



# Release Notes

*Version:* 3.3.0.0  
*Date:* 23.08.2020  
*Status:* Approved  
*Authors:* P. Roberts & N. Sheppard (Intersect), A. Hueni (UZH)  
*File:* SPECCHIO\_ReleaseNotes.docx  
*Pages:* 15

*Classification:*  
*Distribution:* SPECCHIO Users



# 1 Introduction

SPECCHIO is a spectral database combined with user-friendly interface software designed to store spectral data acquired by spectroradiometers and associated metadata.

SPECCHIO was first developed at the Remote Sensing Labs at the Geography Department, University of Zurich to support long term usability and data sharing between researchers. It was then further enhanced through a project run by the University of Wollongong in 2012/2013. This project was supported by the Australian National Data Service (ANDS). ANDS is supported by the Australian Government through the National Collaborative Research Infrastructure Strategy Program and the Education Investment Fund (EIF) Super Science Initiative.

SPECCHIO is an Italian word meaning “mirror” or “looking glass”. It can also be used to refer to a table of data or a scoreboard.

## 1.1 Document scope

SPECCHIO uses a Client-Server architecture. This User Guide details operation of the Client component of SPECCHIO only.

## 1.2 Intended audience

This document assumes that readers are familiar with...

- remote sensing and the disciplines and processes related to it.
- the general operation of their own computer.
- the general concept of a client-server architecture.

## 1.3 SPECCHIO ownership and access

SPECCHIO was originally built by the Remote Sensing Laboratories at the University of Zurich, and extended by Intersect for the School of Earth and Environmental Sciences at the University of Wollongong.

## 1.4 Copyright and licensing

SPECCHIO is licensed under the Creative Commons Attribution-ShareAlike 3.0 Unported Licence. Therefore its source is readily available for inspection and development. It can be found in LICENCE.html and at <http://creativecommons.org/licenses/by-sa/3.0/>.

## 1.5 For Further Information

Please refer to the following documents for more information about SPECCHIO. Unless otherwise stated, they can be found in the SPECCHIO Installation kit.

**SPECCHIO\_ReleaseNotes.pdf** can be found in each Installation Kit and provides installation instructions for the SPECCHIO Client.

**SPECCHIO\_Tutorial.pdf** provides instruction in the operation of key areas of the SPECCHIO Client.

**SPECCHIO\_ServerInstallation.pdf** provides system administrators with information to assist in managing and maintaining a SPECCHIO Server System.

**SPECCHIO\_VM.pdf** provides information about the SPECCHIO Virtual Machine, basing on a VirtualBox CentOS virtual machine. The SPECCHIO VM is an easy solution to running a complete SPECCHIO server without the complex installation procedure of a complete native server installation.

**SPECCHIO Web Site** ([www.specchio.ch](http://www.specchio.ch)) General information about SPECCHIO. Some of this information may be related to other non-UOW versions of SPECCHIO.

**SPECCHIO UoW** (<https://specchio.uow.edu.au>) Installation kits for University of Wollongong version of the SPECCHIO Client and documentation for that version.

**SPECCHIO DC10 GitHub** (<https://github.com/IntersectAustralia/dc10>) Source code for the University of Wollongong version of SPECCHIO. This is a legacy code; the DC10 related updates to SPECCHIO are all included in the general SPECCHIO version on GitHub.

**SPECCHIO GitHub** (<https://github.com/ahueni/SPECCHIO>) Source code for the community version of SPECCHIO, currently developed under the lead of RSL (University of Zurich).

## 2 Installation and Configuration

### 2.1 Before you install

SPECCHIO requires that Java Runtime Environment (JRE) version 1.8<sup>1</sup> is already installed on your computer before installing SPECCHIO itself. To check the Java version on your system open a command window under Windows, or a terminal for Macintosh or UNIX systems, and type:

```
java -version
```

The output will be similar to:

```
openjdk version "1.8.0_265"  
OpenJDK Runtime Environment (AdoptOpenJDK)(build 1.8.0_265-b01)  
OpenJDK 64-Bit Server VM (AdoptOpenJDK)(build 25.265-b01, mixed mode)
```

If you do not have Java installed, or the version number is less than 1.8, you should install an appropriate version of the Java Runtime Environment (JRE) from the internet. We suggest that you use the OpenJDK versions.

To download the installation package, go to: <https://adoptopenjdk.net/> and select OpenJDK 8 (LTS).

