

EVIDEN



Schweizerische Eidgenossenschaft
Confédération suisse
Confederazione Svizzera
Confederaziun svizra

sM-Client: Installationshandbuch und Aktualisierungsinstruktionen

Schweizerische Steuerkonferenz (SSK) / Bundesamt für Sozialversicherungen (BSV)

Version: R7.x

INHALTSVERZEICHNIS

1 ZWECK DES DOKUMENTS	8
2 INSTALLATION ÜBERSICHT	9
3 INSTALLATION UND KONFIGURATION DER	10
INFRASTRUKTUR-SOFTWARE	10
3.1 Installation und Konfiguration.....	10
3.1.1 Java JDK	10
3.1.2 Sedex-Adapter	10
3.1.3 Swisdec-Adapter	11
3.1.4 Datenbanken	11
4 Neu INSTALLATION UND KONFIGURATION DES SM-CLIENT	12
4.1 Datenbank Installation	12
4.1.1 MySQL.....	12
4.1.2 MS-SQL	15
4.1.3 Oracle	17
4.1.4 Überprüfung und Erstellung der Camunda-Tabellen	19
4.1.5 Überprüfung und Erstellung der sM-Client Meldungenbezogenen Tabellen	19
4.2 Installation von smclient.jar.....	20
4.2.1 Application Profile.....	20
4.2.2 Konfiguration des sM-Clients	21
4.2.3 Technische Konfiguration	22
4.2.4 Monitoring Triage Regel	40
4.2.5 Fachliche Konfiguration.....	41
4.3 Neu-Installation mit vollständigem Backup	45
4.4 Erstellen eines neuen Mandanten	46
4.4.1 Mandantenfähigkeit aktivieren	47
4.5 Konfiguration der eCH0058v4 Domain	48
4.6 Konfiguration der ELM-Domain	50
4.6.1 Hauptkonfiguration	50
4.6.2 Konfiguration der kantonalen Ablage.....	55
4.6.3 ELM Triage Rules	56
4.7 Starten des sM-Client	57
4.8 Installation als Service	57
4.9 sM-Client als Linux Service	58
5 Upgrade auf sM-Client R7	59
5.1 ÜBERSICHT.....	59
5.1.1 Systeme Vorbereitung	59

5.1.2 Camunda Datenbanktabellen.....	59
5.1.3 Änderungen an Spalten in Oracle-Datenbanktabellen (Oracle Version > 12C) .	60
5.1.4 Java Version auf JDK 17 aktualisieren	60
5.1.5 sM-Client Applikation	60
6 ÜBERPRÜFUNG DER INSTALLATION	61
6.1 Überprüfung im Log	61
6.2 Aufruf der Versionsseite	61
6.3 Senden und Empfangen einer Testmeldung	62

DOKUMENTENVERWALTUNG

Historie

Version	Geändert	Beschreibung	Datum	Validator
01	Martin Liebrich	Erstellung und Gliederung	30.03.2009	SM
02	Martin Liebrich	Kapitel 3-6	06.04.2009	SM
03	Sergey Abagyan	Zusätzlicher Input	08.04.2009	ML
04	Martin Liebrich	Review für Release 1 Beta	09.04.2009	SM
05	Sergey Abagyan	Review für Release 1 Beta1	17.04.2009	SM
06	Stefan Malär	Restrukturierung und Ergänzungen Update für Release 2 Beta	22.05.2009	ML
07	Martin Liebrich	Beschreibung Installation mit Wizard	27.05.2009	SM
08	Martin Liebrich, Stefan Malär	Updates Release 1.0 RC1	08.06.2009	SM, ML
09	Martin Liebrich, Stefan Malär, Markus Vitalini	Updates, Installation des Testbed	15.06.2009	SM, ML
10	Martin Liebrich, Stefan Malär	Updates Release 1.0 RC2	06.07.2009	SM, ML

11	Martin Liebrich, Stefan Malär	Updates Release 1.0 RC3	27.07.2009	SM, ML
12	Martin Liebrich, Stefan Malär	Updates Release 1.0 RC4	17.08.2009	SM, ML
13	Martin Liebrich, Stefan Malär	Updates Release 1.0 RC5	24.08.2009	SM, ML
1.0	Stefan Malär	Finale Version	04.09.2009	ML

1.1	Sylvain Berthouzoz	Information über Standard benutzer im Kap 5.5, 7, neue MS SQL JDBC Driver, Updates gemäss Emails empfangen für support-smclient.consulting@atos.net	11.09.2009	
1.2	Sylvain Berthouzoz	Korrektion von LDAP Beispiel	15.12.2009	
1.3	Sylvain Berthouzoz	Updates Release 2.0	19.01.2010	
1.4	Sylvain Berthouzoz	Updates Release 2.1	03.03.2010	
1.5	Christian Wattinger	Updates for Tomcat Installation	11.03.2010	SB
1.6	Sylvain Berthouzoz	Updates for Tomcat on Oracle and MS SQL	23.03.2010	
1.7	Sylvain Berthouzoz	Update page 21, Chapter 4.3	8.11.2010	
1.8	Sylvain Berthouzoz	Updates Release 3.0	05.01.2011	SM

1.9	Sylvain Berthouzo	Update Release 3.0 after feedback from Tests	11.02.2011	SM
2.0	Andreas Doebeli	Update paragraph 4) in chapter 4.6	25.02.2011	SM
2.1	Sylvain Berthouzo	Updates Release 4.0 (chapter 4.7, 4.2.2.6)	04.01.2012	CW
2.2	Andreas Doebeli	Updates Release 4.0 (chapter 4.2.2.4)	16.01.2012	CW
2.3	Christian Wattinger	Sprachliche Anpassungen und Update chapter 4.7 (message-handler outbox)	27.01.2012	SB
2.4	Sylvain Berthouzo	Updates während Testphase (4.2.2 Einschränkungen von Backups)	22.02.2012	CW
2.5	Sylvain Berthouzo	SMCTESTING-37: Beim Versenden von ech0058v4 Meldungen wird die Sedex Quittung nicht abgelegt(4.2.2 messagehandler)	29.02.2012	CW
2.6	Andrea Rapanaro	PDF-Splitting von Sammelmeldungen (4.2.2)	15.06.2012	CW
2.7	Sylvain Berthouzo	ELM Konfiguration (4.8)	27.06.2012	CW
2.8	Christian Wattinger	ELM Konfiguration (4.8) Rules	10.07.2012	SB
2.9	Andrea Rapanaro	ELM Konfiguration fehlenden Parameterbeschreibung	26.07.2012	CW
3.0	Sylvain Berthouzo	Remark on installer 64bits, add XML declaration to example rules xml	17.09.2012	CW
3.1	Sylvain Berthouzo	Monitoring triage rule (4.2.3)	24.09.2012	CW

5.0	Andreas Dejung	Informationen für Release 5.0 hinzugefügt	15/04/2013	CW
5.0.1	Andreas Dejung	Informationen für Release 5.0.1 hinzugefügt	23.07.2013	CW
5.0.7	Fei Guo	Informationen für Release 5.0.7 hinzugefügt (3.1.5.1, 3.1.5.2, 4.1.2, 4.1.4)	20.03.2014	CW
5.0.8	Christian Wattinger	Informationen für Release 5.0.8 hinzugefügt (3.1.5.2); Parameter Konfiguration für Stuck Message Detector in global-config.properties	28.07.2014	SB
	Sylvain Berthouzoz	Infos über Anmeldung an der Monitoring Platform im Kapitel 4.2.3	30.09.2014	
5.0.8_01	Sylvain Berthouzoz	Infos über EAP6 als Windows Dienst (Kap 4.2.4.2.4)	23.01.2015	
5.0.8_02	Sylvain Berthouzoz	Aktualisierung von , protocol.receipt.timeout.days' Standardwert (4.2.2.3)	30.11.2015	
6.0 Draft 3	Sylvain Berthouzoz	Update Release 6.0	21.06.2017	
6.0	Sylvain Berthouzoz	Information über die Limitierung der Länge von Dateienpfad auf Window	25.02.2019	
6.1	Sylvain Berthouzoz	Update Release 6.1	28.08.2020	
6.1.5.2	Agung Bondan	Update Release 6.1.5.2	19.02.2024	
7.0	Agung Bondan	Update Release 7.0	10.09.2024	

1 ZWECK DES DOKUMENTS

Das nachfolgende Dokument beschreibt sämtliche notwendigen Installationsschritte und Konfigurationen, um den sM-Client zu installieren und in Betrieb zu nehmen.

sM-Client Release 7 wurde mit **Spring Boot erstellt**, einem modernen, schlanken Framework, das die Entwicklung und Bereitstellung von Java-Anwendungen vereinfacht. Durch die Nutzung von Spring Boot profitiert die Anwendung von einem optimierten Konfigurationsprozess, reduzierter Komplexität und verbesserter Leistung.

Mit Spring Boot sind herkömmliche Anwendungsserver wie Apache TomEE, JBoss oder Oracle WebLogic Server nicht mehr erforderlich. Spring Boot-Anwendungen werden mit einem eingebetteten Server geliefert, der die Anwendung direkt ausführt. Dieser eigenständige Ansatz vereinfacht die Bereitstellung, indem er die Abhängigkeit von externen Anwendungsservern beseitigt und einen optimierten und effizienten Prozess ermöglicht.

Dieses Dokument wird auch eine Anleitung zum Upgrade des sM-Clients von Release 6.1.5 auf Release 7 sein.

2 INSTALLATION ÜBERSICHT

Die Inbetriebnahme des sM-Clients erfordert verschiedene Installations- und Konfigurationsschritte. Es werden Windows- oder Linux-Betriebssysteme unterstützt. Es sind folgende Teile zu installieren:

- OpenJDK Correto 17 (oder ähnliche) bzw. lizenziertes Oracle JDK 17
- Sedex-Adapter (nicht Teil der Lieferung des sM-Clients)
- Swissdec-Adapter (nicht Teil der Lieferung des sM-Clients) (optional; nur für ELM)
- sM-Client Datenbank : MySQL bzw. Oracle oder MS SQL
- sM-Client JAR-File

Die unterstützten Versionen können dem Technischen Design Dokument entnommen werden.

Anschliessend an die Installation sind folgende Konfigurationen vorzunehmen:

- Konfiguration der Adapter (nicht Teil der Lieferung des sM-Clients)
- Konfiguration der Schnittstellenverzeichnisse (sM-Client)
- Technische Konfiguration (sM-Client)
- Fachliche Konfiguration (nach Systemstart; sM-Client) Eine Default-Konfiguration wird mitgeliefert.

3 INSTALLATION UND KONFIGURATION DER INFRASTRUKTUR-SOFTWARE

Als Grundlage für den Betrieb des sM-Clients muss erst die entsprechende InfrastrukturSoftware installiert werden (bzw. schon vorhanden sein). Prinzipiell sind folgende Teile zu installieren:

- JDK 17
- sedex-Adapter - nicht Teil der Lieferung des sM-Clients
- MySQL (oder MS SQL Server oder Oracle DB)
- Ghostscript - nicht Teil der Lieferung des sM-Clients, optional, (benötigt um mehrere Meldungen gleichzeitig zu drucken, siehe Kapitel ["Konfiguration des sM-Clients"](#) → Einrichten von Druckern")

Die unterstützten Versionen können dem [Technischen Design Dokument](#) entnommen werden.

Anschliessend an der Installation sind folgende Konfigurationen vorzunehmen:

- Konfiguration Adapter - nicht Teil der Lieferung des sM-Clients

3.1 Installation und Konfiguration

3.1.1 Java JDK

Eine entsprechende Java-Umgebung (JDK) mit Version 17 kann aus dem Internet [heruntergeladen](https://docs.aws.amazon.com/corretto/latest/corretto-17-ug/downloads-list.html) (https://docs.aws.amazon.com/corretto/latest/corretto-17-ug/downloads-list.html) werden.

[Installationsanweisung](https://docs.aws.amazon.com/corretto/latest/corretto-17-ug) (https://docs.aws.amazon.com/corretto/latest/corretto-17-ug) von Amazon.

3.1.2 Sedex-Adapter

Der sedex-Adapter kann aus dem Internet [heruntergeladen](https://www.bfs.admin.ch/bfs/de/home/register/personenregister/sedex/downloads.html) (https://www.bfs.admin.ch/bfs/de/home/register/personenregister/sedex/downloads.html) werden.

Die Installation und Konfiguration erfolgt entsprechend der Installationsanweisung des BFS ([Sedex Downloads](#) (https://www.bfs.admin.ch/bfs/de/home/register/personenregister/sedex/downloads.html) → Dokumente → "sedex Client: Installation and User Manual").

3.1.3 Swissdec-Adapter

Der Swissdec-Adapter kann aus dem Internet [heruntergeladen](https://www.swissdec.ch/) (<https://www.swissdec.ch/>) werden.

Die Installation und Konfiguration erfolgt entsprechend der Installationsanweisung.

3.1.4 Datenbanken

3.1.4.1 MySQL

MySQL 8.4.2 LTS kann aus dem Internet [heruntergeladen](https://dev.mysql.com/downloads/mysql/) (<https://dev.mysql.com/downloads/mysql/>) werden.

Die Installation erfolgt entsprechend der [Installationsanweisung](https://dev.mysql.com/doc/refman/8.4/en/installing.html) (<https://dev.mysql.com/doc/refman/8.4/en/installing.html>) von MySQL.

3.1.4.1.1 Performance-Verbesserung

Um die Performance zu verbessern empfehlen wir die folgende Konfiguration für MySQL:

- `innodb_lock_wait_timeout` (konfiguriert in `my.ini` von MySQL Installationsverzeichnis oder in `%PROGRAMDATA%\MySQL\MySQL Server 8.4\my.ini`, siehe MySql Dokumentation für mehr Information über die [Konfigurations-Dateien](https://dev.mysql.com/doc/refman/8.4/en/option-files.html) (<https://dev.mysql.com/doc/refman/8.4/en/option-files.html>)) soll auf 150 erhöht werden.

3.1.4.2 Microsoft SQL Server

Während der Entwicklung wird MSSQL Server Version 16 (Express Edition) verwendet.

Link zum SQL Server Download: <https://www.microsoft.com/en-us/sql-server/sql-server-downloads>

3.1.4.2 Oracle

Während der Entwicklung wird Oracle Database 21c Express Edition verwendet.

