `COB_ENV`:

JDBC_PARAM	Wenn Sie unter Postgres aus COB-GX 12 entladen, muss beim Unload ein spezielles JDBC-Kommando abgesetzt werden, das dem Client sozusagen den Weg zum SVA-Schema zeigt. Dieses Kommando wird wie folgt aktiviert: <code>JDBC_PARAM="set search_path to cob;"</code> <code>export JDBC_PARAM</code>
JDBC_CLASSPATH	Wenn Sie den obigen <code>JDBC_PARAM</code> nutzen, aber noch nicht mit dem Kernmodul 4.0 arbeiten, müssen Sie eine spezielle Bibliothek laden mit dem Befehl <code>JDBC_CLASSPATH="\$COB_LOAD_PFAD/lib/superx3.9.jar:\$JDBC_CLASSPATH"</code> <code>export JDBC_CLASSPATH</code> Wenn Sie bereits mit dem Kernmodul 4.0 arbeiten, ist dies nicht nötig.

2.3.3.1.3 Entladen aus Informix/ Windows

Wenn Sie Informix unter Windows betreiben, können Sie das Entladen vom SuperX-Rechner aus durchführen.

Setzen Sie in der Datei

```
$SUPERX_DIR/db/module/cob/rohdaten/COB_ENV
```

die Variable `SX_CLIENT=jdbc` und passen Sie alle benötigten Umgebungsvariablen an.

Kopieren Sie `db-cob.properties.sam` nach `db-cob.properties` und tragen Sie die Verbindungsparameter zur Informixdatenbank mit dem Propadmin ein.

```
propadmin.x /path/to/cob/db-cob.properties
```

Der Entladevorgang wird gestartet mit

```
cob_unload.x
```

Die Rohdaten werden in das Unterverzeichnis `rohdaten/unl` kopiert. Das Entladedatum wird danach in der Textdatei `rohdaten/superx.datum` gespeichert; wenn das Script einen Fehler findet, dann wird das vorherige Datum (in der Datei `superx.datum.alt`) gesetzt.

Ausgaben des Scriptes werden in die Datei `cob_unload.err` geschrieben. Diese Datei wird auf Meldungen über Entladefehler geprüft.

2.3.4 Nur beim PUSH-Verfahren: Kopieren der Rohdaten

In der Datei `COB_ENV.sam` sind noch weitere Umgebungsvariablen definiert (`REMOTE_HOST` etc). Diese werden von dem Script `cob_copy.x` benutzt, das die entladenen Rohdaten automatisch auf den SuperX-Rechner kopiert. Als unterstützte Übertragungsmethoden gelten dabei `scp` und `rsync`. Eine Anleitung zum Kopieren mit `scp/rsync` befindet sich im Adminhandbuch Kernmodul.

2.4 Neuinstallation: Erzeugung der Prozeduren, Schlüssel und Datentabellen

Diese Schritte brauchen nur einmal ausgeführt zu werden.

Unter UNIX:

1. Melden Sie sich als Benutzer `superx` an und wechseln Sie ins Verzeichnis

```
$SUPERX_DIR/db/module/cob.
```

2. Geben Sie ggf. dem Skript `cob_modul_erzeugen.x` (sowie allen anderen Scripten in diesem Verzeichnis) Ausführungsrechte mit

```
chmod +x *.x.
```

3. Starten Sie das Skript durch Eingabe von `cob_modul_erzeugen.x`.

Wenn ein Fehler aufgetreten ist, wird dies in der Datei `L_cob_installieren.log` protokolliert.

Wenn möglich, beheben Sie die Fehlerursache und rufen dann zunächst

```
cob_modul_entfernen.x
```

auf, damit evtl. schon installierte Teile entfernt werden (etwaige Fehlermeldungen beim Entfernen können normalerweise ignoriert werden). Dann probieren Sie erneut die Installation mit

```
cob_modul_erzeugen.x.
```

Neben dem Erstellen der Tabellen und Hinzufügen der Prozeduren und Abfragemasken, werden auch Einträge in den Themenbaum und die Tabelle `sachgebiete` gemacht.

Außerdem erhält der User `admin` Zugriffsrechte für das neue Sachgebiet Nr. 27 Kostenrechnung. Die Zugriffsrechte für andere User müssen mit Hilfe der Administrationsabfragen im XML-Frontend oder mit dem Access-Administrationsfrontend vorgenommen werden.

2.4.1 Konstanten

Nach der Installation gibt es ein paar Konstanten, mit denen der Ladeprozeß gesteuert wird:

apnr	beschreibung	Kommentar	Von HS im DB-FORM änderbar?
1	Organigramm_aus_COB	Schalter, ob das Organigramm aus COB-GX (Tab. inst) bzw. HISinOne-COA (Tab. orgunit) übernommen werden soll. 1=ja, 0=nein	ja
5	COB-Version	Version von HISCOB-GX (wird im Entladescript gesetzt)	nein
2000	COB-Startjahr	Startjahr, ab dem KLR-Daten entladen wurden (wird im Entladescript gesetzt)	nein
6	COB_Quellsystem	COB_Quellsystem enthält einen Wert für COB-GX (10) bzw. HisInOne (6) (wird im Entladescript gesetzt)	nein
0	COB_FIN_TO_BUSA_ERSETZEN	Konfiguration der Übernahme von Primärbuchungen aus FIN (siehe https://wiki.his.de/mediawiki/index.php/Haushalts-Buchungen_aus_Edustore-FIN_-_HISinOne)	ja
1900	COB_FIN_TO_BUSA_STARTJAHR	Startjahr der Übernahme von Primärbuchungen aus FIN (siehe https://wiki.his.de/mediawiki/index.php/Haushalts-Buchungen_aus_Edustore-FIN_-_HISinOne)	ja
	COB_SOSKEY	Diese Variable wird automatisch im SuperX-COB-Modul gesetzt. Es wird festgelegt, ob die Hochschule in ihrer COB-Studiengangstabelle amtliche (0) oder hochschulinterne Schlüssel (1) für Fächer und Abschlüsse setzt (Tabelle sys, Feld txt für msnr='SOSKEY'). Wird im SOS-Modul bei der Übernahme von Studiengängen in die lehr_stg_ab genutzt.	nein
0	COB_PROJ_EXT_VALID	Wenn die Gültigkeit der Projekte beschränkt ist: Suche maximales und minimales Jahr aller Primärbuchungen eines Projektes (egal ob Einnahme oder Ausgabe). Wenn Projekt kürzere Gültigkeit hat, wird es auf 1.1. des minimalen Jahrs (von) und 31.12. des max. Jahrs (bis) gesetzt.	ja

2.5 Upgrade eines vorhandenen COB-Moduls

Falls Sie sich nicht sicher sind welche Modulversion Sie nutzen: Die Version steht seit der Version 0.6 in der Textdatei `$(SUPERX_DIR)/db/module/cob/VERSION`.

Entpacken Sie die neue Version des COB-Moduls. Entladen Sie dann neu aus HISCOB mit dem Entladescript `cob_unload.x`

Starten Sie im Verzeichnis `$(SUPERX_DIR)/db/module/cob/upgrade`, das Skript `cob_modul1.0rc3_to_rc4.x`. Dieses Skript umfasst auch alle Änderungen der vorhergehenden rc1-rc3-Skripte.

2.6 Laden der Daten

2.6.1 Allgemeines

Generell bleiben die Bewegungsdaten im Data Warehouse gespeichert. Beim Laden werden nur jahresweise die neu eingeladenen Buchungsdaten zunächst gelöscht und dann neu eingefügt. Bei der Datenquelle HISinOne werden darüber hinaus Buchungsdaten, die bereits in Verteilrechnungen eingegangen sind, nicht mehr gelöscht (und auch nicht mehr neu eingefügt).

Über das [Startjahr beim Entladen](#) (`START_COB`) können Sie steuern, welche Haushaltsjahre neu eingefügt werden sollen. Aus Performancegründen kann es sinnvoll sein, nur die Daten ab einem neueren Haushaltsjahr X zu laden, wenn

- sichergestellt ist, daß sich Konten und Buchungen vor dem Jahr X nicht mehr ändern.
- Stammdaten (z.B. Kostenstellen), die vor dem Jahr X bebucht wurden, nicht gelöscht werden

2.6.2 Laden mit Shellscripten

Die entladenen KLR-Daten müssen für die Aufnahme nach SuperX im Verzeichnis `$(SUPERX_DIR)/db/module/cob/rohdaten/unl` stehen. Kopieren Sie die Daten dorthin, oder mounten Sie das Verzeichnis vom COB-Rechner als NFS-Verzeichnis. An der Stelle `$(SUPERX_DIR)/db/module/cob/rohdaten` müssen außerdem die vom Entladescript automatisch erzeugten Dateien `superx.datum.alt` und `unload.err` stehen.

Um Probleme der Vergleichbarkeit mit HISCOB aufgrund von Gültigkeitszeiträumen zu vermeiden, sollte die Gültigkeit von Kostenstellen,-arten und -trägern überschrieben und die Standbuttons für die Sichten abgeschaltet werden. Erstellen Sie dazu eine Kopie der Datei `finalize.sql.sam` und nennen Sie sie `finalize.sql`.

Zum Übernehmen der Basisdaten nach SuperX wird das Script

```
$SUPERX_DIR/db/module/cob/cob_update.x
```

gestartet.

(Falls Sie das Organigramm nicht aus der inst-Tabelle von HISCOB übernehmen wollen, lassen Sie den Parameter `--mit-organigramm` weg).

