The SAMI Galaxy Survey: A Prototype Data Archive for Big Science Exploration[☆]

Iraklis S. Konstantopoulos[a,b,1], Andrew W. Green[a], Caroline Foster[a], Nicholas Scott[c,b], James T. Allen[c,b], L. M. R. Fogarty[c,b], Nuria P. F. Lorente[a], Sarah M. Sweet[d], Andrew M. Hopkins[a], Joss Bland-Hawthorn[c], Julia J. Bryant[c,a,b], Scott M. Croom[c,b], Michael Goodwin[a], Jon S. Lawrence[a], Matt S. Owers[e,a], Samuel N. Richards[c,a,b]

[a]*Australian Astronomical Observatory, PO Box 915, North Ryde NSW 1670, Australia; iraklis@aao.gov.au*
[b]*ARC Centre of Excellence for All-sky Astrophysics (CAASTRO), 44 Rosehill Street, Redfern NSW 2016, Australia*
[c]*Sydney Institute for Astronomy (SIfA), School of Physics, The University of Sydney, NSW 2006, Australia*
[d]*Research School of Astronomy and Astrophysics, Australian National University, Cotter Rd., Weston ACT 2611, Australia*
[e]*Department of Physics and Astronomy, Macquarie University, NSW 2109, Australia*

Abstract

We describe the data archive and database for the SAMI Galaxy Survey, an ongoing observational program that will cover ≈ 3400 galaxies with integral-field (spatially-resolved) spectroscopy. Amounting to some three million spectra, this is the largest sample of its kind to date. The data archive and built-in query engine use the versatile Hierarchical Data Format (HDF5), which precludes the need for external metadata tables and hence the setup and maintenance overhead those carry. The code produces simple outputs that can easily be translated to plots and tables, and the combination of these tools makes for a light system that can handle heavy data. This article acts as a contextual companion to the SAMI Survey Database source code repository, `samiDB`, which is freely available online and written entirely in Python. We also discuss the decisions related to the selection of tools and the creation of data visualisation modules. It is our aim that the work presented in this article—descriptions, rationale, and source code—will be of use to scientists looking to set up a maintenance-light data archive for a *Big Science* data load.

Keywords: astronomical databases: miscellaneous, surveys, virtual observatory tools, methods: miscellaneous

1. Introduction

Astronomy has firmly stepped into the *Big Science* realm. While in-depth studies of individual systems are unlikely to ever stop uncovering exciting new physics, such knowledge is now being solidified by investigations of systems in their hundreds, thousands, and even millions in certain disciplines. Surveys of galaxies have certainly come a long way since the CfA redshift survey of Davis et al. (1982) collected some 2400 redshifts. Modern prize contenders now collect orders of magnitude more redshifts, with BOSS planned to exceed one million objects (≈ 1.6 million unique spectra, Dawson et al., 2013).

Integral-field spectroscopy (IFS; also referred to as three-dimensional spectroscopy or hyperspectral imaging) is now entering this stage, with the record-breaking CALIFA survey (Sánchez et al., 2012) having overtaken the 260-galaxy milestone of ATLAS^{3D} (Cappellari et al., 2011). The SAMI Galaxy Survey became the leader in sample size in the second half of 2013 and at the time of writing has exceeded 1000 galaxies, owing to the multiplexing of its instrument: thirteen IFS units (IFUs) feeding one spectrograph (Croom et al., 2012). Not in the millions yet, but the samples of IFS surveys coming from multiplexed instruments—the stuff of imagination a decade or so ago—are projected to soon reach the 100,000 mark (*e. g.*, the Hector survey concept: Lawrence et al., 2014; Bland-Hawthorn, 2014). Since each IFS cube contains many spectra even moderate IFS surveys will soon hold many more unique spectra than those that collect a single spectrum per object (the SAMI Survey will collect some three million spectra).

Needless to say, the data these surveys are collecting will be challenging to archive. In the near future it will not be the volume that imposes limitations, but the complexity of the data and the sort of rapport an end-user will seek to establish with the archive. IFS data are commonly referred to as 'cubes'; in astrophysics the three dimensions to which this alludes are typically a pair of spatial (celestial) ordinates and a wavelength axis, although there are very suitable applications for such data organisation in time-domain astrophysics. Cube elements are named 'spaxels' or 'voxels', the former referring to the spatial sampling element, the latter to a single three-dimensional datum. The majority of current facilities, with the notable exception of the MUSE spectrograph (Bacon et al., 2010), give rise to relatively small IFS datacubes (∼ 100 MB). Data archiving at this order of magnitude might not seem like a great challenge, however, raw and intermediate data, which is nowadays never thrown away, typically increase the size of the archive by a factor of ten or more (to ∼ 1 GB). Thus a survey of 10^5 galaxies, covering the optical spectrum at

[☆]Source code is available on https://bitbucket.org/iraklis_k/samidb
[1]John Stocker Postdoctoral Fellow.

moderate spectral and spatial sampling (for example, the 2500 spaxels mapped by SAMI), would take up no less than $\sim$ 10 TB on disk. This is hardly in the *exabyte* regime, but well beyond the reach of small collaborations without the means to hire computer scientists for data archiving and developing the access tools that come with this.

The SAMI Survey data archive, `samiDB`, looks ahead to these future surveys, as does the instrument itself, a pathfinder for a massively multiplexed successor. The SAMI Survey will collect IFS data of 3,400 galaxies and has already, two years into data collection, exceeded the sample size of past surveys. In this paper we will describe the rationale that led to the design of `samiDB` and provide a textual companion to this publicly available Python source code (see footnote above). We hope this software will help small collaborations get a better grip on their data, either by direct adoption, or through adaptation and further development.

2. Conceptual Design

A database for a large astronomical survey is often built with SQL, the Structured Query Language. Data are stored in a POSIX directory structure using the astronomical FITS file format (flexible image transport system Wells et al., 1981; Pence et al., 2010), *metatables* that describe the data are constructed, a query tags all files that the end-user desires, and the data are usually delivered using a variant of the *wget* or *rsync* mechanism. While the SAMI Survey data reduction process has not moved away from the gold-standard of the FITS file, the databasing effort has. The reasons are two-fold:

1. The IFS-end of SAMI is newly built, but it feeds the existing AAOmega fibre spectrograph (Saunders et al., 2004), and therefore data are reduced with the standard `2dfdr` data reduction package[2].
2. The type of access we need to our data, is in some ways different to that sought by past spectroscopic surveys—for example, directly querying the datacubes as well as high-level science products.

Point (1) presents an obstacle, that the data are packaged in a way suitable for a multi-fibre spectrograph that takes multiple, but unassociated spectra: individual files contain 'row-stacked' spectra that correspond to all spectrograph fibres (all thirteen IFUs), rather than packaging on a per-unit basis. Point (2) refers to the fact that a SAMI Survey database user will often require access to only a part of an observation, and not a data-cube (and therefore FITS file) in its entirety. An example of inventive new data access methods that these new surveys allow is the extraction of a central spectrum, *e. g.* to mimic the fixed 2dF (Colless, 1999) or SDSS (York et al., 2000) aperture, since the information is already available in the integral-field cube. Such an extracted spectrum would help anchor previous results that extrapolate galaxy-wide information from its central spectrum. Such a request may be automatically issued by the user for a long list of targets satisfying some set of criteria, perhaps even all survey targets. The industry standard described above requires the predictive storage and metadata tagging of an all-new data product, packaged as one FITS file per target on disk, which involves opening and closing (or creating) potentially thousands of FITS files.

