A Private Cloud-based Architecture for the Brazilian Weather and Climate Virtual Observatory

Rafael Duarte Coelho dos Santos, Luiz Alberto R. Correa, Eduardo Martins Guerra, and Nandamudi L. Vijaykumar

Brazilian National Institute for Space Research, São José dos Campos - SP, CEP 12227-010, Brazil,
rafael.santos@inpe.br

Abstract. Brazil has a significant amount on a wide range of data about weather and climate, collected from sensors or calculated by numerical models, and are very important for historical reasons for the understanding of climate change and prediction of extreme weather events. This data represent different physical measures, have different temporal and spatial scales and is stored in different formats; there is no unified way to discover which data is available and under which conditions it can be used.
In this paper we describe the architecture of the Brazilian Weather and Climate Virtual Observatory, a set of software tools that works as a Virtual Observatory (VO) to allow weather and data metadata discovery and data access and processing. The VO will be partially deployed in a private cloud; reasons and benefits of doing so will also be explained.

Keywords: Virtual Observatories, Private Cloud, Metadata, Distributed Data Processing

1 Introduction

Brazil now has a significant amount of information on a wide range of climate- and weather-related measured variables and predictions. There is a growing understanding of the expected impacts of climate change and extreme weather phenomena and a strong interest on better understanding it. Strategies to face the effects of climate-related phenomena are important for all the government sectors: national, state and local, for commercial enterprises (tourism, farms and agricultural cooperatives, etc.) and for the community in general.

It is a fact that relevant data and information on climate and weather belong to several organizations that deal with different sectors of expertise. Each organization has its own agenda of policies to operate and therefore such data and information are stored in files and databases with formats and resolutions (time and space) that most suit its needs and applications, often with different public or private access policies and interfaces. That is, data and information are

distributed and diverse in terms of completeness, formats, quality, etc., sometime with incomplete or unavailable metadata, making it hard to know which data with a specific time and/or spatial coverage is even available for queries. Therefore, it is more than natural that decision-makers, researchers, students and common citizens face difficulties to access such data. The difficulty is increased when there is need to use multiple data sources or combine data with different time and spatial scales, moreover when these data products must be kept up-to-date; and sometimes the most difficult task is to discover which type of data is available for a specific need.

Solutions to facilitate planning, policy-making, decision taking related to this kind of data must be made available. In order to achieve this, it is necessary to figure how to make available and accessible complete, reliable, good quality and easy to use information on climate and weather related data, without the need to reformulate the already existing systems, databases and interfaces and allowing the discovery of existing datasets and data processing operations. Along with the data itself, online applications that process this data (e.g. for classification, regression, summarization, prediction, visualization, etc.) could also be catalogued so users could combine applications and datasets to create their own workflows for weather and climate data analysis.

One approach to solve similar data dissemination and utilization problems was proposed by the astronomy and astrophysics community more than ten years ago: Virtual Observatories (VOs). Virtual Observatories are frameworks that use information technology (IT) to organize, maintain and explore information on large, distributed and dynamic datasets [1, 2]. Within this framework it is possible to catalog data; process, visualize or cross-correlate it with tools both on the desktop and on the web; generate new data collections and create and use workflows and processing pipelines to automate new analysis and discoveries. The concept of Virtual Observatories is also being used in other science fields, such as Earth Sciences [3], Solar-terrestrial Physics [4], Environmental Sciences [5] and even Computer Science and IT itself [6]. Virtual Observatories are a natural extension of the paradigm of centralized middleware proposed to allow access to data and tool for specific domains [7, 8], but allowing the inclusion of external data and tools, therefore increasing their usage and possibilities.

Figure 1 illustrates the outline of a Virtual Observatory. Users have access to portals that provide data catalogues, the data itself and processing tools, implemented as Web Services to ensure portability and flexibility. Users may also contribute with data and processing resources to the VO. The primary roles of a VO are to facilitate data discovery (through the portals or a registry of all catalogued data and services), data access (through web services or other methods, also allowing the use of local data) and data federation (combination of data from different sources) [1].

