
sM-Client Projekt

Aktualisierungsinstruktionen für R2.0.4

Finale Version

SSK / BSV eAHV/IV





COPYRIGHT

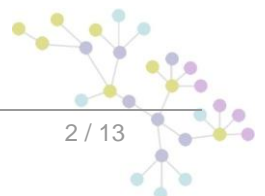
© Copyright - 2010 Cambridge Technology Partners. All Rights Reserved.

For Internal use only

No part of this publication may be reproduced, transmitted, transcribed, stored in a retrieval system or translated into any language in any form by any means without the written permission of Cambridge Technology Partners, Inc.

While every precaution has been taken in the preparation of this document, Cambridge Technology Partners, Inc. assumes no responsibility for errors, omissions, or for damages resulting from the use of the information herein.

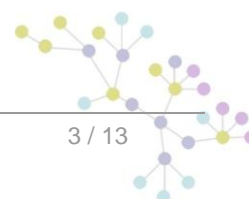
Products or corporate names may be trademarks or registered trademarks of other companies and are used only for the explanation and to the owner's benefit, without intent to infringe.





INHALTSVERZEICHNIS

1	AKTUALISIERUNGSINSTRUKTIONEN FÜR R2.0.4	5
1.1	ZWECK DES DOKUMENTS.....	5
1.2	ÜBERSICHT	5
1.3	AKTUALISIERUNG	5
2	MYSQL	6
2.1	AUSFÜHRUNG	6
2.2	SQL SKRIPT	6
3	ORACLE.....	8
3.1	AUSFÜHRUNG	8
3.2	SQL SKRIPT	8
4	MICROSOFT SQL SERVER	10
4.1	AUSFÜHRUNG	10
4.2	SQL SKRIPT	10





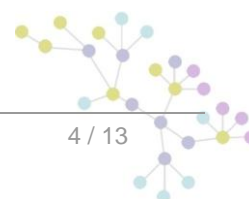
DOKUMENTENVERWALTUNG

Historie

Version	Geändert	Beschreibung	Datum	Validator
1.0	Sylvain Berthouzoz	Finale Version	20.01.2010	

Referenzen

ID	Name	Version	Dokument
01	Installations-handbuch	V1.3	Installations-Handbuch_sM-Client_1.3.pdf





1 AKTUALISIERUNGSIKTRUKTIONEN FÜR R2.0.4

1.1 Zweck des Dokuments

Das nachfolgende Dokument beschreibt sämtliche notwendigen Installationsschritte und Konfigurationen, um den sM-Client von R1.0.x auf R2.0.4 (Release 2010-1) zu aktualisieren.

Im Release 2.0.4 ist die wichtigste Neuheit die Einführung der Mandantenfähigkeit. Das heisst man kann mehrere sM-Client Instanzen auf dem gleichen JBoss Applikation-Server verwalten.

1.2 Übersicht

Die Aktualisierung des sM-Clients erfordert verschiedene Installationsschritte. Es werden Windows-, Linux- und Unix-Betriebssysteme unterstützt.

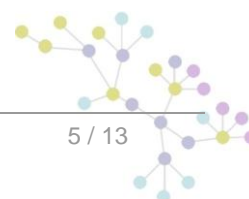
Es sind folgende Teile zu aktualisieren:

- sM-Client WAR-File
- sM-Client Datenbank
- Das Skript enthält am Anfang zwei Variablen. Diese entsprechen Ihrer aktuellen Sedex-ID und der entsprechenden Name.