Für den ordnungsgemäßen Betrieb des COB-Moduls ist es wichtig, dass die Kostenstellen aus HISCOB auch im SuperX-Organigramm wiederfinden, daher empfehlen wir die Angabe des Parameters.

Das Skript füllt die Datentabellen; danach werden die Hilfstabellen neu erzeugt.

[Fall Sie die Hilfstabellen einmal manuell aktualisieren wollen, dann starten Sie das Script

```
$SUPERX_DIR/db/module/cob/hilfstabellen/update_hilfstabellen.x ]
```

Falls Fehler aufgetreten sind, wird ein Verweis auf eine Logdateien angegeben, die wiederum Verweise auf Logdateien für einzelne Arbeitsschritte enthalten kann. Dadurch muss man vielleicht zwei oder drei Logdateien öffnen, aber nicht in einer einzelnen meterlangen Logdatei umständlich die Fehlerstelle suchen.

Beim regelmäßigen Update wird die Übernahme der COB-Daten über Cronjobs erledigt.

Kopieren Sie dazu `$SUPERX_DIR/db/module/cob_update_cron.x.sam` nach `cob_update_cron.x` und tragen Sie es in die Crontab ein.

Derzeit werden bei jedem Update die gesamten COB-Daten neu übernommen, es ist noch keine Funktionalität zur Übernahme von lediglich geänderten Datensätzen in COB vorgesehen, wie es beim SOS-Modul der Fall ist.

Kommen alle Buchungen nach SuperX?

Die Tabelle `busa` wird nicht komplett in SuperX übernommen; in dieser Tabelle werden alle Buchungen, die für COB nicht relevant sind (Schalter "relevant für COB") herausgefiltert. Der Schalter in der Tabelle `fikr`, `proj` und `inst` wird dazu genutzt.

Ebenso werden Storno-Buchungen nicht übernommen.

2.6.3 Hochschulspezifische Transformationen

Nach dem Laden der Rohdaten können vor und nach der Transformation hochschulspezifische Anpassungen vorgenommen werden. Dafür sind jeweils die Scripte

```
$COB_PFAD/preparation.sql
```

sowie

```
$COB_Pfad/finalize.sql
```

vorgesehen.

Der SuperX-ETL-Controller (`$SUPERX_DIR/db/bin/module_etl.x`) prüft nach dem Laden der Rohdaten, ob im `$COB_Pfad` ein Script mit dem Namen `preparation.sql` liegt; wenn ja, dann wird es ausgeführt. Wenn es ein Script `finalize.sql` gibt, wird dies nach der Datentransformation ausgeführt.

Im COB-Modul liegt bereits eine `finalize.sql` vor. Darin werden die Gültigkeiten von Kostenstellen,-arten und -trägern zurückgesetzt und das Anzeigen von „Standauswahl-Buttons“ deaktiviert, weil HISCOB regulär noch keine **zeitraumbezogene Auswertung** unterstützt.

Wenn Sie die Gültigkeitsangaben in COB gut gepflegt haben und in SuperX zeitbezogene Auswertungen machen wollen, können Sie einzelne Abschnitt auskommentieren, in dem Sie zwei Striche davor setzen.

2.6.4 Separate Übernahme des Organigramms

Wenn das Organigramm aus COB übernommen wird, ist es sinnvoll, den Laderhythmus für das Organigramm anders zu betreiben als den gesamten COB-Update. Während ein COB-Update eher stichtagsbezogen läuft (z.B. 2x pro Semester), sollte das Organigramm eher tagesaktuell vorliegen. Dazu gibt es zum **Entladen** ein eigenes Script (`inst_unload.*`), und zur Übernahme nach SuperX ein Script namens

```
$COB_Pfad/update_organigramm.x
```

Dieses Script erzeugt aus einer entladenen `cob_inst` ein neues Organigramm. Hochschulspezifische Transformationen können Sie in der Datei `preparation_cob_inst.sql` eintragen (ein Muster mit der Endung `.sam` liegt vor).

Beispiel für hochschulspezifische Transformationen:

`$(COB_PFAD)/preparation_cob_inst.sql.sam`

--Manche Hochschulen setzen in ihren Inst-Tabellen die übergeordnete Institution in das Namensfeld, so

--dass das Organigramm unleserlich wird. Der folgende Update setzt den Namen gleich dem Drucktext.

update cob_inst_neu set name=drucktext;

--In der cob_inst-Tabelle dient das Feld orgstruktur der Gruppierung von Kostenstellen zu Bereichen.

--Die Konvention in SuperX ist, dass Lehreinheiten mit 30 gekennzeichnet sind,

--Fakultäten mit 20, und Institute mit 40. Wenn Ihre Hochschule andere

--Schlüssel verwendet, müssen diese hier angepasst werden.

--Wenn Sie z.B. Fakultäten mit 40 verschlüsseln:

--Zuerst die vorherigen 20er sichern:

update cob_inst_neu set orgstruktur=99 where orgstruktur=20;

--Dann alle 40er auf 20 setzen.

update cob_inst_neu set orgstruktur=20 where orgstruktur=40;

Diese Änderung würde bei Ihnen aktiv, wenn Sie die Datei umbenennen nach `preparation_cob_inst.sql`.

Vor dem Update wird eine Sicherung des Organigramms gemacht. Bei Update-Fehlern wird das alte Organigramm wiederhergestellt, und eine Mail an den Admin geschickt.

Das Script kann allein in die `crontab` eingetragen werden. Wichtig ist, dass Sie im Kopf der Datei `$(SUPERX_DIR)` richtig setzen.

2.6.5 Geldgeber zu Drittmitteln zuordnen

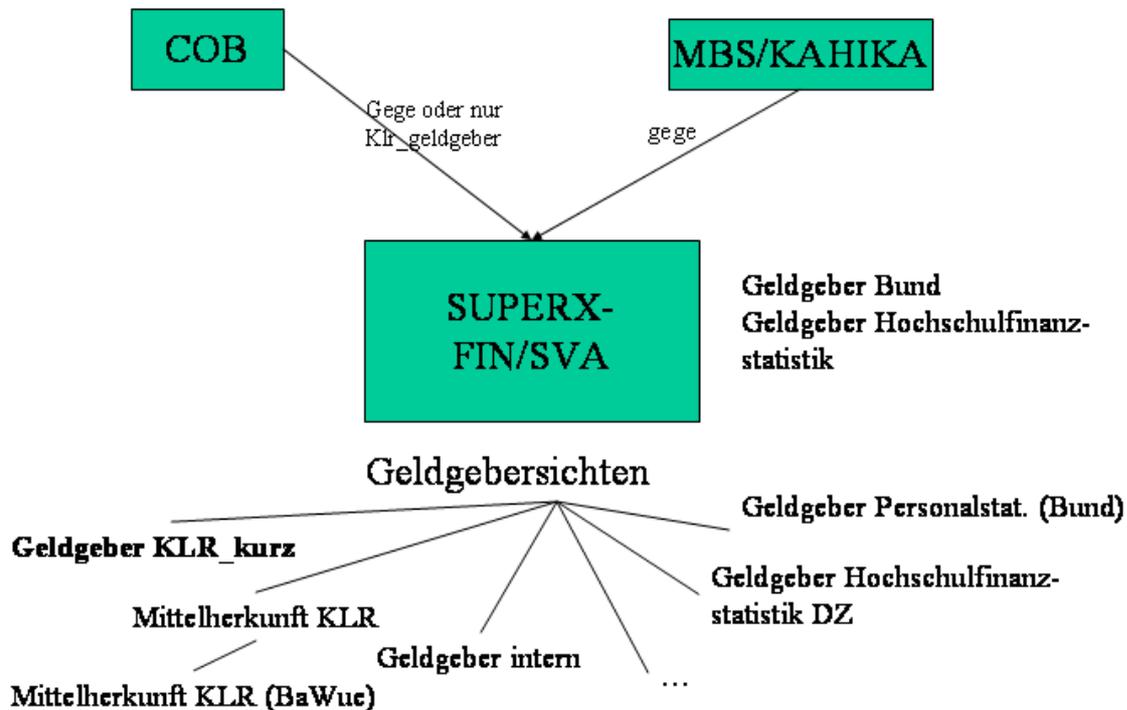
Die Geldgeber werden in unterschiedlichen Sichten gruppiert. Die folgende Abbildung zeigt die verschiedenen Varianten.

Folgende Sichten werden mit SuperX mitgeliefert:

- Mittelherkunft KLR. In dieser Sicht wird neben dem Geldgeber als übergeordnetes Element statt ueberkey das Feld `klr_geldgeber` genutzt. Für Hochschulen in BadenWürttemberg gibt es eine spezielle Sicht mit vorgegebenen Schlüsseln (siehe [Anhang](#), rechte Spalte)
- Geldgeber KLR (Kurz): Hier werden die einzelnen Geldgeber nach dem Muster Drittmittel und zwei weitere beliebige Geldgebergruppen aufgeteilt. Ausschlaggebend ist dabei in der Tabelle `sva_geldgeber` das Feld `klr_geldgeber` das erste Zeichen: Ist es ein "D", dann sind es Drittmittel, die beiden anderen Schlüssel sind frei.
- Geldgeber (intern). Dies ist die hochschulinterne Sicht auf Geldgeber bestehend aus `ggnr` und `ueberkey`.
- Geldgeber Hochschulfinanzstatistik: Hier wird die Aggregation aufgrund des Feldes `fikey` in der Tabelle `sva_geldgeber` genutzt (Siehe [Anhang](#), mittlere Spalte)



Geldgeber-Sichten



Geldgeber Bundes-Stellenstatistik: Hier wird die Aggregation aufgrund des Feldes `bund_geldgeber` in der Tabelle `sva_geldgeber` genutzt (Siehe [Anhang](#), linke Spalte)

Wenn die Gruppierungsfelder in dem entsprechenden Vorsystem (MBS) gepflegt sind, können sie direkt übernommen werden. Die Gruppierungsmerkmale können auch aus COB übernommen werden, wenn die entsprechende [Konstante](#) gesetzt ist.

