PROCEEDINGS OF SPIE

SPIEDigitalLibrary.org/conference-proceedings-of-spie

The HARPS-N archive through a Cassandra, NoSQL database suite?

Molinari, Emilio, Guerra, Jose, Harutyunyan, Avet, Lodi, Marcello, Martin, Adrian

Emilio Molinari, Jose Guerra, Avet Harutyunyan, Marcello Lodi, Adrian Martin, "The HARPS-N archive through a Cassandra, NoSQL database suite?," Proc. SPIE 9913, Software and Cyberinfrastructure for Astronomy IV, 99132A (26 July 2016); doi: 10.1117/12.2233137

SPIE.

Event: SPIE Astronomical Telescopes + Instrumentation, 2016, Edinburgh, United Kingdom

Downloaded From: https://www.spiedigitallibrary.org/conference-proceedings-of-spie on 15 Oct 2020 Terms of Use: https://www.spiedigitallibrary.org/terms-of-use

The HARPS-N archive through a Cassandra, NoSQL database suite?

Emilio Molinari[*a,b], Jose Guerra[a], Avet Harutyunyan[a], Marcello Lodi[a], Adrian Martin[a]
[a]Telescopio Nazionale Galileo, FGG-INAF, Rambla José Ana Fernández Pérez, 7
38712 Breña Baja, TF – Spain
[b]INAF-IASF-Milano Via E. Bassini 15 I-20133 Milano - Italy

ABSTRACT

The TNG-INAF is developing the science archive for the WEAVE instrument. The underlying architecture of the archive is based on a non relational database, more precisely, on Apache Cassandra cluster, which uses a NoSQL technology. In order to test and validate the use of this architecture, we created a local archive which we populated with all the HARPS-N spectra collected at the TNG since the instrument's start of operations in mid-2012, as well as developed tools for the analysis of this data set.

The HARPS-N data set is two orders of magnitude smaller than WEAVE, but we want to demonstrate the ability to walk through a complete data set and produce scientific output, as valuable as that produced by an ordinary pipeline, though without accessing directly the FITS files. The analytics is done by Apache Solr and Spark and on a relational PostgreSQL database. As an example, we produce observables like metallicity indexes for the targets in the archive and compare the results with the ones coming from the HARPS-N regular data reduction software.

The aim of this experiment is to explore the viability of a high availability cluster and distributed NoSQL database as a platform for complex scientific analytics on a large data set, which will then be ported to the WEAVE Archive System (WAS) which we are developing for the WEAVE multi object, fiber spectrograph.

Keywords: Databases, PostgreSQL, Cassandra, NoSQL, spectrograph, HARPS-N, WEAVE

1. INTRODUCTION

HARPS-N is the high resolution, high accuracy visual spectrograph installed at the Italian Telescopio Nazionale Galileo (TNG) in the island of La Palma in the Canary Islands[1,2]. The TNG is a 3.6 meter diameter reflector equipped with four focal plane instruments, for the imaging and for the spectroscopy in the visible and in the near infrared.

Spectroscopy is offered at low and very high resolution, and HARPS-N is capable of R=115000, and thanks to its very high stability (mechanical and thermal) can obtain measurements of radial velocities with a precision better that 1 m/s. This is necessary to characterize the mass of planets of dimensions like Earth or Superearths. Apart from radial velocity, astronomers find it really important to extract more physical parameters, like absorption line intensity and profile, in order to disentangle the activity of the parent star from its gravitational wobbling due to the presence of an orbiting planet.

We use here the complete HARPS-N dataset to test the performances of a database with about 30000 spectra, each with 300k of real numbers, plus around 100 parameters, metadata extracted from their original FITS files. The aim is to create a scientific validated environment with a high load of computation needed, to threat is using just databases, in view of the larger work we are developing for the WAS[3], the scientific archive and database for the WEAVE[4] multi object, fiber spectrograph for the WHT telescope in la Palma.

* emilio.molinari@tng.iac.es; phone +34 922 433-666; fax +34 922 420-508; www.tng.iac.es

Software and Cyberinfrastructure for Astronomy IV, edited by Gianluca Chiozzi,
Juan C. Guzman, Proc. of SPIE Vol. 9913, 99132A · © 2016 SPIE
CCC code: 0277-786X/16/$18 · doi: 10.1117/12.2233137

Downloaded From: https://www.spiedigitallibrary.org/conference-proceedings-of-spie on 15 Oct 2020
Terms of Use: https://www.spiedigitallibrary.org/terms-of-use

After building two separated databases, one using Cassandra and one using PostgreSQL, we try here to perform a bulk computation for a metallicity index, a common case for the astrophysicist which simply search for flux in a given range within the spectra.