4 Neu INSTALLATION UND KONFIGURATION DES SM-CLIENT

Nachdem die Infrastruktur installiert wurde, kann der sM-Client installiert werden. Es sind folgende Teile zu installieren und entsprechende Konfigurationen vorzunehmen:

- sM-Client Datenbank
- sM-Client JAR-File
- Konfiguration der Schnittstellenverzeichnisse
- Technische Konfiguration

4.1 Datenbank Installation

4.1.1 MySQL

Als Beispiel wird hier gezeigt, wie die sM-Client Datenbank unter MySQL installiert werden kann. Unter MySQL muss eine Datenbank mit Name `smclient` angelegt werden:

```
mysql> CREATE DATABASE smclient;
```

Danach muss ein Datenbank-Benutzer angelegt werden, um auf die Datenbank zugreifen zu können.

```
mysql> CREATE USER smclientUser IDENTIFIED BY 'password';  
mysql> GRANT ALL PRIVILEGES ON smclient.* TO smclientUser;
```

4.1.1.1 Konfigurationstabelle

Daraufhin wird die Konfigurationstabelle angelegt und mit den entsprechenden Werten gefüllt. Das untenstehende Script muss dafür angepasst (**Ersetzen von Werten „yourSedexId“**) und ausgeführt werden. Die **confvalue** müssen der jeweiligen sM-Client Installation entsprechen. Das nachstehende Skript wird in Oracle SQL Developer ausgeführt.

```
use smclient;  
  
CREATE TABLE smMandant (sedexId varchar(255) NOT NULL, name varchar(255) DEFAULT  
NULL, active bit DEFAULT false, PRIMARY KEY (sedexId));  
INSERT INTO smclient.smMandant (sedexId,name,active) values  
( 'yourSedexId', 'default', true);  
  
CREATE TABLE domain (  
  id bigint NOT NULL AUTO_INCREMENT,  
  name varchar(255) NOT NULL,  
  PRIMARY KEY (id),  
  UNIQUE KEY name (name));  
INSERT INTO domain (name) values ('smclient');  
INSERT INTO domain (name) values ('ech0058v4');
```

```

CREATE TABLE configuration (keycolumn VARCHAR(255) NOT NULL, confValue VARCHAR(255),
sedexId VARCHAR (255) NOT NULL, domain_id BIGINT NOT NULL DEFAULT '0', PRIMARY
key(sedexId, keycolumn, domain_id));
INSERT INTO configuration (domain_id, keycolumn, confvalue, sedexid ) VALUES
((select id from domain where name like 'smclient'), 'global-config.location',
'C:/smclientbase/config/global-config.properties', 'yourSedexId' );
INSERT INTO configuration
VALUES('config.location','C:/smclientbase/config/config.properties','yourSedexId',
(select id from domain where name like 'smclient'));
INSERT INTO configuration VALUES('message-
handler.location','C:/smclientbase/config/message-handler.properties','yourSedexId',
(select id from domain where name like 'smclient'));
INSERT INTO configuration
VALUES('repository.location','C:/smclientbase/config/repository.properties','yourSed
exId', (select id from domain where name like 'smclient'));
INSERT INTO configuration
VALUES('formservice.location','C:/smclientbase/config/formservice.properties','yourS
edexId', (select id from domain where name like 'smclient'));
INSERT INTO configuration
VALUES('rules.location','C:/smclientbase/config/rules.xml','yourSedexId', (select id
from domain where name like 'smclient'));
INSERT INTO configuration VALUES('timeout.time','30','yourSedexId', (select id from
domain where name like 'smclient'));
INSERT INTO configuration VALUES('keep.deleted.time','10','yourSedexId', (select id
from domain where name like 'smclient'));
INSERT INTO configuration VALUES('message-
handler.location','C:/smclientbase/config/messagehandler-
ech0058v4.properties','yourSedexId', (select id from domain where name like
'ech0058v4'));

```

- **global-config.location:** Definiert wo die Properties Datei für die globale Konfiguration ist. Dies muss nur für einen domain und sedex id definiert werden. Welche benutzt wird ist irrelevant aber sie muss existieren.
- **config.location:** Definiert wo die Properties Datei mit den Konfigurationen, die allgemein für den sM-Client gelten, liegt
- **message-handler.location:** Definiert wo die Properties Datei mit den Konfigurationen für den MessageHandler liegt
- **repository.location:** Definiert wo die Properties Datei mit den Konfigurationen für die Ablage liegt
- **formservice.location:** Definiert wo die Properties Datei mit den Konfigurationen für den Formservice sind
- **rules.location:** Definiert wo die Datei mit den Triageregeln abgelegt ist
- **timeout.time:** Definiert in Minuten wie lange ein Benutzer ohne Aktivität eingeloggt bleibt
- **keep.deleted.time:** Definiert in Tagen wie lange Nachrichten im Papierkorb aufbewahrt werden, bevor sie gelöscht werden

4.1.1.2 Benutzertabellen

Die Benutzertabellen müssen ebenfalls angelegt und mit den entsprechenden Werten gefüllt werden. Das untenstehende Script muss dafür angepasst (**Ersetzen von Werten „yourSedexId“**) und ausgeführt werden.

```
CREATE TABLE smRole (
  id BIGINT NOT NULL AUTO_INCREMENT,
  name VARCHAR(255),
  description VARCHAR(255),
  PRIMARY KEY (id));

INSERT INTO smRole (name, description) values ('Admin', 'Administrator');
INSERT INTO smRole (name, description) values ('ActiveUser', 'Active User (Sender)');
INSERT INTO smRole (name, description) values ('PassiveUser', 'Passive User (Receiver)');
INSERT INTO smRole (name, description) values ('SuperUser', 'Super User');

CREATE TABLE smUser (
  id BIGINT NOT NULL AUTO_INCREMENT,
  username VARCHAR(255),
  password VARCHAR(255),
  firstname VARCHAR(255),
  lastname VARCHAR(255),
  defaultLanguage VARCHAR(255),
  sedexId VARCHAR(255),
  logicSedexId varchar(255) DEFAULT NULL,
  PRIMARY KEY (id));

INSERT INTO smUser (username,password,firstname,lastname, defaultLanguage, sedexId, logicSedexId) values ('Admin', '46/tAEewgFnQ+toQ9ADB5Q==', 'administrator', 'administrator', 'de', 'yourSedexId', null);
INSERT INTO smUser (username,password,firstname,lastname,defaultLanguage, sedexId,logicSedexId) values ('SuperUser', '2atHgjIcy5hGeeRwzY+mhQ==', 'Super', 'User', 'de', null,null);

CREATE TABLE smUser_smRole (
  smUser_id BIGINT NOT NULL,
  role_id BIGINT NOT NULL );

INSERT INTO smUser_smRole (smUser_id,role_id) values((select id from smUser where username='Admin'),(select id from smRole where name='Admin'));
INSERT INTO smUser_smRole (smUser_id,Role_id) values ((select id from smUser where username='SuperUser'),(select id from smRole where name='SuperUser'));
```

Das Script legt einen Standard Benutzer an, welcher anschliessend weitere Benutzer hinzufügen kann. Der Benutzer hat den Benutzernamen: „Admin“ und auch das Passwort „Admin“. Bei der Eingabe ist die Gross- und Kleinschreibung zu beachten. Constraints werden erst nach dem Start von sM-Client appliziert.

4.1.1.3 Transaction-isolation

TRANSACTION_ISOLATION muss auf READ-COMMITTED gesetzt werden:

1. Führen Sie die folgende Abfrage aus:

```
SELECT @@global.transaction_ISOLATION, @@transaction_ISOLATION;
```

2. Falls die TRANSACTION_ISOLATION nicht auf READ-COMMITTED gesetzt ist, dann aktualisieren Sie Ihre MySql Konfigurations-Datei (Standard `my.ini` im Installationsverzeichnis von MySql) mit den folgenden Zeilen:

```
# Set the default transaction isolation level. Levels available are:
# READ-UNCOMMITTED, READ-COMMITTED, REPEATABLE-READ, SERIALIZABLE
[mysqld]
transaction_isolation = "READ-COMMITTED"
```

oder starten Sie den MySql Server mit dem Option `--transaction-isolation=READ-COMMITTED`.

4.1.2 MS-SQL

1. Datenbank "*smclient*" erstellen
2. Login "*smclientUser*" erstellen; zuordnen der Rolle "*db_owner*" für Datenbank "*smclient*".

4.1.2.1 Konfigurationstabelle

Daraufhin wird die Konfigurationstabelle angelegt und mit den entsprechenden Werten gefüllt. Das untenstehende Script muss dafür angepasst (**Ersetzen von Werten** „**yourSedexId**“) und ausgeführt werden.

Die **confvalue** müssen der jeweiligen sM-Client Installation entsprechen. Das nachstehende Skript wird in Oracle SQL Developer ausgeführt.

```
CREATE TABLE smMandant (sedexId varchar(255) NOT NULL, name varchar(255) DEFAULT
NULL, active tinyint DEFAULT 0, PRIMARY KEY (sedexId));
INSERT INTO smMandant VALUES ('yourSedexId', 'mandantName', 1);

CREATE TABLE domain
(
  id BIGINT NOT NULL IDENTITY(1,1),
  name VARCHAR(255) NOT NULL,
  PRIMARY KEY (id),
  UNIQUE (NAME)
);
insert into domain (name) values ('smclient');
insert into domain (name) values ('ech0058v4');

CREATE TABLE configuration (
```

```

keycolumn NVARCHAR(255) NOT NULL,
confValue NVARCHAR(255) NULL,
sedexId NVARCHAR(100) NOT NULL,
domain_id BIGINT NOT NULL,
PRIMARY KEY (keycolumn, sedexId, domain_id)
);

INSERT INTO configuration (domain_id, keycolumn, confvalue, sedexid ) VALUES
((select id from domain where name like 'smclient'), 'global-config.location',
'C:/smclientbase/config/global-config.properties', 'yourSedexId' );
INSERT INTO configuration VALUES('config.location',
'C:/smclientbase/config/config.properties', 'yourSedexId' , (select id from domain
where name like 'smclient'));
INSERT INTO configuration VALUES('message-handler.location',
'C:/smclientbase/config/message-handler.properties', 'yourSedexId' , (select id from
domain where name like 'smclient'));
INSERT INTO configuration VALUES('repository.location',
'C:/smclientbase/config/repository.properties', 'yourSedexId' , (select id from
domain where name like 'smclient'));
INSERT INTO configuration VALUES('formservice.location',
'C:/smclientbase/config/formservice.properties', 'yourSedexId' , (select id from
domain where name like 'smclient'));
INSERT INTO configuration VALUES('rules.location',
'C:/smclientbase/config/rules.xml', 'yourSedexId' , (select id from domain where
name like 'smclient'));
INSERT INTO configuration VALUES('timeout.time','30', 'yourSedexId' , (select id
from domain where name like 'smclient'));
INSERT INTO configuration VALUES('keep.deleted.time','10', 'yourSedexId' ,
(select id from domain where name like 'smclient'));
INSERT INTO configuration VALUES('message-
handler.location','C:/smclientbase/config/message-handler-
ech0058v4.properties','yourSedexId', (select id from domain where name like
'ech0058v4'));

```

4.1.2.2 Benutzertabellen

Die Benutzertabellen müssen ebenfalls angelegt und mit den entsprechenden Werten gefüllt werden. Das untenstehende Script muss dafür angepasst und ausgeführt werden, der Wert "yourSedexId" muss in allen Zeilen mit ihrer Sedex-ID ersetzt werden.

```

CREATE TABLE smRole (
    id BIGINT NOT NULL IDENTITY(0,1),
    name VARCHAR(255),
    description VARCHAR(255),
    PRIMARY KEY (id));

INSERT INTO smRole (name, description) values('Admin', 'Administrator');
INSERT INTO smRole (name, description) values('ActiveUser', 'Active User (Sender)');
INSERT INTO smRole (name, description) values('PassiveUser', 'Passive User
(Receiver)');
INSERT INTO smRole (name, description) values('SuperUser', 'Super User');

CREATE TABLE smUser (
    id BIGINT NOT NULL IDENTITY(0,1),

```



```

username VARCHAR(255) NOT NULL,
password VARCHAR(255),
firstname VARCHAR(255),
lastname VARCHAR(255),
defaultLanguage VARCHAR(255) NOT NULL,
sedexId VARCHAR(255),
logicSedexId varchar(255),
PRIMARY KEY (id)
);

INSERT INTO smUser (username,password,firstname,lastname,defaultLanguage, sedexId,
logicSedexId) values('Admin', '46/tAEewgFnQ+toQ9ADB5Q==', 'administrator',
'administrator', 'de', 'yourSedexId',null);
INSERT INTO smUser (username,password,firstname,lastname,defaultLanguage, sedexId,
logicSedexId) values('SuperUser', '2atHgJicy5hGeeRwzY+mhQ==', 'Super', 'User', 'de',
null,null);

CREATE TABLE smUser_smRole (
smUser_id BIGINT NOT NULL,
Role_id BIGINT NOT NULL
);

INSERT INTO smUser_smRole (smUser_id,Role_id) SELECT (select id from smUser where
username='Admin'),(select id from smRole where name='Admin');
INSERT INTO smUser_smRole (smUser_id,Role_id) SELECT (select id from smUser where
username='SuperUser'),(select id from smRole where name='SuperUser');

```

Das Script legt einen Standard Benutzer an, welcher anschliessend weitere Benutzer hinzufügen kann. Der Benutzer hat den Benutzernamen: „Admin“ und auch das Passwort „Admin“. Bei der Eingabe ist die Gross- und Kleinschreibung zu beachten. Constraints werden erst nach dem Start von sM-Client appliziert.