2.6.5.1 Geldgeber KLR (Kurz)

Die Geldgebersicht `Geldgeber KLR kurz` wird zwar aus COB/MBS übernommen, lässt sich aber manuell ändern. Die Geldgeber werden einzeln zur Kategorie "[Drittmittel](#)" zugeordnet. Gehen Sie dazu im Browser über das XML-Frontend in der **Maske Prüfprotokoll Finanzstatistik**, Link "Geldgeber zu Drittmitteln":

Sie können die Volltexte der Geldgeber ändern und durch Ankreuzen festlegen, dass diese Geldgeber zu Drittmitteln zählen.

Beachten Sie, dass Sie bei jeder Zeile separat speichern müssen.

Geldgeber-Klassifikation des COB-Moduls bearbeiten	In diesem Formular können Sie Geldgeber zu Drittmitteln zuordnen.	Nach Änderungen sollten Sie den Cache aktualisieren SuperXManager
---	---	---

Tid	Wert	Geldgeber	Drittmittel		
754	015	Abfallentsorgungsverband	<input checked="" type="checkbox"/>	Speichern	Löschen
829	11	ABM	<input type="checkbox"/>	Speichern	Löschen
805	066	ACAS Augustusburg Conf	<input checked="" type="checkbox"/>	Speichern	Löschen
763	024	AHABAU Berlin	<input checked="" type="checkbox"/>	Speichern	Löschen
1.734	302	AIF	<input checked="" type="checkbox"/>	Speichern	Löschen
800	061	Alzheimer Gesellschaft Br	<input checked="" type="checkbox"/>	Speichern	Löschen
777	038	Amt "Am Senftenberger S	<input checked="" type="checkbox"/>	Speichern	Löschen
1.756	711	and.Bundesländ.	<input checked="" type="checkbox"/>	Speichern	Löschen
1.743	412	and.Landesbehö.	<input checked="" type="checkbox"/>	Speichern	Löschen
1.757	712	and.Landesbehö.	<input checked="" type="checkbox"/>	Speichern	Löschen
814	079	Arbeitsamt Cottbus	<input checked="" type="checkbox"/>	Speichern	Löschen
798	059	Arte Kunsthandel GmbH	<input checked="" type="checkbox"/>	Speichern	Löschen
1.759	722	AvH	<input checked="" type="checkbox"/>	Speichern	Löschen
1.738	306	BA Naturschutz	<input checked="" type="checkbox"/>	Speichern	Löschen
1.778	938	BASF	<input checked="" type="checkbox"/>	Speichern	Löschen
744	005	BASF Schwarzheide Gmt	<input checked="" type="checkbox"/>	Speichern	Löschen
1.736	304	BAST	<input checked="" type="checkbox"/>	Speichern	Löschen
1.779	939	Bayer AG	<input checked="" type="checkbox"/>	Speichern	Löschen
807	068	Behindertenwerk Sprembe	<input checked="" type="checkbox"/>	Speichern	Löschen
1.733	301	BMBF	<input checked="" type="checkbox"/>	Speichern	Löschen
1.739	307	BMU	<input checked="" type="checkbox"/>	Speichern	Löschen
1.740	308	BMWT	<input checked="" type="checkbox"/>	Speichern	Löschen

Nach einer Änderung muss der SuperX-Update des FIN -Moduls komplett neu gestartet werden.

2.6.5.2 Mittelherkunft KLR

Die Geldgeber-Sichten **Mittelherkunft KLR** (BaWue oder andere) speisen sich aus dem Feld `klr_geldgeber` der Tabelle `gege`. Sie wird zwar aus COB/MBS übernommen, lässt lässt sich aber manuell ändern. Die Geldgeber werden einzeln zu den Kategorien zugeordnet, die im Feld `klr_geldgeber` vorgegeben sind (unterste Ebene). Außerdem lassen sich die Volltexte der Gruppierungsebenen in `klr_geldgeber` ändern (oberste Ebene). Gehen Sie zum Bearbeiten im Browser über das XML-Frontend in der **Maske Prüfprotokoll Finanzstatistik**, auf die

Links unter "Mittelherkunft (KLR)" (jeweils für Hochschulen in BaWue oder außerhalb BaWue, das folgende Beispiel zeigt die Geldgeber außerhalb BaWue):

Oberste Ebene.

Sie können die Volltexte der Geldgeber-Kategorien ändern.



Oberste Ebene der Geldgeber-Klassifikation des COB-Moduls bearbeiten	In diesem Formular können Sie KLR-Gruppierungen bearbeiten.	Nach Änderungen sollten Sie den Cache aktualisieren SuperXManager
---	---	--

In dieser Tabelle definieren Sie Schlüssel und Volltexte der Werte, die Sie in der COB-Tabelle gegen im Feld 'klr_geldgeber' pflegen. Da im entsprechenden HIS-System (COB) nur ein Schlüssel, kein Volltext vorgesehen ist, können Sie den Volltext hier pflegen.

Tid	Wert	Bezeichnung		
627	KLR_D	D	<input type="button" value="Speichern"/>	<input type="button" value="Löschen"/>
1.633	KLR_H	H	<input type="button" value="Speichern"/>	<input type="button" value="Löschen"/>
628	KLR_L	L	<input type="button" value="Speichern"/>	<input type="button" value="Löschen"/>
629	KLR_S	S	<input type="button" value="Speichern"/>	<input type="button" value="Löschen"/>
626	KLR_	Sonstige	<input type="button" value="Speichern"/>	<input type="button" value="Löschen"/>

Unterste Ebene

Sie können die Volltexte der Geldgeber ändern und die Gruppierungsebene zuweisen. Die Gruppierungsebene entspricht der obigen "Obersten Ebene".



Geldgeber-Klassifikation des COB-Moduls bearbeiten	In diesem Formular können Sie Geldgeber zu KLR-Gruppierungen zuordnen.	Nach Änderungen sollten Sie den Cache aktualisieren SuperXManager
---	--	--

Tid	Wert	Geldgeber	Gruppierung		
536	015	Abfallentsorgungsverband	KLR_D-D	<input type="button" value="Speichern"/>	<input type="button" value="Löschen"/>
611	11	ABM	KLR_L-L	<input type="button" value="Speichern"/>	<input type="button" value="Löschen"/>
587	066	ACAS Augustusburg Conf	KLR_D-D	<input type="button" value="Speichern"/>	<input type="button" value="Löschen"/>
545	024	AHABAU Berlin	KLR_D-D	<input type="button" value="Speichern"/>	<input type="button" value="Löschen"/>
1.585	302	AIF	KLR_D-D	<input type="button" value="Speichern"/>	<input type="button" value="Löschen"/>
582	061	Alzheimer Gesellschaft Br	KLR_D-D	<input type="button" value="Speichern"/>	<input type="button" value="Löschen"/>
559	038	Amt "Am Senftenberger S	KLR_D-D	<input type="button" value="Speichern"/>	<input type="button" value="Löschen"/>
1.607	711	and Bundesländ.	KLR_D-D	<input type="button" value="Speichern"/>	<input type="button" value="Löschen"/>
1.608	712	and.Landesbehö.	KLR_D-D	<input type="button" value="Speichern"/>	<input type="button" value="Löschen"/>
1.594	412	and.Landesbehö.	KLR_D-D	<input type="button" value="Speichern"/>	<input type="button" value="Löschen"/>
596	079	Arbeitsamt Cottbus	KLR_D-D	<input type="button" value="Speichern"/>	<input type="button" value="Löschen"/>
580	059	Arte Kunsthandel GmbH	KLR_D-D	<input type="button" value="Speichern"/>	<input type="button" value="Löschen"/>
1.610	722	AvH	KLR_D-D	<input type="button" value="Speichern"/>	<input type="button" value="Löschen"/>
1.589	306	BA Naturschutz	KLR_D-D	<input type="button" value="Speichern"/>	<input type="button" value="Löschen"/>
1.629	938	BASF	KLR_D-D	<input type="button" value="Speichern"/>	<input type="button" value="Löschen"/>
526	005	BASF Schwarzheide Gmb	KLR_D-D	<input type="button" value="Speichern"/>	<input type="button" value="Löschen"/>
1.587	304	BAST	KLR_D-D	<input type="button" value="Speichern"/>	<input type="button" value="Löschen"/>
1.630	939	Bayer AG	KLR_D-D	<input type="button" value="Speichern"/>	<input type="button" value="Löschen"/>
589	068	Behindertenwerk Spremb	KLR_D-D	<input type="button" value="Speichern"/>	<input type="button" value="Löschen"/>
1.584	301	BMBF	KLR_D-D	<input type="button" value="Speichern"/>	<input type="button" value="Löschen"/>
1.590	307	BMU	KLR_D-D	<input type="button" value="Speichern"/>	<input type="button" value="Löschen"/>
1.591	308	BMAKT	KLR_D-D	<input type="button" value="Speichern"/>	<input type="button" value="Löschen"/>

Beachten Sie, dass Sie bei jeder Zeile separat speichern müssen.