1.3 Aktualisierung

- 1) JBoss stoppen
- 2) Datenbank mittels Ausführung des Skript aktualisieren. Siehe dafür den Kapiten entsprechend ihre Datenbank (MySQL, Oracle oder MS SQL Server)
- 3) WAR-Datei ersetzen
 - a. alte Version von smclient.war löschen von JBOSS_HOME/Server/default/deploy
 - b. neue Version herunterladen. (smclient-r2.0.4.war for mysql)
 - c. WAR-File ins Verzeichnis JBOSS_HOME/Server/default/deploy kopieren
- 4) JBoss starten

Bitte beachten Sie, dass bei einem Download mittels Internet Explorer, die File-Extension von .war auf .zip geändert werden kann. In diesem Falle ist nach erfolgtem Download einfach die Endung wieder in .war zu ändern. **Das heruntergeladene File darf nicht mittels Win-Zip (oder ähnlichen Programmen) ausgepackt werden.**





2 MySQL

Das MySQL Skript „*mandant_update_mysql*“ muss mittels den mysql Client ausgeführt werden.

Datei „*mandant_update_mysql*“ öffnen und editieren.

- 1) Wenn die Datenbank nicht „*smclient*“ heisst, editieren Sie die folgende Zeile:

```
use smclient;
```

- 2) Die Variablen „sedexId“ und „mandantName“ editieren: Hier bitte Ihre aktuelle Sedex-ID und der entsprechende Name einfügen

```
SET @sedexId = 'yourSedexId';
```

```
SET @mandantName = 'default';
```

Ein Mandant ist sozusagen eine Instanz von einem sM-Client.

Sedex-ID war bevor im [smclient_installDir]/config/config.properties als sedexid konfiguriert. MandantName beschreibt ihre Organization.

Zum Beispiel ist „2-ZH-5“ ist ein Sedex-ID und „Steueraamt Zürich“ wäre der Mandant-Name, die Beschreibung von 2-ZH-5

Zum Beispiel:

```
SET @sedexId = '2-ZH-5';
```

```
SET @mandantName = 'Steueramt Zürich';
```

2.1 Ausführung

Eine Kommando-Fenster an der Stelle öffnen, wo das Skript liegt und dann das mysql-Programm ausführen (Ändern Sie den Benutzernamen, falls nötig):

```
mysql -u smclientUser -p
```

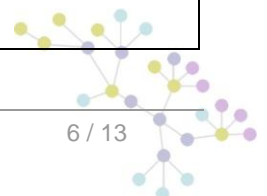
Skript ausführen

```
mysql> source mandant_update_mysql.sql
```

2.2 SQL Skript

```
-- Syntax is for mysql query with mysql program
use smclient;
-- create smMandant table
CREATE TABLE `smMandant` (`sedexId` varchar(255) NOT NULL, `name` varchar(255)
DEFAULT NULL, `active` bit DEFAULT false, PRIMARY KEY (`sedexId`));
SET @sedexId = 'yourSedexId';
SET @mandantName = 'default';
INSERT INTO smMandant VALUES (@sedexId,@mandantName,true);

-- alter tables (smUser, Message, smAudit)
ALTER TABLE smUser ADD COLUMN `sedexId` varchar(255);
```

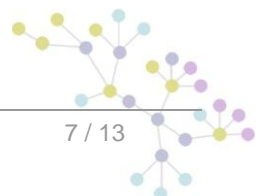




```
ALTER TABLE Message ADD COLUMN `sedexId` varchar(255) NOT NULL;
ALTER TABLE smAudit ADD COLUMN `sedexId` varchar(255) NOT NULL;

UPDATE smUser SET sedexId = @sedexId WHERE sedexId IS NULL OR sedexId LIKE '';
UPDATE Message SET sedexId = @sedexId WHERE sedexId IS NULL OR sedexId LIKE '';
UPDATE smAudit SET sedexId = @sedexId WHERE sedexId IS NULL OR sedexId LIKE '';
-- alter table configuration add column sedexId
ALTER TABLE configuration ADD COLUMN `sedexId` varchar(255) NOT NULL;
UPDATE configuration SET sedexId = @sedexId WHERE sedexId IS NULL OR sedexId
LIKE '';
-- alter table configuration set primary key.
ALTER TABLE configuration DROP PRIMARY key;
ALTER TABLE configuration ADD PRIMARY key(`sedexId`, `keycolumn`);

-- alter table ProcessMapping
ALTER TABLE ProcessMapping DROP PRIMARY key;
ALTER TABLE ProcessMapping ADD COLUMN `sedexId` varchar(255) NOT NULL;
UPDATE ProcessMapping SET sedexId = @sedexId WHERE sedexId IS NULL OR sedexId
LIKE '';
ALTER TABLE ProcessMapping ADD PRIMARY key(`messageId`, `sedexId`, `processId`);
-- set foreign key
ALTER TABLE configuration ADD FOREIGN KEY (`sedexId`) REFERENCES
smMandant(`sedexId`);
ALTER TABLE Message ADD FOREIGN KEY (`sedexId`) REFERENCES smMandant(`sedexId`);
ALTER TABLE smAudit ADD FOREIGN KEY (`sedexId`) REFERENCES smMandant(`sedexId`);
-- Uncomment the next line if the underlying mysql don't use auto_commit
-- commit;
```