4.1.3 Oracle

1. Datenbank "smclient" erstellen
2. Login "smclientUser" erstellen; zuordnen der Rolle "DBA" für Datenbank "smclient".

4.1.3.1 Konfigurationstabelle

Daraufhin wird die Konfigurationstabelle angelegt und mit den entsprechenden Werten gefüllt. Das untenstehende Script muss dafür angepasst (**Ersetzen von Werten** „**yourSedexId**“) und ausgeführt werden. Die **confvalue** müssen der jeweiligen sM-Client Installation entsprechen. Das nachstehende Skript wird in Oracle SQL Developer ausgeführt.

```

CREATE TABLE smMandant (
sedexId varchar2(255 char) NOT NULL PRIMARY KEY,
name varchar2(255 char) NOT NULL,
active number(1) DEFAULT 0);

INSERT INTO smMandant VALUES ('yourSedexId', 'mandantName',1);

CREATE TABLE domain (

```

```

id NUMBER(19,0) NOT NULL PRIMARY KEY,
name VARCHAR2(255 CHAR) NOT NULL,
CONSTRAINT "UNIQUE_NAME" UNIQUE ("NAME"));

INSERT INTO domain (id,name) values (1,'smclient');
INSERT INTO domain (id,name) values (2,'ech0058v4');

CREATE TABLE configuration( keycolumn varchar2(255) NOT NULL, confvalue
varchar2(255) NOT NULL, sedexId varchar2(255) NOT NULL, domain_id NUMBER(19,0)
NOT NULL, PRIMARY KEY (sedexId, keycolumn, domain_id));

INSERT INTO configuration (domain_id, keycolumn, confvalue, sedexid ) VALUES
((select id from domain where name like 'smclient'), 'global-config.location',
'C:/smclientbase/config/global-config.properties', 'yourSedexId' );
INSERT INTO configuration VALUES('config.location',
'C:/smclientbase/config/config.properties', 'yourSedexId', (select id from domain
where name like 'smclient'));
INSERT INTO configuration VALUES('messagehandler.location',
'C:/smclientbase/config/messagehandler.properties', 'yourSedexId', (select id from
domain where name like 'smclient'));
INSERT INTO configuration VALUES('repository.location',
'C:/smclientbase/config/repository.properties', 'yourSedexId', (select id from
domain where name like 'smclient'));
INSERT INTO configuration VALUES('formservice.location',
'C:/smclientbase/config/formservice.properties', 'yourSedexId', (select id from
domain where name like 'smclient'));
INSERT INTO configuration VALUES('rules.location',
'C:/smclientbase/config/rules.xml', 'yourSedexId', (select id from domain where name
like 'smclient'));
INSERT INTO configuration VALUES('timeout.time','30', 'yourSedexId', (select id from
domain where name like 'smclient'));
INSERT INTO configuration VALUES('keep.deleted.time','10', 'yourSedexId', (select id
from domain where name like 'smclient'));
INSERT INTO configuration VALUES('message-
handler.location','C:/smclientbase/config/message-handler-
ech0058v4.properties','yourSedexId', (select id from domain where name like
'ech0058v4'));

```

4.1.3.2 Benutzertabellen

Die Benutzertabellen müssen ebenfalls angelegt und mit den entsprechenden Werten gefüllt werden. Das untenstehende Script muss dafür angepasst und ausgeführt werden, der Wert "yourSedexId" muss in allen Zeilen mit ihrer Sedex-ID ersetzt werden.

```

CREATE TABLE smUser (
id NUMBER NUMBER GENERATED by default on null as IDENTITY,
username VARCHAR2(255) NOT NULL,
password VARCHAR2(255),
firstname VARCHAR2(255),
lastname VARCHAR2(255),
defaultLanguage VARCHAR2(255) NOT NULL,
sedexId VARCHAR2(255),
logicSedexId VARCHAR2(255),
PRIMARY KEY (id)
);

```

```

INSERT INTO smUser (id,username,password,firstname,lastname,defaultLanguage,
sedexId, logicSedexId) values (1,'Admin', '46/tAEewgFnQ+toQ9ADB5Q==',
'administrator',
'administrator', 'de', 'yourSedexId', null);
INSERT INTO smUser (id,username,password,firstname,lastname,defaultLanguage,
sedexId, logicSedexId) values (2,'SuperUser', '2atHgJicy5hGeeRwzY+mhQ==', 'Super',
'User', 'de', null, null);

CREATE TABLE smRole ( id NUMBER(37) NOT NULL, name VARCHAR2(255), description
VARCHAR2(255),
PRIMARY KEY (id)
);

INSERT INTO smRole (id,name, description) values (1,'Admin', 'Administrator');
INSERT INTO smRole (id,name, description) values (2,'ActiveUser', 'Active User
(Sender)');
INSERT INTO smRole (id,name, description) values (3,'PassiveUser', 'Passive User
(Receiver)');
INSERT INTO smRole (id, name, description) values (4,'SuperUser', 'Super User');

CREATE TABLE smUser_smRole (
smUser_id NUMBER(37) NOT NULL,
Role_id NUMBER(27) NOT NULL
);

INSERT INTO smUser_smRole (smUser_id,Role_id) VALUES ((select id from smUser where
username='Admin'),(select id from smRole where name='Admin'));
INSERT INTO smUser_smRole (smUser_id,Role_id) VALUES ((select id from smUser where
username='SuperUser'),(select id from smRole where name='SuperUser'));

```

Das Script legt einen Standard Benutzer an, welcher anschliessend weitere Benutzer hinzufügen kann. Der Benutzer hat den Benutzernamen: „Admin“ und auch das Passwort „Admin“. Bei der Eingabe ist die Gross- und Kleinschreibung zu beachten. Sequences und Constraints werden erst nach dem Start von sM-Client appliziert.

4.1.4 Überprüfung und Erstellung der Camunda-Tabellen

Beim Start des **sM-Client** wird automatisch überprüft, ob die erforderlichen Camunda-Tabellen in der Datenbank vorhanden sind. Falls die Tabellen nicht existieren, werden sie während des Startvorgangs automatisch erstellt. Dieser Prozess stellt sicher, dass die Anwendung korrekt funktioniert, ohne dass manuelle Eingriffe erforderlich sind.

4.1.5 Überprüfung und Erstellung der sM-Client Meldungenbezogenen Tabellen

Beim Start des **sM-Client** wird automatisch überprüft, ob die folgenden nachrichtenbezogenen Tabellen in der Datenbank vorhanden sind: **messages**, **smaudit**, **groupedmessageaudit** und **processmapping**. Falls diese Tabellen nicht existieren, werden sie während des Startvorgangs automatisch erstellt. Dieser Mechanismus

gewährleistet, dass die Nachrichtenverarbeitung ordnungsgemäss funktioniert, ohne dass zusätzliche manuelle Konfigurationen erforderlich sind.

4.2 Installation von smclient.jar

Die letzte Version des sM-Client kann [hier](https://sm-client.atos-solutions.ch) (https://sm-client.atos-solutions.ch) heruntergeladen werden. Bitte beachten Sie, dass bei einem Download mittels Internet Explorer, die File-Extension von .jar auf .zip geändert werden kann. In diesem Falle ist nach erfolgtem Download einfach die Endung wieder in .jar zu ändern. **Das heruntergeladene File darf nicht mittels Win-Zip (oder ähnlichen Programmen) ausgepackt werden.**

4.2.1 Application Profile

Vorbereiten application-**smclient**.properties (properties für profile **smclient**)

Die Datei application-**smclient**.properties enthält die verschiedenen Konfigurationen, die erforderlich sind, um die Anwendung in verschiedenen Umgebungen auszuführen, wobei für jede Umgebung eine andere Eigenschaft definiert wird.

Muster application-**smclient**.properties (Bitte passen Sie die Werte entsprechend Ihrer Einrichtung an, z.b. url, username, password)

```
## Logging configuration
logging.file.path=C:/smclientbase/log
logging.file.name=${logging.file.path}/smclient.log
logging.level.root=INFO
logging.level.ch.admin.smclient=INFO
logging.level.org.apache.fop=ERROR
logging.level.ch.admin.smclient2.web.service=DEBUG

## Database config ##
## Enable only ONE of below database configs!!!
## Configuration for MySQL
spring.datasource.url=jdbc:mysql://localhost:3306/smclientpoc
spring.datasource.driver-class-name=com.mysql.cj.jdbc.Driver
spring.datasource.username=smclientUser
spring.datasource.password=password

# OR

## Configuration for MS SQL
spring.datasource.url=jdbc:sqlserver://localhost\\SQLEXPRESS:1433;databaseName=smcli
ent;SelectMethod=cursor;encrypt=true;trustServerCertificate=true
spring.datasource.driver-class-name=com.microsoft.sqlserver.jdbc.SQLServerDriver
spring.datasource.username=smclientUser
spring.datasource.password=password

# OR

## Configuration for Oracle
spring.datasource.url=jdbc:oracle:thin:@localhost:1531/xePDB1
spring.datasource.driver-class-name=oracle.jdbc.OracleDriver
spring.datasource.username=smclientUser
spring.datasource.password=password
```

Die unnötigen Datenbank-Konfigurationen (## Configuration for Oracle) müssen gelöscht werden.

Speichern Sie application-**smclient**.properties in einer Datei, z. B. in C:\smclientbase\config\application-**smclient**.properties.

4.2.2 Konfiguration des sM-Clients

4.2.2.1 Konfiguration der Schnittstellenverzeichnisse

Die Schnittstellen sind im Property-File message-handler.properties definiert. Es sind die Schnittstellenverzeichnisse zu konfigurieren:

- Zum sedex-Adapter
- Zu den Fachapplikationen/externe Triage
- Ablage
- Interne Schnittstellen

Folgende Verzeichnisse sind als Schnittstelle zum **sedex-Adapter** zu konfigurieren. Die Verzeichnisse wurden bei der Installation des Adapters schon angelegt:

- Meldung Eingang
- sedex-Quittung Eingang
- Meldung Ausgang

Details können dem Kapitel [6.1](#) des Technischen Design Dokumentes entnommen werden.

Folgende Verzeichnisse sind als Schnittstelle zu den **Fachapplikationen/externer Triage, bzw. als interne Schnittstellen** zu konfigurieren. Die Verzeichnisse sind gegebenenfalls anzulegen und mit entsprechenden Lese- und Schreibrechten für den sMClient auszustatten:

- Meldung Eingang
 - Unterverzeichnisse für jede Fachapplikation/Triage
 - Unterverzeichnis für Postfach
- Meldung Ausgang
 - Unterverzeichnisse für jede Fachapplikation/Triage
- Gesendete Meldungen
 - Unterverzeichnisse für jede Fachapplikation/Triage
- Gescheiterte Meldungen
 - Unterverzeichnis für Meldungen welche nicht gesendet werden konnten
 - Unterverzeichnis für Meldungen welche nicht übertragen werden konnten

Die Namen der Verzeichnisse sind frei wählbar, müssen aber über ein Property-File definiert werden.

Details können den Kapiteln [6.2](#) und [6.3](#) des Technischen Design Dokumentes entnommen werden.

Folgende Verzeichnisse sind als **Ablage** zu konfigurieren. Die Verzeichnisse sind gegebenenfalls anzulegen und mit entsprechenden Lese- und Schreibrechten für den SMClient auszustatten:

- Ablage
 - Dokumentation
 - Integrity Check
 - Teilnehmerliste
 - XSD_XSLT

Die Namen der Verzeichnisse sind frei wählbar, müssen aber über ein Property-File definiert werden.

Details können den Kapiteln [6.2](#) und [6.3](#) des Technischen Design Dokumentes entnommen werden.

Die Ablage wird zur Verfügung gestellt oder kann von unserer Webseite heruntergeladen [Link](http://sm-client.atos-solutions.ch/) (<http://sm-client.atos-solutions.ch/>). Die Produktive Ablage oder Test Ablage für Test Instanzen [T sedexID]) und muss an den entsprechenden Ort, welcher im repository.properties definiert wird (z.B. C:/smclientbase/interface/repository/), kopiert werden.

4.2.3 Technische Konfiguration

Die Liste der technischen Konfigurationen kann dem Kapitel 5.5.3 des Technischen Design Dokumentes entnommen werden. Die Konfiguration wird per Hand in den entsprechenden Konfigurations-Files vorgenommen.

Wichtig: Stellen sie sicher, dass alle angegebenen Verzeichnisse auch vorhanden sind. Ebenfalls sollten die Zugriffsrechte auf die Verzeichnisse bezüglich der Sicherheitsbedürfnisse des Betriebs vergeben werden. Die Applikation braucht aber auf jeden Fall, Lese- und Schreib-Rechte.

In sämtlichen .properties Dateien dürfen die Werte kein Leerzeichen am Anfang oder am Ende haben.

4.2.3.1 global-config.properties

jms.delay.delivery.millis	Zeitabstand zwischen dem Versand einer JMS-Meldung und ihrem Empfang, damit die Datenbank in dieser Zeit aktualisiert werden kann. (Defaultwert ist 1000, = 1 Sekunde)
directory.min.size	Minimale freie Größe, die für ein Verzeichnis verfügbar sein soll, damit der sM-Client laufen kann (in KiloBytes) (Defaultwert ist 1048576, = 1GB)
script.runtime.timeout	Maximale Zeitspanne (in Stunden), um ein groovy Skript laufen zu lassen, bevor man es als fehlgeschlagen betrachtet. (Defaultwert ist 3, = 3 Std)

Beispielkonfiguration:

```
## Amount of Time (in milisecond) to delay the sending of a JMS Message, which
allows the data to be written in the Database (Default 1000)
## can be increased if you receive a lot of grouped messages or big messages
jms.delay.delivery.millis=1000
## Minimal size that a directory must have for the sM-Client to run (in
KiloBytes)
## default is 1GB (1048576KB)
directory.min.size=1048576
## Maximum amount of time (in HOURS) to let a groovy script running before
considering it as failed. (Default is 3 hours)
script.runtime.timeout=3
```