In this paper we describe the architecture of the Brazilian Weather and Climate Virtual Observatory, a set of software tools and data access tools that will enable users in different levels and with different skills to discover data, do basic analysis and visualization, using uniform data access protocols. This Virtual

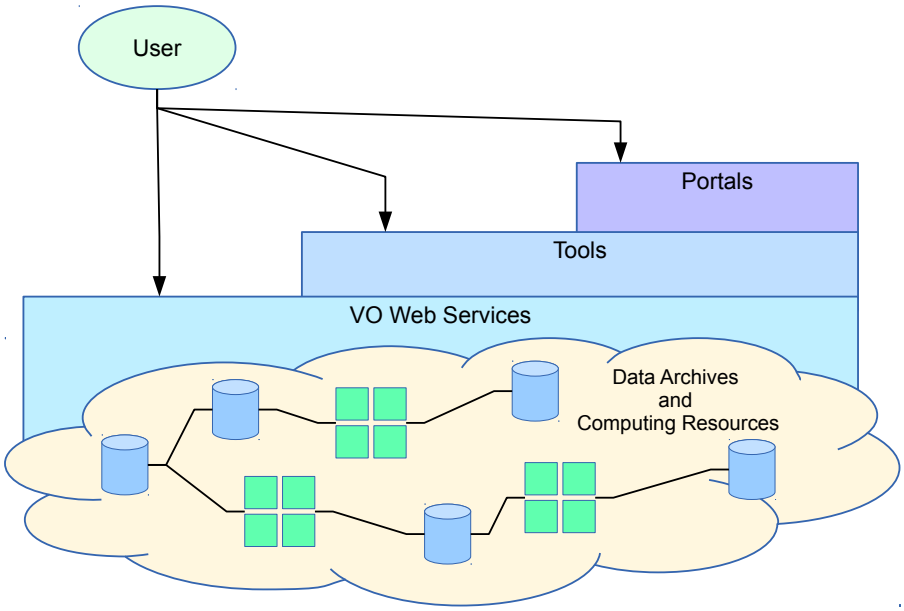

Fig. 1. Conceptual outline of a VO (adapted from [1])

Observatory will also enable users to include their own datasets in a catalog so other users can also access and use it.

This paper is organized as follows: Section 2 presents the general architecture for the Virtual Observatory, detailing its software components and the role of a private cloud for deploying the core functions of the Virtual Observatory. Section 3 comments on the present status of development and deployment of the VO and also on the future steps for the project and Section 4 present our conclusions.

2 The Weather and Climate VO Architecture

2.1 Introduction

In this section we describe the Brazilian Weather and Climate VO Architecture, in particular, its software components, their functions and how those components are integrated. We also explain the reasons to deploy the core part of the VO in a private cloud, and the expected problems and benefits of it. Finally, we describe a middleware that allows the execution of user-defined code inside the private cloud, effectively bringing the code "close" to the data (in the sense of reducing the need for sending the data through the network for processing), making possible the execution of algorithms that use large amounts of the data in an efficient way.

