Scientific Visualisation of Extremely Large Distributed Astronomical Surveys

Singh, S.; Valentijn, E. A.; Belikov, A. N.; Buddelmeijer, H.; Roerdink, J. B. T. M.

Published in:
Astronomical Data Analysis Software and Systems XXIX

Publication date:
2020

Astronomical Data Analysis Software and Systems XXIX
ASP Conference Series, Vol. 527
R. Pizzo, E.R. Deul, J-D. Mol, J. de Plaa and H. Verkouter, eds.

Scientific Visualisation of Extremely Large Distributed Astronomical Surveys

S. Singh,[1] E. A. Valentijn,[2] A. N. Belikov,[2] H. Buddelmeijer[2] and J. B. T. M. Roerdink[1]

[1]*Bernoulli Institute for Mathematics, Computer Science and Artificial Intelligence, University of Groningen, the Netherlands;* rnsweta@gmail.com

[2]*Kapteyn Astronomical Institute, University of Groningen, the Netherlands;*

Abstract. Interactive real-time visualisation of large data sets plays an important role in scientific research. It is even more relevant for astronomy where new cutting edge large telescopes will generate tens of petabytes sky surveys. We describe our solution, developed in context of the Euclid space mission whose large astronomical imaging data will be distributed over several heterogeneous Science Data Centres (SDCs) across the world. In our visualisation architecture for distributed data, millions of survey images (HiPS) distributed over SDCs are efficiently transported and combined to deliver image(s) of interest at the desired resolution (up to pixel level) to the user. This is achieved by optimally utilising a combination of several modern tools consisting of *http* servers, a Front-End Node and load-balancer (FEN), reverse proxies, PHP/Python scripts, MySQL databases, including on the fly image generation/combination which all feed (only) the required information to the Aladin interactive visualisation tool at the remote user's Personal Computer (PC). It has potential applications for large projects (e.g., Square Kilometre Array) having data distributed across several locations.

1. Introduction

Euclid (http://sci.esa.int/euclid) is a space mission led by Euclid Consortium and the European Space Agency with the main scientific objective of carrying out visible and near infrared surveys of the entire extra-galactic sky to investigate the origin of the Universe's accelerating expansion and the nature of dark energy, dark matter and gravity. Due to its unprecedented accuracy similar to that of the Hubble space telescope, Euclid data will be useful for most astronomical studies. The Euclid consortium (https://www.euclid-ec.org) has organised the scientific and technical community into a number of working groups for efficient workload distribution. The resulting data volume comprising several tens of PB (Dubath et al. 2017) will be distributed over several SDCs (in Europe, USA) responsible for data storage and processing.

One of the main outputs at each of these SDCs will be high-quality astronomical survey images (including from related ground based telescopes and simulations) and source catalogues. The traditional approach involving local copies at user PCs or large compute clusters, or even at one physical location may be difficult due to large data volumes and costs involved. Thus distributed, real-time interactive visualisation of the survey is critical to gain needed insight to maximise scientific astronomical research.

2. Distributed Visualisation Pipeline

The two ends of any visualisation pipeline framework are (i) *The Demand (User) Side* : – the visualisation software at the users end (client) and (ii) *The Supply Side* : – a cohesive data storage, management and transport framework able to provide the required data/information in real time in an efficient and transparent manner.

For Euclid, Aladin sky atlas (`https://aladin.u-strasbg.fr`) will be the main visualisation analysis tool on the user (client) side (e.g. an astronomer). The default format for Euclid surveys will be based on Hierarchical Progressive Surveys (HiPS). HiPS makes use of the hierarchical features of HEALPix (Górski et al. 2005) to organise astronomical data into HEALPix maps of different resolutions (Fernique et al. 2015).

We followed a two-pronged approach, firstly to develop a visualisation framework for distributed data to address the challenges at the supply side of the pipeline. Secondly we worked on ways to minimise the size of survey data itself. In this paper we present only the work related to the visualisation framework for distributed data.

3. Architectural Design

The *http* compliant nature of HiPS (`http://www.ivoa.net/documents/HiPS`) allows it to be accessed via any *http* compliant server. We make use of this useful feature to arrive at a client-server based architectural design which binds the distributed resources across the SDCs into the visualisation framework. This flexible architecture also empowers us to utilise the *http* protocols and associated tools. As an example, while the HiPS does prescribe the storage of the HiPS image (survey) in a simple hierarchy of directory and files, the *http* compliance allows the actual storage behind the *http* server to be implemented in any desired manner, provided the critical requirement of supplying the requested information to the client (e.g., Aladin) is correctly met.

