

# The SSCd Handbook

Peter H. Grasch



# The SSCd Handbook

# Contents

|          |  |           |
|----------|--|-----------|
| <b>1</b> | <b>Introduction</b>                    | <b>6</b>  |
| <b>2</b> | <b>Using SSCd</b>                      | <b>7</b>  |
| 2.1      | Base folder . . . . .                  | 7         |
| 2.2      | Configuration . . . . .                | 7         |
| 2.3      | Database . . . . .                     | 8         |
| 2.4      | Locked mode . . . . .                  | 8         |
| 2.5      | Extracting collected samples . . . . . | 9         |
| <b>3</b> | <b>Questions and Answers</b>           | <b>10</b> |
| <b>4</b> | <b>Credits and License</b>             | <b>11</b> |
| <b>A</b> | <b>Installation</b>                    | <b>12</b> |

# List of Tables

|     |                           |   |
|-----|---------------------------|---|
| 2.1 | SSC base folder . . . . . | 7 |
|-----|---------------------------|---|

### **Abstract**

SSCd is the server component of the SSC sample acquisition tool.

# Chapter 1

## Introduction

SSCd is the server component of the SSC sample acquisition tool.

It manages speaker data (users, institutions) as well as sample meta data using a database and stores the samples.

It receives input from the SSC client(s) which connect to the server using TCP/IP.

For more information on the general architecture of the Simon suite please see the [Simon manual](#).  
For information about the SSC client, please refer to the [SSC manual](#).

# Chapter 2

## Using SSCd

SSCd is a command line application which does not have any user interface. There are no special launch parameters.

### 2.1 Base folder

The base folder of SSCd contains the configuration file `sscd.conf` and an error log in case something goes wrong. This folder also contains the `samples` subfolder where all the samples are stored.

The location of the SSCd folder depends on your operating system:

| Microsoft Windows  | GNU/Linux                    |
|--|------------------------------|
| Installation folder of <code>sscd.exe</code><br>(usually: <code>C:\Program Files\simon<br/>0.3\bin\sscd.exe</code> ) | <code>/usr/share/sscd</code> |

Table 2.1: SSC base folder

### 2.2 Configuration

There is no graphical configuration of SSCd, but there is a configuration file (`sscd.conf`) stored in the [SSCd folder](#).

The default configuration file is heavily commented and should be self-explanatory.

Before running SSCd, you will want to at least change the `DatabaseUser` and `DatabasePassword` entry of the configuration file. See [the database section](#) for more information.

```
; This is an example config file and displays the built-in defaults
; SSCd will look for this file in:
; Linux:
;   /usr/share/sscd/sscd.conf
; Windows:
;   <sscd installation path>\sscd.conf

[General]
```

## The SSCd Handbook

```
; Change this to use a different database; Because SSCd uses db-specific
; commands in places, only QMYSQL is supported at the moment.
; Support for other DBMS can be added extremely easily, though so please
; feel free to request support through kde-accessibility@kde.org
DatabaseType=QMYSQL

; The host of the DBMS
DatabaseHost=127.0.0.1

; The port of the DBMS; 3306 is the default port of MySQL
DatabasePort=3306

; The database to use; Make sure that you run the supplied create script
; before you use SSCd
DatabaseName=ssc

; The username to use when connecting to the DBMS
DatabaseUser=sscuser

; Database password. The default one will obviously not work in most cases
DatabasePassword=CHANGE ME NOW

; Database options. Refer to Qts documentation of QSqlDatabase for details
DatabaseOptions=MYSQL_OPT_RECONNECT=1

; The port the server will listen to; Default: 4440
Port=4440

; Bind the server to a specific client IP; If this is true, the server
; will ignore requests from all but the BoundHost (see below)
Bind=false

; IP of the bound host (if Bind is active)
BindHost=127.0.0.1
```

### 2.3 Database

SSCd stores the speaker and sample data (but not the samples themselves) in a database. At the moment, only MySQL databases are fully supported. Adding support for a new database is trivial, though. Contact the [Simon team](#) if you want to help.

To set up the required tables SSCd ships with an appropriate create script `mysql_create_script.sql` installed in the [base folder](#) of SSCd.

Database errors can be found in the `error.log` which also resides in the base folder.

### 2.4 Locked mode

SSCd optionally takes the command line argument `-l` (or `--locked`) to turn on “locked” mode (off by default).

In locked mode, clients can still connect and upload samples as well as create new microphones and sound cards but do not have any access (read or write) to any personal- or institutional data besides the users name and id. This can be helpful to limit the amount of private patient information shown to recording teams.



While locked mode is active, there is also no search of users so make sure your recording team is provided with a list of user ids beforehand.

## 2.5 Extracting collected samples

To build models using the samples collected with SSCd you first have to extract them from the database.

### WARNING

Because SSCd is designed for large scale sample acquisition this is not end user friendly. The documentation below is mainly provided for technically skilled professionals.  
Most of the scripts below require the GNU tools (usually available by default on GNU/Linux).

You can use the following query (minor adjustments will be necessary depending on what samples exactly you need):

```
use ssc;

select s.Path, s.Prompt
  from Sample s inner join User u
    on s.UserId = u.UserId inner join UserInInstitution uii
    on u.UserId = uii.UserId inner join SampleType st
    on s.TypeId = st.SampleTypeId inner join Microphone m
    on m.MicrophoneId = s.MicrophoneId
 WHERE st.ExactlyRepeated=1 and uii.InstitutionId = 3
    and (m.MicrophoneId = 1);
```

This query will list all samples from institution 3 that were recorded with microphone 1.

You can then for example use this script to create a prompts file:

```
#!/bin/bash
sed 'ld' $1 > temp_out
sed -e 's/\\\\\\\\\\\\\\\\/g' -e 's/.*Samples\\\\\\\\/g' -e 's/\\.wav\\t/ //' temp_out <->
  > $1
rm temp_out
```

This prompts file can then be [imported in Simon](#).

To build the appropriate dictionary to compile the model you might also want to list all the sentences contained in the prompts file. You can do this with this script:

```
#!/bin/bash
cat $1 | sed -e 's/[0-9\\/]*/ //' | sort | uniq
```

## Chapter 3

# Questions and Answers

In an effort to keep this section always up-to-date it is available at our [online wiki](#).

## Chapter 4

# Credits and License

SSCd

Program copyright 2008-2010 Peter Gräsch [peter.grasch@bedahr.org](mailto:peter.grasch@bedahr.org)

Documentation Copyright (c) 2009-2010 Peter Gräsch [peter.grasch@bedahr.org](mailto:peter.grasch@bedahr.org)

This documentation is licensed under the terms of the [GNU Free Documentation License](#).

This program is licensed under the terms of the [GNU General Public License](#).

# Appendix A

## Installation

Please see our [wiki](#) for install instructions.