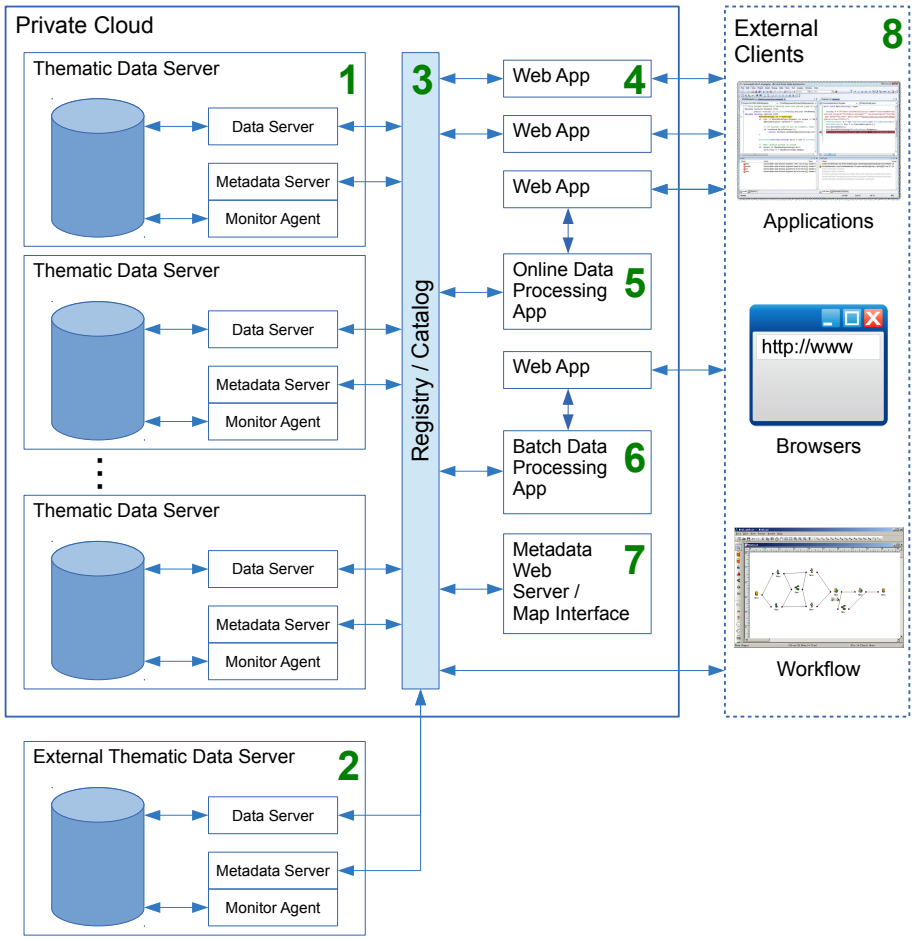

Fig. 2. General architecture of the VO

2.2 Architecture of the VO

Figure 2 presents the general architecture of the Virtual Observatory. The software components of the VO are grouped into three categories: private-cloud based server databases and applications (the VO core), external data server and external clients. Each major software component of the VO is labeled and will be commented in this section.

1: A **Thematic Data Server** contains data about a specific data collection or generation effort, usually with data related to a specific theme (e.g. rainfall, temperature, atmospheric electrical discharges, wind speed and direction, rainfall calculated from numerical models, etc.). The data on the server can also be spatially or temporally limited, e.g. there are databases that cover

historic rainfall records for a particular state or only for a specific sensor which operated in a period of time.
The thematic data servers are self-contained systems composed of a database server with an associated data server that allows access to its data (optionally with constraints to avoid improper use). It is important to point out that under the VO architecture the databases are exposed to client applications only through their associated web services.
The thematic data servers also contain other two important software components: a Monitor Agent and a Metadata Server. The Monitor Agent regularly queries the database to extract metadata about it. In the VO context, metadata is information about the weather or climate data stored in the database, particularly, information about the data coverage (the spatial or temporal extent of the data on that database). This information is relayed to users and to the registry so users and applications can get information about the data before querying it.

2: **External Thematic Data Servers** are hosted outside of the private cloud, which is the expected scenario for collaborators who have their own databases already operational, possibly in other locations. In order to be able to share information with the VO these data servers must implement the Monitor Agent and Metadata Server, but this can be done without any real changes to the database itself, making possible the linking of legacy databases to the VO.

3: The **Registry** or Catalog is the most important component on the Weather and Climate Virtual Observatory. It is a central repository of information about data, metadata and tools for the VO, and can be searched by geographic coordinates, time intervals, data type, provider, keywords and combinations of those; returning a set of resources (usually web services) that can be used to get the data itself. Interaction with the registry can be done via a web interface or via web services, so they can be used directly by other applications.
The registry will be feed metadata from each thematic data servers' monitor agents through their metadata servers, ensuring that at any time the final user or application can find what data is available and its restrictions.

4: **Web Applications** are interfaces to the VO registry and to the databases associated to the VO. These applications are implemented as simple web services that perform queries on the databases or registry and return the results to the client applications. These web applications can also compose results from databases or other applications; distribute and aggregate queries, execute specific algorithms. The main difference between the data servers that are part of the thematic data servers and the web applications is that the former are designed to allow access to chunks of data with as little processing as possible (e.g. extrema and averages on time series), while the latter are applications that may be able to answer more complex queries that may involve more complex algorithms.

5: **Online data processing applications** are applications specifically designed to query and process one or more data servers and/or the registry

to create results in almost real-time (i.e. being able to use the most recent records in the database). These applications, by their nature, must not be data- or CPU-intensive. These applications will interface with external users and applications through specific web applications that will work as portals to the data processing applications, i.e. interfaces that are able to call the applications, passing parameters and returning results to the final user.
Some possible examples of these applications are static visualization tools, for example, tools that overlay data points over a map.

