Journal of Computational Interdisciplinary Sciences (2011) 2(1): 47-56

Printed version ISSN 1983-8409 / Online version ISSN 2177-8833
http://epacis.org

The VLADA white paper: building an active Virtual Lab for Advanced Data Analysis

Murilo da S. Dantas[1,2], Reinaldo R. Rosa[1], Nilson Sant'Anna[1], Moacyr G. Cereja Jr[3], Thalita B. Veronese[1], Silvia Bianchi[1], Julia C. Rosa[1], Kiril M. Alexiev[4] and José D.S. da Silva[1]

Manuscript received on November 11, 2010 / accepted on March 15, 2011

ABSTRACT

This technical white paper describes the design and initial implementation of a virtual environment for straightforward and robust data analysis intended for students and researchers acting in science and technology. The Virtual Laboratory for Advanced Data Analysis (VLADA) aims to fill a growing demand for scientific mathematical and statistical tools validated and coupled with appropriate high performance computing infrastructure into a single computing environment available on the Web using advanced parallel processing and object-oriented programming. This work proposes to provide: (i) a detailed study on the feasibility of building a such virtual environment with large international access, and (ii) a description of a preliminary single prototype including a standard method for advanced time series analysis. The main steps taken to develop such a laboratory, including preliminary software engineering implementation, are shown in this paper.

Keywords: computational data analysis, virtual systems, advanced parallel processing, object-oriented programming, software engineering.

1 INTRODUCTION

In recent decades, due to technological advancement in sensor data acquisition, the observation and collection of scientific data became almost automatic procedures in many applications. Similarly, the development of advanced mathematical tools in computer enabled the automation of data analysis methods. Hence, the advanced mathematical analysis of massive data bases is rapidly becoming a key component of scientific research and related technological applications. More specifically, there is an increasing use of computers in data processing, visualization and analysis involving new methods to improve data mining in order to provide new scientific knowledge, variability pattern characterization, system identification, control and warning from real-time monitoring. Therefore, there are many computational packages available for scientific data analysis to assist scientists in this effort. Most of them are free software packages (GNU-style) with appropriate syntax that can be downloaded and installed on personal computers as stand-alone (offline, non-Internet) programs only.

Correspondence to: Murilo da S. Dantas – E-mail: murilo.dantas@lac.inpe.br
[1]Lab for Computing and Applied Mathematics (LAC), National Institute for Space Research (INPE), S.J. dos Campos, SP, Brazil.
[2]FATEC, S.J. dos Campos, SP, Brazil.
[3]SESIS Sistemas de Engenharia de Software Ltda, S.J. dos Campos, SP, Brazil.
[4]Institute of Information and Communication Technologies (IICT), Bulgarian Academy of Sciences, Sofia, Bulgaria.

The usual personal computers normally have limited computing resources by using only one or by combining two multicore processors and graphics card with graphics accelerator. Thus, the development of online avaliable electronic packages and corresponding cyber-infrastructure where researchers can perform advanced data analysis are recently required. Note that, for analysing data in a such virtual environment, there is no need to install any scientific package and the task can be performed using a low-cost basic plataform as standard personal computers. In fact, online virtual environments where people can work and interact in a somewhat realistic manner have been gaining great interest and potential as sites for virtual universities, observatories and labs [1]. On this basis, VLADA has been designed as the first virtual data analysis computing environment for the World Wide Web scientific community. The pilot prototype of VLADA includes time series analysis. In the wide range of data science, time series analysis has been important in a number of different scientific communities, the most important of which are statistical physics and nonlinear dynamics with several applications in environmental, space and material sciences, genomics and econometrics.

An easy-to-use virtual advanced time series analysis package should contain a specific selection of common statistical, plotting and modelling functions as, for example, Detrended Fluctuation Analysis (DFA) [2, 3]. It is usually applied for robust characterization of long-range correlation in relatively short nonlinear time series (e.g., [4, 5]). In general, all data analysis packages support a wide variety of traditional methods, but are limited when handling nonlinear time series using advanced tools. Some examples of such mathematical tools are given in the Table 1. Considering that DFA is the most commonly used technique and its implementation is relatively easy, the feature chosen to be implemented in the prototype for VLADA is the computation of the scaling exponent for characterization of long-range correlations in nonlinear time series. In particular, applications of the DFA-technique have therefore gained increasing popularity. Thus, DFA has been selected as the canonical advanced data analysis technique in the prototype of VLADA (see Appendix).

The rest of the paper is organized as follows: Section 2 gives a technical overview of the system. The implemented software engineering resources are described in Section 3. Finally, in Section 4 we discuss the expected short-term results and outline some concluding remarks and challenges that may motivate the further steps in this project.