4.2.3.2 config.properties

sedexId	Ihre Sedex ID (nicht benutzt seit Version 2.0.0)
default.locale	Standard Sprache (z.B. für Export)
filechange.listener.interval	Polling Intervall des sM-Client (in Sekunden)
message.batch.pickup.size	[Optional] Wieviele Meldungen soll der sM-Client in ein Mal verarbeiten. (Default ist 50)
should.message.be.validated	Müssen die Meldungen gemäss XML Schema validiert werden. Siehe auch message-process.xml > Tag message-validation
should.attachments.be.validated	Müssen die Attachments Mime-Types validiert werden
backup.enabled	Defniert ob das Backup aktiv ist
backupDirectory	Verzeichnis, wo die Backups abgelegt werden
time.of.day.for.backup	Tageszeit zu welcher Backups durchgeführt werden sollen

backup.fullIDB.enabled	Definiert ob beim Backup auf ein kompletter Dump der Datenbank gemacht werden soll. Dieser kann für eine manuelle komplette Wiederherstellung verwendet werden. Bemerkung: Mit mehreren Mandanten gibt es beim Backup die unteren Einschränkungen.
backupBatch	Ort wo die Datei MySQLSmClientBackup.bat (für Windows) bzw. MySQLSmClientBackup.sh (für Linux). (Bei einer manuellen Installation muss die Datei vom Installateur dorthin kopiert werden)
inbox.enabled	Definiert ob die Inbox aktiv ist
formservice.enabled	Definiert ob die Formservice aktiv ist
LDAP.enabled	Definiert ob LDAP für die Benutzerverwaltung verwendet werden soll. Falls auf true gesetzt müssen zusätzlich noch die Konfigurationen, welche in Kapitel 4.2.5 beschrieben sind, gemacht werden.
filterfolders	[Optional] Liste von Filtern für die Inbox, mit einem Komma getrennt. Standardwert: unread,important. Zum Beispiel: filterfolders =read,unread,important
print.message.smaller	[Optional] Meldungen, die als XML grösser als diesen Wert (in KB), werden nicht gedruckt, exportiert bzw. ein PDF beim Empfang/Versand dafür generiert. Falls höhere Werte eingestellt werden, muss dafür auch entsprechend mehr Memory konfiguriert werden. Standardwert: 2048
validate.recipients.message.types	Meldungstypen, getrennt mit Komma, für welchen den Empfänger geprüft wird. Alle Meldungen müssen dann den gleichen EmpfängerId haben. Standard werden alle Meldungstypen geprüft (kein Wert im config.properties) Leeres Wert deaktiviert diese Prüfung
should.wait.for.envelope	Meldungen ohne Umschlag werden aus der Sedex Inbox ignoriert, wenn dieses Parameter auf true gesetzt ist (Standard ist inaktiv)

Einschränkungen für Backups

Die interne Backup-Funktion soll nur für kleine Datenmengen (< 10000 Meldungen in der Tabelle Message) verwendet werden und grundsätzlich ist ein sM-Client-externer BackupMechanismus empfohlen. (z.B. durch mysql dump)

Einschränkungen für Backups mit mehreren Mandanten

- Es gibt bis anhin beim Backup die Möglichkeit eines einfachen Backups (DB ohne Prozesse, Filesystem) sowie eines kompletten Backups (kompletter DB-Inhalt einschliesslich Prozess) zu erstellen. Entsprechende Restores können abhängig vom gewählten Backup vom Admin durchgeführt werden. Für den Ein-Mandant-Betrieb ändert sich mit dem neuen Release hierbei nichts.
- Ist mehr als ein Mandant konfiguriert, ergeben sich folgende Änderungen:
 - Einfache Backups können pro Mandant konfiguriert werden und vom entsprechenden Admin des Mandanten auch wieder restored werden.
 - Wird ein komplettes Backup gewünscht, funktioniert dies nur für die ganze sMClient-Installation. Es sollte daher darauf geachtet werden, dass dieses Flag nur für einen Mandanten gesetzt wird (mehrere Flags würden einfach zum mehrfachen Backup der DB führen, hätten also keine schlimmen Auswirkungen) und dieses Backup **nicht** gleichzeitig (=andere Werte für time.of.day.for.backup) und nicht im gleichen Verzeichnis wie die von anderen Mandanten. Nach einem Komplet-Restore der DB (findet ausserhalb des sm-client statt) müssten dann alle Admins der Mandanten noch die Files über die Oberfläche wiederherstellen. Dies setzt voraus, dass das einfache Backup für alle Mandaten gesetzt ist (da sonst keine Files vorhanden sind). Der sM-Client nimmt hier keine Plausibilitätsprüfung vor.

Beispielkonfiguration:

```
#your sedex id
sedexId=6-900014-1
##default application language
default.locale=de
##interval for checking files in seconds
filechange.listener.interval=60
##maximum number of files to process at the same time.
##default is 50
message.batch.pickup.size=10
##should messages.xml files be validated with the schema
should.message.be.validated=true
##should attachment mime types be validated
should.attachments.be.validated=false
##folders displayed under the inbox
##default is unread,important
filterfolders=read,unread,important
##max size for generating pdf from message (in KB)
##bigger messages won't be printed. (default is 2048)
print.message.smaller=2048
##defines if backup is enabled
backup.enabled=false
folder used for backup and restore, folder must exist
backupDirectory=C:/smclientbase/backup
##time of backup (hour of day)
time.of.day.for.backup=12
is full backup (DB dump) enabled
backup.fullDB.enabled=false
path to backup script
backupBatch=C:/smclientbase/scripts/windows/MySQLSmClientBackup.bat
defines if inbox is enabled
inbox.enabled=true
defines if formservice is enabled
formservice.enabled=true
defines if LDAP should be used
LDAP.enabled=false
```

4.2.3.3 message-handler.properties

Parameter zum Prozess	
should.sedex.receipt.be.added	[Optional] Seit R4.0.0 Sedex-Quittung im Zip Datei der Meldung beim Versandsprozess ablegen. Funktioniert nur für Meldungen im Zip Format (Also nicht für eschkg ohne Konvertierung)

export.pdf.splitting	[Optionnel] depuis R4.1 Der Tag export-pdfsplitting bestimmt ob die Exportierung einer Sammelmeldung mehrere PDF Dateien (eine pro Meldung der Sammelmeldung) (true) oder eine einzige PDF Datei generieren soll (false). Konfiguration für alle Meldungstypen gültig. Standard ist false. Es kann auch für bestimmte Meldungstypen konfiguriert werden, Siehe dafür 4.2.2.6 message-process.xml
Schnittstelle zu Fachapplikationen	
base.dir.smclient.interface	Basis-Verzeichnis für sM-Client; die anderen Pfade sind relative zu dieses Verzeichnis
sent.location	Verzeichnis für Meldungen die versendet worden sind
outbox.location	Meldung Ausgang, für Fachmeldungen die verschickt werden sollen
inbox.location	Meldung Eingang, für Meldungen die empfangen worden sind und Postfach angezeigt werden sollen
failed.location	Enthält Meldungen, welche nicht verarbeitet werden können
failed.tosend.location	Enthält Meldungen, welche schon vor dem Versand scheitern
failed.transmit.location	Enthält Meldungen, welche beim Versand scheitern
deleted.location	Enthält Meldungen, welche in den Papierkorb verschoben wurden
exported.location	Enthält die exportierten Meldungen

Schnittstelle zu Sedex Adapter	
base.dir.adapter	Verzeichnispfade der sedex Adapter Schnittstelle
adapter.inbox.location	Meldung Eingang
adapter.receipt.location	Quittung Eingang
adapter.outbox.location	Meldung Ausgang
JMS	
max.num.msg	Maximale Anzahl Submeldungen, die in die PrioQueue gesendet werden. Standard 1
max.num.msg.pdf	Maximale Anzahl Submeldungen, die in die PrioQueue gesendet werden bei der Erzeugung von PDF Dateien. Standard 1
max.size	Maximale Grösse in kB von XML Dateien innerhalb einer Sammelmeldung für die Verarbeitung in priority queue (für die Validierung, Standard 1024)
max.size.pdf	Maximale Grösse in kB von XML Dateien innerhalb einer Sammelmeldung für die Verarbeitung in priority queue (für die PDF Generierung, Standard 1024)

Beispielkonfiguration:

```
####
#message-handler.properties for the default domain
####
##since 4.0.0. Add the Sedex Receipt to the zip of the message,
##default is false
should.sedex.receipt.be.added=false
##base directory for messages all directories below are relative to base
directory
base.dir.smclient.interface=C:/smclientbase/interface/
##location for sent files
sent.location=sent/
##location form messages to be send
outbox.location=outbox/
##location for incoming messages
inbox.location=inbox/post_box
##location for deleted messages
deleted.location=deleted/
##location for exported messages
exported.location=exported/
##location of failed messages
failed.location=failed/
##failed to send it could not be send by the smclient
failed.tosend.location=failed/failed_to_send/
##failed to transmit means that in the send the sedex client could not transmit
the message
failed.transmit.location=failed/failed_to_transmit/
##configuration of sedex adapter
##working directory of sedex adapter all directories below are relative to base
directory
base.dir.adapter=C:/smclientbase/adapter/
##directory for incoming messages
adapter.inbox.location=inbox/
##directory for technical receipts
adapter.receipt.location=receipts/
##directory for outgoing messages
adapter.outbox.location=outbox/
##max number of xml message which still go into the prio queue
max.num.msg=1
##for pdf generation
max.num.msg.pdf=1
##max size of xml messages which still go in to prio queue
max.size=1024
##for pdf generation
max.size.pdf=1024
```

4.2.3.4 formservice.properties

Formular Service	
messageId.prefix.number	4-stellig Zahl, welche der MeldungsId, von im Formservice generierten Meldungen, vorangestellt werden

Beispielkonfiguration:

```
#file contains settings form service of sM-Client
#prefix for messageIds
messageId.prefix.number=4444
```

4.2.3.5 repository.properties

Ablage	
ablage.base.dir	Ablage für Repository des sM-Client. Die anderen Pfade sind relative zu dieses Verzeichnis
xsd_xslt	Sämtliche Meldungsformate (XSD) Zugehörige Umwandlungsanweisungen (XSLT)
documentation	Enthält Dokumentation zum sM-Client (z.B. Benutzerhandbuch); kann Unterordner enthalten (wenn so gesendet)
integrity	Enthält allenfalls Dokumente, welche zur Integritätsprüfung nur vom sM-Client benutzt werden; das Verzeichnis ist deshalb auch nur für den sM-Client zugänglich
participants	Enthält Dokumente, welche die Teilnehmer, Meldungen und Rechte beschreiben
participants.list	Enthält die Liste der aktuellen Teilnehmer
message.list	Enthält die Liste der aktuellen Meldungen; beschreibt, welche Meldungen aktiv sind
message.participants.list	Enthält das Teilnehmergeverzeichnis; beschreibt welche Meldungen Teilnehmer austauschen dürfen

version	[Optional] Enthält den relativen Pfad (ausgehend von ablage.base.dir) zur version.properties Datei. Diese Property ist optional und enthält standardmässig den Wert "participants_list/version.properties". Dieser Wert sollte bei Verwendung der Standardablage nicht verändert werden.
plugins.dir	[Optional] Kann einen absoluten oder einen relativen Pfad beinhalten. Im Falle eines relativen Pfades ist dieser ausgehend vom ablage.base.dir. Der Pfad muss auf ein existierendes Verzeichnis zeigen, welches Plug-ins (JAR Dateien) enthält. Standardmässig wird der Wert "plugins/" verwendet.

Beispielkonfiguration:

```
#ablage is file repository used to store xsd xslts and documentation
ablage.base.dir=C:/smclientbase/interface/repository/
#directory for all xsds and xsls
xsd_xslt=xsd_xslt
#directory for documentation
documentation=documentation
#directory for optional integrity check files
integrity=integrity_check
#directory of participants/message lists
participants=participants_list
#location of participants list
participants.list=participants_list/participants-list.xml
#location of messages list
message.list=participants_list/message-list.xml
#location of message-participants list
message.participants.list=participants_list/message-participants-list.xml
#version properties
version=participants_list/version.properties
#plugins.dir plugins.dir=plugins/
```

4.2.3.6 message-process.xml (Optional)

Einfügen der message-process.xml Datei in der sM-Client Datenbank (pro Domain)

Die Datei " message-process.xml" muss in die Tabelle " Configuration" pro Domain deklariert werden. Benutzen Sie dafür die folgende Abfrage nach der Anpassung von dem Wert **yourSedexId**. Dieser Wert soll einem Mandant entsprechen.

```
INSERT INTO configuration VALUES('message-
process.location','C:/smclientbase/config/message-process.xml', 'yourSedexId', (select id
from domain where name like 'smclient'));
```

Die Datei message-process.xml erlaubt einige Prozessparameter pro Meldungstyp zu definieren, welche dann die Standardwerte überschreiben und auch ein Skript konfiguriert.

Dieses Skript wird für jede empfangene und/oder sendende Meldungen von den entsprechenden Domain ausgeführt.

Beispielkonfiguration:

```

<!--
Document : rules.xml
Created on : October 04, 2010,
Author : Sylvain Berthouzoz
Description:
Test/Example message-process.xml file to define scripts that the sM-Client can
perform before the triage.
-->
<processes>
  <!-- only the most specific one is executed -->
  <!-- script for every message --> {1}
  <script path="./target/test-classes/test.groovy">
    <args>default</args>
    <args>config</args>
  </script>
  <!-- script only for one message type/subtype -->
  <message type="2003" subType="000101"> {2}
  <message-validation>false</message-validation>
  <script path="./src/test/resources/test-exception.groovy"
noPdfGeneration="true" process="receive"> {3 }
    <args>EXCEPTION</args>
    <args>arg2</args>
  </script>
  <script path="./src/test/resources/test-send.groovy"
noPdfGeneration="true" process="send"> {3 }
    <args>default</args>
    <args>config</args>
  </script>
  <export-pdf-splitting>false</export-pdf-splitting>
</message>
</processes>

```

Das Skript kann für alle Meldungen konfiguriert werden {1} oder für einen bestimmten Meldungstyp {2}. Für jede empfangene Meldungen wird nur ein Skript ausgeführt, der sMClient sucht das Skript in der folgenden Reihenfolge:

1. Für Meldungstyp und Meldungssubtyp, wie im {2} hier oben. In diesem Beispiel ist auch die Validierung ausgeschaltet
2. Für Meldungstyp
3. Für alle Meldungen, wie im {1} hier oben
4. Nur für Empfang oder Versand {3}

Tag message

Der Tag "message" enthält die folgenden Attribute:

Name	Beschreibung
Type	Sedex Meldungstyp, zum Beispiel 2003

subType	(Optional) Meldungssubtyp gemäss eCH-0058 standard, zum Beispiel 000101
---------	---

Tag message-validation

Mit diesem Tag kann man die Validierungs-Property (true/false) aus dem config.properties für einen bestimmten Meldungstyp und Meldungssubtyp überschreiben.

Tag script

Der Tag "script" enthält die folgenden Attribute:

Name	Description
path	Absoluter Pfad zum Skript
noPdfGeneration	(Optional, Standardmässig false) True wenn das Skript der generierten PDF der Meldung NICHT braucht. Standardmässig wird der PDF generiert.
process	(Optional, Standardmässig receive) Definiert ob den Skript muss bei Empfang oder Versand oder beide Prozesse durchgeführt werden (receive, send, both).