2. DATABASE CONSTRUCTION

All the parameters and the calibrated spectra inserted into the database(s) are produced by the HARPS-N Data Reduction Sotware (DRS). After the technical manual, he DRS runs on-line on a dedicated Linux Workstation. The DRS deals with all the aspects of the scientific reduction of the raw data, as well as the processing of these reduced data to extract the radial velocities.

```
                                    Table "public.header"
               Column               |            Type             |             Modifiers
------------------------------------+-----------------------------+-------------------------
 id                                 | uuid                        | not null default
gen_random_uuid()
 date_obs                           | timestamp without time zone |
 mjd_obs                            | double precision            |
 program                            | text                        |
 ra_deg                             | double precision            |
 dec_deg                            | double precision            |
 el                                 | double precision            |
 coords                             | text                        |
 airmass                            | double precision            |
 exptime                            | double precision            |
 s1d_cdelt1                         | double precision            |
 s1d_crval1                         | double precision            |
 s1d_naxis1                         | double precision            |
 e2ds_naxis1                        | double precision            |
 e2ds_naxis2                        | double precision            |
 s1d_filename                       | text                        |
 blaze_filename                     | text                        |
 ccf_filename                       | text                        |
 e2ds_filename                      | text                        |
 hierarch_tng_obs_targ_name         | text                        |
 hierarch_tng_tel_targ_alpha        | text                        |
 hierarch_tng_tel_targ_delta        | text                        |
 hierarch_tng_drs_berv              | double precision            |
 pg_hk                              | double precision            |
 pg_hk_err                          | double precision            |
Indexes:
    "header_pkey" PRIMARY KEY, btree (id)



                    Table "public.s1d_flux"
  Column  |        Type        |             Modifiers
----------+--------------------+-----------------------------------
 id       | uuid               | not null default gen_random_uuid()
 flux     | text               |
 flux_arr | double precision[] |
Indexes:
    "s1d_flux_pkey" PRIMARY KEY, btree (id)
```

Table 1. The data model in the PostgreSQL couple of tables for the HARPS-N database. Table “public.header” has been deliberately truncated and only a small portion of the fields is shown.

Downloaded From: https://www.spiedigitallibrary.org/conference-proceedings-of-spie on 15 Oct 2020
Terms of Use: https://www.spiedigitallibrary.org/terms-of-use

The DRS does not interfere with the operation of the instrument, i.e. observations are independent from the Data Reduction activity. The DRS is designed to run automatically (no user interaction) like a batch process controlled by the Trigger right after the end of each exposure or at the completion of a sequence of exposures. It is also possible to use a DRS off-line to display and analyze reduced data, but here we rely on the produced one dimensional spectra.

```
                                        Table "header2"

CREATE TABLE harps.fluxes (
    id uuid PRIMARY KEY,
    cdelt1 float,
    cdelt1b float,
    crval1 float,
    crval1b float,
    fluxa text,
    fluxb text,
    s1da_filename text,
    s1db_filename text


                                        Table "fluxex"

CREATE TABLE harps.header2 (
    id uuid PRIMARY KEY,
    airmass float,
    bisa_filename text,
    ccfa_filename text,
    cdelt1 float,
    cdelt1b float,
    coords text,
    crval1 float,
    crval1b float,
    date_obs timestamp,
    dec_deg float,
    el float,
    exptime float,
    fitgderx float,
    fitgdery float,
    fitseex float,
    fitseey float,
    hierarch_tng_drs_berv float,
    hierarch_tng_drs_bis_span float,
    s1da_filename text,
    s1db_filename text
```

Table 2. The data model in Cassandra-Solr couple of tables for the HARPS-N database. Table “harps.header2” has been deliberately truncated and only a portion of the fields is shown.

These one dimensional spectra have a wavelength step of 0.01 Å and a range typically from 3830 to 6930 Å, yielding a 300k pixel FITS file. To this a series of calibration parameters is added, of the order of a hundred.

Each FITS file has thus around 300k real values which we ingest into the database, with different formats so to test the performance of data access, retrieval and table update. Figure 1 shows a typical star spectrum as produced by the DRS after matching every order of the echelle spectrum, taking into account blaze function and Thorium Argon calibration lamp. We will then focus on the bluest part of each spectrum in order to compute the Calcium H+K index, see par. 3.