6: **Batch data processing applications** are also applications that will access data on the data servers and registry, process this data and return it to the final users, through specific web applications that control the execution of the data processing applications. As the name of this software component points out, execution of applications will be done in batch, therefore it is possible to run more data-intensive and/or CPU-intensive algorithms but without guarantee that the results will be delivered in real time.
Since applications developed with this model will be executed in batch the web applications that control it must implement basic batch processing techniques: implementation of priority queues, running and monitoring processes, batch communication of results to users, basic authentication mechanisms, etc.
One important aspect of batch data processing applications is that they will use a framework that allows the execution of user-defined code in a sandbox. This will be described in subsection 2.4.

7: A **Metadata Web Server / Map Interface** that is a specific web application that allows the visual discovery of the available data. This application will present two ways to discover data: one by web services, proper for interaction with other applications, that will be able to list all data sources corresponding to specific constraints (e.g. data type, time and spatial limits, data quality/completeness, etc.). This application also will present to the users a visual interface, with the results for sources overlaid in a map, similar to SciScope (www.sciscope.org). With this application users will be able to quickly locate regions in space and time which contains the data of interest.

8: **External clients** that use the metadata and data available through the VO. We expect to have several types of clients of the VO data and metadata, such as applications developed by expert users that access the data and metadata through web services, simple clients like browsers and workflow management systems, that are able to visually compose the web services available to answer specific questions.

2.3 A Private Cloud for the VO

As described in Figure 2 the core functionality of the VO (some of its data servers, the registry, some applications) will be deployed in a private cloud, i.e. a cloud computing environment deployed in and operated by a single company or institution.

One could expect that the deployment of VO tools in a private cloud is a contradiction of the open, distributed nature of resources a VO is supposed to provide. There are several reasons for deploying the core of the VO infrastructure in a private cloud, which, in our opinion, more than justify its adoption:

- The web applications and the registry will be hosted in the same physical environment: hardware that makes part of the private cloud are connected through a high-performance internal network, ensuring fast access to the data servers.
- Thematic data servers may have lots of features in common; templates for the virtual servers can be created in order to facilitate the deployment of new data servers.
- The generic advantages of the cloud (sharing of resources, expected reduced costs, quick and easy deployment of servers, tools to increase the pool of resources, etc.) also apply.
- Monitoring the performance of the virtual data servers and web applications could give interesting insights about usage, which could lead, for example, to changes in the resources allocated to the servers. Since all those servers are hosted in a private cloud, the cloud manager itself could give information on the performance and loads on the servers.

Additionally it must be pointed that deploying the VO core infrastructure in a private cloud does not prevent or hinder the deployment of other tools or external data servers outside the cloud.

2.4 Running User-Defined Code on the VO Servers

Let's consider a simple use case of the VO: calculating simple statistics on a set of data with some constraints. For example, one could want to discover the largest difference between consecutive monthly averages of temperatures (i.e. greatest variation in consecutive months), constrained or not to a geographic region. Implementation of this algorithm is straightforward, and it can be made simpler if there are already web services to provide monthly temperature averages from some weather stations.

Expert users could implement this algorithm and query the appropriate web services to get all the data they need, but that would imply in running several web services and transferring their results over the Internet to the client application. It would be more convenient to prototype the algorithm in a small subset of the data, then transfer the algorithm implementation to the Batch Data Processing App (shown in Figure 2) so it would run "close to the data" improving its efficiency.

In order to achieve this we propose a middleware that will be one of the components of the VO and that allows the execution of user-defined code in a sandbox. By using this feature, a user will be able to develop his own processing algorithm and submit it to be executed in the server. The code submitted by the user should be described by custom metadata that will be used by the middleware