Der Tag script enthält dann 0 bis n <args> Elementen, die Werte von diesen Elementen werden zum Skript weitergegeben.

Tag args

Der Tag args entspricht ein Parameter des Skripts

Für mehr Information über den Parametern des Skripts, lesen Sie die Dokumentation des Skripts

Tag export-pdf-splitting

Überschreibt den Wert von message-handler.properties

Der Tag export-pdf-splitting bestimmt ob die Exportierung einer Sammelmeldung mehrere PDF Dateien (eine pro Meldung der Sammelmeldung) oder eine einzige PDF Datei generieren soll.

- *True*: die Exportierung der Sammelmeldung generiert eine PDF Datei pro Meldung
- *False*: die Exportierung der Sammelmeldung generiert eine einzige PDF Datei

Groovy Skript

Das Skript muss den Schnittstelle "ch.admin.smclient.service.script.GroovyScriptController" implementieren. Diese Schnittstelle definiert nur eine Methode:

```
String process(ch.admin.smclient.model.Message msg, File xml, File pdf, String[] args)
```

Hier ist eine *HelloWorld.groovy* Implementierung als Beispiel

```
class HelloGroovy implements ch.admin.smclient.service.script.GroovyScriptController
{
String process(ch.admin.smclient.model.Message msg, File xml, File pdf, String[]
args) {
    println "Hello groovy $args for message $msg.messageId"
    "success"
}
}
```

Für mehr Information über Groovy, sehen Sie auf <http://groovy-lang.org> (<http://groovylang.org>). Der sM-Client benutzt die Version 2.4.9 von Groovy

4.2.3.7 Einrichten von Druckern

Für das Drucken von mehreren Meldungen müssen auf dem Server einer oder mehrere Drucker eingerichtet sein. Das Einrichten erfolgt abhängig vom Betriebssystem des Servers.

Die im Betriebssystem vorhandenen und **NUR** die **PostScript kompatiblen** Drucker, werden automatisch vom sM-Client verwendet. Es sollte jedoch überprüft werden, ob der direkte Druck auch funktioniert bezüglich des Druckformats.

Installation von GhostScript

Windows (32 bits)

Der sM-Client wird mit der 32-bit Version von ghostscript nicht mehr getestet.

Windows (64 bits)

Auf Windows kann Ghostscript von hier <http://www.ghostscript.com/download/gsdnld.html> (<http://www.ghostscript.com/download/gsdnld.html>) heruntergeladen werden. Die Version 9.04 (gs904w64.exe) oder neuer sollte installiert werden. Führen Sie das Programm aus. Dann setzen Sie den Installationspfad von Ghostscript bin-Verzeichnis in dem Umgebungsvariable PATH.

1. Öffnen Sie die Computerverwaltung.

2. Klicken Sie in der Konsolenstruktur mit der rechten Maustaste auf **Computerverwaltung (Lokal)**, und klicken Sie dann auf **Eigenschaften**.
 3. Klicken Sie auf der Registerkarte **Erweitert** unter **Umgebungsvariablen** auf **Einstellungen**.
 4. Wählen Sie einen Benutzer in der Liste **Benutzervariablen für** aus.
 5. Klicken Sie auf den Namen (*Path*) der benutzer- oder systemspezifischen Variablen, die geändert oder gelöscht werden soll.
 6. Klicken Sie auf **Bearbeiten**, um den Wert der Variablen mit dem Namen *Path* zu ändern.
 7. Ergänzen Sie im Feld Wert (Variable Value) zunächst ein Semikolon und dann das binVerzeichnis im Installationspfad des Ghostscript. Zum Beispiel *C:\Program Files\gs\gs9.04\bin*.
 8. Als Prüfung, öffnen Sie eine Kommando-Fenster (Start → Ausführen: cmd) und geben sie das folgende Kommando aus: gswin64. Eine Ghostscript Fenster sollte sich öffnen.
- Starten Sie den sM-Client neu.

Linux

Laden sie das ghostscript Paket, zB apt-get install ghostscript herunter oder finden Sie hier die Installationsanleitung: <http://www.ghostscript.com/download/gsdnld.html>
(<http://www.ghostscript.com/download/gsdnld.html>)

4.2.3.8 rules.xml

Das rules.xml-File beinhaltet die Regeln für die Triage. Auf Grund der Komplexität ist es als XML-File angelegt.

Beispielkonfiguration:

```

<trriage xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance"
xsi:noNamespaceSchemaLocation="rules.xsd">
  <rule>
    <criteria>
      <messageType>2003</messageType>
      <messageSubType>000101</messageSubType>
      <messageAction>1</messageAction>
    </criteria>
<destinationFolder>C:/smclientbase/interface/inbox/application</destinationFolder>
  <PDFGeneration boolean="false" pdfSplitting="false"/>
  <useCase>receive</useCase>
</rule>
<rule>
  <criteria>
    <messageType>2015</messageType>
    <messageSubType>000101</messageSubType>
  </criteria>
<destinationFolder>C:/smclientbase/interface/inbox/application_1</destinationFolder>
  <PDFGeneration boolean="true" pdfSplitting="true"/>
  <useCase>receive</useCase>
</rule>
  <!-- Rules with TestMessage. Applied from top to bottom, so the less specific
one should be at the bottom -->
  <rule>
    <criteria>
      <messageAction>2</messageAction>
      <testMessage/>
    </criteria>
    <destinationFolder>/inbox/test-multiple-messageslocation</destinationFolder>
    <useCase>receive</useCase>
  </rule>
  <rule>
    <criteria>
      <testMessage/>
    </criteria>
    <destinationFolder>/inbox/test-messages-location</destinationFolder>
    <useCase>receive</useCase>
  </rule>
  <!-- default rule has no criteria -->
  <rule>
<destinationFolder>C:\smclientbase\interface\inbox\post_box</destinationFolder>
    <PDFGeneration boolean="false" pdfSplitting="false"/>
    <useCase>receive</useCase>
  </rule>
  <rule>
    <destinationFolder>C:\smclientbase\interface\sent</destinationFolder>
    <PDFGeneration boolean="false" pdfSplitting="false"/>
    <useCase>send</useCase>
  </rule>
  <!-- default rule for adapter so all messages are picked up -->
  <rule>
    <useCase>adapter</useCase>
  </rule>
</trriage>

```

Erklärung Verwendung von Regeln

Das Element `<triage>` enthält beliebige Anzahl von `<rule>` Elementen. Die Regeln werden in der Reihenfolge, in der sie auftreten abgearbeitet. Die spezifischen Regeln sollten also zuoberst stehen.

Für die Use Cases (`<useCase>`) "send" und "receive" muss zwingend eine Default-Regel vorhanden sein. Eine Default-Regel hat keine Kriterien (`<criteria>`).

Die Standardkonfiguration beim Empfang ist so, dass die Meldung für die keine spezifischen Regeln bestehen, diese in den Eingang des Postfaches (sollte mit der `inbox.location` im `messagehandler.properties` übereinstimmen) verschoben werden. Für das Senden, werden diese in den Sent Ordner (sollte mit der `sent.location` im `messagehandler.properties` übereinstimmen) abgelegt, welcher im Postfach angezeigt wird. Beim Adapter werden alle Meldung eingelesen.

Es gibt die Use Cases "send", "receive", "failed" und "adapter"; die Use Cases bezeichnen, in welchem Fall die Regeln angewandt werden; "send" wird verwendet beim Versenden von Meldungen, "receive" beim Empfang von Meldungen, "failed" bei fehlgeschlagene Meldungen und "adapter" definiert welche Meldung aus der Inbox eingelesen werden (z.B. wenn mehrere sM-Client einen Adapter verwenden).

`<rule>` enthält

- `<criteria>` - Kriterien der Regel welche ausgewertet werden; nicht vorhanden für Default-Regeln
- `<destinationFolder>` - Verzeichnis, in welches die Meldung im Falle einer anschlagenden Regel gespeichert wird (nur für "send" und "receive")
- `<PDFGeneration>` - boolean (true/false): je nach Wert wird ein PDF in diesem Fall generiert oder nicht wird (nur für "send" und "receive")
- `<PDFGeneration>` - `pdfSplitting` (true/false) (seit 4.1): je nach Wert werden die Meldungen einer Sammelmeldung in einem einzigen PDF (false) oder als einzelne PDFDateien (true) gespeichert.
- `<useCase>` - wie oben beschrieben.

`<criteria>` kann bei "send" und "receive" auf folgende Werte prüfen

- `<messageType>`
- `<messageSubType>`
- `<messageAction>`
- `<testMessage/>`

All diese Werte beziehen sich auf die entsprechenden Werte in der Meldung.

Es sind folgende Wertekombinationen für `<criteria>` möglich

- keine → Default-Regel
- `<messageType>`
- `<messageAction>`

- `<messageType>` und `<messageAction>`
- `<messageType>` und `<messageSubType>`
- `<messageType>`, `<messageSubType>` und `<messageAction>`

`<testMessage/>` soll immer ein leeres Tag sein und prüft ob es sich um eine Testmeldung handelt (TestDeliveryFlag wurde in der Meldung gesetzt). Dieser Tag kann mit die oben genannten Wertekombinationen kombiniert werden. Regeln mit `<testMessage/>` werden in der gleiche Reihenfolge (von oben nach unten), wie im rules.xml stehen, durchgeführt. So soll die wenigste spezifische Regel am Ende des rules.xml stehen.

`<criteria>` kann bei "adapter" auf folgende Werte prüfen

- `<messageType>`
- `<recipientId>`

All diese Werte beziehen sich auf die entsprechenden Werte in der Meldung.

Es sind folgende Wertekombinationen für `<criteria>` möglich

- keine → Default-Regel
- `<messageType>`
- `<recipientId>`
- `<messageType>` und `<recipientId>`

Falls keine Default-Regel verwendet wird muss sichergestellt werden, dass die Update Meldungen so definiert sind, dass sie empfangen werden.

Beispiel Adapter Regeln

Hier einige Beispiel, wie die Adapter Regeln verwendet werden können, falls an einem Adapter mehrere sM-Client oder Applikationen hängen.

Mit dieser Regel, werden alle Meldungen vom Typ 3001, welchen an den Empfänger T6900014-2 gehen eingelesen.

Bemerkung: Default Adapter Regel (adapter Regel ohne `<criteria>`) muss dann gelöscht werden, sonst werden immer alle Meldungen verarbeitet.

```
<rule>
  <criteria>
    <recipientId>T6-900014-2</recipientId>
    <messageType>3001</messageType>
  </criteria>
  <useCase>adapter</useCase>
</rule>
```

Mit dieser Regel, werden alle Meldungen vom Typ 3002 eingelesen:

```
<rule>
  <criteria>
    <messageType>3002</messageType>
  </criteria>
  <useCase>adapter</useCase>
</rule>
```

Mit dieser Regel, werden alle Meldungen welchen an den Empfänger T6-900014-1 gehen eingelesen:

```
<rule>
  <criteria>
    <recipientId>T6-900014-1</recipientId>
  </criteria>
  <useCase>adapter</useCase>
</rule>
```

4.2.4 Monitoring Triage Regel

Monitoring-Meldungen werden jeden Tag von smc-monitoring.atos-solutions.ch verschickt. Damit diese Meldungen die Fachapplikationen nicht stören, muss eine Triage Regel im rules.xml erstellt werden (Siehe auch [4.2.2.8](#) im Installations-Handbuch) Monitoring und Zugriff erfolgt nur nach Anmeldung bei support-smclient.consulting@atos.net (mailto:support-sm-client.consulting@atos.net) – bei dap@bsv.admin.ch (mailto:dap@bsv.admin.ch) für die Teilnehmer von der BSV - mit Angabe von Ihren sedexId und die E-Mail Adresse, welche die Emails von der Monitoring Plattform empfangen wird.

Diese Regel setzt via den Meldungstyp für empfangene Meldungen an:

- Für BSV Meldungstyp 2991
- Für SSK Meldungstyp 3991

4.2.4.1 SSK

Beispiel-Regel zur Monitoring Meldung für SSK im File rules.xml

Bitte *destinationFolder* an Ihre lokale Installation anpassen und das Verzeichnis erstellen.

```
...
  <!-- Siehe Kapitel rules.xml für die komplette Konfiguration -->
<rule>
  <criteria>
    <messageType>3991</messageType>
  </criteria>
  <PDFGeneration boolean="false" pdfSplitting="false"/>

<destinationFolder>C:/smclientbase/interface/inbox/monitoring</destinationFolder>
  <useCase>receive</useCase>
</rule>
...
```


4.2.4.2 BSV

Beispiel-Regel zur Monitoring-Meldung für BSV im File rules.xml:

Bitte *destinationFolder* an Ihre lokale Installation anpassen und das Verzeichnis erstellen.

```
...
    <!-- Siehe Kapitel rules.xml für die komplette Konfiguration -->
<rule>
    <criteria>
        <messageType>2991</messageType>
    </criteria>
    <PDFGeneration boolean="false" pdfSplitting="false"/>

<destinationFolder>C:/smclientbase/interface/inbox/monitoring</destinationFolder>
    <useCase>receive</useCase>
</rule>
...
```

4.2.5 Fachliche Konfiguration

Die Liste der fachlichen Konfigurationen kann dem Kapitel 5.5.1 des [Technischen Design Dokumentes](#) entnommen werden. Die Konfiguration wird über ein Web Interface vorgenommen.

Die Beschreibung davon ist im Benutzerhandbuch [02] zu finden.

4.2.5.1 Logging

Das Logging kann wie folgt konfiguriert werden in der application-**smclient**.properties application profile (wie in Kapitel 4.2.1 "Application Profile" beschrieben) :

```
logging.file.name=C:/smclientbase/log/smclient.log
logging.level.root=INFO

logging.level.ch.admin.smclient=INFO
logging.level.org.apache.fop=ERROR
logging.level.ch.admin.smclient2.web.service=DEBUG
logging.level.ch.admin.smclient.service=DEBUG
```

4.2.5.2 Optionale Konfigurationen

Einige weitere Konfigurationen der sM-Client sind optional und sollten nur gemacht werden, wenn sie benötigt werden.