The builds of Java 8 after JAN 2020 include a new security feature that is not compatible with the current SPECCHIO certificate. The SPECCHIO client is updated to include a Java command line option to avoid this problem. In case you experience some issues nonetheless, then please use this specific version of the OpenJDK and download the package for your operating system:

<https://github.com/AdoptOpenJDK/openjdk8-binaries/releases/tag/jdk8u212-b04>

For more information on the certificate problem, see the SPECCHIO FAQ:  
<https://specchio.ch/faq/>

### 2.2 Installing the SPECCHIO Application Bundle

#### 2.2.1 Installing SPECCHIO for the First Time

The SPECCHIO application plus the libraries it uses are supplied as an application bundle in JAR file format. The installation bundle is usable on Windows, Mac and Unix systems. For Mac OS X, a special installation bundle is additionally provided, packaging the SPECCHIO application as Mac OS application.

1. Open `specchio-client-installer.jar` by double-clicking on it, or executing `java -jar specchio-client-installer.jar` from the command line.  
For Mac OS X open the `specchio-client-MacOSX-installer.jar` respectively.
2. Follow the prompts to install the software into a directory of your choice.

**Note** SPECCHIO does not currently support multi-user installations. SPECCHIO should usually be installed within the personal folder of each user wanting to use the software. This is the folder called `C:\Users\UserName` in Windows; `/Users/username` in Mac OS X; or `/home/username` in Unix and Linux.

<sup>1</sup> Later versions of Java may work, but have not been officially tested.

### 2.2.2 Upgrading an Existing Installation

You can upgrade an existing installation by following the same procedure as for a new installation. Just install the new version of SPECCHIO into the same folder as your existing installation. You may wish to create a backup of your existing installation first.

Older versions of SPECCHIO store the database account information in a file named `db_config.txt`. New versions store this information in system preferences. Installations that still contain a `db_config.txt` file will continue to use it. Upgrading to a new version of SPECCHIO will not change the content of an existing `db_config.txt` file.

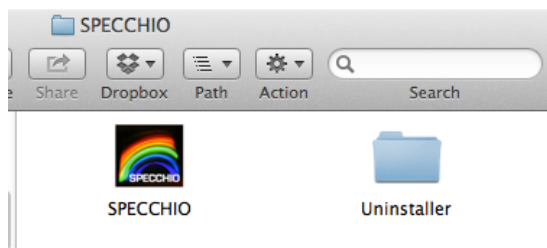
## 2.3 Launching the SPECCHIO Application

### 2.3.1 Windows

The installer creates a folder on the "Start" menu. This folder contains options for launching SPECCHIO on both 32-bit versions and 64-bit versions of Windows. Select the option that matches your version of Windows.

### 2.3.2 Mac OS X

Open Finder, and navigate to the folder into which you installed SPECCHIO. Double click the SPECCHIO application icon.



In case you installed the standard bundle (i.e. non Mac OS specific bundle), navigate to the `macosx` folder and double-click on `specchio`.

### 2.3.3 Unix and Linux

Navigate to the folder into which you installed SPECCHIO. If using a graphical interface that supports it, you may be able to start the application by double-clicking on `specchio-client.jar`. Otherwise, execute `java -jar specchio-client.jar` from the command line.

If your Java build version is  $\geq 241$ , then you may experience certificate problems.

To circumvent this, use the following additional line to start SPECCHIO from the command line, or put it into a shell script or desktop shortcut:

```
export _JAVA_OPTIONS="-Djdk.security.allowNonCaAnchor=true"
java -jar specchio-client.jar
```

## 2.4 Creating a User Account

- Start the SPECCHIO client application.
- Select *Database*, then *Create a new user account* from the main menu.
- Complete the server details as follows, depending on the server you want to connect to:

Worldwide SPECCHIO Online System, hosted by University of Zurich

Web Application Server: `sc22.geo.uzh.ch`

Port: `443`

Application Path: `/specchio_service`

Data Source: Choose between productive and test servers:

`jdbc/specchio_prod`

`jdbc/specchio_test`

- Press **Connect**.
- Complete the *title*, *first name*, *last name*, *institute*, *e-mail* and *WWW* fields as desired.

The *first name*, *last name*, *institute* and *e-mail* are required.

Add new institutes as necessary using the **Add new institutes...** button.

SPECCHIO will not send any e-mail to the e-mail address entered and the address is not visible to other users.

- Press **Create**.

You should see a message saying that an account has been created and its details added to your configuration file.