2 DESCRIPTION OF THE VIRTUAL LABORATORY

2.1 A technical overview

The VLADA working group (VWG) has been organized into the following four initial teams: Management Resources Team (MRT), High Performance Networks Team (HPNT), Software Engineering Team (SET) and Data Analysis Algorithm Team (DAAT). In this logical collection of basic work tasks, it is worthly to mention that the main goal of the DAAT is to deploy, certify and develop, more efficiently, useful data analysis algorithms in a transparent and public collaborative forum (analogous to the teams working on open-source software development projects). It should be emphasized that the success of the project critically depends on the VWG Long-Term Strategic Plan which should provide a virtual laboratory that must have a final structure that is as close as possible to an actual laboratory for data analysis procedures. Ongoing improvements, both in software and hardware, and in quality and speed of internet connections, will enable us to enhance the virtual laboratory prototype and complement it with new elements which are defined as software modules, called Virtual Labs (VLab). A possible structure of VLADA, containing six virtual labs, is the following:

- VLab1: Visualization Tools and Standard Data Analysis;
- VLab2: Advanced Statistical Tools;
- VLab3: Advanced Tools for Time Series Analysis;
- VLab4: Advanced Tools for Image and Spatio-temporal Analysis;
- VLab5: Advanced Tools for Multivariate Data Systems;
- VLab6: Advanced Data Mining Techniques.

VLab1 is divided into visualization and standard data analysis as, for instance, the calculation of statistical moments, histograms and autocorrelation functions. Once the data is activated in VLADA specifying the required analysis, the chosen tasks from a large list of traditional data analysis routines (found in many text books of classical statistics) are generated automatically by VLab1. Hence, as shown in Figure 1, VLab1 is expected to enclose the major amount of standard techniques. Moreover, all techniques in this module are supposed to be well known in the academic community. Consequently, the VLab1 implementation procedure although simple, is quantitatively more complex and, from the point of view of a mature user, its merit will be hard to be assessed from a virtual environment. However, due to academic and completeness purposes of this project, we understand that VLab1 should be part of the system in a long term open project as VLADA is.

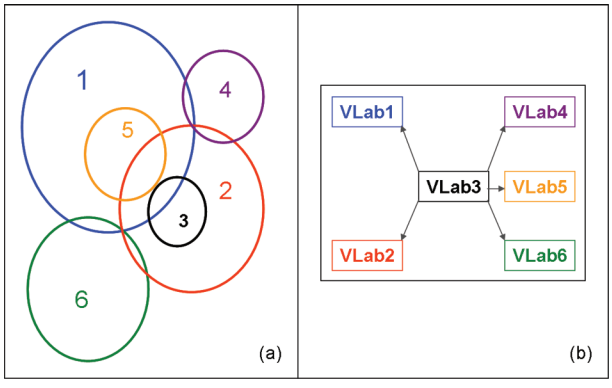

Figure 1 – (a) The VLab domains and (b) the strategy of implementation after VLab 3.

Table 1 – A reference list of relatively recent techniques that have been used, specially in physical sciences, for advanced time series analysis over the last twenty years.

Technique	Main Output	Year [Main Ref.]
Detrended Fluctuation Analysis (DFA)	Scaling Exponent	1992 [2-9]
Multifractal Spectral Analysis (MSA)	Singularity Spectra	1992 [10-12]
Gradient Spectral Analysis (GSA)	Gradient Asymmetry Spectra	1999 [13-15]
Trajectory Parallel Method (TPM)	Trajectory Curve	1999 [16,17]

VLab2 may be a complementary open module to pursue as much as possible the most important advanced statistical tools, previously validated by the DAAT. A schematic overview of all VLabs as interconnected and complementary software modules is given in Figure 1, where VLab3 has been selected to be the initial core of the VLab system. After development of the prototype module (VLab3), the complementary modules (1, 2, 4, 5 and 6) will be simultaneously developed, so that the entire systems will be expanded further. This is illustrated in Figure 1(b). Besides the DFA, other some typical advanced techniques which would be compatible with VLab3 are listed in Table 1.

In addition, each VLab must contain the all necessary information to be carried out the virtual analysis and it must be as easy as possible to help those users who may not be familiar with the available analytical tools. In this way, the users will be able to access independently a Virtual Knowledge Repository (VKR) composed of three virtual knowledge libraries (VKLib):

- VKLib1: Techniques for Data Analysis: Basic Text, References and Links;
- VKLib2: Packages and Codes;
- VKLib3: Samples and Data Repository.

These multi-module elements might be used to enable data analysers (scientists and students) to perform virtual data analysis as a straightforward procedure. Therefore, the architecture implementation group should explore the variety of possible network topologies and discuss methods of collecting information (data and analytical requirements) from users such as scheduled and on-demand harvesting, taking into account management overhead, adaptability and timeliness. It should also be determined how to combine searching and harvesting services as part of a comprehensive solution. Some multi-tier network topologies are available performing combination of repositories, registries, and access points. A multiple repository with single registry, corresponding appropriate topology and archive perspective should be the first to be addressed. The mathematical tools for data analysis will define a kernel accessed through an interface, as shown in Figure 2.