SSL Zertifikat

Das bei der gewünschten Verwendung eines Zertifikates für den Betrieb, kann im application-**smclient**.properties application profile entsprechend konfiguriert werden, (in Kapitel 4.2.1 "Application Profile" beschrieben).

```
server.ssl.enabled = true
# The format used for the keystore. It could be set to JKS in case it is a JKS file
server.ssl.key-store-type=PKCS12
# The path to the keystore containing the certificate
server.ssl.key-store=smclient7.p12
# The password used to generate the certificate
server.ssl.key-store-password=password
# The alias mapped to the certificate
server.ssl.key-alias=smclient7
```

Ändern von Ports

MySQL Port

Je nachdem wie MySQL installiert wird, kann der Port angepasst werden.

- Bei einer manuellen Installation kann in der Datei my.cnf im Verzeichnis mysql_home/bin der Port geändert werden. Der Port muss unter Client und Server geändert werden, die Linie ist an beiden Stellen dieselbe: port=3306

Ändern des temporären Verzeichnisses

Sollte eine anderes temporäres Verzeichnis als das System Temp Verzeichnis gewünscht sein, kann diese als Command line paramter gesetzt werden. set

```
JAVA_OPTS=%JAVA_OPTS% -Djava.io.tmpdir=C:/smclientbase/temp
```

Wichtige Informationen: Bei Windows-Systemen gibt es eine technische Limitierung bei Dateilängen (Genauer 255 Zeichen). Stellen Sie daher sicher, dass der ausgewählte Pfad (zum Beispiel

C:\smclientbase\adapter\interface\outbox\data_sendprocess5336001870260660156new4662471672881784255_444437efc9947f4d4e57b38f0e b1a4ac45d1T2-VD-5.zip) die Beschränkung auf **255 Zeichen** nicht überschreitet.

4.2.5.3 Konfiguration LDAP

Falls für die Benutzerverwaltung ein LDAP statt der internen Benutzerverwaltung verwendet werden soll, müssen folgende Werte im Konfigurationsfile **config.properties** **zusätzlich** zu den unter 4.2.2.1 beschriebenen konfiguriert werden. Damit der LDAP benutzt wird muss der Wert LDAP.enabled auf true gesetzt werden.

NOTIZ: Der Wert für 'username' muss durch den Login-User-Namen ersetzt werden.

ldap.provider.url	URL unter welcher LDAP oder AD erreichbar sind
-------------------	--

authentication.type	Muss immer auf ‚simple‘ gesetzt sein
bindDN (optional)	<p>Directory Node, welcher vom sM-Client als Basis für die Verbindung gesetzt wird. Der bind DN ist der Benutzer, welcher das Recht besitzt das LDAP innerhalb der definierten Suchbasis zu durchsuchen. In Normalfall, hat der bind DN die Berechtigung, das ganz LDAP zu durchsuchen. Beispiel: Active Directory bind DN cn=administrator,cn=Users,dc=domain,dc=com OpenLDAP bind DN cn=root,dc=domain,dc=com Bemerkung: Falls bindDN/bindCredential nicht gesetzt werden, wird bindDN=baseFiler='username',baseCtxDN benutzt</p>
bindCredential	Passwort damit sM-Client sich beim LDAP anmelden kann
baseCtxDN	<p>Directory Node, welcher vom sM-Client als Basis für Benutzerabfragen verwendet wird. Beispiel: OU=ZUR,OU=Accounts,DC=ctp,DC=com</p>
baseFilter	<p>Filter, nach welchem Benutzer gesucht werden. Bemerkung: ist kein LDAP Filter Für die Suche wird der folgende Filter benutzt: (baseFiler='username',baseCtxDN) Beispiel: cn</p>
languageAttributeID	<p>Zeigt auf das Feld, in welchem die StandardSprache des Benutzers gespeichert ist. Als Sprache kann de, fr oder it gewählt werden.</p>
rolesCtxDN	Directory Node, welcher vom sM-Client als Basis für Rollenabfragen verwendet wird.
roleFilter	<p>Filter, nach welchem Rollen gesucht werden Beispiel: (member=cn='username',OU=ZUR,OU=Accounts,DC=ctp,DC=com)</p>

roleAttributeID	Attribut, in welchem die Rollen gespeichert sind. Beispiel: name
AdminRoleMapper (optional)	Gruppennamen, der mit der Rolle 'Admin' des sM-Client gemappt werden soll Beispiel: smClinetAdministrator Wenn nicht gesetzt, wird standardmässig 'Admin' benutzt
ActiveUserRoleMapper (optional)	Gruppennamen, der mit der Rolle 'ActiveUser' des sM-Client gemappt werden soll Beispiel: smClinetActiveUser Wenn nicht gesetzt, wird standardmässig 'ActiveUser' benutzt
PassiveUserRoleMapper (optional)	Gruppennamen, der mit der Rolle 'PassiveUser' des sM-Client gemappt werden soll Beispiel: smClinetPassiveUser Wenn nicht gesetzt, wird standardmässig 'PassiveUser' benutzt
SuperUserRoleMapper (optional)	Gruppennamen, der mit der Rolle 'SuperUser' des sM-Client gemappt werden soll Beispiel: smClientSuperUser Wenn nicht gesetzt, wird standardmässig 'SuperUser' benutzt

Beispiel einer LDAP Konfiguration:

Annahme der folgenden LDAP-Struktur (LDIF)

```
Benutzer:
  DN: CN=John Frei,OU=ZUR,OU=Accounts,DC=ctp,DC=com
  ObjectClass: OrganizationalPerson
CN: John Frei
  Gruppe:
  DN: CN=SmClientAdmin,OU=Groups,DC=ctp,DC=com
  Member: CN= John Frei,OU=ZUR,OU=Accounts,DC=ctp,DC=com
  Name: smClientAdministrator
```

In diesem Fall sollte die folgende Konfiguration verwendet werden:

```
#LDAP configuration (optional)
ldap.provider.url=ldap://hostname:389/
#authentication type, used to obtain user's roles
#should be "simple" authentication.type=simple
#bind credentials (optional), if not set than #baseFilter=username,baseCtxDN will be
used bindDN=CN=Manager,OU=ZUR,OU=Accounts,DC=ctp,DC=com bindCredential=secret
#from where to start search users
baseCtxDN=OU=ZUR,OU=Accounts,DC=ctp,DC=com
#base filter to search user in ldap baseFilter=cn
#optional, attribute containing user's language, if not
#set "default.locale" will be used languageAttributeID=msExchUserCulture #from where
to start search roles
rolesCtxDN=OU=ZUR,OU=Groups,DC=ctp,DC=com
#matching filter for searching roles,
#'username' will be substitute with login name
roleFilter=(member=cn='username',OU=ZUR,OU=Accounts,DC=ctp,DC=com)
roleAttributeID=name
AdminRoleMapper=smClientAdministrator
ActiveUserRoleMapper=smClientActiveUser
PassiveUserRoleMapper=smClientPassiveUser
```

Konfiguration LDAP mit SSL

Für eine LDAPS-Anbindung müssen sie den Zertifikat vom LDAP-server auf Java installieren. Sie können das folgende machen:

1. Der CA Zertifikat von dem LDAP server in .crt, .pem, or .der Format erhalten
2. Keytool Import Tool, das mit der JAVA Installation kommt, benützen:

```
keytool -importcert -trustcacerts -file <certificate.crt> -keystore
<JAVA_HOME>/lib/security/cacerts -alias <alias-name>
```
3. Wiederholen Sie Schritt 1) & 2) für jede LDAP Server, der vom sM-Client benutzt wird
4. In der Datei config.properties die URL des LDAP-Servers ändern. Zum Beispiel:

```
ldap.provider.url=ldaps://ldapServerUrl:636/
```

4.3 Neu-Installation mit vollständigem Backup

Falls der sM-Client für Full-Backup konfiguriert ist, kann der sM-Client nach schwerwiegenden Problemen (Systemabsturz, defekte Disk) neu installiert und wieder in den Ausgangszustand gebracht werden. Dafür sind folgende Schritte notwendig:

1. Falls die Prozesse des sM-Client (Applikationsserver und/oder Datenbank) noch laufen, sollten diese vorgängig beendet werden.
2. Danach sollte der sM-Client wie beschrieben komplett neu installiert werden. Eventuell noch vorhandene Verzeichnisse sollten vorgängig komplett deinstalliert/gelöscht werden.
3. Konfiguration des sM-Client wie beschrieben. Datenbank muss zuerst gestartet werden.

4. Ausführen des Scripts MySQLSmClientBackup.bat auf Windows, bzw. MySQLSmClientBackup.sh auf Linux/Unix. Dadurch wird die Datenbank wiederhergestellt
 - MySQLSmClientBackup.bat restore
 - MySQLSmClientBackup.sh restore
1. Starten des sM-Clients
2. Einloggen als Admin
3. Auf der Seite „Configuration->Backup“ können per Knopf „Restore Files“ die Meldungsverzeichnisse (einschliesslich temp) wiederhergestellt werden.

4.4 Erstellen eines neuen Mandanten

Zuerst muss man sich als SuperUser anmelden. Falls es noch kein SuperUser gibt, muss man diesen erstellen. Siehe dafür Kapitel 4.5.1 Mandantenfähigkeit aktivieren.

Der SuperUser kann die Mandanten verwalten:

- Mandant erstellen
 - Mandanten aktivieren
 - Mandanten deaktivieren
1. Auf der Benutzeroberfläche als SuperUser angemeldet (Standardbenutzer ist SuperUser mit Kennwort SuperUser), kann ein neuer Mandant mit einem eigenen Administrator erstellt werden. Dafür muss das Formular „Mandant hinzufügen“ ausgefüllt werden.
 2. Mit dem folgenden Skript kann die Konfiguration für einen Mandant erstellt werden. Hier soll das Skript mit dem richtigen Pfad und der SedexId aus Schritt 1 angepasst werden.

```
INSERT INTO configuration
VALUES('config.location','C:/smclientbase/config_newMandant/config.properties', '
newMandantSedexId', (select id from domain where name like 'smclient'));
INSERT INTO configuration
VALUES('message-handler.location','C:/smclientbase/config_newMandant/message-handler
.properties',' newMandantSedexId', (select id from domain where name like
'smclient'));
INSERT INTO configuration
VALUES('repository.location','C:/smclientbase/config_newMandant/repository.properties', '
newMandantSedexId', (select id from domain where name like 'smclient'));
INSERT INTO configuration
VALUES('formservice.location','C:/smclientbase/config_newMandant/formservice.properties'
,' newMandantSedexId', (select id from domain where name like 'smclient'));
INSERT INTO configuration
VALUES('rules.location','C:/smclientbase/config_newMandant/rules.xml',' newMandantSedexId',
(select id from domain where name like 'smclient'));
INSERT INTO configuration VALUES('timeout.time','30',' newMandantSedexId', (select id from
domain where name like 'smclient'));
INSERT INTO configuration VALUES('keep.deleted.time','10',' newMandantSedexId', (select id
from domain where name like 'smclient'));
```

1. Der neue Mandant muss, wie im Kapitel 4.2 Konfiguration des sM-Clients beschrieben, konfiguriert werden.
2. Auf den Benutzeroberfläche kann man jetzt als SuperUser den neuen Mandanten aktivieren. Dafür muss der neue Mandant selektiert und dann durch den Link „Mandanten ein/ausschalten“ aktiviert werden.
3. Der Benutzer, der in Schritt 1 erstellt wurde, kann sich nun einloggen und neue Benutzer erstellen.

4.4.1 Mandantenfähigkeit aktivieren

Um die Mandantenfähigkeit zu benutzen, muss ein SuperAdmin verfügbar sein. Wenn die LDAP Authentifizierung benutzt wird, muss entsprechend eine LDAP Rolle als SuperUser im config.properties definiert sein. (siehe Kapitel 4.2.4.3 Konfiguration LDAP). Ansonsten muss dieser Benutzer in der Datenbank erstellt werden:

1. Erstellen der SuperUser Rolle:

```
INSERT INTO smRole (name, description) values('SuperUser', 'Super User');
```

2. Erstellen des SuperUser Benutzers:

```
insert into smUser (username,password,firstname,lastname,defaultLanguage, sedexId)
values('SuperUser', '2atHgjIcy5hGeeRwzY+mhQ==', 'Super', 'User', 'de', null);
INSERT INTO smUser_smRole (smUser_id,Role_id) VALUES (
(select id from smUser where username='SuperUser'),
(select id from smRole where name='SuperUser'));
```

3. Jetzt kann man sich als SuperUser einloggen (Kennwort: SuperUser, ohne Mandant).

4.5 Konfiguration der eCH0058v4 Domain

Diese Konfiguration ist nur notwendig, um eCH0058v4 Meldungen zu verschicken oder zu empfangen.

Ab R6.0 ist der eCH0058v4 Domain Teil der Standard-Installation und muss also nicht mehr separat installiert werden

1. Erstellen der eCH0058v4 Domain in der Datenbank:

```
insert into domain (name) values ('ech0058v4');
```

2. Einbinden der Konfigurationsdatei (Passen Sie zuerst Ihre sedexId und den Pfad an):

```
INSERT INTO configuration VALUES('message-  
handler.location','C:/smclientbase/config/message-handler-ech0058v4.properties','yo  
urSedexId', (select id from domain where name like 'ech0058v4'));
```

3. Jede neue Domain kann auch eigene Verzeichnisse brauchen, deshalb gibt es ein message-handler.properties pro Domain, in welchen im Minimum die Verzeichnisse definiert sein müssen, damit der sM-Client den richtigen Prozess startet. Hier wird nur die *outbox.location* von den Standard message-handler.properties unterschieden.

```
####  
#Message handler ech0058v4 configuration  
####  
#base directory for messages all directories below are relative to #base directory  
base.dir.smclient.interface=C:  
smclientbase/interface/  
#location for sent files  
sent.location=sent/  
#location form messages to be send  
outbox.location=outbox/  
#location for incoming messages  
inbox.location=inbox/  
#location for deleted messages  
deleted.location=deleted/  
#location for exported messages  
exported.location=exported/  
#location of failed messages  
failed.location=failed/  
#failed to send it could not be send by the smclient  
failed.tosend.location=failed/failed_to_send/  
#failed to transmit means that in the send the sedex client could not transmit the  
message  
failed.transmit.location=failed/failed_to_transmit/  
#configuration of sedex adapter  
#working directory of sedex adapter all directories below are relative to base  
directory  
base.dir.adapter=C:\\smclientbase/adapter/  
#directory for incoming messages
```



```
adapter.inbox.location=inbox/  
#directory for technical receipts  
adapter.receipt.location=receipts/  
#directory for outgoing messages  
adapter.outbox.location=outbox/
```

4.6 Konfiguration der ELM-Domain

4.6.1 Hauptkonfiguration

Wichtig

Diese Domain benötigt einen Swisdec Adapter (sdA). Der sdA muss vorher installiert werden. Siehe dafür Benutzerhandbuch swisdec Adapter
Diese Konfiguration ist nur notwendig, um ELM- (Swisdec Quellensteuer und Lohnausweis) Meldungen zu verschicken oder zu empfangen.