The account username and password are automatically stored in the system preferences of your computer. You essentially do not need to know or record them yourself; it may however be prudent to take a note in case your computer crashes and the preferences are lost.

Older versions of SPECCHIO stored them in a configuration file (`db_config.txt`). If you have upgraded from previous versions then SPECCHIO will continue to read account data from the configuration file. Newer versions can be enforced to write data into the configuration file during account creation; this is achieved by selecting *Database*, then *Edit db\_config file* from the main menu. Note: the menu item is greyed out by default; to enable it set the according switch in the SPECCHIO Preferences.

## 2.5 Using SPECCHIO

- Select *Database*, then *Connect to database* from the main menu.
- If you have only created one account, this account should be selected in the *Known connections* selector, and the dialogue should be pre-filled with the account details.  
If you have created more than one account, you can choose between them using

the *Known connections* selector.

- Press **Connect**.

You are now ready to use SPECCHIO as described in the User Guide.

## Appendix A: Change History

### V3.0.1

- DC10-301 Added a researcher description field for Research Data Australia
- DC10-302 Corrected a JAXBException when viewing an instrument picture
- DC10-303 Preserved db\_config.txt when upgrading an existing installation
- DC10-304 Corrected account name generation for names containing punctuation
- DC10-306 Cleared the progress message when loading campaign data fails

### V3.1.0

- Added support for automatic instrument and sensor inserts
- Added support for multiple data sources (multiple database on same server)
- Added packaging as Mac OS X application

### V3.1.3

- Database updates (CORINE landcover support, instrument calibration support)
- Updates to increase speed during data selection and loading
- Enhanced API
- Interactive database config file editing

#### Bugfixes:

- Increased stability of system over time and multiple users
- Timestamp issue when operating the system in different timezones
- Increased file reader stability for UniSpec and ASD

### V3.2.0 Alpha – Gamma Versions

- Timeline plot updated to show datapoints as squares.
- New Data Link function to create links between spectra to model Target-Reference links, Reference-Target links, Target-Target links and Provenance links.
- Query Builder has new method to set spectrum ids from an external process; useful for using query builder functionality with an existing set of spectrum ids.
- Speed improvement for data removal and file loading.
- Update to support latest file versions produced by SVC HR1024i.
- File loader for file produced by Ocean Optics new Ocean View software.
- File loader for Microtops text file and new attributes to support the metadata of the Microtops.
- File loader for Bruker FTIR dpt files.
- Ocean Optics Spectra Suite files: update to deal with corrupt files where the spectral data block was not written properly.
- Better reporting support for loaded/parsed files.
- Returning empty metaparameters when a metaparameter does not exist for a supplied spectrum. Applies to getMetaparameterValues and getMetaparameters



methods of the SpecchioClient. Non-existing parameters have their eav\_id set to 0.

- Improved matching of sensors and instruments based on full wavelengths information during data loading.
- Radiance to Reflectance conversion: only target radiances are converted to reflectances; reference readings are ignored.
- Removed restriction in Data Loader that one directory can only hold one spectral file type: files may now be freely mixed.
- Support for new GPS format in SVC files produced by HR-1024i

#### Bugfixes:

- Metaparameter loading bug, appearing during target-reference linking.
- Support ASD Handheld files during DN to Radiance conversion
- Delta loading bug for files containing several spectra
- Campaign removing bug
- Bug that caused instruments with spaces and other special characters in their instrument name not to be inserted into the database during data ingestions, resulting in repeated instrument inserts.
- Bug in exclusive EAV selection method
- Bug that caused CSV writer to fail due to internal upgrade to Joda time.
- Wrong default storage field of Optics Name attribute.
- SVC time reading bug for PM time format

#### V3.2.0 Zeta

- Speedup: Added storage of previous path once a metadata file, e.g. a picture, has been loaded in the Metadata editor
- Fix for the OceanView file format to extract the instrument name: note that this can currently lead to wrong results when more than one instruments are connected and run; in our experience all spectral files will contain the same instrument name due to a bug in the OceanView software.
- Updates for the HR1024i loader:
  - Support for the Irradiance unit
  - More support for date formats. This is still a mess as the date format depends on the language setting of the machine controlling the spectrometer. If you experience an error while loading please forward an example of the file to the SPECCHIO team.
  - Support for commas used in floating point representations (again a language setting issue)

- Intermediate fix for dealing with Unix style paths; for some reasons in a mixed environment it happened that Unix forward slashes were converted to backward slashes when having a shared campaign with Unix and Windows machines.
- Support for different wavelength calibrations of the same instrument.
- Handling of errors when inserting campaigns via exceptions and printing into the server log
- Handling of empty spectral files lists during inserts; a condition that can appear when implementing file loaders in 3rd party languages like Matlab

#### Bugfixes:

- HR1024i: bugfix that prevented the wavelength calibration of an instrument to be inserted properly.
- Bugfix to display all input fields in the account creation dialogue also on some versions of Java on Windows machines
- Instrument type number used for instrument identification to prevent mismatches of instruments or re-inserts of instruments.