2.6.6 Validierung KLR-Daten im Vorsystem

2.6.6.1 Validierung in COB-GX

Zur Validierung von Primär- und Sekundärkosten können Sie in COB-GX mit dem Schnell-Info arbeiten. Die folgende Maske gibt eine analoge Ausgabe:

Kosten und Erlöse (Primär und Sekundär)

Legende															
Kostenträger Kosten in Prim.Kosten von KoSt: Inklusive ; Jahr von: 2013 ; Jahr bis: 2013 ; Verteilschritt: KoTr Lehre (Verteilschritt); Ausgabe: nach Kostenstelle ; Untergeordnete Elemente: Inklusive ; User: admin															
Stand: 05.12.2013															
Ebene	Nummer	Name	Jahr	Primärkosten	Primärerlöse	Saldo Primär	Plan Primärkosten	Plan Primärerlöse	Sekundäre Kosten	Sekundäre Erlöse	Verteilte Kosten	Verteilte Erlöse	Gesamt Kosten	Gesamt Erlöse	Gesamt Saldo
1	1	1-KoSt gesamt	2013	454.303,53	1.002.000,00	-547.696,47	0,00	0,00	268.524,60	1.003.455,89	703.000,22	2.003.455,89	19.827,91	2.000,00	17.827,91
2	100	100-Wissenschaft	2013	204.146,73	2.000,00	202.146,73	0,00	0,00	246.091,39	974.499,80	430.410,21	974.499,80	19.827,91	2.000,00	17.827,91
3	01	01-Mathematik	2013	0,00	0,00	0,00	0,00	0,00	1.736,79	6.983,88	1.736,79	6.983,88	0,00	0,00	0,00
3	02	02-Physik	2013	0,00	0,00	0,00	0,00	0,00	651,30	2.618,95	651,30	2.618,95	0,00	0,00	0,00
3	05	05-Biologie	2013	0,00	0,00	0,00	0,00	0,00	651,30	2.618,95	651,30	2.618,95	0,00	0,00	0,00
3	06	06-Theologische Fakultät	2013	149.017,07	2.000,00	147.017,07	0,00	0,00	139.280,61	550.578,54	268.469,77	550.578,54	19.827,91	2.000,00	17.827,91
3	07	07-Fakultät Sprach-u. Litera.	2013	0,00	0,00	0,00	0,00	0,00	9.291,83	37.363,74	9.291,83	37.363,74	0,00	0,00	0,00
3	08	08-Fakultät Erziehungswissen	2013	0,00	0,00	0,00	0,00	0,00	6.252,44	25.141,95	6.252,44	25.141,95	0,00	0,00	0,00
3	09	09-Juristische Fakultät	2013	0,00	0,00	0,00	0,00	0,00	13.025,92	52.379,07	13.025,92	52.379,07	0,00	0,00	0,00
3	10	10-Fakultät Wirtschaftswisse	2013	0,00	0,00	0,00	0,00	0,00	12.591,73	50.633,10	12.591,73	50.633,10	0,00	0,00	0,00
3	11	11-Fakultät Informatik	2013	0,00	0,00	0,00	0,00	0,00	18.670,49	75.076,67	18.670,49	75.076,67	0,00	0,00	0,00
3	12	12-Fakultät Elektrotechnik	2013	53.196,05	0,00	53.196,05	0,00	0,00	6.083,64	19.205,66	59.279,69	19.205,66	0,00	0,00	0,00

Alternativ können Sie auch folgende SQLs direkt in COB-GX absetzen, um die obige Auswertung zu reproduzieren.

2.6.6.1.1 Validierung von Primärbuchungen in COB-GX

Die Primärkosten im obigen [Beispiel](#) von 454.303,53 EUR lassen sich wie folgt reproduzieren:

```
-- primaere kosten
SELECT SUM( B.busa_betrag) AS betrag
FROM
        busa B, fikt F, inst I
WHERE B.busa_fiktkey=F.key
and B.busa_instnr=I.inst_nr
and B.busa_jahr=2013
AND F.cobrel = '1'
AND F.kokl = 'K'
--keine Storno-und Planbuchungen
AND B.busa_bukz not in ( 'IES','PES','PE')
and I.cobrel='1'
;
```

Die Erlöse lassen sich analog berechnen:

```
-- primaere erlöse
SELECT SUM( B.busa_betrag) AS betrag
FROM
        busa B, fikt F, inst I
WHERE B.busa_fiktkey=F.key
and B.busa_instnr=I.inst_nr
and B.busa_jahr=2013
```

```

AND F.cobrel = '1'
AND F.kokl = 'E'
--keine Storno-und Planbuchungen
AND B.busu_bukz not in ( 'IES','PES','PE' )
and I.cobrel='1'

```

;

ergeben 1.002.000,00 EUR. Wir fügen bei Erlösen in manchen Masken lediglich noch das negative Vorzeichen hinzu.

2.6.6.1.2 Validierung von Sekundärbuchungen in COB-GX

Für Sekundärbuchungen müssen Sie zunächst den jew. Verteilschritt auswählen, bis zu dem Sie auswerten wollen. Ermitteln Sie dazu bitte die Nummern aller Verteilschritte bis zum wählen, beim obigen [Beispiel](#) sind dies die Nummern 1-5.

Die Sekundärkosten im obigen [Beispiel](#) von 268.524,60 EUR lassen sich wie folgt reproduzieren:

```

-- Sekundärkosten:
SELECT SUM( B.vtbu_betrag) AS betrag
FROM
        vtbu B, inst I
WHERE B.vtbu_ziel_instnr=I.inst_nr
and B.vtbu_jahr=2013
and I.cobrel='1'
--Verteilschritte bis zum untersten:
and vtbu_varnr in (5,4,3,2,1);

```

Die Sekundärerlöse analog:

```

-- Sekundärerlöse:
SELECT SUM( B.betrag) AS betrag
FROM
        vtbu_e B, inst I
WHERE B.ziel_instnr=I.inst_nr
and B.jahr=2013
and I.cobrel='1'
--Verteilschritte bis zum untersten:
and B.varnr in (5,4,3,2,1);

```

Die verteilten Kosten von 703.000,22 EUR ergeben sich aus:

```

-- verteilte Kosten
SELECT SUM( B.vtbu_betrag) AS betrag
FROM
        vtbu B, inst I
WHERE B.vtbu_quell_instnr=I.inst_nr
and B.vtbu_jahr=2013
and I.cobrel='1'
and vtbu_varnr in (5,4,3,2,1)

```

;

Die verteilten Erlöse von 2.003.455,89:

```
-- Verteilte Erlöse:
SELECT SUM( B.betrag) AS betrag
FROM
        vtbu_e B,  inst I
WHERE  B.quell_instnr=I.inst_nr
and B.jahr=2013
and I.cobrel='1'
--Verteilschritte bis zum untersten:
and B.varnr in (5,4,3,2,1);
```

2.7 Vergeben von Rechten für eingeschränkte User

Damit ein Nicht-Administrator mit dem Cob-Modul arbeiten kann, müssen Sie Institutionsrechte und Sichtenrechte vergeben.

Melden Sie sich dazu als `admin` oder `superx` im XML-Frontend an.

2.7.1 a. Institutionsrechte

Wenn Sie einen neuen User anlegen wollen, benutzen Sie die Abfrage *User einrichten*, dort könnten Sie bei *Inst-Rechte* auch angeben, für welche Institution (=Kostenstelle) der User eine Berechtigung erhalten soll.

Wenn der User schon angelegt ist oder Sie weitere Institutionsrechte vergeben wollen, benutzen Sie die Abfrage *User suchen*. Es ist nicht notwendig eine Einschränkung zu machen. Klicken Sie Abschicken. In der Userliste klicken Sie neben dem gewünschten User auf *Bearbeiten*. Auf dem sich öffnenden Formular können Sie neue Institutionsrechte einfügen, in dem Sie unten eine Institution auswählen und *Neue Institution einfügen* anklicken.

Sie können auch vergebene Institutionsrechte entfernen. Markieren Sie die Institution und klicken Sie *Markierte Institution löschen*. Wenn Sie die Gültigkeit der Berechtigung ändern, müssen Sie anschließend *Markierte Institution speichern* anklicken.

2.7.2 b. Sichtrechte

Der User (oder einer der Gruppen zu denen der User gehört) muss Rechte für Sichten auf Kostenstellen, -arten und -träger erhalten.

Sie könnten z.B. eine Gruppe Cob-User anlegen, in dem Sie sich als `admin` oder `superx` im XML-Frontend anmelden und dann die Abfrage *Tabelle suchen* starten. Dann klicken Sie neben der Tabelle *groupinfo* auf *Bearbeiten* und im sich öffnenden Formular auf *neu*. Geben Sie

eine tid und einen Gruppennamen ein. Zum Abschluß klicken Sie auf *einfügen* und schließen die Fenster.

2.7.2.1 Kostenstellen

Falls noch nicht gemacht, melden Sie sich im XML-Frontend als `admin` oder `superx` im XML-Frontend an.

Rufen Sie die Abfrage *Sicht suchen* mit der Einschränkung *Kostenstellen-Sicht* auf.

Klicken Sie in der Zeile **reguläre Sicht** auf *User- und Gruppenrechte*.

Sicht-Art: Kostenstellen-Sicht ; Stand: 01.01.2003				
Name	Beschreibung	Art	Bearbeiten	User- und Gruppenrechte
reguläre Sicht	Kostenstellen COB-Import	Kostenstellen-Sicht		

Es öffnet sich ein Formular. Wählen Sie einen User oder eine Gruppe aus und klicken Sie „Neuen User einfügen“ bzw. „Neue Gruppe einfügen“, bis diese in der Liste der User bzw. Gruppen aufgeführt sind, die die Sicht sehen dürfen.