Downloaded From: https://www.spiedigitallibrary.org/conference-proceedings-of-spie on 15 Oct 2020
Terms of Use: https://www.spiedigitallibrary.org/terms-of-use

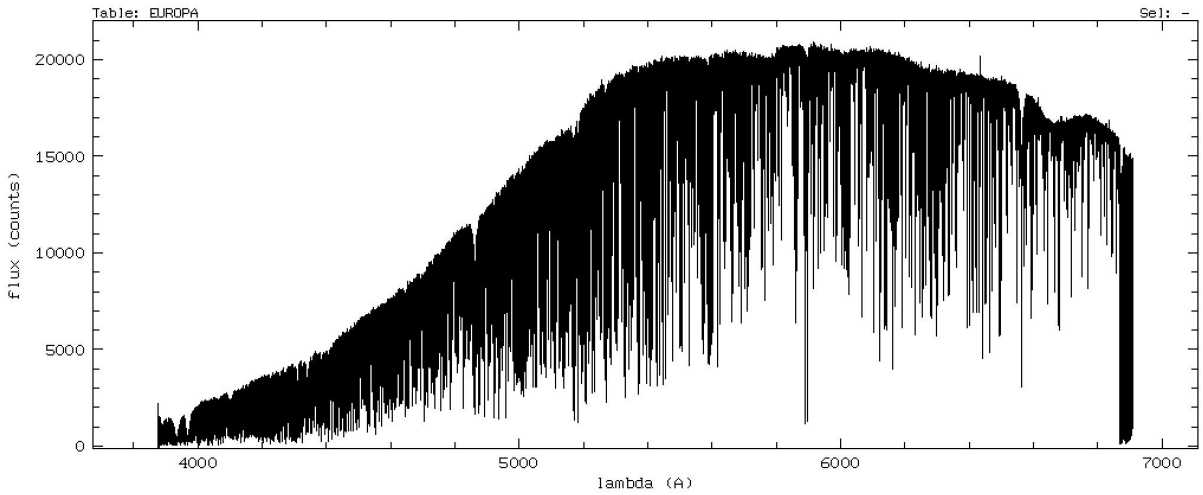

Figure 1 Full spectrum for a Solar like source observed with HARPS-N and reduced by the DRS in a one dimensional single spectrum.

We thus populated two different databases by means of a scanning of the complete directory tree of the FITS file repository, creating for each two separated tables: a header table containing all the input and derived parameters for each observed spectra, and a flux table, containing the 300k pixel values. Both tables contain a common column as a unique UUID identifier. Tables 1 and 2 show the two different, but very similar data models for the PostgreSQL and the Cassandra databases. We first decided to store fluxes as a compressed, csv text. In the PostgrSQL table there is also a version of these data stored as an array (see paragraph 3.2).

The two architectures do have major differences. The PostgreSQL is running on a single virtual machine, with 6 VCPUs and 4 GB RAM. To this a Gluster file system is connected with all the original FITS files.

The Cassandra cluster has two Solr nodes and three Spark nodes, running in virtual machines. The database is replicated twice in the Spark cluster. We opted for a deal with the private company Datastax to install its complete DSE, receiving free access to the suite in change of being an academic test case (no profit).

3. PERFORMANCES

We decided to concentrate on the calculation of one single metallicity index for stars. The Mount-Wilson S-index is a measure of the emission-line cores of the Ca II H and K lines at about 3933 A and 3968 A in two narrow bands normalized by two adjacent continuum bands. The activity index RHK is closely related to the S-index. In particular, it gives the emission in the narrow bands normalized by the bolometric brightness of the star, but here we limit ourselves to the S.-index, as a pure computation extracted from the spectrum. Figure 2 clearly shows the spectral region of the fluxes inside the lines (K and H) and the two continuum regions (V and R).

Downloaded From: https://www.spiedigitallibrary.org/conference-proceedings-of-spie on 15 Oct 2020
Terms of Use: https://www.spiedigitallibrary.org/terms-of-use