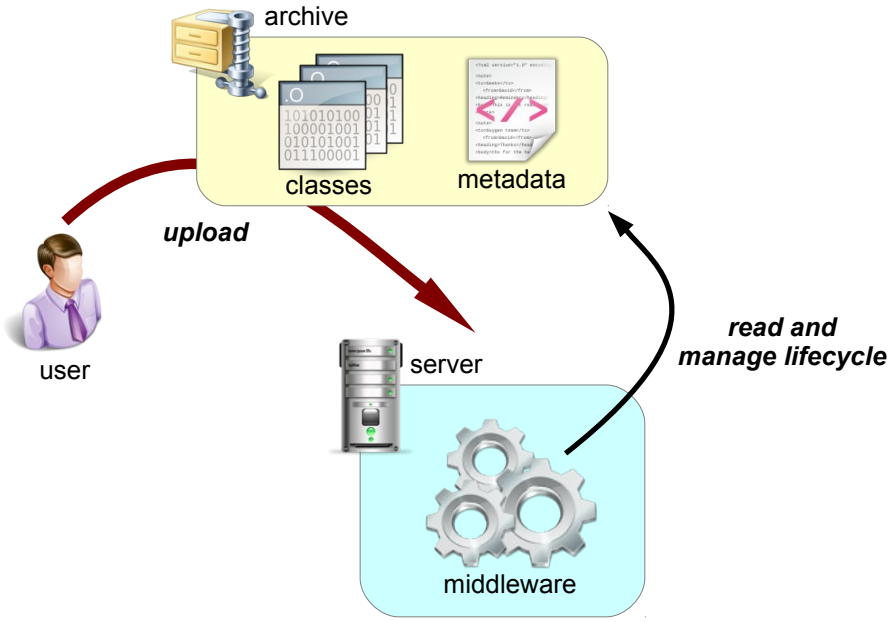

Fig. 3. Submission of user-defined code for execution on the server

to provide the right services to its classes [9]. The submission will be composed by an archive with classes and metadata descriptors as presented in Figure 3.

The code submitted to the server is able to define parameters to be received in each execution request as input. As output the class should have a result and optionally log entries. The class attributes used to store the processing inputs and outputs will be defined using metadata and can be used to dynamically generate graphical or programmable interfaces to a request submission. For a safe execution, this class will have restricted access to resources in the server. The middleware will be responsible to inject into the class the services necessary for its execution [10]. The instance injected is encapsulated with a proxy that is responsible to monitor the service access. The services retrieved by the middleware and inserted in the user class are also determined by metadata. Figure 4 presents a representation of this architecture.

The execution life cycle will be managed by the middleware. It should be able to ensure restrictions in the execution of external classes, being allowed to interrupt the execution if necessary. A configurable policy to limit the execution time or the number of services requests can be used to avoid the consumption of services resources by only one process. The metadata-based API used to define the classes allows a flexible definition of the services provided by the server [11]. This kind of solution also increases the decoupling between the framework and the submitted class, allowing each one of them to evolve independently. Consequently, a code submitted in an earlier version of the middleware will work on