Nachdem Sie dem User bzw. der Gruppe Rechte für die reguläre Kostenstellensicht gegeben haben, können Sie das Formular schließen und ggfs. weitere Rechte für einzelne alternative Hierarchien vergeben.

2.7.2.2 Kostenträger

Bei Kostenträgersichten sehen eingeschränkte User nur die Kostenträger, die Institutionen zugeordnet sind, für die sie Rechte haben. (intern wird die `cob_proj_to_inst` Tabelle ausgewertet).

Sichten In diesem Formular können Sie die Le

Name der Sicht	reguläre Sicht
tid	111
Art	Kostenstellen-Sicht
Interner Name	Hauptsicht Kostenstellen
User-Rechte	<input checked="" type="checkbox"/>
User, die diese Sicht sehen dürfen:	User Neuen User <input type="text"/> Neuen User einfügen
Gruppe, die diese Sicht sehen dürfen:	Gruppe <input checked="" type="radio"/> <input type="text" value="Dezernenten"/> <input type="radio"/> <input type="text" value="Rektorat/Kanzler"/> Markierte Gruppe löschen Neue Gruppe <input type="text" value="Dezernenten"/> Neue Gruppe einfügen

Sie können für Kostenträger Sichtrechte nach dem gleichen Vorgehen wie bei Kostenstellen vergeben. Sie können aber auch Rechte für alle Kostenträgersichten auf einmal vergeben, wie im folgenden Abschnitt für Kostenarten beschrieben.

2.7.2.3 Kostenarten

Sie können bei der Vergabe von Rechten für Kostenartensichten genauso vorgehen wie bei Kostenstellen und für jede Sicht einzeln Rechte vergeben, Sie können es sich aber auch einfacher machen und eine Berechtigung für sämtliche Kostenartensichten vergeben.

Rufen Sie dazu die Abfrage *Sicht suchen* mit der Einschränkung „Kosten-/Erlösarten-Sicht“ auf und klicken Sie bei einer Sicht auf *User- und Gruppenrechte* vergeben.

Dann scrollen Sie ans Ende des Formulars. Dort können Sie einzelnen Usern oder Gruppen Rechte für die *ganze Sichtart* geben.

User, die die ganze Sichtart sehen dürfen:	User Neuen User <input type="text"/> Neuen User einfügen
Gruppen, die die ganze Sichtart sehen dürfen:	User Neue Gruppe <input type="text" value="Dezernenten"/> Neue Gruppe einfügen

Wählen Sie einen User oder eine Gruppe aus und klicken Sie „Neuen User einfügen“ bzw. „Neue Gruppe einfügen“, bis diese in der Liste der User bzw. Gruppen aufgeführt sind, die die *ganze Sichtart* sehen dürfen.

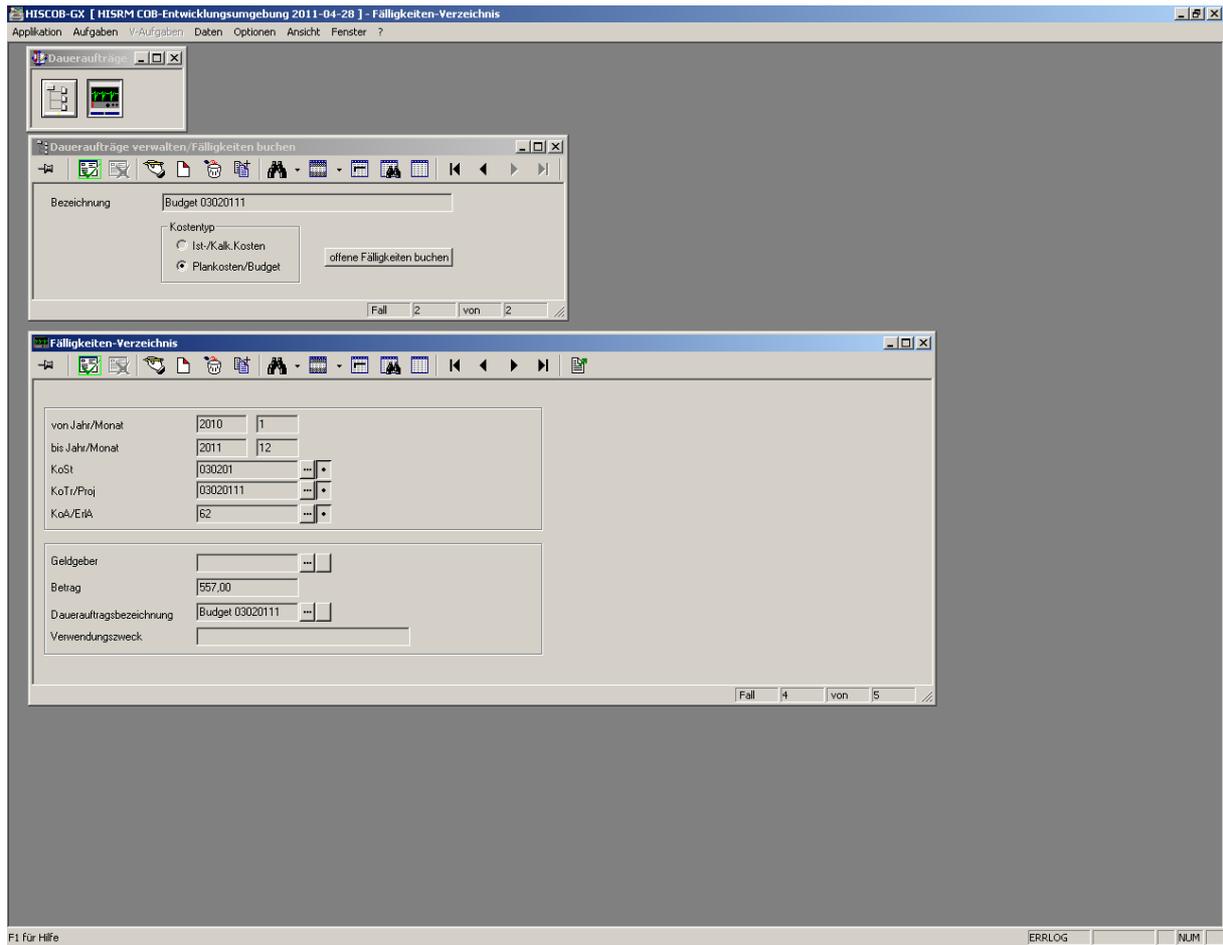
Das bedeutet, dass der User/die Gruppe *alle* Kostenartensichten sehen darf. Sie brauchen nicht mehr für jede Sicht (alternative Hierarchie) einzeln Rechte vergeben. Auch wenn neue Kostenartensichten dazukommen, erhält der User/die Gruppe bei der nächsten Anmeldung automatisch Rechte dafür.

2.8 Planbuchungen

In COB-GX sind in der Tabelle busa in dem Feld cob_bukz die Primärbuchungen nach IM=Istbuchungen und PE=Planbuchungen aufgeteilt. Die Sekundärbuchungen sind auf die Tabellen vtbu und busa verteilt. Die sekundären Istbuchungen sind in der Tabelle vtbu, die sekundären Planbuchungen in der Regel in der busa mit der Bedingung fikrkey='97000'. Da es hier unterschiedliche Bedingungen geben kann, gibt es für diese Regel in SuperX eine Repository Variable 'COB_SEK_PLAN', in der die Bedingung steht. Ausgeliefert wird diese mit dem Inhalt: "fikrkey='97000'". Wenn dies an Ihrer Hochschule anders gehandhabt wird, ändern Sie diese Variable bitte dementsprechend.

Eingabe in COB-GX:

Die Eingabe solcher Planbuchungen in COB-GX sieht folgendermaßen aus:



In der Datenbank wird dann der Eintrag hinzugefügt:

VerbNr.	von	bis	nächste	Dauerauftragsbez.	Zweck2	KoSt	KoTr/Proj	KoErArt	Geldgeber	Betrag
1	1 2010	12 2011	12 2011	Budget 03020111		030201	03020111	60		5.000,00
2	1 2010	12 2011	12 2011	Budget 03020111		030201	03020111	61		3.000,00
4	1 2010	12 2011	12 2011	Budget 03020111		030201	03020111	68		6.000,00
5	1 2010	12 2011	12 2011	Budget 03020111		030201	03020111	62		557,00
6	1 2010	12 2011	12 2011	Budget 03020111		030201	03020111	66		56,00

2.9 Anpassungen der Berichte

2.9.1 Berichtsblatt Kennzahlen aus der Kostenrechnung (nur relevant für NRW)

Das Berichtsblatt betrifft in erster Linie die Hochschulen in NRW. Es besteht aus den Einzelabfragen Kostengrunddaten und Personal nach Landes-/Drittmitteln. Das XSL-Stylesheet für das **Berichtsblatt Kennzahlen aus der Kostenrechnung** kann angepasst werden (§\$U-

PERX_DIR/webserver/tomcat/webapps/superx/xml/cob/tabelle_html_12000.xml), es enthält den Briefkopf der Uni Duisburg-Essen. Das zugehörige html-Systemsheet liegt im gleichen Verzeichnis (tabelle_html_mswf.css). Hier werden Farben, Schriftarten etc definiert.

2.9.2 Auslastung (Lehrangebot / Lehrnachfrage)

In dem Bericht Auslastung (Lehrangebot / Lehrnachfrage) können Sie das Lehrangebot über die Studienplätze (cob_stupl) ermitteln. Sie können alternativ auch das Lehrangebot über die manuelle Schnittstelle des KENN Moduls eingeben, hier ist dies die Kennzahl LEHRE_SWS (Studienangebot (SWS)). Dazu müssen Sie aber auch die Konstante KENN_LEHRANG_MAN=1 setzen (Default ist 0). .