3 ORACLE

Das sqlplus Skript „*mandant_update_oracle*“ muss mittels den sqlplus Client ausgeführt werden.

Datei „*mandant_update_oracle*“ öffnen und die **markierten** Werte editieren.

- 1) Die Variablen „sedexId“ und „mandantName“ editieren: Hier bitte Ihre aktuelle Sedex-ID und der entsprechende Name einfügen

```
define sedexId = 'yourSedexIdHere';
```

```
define mandantName = 'yourMandantName';
```

Sedex-ID war bevor im [smclient_installDir]/config/config.properties als sedexid konfiguriert. MandantName beschreibt ihre Organization.

Zum Beispiel ist „2-ZH-5“ ist ein Sedex-ID und „Steueraamt Zürich“ wäre der Mandant-Name, die Beschreibung von 2-ZH-5

3.1 Ausführung

Eine Kommando-Fenster an der Stelle öffnen, wo das Skript liegt und dann sqlplus-Programm ausführen (Ändern Sie den Benutzernamen, wenn nötig):

```
sqlplus
```

Dann Benutzer name und Passwort eingeben

Benutzernamen eingeben: smclient

Kennwort eingeben:

SQL Skript ausführen

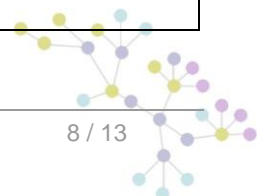
```
SQL> @mandant_update_oracle.sql
```

3.2 SQL Skript

```
-- Syntax is for SQLPLUS query
CREATE TABLE smMandant (
    sedexId varchar2(255 char) NOT NULL PRIMARY KEY,
    name varchar2(255 char) NOT NULL,
    active number(1) DEFAULT 0);

define sedexId = 'yourSedexIdHere';
define mandantName = 'yourMandantName';
INSERT INTO smMandant VALUES ('&sedexId', '&mandantName', 1);

--alter tables (smUser, Message, smAudit)
ALTER TABLE smUser ADD sedexId varchar2(255) DEFAULT '&sedexId';
ALTER TABLE Message ADD sedexId varchar2(255) DEFAULT '&sedexId' NOT NULL;
ALTER TABLE smAudit ADD sedexId varchar2(255) DEFAULT '&sedexId' NOT NULL;
```

