Rapporti Tecnici INAF
INAF Technical Reports

Number	245
Publication Year	2023
Acceptance in OA@INAF	2023-02-06T15:51:10Z
Title	SLab Database
Authors	CARRARO, Francesco; FONTE, SERGIO; CARLI, CRISTIAN
Affiliation of first author	IAPS Roma
Handle	http://hdl.handle.net/20.500.12386/33196; https://doi.org/10.20371/INAF/TechRep/245

SLab Database

(Part of SLab HYDRA)

Francesco Carraro

Sergio Fonte

Cristian Carli

1) Overview

1.1)   What is SLab Database?

The *SLab Database* is the core of the *SLab HYDRA* project and consists of 2 databases aimed to save and create backups of both data related to the SLab activities @ IAPS and data related to the users interested in participating the activities. The nature of above mentioned two kinds of data is quite different, being the first type related to science and the second type to personal information. This led to the decision of creating two different databases, called *SLabData* and *SLabUsers*.

In this user manual it will be explained how the databases are structured and how they interact with other components of the *SLab HYDRA* project.

1.2)   SLab HYDRA (HYperspectral Data Reduction and Analysis)

SLab Data Manager is part of a set of informatic tools consisting of in:

- *SLab Data Manager*: UWP app for processing, analyzing and saving data (INAF handle)
- *SLab Web Data Manager*: web app that is being developed at this time
- *SLab Database*: the centralized database containing all the info about registered users and data, discussed in present manual.

These tools build up an integrated environment for easing and speeding up as much as possible the reduction and the analysis of spectra acquired during activities carried on @ IAPS SLab.

2) SLabData database

The *SLabData* database is designed to host all data about science activities carried on @ SLab laboratory. This means that all information about spectra, processing steps, sample composition and geometry of the measurement have dedicated tables.

A complete schema of the *SLabData* database, showing relationships between tables, is shown in Figure 1.

2.1) Rules for nomenclature

Before starting the description of the tables constituting the database, a brief description of rules for nomenclature used in assigning names to tables is provided.

- Tables containing main data have representative names
- Tables with read-only data (no adding of new data by users is expected) have name starting with *ReadOnly* prefix
- Tables linking two other tables by means of IDs have name starting with *Link* prefix and the remaining part of the name is the conjunction of names of linked tables (or the main representative section of the names, to avoid excessively long names).

2.2) Info about operating user

A best practice widely used during the design phase of a database, is considering a way to save data about user (called operating user) who added/changed/deleted a row in a table.

Following this practice, all tables, except the *ReadOnly* ones, contain fields about the ID of the user who operated a change in a row. Fields for saving info about operating user are the following ones:

- *InsertUserId* filled when the row is added
- *EditUserId* filled when content of the row is changed
- *DeleteUserId* filled when row is deleted

2.3) Timestamp

As long as info about operating user are saved, also timestamp of the operation is saved for each added/changed/deleted row. Fields for saving timestamp are the following ones:

- *InsertDateTime* filled when the row is added
- *EditDateTime* filled when content of the row is changed
- *DeleteDateTime* filled when row is deleted