Our framework is based on the generic software LAMP stack model founded on four open source software components, namely **L**inux, **A**pache, **M**ySQL (and alternatives) and **P**HP (and alternatives). Figure 1(a) shows the three main constituents in the framework, namely : (i) Astronomers using Aladin at their PCs to remotely visualise Euclid surveys. (ii) A FEN acts as the main interface for Aladin requests and hosts the mandatory and recommended (optional) files necessary to initiate survey visualisation (e.g., properties, index.html, moc.fits etc.). It has a Linux operating system (e.g., Ubuntu) with a listening Apache *http* server, a MySQL server/database and PHP/Python scripts. The database contains the necessary entries to locate each survey file constituting the entire survey (across all SDCs). The PHP/Python scripts dynamically respond with appropriate action to the incoming *http* requests. (iii) The SDCs also have a listening Apache *http* server and collectively host the actual survey images.

The visualisation framework can be set up in two modes namely (i) A fully distributed visualisation mode and (ii) A reverse proxy mode. Figure 1(b) shows a simplified flowchart for the fully distributed mode setup. An astronomer initiates a survey visualisation using Aladin from the PC. Aladin makes a *http* request for the desired information to the *http* server on the FEN. There could be (mainly) three possible scenarios : (a) The request is for one of the basic mandatory (or recommended) files (hosted at FEN) (b) The requested file is unique and present at one or more SDCs and (c) The requested file is non-unique and present at one or several SDCs. Here the term ‘unique’ refers to files which have only identical copies at one or more SDCs. The term ‘non-