To avoid such pitfalls, the SAMI Survey data archive is instead dynamic, hierarchical, and monolithic, its specification based on three pillars: open source code, ease of maintenance, and efficient storage. The maintenance clause has the highest impact, as it imposes a widely used programming language, rather than a blend of tools in various languages. The *lingua franca* of the SAMI Survey is Python, which is certainly broadly used and open-source, but as an interpreted language introduces some overheads. The biggest drawback we immediately encountered in the initial design phase was the large input/output overhead of FITS files using the standard `pyfits` package (now incorporated into the `io` module of the `astropy` library). Reading a pair of SAMI data-cubes, each $50\times50\times2048 = 5.12\times10^6$ elements, to perform a query takes approximately a tenth of a second (according to benchmark tests by the astropy group[3]). Assuming that a user may need to query header information in each and every file, that would add up to a few hundred seconds to peruse the whole catalogue. Partly to take advantage of new features and partly to fulfil the specification we moved away from FITS files for data archiving, opting instead for the Hierarchical Data Format, HDF5[4]. A detailed comparison between the two formats will follow in Section 2.1. In brief HDF5 is a smart data container, not a filesystem or a data format. The source code that arose from this specification can be found on https://bitbucket.org/iraklis_k/samidb, and relevant documentation is also available online[5]. This setup reduces complexity for the above use-case of selecting spectra from a cube, as the hypothetical fibre spectra would be created on-the-fly from the data cube, and only require opening a single, albeit large, archive file, rather than thousands of FITS files. Finally, one could easily write a Python code to execute a custom data request, one that will not have been foreseen by the designers of the data archive and distribution system, thus replicating—arguably exceeding—the functionality of complex *table join* functions readily available in SQL.

2.1. HDF5 vs FITS

The FITS format has been the *de facto* choice in Astronomy for decades, owing to its capability to store mul-

[2] http://www.aao.gov.au/get/document/2dF-AAOmega-obs-manual-part4.pdf

[3] http://astropy.readthedocs.org/en/latest/io/fits/appendix/faq.html

[4] http://www.hdfgroup.org/HDF5/doc/H5.format.html

[5] http://irakliskonstantopoulos.com/samiDBrtd/

tiple datasets and limitless ancillary information in well-structured headers, and its interoperability across platforms. Virtually all software designed to interact with observational data is written with the format in mind. With the world of computing moving quickly around us, and with the dominance of the Macintosh operating system among astronomers over more stable flavours of Unix, it is becoming a challenge for our rather small scientific community to maintain the format and to keep the crucial interoperability factor that established FITS in the first place—one of the main advances of the FITS format is that it replaced the plethora of observatory-specific file formats in use by the community at the time. Two recent additions to the literature have delved into these issues in detail: Thomas et al. (2015) and Mink et al. (2014). While FITS is an obvious choice for any practical astronomical application, in archiving data it can become rather cumbersome on a regular desktop computer. As outlined in the above section, it is not designed for use with any high-level coding environments apart from the ones developed for this reason alone—`IRAF`, `MIDAS`, and `starlink`, the latter of which developed a pioneering hierarchical data format. The main obstacles we faced with manipulating FITS files with our Python codes were the following:

1. The relatively slow i/o (≈ 0.1 sec/file) prohibits querying and manipulating thousands of files in quick succession.
2. FITS headers are programmatically quite structured, rather than simple key-value pairs.
3. The distinction between 'primary' and 'image' or 'table' header units introduces another overhead in writing files, as an IFS data archive is meant to seamlessly combine tables, images, and spectra.

The i/o issues are not inherent to FITS, but its Python access module. The two other issues, however, are difficult to circumnavigate. Instead, HDF5 allows us to house a mix of two- and three-dimensional data products (even of mixed data types) for each survey target in the same conceptual and bitwise space—that is to say, not just as related files in a folder, but in the same file.

The two formats compare rather favourably, with HDF5 able to replicate most of the FITS functionality and adding some crucial extra features. All elements of a FITS file can be transcribed aptly: header data units become N-dimensional *datasets* that can have mixed data types (useful for storing tables); the filesystem is replaced by a POSIX-like linkage of datasets into *groups*; and the header becomes a set of *attributes* that describe any single dataset or group. Data are packed into *chunks* of a given size that can either be automatically optimised or user-defined, and zip-compressed for efficient storage with an equally short i/o overhead as FITS. There is no advantage over small FITS files (of order MB), but the *chunking* efficiency[6] begins to show past the 100 MB (raw) file size[7]. Since an HDF5 file can house any number of datasets this advantage begins to show once we combine data from only a dozen or so targets. Additionally, the HDF5 i/o overhead scales linearly with the number of datasets, while always outperforming FITS (accessed via `fitsio`) according to benchmark tests performed by the HDF Group (Pourmal, 2002). Meanwhile, attributes are kept accessible at the top level by the HDF5 filesystem, enabling very fast queries of attributes across thousands of datasets.

The interoperability that FITS files brought to our field enabled astronomers to easily share data. This is without a doubt a major advance. In a way HDF5 (or another widely used file format) can extend this effect and bring about an even greater benefit: by employing a file format that is used by millions of scientists around the world we may be able to improve our data processing, not to mention to share the burden of maintaining our file format of choice.

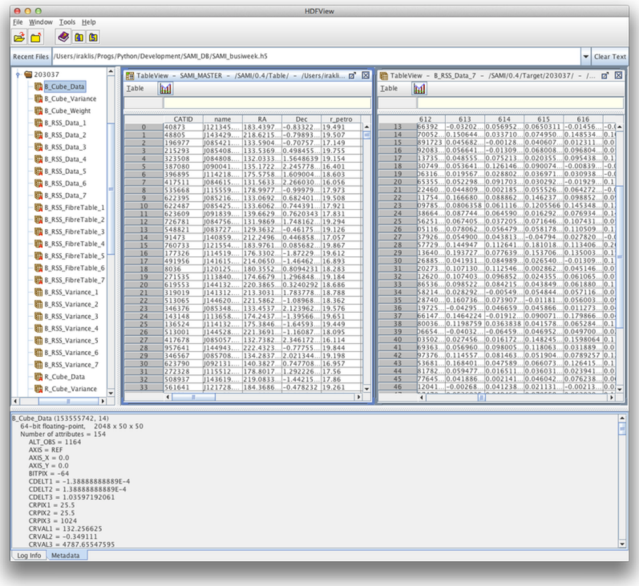

Figure 1: A visualisation of the contents of an HDF5 file containing the SAMI Survey archive, captured using the HDFView application provided by the HDF Group. The sidebar lists all datasets held within the group being perused. The main window tabulates two separate datasets: a stack of single spectra of like datatype (right), and a target table that contains a mix of datatypes (compound dataset). There is no need to save tables in separate files to images and spectra, as dictated by the FITS format. The bottom panel lists the attributes of the displayed spectroscopic dataset, a key-value pair dictionary. Data can be accessed directly via a prescriptive universal resource indicator that resembles a POSIX nested file path.

[6] When data are compressed into HDF5 packages, arrays are divided into appropriately sized *chunks*. HDF5 takes care of this by applying an algorithm to optimise the process, but the expert user can manually set the chunk size.