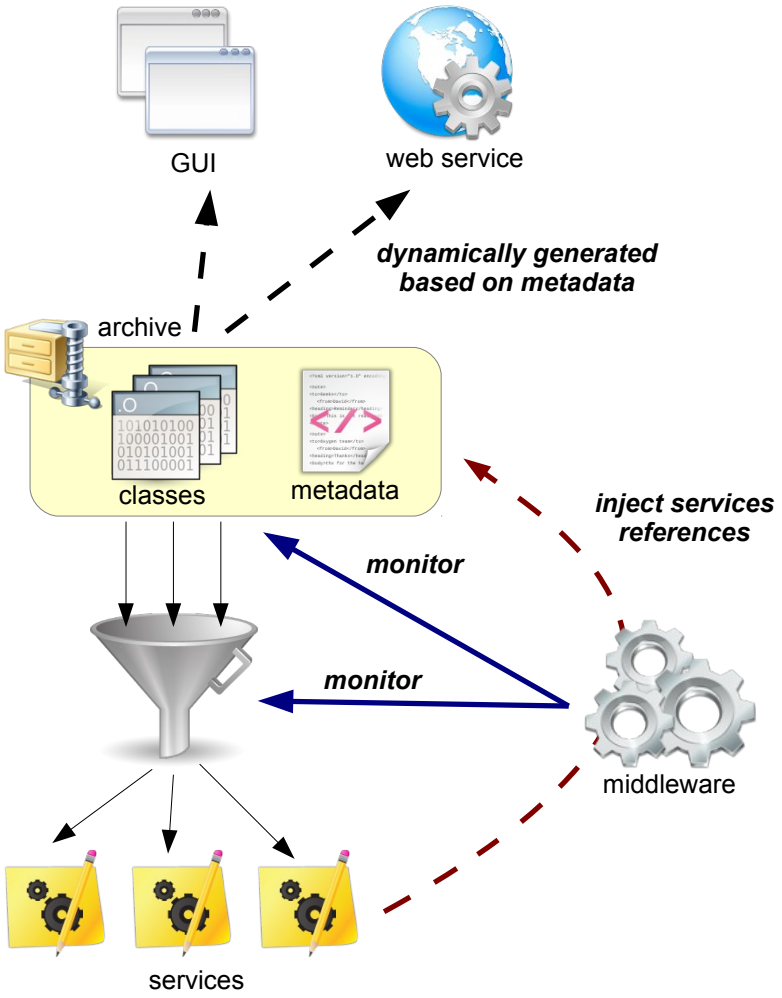

Fig. 4. Interaction between middleware and the submitted archive

new versions, even if it evolve the existing metadata schema. The motivation of this architecture is to allow users to available new executions on the server. The middleware is used to ensure that the process will have access to the services that it needs and that its execution will not harm other processes spending excessive computational resources. As a result, it will provide a safe environment, in which the algorithms execution is performed close to the data, consequently avoiding excessive network invocations and having a better performance.

The middleware implementation would allow the execution of specific, user-defined algorithms with a more efficient access to the data. This would allow the

implementation and deployment of data-intensive algorithms such as the used in data mining and static visualization applications [12].

3 Present Status

The Brazilian Weather and Climate Virtual Observatory is an ongoing research and development project – working results are, so far, not visible to the public. We already have three thematic data servers (one with precipitation data, one with atmospheric discharges and one with ground temperature). The next steps for those thematic data servers will be the implementation of the monitor agent and metadata server. After that we will able to deploy the registry and start publishing data and services, and after that the web applications. We expect to be able to get a first working implementation of the basic services and tools shown in Figure 2 by the end of 2013 – this step is called "first light", and it is borrowed from astronomy, meaning the first data collection from an astronomical instrument.

3.1 Next Steps

Future steps in this project will be based on users' demand. There is an existing "wish-list" of tools and services that was collected by CPTEC (INPE's Center for Weather Forecasting and Climate Research) from its data users and researchers. The online and offline processing tools will be chosen to satisfy some needs of the users of the data and to show the potential of using the VO paradigm for the development of new tools and solutions.

Another important future step is the validation of the web services included in the VO for use with workflow tools (e.g. Taverna [13]). These tools allow the visual definition of processing steps to find, collect and process distributed data and will prove very valuable for quick exploration and prototyping of additional tools for the VO.

3.2 Research and Implementation Challenges

Some noteworthy research and implementation challenges being considered at the moment for the Brazilian Weather and Climate Virtual Observatory are:

- Develop a metadata scheme that can deal with spatiotemporal data coverage in an efficient and compact way. Some of the data that will be stored in the thematic data servers can be represented as time series containing specific physical measures related to a geographic location (e.g. daily rainfall in a city). The metadata for this data server must represent the period in which the data was collected and its frequency, but also must somehow represent existing gaps in the data, so a potential user can know, beforehand, if that dataset will be suitable for his/her needs. The specific challenge is to represent the gaps in a way that will not make the metadata itself too large or complex.

– It is expected that the VO may collect redundant data or data that could be substituted by alternative data under certain conditions. For example, some large cities have their own meteorological stations, that may collect data with a different time frequency from other sources. Mechanisms to indicate alternative data sources could be implemented, considering spatiotemporal coverage and data quality indicators [14, 15].
– Some of the tools for the VO (e.g. the ones implemented in the Batch Data Processing Application) may require a large amount of CPU cycles, but at the same time they may not need to be executed frequently – one example is the visual outliers map tool [12], that shows which time series are too divergent from similar time series in a neighborhood, and can be used to detect problems in the data collection. These data- and CPU-intensive applications may demand additional resources from the private cloud, so we must investigate ways to automatically deploy additional resources when needed, and release them after computation is complete. As we mentioned in subsection 2.3, monitoring the performance of the virtual servers in the cloud may also give interesting insights on its operation, which could be used to optimize the resources available.

4 Conclusions

This paper presented the proposed architecture for a private cloud-based Virtual Observatory (VO) for Weather and Climate data, still under development. Some tools for real use will be created, and efforts to garner support from the scientific community and population in general will be done as part of the VO objectives.

Most of the resources (hardware, software and data) for the Brazilian Weather and Climate Virtual Observatory ill be initially developed by and hosted at INPE, the Brazilian National Institute for Space Research. INPE collects and generates, through its several scientific missions, a very large amount of earth observation data, including meteorological data from sensors and weather model simulations. Some of this data is presently available through web interfaces designed for human use [16], but without data access integration, full metadata or centralized information about it.