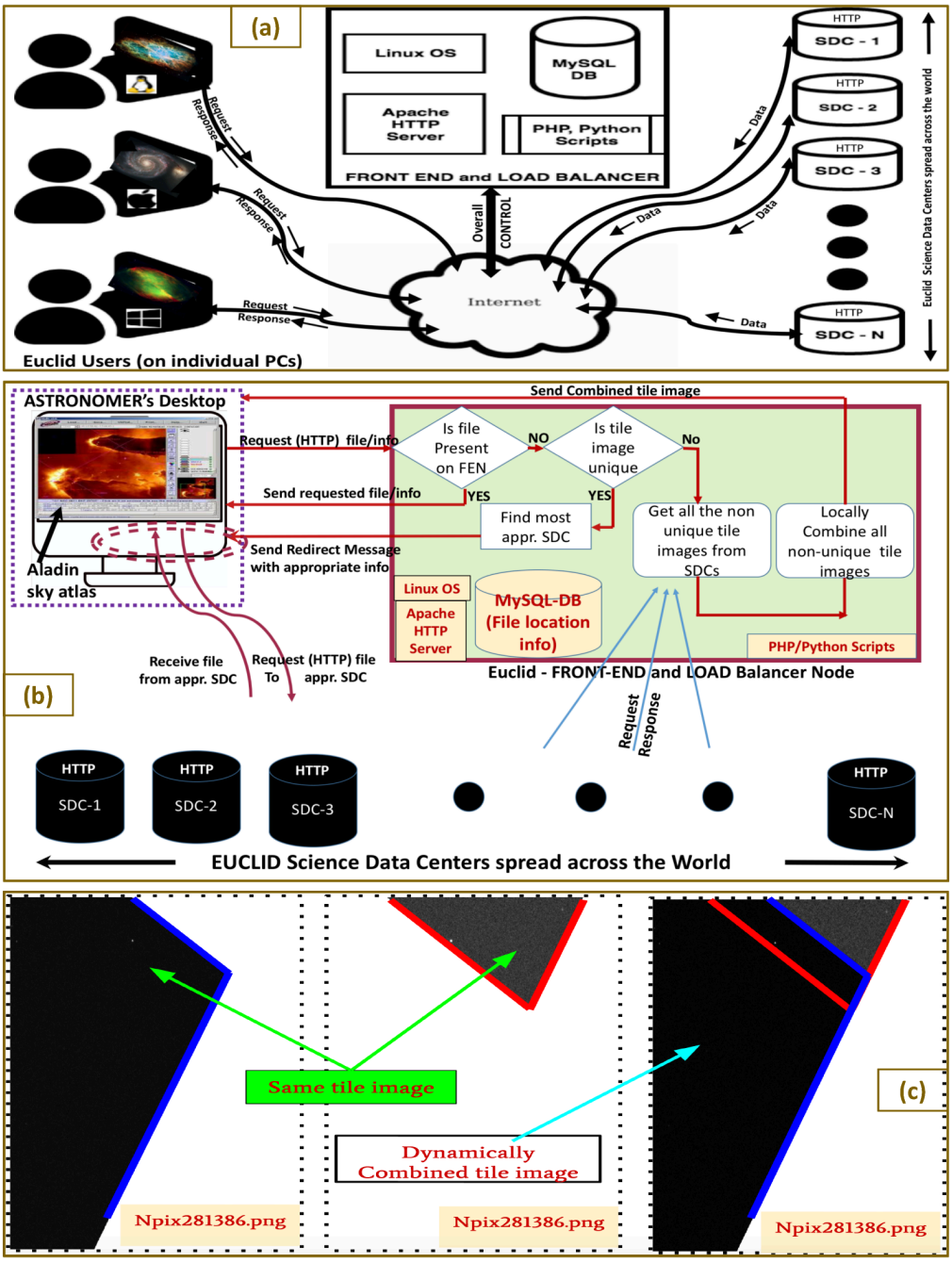

Figure 1. (a) An overall schematic of visualisation framework for distributed data. Using Aladin, astronomers visualise their regions of interest (Euclid surveys) at their individual PCs. The FEN is the main interface with overall control. (b) A simplified flowchart depicting how Aladin's *http* requests are handled in the fully distributed visualisation mode setup (c) An example of two same tile images but with different partial sky coverage which are dynamically combined during visualisation.

unique' refers to a file present at multiple SDCs some of which contain non-identical information (e.g., the same tile image but with different partial coverage).

In case (a) the *http* server at the FEN responds with the requested file. In case (b) it responds with a *redirect message* to Aladin with the details of the most appropriate SDC to which the request must be re-addressed. Aladin re-requests to the appropriate *http* server which responds with the desired file. In case (c) a PHP script is instantiated which requests and receives all the non-unique instances of the requested file from the SDCs, combines them locally and responds with the resulting combined file to Aladin as shown in Figure 1(c). The fully distributed mode setup is network efficient as most of the data transfer to the clients happens directly from the SDCs hosting it. The reverse proxy mode setup is convenient for overall monitoring and system security as clients communicate only with the FEN and the SDCs are shielded from direct access.

4. Conclusions and Future work

We have successfully developed a distributed visualisation framework with the aim of enabling users to visualise all sky HiPS surveys as if they were stored locally on their individual PCs. It places no constraints on the inherent capabilities of Aladin and on the SDCs including their heterogeneity in terms of their size, system design, operating system, file system and intra-SDC bandwidth. An user on his PC screen can visualise either a small sky area at high angular resolution or a large sky area at low angular resolution given the limited number of screen pixels. Thus the visualisation framework has to transfer only a limited amount of data which can be visualised at a given time. Every *http* request from all the users is handled simultaneously (in parallel), and independently (including the data transfer) due to stateless nature of the http protocol.

Our framework can be used for visualisation of distributed data in general and will be also applied for Euclid mission. The present work has been carried out using *Virtual Machines* (`https://www.oracle.com/virtualization`) as SDCs and the PC as the FEN. We are now implementing the visualisation framework on OmegaCEN infrastructure (`http://www.astro.rug.nl/~omegacen`) for the 4^{th} data release of Kilo-Degree Survey (Kuijken et al. 2019). The documentation including the installation instructions and software code will be available for open access.

We expect to make further improvements by exploring options which include but are not limited to: NoSQL distributed databases like Apache Cassandra, NGINX *http* server on SDCs, key-value stores, replacing PHP scripts by daemon processes, caching, ELK stack (`https://www.elastic.co`) with Kibana as Web-UI for integrated performance monitoring. The International Virtual Observatory Alliance will play a critical role for future big international projects where single entities cannot efficiently tackle the big data volumes. In that context, we expect our work to play a useful role.

References

Dubath, P., Apostolakos, N., Bonchi, A., Belikov, A., et al. 2017, in Astroinformatics, edited by M. Brescia, S. G. Djorgovski, E. D. Feigelson, G. Longo, & S. Cavuoti, vol. 325 of IAU Symposium, 73. 1701.08158

Fernique, P., Allen, M. G., et al. 2015, A&A, 578, A114. 1505.02291

Górski, K. M., Hivon, E., et al. 2005, ApJ, 622, 759. astro-ph/0409513

Kuijken, K., Heymans, C., et al. 2019, A&A, 625, A2. 1902.11265