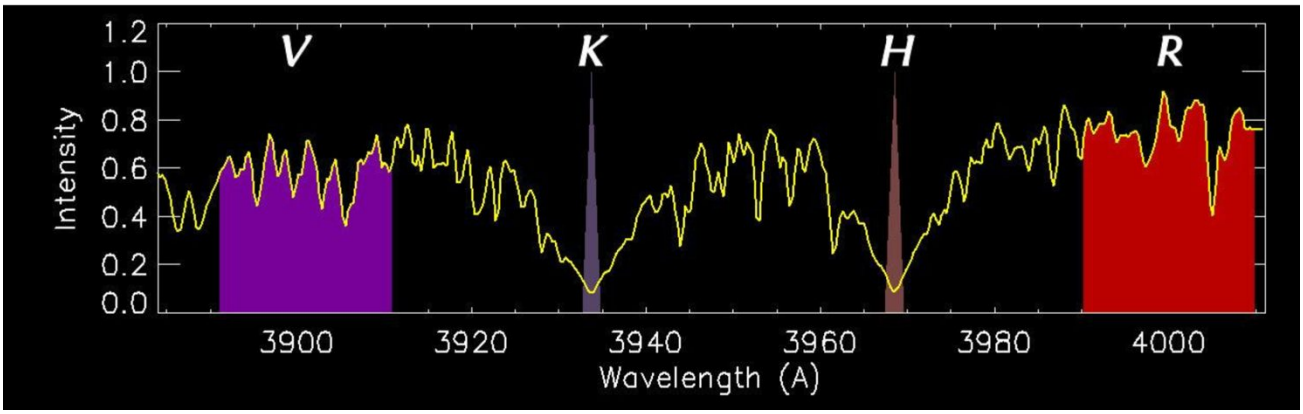

Figure 2. Graphical explanation of the S-index computation method. The enlarged spectral range in the 3900-4000 A region shows the continuum reference level (V and R) and the narrow bands used well inside the absorption lines (K and H). Adapted from Richard R. Radick AFRL/RVBXS.

The S-index is thus computed by the simple formula

$$S_{Ca_{II}} = \frac{f_H + f_K}{f_V + f_R} \tag{1}$$

where:

$f_H = \int_{3968.17}^{3968.77} flux(\lambda)d\lambda$, $f_K = \int_{3933.36}^{3933.96} flux(\lambda)d\lambda$, $f_V = \int_{3890}^{3910} flux(\lambda)d\lambda$, $f_R = \int_{3990}^{4010} flux(\lambda)d\lambda$,

for which all the spectra had to be beforehand shifted back to the restframe, making good use of the accurate radial velocity produced by the DRS of the HARPS-N instrument.

As a check on our computing method, we plot in Figure 3 the comparison with the S-index computed by the GAPS[5] group. The two set of values are identical within the errors,

3.1 Cassandra Spark unsuccessful experiment

Cassandra's data-model is basically a multi-dimensional map structure indexed by a key. The value is a highly structured object. Every operation on a row key is atomic per replica no matter how many columns are being read or written. Columns are grouped in what it is called Column Families (tables) similar to Bigtable structure.

In order to run this computation, a new data model with two tables was set up to in the cluster. One table keeps all the FITS header information and in another table the fluxes, as shown in Table 3. The sizes per node on disk of these two tables are 86GB for the spectra and 18GB for the header table.

To run this procedure as a Spark job, a Python script was written importing Sparks's DataFrames library to run a native SQL over the mentioned two Cassandra tables. Basically the algorithm is performed into two phases. The first one is the selection of targets and the second one is over the spectra table.

The algorithm at first performs a select in the header table looking for names of the target like M1, M2, .., M199, to locate the M type stars observed for a given program (we used here the GAPS research on exoplanet system architectures). Secondly, the identified IDs (uuid) are searched in the flux table and the spectra extracted to compute the S-index (formula 1).

Downloaded From: https://www.spiedigitallibrary.org/conference-proceedings-of-spie on 15 Oct 2020
Terms of Use: https://www.spiedigitallibrary.org/terms-of-use

