

**Inhaltsverzeichnis**

Inhaltsverzeichnis	1
Verbindung zu externen Datenbanken (operatives Reporting)	2
Verbindung mit der externen Datenbank	2
Passwortverschlüsselung	3
Sicherheitsmaßnahmen	3
Allgemeine Rechteeinschränkung	3
Rechte auf einzelne Tabellen oder Spalten	3
Nutzung der dbconnections	4
Maskenfelder	4

# Verbindung zu externen Datenbanken (operatives Reporting)

---

SuperX/BI kann in den Standardberichten auch externe Datenbanken direkt anbinden, um z.B. direkt Berichte auf dem operativen System auszuführen. Dies ist insbesondere dann praktisch, wenn die zugehörigen Daten

- noch nicht per Konnektor vom Vorsystem zur BI übertragen kann
- die Daten aktueller als vom Vortag benötigt werden

## Verbindung mit der externen Datenbank

---

Die Verbindungen zu externen Datenbanken werden in der Tabelle **dbconnections** erfasst.

Um [dbconnections](#) zu bearbeiten kann ein Formular genutzt werden. Dazu suchen wir zunächst die Tabelle über die Maske "Tabelle suchen":

imgsrcTabelleSuchen.pngwidth600captionTabelle suchen

Es eignet sich an dieser Stelle die Stichwortsuche:

imgsrcStichwortsuche.pngwidth600captionStichwortsuche

Um die Tabelle zu bearbeiten wird der Button auf der rechten Seite geklickt:

imgsrcTabelleBearbeiten.pngwidth600captionTabelle bearbeiten

Es öffnet sich ein neues Fenster mit dem Bearbeitungsformular:

imgsrcBearbeitungsformularDBConnections.pngwidth600captionBearbeitungsformular DBConnections

In diesem Falle ist bereits eine externe Datenbankverbindung angelegt. Es können die Details dieser Datenbankverbindung bearbeitet oder ein neuer Datensatz angelegt werden.

Beim Anlegen eines neuen Datensatzes werden zunächst nur die Felder "Id" und "Name" gefüllt:

imgsrcNeuerDatensatzDBConnections.pngwidth600captionNeuer Datensatz DBConnections

Anschließend können die Details dieser Verbindung bearbeitet werden:

imgsrcDBConnectionsDetailsBearbeiten.pngwidth600captionDBConnections Details bearbeiten

Wichtig: Nach Änderungen an den Connections muss der SuperXManager-Cache aktualisiert werden, damit diese neu aufgebaut werden.

## Passwortverschlüsselung

---

Das Passwort kann entweder im Klartext angegeben werden oder verschlüsselt.

Dazu kopieren Sie unter Linux eine db.properties nach db\_test.properties bearbeiten diese bis auf das Passwort und rufen auf :

propadmin.x db\_test.properties. Geben Sie das Passwort ein, es wird ein Verbindungsaufbau probiert und bei Erfolg das Passwort verschlüsselt.

Tragen Sie es dann ohne \ statt des Klartextpassworts ein, z.B. sx\_des#-107#-62#90#114#112#-9#-73#86#116#120#-97#-28#-54#-55#11#8

Nach Änderungen an den Connections muss der Webanwendung-Manager-Cache aktualisiert werden, damit diese neu aufgebaut werden.

## Sicherheitsmaßnahmen

---

### Allgemeine Rechteeinschränkung

---

Es bietet sich an. Im Vorsystem eine spezielle Kennung anzulegen, die nur Leserecht auf Datenbankobjekte hat.