2.10 Entfernen des COB-Moduls

Wenn Sie das COB-Modul nicht mehr benötigen starten Sie das Script
\$SUPERX_DIR/db/module/cob/cob_modul_entfernen.x.

Dieses Script löscht alle Tabellen, Prozeduren und Abfragen aus der Datenbank, und löscht auch die Einträge im Themenbaum. Danach können Sie den Pfad \$SUPERX_DIR/db/module/cob löschen.

Wenn Sie nur die Inhalte der Daten- und Hilfstabellen des COB-Moduls löschen wollen (z.B. aus Datenschutzgründen), ohne das ganze Modul zu deinstallieren, können Sie dies mit folgendem Befehl tun:

DOSQL \$SUPERX_DIR/db/module/cob/cob_purge_pg.sql (für Postgres)

bzw.

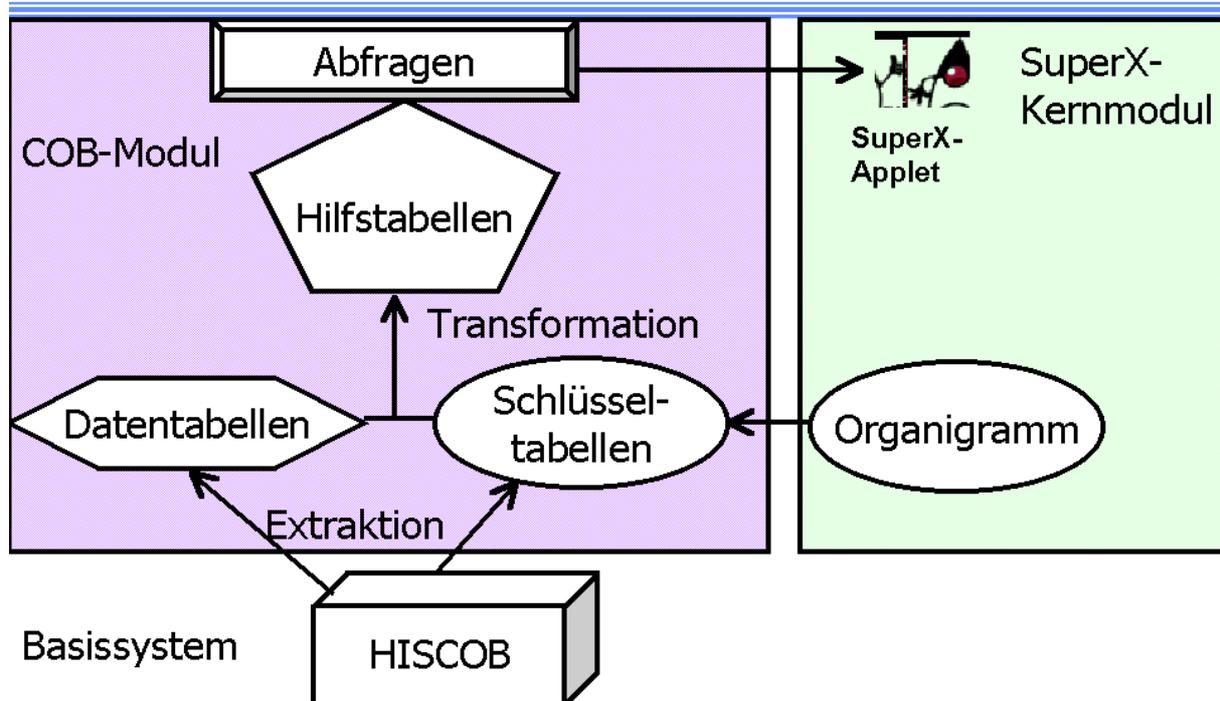
DOSQL \$SUPERX_DIR/db/module/cob/cob_purge_ids.sql (für Informix)

3 Bestandteile des COB-Moduls

Im COB-Modul sind die Komponenten von der Datenextraktion bis zur Präsentation enthalten. Die folgende Abbildung zeigt den Datenfluss.



Architektur des COB-Moduls



Die Daten werden aus dem Basissystem extrahiert, und die resultierenden Datentabellen werden mit Schlüsseln verknüpft. Daraus werden aggregierte Hilfstabellen erzeugt, die wiederum als Basis für die Abfragen dienen.

3.1 Datentabellen

Die wichtigsten Tabellen des COB- Moduls sind die Grundtabellen

- cob_busa <- busa
- cob_vtbu <- vtbu
- cob_umks <- umks
- cob_su_imp_stud <- su_imp_stud

Aus diesen Tabellen werden die wichtigsten Hilfstabellen vom COB-Modul erzeugt..

3.1.1 Tabelle cob_busa: Primärbuchungen

Die Tabelle `cob_busa` in SuperX ist ein Auszug der `busa`-Tabelle im COB-System. Das Feld `ch110_institut` ist für eine Verknüpfung zum Organigramm des Kernmoduls vorgesehen. Momentan enthält das Feld den gleichen Wert wie `instnr`.

Weitere Details siehe Merkmalsbeschreibung ([cob_sx_beschreibung](#)).

3.1.2 Tabelle cob_vtbu: Verteilbuchungen

Die Tabelle `cob_vtbu` in SuperX entspricht einem Auszug der `vtbu`-Tabelle im COB-System. Darüber hinaus enthält sie wie die Tabelle `cob_busa` die Felder `quell_ch110_inst`, und `ziel_ch110_inst`, die jeweils für Verknüpfungen zum SuperX-Organigramm vorgesehen sind.

Weitere Details siehe Merkmalsbeschreibung ([cob_sx_beschreibung](#)).

3.1.3 Tabelle cob_umks: Verzeichnis der Verrechnungssätze/Festpreise

Die Tabelle `cob_umks` in SuperX entspricht einer Kopie der `umks`-Tabelle im COB-System, die das Verzeichnis der Verrechnungssätze/Festpreise darstellt.

Weitere Details siehe Merkmalsbeschreibung ([cob_sx_beschreibung](#)).

3.1.4 Tabelle cob_su_imp_stud: Studierendendaten in COB

In der Tabelle `cob_su_imp_stud` werden die Importdaten von HISSOS nach HISCOB gehalten, die dann in der Abfrage **Studierende (gewichtet) für die Kostenrechnung** ausgewertet werden.

Wenn in dieser Abfrage keine Ergebnisse erscheinen, kann dies am automatischen Import von HISSOS nach HISCOB liegen. Bitte beachten Sie, dass Sie vor dem Export von HISCOB nach SuperX in HISCOB die Studierendensummen berechnen: Die Tabelle wird in COB beim Import der SOS Daten nur teilweise gefüllt. Erst beim "Berechnen der Studierendensummen" werden die `stugkey` - Felder gefüllt, die dann in der Auswertung genutzt werden.

Weitere Details zu der Tabelle siehe [Merkmalsbeschreibung](#).

3.1.5 Tabelle cob_imp_flaeche: Flächen in der KLR

Die Flächen werden komplett gelöscht und wieder neu eingefügt, bei jeder Laderoutine. Nur die Flächendaten vom jeweils letzten Importdatum gelangen nach Fleda.

3.2 Schlüsseltabellen

Die Schlüsseltabellen stellen die Metadaten für das COB-Modul dar. Sie sorgen für eine sinnvolle Aggregation der Hilfstabellen. Sie werden direkt von COB übernommen und dürfen nicht manuell nachgearbeitet werden. Nur die Tabelle `cob_inst` dient als Vorlage für das Organigramm und kann falls gewünscht, durch das Skript `preparation_cob_inst.sql` nach Ihren Wünschen angepasst werden. Es ist aber auch möglich, das Kostenstellenverzeichnis aus COB ohne Änderung in das SuperX-Organigramm einzufügen (s.u.).

3.2.1 Die Tabelle `cifx`

Die Tabelle `cifx` ist Teil des Kernmoduls und wird mit Schlüsseln aus COB gefüllt.

key	hs	Bereich	Bedeutung	Schlüsseltabelle
38	0	COB SVA	Hochschulnummer	<code>k_hochschule</code>

3.2.2 Die Tabellen `cob_cif` / `cob_cifx`

Die Tabellen `cob_cif`/`cob_cifx` sind "angereicherte" und nur für COB gültige Varianten der Tabellen `cif`/`cifx` mit den zusätzlichen Feldern "ueberkey" und "sort_key"; sie werden mit Schlüsseln aus COB gefüllt. Die Werte für den key entsprechen denen der Kernmodul-`cif`, COB- interne Schlüssel werden mit key von 700-720 verschlüsselt.

key	hs	Bedeutung	Schlüsseltabelle
30*	<>0	Verzeichnis der Studienfaecher aus HISSOS	<code>k_fach</code>
109*	<>0	Verzeichnis der Verguetungsgruppen	<code>k_bvlgruppe</code>
700*	<>0	Verzeichnis der Flächen-Nutzungsarten nach DIN 277	<code>k_nadin</code>
701*	<>0	Verzeichnis der Kostenflaechenarten	<code>k_kfa</code>
710*	<>0	Schlüsseltabelle fuer (Studien-)Hauptfach/Nebenfach-Kennzeichen	<code>k_kzfa</code>
711*	<>0	Verzeichnis der (SOS-)Studienarten	<code>k_stuart</code>
712*	<>0	Verzeichnis der (SOS-)Studienformen	<code>k_stufrm</code>
713	<>0	Semesterbezeichnung (WS/SS)	<code>zeit</code>
714	<>0	Verzeichnis der Verteilschritte	<code>vari</code>

3.2.3 Die Tabelle `cob_inst` (Verzeichnis der Kostenstellen)

Die Tabelle `cob_inst` ist ein Auszug der COB-Tabelle `inst`, wobei die Felder und Datentypen an das SuperX-Organigramm im Kernmodul angeglichen sind. Hochschulen, die SuperX erstmalig einführen wollen, können also eine vorhandene `inst`-Tabelle benutzen, um das Organigramm für SuperX zu erzeugen.