1. Erstellen der ELM Domain in der Datenbank:

```
insert into domain (name) values ('elm');
```

2. Einbinden der Konfigurationsdatei (Passen Sie zuerst Ihre sedexId und den Pfad an):

```
INSERT INTO configuration  
VALUES ('message-handler.location','C:/smclientbase/config/message-handler-elm.prope  
rties','yourSedexId', (select id from domain where name like 'elm'));
```

3. Jede neue Domain kann auch eigene Verzeichnisse benutzen, deshalb gibt es ein message-handler.properties File pro Domain, in welchem im Minimum die benötigten Verzeichnisse definiert sein müssen, damit der sM-Client den richtigen Prozess startet kann. Hier wird nur die *outbox.location* von den Standard message-handler.properties unterschieden. Die Verzeichnisse, die hier definiert werden, müssen mit den Verzeichnisse übereinstimmen, die für den sdA konfiguriert sind. **Wichtig:** Diese Verzeichnisse müssen vor dem Start des sM-Client vorhanden sein. Siehe die Tabelle unten, die die Abhängigkeiten zwischen sM-Client und sdA beschreibt.

```
####
#Message handler ELM configuration
####
elm.timeout.days=60
#base directory for messages all directories below are relative to base directory
base.dir.smclient.interface=C:/smclientbase/interface/
#location for sent files
sent.location=elm/sent/
#location form messages to be send
outbox.location=elm/outbox/
#location for incoming messages
inbox.location=inbox/post_box
#location for deleted messages
deleted.location=elm/deleted/
#location for exported messages
exported.location=elm/exported/
#location of failed messages
failed.location=failed/
#failed to send it could not be the send the sedex client could not transmit the
message
failed.transmit.location=failed/failed_to_transmit/
#configuration of swissdec adapter
#working directory of swissdec adapter all directories below are relative to base
directory
base.dir.adapter=C:/smclientbase/adapterSwissdec
#base.dir.adapter=//VBOXSVR/Temp
#directory for incoming messages
adapter.inbox.location=inbox/
#directory for technical receipts
#[base.dir.adapter] + [adapter.receipt.location] should point to the sedex adapter
receipts
adapter.receipt.location=../adapter/receipts/
#directory for outgoing messages
adapter.outbox.location=outbox/
##specific for swissdecadapter and ELM domain
adapter.failed.receive=failed/
adapter.sent.location=sent/
adapter.failed.transmit.location=failed_to_transmit
#configuration of sedex adapter
#base directory is taken from 'base.dir.adapter' of the default (smclient)
#domain (message-handler.properties)
#used for forwarding to commune.
#directory for outgoing messages
sedex.adapter.outbox.location=outbox/
```

Die folgende Tabelle beschreibt die Abhängigkeiten zwischen der Konfiguration von sMClient und sdA.

sM-ClientMessage-handler-elm.properties	PathID	Swissdec AdapterswissdecAdapter.properties
Im sM-Client sind diese Verzeichnisse relativ zu base.dir.adapter definiert.		Der sdA braucht jeweils den absoluten Pfad in dieser Konfigurationsdatei.

adapter.inbox.location	PATH_ELM_IN	released.filesystem.parent.xml
adapter.outbox.location	PATH_ELM_OUT	result.filesystem.parent.xml
adapter.failed.receive	PATH_ELM_FAILED_TO_RX	failed.filesystem.parent.xml
adapter.sent.location	PATH_ELM_SENT	sent.filesystem.parent.xml
adapter.failed.transmit.location	PATH_ELM_FAILED_TO_TX	undeliverable.filesystem.parent.xml

Die folgende Tabelle beschreibt die Properties, die für die Verzeichniskonfiguration gebraucht werden.

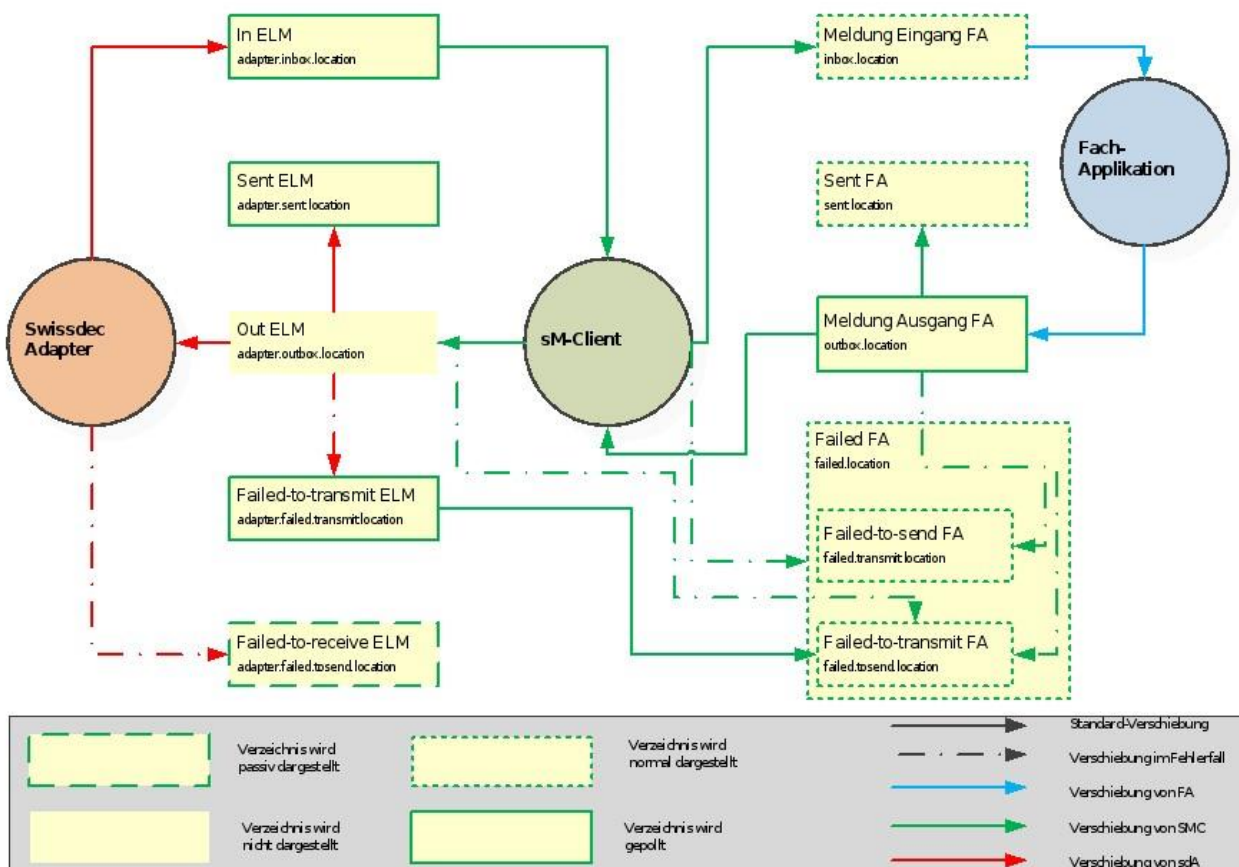
Ordner-Name im Tech Design	Config File	Property Key	Beschreibung
<i>In ELM</i>	message-handler-elm.properties	adapter.inbox.location	Dieser Ordner wird die erfolgreich vom sdA empfangenen Meldungen zur Übergabe and den SMClient enthalten.
<i>Out ELM</i>	message-handler-elm.properties	adapter.outbox.location	In diesen Ordner wird der sM-Client die Antwortmeldungen zur Abholung durch den sdA ablegen.
<i>Sent ELM</i>	message-handler-elm.properties	adapter.sent.location	Hierhin werden die erfolgreich verschickten Meldungen vom sdA verschoben.
<i>Failed-to-transmit ELM</i>	message-handler-elm.properties	adapter.failed.transmit	In diesen Ordner werden Meldungen durch den sdA verschoben, falls der sdA eine Antwort vom SMC nicht übermitteln kann.

<i>Failed-to-receive ELM</i>	message-handler-elm.properties	adapter.failed.receive	Bei fachlichen Fehlern, die von sdA entdeckt werden, wird der sdA die entsprechenden Meldungen ergänzt mit dem Error Code in diesen Ordner ablegen.
<i>Meldung Eingang FA</i>	message-handler-elm.properties	inbox.location	Meldung Eingang, für Meldungen die empfangen worden sind und Postfach angezeigt werden sollen
<i>Meldung Ausgang FA</i>	message-handler-elm.properties	outbox.location	Meldung Ausgang, für Fachmeldungen die verschickt werden sollen.
<i>Sent FA</i>	message-handler-elm.properties	sent.location	Verzeichnis für Meldungen die versendet worden sind.
<i>Failed FA</i>	message-handler-elm.properties	failed.location	Enthält Meldungen, welche nicht verarbeitet werden können.
<i>Failed-to-send FA</i>	message-handler-elm.properties	failed.transmit.location	Enthält Meldungen, welche beim Versand scheitern.
<i>Failed-to-transmit FA</i>	message-handler-elm.properties	failed.tosend.location	Enthält Meldungen, welche schon vor dem Versand scheitern.

Die folgende Tabelle beschreibt die Properties, die für verschiedene Konfigurationen gebraucht werden.

Property Key	Typ	Config File	Beschreibung
elm.timeout.days	Integer	message-handler-elm.properties	Dieses Parameter beschreibt die Zeit die der sM-Client wartet, bevor er eine geschickte Meldung vom <i>Out ELM</i> zu dem <i>Failed-to-send FA</i> Ordner verschiebt wenn sie vom sDA nicht abgenommen wird.
forward.message.zipped	Boolean	message-handler-elm.properties	Wenn "true", werden die Meldungen bei Gemeindefortleitung gezippt.

Dieses Diagramm beschreibt die Schnittstellen in der ELM-Domäne (ohne Gemeindefortleitung):



4.6.2 Konfiguration der kantonalen Ablage

Die kantonale Ablage wird nur für die Weiterleitung von Quellensteuer-Meldungen (qst) vom Kanton zu einer Gemeinde via Sedex benötigt.

1. Zuerst muss dafür die Datei repository.properties ergänzt werden:

```
#FOR ELM domain only
    ablage.kantonal.base.dir=C:/smclientbase/interface/cantonal_repository
cantonal.message.list=participants_list/message-list.xml

cantonal.message.participants.list=participants_list/message-participants-list.xml
    cantonal.participants.list=participants_list/participants-list.xml
cantonal.xsd_xslt=xsd_xslt
```

2. Im File message-list.xml der kantonale Ablage wird der Meldungstyp von qst Meldungen für Sedex festgelegt. Die Werte, die von # umgeben sind, müssen für die jeweilige Umgebung angepasst werden. Der Meldungstyp muss zuerst bei Sedex registriert werden. (Siehe Sedex-Handbuch und Sedex-Betrieb)

```
<messages xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance"
xsi:noNamespaceSchemaLocation="message-list.xsd">
    <message type="#9999#" subtype="#000909#"
informService="false" sendWithoutProtocolReceipt="true" domain="elm">
<redirectFrom type="elm" subtype="qst" />
    <name>#elm-9999-000909#</name>
    <version number="#http://www.elm.ch/xmlns/elm-9999-000909/1#">
        <valid_from>01.01.2010</valid_from>
        <valid_to>01.09.2999</valid_to>
    </version>
</message>
</messages>
```

3. Im File participants-list.xml der kantonale Ablage können die Gemeinden konfiguriert werden. Wenn eine Gemeinde die Weiterleitung nicht braucht, löschen Sie das Attribut municipalityId dieser Gemeinde. Werte, die von # umgeben sind, müssen angepasst werden.

```
<participants xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance"
xsi:noNamespaceSchemaLocation="participants-list.xsd">
    <participant sedexId="#99-ZH-99#" municipalityId="#1234#">
        <shortName>Musterdorf</shortName>
        <longName>Gemeinde Musterdorf</longName>
    </participant>
    <participant sedexId="#99-ZH-98#" municipalityId="#4321#">
        <shortName>Musterdorf 2</shortName>
        <longName>Gemeinde Musterdorf 2</longName>
    </participant>
</participants>
```

4. Im File message-participants-list.xml wird festgelegt, welche Teilnehmer die Meldungen empfangen dürfen. Hier soll also Ihre sedexId und auch die Gemeinden von Schritt 3 participants-list.xml stehen. Meldungstyp und Meldungsstyp müssen also mit den Werten im File message-list.xml aus Schritt 2 übereinstimmen.

```
<groups xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance"
xsi:noNamespaceSchemaLocation="message-participants-list.xsd">
  <group name="elm-1">
    <participant sedexId="99-ZH-98" />
    <participant sedexId="99-ZH-99" />
    <participant sedexId="T6-900014-1" />
    <participant sedexId="T6-900014-2" />
    <message type="9999" subtype="000909" />
  </group>
</groups>
```

4.6.3 ELM Triage Rules

Die ELM-Triage-Konfiguration folgt der Standard-Konfiguration (siehe [Konfiguration des sMClients](#) → rule.xml), hat aber ein paar Besonderheiten:

- Automatische PDF und CSV-Generierung sind bei ELM ab Version 5.0 möglich
- Element <messageAction> ist bei <criteria> nicht erlaubt
- Meldungstyp und SubTyp sind keine sedex Meldungstypen, sondern fix definiert in der Ablage inmessage-list.xml.
 - Meldungstyp: elm
 - Subtyp: qst oder la
 Beispielregel für ELM:

```
<trriage xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance"
xsi:noNamespaceSchemaLocation="rules.xsd">
  <!-- ...
  Other rules goes here
  ...-->
  <!-- Siehe Kapitel 4.2.2.5 für die komplette Konfiguration -->
<rule>
  <criteria>
    <messageType>elm</messageType>
    <messageSubType>qst</messageSubType>
  </criteria>
  <destinationFolder>C:/smclientbase/interface/inbox/elm</destinationFolder>
  <useCase>receive</useCase>
</rule>
  <!-- Other rules goes here
  ...-->
</trriage>
```


4.7 Starten des sM-Client

Bevor Sie den sM-Client starten, stellen Sie sicher, dass die smclient-Datenbank gestartet und verfügbar ist.

```
c:\java -jar -Dspring.profiles.active=smclient smclient-7.0.1.jar
```

- -Dspring.profiles.active=**smclient**. **smclient** ist das Profil, das als application profiles verwendet wird, wie in Kapitel 4.2.1 beschrieben.

spring.profiles.active kann auch in der Umgebungsvariablen SPRING_PROFILES_ACTIVE gesetzt werden.