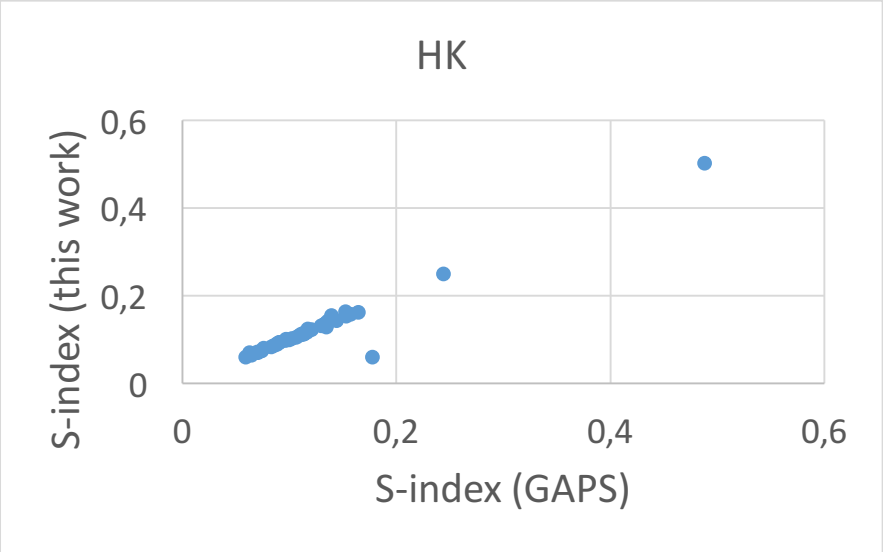

Figure 3. Comparison of this work with S-index values as computed by the GAPS group. The remarkable outlier is due to its unknown error value.

Selecting different subsets we can put different load on the system. Table 3 reports the numbers of selected stars. In our experiment the first selection of the header table took 17 seconds which was a very long time for a simple selection.. The second phase of retrieving the found uuid's in the fluxes table yielded a mean time of 0.7 seconds per spectra to resolve the query, which is also clearly slow.

Our conclusion is that a Spark cluster needs a long fine tuning before being effective, and turned to PostgreSQL to get a reference benchmark to be applied in the future.

Table 3. Subset of the M dwarf star program for the GAPS project, as used in this work to measure performances.

Star name	Number of spectra
M102	32
M0..M1	123
M0..M2	882
M0..M3	1118
M0..M7	1720
M0..M999	2931

3.2 PostgreSQL experiment with different data formats

We thus use a more classis PostgreSQL database, coding the S-index computation as a SQL function (language plpythonu).

Despite this database was running on a single machine we could establish a better performance than we obtained with Cassandra-Spark, so setting a firm bottom level for any subsequent evolution of our processing machine.

Downloaded From: https://www.spiedigitallibrary.org/conference-proceedings-of-spie on 15 Oct 2020
Terms of Use: https://www.spiedigitallibrary.org/terms-of-use

Using the same subset of Table 3, we applied the plpythonu to the tables, first selecting from header and then retrieving from fluxes. We then either printed the results to the screen/file or we updated the header table with the newly computed values of S-index and its error. Also, we tried using three different data storage for the spectra fluxes: csv text (like in Cassandra), direct access to the FITS files in a Gluster file system, and a sql array of float values.

The second access to the table for updating does not affect the performances but in the FITS file case, when the simple access to header table for the selection, followed by FITS file opening for the computation showed to be one of the fastest methods.

Figure 4 plots the timing per spectrum used in each case: as for low number of spectra the timing looks affected by the program setup, we report in Table 4 the time per spectrum of the longest case of almost 3000 S-indexes computed, possibly inserted in the header table.

Table 4. Mean time per computation of an S-index in a spectrum, using different methods of data storage, access and table update.

Method	Mean time (msec/spectrum)
PG+PG compute	267
PG+PG update	270
PG+FITS compute	86
PG+FITS update	183
PG+arr compute	171
PG+arr update	183
PG-array23k compute	38
PG-array23k update	41

To complete this experiment, we used all the available database, limiting the initial select query on header table with a LIMIT N instead of a search for star name. This gave better results, using as low as a 25 msec per spectrum, independently of the number N in the LIMIT.

4. CONCLUSIONS AND THE FUTURE

From the experiments reported in previous paragraph we cannot conclude anything on the use of a NoSQL database like Cassandra when we try to apply a distributed computation through Spark technology. Unfortunately Cassandra-Spark test was unsuccessful even showing different results for very basic algorithms resolved on Python and Scala. This shows Spark's DataFrames is a nice API allowing to work with native SQL over different structures however it seems get stuck with heavy data structures such the one for the spectra. It was also scaled up to an Amazon EC2 cluster with five nodes running Spark with the same dataset and Cassandra & Spark's versions, and we obtained a similar issue with the DataFrame's join SQL feature. Our conclusion is that a Spark's clusters are necessary to work with a different data model and programming model in order to make it an effective tool on the area. The result of this test definitively encourage to make a depth analysis of new structures in order to change the current model of data and programming. The expectations are indeed supported by the experiment results made with the PostgreSQL database.

At first sight the access to the original FITS file instead of ingesting the numbers in the database may be a good solution, especially if no update of the original table is programmed. The possibility to access just a subrange in the SQL array gave

Downloaded From: https://www.spiedigitallibrary.org/conference-proceedings-of-spie on 15 Oct 2020
Terms of Use: https://www.spiedigitallibrary.org/terms-of-use

indeed the best results, but this will be to be balanced with the inevitable higher use of disk space and the necessity to assess the array index range before the SELECT query is issued.