Wichtig: Damit der Kostenstellenplan aus der `inst`-Tabelle in COB übernommen werden kann, muss die Tabelle genau ein root-Element enthalten, das in dem `uebinst_nr`-Feld ein leeres Element enthält (null oder ""). Dieses Element ist notwendig, um den Baum aufbauen zu können.

Da für den ordnungsgemäßen Betrieb des COB-Moduls die Kostenstellen auch im Organigramm vorkommen müssen, empfehlen wir, beim den Datenupdate des Cob-Moduls mit dem Parameter `--mit-organigramm` laufen zu lassen. In `$$SUPERX_DIR/db/module/cob/`

```
cob_update.x --mit-organigramm
```

Unabhängig vom Update des gesamten Cob-Moduls können Sie das Organigramm aktualisieren mit dem Script

```
$$SUPERX_DIR/db/module/cob/update_organigramm.x
```

Das Organigramm wird dann aus `cob_inst` gebildet.

Weitere Details siehe [Merkmalsbeschreibung](#).

3.2.4 Die Schlüsseltabelle `cob_fikr` (Verzeichnis der Kosten-/Erlösarten)

Eine Schlüsseltabelle mit dem Kostenartenplan. Diese Tabelle wird aus COB entladen und leicht abgewandelt.

Weitere Details siehe [Merkmalsbeschreibung](#).

Die Schlüsseltabelle enthält zusätzlich das Feld "Hierarchieebene". Sie wird in Abfragen bei der Kostenartenrechnung benutzt.



Wichtig: Damit der Kostenartenplan aus der `fikr`-Tabelle in COB übernommen werden kann, muss die Tabelle genau ein root-Element enthalten, das in dem `ueberg`-Feld ein leeres Element enthält (null oder ""). Dieses Element ist notwendig, um den Baum aufbauen zu können.

3.2.5 Die Tabelle `cob_proj` (Verzeichnis der Kostenträger/Projekte)

Die Tabelle `cob_proj` ist ein Auszug der COB-Tabelle `proj`.



Wichtig: Damit der Kostenträgerplan aus der `proj`-Tabelle in COB übernommen werden kann, muss die Tabelle genau ein `root`-Element enthalten, das in dem `ueberkey`-Feld ein leeres Element enthält (null oder ""). Dieses Element ist notwendig, um den Baum aufbauen zu können.

Weitere Details siehe [Merkmalsbeschreibung](#).

3.2.6 Die Tabelle `cob_extkotr` (Verzeichnis der externen Kostenträger)

Die Tabelle `k_extkotr` wird regelmäßig mit aus COB entladen und mit dem Script

```
$$SUPERX_DIR/db/module/cob/schluesseltabellen/trans_cob_extkotr.sql
```

für SuperX angepasst.

Weitere Details siehe [Merkmalsbeschreibung](#).

3.2.7 Die Schlüsseltabelle `cob_kpum` (Verzeichnis der Leistungsarten)

Die Schlüsseltabelle `cob_kpum` enthält das Verzeichnis der Leistungsarten. Die Tabelle stammt aus COB.

Weitere Details siehe [Merkmalsbeschreibung \(cob_sx_beschreibung\)](#).

3.2.8 Die Schlüsseltabelle `cob_proj_to_inst`

Die Tabelle `cob_proj_to_inst` enthält eine Zuordnung von Projekten zu Institutionen und kommt ebenfalls aus COB. Wenn ein einschränkter User nur bestimmte Institutionen (Kostenstellen) sehen darf, werden bei Kostenträgersichten auch nur die in dieser Tabelle zugeordneten Projekte dargestellt.

Weitere Details siehe [Merkmalsbeschreibung](#).

3.2.9 Die Schlüsseltabelle `aggregation`

Die Tabelle `aggregation` ist eigentlich Bestandteil des Kernmoduls. Für HISCOB werden die relevanten Dimensionen aus der COB-Tabelle `zeit` übernommen. Die Werte wiederum werden ist fast allen COB-Abfragen für den Dialog Zeitraum verwendet.

3.2.10 Die Schlüsseltabelle cob_drittmittel (obsolet)

Die Tabelle `cob_drittmittel` wird ohne Änderung aus COB übernommen (`drittmittelherk`) und enthält diejenigen Titel / Kapitel, die Haushaltsbuchungen aus Drittmitteln kennzeichnen.

Beim Update von COB wird diese Schlüsseltabelle neu übernommen. Sie wird für die Kennzeichnung von Drittmitteln verwendet, wenn Quellsystem HISCOB5 ist, oder bei den folgenden HISCOB-Versionen bei Daten von vor 2003 stammen.

Weitere Details siehe Merkmalsbeschreibung.

SuperX geht wie COB davon aus, dass das Kapitel fünfstellig und die Titelgruppe zweistellig ist. Das Haushaltsjahr muss vierstellig angegeben werden.

3.2.11 Die Schlüsseltabelle cob_stug (Verzeichnis der Studiengänge)

Die Tabelle `cob_stug` wird ebenfalls ohne Änderung aus COB übernommen und enthält die Studiengänge und deren Regelstudienzeiten, Curricularnormwerte und Lehreinheits-Schlüssel. Die Tabelle wird auch für die Erzeugung der Hilftabelle `lehr_stg_ab` genutzt, die Teil des SOS-Modul von SuperX ist.

Weitere Details siehe [Merkmalsbeschreibung](#).

3.2.12 Die Schlüsseltabelle cob_geldgeber

Die Tabelle entspricht der Tabelle `gege` in COB.

3.2.13 cob_alt_hier, cob_trees, cob_tree_cfg, cob_alt_keys

Die Tabellen `cob_alt_hier`, `cob_trees` und `cob_tree_cfg` werden aus COB für die Darstellung alternativer Hierarchien übernommen.

In `cob_alt_keys` werden die Tabellen `alt_inst`, `alt_fikr` und `alt_proj` aus HISCOB zusammengefasst.

3.3 Hilfstabellen

Hilfstabellen sind aggregierte Datentabellen, die aus den Basisdatentabellen gebildet werden. Sie erhöhen die Performance der Abfragen, da die Tabellen sinnvoll für einige Abfragen summiert werden.

Wie die COB- Datentabellen werden die Hilfstabellen bei jedem Update komplett neu erzeugt. Je nach Datenvolumen und Rechnerkapazität können sehr unterschiedliche Laufzeiten

resultieren. Bei der Installation und für erste Tests sollte deshalb vorsorglich ein abgeschottetes Rechnersystem verwandt werden. Das Script zum Erzeugen der Hilfstabellen lautet

```
$SUPERX_DIR/db/module/cob/hilfstabellen/update_hilfstabellen.x
```

und wird, wenn die Datentabellen ohne Fehler gefüllt werden, im Script

```
$SUPERX_DIR/db/module/cob/cob_update.x gestartet.
```

3.3.1 cob_busa_aggr

Die Prozedur `sp_cob_busa_aggr` erzeugt aus der Tabelle `cob_busa` eine aggregierte Tabelle mit den Kosten / Erlösen einer Institution bzw. eines Projektes / Studiengangs pro Monat, Jahr und Kostenart. Die Beträge werden ja nach Kostenart als Kosten- oder Erlösbuchung summiert.

Weitere Details siehe Merkmalsbeschreibung (`cob_sx_beschreibung`).

3.3.2 cob_vtbu_aggr

Analog zur obigen Prozedur erzeugt die Prozedur `sp_cob_vtbu_aggr` eine aggregierte Tabelle mit den Verteilbuchungen einer abgebenden und einer empfangenden Institution bzw. Projekts / Studiengangs pro Monat und Jahr.

Weitere Details siehe Merkmalsbeschreibung (`cob_sx_beschreibung`).

3.3.3 cob_busa_vtbu

Die Prozedur `sp_cob_busa_vtbu` erzeugt aus den Hilfstabellen `cob_busa_aggr` und `cob_vtbu_aggr` eine aggregierte Zusammenstellung der Kosten und Erlöse auf primärer und sekundärer Ebene. Die resultierende Tabelle wird für die Abfrage Kosten und Verteilbuchungen: Schnellinformation verwendet.

Besonderheit NRW:

Wegen den besonderen Anforderungen in NRW wurden die Sekundärkosten auf Ziel-Kostenarten aufgesplittet nach Sekundärkosten auf Ziel-Kostenarten (`vert_ziel`) und Kosten, bei denen das Feld `zfikr` gefüllt ist (`vert_beih`); dies ist sinnvoll für Auswertungen, wo z. B. Beihilfen, die zentral gebucht werden, künstlich den Kostenstellen zugerechnet werden. Die Abfrage Kosten und Verteilbuchungen (inkl. Beihilfen) weist diese Kosten separat aus, in der Abfrage Kosten und Verteilbuchungen (Schnell-Info) werden die Kosten addiert.

Weitere Details siehe Merkmalsbeschreibung (`cob_sx_beschreibung`).

3.4 Alternative Hierarchien / Sichten

In HISCOB angelegt alt. Hierarchien bzw. Auswertungshierarchien (mit eingestelltem Aufklappstatus) werden in den Tabellen `cob_alt_hier`, `cob_alt_keys`, `cob_trees` und `cob_tree_cfg` übernommen.