2.4) Rules for deletion of a row

No physical deletion of rows is available in *SLabData* database, only logical deletion is possible. This choice allows the recovery of critical information that has been deleted by mistake. Information about deletion of a row is provided by the *IsDeleted* field, containing a bit (0/1) value.
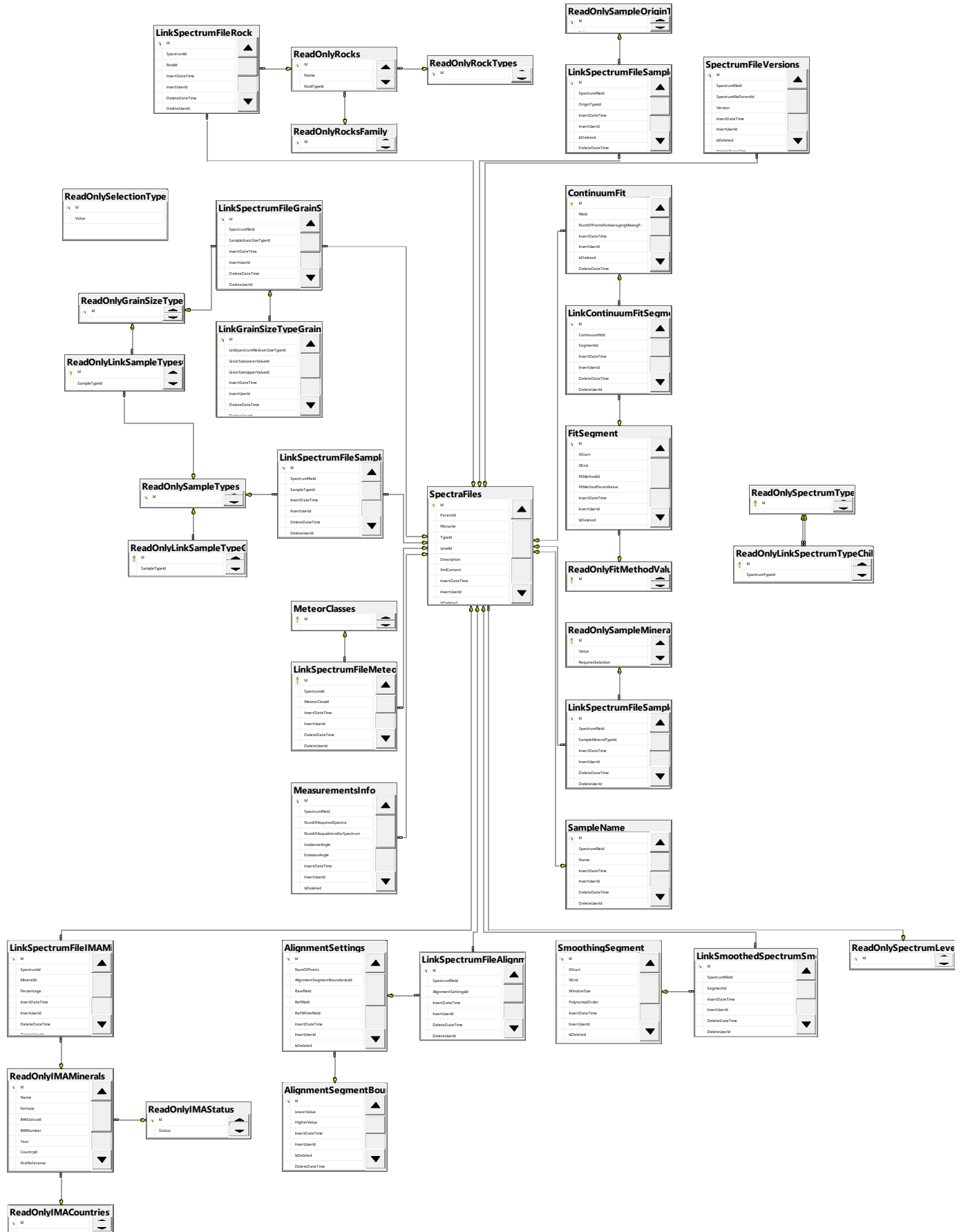
Figure 1 Schema of the SLabData database

2.5) Tables overview

Tables forming the database can be divided in groups, each related to a specific kind of information. Following an overview of those groups is presented.

2.5.1) Main info tables

This is the group holding the core tables which store information about spectra. The core table is named *SpectraFiles*. It contains main information about the spectrum, including the XML version of the spectrum file into the *XmlContent* column. Other info regarding the spectrum type (the kind of processing), parent files, the version of spectrum (when many processed files have been calculated by using different parameter values starting from the same spectrum), etc. are stored in those tables.

2.5.2) Alignment settings tables

Tables belonging to this group, contains settings on the alignment applied by the user to the raw spectrum.

2.5.3) Continuum fit settings tables

This group includes tables containing information about segments used for fitting the aligned spectrum: spectral range of the segment, method and parameter, when available.

2.5.4) Smoothing settings tables

This group includes tables storing information about settings used for calculating smoothed spectrum: usage of segments or whole spectrum, segments spectral range (if used) and parameters for the *Savitzky – Golay* algorithm.

2.5.5) Read-only tables

This is a group holding several tables containing data related to different kind of information, i.e.: sample composition, grain size, origin of sample, available spectrum types, available methods for fitting or for smoothing, etc.

3) SLabUsers database

The *SLabUsers* database hosts information related to users who are registered for logging into the *SLab HYDRA* system and have been granted with a level of authorization.

3.1) Impact of GDPR in designing the database

Information about registered users has been segregated into a dedicated database, separated from the one holding science data, in order to respect the E.U. *General Data Protection Regulation* (GDPR) that implies several cares in storing data also protecting privacy of users. One of the first requirements to fit the GDPR, as example, is encrypting the database.

Being personal data subject to a series of rules, the database containing this kind of data requires a bit more accuracy and complexity in being managed.

To avoid those complications in managing scientific data, which constitute the major amount of data with respect to the minimal set of personal information used by *SLab HYDRA* system for allowing users login, the decision of separating scientific data from personal data has been taken at the beginning of the design process.

A complete schema of the *SLabUsers* database, showing relationships between tables, is shown in Figure 2.