[7] For a thorough analysis of chunking optimisation see https://pytables.github.io/usersguide/optimization.html

2.2. HDF5 vs SQL

Another astronomy standard is the use of SQL, the Structured Query Language, for all database needs. The 'gold standard' is the SDSS database, which holds an intricate relational map linking a remarkable (and rather large) suite of metadata that describe the data and specific details of their acquisition. SQL was the preferred database language when the SDSS archive was designed and therefore constituted a natural choice. These days favour is shifting slightly away from strict relational databases, partly motivated by the 'NoSQL' movement (*e. g.* Sadalage, 2014) and the Apache-Hadoop filesystem in particular[8]. At its core sits the Hadoop Distributed File System, which is designed to take the computation requirement away from metadata tables, and bring it to the actual data. With the continuing growth in computing power this approach has become more and more popular, powered by MapReduce and other smart algorithms that process data in parallel, rather than swiftly but linearly, navigating a relational map. A debate of relative merits between the two approaches is beyond the scope of this article. Suffice it to say that the advantage a modern (non-relational) management schema brings to a scientific collaboration, and specifically a smaller project such as the SAMI Survey, is that it limits the overheads related to the setup and upkeep of a complex relational database. This is necessary in a very practical sense: while issuing SQL queries is becoming somewhat commonplace among astronomers, it is a very small subset of scientists in the field who are able to set up and maintain such infrastructure.

HDF5 offers a conceptually similar system as the Hadoop hierarchical filesystem. HDF5 is not a filesystem, however, but a smart data container (for a full discussion see Heber et al., 2014). *Datasets* are associated with *groups* and both types of data can be described by any number of *attributes*. The Python packages built to interface with HDF5, `h5py` and `PyTables`, take advantage of this simple data model by making an object out of each group and dataset when processing information. The HDF5 software automatically organises a set of metadata tables that reflect the relationship between datasets (and their attributes) and groups, without taxing the archivist. And above all, it, too, is designed to bring the computation to the data, granting versatility to the user to make queries up as she goes along, rather than being restricted to what is on the relational map. The compliance of the SAMI Survey data archive with the *ACID* principles of databasing will be discussed in Section 3.1. When it comes to selecting between SQL and HDF5 it is not so much a matter of superiority, but perhaps a matter of pragmatism. As scientists we tend to build our own software products, even though we often lack the relevant expertise, rather than outsourcing to professionals. In that sense, one should opt for the system with the fewest free parameters, an Occam's razor of practicality. This was the driver behind the decision for the SAMI Survey: when the maintenance of the data archive passes from one data archivist to his successor, the only requirement is a basic understanding of HDF5 containers (which are conceptually POSIX-like) and a working knowledge of Python.

[8] http://hadoop.apache.org/

2.3. Types of Data

In order to enhance the survey we store more than just the IFS cubes that come out of the SAMI Survey processing pipeline (Allen et al., 2014). From the telescope we receive raw data from seven dithering positions, which are arranged in such a way as to fill in the physical gaps between the 61 fibres that make up a SAMI 'hexabundle'. Since the light from all thirteen IFUs is fed to the AAOmega spectrograph, this is reduced using 2dfdr to produce 'row-stacked spectra' for all thirteen IFUs and 26 associated sky fibres. We provide this original dataset to benefit users who wish to delve deeply into the particularities of the data, *e. g.*, physics that may become complicated to deduce by the *drizzling* process, as described in Sharp et al. (2015). These two-dimensional data products therefore complement the drizzled three-dimensional datacubes.

In addition to the processed data themselves we also provide 'high-level science products' (HLSP) derived from or associated with the data, which can assume many forms of dimensionality: tables, images, or (hyper-)cubes. A complex example HLSP is a set of cubes containing fits to the spectral continuum and a large complement of emission lines (Ho et al, LZIFU, in preparation), as well as detailed fits to stellar kinematics (Fogarty et al., in preparation). The structure of these cubes is quite complex, reflecting the number of fits performed, and so the cubes are stored as a set of HDF5 datasets. This is the main form of HLSP we provide as part of the survey, but there will be many more types of ancillary data offered. A simpler and more common sort of HLSP is a table describing the sample of galaxies in its entirety.

These collections of data are stored in a logical structure that distinguishes between target-specific data and those that pertain to the sample as a whole. From this concept arises a simple data structure within the HDF5 file that is best expressed as a POSIX filesystem (see Figure 1):

```
./SAMI
./SAMI/vXX
./SAMI/vXX/Table
./SAMI/vXX/Target
./SAMI/vXX/Calibrator
```

The versioning loop (`vXX`, where `XX` is the version number) sits at the top of the package to facilitate quality control. In a system where more intricate control and testing features can be implemented, the versioning could take the form of higher dimensionality in the data, *i. e.*, new

versions are added as a 'veneer', a top layer, to an existing array[9]. The `Table` *group* contains the target table and all derived HLSPs that convey information about the whole sample, or any subset of more than one galaxy (target). Any IFS observation that is taken to calibrate data past the basic reduction stage is stored in the `Calibrator` group. Here we store 'secondary stars', observations of F-type stars taken at the same time as the galaxy data, in order to assess the extent of the seeing disk (as well as calibrating the data). A `Target` *group* contains any number of information only on a single target: two IFS cubes, one for each arm of the AAOmega spectrograph; the original stack of pre-drizzling spectra; LZIFU cubes; and any other HLSPs.

In terms of book-keeping, keeping calibrators in their own *group* helps to avoid needless CPU cycles during queries. Keeping single-galaxy and sample-wide HSLPs separate helps to direct the operation of the query code by starting from the top-down—that is, scan through tables before going into the heavy processing of looking for values in the large IFS cube datasets.

2.4. Schema Browser

One major benefit of this HDF5 archive and filesystem combination is that information can be (re)generated on-the-fly through a very simple code. Once a new HLSP is introduced its record is simply entered in the HDF5 file. Meanwhile, a simple Python code periodically scans through the HDF5 attributes and judges whether it needs to regenerate an HTML table that forms the schema browser on the SAMI Survey website. By saving on quality control and eliminating manual input of information into an HTML table, this zero overhead approach is ideal for small collaborations.

2.5. Visualisation Tools

Another great benefit of the HDF5 data archive is the ease with which web-based modules can access information. It is rather simple to create a figure given either a single or multiple HLSPs, or the collective attributes of a range of targets. The software tasked with generating these plots can take advantage of the simple i/o and only have to access one file to draw all the information required to create a figure. The setup and maintenance overheads of such operations are minimal. We will discuss particular tools in Section 6.

[9] Deleting data from an HDF5 package does, however, entail a potentially significant overhead. To *delete* data in HDF5 is to *unlink* it, meaning that the information remains within the HDF5 package, but is not referenced in metadata. True deletion follows the copying of a file to a new HDF5 package, during which no unlinked information is propagated. Unless data are constantly being refreshed this should not introduce a significant overhead, but the relative merits of the two methods are debatable.

3. Code Organisation

The code that pertains to all aspects of the data archiving effort is organised mostly in three management units, plus some `utils` packages for common use functions:

- `ingest.py` manages the importing of two- and three-dimensional data and HLSPs;
- `query.py` contains the set of tools that perform queries across all types of data;
- `export.py` handles the delivery of data products to the end user; and,
- smaller code units interface with the SAMI Survey website (http://sami-survey.org).

The conceptual flow of a `samiDB` query operation is as follows:

- Our hypothetical user enters a query on the web interface;
- a `PHP` handler script invokes `query.py`;
- `query.py` produces a list of target identifiers for which the query is satisfied;
- the ID list is passed to `export.py` which packages the data in the requested format.

For details of code operation and usage we refer the reader to the online documentation repository: http://irakliskonstantopoulos.com/samiDBrtd/. In the remainder of this article we will describe the web interface through which a user gains access to the data archive.