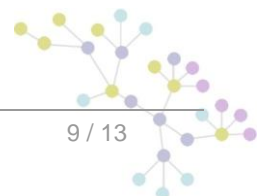




```
UPDATE smUser SET sedexId = '&&sedexId' WHERE sedexId IS NULL OR sedexId LIKE
'';
UPDATE Message SET sedexId = '&&sedexId' WHERE sedexId IS NULL OR sedexId LIKE
'';
UPDATE smAudit SET sedexId = '&&sedexId' WHERE sedexId IS NULL OR sedexId LIKE
'';
--alter table configuration add column sedexId
ALTER TABLE configuration ADD sedexId varchar2(255) DEFAULT '&&sedexId' NOT
NULL;
UPDATE configuration SET sedexId = '&&sedexId' WHERE sedexId IS NULL OR sedexId
LIKE '';
--alter table configuration set primary key.
ALTER TABLE configuration DROP PRIMARY key;
ALTER TABLE configuration ADD PRIMARY key(sedexId, keycolumn);

-- alter table ProcessMapping
ALTER TABLE ProcessMapping DROP PRIMARY key;
ALTER TABLE ProcessMapping ADD sedexId varchar2(255) DEFAULT '&&sedexId' NOT
NULL;
UPDATE ProcessMapping SET sedexId = '&&sedexId' WHERE sedexId IS NULL OR sedexId
LIKE '';
ALTER TABLE ProcessMapping ADD PRIMARY key(messageId, sedexId, processId);
--set foreign keys
ALTER TABLE configuration ADD FOREIGN KEY (sedexId) REFERENCES
smMandant(sedexId);
ALTER TABLE Message ADD FOREIGN KEY (sedexId) REFERENCES smMandant(sedexId);
ALTER TABLE smAudit ADD FOREIGN KEY (sedexId) REFERENCES smMandant(sedexId);

commit;
```





4 MICROSOFT SQL SERVER

Das SQL Skript „mandant_update_mssql“ muss mit dem Microsoft SQL Query Tool (Microsoft SQL Server Management Studio für die 2008 Version) geöffnet werden.

Die Datei „mandant_update_mssql“ öffnen, Benutzername, Kennwort und Datenbank eingeben. Dann die **markierten** Werte editieren.

- 1) Die Variablen „sedexId“ und „mandantName“ editieren: Hier bitte Ihr aktueller Sedex-ID und den entsprechenden Name einfügen.

```
SET @sedexId = 'yourSedexId';
```

```
SET @mandantName = 'default';
```

Sedex-ID war bevor im [smclient_installDir]/config/config.properties als sedexid konfiguriert. MandantName beschreibt ihre Organization.

Zum Beispiel ist „2-ZH-5“ ist ein Sedex-ID und „Steueraamt Zürich“ wäre der Mandant-Name, die Beschreibung von 2-ZH-5

4.1 Ausführung

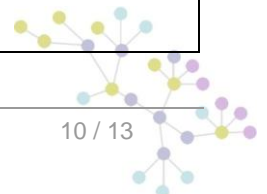
Klicken sie auf den “Execute” Button (dies ist *F5* in der Version 2008)