N.B. In designing the database, same rules applied to *SLabData* database for nomenclature, operating user, timestamp and rows deletion have been followed.
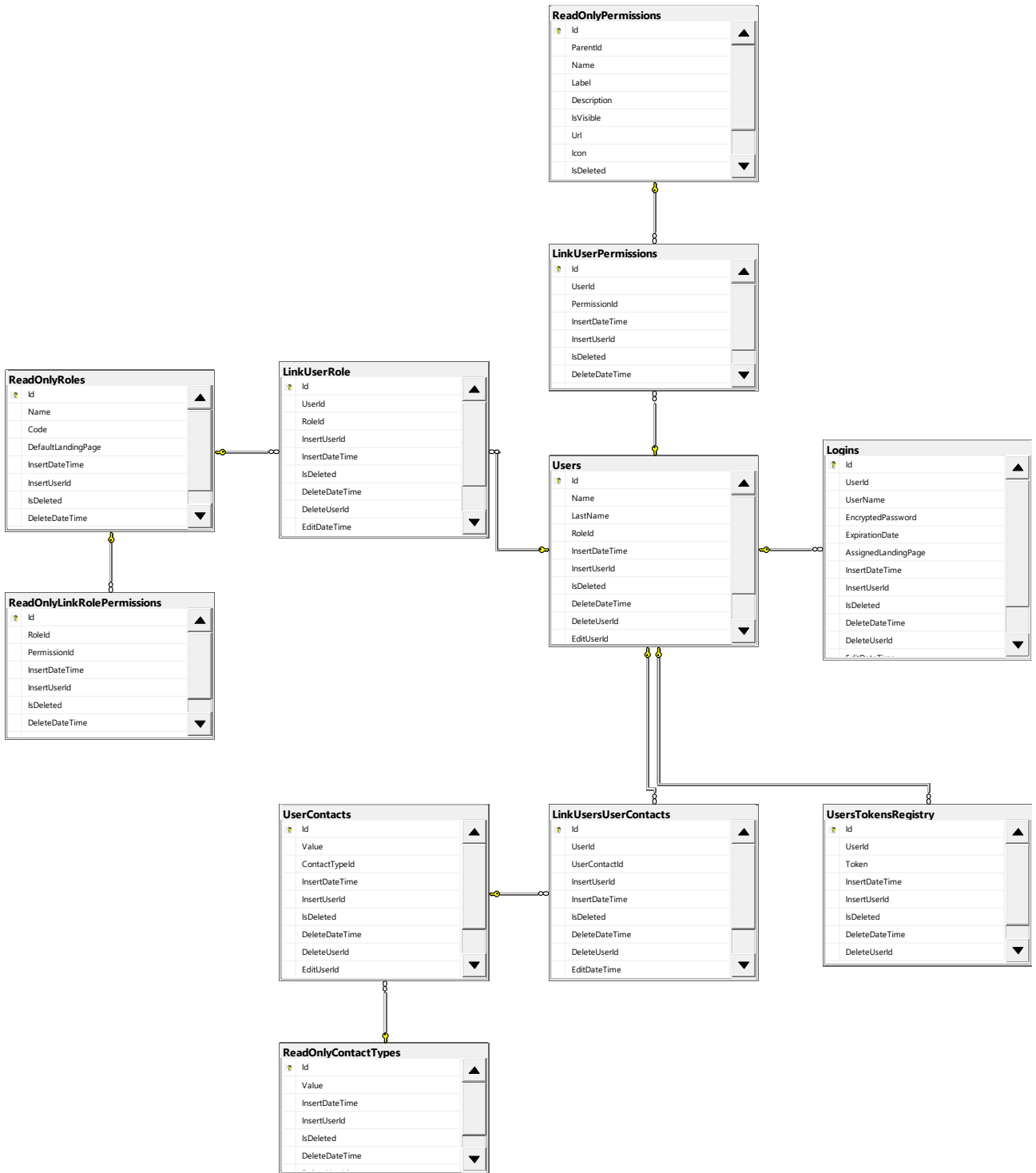
Figure 2 Scheme of SLabUsers database.

3.2) Tables overview

Tables constituting the database can be divided in a few groups, each related to a specific kind of information. Following an overview of those groups is presented.

3.2.1) User info table

The main group of tables includes the one dedicated to the small amount of personal info, the role and the email contact. This group also include the login information and the user token, automatically created for each user when logging into the *SLab HYDRA* system by using the *SLab Data Manager* app.

3.2.2) Role and permissions

A group of tables is devoted to defining the whole set of available permissions valid in the SLab HYDRA system, the available roles, the default sub-set of permissions related to each role and the permissions granted to each user when registered.

4) Acknowledgments