3.1. The ACID Test

The code is written with the *ACID* principles of databasing in mind. These were defined by Haerder and Reuter (1983) and summarised by Cook (2009) as follows:

> ***Atomic**: Everything in a transaction succeeds or the entire transaction is rolled back.*
>
> ***Consistent**: A transaction cannot leave the database in an inconsistent state.*
>
> ***Isolated**: Transactions cannot interfere with each other.*
>
> ***Durable**: Completed transactions persist, even when servers restart etc.*

`samiDB` is made *atomic* through the methodic catching of exceptions ('Python-`try`' rather than 'do'), such that transactions can exit cleanly, rather than being aborted. Through a rigorous quality control process we ensure that datasets are qualitatively correct (*e. g.* their dimensions and attributes are as expected) and hence ensure *consistency* in the database; the majority of lines in the `ingest.py` code is exception catching loops. We pursue *isolation* by

only updating the database when a new version of the data appears; this includes a fully reprocessed archive and a related set of HLSPs. Toward that end, HSLPs are version-controlled along with each data release, rather then being stored in multi-dimensional, version-controlled tables. This reduces efficiency but mitigates a severe risk. Few team members are given write access to the database to further manage risk. Finally, the database is made *durable* by never allowing the change of existing data. Instead data are version-controlled, as described above, which means that no datum is ever overwritten. In addition, a default setting in the ingest code creates a backup loop whereby a file is duplicated and time-stamped before a new version of the SAMI Survey data archive is ingested. Only once the process is completed and all quality control checks made is the code allowed to change the names of the two files: the backed-up duplicate retains the time stamp name, while the new file is now linked to the default filename, the object referenced by the server.

4. Programmatic Query Interface

The query language employed by `samiDB` is purpose-made under the hood but is formatted using standard Python notation of comparison operators (`==`, `!=`, `<>`, `>`, `<`, `>=`, `<=`), which can be joined with bitwise operators (`&`, `|`). The syntax intersperses the names of tables the user wishes to query with rows of comparison (query) syntax. The query code can receive this either as a string argument or as a text file. For example, a user wanting to search for the most massive galaxies might simply request information from a hypothetical table named `STELLAR MASS`:

```
STELLAR MASS
(logMstar > 10.0)
```

For a range of masses (`logMstar`) this would simply be:

```
STELLAR MASS
(logMstar > 8.0) & (logMstar < 10.0)
```

If the user wants to add, say, redshift (`z_spec`) information from another table she will need to name both tables in a stacked syntax to perform a *table join* operation:

```
STELLAR MASS
(logMstar > 8.0) & (logMstar < 10.0)
REDSHIFT
(z_spec > 0.02)  & (z_spec < 0.1)
```

Perhaps a user wishes to probe two parts of a certain parameter space, for example to find outliers in terms of stellar mass within a redshift bin:

```
STELLAR MASS
(logMstar < 8.0) | (logMstar > 10.0)
REDSHIFT
(z_spec > 0.02)  & (z_spec < 0.1)
```

Performing numerics within a table is handled entirely by Python:

```
PHOTOMETRY
(g_mag - i_mag > 1.0)
```

Numerical operations that cross tables must be handled by `query.py` using a function that is currently being tested by our team. Once testing is complete the following SQL-like syntax will be available to SAMI Survey users:

```
t1 = STELLAR MASS
t2 = SFR
(t1.sfr / t2.logMstar > 1e-9)
```

which would return galaxies with specific star formation rates in the star formation main sequence (Noeske et al., 2007). Table joins are performed automatically based on a unique SAMI Survey galaxy identifier.

5. The Data Browser

In the early data release of the SAMI Survey (EDR, Allen et al., 2015, sami-survey.org/edr/browser) we presented data through a simple interface, the Data Browser, as shown in Figure 2. The code that gives rise to this table will be used for all SAMI Survey queries in the future, as we will described in Section 6.

The Data Browser is a tabulation of descriptive imagery. It presents basic information for each target not in terms of representative numbers, but instead as three images and a schematic, in order to convey the three-dimensionality of the data and the multi-dimensionality of the ancillary data available via the GAMA Survey (Driver et al., 2011), from which the majority of the SAMI Survey sample is drawn (Bryant et al., 2015). Each row in the browser presents a *gri* colour image composite from SDSS (York et al., 2000) accompanied by two IFS maps (two-dimensional cuts): one of flux and another of velocity. A simple JavaScript code incorporates some minimal interactivity: by clicking and holding on either the SDSS image or the SAMI flux map, the display changes to show the velocity field. This is very helpful when perusing the data for striking features. In the case of the EDR and its a relatively small number (107) of galaxies, we have curated the table and selected the most appropriate SAMI Survey data products to display here: either emission line fluxes, or continuum. In future releases we plan to give the user control of this option, as curation at this level for a sample of 3,400 galaxies is beyond the means of the project.

The fourth and fifth columns of the browser contain more information in the form of (4) a *starfish diagram*, and (5) some pertinent information, such as identifiers, celestial coordinates, and links to the data plus other surveys that have covered these targets. The starfish diagram (Figure 2, right; Konstantopoulos, 2014, 2015) is a simultaneous visualisation of seven galaxy properties in a friendly set of glyphs–five arms and two conveyed through