VLADA has been designed to work as a dedicated system that allows the registration of users through a robust interface. The registered users perform a Basic Logical Procedure (BLP) for Data Analysis Services (DAS) based on Laboratory Modules (VLab) and Knowledge Library Modules (VKLib), as shown in Figure 3. In this figure, an important part of the process is the step 6. In this last procedure the users could send a standard statement to VLADA, reporting the quality of the obtained results.

As it can be seen in Figure 4, VLADA has been designed to be an active collaborative open project. It may require additio-

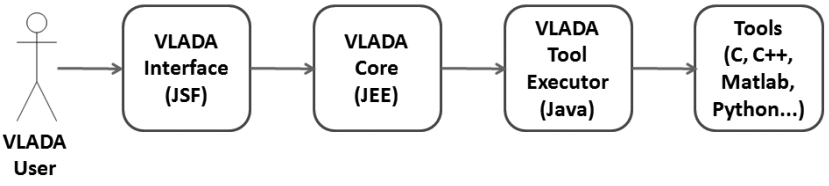

Figure 2 – The routines at the core of VLADA are being developed in JEE, while the interface will be developed in JSF. The data analysis algorithms, usually written in $C/C++$ or in higher level scientific computing languages, will be executed from the VLADA Tool Executer developed in JAVA.

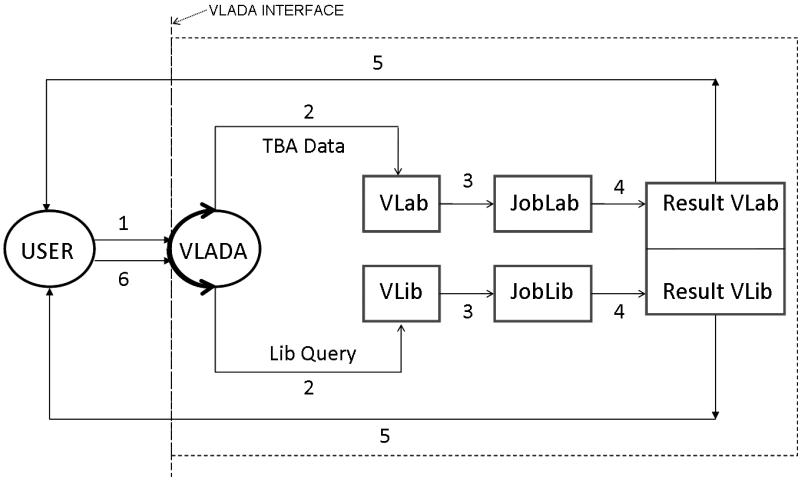

Figure 3 – A registered user access VLADA for VLab and/or VKLib services. VLADA returns to the user the result after Job module response. Two possible inputs from users are "to be analyzed data" (TBA Data) and "Library Information Requirement" (LI Query). The six basic logical steps are identified. The user is an active component only for steps 1, 2 and 6. In the step 6 the user returns to VLADA the service evaluation.

nal partners, for client advances, in the allocation of hardware and software development in order to form a global network of virtual labs, increasing the quantity and variety of analysis tools and computational infrastructure. Nevertheless, as shown in Figure 5, a preliminary virtual server environment for routine integrity should be designed to optimize server utilization in real time allowing management of physical and virtual resources. The minimal software and hardware requirements for an expandable local prototype based on LAC-INPE network and LAC-INPE community has been designed based on DFA technique and INPE's users only.

2.2 The prototype from the user's interface perspective

The VLADA prototype website (VPW) will provide an easy to use interface to access the VLab3 and its respective VKLib prototype resources. Those resources will be restricted to the VLab3 where the only technique avaliable will be DFA. The VPW will lead the INPE's user through step-by-step instructions for doing common tasks with the application of DFA. To getting started with VLADA the first step is to get the username and password using the register link. The registered user will access the Command Line Interface (CLI), which provides immediate access to the VLab (DFA) and VKLib (on DFA). The web page for each service introduces its capabilities and provides online documentation of the task.

The default task using VLab is to post your own time series, in ASCII-like format, to be analyzed by means of the DFA technique. During this procedure the user will get a WebServiceId which should be used in order to get the result of his analysis in the ResultVLab. For users without expertise on DFA, the files from VKLib are distributed as a GZip-compressed *tar* file containing source code and complete documentation on DFA. An outline of a virtual analysis, via VLab3 based on DFA, using the VPW may be seen in Figure 6.