4.2 SQL Skript

```
begin transaction
CREATE TABLE smMandant (sedexId varchar(255) NOT NULL, NAME varchar(255) DEFAULT
NULL, active tinyint DEFAULT 0, PRIMARY KEY (sedexId));

DECLARE @sedexId varchar(max), @mandantName varchar(max);
DECLARE @executeString nvarchar(max);
SET @sedexId = 'yourSedexId';
SET @mandantName = 'default';
INSERT INTO smMandant VALUES (@sedexId,@mandantName,1);

-- alter smUser
SET @executeString = 'alter table smUser add sedexId varchar(255) DEFAULT ''
SET @executeString = @executeString + cast(@sedexId as nvarchar(max)) + ''';
-- select @executeString;
EXECUTE (@executeString)
SET @executeString = 'UPDATE smUser SET sedexId = '' + cast(@sedexId as
nvarchar(max)) + '' WHERE sedexId IS NULL OR sedexId LIKE ''';
EXECUTE (@executeString)
-- alter Message
SET @executeString = 'alter table Message add sedexId varchar(255) DEFAULT ''
SET @executeString = @executeString + cast(@sedexId as nvarchar(max)) + '' NOT
NULL;';
-- select @executeString;
EXECUTE (@executeString)
```





```
SET @executeString = 'UPDATE Message SET sedexId =''' + @sedexId + ''' WHERE
sedexId IS NULL OR sedexId LIKE '''';';
EXECUTE (@executeString)

-- alter smAudit
SET @executeString = 'alter table smAudit add sedexId varchar(255) DEFAULT '''
SET @executeString = @executeString + cast(@sedexId as nvarchar(max)) + ''' NOT
NULL;';
-- select @executeString;
EXECUTE (@executeString)
SET @executeString = 'UPDATE smAudit SET sedexId =''' + @sedexId + ''' WHERE
sedexId IS NULL OR sedexId LIKE '''';';
EXECUTE (@executeString)

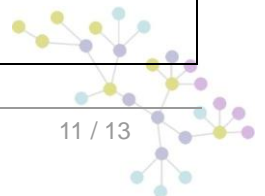
-- alter configuration
SET @executeString = 'alter table configuration add sedexId varchar(255) DEFAULT
'''
SET @executeString = @executeString + cast(@sedexId as nvarchar(max)) + ''' NOT
NULL;';
-- select @executeString;
EXECUTE (@executeString)
SET @executeString = 'UPDATE configuration SET sedexId =''' + @sedexId + '''
WHERE sedexId IS NULL OR sedexId LIKE '''';';
EXECUTE (@executeString)

DECLARE @pkname varchar(max);
SET @pkname = (SELECT [name] FROM sysobjects WHERE [xtype] = 'PK' AND
[parent_obj] = OBJECT_ID(N'[dbo].[' + 'configuration' + ']'));
SET @executeString = 'alter table configuration drop constraint [' + @pkname +
'];';
EXECUTE (@executeString)

SET @executeString = 'ALTER TABLE configuration ADD CONSTRAINT PK_configuration
PRIMARY KEY (sedexId, keycolumn);';
EXECUTE (@executeString)

SET @pkname = (SELECT [name] FROM sysobjects WHERE [xtype] = 'PK' AND
[parent_obj] = OBJECT_ID(N'[dbo].[' + 'ProcessMapping' + ']'));
SET @executeString = 'alter table ProcessMapping drop constraint [' + @pkname +
'];';
EXECUTE (@executeString)
ALTER TABLE ProcessMapping ALTER COLUMN messageId VARCHAR(255) NOT NULL;
SET @executeString = 'alter table ProcessMapping add sedexId varchar(255)
DEFAULT '''
SET @executeString = @executeString + cast(@sedexId as nvarchar(max)) + ''' NOT
NULL;';
-- select @executeString;
EXECUTE (@executeString)
SET @executeString = 'UPDATE ProcessMapping SET sedexId =''' + @sedexId + '''
WHERE sedexId IS NULL OR sedexId LIKE '''';';
EXECUTE (@executeString)

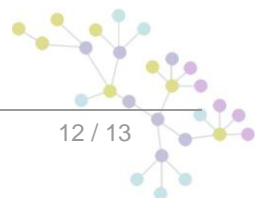
SET @executeString = 'ALTER TABLE ProcessMapping ADD CONSTRAINT PK_ProcMapping
PRIMARY KEY (messageId, sedexId, processId);';
EXECUTE (@executeString)
-- set foreign key
SET @executeString = 'ALTER TABLE configuration ADD CONSTRAINT
FK_configuration_smMandant FOREIGN KEY (sedexId) REFERENCES
smMandant(sedexId);';
EXECUTE (@executeString)
```





```
SET @executeString = 'ALTER TABLE Message ADD CONSTRAINT FK_Message_smMandant
FOREIGN KEY (sedexId) REFERENCES smMandant(sedexId)';
EXECUTE (@executeString)
SET @executeString = 'ALTER TABLE smAudit ADD CONSTRAINT FK_smAudit_smMandant
FOREIGN KEY (sedexId) REFERENCES smMandant(sedexId)';
EXECUTE (@executeString)

commit;
```



Cambridge Technology Partners (Switzerland) S.A.

Nyon

27 Ch. de Précossy
CH-1260 Nyon

Tel: +41 (0)22 306 4646

Fax: +41 (0)22 306 4647

Zurich

Imperial – 4th floor
Leutschenbachstrasse 41
8050 Zürich

Tel: +41 (0)43 456 2300

Fax: +41 (0)43 456 2301