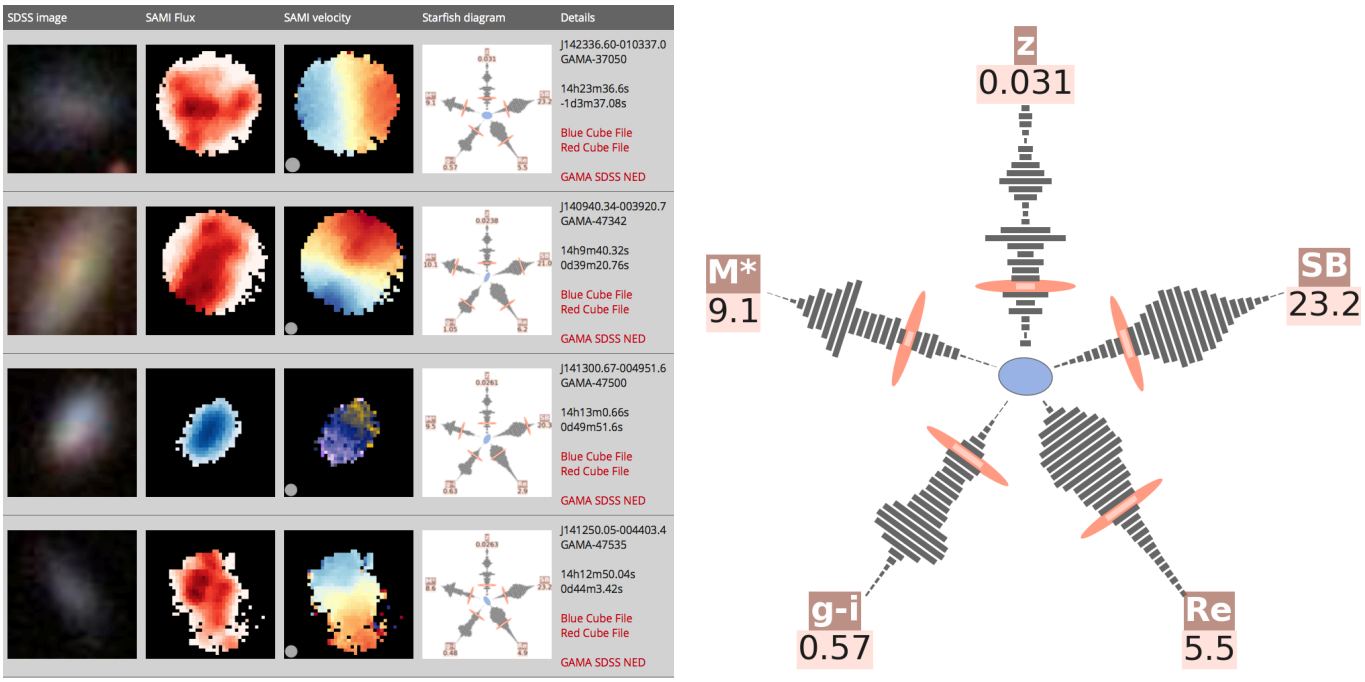

Figure 2: **Left**: The first few rows of the Early Data Release data browser, as described in the text. The first column shows an SDSS thumbnail, matched to the coordinates and spatial scale of the SAMI maps of flux and velocity that can be found in the following two columns. The circle in the bottom left corner of the velocity map represents the full width at half-maximum of the seeing disk over the course of the observation, which is determined through the simultaneous observation of a secondary standard star. The starfish diagram in the fourth column represents a set of galaxy properties in the context of the sample from which it is drawn (in this case the subset of SAMI Survey targets drawn from GAMA). **Right**: A close-up view of the starfish diagram on the first row. Each arm of the diagram shows a histogram of the distribution of a certain sample-wide property, with the value for the specific galaxy indicated by an ellipse. The 'glyph' in the centre of the diagram illustrates the major axis and orientation of the galaxy.

the shape of the central glyph. For the EDR these are redshift, stellar mass, colour excess, effective radius, and surface brightness at this radius, as well as position angle and ellipticity.

6. Web Interface

We are currently developing the tools that will allow our users to interface with `samiDB` via sami-survey.org. These features are not yet available to the community and so we present a concept-level description here. The inner workings of the portal itself[10] will not be discussed here.

The query box, an ASCII-input box embedded in a web browser window, is the main portal to the data archive. It directly invokes `query.py`, as per the programmatic interface described in Section 4. The box will receive either text or file input and return a Data Browser (see Section 5) tabulating only those galaxies that satisfy the query executed. The user will be able to further refine source selection with simple 'select' and 'select all' tick-boxes.

Another future portal into the data and the SAMI Survey sample is the interactive target table, created with the `bokeh` module for Python. The plot code is invoked via Python, but generates plots with JavaScript therefore creating an ideal balance between CPU requirement, maintenance overhead, and diagnostic power—with pleasing aesthetics out-of-the-box as an added bonus. Given the rapid development of the `bokeh` project we are concurrently developing a version of this tool using the much more stable `mpld3` library. This tool also receives input in Python and takes advantage of the commonly used `matplotlib` visualisation module to draw the vectors, before converting them to a JavaScript canvas with the `d3` library.

The target table, shown as a static image in Figure 3, displays the mass of every target in the 'master' table from which the majority of the SAMI Survey sample is drawn, against redshift. It features box and scroll zoom functions, a box selection tool, and a hover tool that reveals additional information about a target when the cursor meets its datapoint. The interactive table is designed for those users who wish to peruse the physical properties of the targets to inform their sub-sample selection. In a future release we intend to add a second step to the hover and selection tools, whereby through a second click the query command can be issued and the sub-sample prepared for the user.

Once past this first level of interaction with the sample the user will be able to peruse the data. A single-object viewer, currently in its design phase, will generate

[10] Developed by Andrew Green.

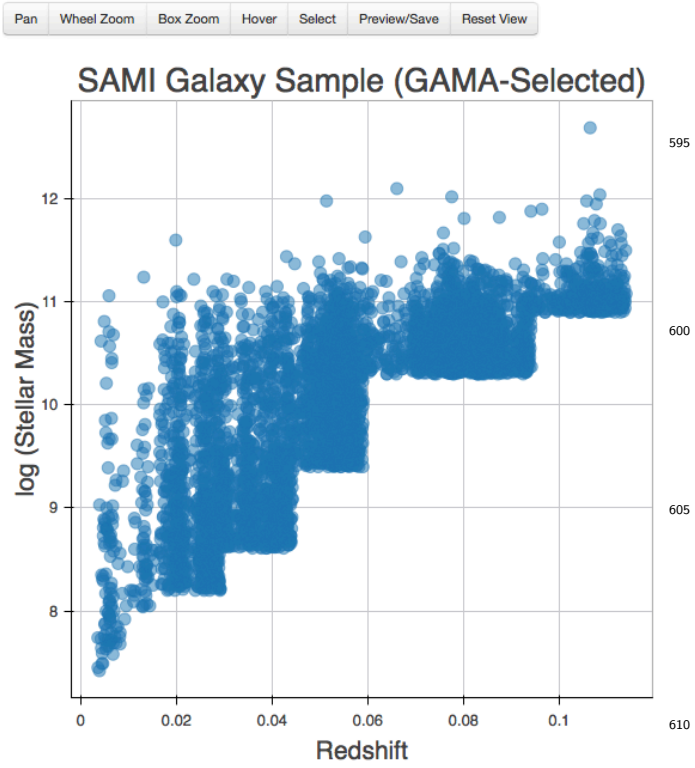

Figure 3: A screen-grab of the interactive target table designed for sample (pre-)selection. At the top of the screen the user is provided with a toolbar, which includes a box selection tool and a hover application, which reveals additional information when the user hovers the cursor above a data-point. This way the user can extract information that will inform their sample selection.

a customisable page of target information. This will act as an optional second level of interaction between the user and the data archive. The concept combines the qualities of two extremely successful viewers. First is the SDSS viewer, a standard reference for survey and single-object work alike. It presents a visual summary of a target with tabulated information as a secondary resource. The second is the GAMA viewer, which takes advantage of the treasury of ancillary information to visualise and tabulate information.

For the SAMI Survey we will create a tool that does both: a visual single-object viewer where the user selects what two-dimensional information is displayed. At the top of the concept design sits a single row of the Data Browser as a header field that can be selected to be 'sticky' at the top of the page. Below it will be a set of blank canvasses for the user to populate with their selected two-dimensional information.

6.1. Two Illustrative Use Cases

- *An investigation of star formation morphology in low-mass galaxies.* The user will have the option of either drawing a box on the target table, or executing a textual query into the box to restrict selection to nearby dwarf galaxies:

```
SAMI_MASTER
(logMstar < 9.0) & (z < 0.02)
```

The interface will present the user with a browser table where she can peruse the sample for the type of galaxies she wishes to study. The tick boxes or 'select all' tools will restrict the sample and the user will choose between a compressed package of FITS files or a single HDF5 file.

- *The Brightest H II Regions.* In this case the user wishes to query the data directly, rather than using metadata tables, to find any spaxels with fluxes higher than 10^{-12} erg/s/cm/Å. To achieve this the user will invoke the 'DATA' tag and point the query to a particular dataset:

```
DATA
(R_CUBE > 1e-12)
```

Since active galaxies will contaminate a sample selected purely on flux, the user could also use metadata; let us assume that an AGN classification HLSP is accessible:

```
DATA
(R_CUBE > 1e-12)
EMISSION-LINE-PHYSICS
(AGN == 0)
```

where the 'AGN' column is assumed to be a generic or bitwise boolean flag, with zero indicating no central activity. Once the query is complete, the interface will present the user with a browser table where she can peruse the data for the sorts of targets she was originally after and select all or a subset, or delve into the single-object viewer where emission line physics can be studied in more detail.