Nevertheless, the linear behavior is a clear sign of the non parallel computing of this PostgreSQL experiment. This is loudly calling for a solution for large databases, like we will face for the WEAVE instrument. We will continue investigating the Apache Spark environment as well as distributed file systems.

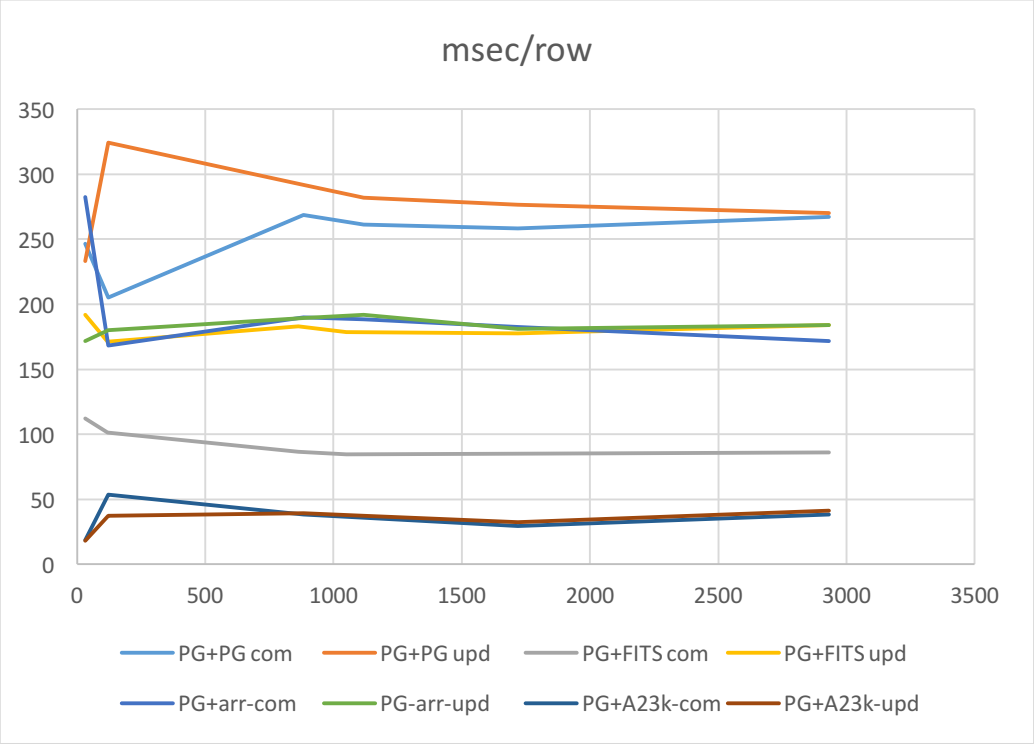

Figure 4. Mean time spent calculating the S-index for the sets of stars as in Table 3. Four methods are plotted, each with its values for just computing (com) and those also for updating (upd) the header tables with the new values. PG+PG: PostgreSQL with spectra stored as csv text; PG+FITS: spectra values are accessed opening the corresponding FITS file; PG-arr: spectra stored as SQL array; PG-A23k: spectra stored as SQL array but only the first 23k values (of 300k) are retrieved by the query.

REFERENCES

[1] Cosentino R., et al. "Harps-N: the new planet hunter at TNG ", Proc. SPIE 8446, 1 (2012).

[2] Cosentino R., et al. "HARPS-N @ TNG, two year harvesting data: performances and results ", Proc. SPIE 9147, 8 (2014)

[3] Guerra J., et al. "WAS: The Archive for the WEAVE Spectrograph", Proc. SPIE 9913-71 in this conference (2016)

[4] Dalton, G. et al. "Project overview and update on WEAVE: the next generation wide-field spectroscopy facility for the William Herschel Telescope", Proc. SPIE 9147, 0L (2014)

[5] Covino E., et al. "The GAPS programme with HARPS-N at TNG. I. Observations of the Rossiter-McLaughlin effect and characterisation of the transiting system Qatar-1 ", A&A 554, a28 (2013)

Downloaded From: https://www.spiedigitallibrary.org/conference-proceedings-of-spie on 15 Oct 2020
Terms of Use: https://www.spiedigitallibrary.org/terms-of-use