### V3.2.0 Eta

- New application version number and build handling, see 'Help'->'About' for the build information:

```
Version: 3.2.0 Eta
Build Number: 23
Build Date: Fri Aug 14 17:26:06 CEST 2015
```

#### Bugfixes:

- Implementation of new sun angle correction class by Klaus A. Brunner. [git@github.com:KlausBrunner/solarpositioning.git](https://github.com/KlausBrunner/solarpositioning.git). Corrects a bug in the solar angle calculation.
- Added exception handling for malformed auto numbered ASD file names
- Added a method to strip unprintable characters from comment strings in ASD files; such characters could mess up the JAXB encoding resulting in error code 400.
- Fixed a bug in the copySpectrum method that failed because of a non-existing field in the spectrum table

### V3.2.1.2

- Colouring of the Metadata Editor update button in red when changes are pending
- Added support for new timestamp field in the eav table
- Improved error handling for null value EAVs

- Update to allow inserting campaigns without a path (useful for programmatic access)
- Massive speed improvement for spectral databrowser sorting by attribute
- Improved API for easier programming
- Changed CSV writer to e notation
- Enable entering of spectrum number in Spinner component of spectrum report
- Added tooltip how to add a new metaparameter in Metadata Editor
- New functions to copy hierarchy and campaign ids to clipboard.
- Update to allow admin users to edit metadata by showing all campaigns.
- New feature: add new institute button for instrumentation panels.

#### Bugfixes:

- Bugfix to properly update instrument cache when new calibration is inserted. Stops the generation of new instrument inserts during each load.
- Bugfix for filter by eav methods.
- Clearing of redundancy buffer when removing metaparameters to avoid errors due to linking to deleted parameters

### **V3.2.1.3 Alpha**

Integrated changes applied within the dc-10 project:

- Update for new account detail storage and error handling
- Internal interface updates for ANDS server communication

General changes:

- Bugfix to include spectra not having attribute that is selected for sorting by.
- Bugfix for an error during inserting new instruments from a spectral file that had no spectral acquisition time.

### **V3.2.1.4 Alpha**

General Changes:

- New methods to work with hierarchies: getHierarchyName, getHierarchyFilePath, renameHierarchy. Currently only available via the specchio\_client when using higher level languages.
- Support for optional creation of measurement unit folders for old ASD file format.
- New menu item for SPECCHIO preferences. Renamed menu item: Compute UTC
- SPECCHIO Preferences dialog: optional creation of measurement unit folders for old ASD file format
- Sun angle calculation: Update to use the attribute 'Acquisition Time (UTC)' as time input.

- UTC time: UTC time is now saved in new attribute 'Acquisition Time (UTC)', making a clear distinction between acquisition time logged by the spectrometer and actual UTC time.
- Build number added to version in main window and About window.

Bugfixes:

- Bug fix for very long integration times in ASD spectrometers
- Bugfix to have correct path to keystore file in deployed applications.
- Re-enabled the deleting of old datalinks to allow removing data loaded with previous specchio versions. Fix to allow data deletion code to run on newer databases as well.
- Bugfix to allow removing obsolete datalinks.
- XLS file loader: Bugfix to load data with all decimals.

### **V3.2.1.5 Alpha**

General Changes:

- Order folders below the campaign level by name.
- Metadata Editor: Support for putting the input focus on newly added metaparameter fields for faster data input.
- Metadata augmentation from XLS files: Improved matching of strings and numbers: number in XLS can be matched versus a string field in the DB.
- Edit db\_config file now creates the file automatically and pre-fills it with account information taken from the preferences store. Menu function is disabled by default and must be enabled via the SPECCHIO Preferences.

Bugfixes:

- Data loading: Files with identical file names on higher hierarchy levels did not load.
- Bugfix for reflectance plots of spectra that are beyond the visible range: auto-y axis range now also working for single spectra.