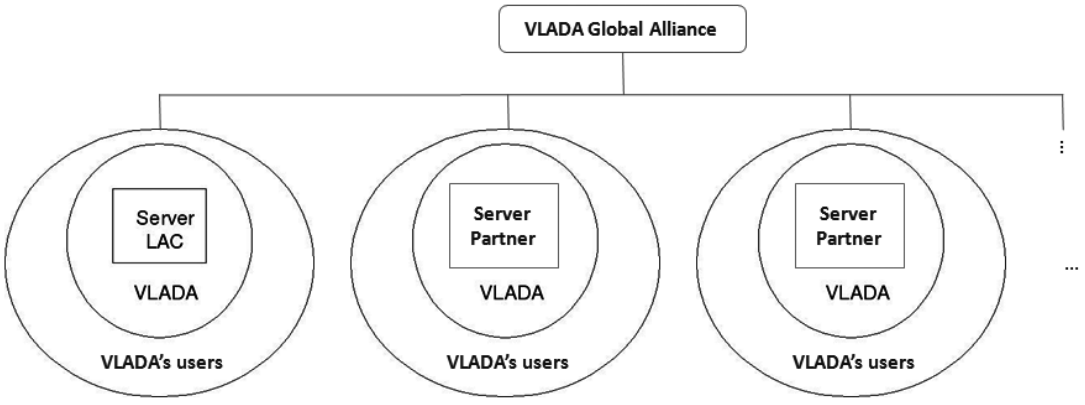

Figure 4 – VLADA Global Alliance.

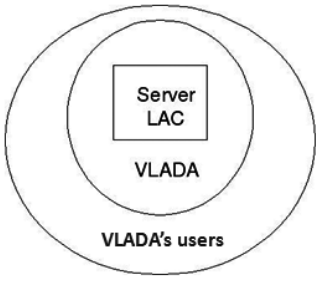

Figure 5 – VLADA prototype based on LAC-INPE.

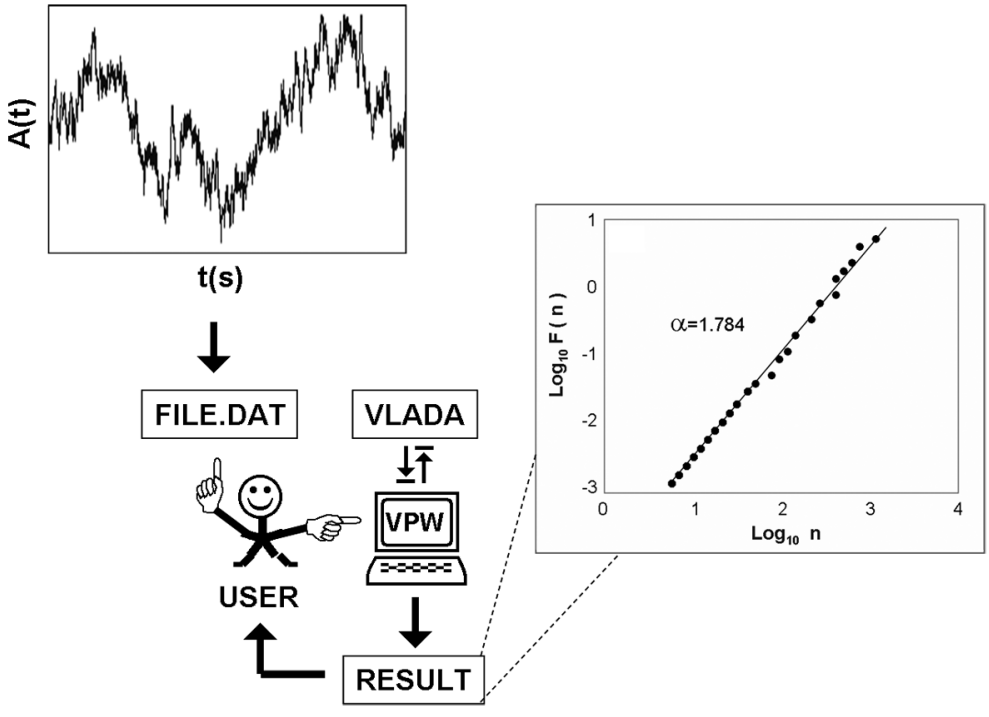

Figure 6 – A virtual Detrended Fluctuation Analysis using the VLADA prototype web site. In this example, the input is a generic time series: Amplitude $\times$ time (in seconds) written as a *.dat* format. The output is the DFA slope with the respective scaling exponent used to characterize the persistence level of the fluctuations observed in a nonlinear time series (see Appendix).

The Lab for Computing and Applied Mathematics (LAC) at INPE has been providing an appropriate initial environment to deploy the VLADA prototype. This environment is composed by 4 storages HP with 3.6 TB of storage space and 3 servers HP 2U with the following configuration: 2 processors quad-core, 32 GB RAM and 4 TB hard drives. To allow multiple tests for the VLADA prototype performance the initial software engineering infrastructure has been developed by the SE team and will be described in the next section.

3 SOFTWARE ENGINEERING RESOURCES

Selected process techniques have been used by SET to improve the quality of the VLADA prototype development effort. The documented collection of policies, processes and procedures developed/chosen by SET follows the rigid protocols for software development methodology (SDM) and system development life cycle (SDLC). In this project, we are defining some issues regarding the state of art in Software Engineering. These elements are related to VLADA's functional and non-functional requirements, architecture, open source components, etc. Some examples of these elements are:

- VLADA environment should have a very sophisticated "Access Control System", in such a way that users can have profiles, roles and permissions. This component will allow users to effectively and efficiently fulfill their jobs (high level of Usability) while accomplishes security requirements.
- We should provide a very robust architecture to keep VLADA working reliably as long as possible. In order to materialize this issue, we can use, for example, "Design Patterns", dividing the system architecture in "Layers" and using Software Engineering best practices.
- VLADA's architecture will need some components like a data base management system and an application server. For these components, we are considering PostgreSQL and JBOSS, both open source components.