Besides the VO itself, some of the results we expect to achieve with the development and implementation of it are:

– Educational and outreach material for users of the VO tools (e.g. code samples, simple fully documented applications, tutorials for using the available data and including the users' own data on the VO, etc.).
– Know-how on data federation, curation, publication and distributed processing that can be applied to other data-related research at INPE and other institutions.
– Studies on how the Brazilian Weather and Climate Virtual Observatory can integrate with other existing frameworks that could benefit of data interchange (e.g. the Environmental Virtual Observatory, http://www.evo-uk.org/).

Acknowledgments. The authors would like to acknowledge the grants provided by the Brazilian Space Agency (AEB) and Brazilian Research Council (CNPq), process number 560188/2010-2.

References

1. Djorgovski, S., Williams, R.: Virtual observatory: From concept to implementation. In: Proceedings of the Astronomical Society of the Pacific Conference Series. Volume 345., Astronomical Society of the Pacific (2005) 1–14
2. Szalay, A.S.: The national virtual observatory. In: Astronomical Data Analysis Sofware and Systems X. Volume 238., ASP Conference Series (2001) 3–12
3. Donnellan, A., Rundle, J., Fox, G., McLeod, D., Grant, L., Tullis, T., Pierce, M., Parker, J., Lyzenga, G., Granat, R., Glasscoe, M.: Quakesim and the solid earth research virtual observatory. In Yin, X., Mora, P., Donnellan, A., Matsu'ura, M., eds.: Computational Earthquake Physics: Simulations, Analysis and Infrastructure, Part II. Springer (2007) 2263–2279
4. Fox, P., McGuinness, D., Cinquini, L., West, P.G., Benedict, J.L., Middleton, D.: Ontology-supported scientific data frameworks: The virtual solar-terrestrial observatory experience. Computers and Geosciences (2009) 724–738
5. Gurney, R., Emmett, B., McDonald, A., Blair, G., Buytaert, W., Freer, J.E., Haygarth, P., Rees, G., Tetzlaff, D., EVO Science Team: The environmental virtual observatory: A new vision for catchment science. In: American Geophysical Union, Fall Meeting. (2011)
6. Matray, P., Csabai, I., Haga, P., Steger, J., Dobos, L., Vattay, G.: Building a prototype for network measurement virtual observatory. In: Proceedings of the 3rd annual ACM workshop on Mining network data (MineNet '07), ACM (2007) 23–28
7. Kiemle, S.: From digital archive to digital library – a middleware for earth-observation data management. In: Research and Advanced Technology for Digital Libraries. Springer (2002) 61–73
8. Sinderson, E., Magapu, V., Mak, R.: Middleware and web services for the collaborative information portal of nasa's mars exploration rovers mission. In: Proceedings of Middleware 2004, Springer (2004) 1–17
9. Guerra, E., Oliveira, E.: Metadata-based frameworks in the context of cloud computing. In Zaigham, M., ed.: Cloud Computing - Methods and Practical Approaches. Springer (2013) 2263–2279
10. Fowler, M.: Inversion of control containers and the dependency injection pattern. http://martinfowler.com/articles/injection.html (2004)
11. Guerra, E., Fernandes, C., Silveira, F.: Architectural patterns for metadata-based frameworks usage. In: Proceedings of Conference on Pattern Languages of Programs. (2010)
12. Garcia, J.R., Monteiro, A.M., Santos, R.D.C.: Visual data mining for identification of patterns and outliers in weather stations data. In: Proceedings of IDEAL 2012, Springer (2012) 245–252
13. Hull, D., Wolstencroft, K., Stevens, R., Goble, C., Pocock, M.R., Li, P., Oinn, T.: Taverna: a tool for building and running workflows of services. Nucleic Acids Research **34**(2) (2006) W729–W732
14. Cruz, S.A.B., Monteiro, A.M.V., Santos, R.: Increasing Process Reliability in a Geospatial Web Services Composition. In: The 17th International Conference on Geoinformatics 2009, Proceedings. (2009)

15. Cruz, S.A., Monteiro, A.M., Santos, R.: Automated geospatial web services composition based on geodata quality requirements. Computers & Geosciences **47** (2011) 60 – 74
16. Andrade, R.B., Nunes, L.H., Barbosa, E.B., Vijaykumar, N.L., Santos, R.D.C.: A web service-based framework for temporal/spatial environmental data access. In: Proceedings of the 12th International Conference on Computational Science and Its Applications, IEEE (2012) 7–13