7. Summary and Discussion

In this article we have introduced `samiDB`, the archiving and query facility for the SAMI Galaxy Survey. The archive code was created with small scientific collaborations in mind and looks ahead to the advances and data volumes the next decade or so will bring to astrophysics and other sciences. The code is open-source, readily available, and well documented on the web. It is meant to be adapted to the needs of scientific (and other) collaborations and as such does not use technology specific to a small field of scientific endeavour—apart from the segment that imports images in the astronomical FITS format. The engine behind `samiDB` is HDF5, a technology that packages data into a hierarchical format thereby saving space on disk and requiring minimal processing prowess

to plough through complex data. The programmatic interface is written entirely in Python and it plugs neatly into a web front-end built with the `Drupal` content management system (the interface is under development). We welcome scientist-archivists to contribute functionality to the code and improve it.

The idiosyncrasies of the SAMI project dictate a very specific course for the manner in which data are to be archived and accessed. It has to be open source and ideally use as few languages as possible, so Python was chosen as the interface. It does not run on a server farm but a virtual machine, so it must be lightweight, therefore the processing is passed on to the client as much as possible. The reality of academia means that the original designer will be succeeded by at least one other archivist over the life of the project, so HDF5 was chosen to encapsulate all aspects of the back end and database; this way the collaboration need not recruit a specialist database administrator.

While this system is very easy to adapt and very well suited to a large range of scientific applications, archivists should be aware of other tools that are readily available to manage their data. A full discussion is beyond the scope of this work, but there are two free, open-source solutions that may be of particular interest to archivists in the physical, mathematical, and life sciences. One model for science data management is SciDB[11], a fully-fledged database management system (DBMS) based on exploiting arrays, rather than relational database tables. SciDB interfaces with R, Python, and more. The Large Survey Database[12] (LSD; Juric, 2011, 2012) was developed by Mario Juric for the PanSTARRS project (Chambers et al., 2007) in order to swiftly navigate very large catalogues.

Both these DBMS are designed with high-performance computing in mind. LSD is meant to be run on multiple cores in parallel, and SciDB is optimised to be scaled onto multiple nodes[13] The advantage that the modest `samiDB` solution offers is its lightness. While professional solutions will outperform it, `samiDB` makes up by being very simple to set up and adapt to any collaboration. It allows the scientist-archivist to edit the code and make it specific to the project as it is not generalised to begin with. And it does not push data to a specific format or organisation chart, as we are aware and sensitive to the fact that a data archive made for a state-of-the-art scientific endeavour will likely not fit comfortably to a pre-existing norm. In addition, `samiDB` is light enough to run on a virtual machine. We built it with modularity in mind; by keeping communication between individual codes very simple and unambiguous, a different interface can be implemented without changing any of the code, and more packages can be stacked on top.

In summary, the versatility that `samiDB` offers is a major boon for the science world where software is often an afterthought. Rather than trying to adjust data after the fact to fit a sensible archiving norm, reality may have it that the data reduction pipeline creates a unique and unpredictable form. The value of `samiDB` is that it introduces a realistic facility for scientific data archiving, one that will potentially help many science archivists while the science world catches up with the current technological boom, and software specification becomes a matter of habit.

Acknowledgements

ISK and JTA are both recipients of a John Stocker Postdoctoral Fellowship from the Science and Industry Endowment Fund.

MSO acknowledges the funding support from the Australian Research Council through a Future Fellowship Fellowship (FT140100255).

The SAMI Galaxy Survey is based on observations made at the Anglo-Australian Telescope. The Sydney-AAO Multi-object Integral field spectrograph (SAMI) was developed jointly by the University of Sydney and the Australian Astronomical Observatory. The SAMI input catalogue is based on data taken from the Sloan Digital Sky Survey, the GAMA Survey and the VST ATLAS Survey. The SAMI Galaxy Survey is funded by the Australian Research Council Centre of Excellence for All-sky Astrophysics (CAASTRO), through project number CE110001020, and other participating institutions. The SAMI Galaxy Survey website is http://sami-survey.org/.

References

References

Allen, J.T., Croom, S.M., Konstantopoulos, I.S., Bryant, J.J., Sharp, R., Cecil, G.N., Fogarty, L.M.R., Foster, C., Green, A.W., Ho, I.T., Owers, M.S., Schaefer, A.L., Scott, N., Bauer, A.E., Baldry, I., Barnes, L.A., Bland-Hawthorn, J., Bloom, J.V., Brough, S., Colless, M., Cortese, L., Couch, W.J., Drinkwater, M.J., Driver, S.P., Goodwin, M., Gunawardhana, M.L.P., Hampton, E.J., Hopkins, A.M., Kewley, L.J., Lawrence, J.S., Leon-Saval, S.G., Liske, J., López-Sánchez, Á.R., Lorente, N.P.F., McElroy, R., Medling, A.M., Mould, J., Norberg, P., Parker, Q.A., Power, C., Pracy, M.B., Richards, S.N., Robotham, A.S.G., Sweet, S.M., Taylor, E.N., Thomas, A.D., Tonini, C., Walcher, C.J., 2015. The SAMI Galaxy Survey: Early Data Release. MNRAS 446, 1567–1583. doi:10.1093/mnras/stu2057, arXiv:1407.6068.

Allen, J.T., Green, A.W., Fogarty, L.M.R., Sharp, R., Nielsen, J., Konstantopoulos, I., Taylor, E.N., Scott, N., Cortese, L., Richards, S.N., Croom, S., Owers, M.S., Bauer, A.E., Sweet, S.M., Bryant, J.J., 2014. SAMI: Sydney-AAO Multi-object Integral field spectrograph pipeline. Astrophysics Source Code Library. arXiv:1407.006.

Bacon, R., Accardo, M., Adjali, L., Anwand, H., Bauer, S., Biswas, I., Blaizot, J., Boudon, D., Brau-Nogue, S., Brinchmann, J., Caillier, P., Capoani, L., Carollo, C.M., Contini, T., Couderc, P., Daguisé, E., Deiries, S., Delabre, B., Dreizler, S., Dubois, J., Dupieux, M., Dupuy, C., Emsellem, E., Fechner, T., Fleischmann, A., François, M., Gallou, G., Gharsa, T., Glindemann, A., Gojak, D., Guiderdoni, B., Hansali, G., Hahn, T., Jarno, A., Kelz, A.,

[11] http://www.scidb.org

[12] http://research.majuric.org/trac/wiki/LargeSurveyDatabase

[13] This is not a high-performance computing premise, rather computation is spread across many cheap nodes; this is still beyond the means of most scientific collaborations.

Koehler, C., Kosmalski, J., Laurent, F., Le Floch, M., Lilly, S.J., Lizon, J.L., Loupias, M., Manescau, A., Monstein, C., Nicklas, H., Olaya, J.C., Pares, L., Pasquini, L., Pécontal-Rousset, A., Pelló, R., Petit, C., Popow, E., Reiss, R., Remillieux, A., Renault, E., Roth, M., Rupprecht, G., Serre, D., Schaye, J., Soucail, G., Steinmetz, M., Streicher, O., Stuik, R., Valentin, H., Vernet, J., Weilbacher, P., Wisotzki, L., Yerle, N., 2010. The MUSE second-generation VLT instrument, in: Society of Photo-Optical Instrumentation Engineers (SPIE) Conference Series, p. 8. doi:10.1117/12.856027.