Um diese im SuperX-COB-Modul verwenden zu können, wird die Sichten-Funktionalität des Kernmoduls benutzt (vergl. Handbuch Kernmodul).

Die Prozedur `sp_cob_alt_hier`, die während des COB-Datenupdates aufgerufen wird, macht Einträge in die Sichtentabelle und zwar für die reguläre „Standardsicht“ auf Kostenstellen,-arten und –träger als auch für konfigurierte Auswertungshierarchien. Das heißt, dass das nicht alle in COB definierten alt. Hierarchien angezeigt werden, sondern nur solche für die eine „Auswertungshierarchie“ mit spezifischem Aufklappstatus angelegt wurde. Dadurch werden doppelte Anzeigen von alt. Hierarchien vermieden.

Wenn neue alt. Hierarchien aus COB übernommen wurden, müssen sich Benutzer des XML-Frontends neu anmelden, nachdem entweder Tomcat neu gestartet wurde oder der Server-Cache aktualisiert unter <http://rechnername:8080/superx/servlet/SuperXManager>.

3.4.1 Prozedur sp_cob_alt_hier

Die Prozedur sp_cob_alt_hier, die während des COB-Datenupdates aufgerufen wird, macht Einträge in die Sichtentabelle und zwar für die reguläre „Standardsicht“ auf Kostenstellen,-arten und -träger als auch für konfigurierte Auswertungshierarchien. Das heißt, dass das nicht alle in COB definierten alt. Hierarchien angezeigt werden, sondern nur solche für die eine „Auswertungshierarchie“ mit spezifischem Aufklappstatus angelegt wurde. Dadurch werden doppelte Anzeigen von alt. Hierarchien vermieden.

3.4.2 Prozedur sp_cob_inst_hier

Die Prozedur wird benutzt, um die reguläre Kostenstellen-Sicht für einen bestimmten Stand aufzubauen.

Sie benutzt die Prozedur sp_user_orga aus dem SuperX-Kernmodul, um Rechteeinschränkungen vorzunehmen: welche Kostenstellen darf der User sehen?

Alternative Hierarchien werden Java-intern basierend auf der regulären Sicht mittels der Einträge in der Tabelle cob_alt_keys aufgebaut.

Die tid der ausgewählten Sicht wird standardmäßig als Parameter übergeben, aber bisher noch nicht ausgewertet.

3.4.3 Prozedur sp_cob_ktr_hier

Die Prozedur wird benutzt, um die reguläre Kostenträger-Sicht für einen bestimmten Stand aufzubauen.

Sie benutzt die Prozedur sp_user_orga aus dem SuperX-Kernmodul sowie die Tabelle cob_proj_to_inst um Rechteeinschränkungen vorzunehmen: basierend auf den erlaubten Kostenstellen – welche Kostenträger darf der User sehen?

Alternative Hierarchien werden Java-intern basierend auf der regulären Sicht mittels der Einträge in der Tabelle cob_alt_keys aufgebaut.

Die tid der ausgewählten Sicht wird standardmäßig als Parameter übergeben, aber bisher noch nicht ausgewertet.

3.4.4 Prozedur sp_cob_fikr_hier

Die Prozedur wird benutzt, um die reguläre Kostenarten-Sicht für einen bestimmten Stand aufzubauen.

Rechteeinschränkungen auf einzelne Kostenarten sind bisher nicht vorgesehen.

Alternative Hierarchien werden Java-intern basierend auf der regulären Sicht mittels der Einträge in der Tabelle cob_alt_keys aufgebaut.

Die tid der ausgewählten Sicht wird standardmäßig als Parameter übergeben, aber bisher noch nicht ausgewertet.

4 Anhang

4.1 Geldgebersichten und Drittmittel

Die folgende Tabelle aus einer Arbeitsgruppe der HIS GmbH und diversen Hochschulen in BadenWürttemberg zeigt drei verschiedene Geldgebersichten:

Finanzierung einer Stelle, Bundesstatistik StBA 5.2		Drittmittelart des Drittmittelzusatzbogens für Drittmittelinnahmen der jährl. Hochschulfinanzstatistik bis 2005, etwas verfeinert für Zwecke der Nachweisung für die Verzinsung		Grunddatenkatalog des MWK Baden-Württemberg ab 2006 vom 06.04.2006 auf Basis des Drittmittelzusatzbogens zur Hochschulfinanzstatistik des StaLa		
gege.bund_geldgeber (2-stellig) "Finanzierung (StBA 5.2)" verwendet für Stellenstatistik (SVA)		gege.fikey (3-stellig) "Drittmittelart (HFinSt DZ)" verwendet für HFinSt-Drittmittelzusatzbogen und FSV-Bericht für Drittmittelinnahmen für die Verzinsung für FH Ba-Wü		gege.klr_geldgeber (3-stellig) "Mittelherkunft (KLR)" verwendet für das Ministerium (COB, SuperX)		
Schl	Art der Finanzierung	Schl	Drittmittelart	Kategorie	Schl	Drittmittelart
		01	1. von Deutsche Forschungsgemeinschaft	öffentlich Inland		
05	Drittmittel DFG	02	1(a) von DFG für Sonderforschungsbereiche (Teilmenge von 01)	öffentlich Inland	301	DFG für Sonderforschungsbereiche
		03	1(b) von DFG für Normal- und Schwerpunktverfahren, Graduiertenkollegs usw. (ohne Habilitandenstipendien, Rechenanlagen, Großgeräte, Baumaßnahmen und Heisenberg-Programm) (Teilmenge von 01)	öffentlich Inland	302	von DFG für Normal- und Schwerpunktverfahren, Graduiertenkollegs usw. (ohne Habilitandenstipendien, Rechenanlagen, Großgeräte, Baumaßnahmen und Heisenberg-Programm)
03	Drittmittel Bund	04	2. vom Bund für die Förderung der "Wissenschaft", "Forschung und Lehre" (ohne Zuweisungen nach dem Hochschulbauförderungsgesetz, Überlastprogramm, dem Graduierten- und Bundesausbildungs-förderungsgesetz sowie für die sonstige Förderung von Studenten)	öffentlich Inland	303	vom Bund (ohne Zuweisungen nach dem Hochschulbauförderungsgesetz, Überlastprogramm, dem Graduierten- und Bundesausbildungsförderungsgesetz sowie für die sonstige Förderung von Studenten)
11	ABM	05	3. von der Bundesanstalt für Arbeit, soweit hieraus Personal mit Lehr- und Forschungsaufgaben finanziert wird	öffentlich Inland	304	von der Bundesagentur für Arbeit soweit hieraus Personal mit FuE- Aufgaben finanziert wird
04	Drittmittel Land	06	4. von Ländern für Lehr- oder Forschungszwecke (ohne Mittel vom Träger der Hochschule)	öffentlich Inland	305	von Ländern für Lehr- oder Forschungszwecke (ohne Mittel vom Träger der Hochschule)
07	sonstige öffentliche Drittmittel	07	5. von Gemeinden, Gemeinde- und Zweckverbänden für Lehr- oder Forschungszwecke (d.h. ohne Erstattungen für Sportanlagen, Bibliotheken u. dgl.)	öffentlich Inland	306	von Gemeinden, Gemeinde- und Zweckverbänden (d.h. ohne Erstattungen für Sportanlagen, Bibliotheken u. dgl.)
		08	6. von sonstigen öffentlichen Bereichen für LuF-Zwecke (z.B. ERP, Lastenausgleichsfonds, Sozialversicherung)	öffentlich Inland	307	von sonstigen öffentlichen Bereichen (z.B. ERP, Lastenausgleichsfonds, Sozialversicherung)
06	Drittmittel EU	09	7. von internationalen Organisationen (z.B. EU, OECD, UN)	sonstige	308	von anderen intern. Organisationen (z.B. OECD, UN)
07	sonstige öffentliche Drittmittel	091	EU-Mittel (als Teilmenge von 09)	EU-Mittel	309	EU-Mittel
08	Drittmittel VW-Stiftung	10	8. von Stiftungen u. dgl. für Lehr- oder Forschungszwecke	privat	310	von Stiftungen u. dgl. für Lehr- und Forschungszwecke
09	sonstige private Drittmittel					
09	sonstige private Drittmittel	11	9. von der gewerblichen Wirtschaft und sonstigen nichtöffentlichen Bereichen für Lehr- und Forschungszwecke (ohne Einnahmen für Materialprüfungen u. dgl., aus Veröffentlichungen, Gebühren, aus wirtschaftlicher Tätigkeit und aus Vermögensveräußerungen)	privat	311	von der gewerblichen Wirtschaft und sonstigen nichtöffentlichen Bereichen für Lehr- und Forschungszwecke (ohne Einnahmen für Materialprüfungen u. dgl., aus Veröffentlichungen, Gebühren, aus wirtschaftlicher Tätigkeit und aus Vermögensveräußerungen)
Erläuterungen zu den darüberliegenden Spalten						
Es gibt weitere Schlüssel für Nicht-Drittmittel, die nicht aufgelistet sind.		Der 2-stellige Schlüssel in der Spalte "Schl." ist derjenige des Drittmittelzusatzbogens der Hochschulfinanzstatistik. Die Kategorisierung ist diejenige für den FSV-Bericht an das MWF für die		Für die Einteilung der Drittmittel für den Grunddatenkatalog wird der folgende 3-stellige Schlüssel vorgeschlagen, der auch die Nicht-Drittmittel einteilt, weil auch über diese Berichte erstellt werden müssen. Eine		