The initial VLADA's software project planning provides a set of diagrams to depict software structures graphically. Figure 7 shows the main components of VLADA (high level software modules, interfaces and kernels), the *VLADA-ComponentDiagram*, having the access flow from tool executor for two possible advanced analytical tools listed in Table 1. The distribution of software modules in the hardware infra-structure is shown in Figure 8, the so-called *VLADA-DeploymentDiagram*. The possible states of a generic analytical tool, from its initial deposition to its certification, is shown as the *VLADA-ToolStateChartDiagram* (Fig. 9). The proposed functionalities defining the VLADA system are organized in the *VLADA-UseCaseDiagram* shown in Figure 10. These diagrams are shown as products that have been implemented in the present phase of this project.

The SET diagrams provide a single framework for organizing, relating, and viewing several diverse aspects of the project. It has been successfully used providing: a framework for project planning, identification of intermediate and final deliverables, a systematic method for deriving a work breakdown structure and a framework for tracking progress in terms of completed/not completed status of all activities. They are also supporting the tracing of requirements through all stages of VLADA software development.

4 EXPECTED RESULTS AND CONCLUDING REMARKS

The initial VLADA's team aspires to cooperate with others in the development of international standards for VLADA. We intend to extend the interactions of INPE, FATEC-SJC and IICT-BAC with partners already interested in this project, as groups from University of Western Ontario (Canada), Royal Institute of Technology (Sweden), Russian Academy of Science (Russia), Massachusetts Institute of Technology (USA), University of Louisiana (USA), and Universite de Caen (France). It will facilitate the international exchange of VLADA resources and allow for the standardization of the analytical tools and data format. It is also intended that, at some point in next year, the VLADA prototype will be accessible through the Internet.

For this open on-line implementation, we must evaluate computer languages, hardware, technical and algorithmic complexity and interface with users and, thus, propose scenarios for providing the most appropriate virtual environment. Formally, the next steps to be taken are the following:

- To establish criteria for the development of large systems in parallel to provide a cloud of tools for advanced data analysis;
- To make a detailed survey of technologies for software and hardware to be used in the project with growing perspectives.
- To implement analysis tools according to the technologies of software and hardware tested.