Bland-Hawthorn, J., 2014. The Hector Survey: integral field spectroscopy of 100,000 galaxies. ArXiv e-prints arXiv:1410.3838.

Bryant, J.J., Owers, M.S., Robotham, A.S.G., Croom, S.M., Driver, S.P., Drinkwater, M.J., Lorente, N.P.F., Cortese, L., Scott, N., Colless, M., Schaefer, A., Taylor, E.N., Konstantopoulos, I.S., Allen, J.T., Baldry, I., Barnes, L., Bauer, A.E., Bland-Hawthorn, J., Bloom, J.V., Brooks, A.M., Brough, S., Cecil, G., Couch, W., Croton, D., Davies, R., Ellis, S., Fogarty, L.M.R., Foster, C., Glazebrook, K., Goodwin, M., Green, A., Gunawardhana, M.L., Hampton, E., Ho, I.T., Hopkins, A.M., Kewley, L., Lawrence, J.S., Leon-Saval, S.G., Leslie, S., McElroy, R., Lewis, G., Liske, J., López-Sánchez, Á.R., Mahajan, S., Medling, A.M., Metcalfe, N., Meyer, M., Mould, J., Obreschkow, D., O'Toole, S., Pracy, M., Richards, S.N., Shanks, T., Sharp, R., Sweet, S.M., Thomas, A.D., Tonini, C., Walcher, C.J., 2015. The SAMI Galaxy Survey: instrument specification and target selection. MNRAS 447, 2857–2879. doi:10.1093/mnras/stu2635, arXiv:1407.7335.

Cappellari, M., Emsellem, E., Krajnović, D., McDermid, R.M., Scott, N., Verdoes Kleijn, G.A., Young, L.M., Alatalo, K., Bacon, R., Blitz, L., Bois, M., Bournaud, F., Bureau, M., Davies, R.L., Davis, T.A., de Zeeuw, P.T., Duc, P.A., Khochfar, S., Kuntschner, H., Lablanche, P.Y., Morganti, R., Naab, T., Oosterloo, T., Sarzi, M., Serra, P., Weijmans, A.M., 2011. The ATLAS^{3D} project - I. A volume-limited sample of 260 nearby early-type galaxies: science goals and selection criteria. MNRAS 413, 813–836. doi:10.1111/j.1365-2966.2010.18174.x, arXiv:1012.1551.

Chambers, K., Pan-STARRS Telescope No. 1, PS1 Science Consortium, 2007. The PS1 Sky Surveys and Science Mission, in: American Astronomical Society Meeting Abstracts #210, p. 171.

Colless, M., 1999. First results from the 2dF Galaxy Redshift Survey, in: Efstathiou, G., et al. (Eds.), Large-Scale Structure in the Universe, p. 105.

Cook, J.D., 2009. ACID vs BASE for database transactions. URL: http://www.johndcook.com/blog/2009/07/06/brewer-cap-theorem-base/.

Croom, S.M., Lawrence, J.S., Bland-Hawthorn, J., Bryant, J.J., Fogarty, L., Richards, S., Goodwin, M., Farrell, T., Miziarski, S., Heald, R., Jones, D.H., Lee, S., Colless, M., Brough, S., Hopkins, A.M., Bauer, A.E., Birchall, M.N., Ellis, S., Horton, A., Leon-Saval, S., Lewis, G., López-Sánchez, Á.R., Min, S.S., Trinh, C., Trowland, H., 2012. The Sydney-AAO Multi-object Integral field spectrograph. MNRAS 421, 872–893. doi:10.1111/j.1365-2966.2011.20365.x, arXiv:1112.3367.

Davis, M., Huchra, J., Latham, D.W., Tonry, J., 1982. A survey of galaxy redshifts. II - The large scale space distribution. ApJL 253, 423. URL: http://dx.doi.org/10.1086/159646, doi:10.1086/159646.

Dawson, K.S., Schlegel, D.J., et al., C.P.A., 2013. THE BARYON OSCILLATION SPECTROSCOPIC SURVEY OF SDSS-III. The Astronomical Journal 145, 10. URL: http://dx.doi.org/10.1088/0004-6256/145/1/10, doi:10.1088/0004-6256/145/1/10.

Driver, S.P., Hill, D.T., Kelvin, L.S., Robotham, A.S.G., Liske, J., Norberg, P., Baldry, I.K., Bamford, S.P., Hopkins, A.M., Loveday, J., Peacock, J.A., Andrae, E., Bland-Hawthorn, J., Brough, S., Brown, M.J.I., Cameron, E., Ching, J.H.Y., Colless, M., Conselice, C.J., Croom, S.M., Cross, N.J.G., de Propris, R., Dye, S., Drinkwater, M.J., Ellis, S., Graham, A.W., Grootes, M.W., Gunawardhana, M., Jones, D.H., van Kampen, E., Maraston, C., Nichol, R.C., Parkinson, H.R., Phillipps, S., Pimbblet, K., Popescu, C.C., Prescott, M., Roseboom, I.G., Sadler, E.M., Sansom, A.E., Sharp, R.G., Smith, D.J.B., Taylor, E., Thomas, D., Tuffs, R.J., Wijesinghe, D., Dunne, L., Frenk, C.S., Jarvis, M.J., Madore, B.F., Meyer, M.J., Seibert, M., Staveley-Smith, L., Sutherland, W.J., Warren, S.J., 2011. Galaxy and Mass Assembly (GAMA): survey diagnostics and core data release. MNRAS 413, 971–995. doi:10.1111/j.1365-2966.2010.18188.x, arXiv:1009.0614.

Haerder, T., Reuter, A., 1983. Principles of transaction-oriented database recovery. ACM Comput. Surv. 15, 287–317. URL: http://doi.acm.org/10.1145/289.291, doi:10.1145/289.291.

Heber, G., Folk, M., Koziol, Q.A., 2014. Everything that HDF Users have Always Wanted to Know about Hadoop... But Were Ashamed to Ask. URL: http://www.hdfgroup.org/pubs/papers/Big_HDF_FAQs.pdf.

Juric, M., 2011. Large Survey Database: A Distributed Framework for Storage and Analysis of Large Datasets, in: American Astronomical Society Meeting Abstracts #217, p. 433.19.

Juric, M., 2012. LSD: Large Survey Database framework. Astrophysics Source Code Library. arXiv:1209.003.

Konstantopoulos, I.S., 2014. The Starfish Diagram: Statistical visualization tool. Astrophysics Source Code Library. arXiv:1407.001.

Konstantopoulos, I.S., 2015. The starfish diagram: Visualising data within the context of survey samples. Astronomy and Computing 10, 116–120. doi:10.1016/j.ascom.2015.01.007, arXiv:1407.5619.

Lawrence, J.S., Bland-Hawthorn, J., Brown, D., Bryant, J.J., Cecil, G., Content, R., Croom, S., Gers, L., Gillingham, P.R., Richards, S., Saunders, W., Staszak, N., 2014. Towards a spectroscopic survey of one hundred thousand spatially resolved galaxies with Hector, in: Society of Photo-Optical Instrumentation Engineers (SPIE) Conference Series, p. 6. doi:10.1117/12.2055734.

Mink, J., Mann, R.G., Hanisch, R., Rots, A., Seaman, R., Jenness, T., Thomas, B., O'Mullane, W., 2014. The Past, Present and Future of Astronomical Data Formats. ArXiv e-prints arXiv:1411.0996.