### **V3.2.1.6**

General Changes:

- New methods for full text search (all string\_val fields in the EAV table)
- Added full user information to Campaign class
- Better GUI design by making spectral databrowsers scale when the window is enlarged.
- Improved GUI Layout for Data Remover Dialog
- Improved error message for renameHierarchy method. Automatic wildcards for fulltext search method.

- Update for Boolean metaparameter and checkbox support.
- Update for tooltips for taxonomies.
- Improved GUI layout of time shift dialogue
- Updates to allow reading ASD calibration files.
- Added code to detect corrupted ASD files.
- Update of the conflict detection algorithm for more coherent results
- Added Irradiance as auto-folder option for old ASD files
- New specchio client methods: get instrument object for a spectral file, get a list of metaparameter lists (i.e. get several metaparameters at once), added distinct control over selected metaparameter values, get the total number of spectra in the database, test the existence of a calibration in the database
- Improved exception handling for JAXB marshalling errors
- Support to insert ASD calibration information during spectral file loading.
- Support for new preference allowing control of ASD DN files are to be inserted in addition to the primary spectrum (e.g. Radiance)
- Support for new preference allowing control over the creation of unit folders for old ASD files
- Improved GUI layout of instrumentation metadata editor
- New admin function to get user contacts of current database.
- New functions in the metadata editor: switch off conflict detection to speed up data selection, new menu to access new functions: calculation of measurement support, altitude augmentation and E-W longitude switch
- New preferences to control the automatic insert of DN files for new ASD files and automatic insert of unit folders for old ASD files
- Modality update for progress reports to avoid changes in the calling GUI while updates are being carried out.
- Updated GUI layout of the Query Builder plus a new Run button to avoid time consuming query running in the background
- Add database info panel to main window
- Metadata Editor: Improved image support and firing of change events when a new field with a default value is added
- Added query support for taxonomies.
- Conflict detection speed-up in the metadata editor
- Automatic image resize support to reduce memory footprint (target height is 400 pixels)

#### Bugfixes:

- UTC time computation to update already existing UTC time stamps
- Bugfix when converting to Date: avoids conversion from UTC to local time. This impacts the time shown in the metadata editor, which was usually wrong, compared to the spectrum report.

- Bug fix of the target-reference linking function that used to hang.
- Bug fix for the auto-scaling of reflectance values in spectral plots
- Database insert updates to avoid SQL injections (those could lead to errors in string values)

### **V3.3.0.0**

#### General Changes:

- Spatial geometries support based on MySQL Spatial Extension, enabling spatial queries and new geometry types: point, polyline and polygon
- Application Domain support: configurable metadata categories per application domain for less cluttered metadata entry
- KML file reading to populate new spatial metaparameters using drag and drop
- Database upgrade function built into client and server application (for administrators only) with intelligent routine to apply required upgrades sequentially
- Import of campaigns from XML files stored on the Glassfish server (may be a better option to import very large campaigns)
- Support for SPECCHIO client preferences: input and output directories
- Corrected GPS longitude to standard format: positive is east of GMT. ASD and SVC file readers are updated to carry out the required conversion.
- ASD file reader update to support first generation ASD handhelds (old binary file version)
- Copy images to clipboard or open image in external viewer (from metadata editor and spectrum report)
- Visualisation of selected spectra from Metadata Editor
- E-W switch function in metadata editor for longitudes (updated to work with new spatial features)
- File drop for XML campaign import (drop on SPECCHIO Icon in main window)
- Extended information on database connection: version and indication of spatial DB support
- Lazy loading feature with progress report for bigger binary objects, (> 1MB) e.g. PDFs
- Range display for numeric data in case of metadata conflicts
- Sun angle calc enabled for all campaigns for admin users
- Update to show selected element when opening already defined taxonomies in Metadata Editor
- UTC computational meta entries are overwritten when re-running the UTC calculator function
- Progress bar update to be more informative
- Improved error messages by XML and HTML parsing of returned error data

- Improved metadata conflict detection to avoid wrong conflicts in some rare cases (same values stored under different eav entries)
- Campaign import: Reloading of caches to show newly imported attributes
- Added server app version and build

Bugfixes:

- Sun angle calc: angles were calculated wrongly in previous code versions: longitudes east of Greenwich must be positive, west longitudes are negative.
- Bugfix for the handling of Provenance Data Links
- Improved error handling and memory usage for campaign import and export routines to avoid crashes due to a list of reasons
- Prepared statements for inserts to avoid problems with special characters in strings