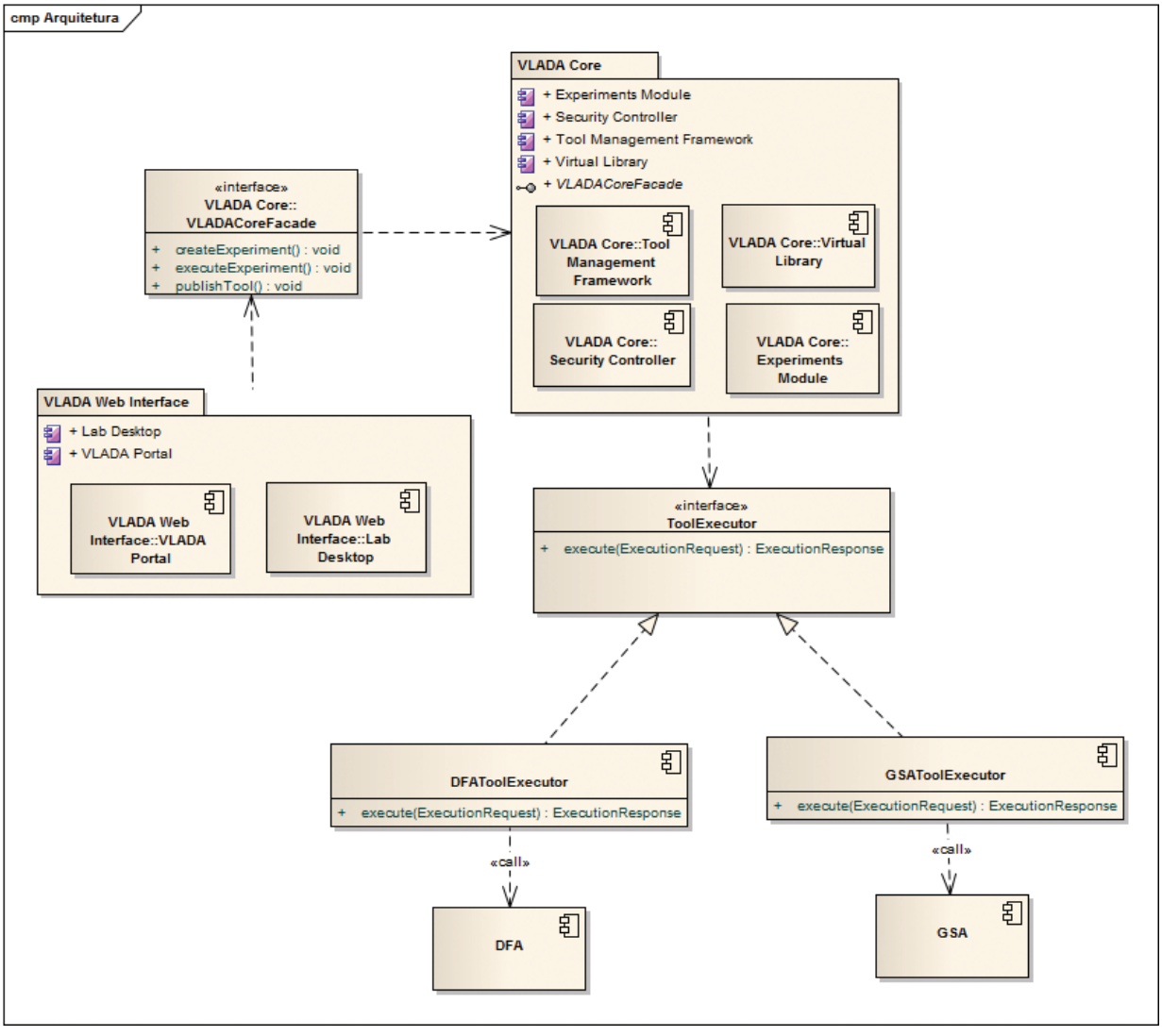

Figure 7 – VLADA ComponentDiagram.

- To develop the prototype of VLADA based on the available hardware described in Section 2.

It is expected that the results of this project will be of direct interest and use to the scientific community, especially to researchers who want to perform straightforward time series analysis having both the minimal hardware and software requirements at the moment when the analysis should be performed. Also, it will be important for academic purposes, including non-specialist users, but have no affinity with mathematical or computational technology for the development of data analysis tools. That is, VLADA attempts to offer a contribution to data analysis, developing new methodologies and providing a distributed virtual environment for the application of advanced tools not easily found in conventional data analysis packages. As a result of this effort, we will provide a new and complementary infrastructure to allow multiple users to search, discover, view, and share technical and instructional content on advanced data analysis.

ACKNOWLEDGMENTS

This work has been partially supported by LAC-INPE. The authors are very grateful to M.D.Todorov, E.F.P. da Luz, R.R. de Carvalho, H.F. Campos Velho and H.V. Capelato for fruitful technical discussions on VLADA, and to N.L. Vijaykumar and an anonymous reviewer, for helpful suggestions and comments in improving this manuscript.

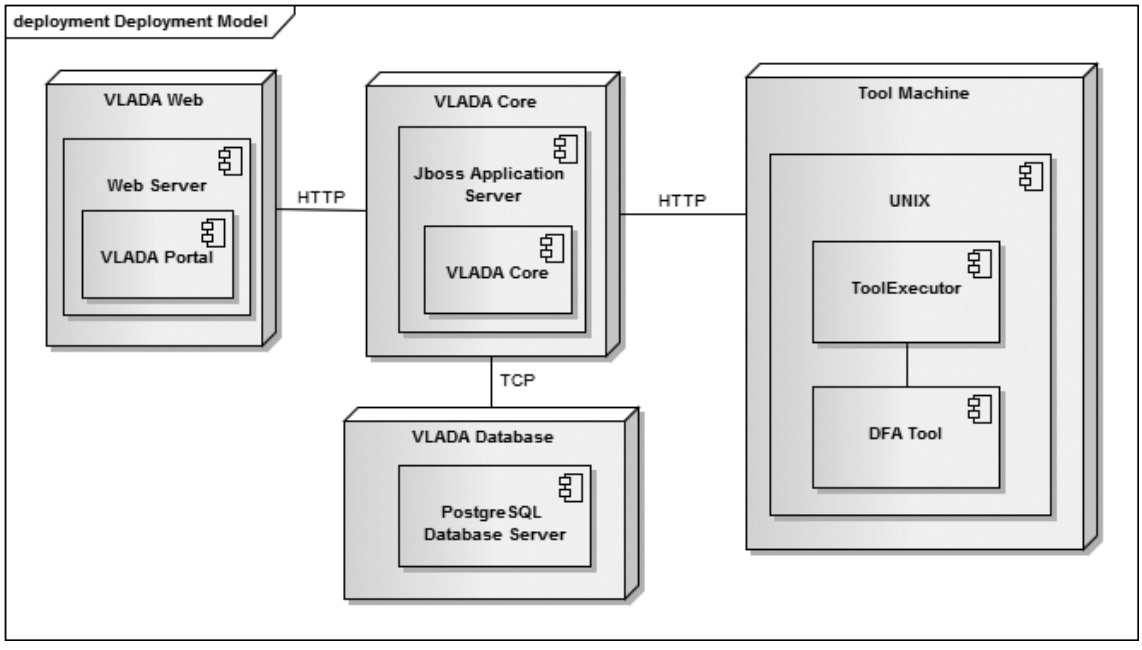

Figure 8 – VLADA DeploymentDiagram.