Noeske, K.G., Weiner, B.J., Faber, S.M., Papovich, C., Koo, D.C., Somerville, R.S., Bundy, K., Conselice, C.J., Newman, J.A., Schiminovich, D., Le Floc'h, E., Coil, A.L., Rieke, G.H., Lotz, J.M., Primack, J.R., Barmby, P., Cooper, M.C., Davis, M., Ellis, R.S., Fazio, G.G., Guhathakurta, P., Huang, J., Kassin, S.A., Martin, D.C., Phillips, A.C., Rich, R.M., Small, T.A., Willmer, C.N.A., Wilson, G., 2007. Star Formation in AEGIS Field Galaxies since z=1.1: The Dominance of Gradually Declining Star Formation, and the Main Sequence of Star-forming Galaxies. ApJL 660, L43–L46. doi:10.1086/517926, arXiv:astro-ph/0701924.

Pence, W.D., Chiappetti, L., Page, C.G., Shaw, R.A., Stobie, E., 2010. Definition of the Flexible Image Transport System (FITS), version 3.0. A&A 524, A42. doi:10.1051/0004-6361/201015362.

Pourmal, E., 2002. ?FITSIO, HDF4, NetCDF, PDB and HDF5 Performance Some Benchmarks Results. URL: http://www.hdfgroup.org/HDF5/RD100-2002/HDF5_Performance.pdf.

Sadalage, P., 2014. NoSQL Databases: An Overview. URL: http://www.thoughtworks.com/insights/blog/nosql-databases-overview.

Sánchez, S.F., Kennicutt, R.C., de Paz et al., A.G., 2012. CALIFA the Calar Alto Legacy Integral Field Area survey. Astronomy & Astrophysics 538, A8. URL: http://dx.doi.org/10.1051/0004-6361/201117353, doi:10.1051/0004-6361/201117353.

Saunders, W., Bridges, T., Gillingham, P., Haynes, R., Smith, G.A., Whittard, J.D., Churilov, V., Lankshear, A., Croom, S., Jones, D., Boshuizen, C., 2004. AAOmega: a scientific and optical overview, in: Moorwood, A.F.M., Iye, M. (Eds.), Ground-based Instrumentation for Astronomy, pp. 389–400. doi:10.1117/12.550871.

Sharp, R., Allen, J.T., Fogarty, L.M.R., Croom, S.M., Cortese, L., Green, A.W., Nielsen, J., Richards, S.N., Scott, N., Taylor, E.N., Barnes, L.A., Bauer, A.E., Birchall, M., Bland-Hawthorn, J., Bloom, J.V., Brough, S., Bryant, J.J., Cecil, G.N., Colless, M., Couch, W.J., Drinkwater, M.J., Driver, S., Foster, C., Goodwin, M., Gunawardhana, M.L.P., Ho, I.T., Hampton, E.J., Hopkins, A.M., Jones, H., Konstantopoulos, I.S., Lawrence, J.S., Leslie, S.K., Lewis, G.F., Liske, J., López-Sánchez, Á.R., Lorente,

N.P.F., McElroy, R., Medling, A.M., Mahajan, S., Mould, J., Parker, Q., Pracy, M.B., Obreschkow, D., Owers, M.S., Schaefer, A.L., Sweet, S.M., Thomas, A.D., Tonini, C., Walcher, C.J., 2015. The SAMI Galaxy Survey: cubism and covariance, putting round pegs into square holes. MNRAS 446, 1551–1566. doi:10.1093/mnras/stu2055, arXiv:1407.5237.

Thomas, B., Jenness, T., Economou, F., Greenfield, P., Hirst, P., Berry, D.S., Bray, E., Gray, N., Muna, D., Turner, J., de Val-Borro, M., Santander-Vela, J., Shupe, D., Good, J., Berriman, G.B., Kitaeff, S., Fay, J., Laurino, O., Alexov, A., Landry, W., Masters, J., Brazier, A., Schaaf, R., Edwards, K., Redman, R.O., Marsh, T.R., Streicher, O., Norris, P., Pascual, S., Davie, M., Droettboom, M., Robitaille, T., Campana, R., Hagen, A., Hartogh, P., Klaes, D., Craig, M.W., Homeier, D., 2015. Learning from FITS: Limitations in use in modern astronomical research. ArXiv e-prints arXiv:1502.00996.

Wells, D.C., Greisen, E.W., Harten, R.H., 1981. FITS - a Flexible Image Transport System. A&A Supplement 44, 363.

York, D.G., Adelman, J., Anderson, Jr., J.E., Anderson, S.F., Annis, J., Bahcall, N.A., Bakken, J.A., Barkhouser, R., Bastian, S., Berman, E., Boroski, W.N., Bracker, S., Briegel, C., Briggs, J.W., Brinkmann, J., Brunner, R., Burles, S., Carey, L., Carr, M.A., Castander, F.J., Chen, B., Colestock, P.L., Connolly, A.J., Crocker, J.H., Csabai, I., Czarapata, P.C., Davis, J.E., Doi, M., Dombeck, T., Eisenstein, D., Ellman, N., Elms, B.R., Evans, M.L., Fan, X., Federwitz, G.R., Fiscelli, L., Friedman, S., Frieman, J.A., Fukugita, M., Gillespie, B., Gunn, J.E., Gurbani, V.K., de Haas, E., Haldeman, M., Harris, F.H., Hayes, J., Heckman, T.M., Hennessy, G.S., Hindsley, R.B., Holm, S., Holmgren, D.J., Huang, C., Hull, C., Husby, D., Ichikawa, S., Ichikawa, T., Ivezić, Ž., Kent, S., Kim, R.S.J., Kinney, E., Klaene, M., Kleinman, A.N., Kleinman, S., Knapp, G.R., Korienek, J., Kron, R.G., Kunszt, P.Z., Lamb, D.Q., Lee, B., Leger, R.F., Limmongkol, S., Lindenmeyer, C., Long, D.C., Loomis, C., Loveday, J., Lucinio, R., Lupton, R.H., MacKinnon, B., Mannery, E.J., Mantsch, P.M., Margon, B., McGehee, P., McKay, T.A., Meiksin, A., Merelli, A., Monet, D.G., Munn, J.A., Narayanan, V.K., Nash, T., Neilsen, E., Neswold, R., Newberg, H.J., Nichol, R.C., Nicinski, T., Nonino, M., Okada, N., Okamura, S., Ostriker, J.P., Owen, R., Pauls, A.G., Peoples, J., Peterson, R.L., Petravick, D., Pier, J.R., Pope, A., Pordes, R., Prosapio, A., Rechenmacher, R., Quinn, T.R., Richards, G.T., Richmond, M.W., Rivetta, C.H., Rockosi, C.M., Ruthmansdorfer, K., Sandford, D., Schlegel, D.J., Schneider, D.P., Sekiguchi, M., Sergey, G., Shimasaku, K., Siegmund, W.A., Smee, S., Smith, J.A., Snedden, S., Stone, R., Stoughton, C., Strauss, M.A., Stubbs, C., SubbaRao, M., Szalay, A.S., Szapudi, I., Szokoly, G.P., Thakar, A.R., Tremonti, C., Tucker, D.L., Uomoto, A., Vanden Berk, D., Vogeley, M.S., Waddell, P., Wang, S., Watanabe, M., Weinberg, D.H., Yanny, B., Yasuda, N., 2000. The Sloan Digital Sky Survey: Technical Summary. AJ 120, 1579–1587. doi:10.1086/301513, arXiv:arXiv:astro-ph/0006396.