Beim Zugriff auf HISinone könnte man die Kennung nutzen, die auch im qjsserver (databases.xml) eingerichtet ist. Diese Kennung hat maximale Rechte, was wir aber für BI nicht brauchen / wollen

Besser wäre, wenn Sie eine eigene Kennung einrichten (lassen), die nur Leserecht im Vorsystem hat. Hier ein Beispiel für Postgres/HISinOne, die SQL Statements haben wir bereits vorbereitet, nehmen wir an die Kennung lautet "biuser":

```
CREATE USER biuser WITH NOSUPERUSER PASSWORD 'password';
GRANT USAGE ON SCHEMA hisinone TO biuser ;
GRANT SELECT ON ALL TABLES IN SCHEMA hisinone TO biuser ;
ALTER DEFAULT PRIVILEGES IN SCHEMA hisinone GRANT SELECT ON TABLES TO biuser ;
```

Hintergrundinfos siehe [http://www.superx-projekt.de/doku/kern\\_modul/admin/f\\_Postgres-NutzerkennungenmitLeserecht.htm](http://www.superx-projekt.de/doku/kern_modul/admin/f_Postgres-NutzerkennungenmitLeserecht.htm)

### Rechte auf einzelne Tabellen oder Spalten

---

Zu Datenschutzzwecken kann man eingeschränkte User anlegen, die nur einzelne Tabellen oder auch nur Views darauf sehen dürfen.

Dabei kann z.B. auch ein View angelegt werden, der nicht den vollständigen Namen von Prüflingen enthält, sondern nur die Anfangsbuchstaben.

```
create view pos_schwarz as
SELECT L.mtknr,
L.abschl,
A.dtxt as abschluss_str,
L.stg,
G.dtxt as stg_str,
```

```
L.pversion,  
L.pstatus,  
L.pnote,  
substring(S.nachname from 1 for 1) as nachname,  
substring(S.vorname from 1 for 1) as vorname,  
F.hrst,  
L.psem  
FROM k_stg G, k_abint A, sos S, lab L left outer join stg F  
on (L.mtknr=F.mtknr  
and F.semester=L.psem  
and F.stgnr=L.stgnr  
)  
where G.refstg=L.stg  
and A.abint=L.abschl  
and S.mtknr=L.mtknr  
and L.pnr=9000;
```

Für diesen View könnte man dann mit

```
grant select on pos_schwarz to biuser;
```

Leserecht vergeben.

## Nutzung der dbconnections

---

Es können ganze Ergebnistabellen sowie einzelne Maskenfelder auf das operative System zugreifen. Wenn eine Abfrage auf eine externe Datenbank gehen soll, kann man dies derzeit spezifizieren

- felderinfo.relation
- felderinfo.defaultwert
- maskeninfo.select\_stmt
- maskeninfo.hinweis

Im jew. SQL muss sich ein Kommentar in einem bestimmten Format befinden, die Syntax lautet

```
--#[dbid:ANGELEGENE_ID]#
```

Die angegebene ID bezieht sich auf die erste Spalte in der Tabelle dbconnections.

Wenn z.B. --#dbid:sospos# angegeben ist, wird nach der ID sospos gesucht.

Man hat aber auch die Möglichkeit pro Maske eine "Spezialmodulversion" zu hinterlegen.

Wenn man z.B. in Maske 888889000 angegeben hat --#dbid:sospos# wird zunächst geschaut, ob es eine spezielle Datenbankverbindung mit der id sospos\_888889000 gibt, falls ja wird diese genommen, falls nein, wird nach einer Datenbankverbindung mit id sospos gesucht.

Falls keine Verbindung gefunden wurde oder diese nicht aufgebaut werden kann, erscheint eine Fehlermeldung.

## Maskenfelder

---

Nachdem die Verbindungsinformationen in [dbconnections](#) erfasst sind, kann in Masken und Maskenfeldern auf die entsprechende Datenbank zugegriffen werden.

Zur Erläuterung sei ein Beispiel eines Maskenfeldes "Studiengang" betrachtet, welches auf eine externe Datenbank zugreift:

imgsrcVerbindungExterneDB.pngwidth600captionVerbindung externe DB

Die grundlegende Syntax lautet `--#dbid:ANGELEGENE_ID#`. ANGELEGENE\_ID wird in diesem konkreten Beispiel durch `hisinone` ersetzt, da `hisinone` zuvor in [dbconnections](#) als ID angegeben wurde.

Bei `hisinone.course_of_study` handelt es sich nicht um die `dbid`, sondern um das Schema `hisinone` auf der Datenbank.