Spring Boot sucht und lädt automatisch application.properties- und application.yaml-Dateien aus den folgenden Speicherorten, wenn Ihre Anwendung gestartet wird:

1. Aus dem Klassenpfad
 - a) Der Klassenpfad root
 - b) Das Paket classpath /config
 - c) Aus dem aktuellen Verzeichnis
2. Das aktuelle Verzeichnis
 - a) Das Unterverzeichnis config/ im aktuellen Verzeichnis
 - b) Unmittelbare untergeordnete Verzeichnisse des Unterverzeichnisses config/

Wenn application-smclient.properties sich nicht am oben genannten Speicherort befindet, muss Folgendes hinzugefügt werden: -Dspring.config.location=<application-smclient.properties location>/. z.B.

```
c:\java -jar -Dspring.config.location=c:\other\location\ -Dspring.profiles.active=smclient smclient-7.0.1.jar
```

4.8 Installation als Service

In einer Windows-Umgebung kann der sM-Client als Windows-Dienst ausgeführt werden, indem ein Windows-Dienst-Wrapper eines Drittanbieters verwendet wird.

Ref: <https://github.com/winsw/winsw>

Beispiel für eine Konfigurationsdatei (smclient.xml)

```
<service>
  <id>smclient</id>
  <name>sm-Client 7</name>
  <description>This service runs sm-Client7</description>
  <env name="SMCLIENTHOME" value="c:\smclientbase"/>
  <workingdirectory>c:\CENTRAL\jdk-17.0.9\bin\java</ workingdirectory>
  <executable>c:\CENTRAL\jdk-17.0.9\bin\java</executable>
```

```
<arguments>
  -jar -Dspring.config.location=%SMCLIENTHOME%\
  -Dspring.profiles.active=%SMCLIENTHOME%\smclient2-7.jar</arguments>
<log mode="roll"/>
</service>
```

Die obige sampe-Konfigurationsdatei wird davon ausgegangen, dass sich sowohl smclient2-7.jar als auch application-smclient.properties-Dateien im Ordner c:\smclientbase befinden.

Hier ausführlichen Schritten:

1. Den Service von sM-Client 6 deinstallieren (bei Tomee gibt es eine remove.as.admin Datei)
2. Den JAVA_HOME Pfad auf Java 17 setzen.
3. Winsw.exe herunterladen. In das Verzeichnis von der sM-Client JAR kopieren und als «smclient.exe» umbenennen.
4. Die Konfigurations-XML (smclient.xml) erstellen. Siehe Beispiel oben. Bei mir schaut diese so aus:
5. In der Konsole ausführen «smclient install»

4.9 sM-Client als Linux Service

Unter Linux lesen Sie bitte unter <https://www.baeldung.com/spring-boot-app-as-a-service#on-linux> nach.

5 Upgrade auf sM-Client R7

Voraussetzung für das Update ist, dass der SEDEX-Adapter UND der Swissdex-Adapter gestoppt werden müssen. Es wird empfohlen, zu warten, bis alle Transaktionen ordnungsgemäss verarbeitet und abgeschlossen sind, bevor sM-Client gestoppt wird.

Es wird davon ausgegangen, dass das Upgrade von einer Person mit Zugang zur Datenbank und dem Datenbank Client (z.B. mysql console für MySQL, sqlcmd für MSSQL, sqlplus für Oracle, oder Oracle SQL Developer) durchgeführt wird.

5.1 ÜBERSICHT

Für die Aktualisierung des sM-Clients werden die unten gelisteten Installationsschritte benötigt. Der sM-Client Release 7 kann sowohl mit Windows- wie auch LinuxBetriebssystemen verwendet werden. Es sind folgende Komponenten zu aktualisieren:

- Altes System stoppen
- Camunda Datenbanktabellen
- Oracle Datenbanktabellen (nur bei Verwendung der Oracle-Datenbank)
- Java Version auf Amazon Corretto 17 JDK aktualisieren
- sM-Client Applikation

Wichtig!

Bitte machen Sie ein Backup der sM-Client Datenbank.

5.1.1 Systeme Vorbereitung

Damit die Datenbank konsistent bleibt, soll der sM-Client wie folgt für eine Aktualisierung vorbereitet werden.

1. Sedex-Adapter stoppen.
2. SwissDec-Adapter stoppen.
3. Alle sM-Client Benutzer müssen sich ausloggen.
4. 10 Minuten warten, bis alle internen Aktivitäten im sM-Client fertig sind.
5. sM-Client stoppen.

5.1.2 Camunda Datenbanktabellen

Für sM-Client Release 7 muss auch die Camunda Version aktualisiert werden, d.h. die Camunda Datenbanktabellen müssen von Version 7.7 auf 7.21.0 aktualisiert oder gepatcht werden.

So aktualisieren auf die im sM-Client R7 verwendete Camunda-Version 7.21

- Laden die camunda-sql-scripts-7.21.0.zip von hier [herunter](#) (<https://artifacts.camunda.com/artifactory/camunda-bpm/org/camunda/bpm/distro/camunda-sql-scripts/7.21.0/camunda-sql-scripts-7.21.0.zip>).
- Entpacken die Zip-Datei in ein Verzeichnis, z. B. **c:\temp\camunda-sql-scripts-7.21.0**
- Vom database client, führen die Datei in dem oben genannten Verzeichnis:
 - o drop**<dbType>**_engine_7.21.0.sql
 - o drop**<dbType>**_identity_7.21.0.sql
 - o create**<dbType>**_engine_7.21.0.sql
 - o create**<dbType>**_identity_7.21.0.sql

Hier **<dbType>** soll mit dem richtigen DB-Typ ersetzt werden. Die möglichen Werten sind: mysql, oracle, mssql.

5.1.3 Änderungen an Spalten in Oracle-Datenbanktabellen (Oracle Version > 12C)

Für die Migration der Oracle-Datenbank zur Verwendung mit sM-Client Version 7 wird hier ein Skript bereitgestellt (<https://sm-client.atos-solutions.ch/sM-Client7/scripts/update-smclient-Oracle-6.1.5.x-7.0.5.sql>).

Dieses Skript soll auf dem OracleDB ausgeführt werden.

5.1.4 Java Version auf JDK 17 aktualisieren

Für den sM-Client Release 7 ist ausserdem ein Wechsel auf eine Open-Source Java Distribution der Version 17 vorgesehen. Bitte vergewissern Sie sich, dass Sie entweder eine **lizenzierte** Oracle JDK 17 64 Bit Version verwenden, oder eine entsprechende Open Source Variante installiert haben.

Wir empfehlen die Verwendung von:

[Amazon Corretto 17 JDK](https://docs.aws.amazon.com/corretto/latest/corretto-17-ug/downloads-list.html) (<https://docs.aws.amazon.com/corretto/latest/corretto-17-ug/downloads-list.html>) in einer 64 Bit Version (jdk17 64 bits)

5.1.5 sM-Client Applikation

Der sM-Client 7 soll jetzt konfiguriert werden. Die Version 7 verlangt kein Applikationsserver (TomEE, JBoss, ...) mehr. Die Server können entfernt werden.

Die Konfiguration/Installation ist im Kapitel 4.2 beschrieben.

6 ÜBERPRÜFUNG DER INSTALLATION

Anschliessend an die Installation des sM-Client können folgende 3 Tests durchgeführt werden.

- [Überprüfung im Log](#)
- [Aufruf der Versionsseite](#)
- [Senden und Empfangen einer Testmeldung](#)

6.1 Überprüfung im Log

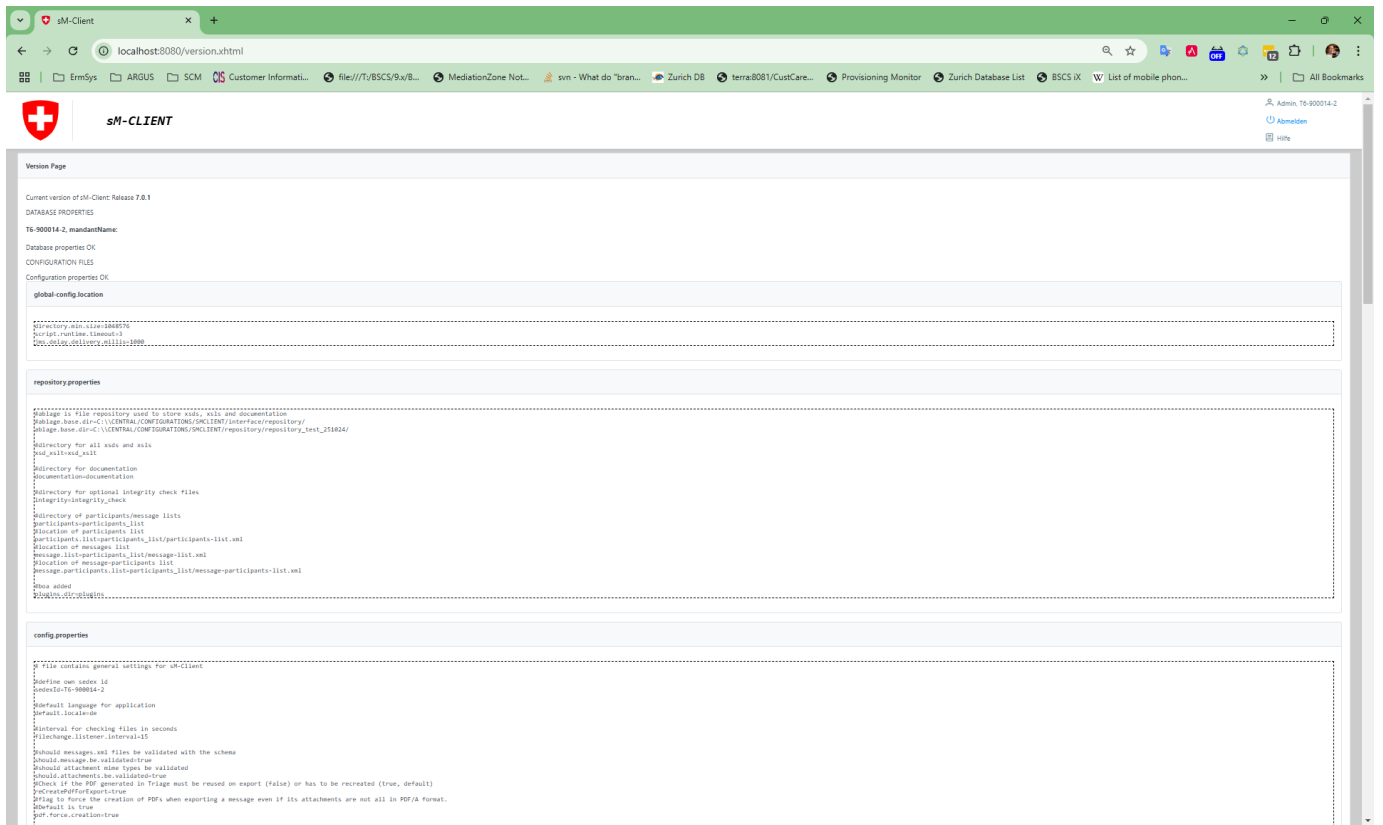
Um sicherzustellen, dass die Installation von **sM-Client** erfolgreich war, kann im `smclient.log` Datei überprüft werden, ob irgendwelche Fehler aufgetreten sind. Wenn keine übereinstimmenden Zeilen gefunden werden, war die Installation erfolgreich. Sollten Fehler- oder schwerwiegende Meldungen vorhanden sein, überprüfen Sie diese, um mögliche Probleme zu identifizieren und zu beheben.

6.2 Aufruf der Versionsseite

Nach dem Einloggen in den **sM-Client** können Sie die Konfiguration über die **Version**-Seite aufrufen. Klicken Sie dazu im Hauptmenü auf den Menüpunkt **Version**.

Die Seite sollte die aktuelle Version des installierten sM-Clients anzeigen, analog der folgenden Abbildung.

Auf der Seite werden sämtliche Konfigurationen aufgelistet.



Beim ersten Login können Sie mit dem Standard Benutzer einloggen. (Benutzernamen: **"Admin"** und auch das Passwort **"Admin"**). Bei der Eingabe ist die Gross und Kleinschreibung zu beachten.

6.3 Senden und Empfangen einer Testmeldung

Um zu überprüfen ob das Senden und Empfangen korrekt funktionieren kann eine Testmeldung in eines der gültigen Ausgangsverzeichnisse kopiert werden. Die Meldung sollte dabei an den Absender adressiert werden.

Die Meldung sollte daraufhin durch den sM-Client versandt und auch von ihm empfangen werden (Selbstversand). Es kann hiermit mittels Audit-Trail und ggf. Logs die ganze Meldungsverarbeitung verfolgt werden. Darüber hinaus kann überprüft werden, ob sich nach Abschluss von Versand und Empfang alle Meldungen und Quittungen in den korrekten Verzeichnissen befinden.

EVIDEN AG

<https://eviden.com/ch-en/>

Lokaler Hauptsitz Zürich

Freilagerstrasse 28

8047 Zürich

Bern

Belpstrasse 26

3007 Bern

Eysins

Route de Crassier 7

1262 Eysins

Basel

Aeschenvorstadt 67- 75

4051 Basel

Vevey

Rue des Bosquets 31

1800 Vevey

Hauptsitz der Gruppe

River Ouest, 80 quai Voltaire

95877 Bezons cedex – France