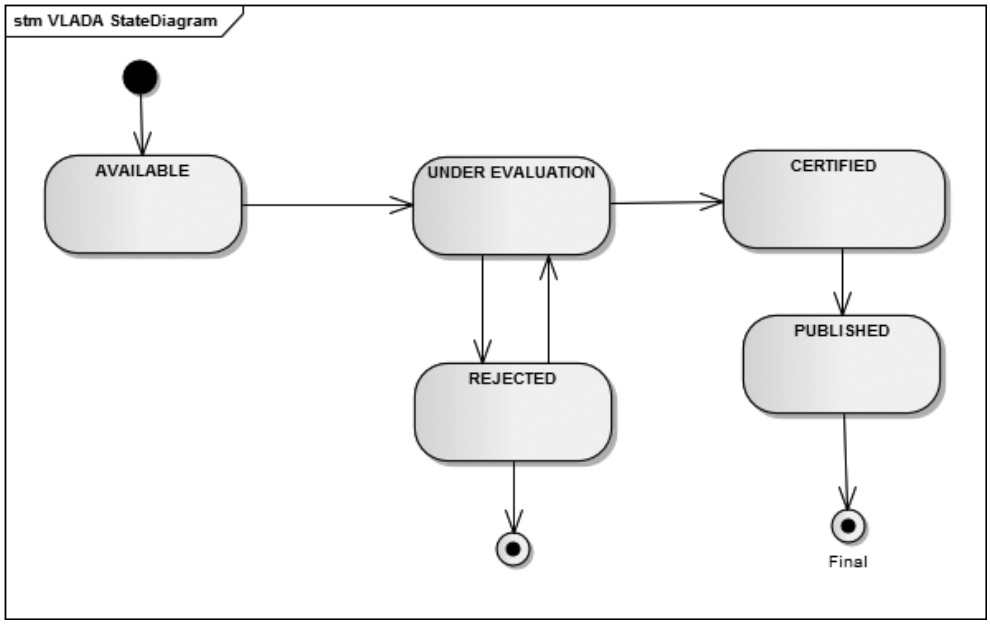

Figure 9 – VLADA ToolStateChartDiagram.

APPENDIX – Detrended Fluctuation Analysis

Detrended Fluctuation Analysis (DFA) measures the so-called *scaling exponents* from non-stationary time series. It is useful for characterizing variability patterns that appear to be due to long-range temporal correlations. The DFA technique has been used, in the last twenty years, to compute the values of the scaling exponents in several applications from physiological data to signals in physics and finance (e.g., [6, 7, 4, 8]).

The standard DFA algorithm ([9]), is composed of four main computational operations starting here on a discrete series of amplitudes $\{A_i\}$:

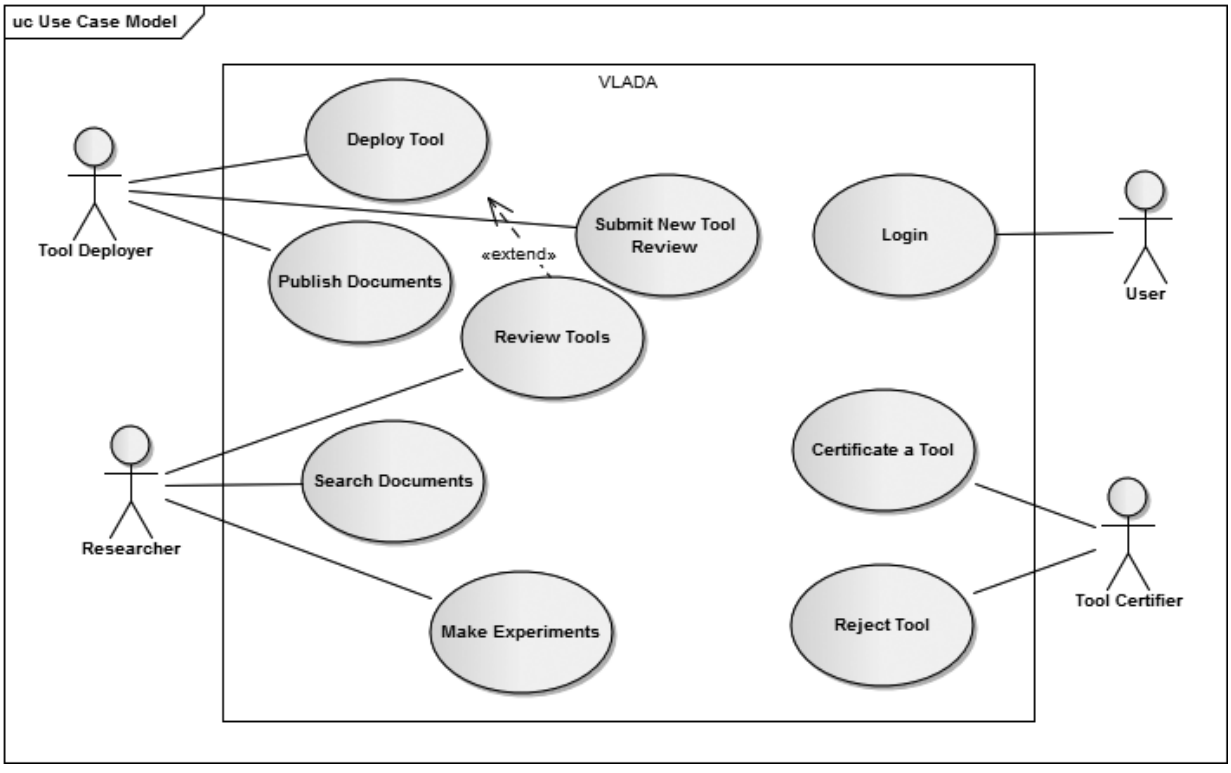

Figure 10 – VLADA UseCaseDiagram.

1. *Discrete Integration and Windowing*: Compute the cumulative representation of $\{A_i\}$ as

$$y(k) = \sum_{i=1}^{k} (A_i - \langle A \rangle) ,$$

with $k = 1, 2, \ldots, N$, where

$$\langle A \rangle = \sum_{i=1}^{N} A$$

is the average of $\{A_i\}$. Using an arbitrary local window of length n, divide $y(k)$ into non-overlapping $N_n = int(N/n)$ sub-interval y_j ($j = 1, 2, \ldots, N_n$). Note that each sub-interval y_j has length n and N may not be the integer multiple of n. Then, the series $y(k)$ is divided once more from the opposite side to make sure all points are addressed, performing at the end of this operation $2N_n$ sub-intervals on each profile.

2. *Fitting and Variance*: In each sub-interval, calculate the least-square fits $p_j^m(k)$ where m is interpreted as the *order of the detrended trend*, and compute the cumulative deviation series in every sub-interval, where the trend has been subtracted: $y_j(k) = y(k) - p_j^m(k)$. Then, calculate the variance of the $2N_n$ sub-intervals for $j = 1, 2, \ldots, N_n$ and $j = N_n + 1, N_n + 2, \ldots, 2N_n$.

3. *Fluctuation*: Calculate the average of all the variances and the square root. Then get the fluctuation function $F(n)$:

$$F(n) = \left[\frac{1}{2N_n} \sum_{j=1}^{2N_n} F^2(j, n) \right]^{1/2} . \qquad (1)$$

4. *Scaling Exponent*: Perform again, recursively, computation from windowing to calculation of corresponding $F(n)$ with different $n([N/4] > n = 2m + 2)$ box lengths. In the presence of power law: $F(n) = Kn^{\alpha}$, $F(n)$ increases linearly with increasing n. Then, get the slope α using the linear least-square regression on the double log plot $\log F(n) = \log K + \alpha \log n$ (see Fig. 6). The scaling exponent $\alpha = 0.5$ characterizes that the fluctuations are uncorrelated. When $\alpha > 0.5$ the auto-correlation is persistent. Values of $\alpha < 0.5$ corresponds to long-term anticorrelations, meaning that large values are most likely to be followed by small

values and vice versa characterizing anti-persistence. Higher values of alpha characterizes stronger auto-correlations in the signal ($\alpha > 1$ indicates a non-stationary local average of the data).

REFERENCES

[1] BAINBRIDGE WS. 2007. The Scientific Research Potential of Virtual Worlds. Science 317(5837): 472–476. DOI: 10.1126/science.1146930.

[2] PENG CK, BULDYREV S, GOLDBERGER A, HAVLIN S, SCIORTINO F, SIMONS M & STANLEY HE. 1992. Long-range correlations in nucleotide sequences. Nature, 356: 168.

[3] HU K, IVANOV PC, CHEN Z, CARPENA P & STANLEY HG. 2001. Effect of trends on detrended fluctuation analysis. Phys. Rev. E, 64: 011114.

[4] BARONI MPMA, DE WIT A & ROSA RR. 2010. Detrended fluctuation analysis of numerical density and viscous fingering patterns. EPL 92, 64002. DOI: 10.1209/0295-5075/92/64002.

[5] VERONESE TB, ROSA RR, BOLZAN MJA, FERNANDESC FCR, SAWANT HS & KARLICKY M. 2011. Fluctuation analysis of solar radio bursts associated with geoeffective X-class flares. Journal of Atmospheric and Solar-Terrestrial Physics, in press. doi:10.1016/j.jastp.2010.09.030.

[6] BUNDE A et al. Phys. Rev. E, 85: 3736.

[7] BULDYREV SV, GOLDBERGER AL, HAVLIN S, MANTEGNA RN, MATSA CK & PENG C-K et al. 1995. Phys. Rev. E, 51: 5084.

[8] BAI MY & ZHU HB. 2010. Physica A, 389: 1883.

[9] PENG C-K, BULDYREV SV, HAVLIN S, SIMONS M, STANLEY HE & GOLDBERGER AL. 1994. Phys. Rev. E, 49: 1685.

[10] MUZY JF, BACRY E & ARNEODO A. 1991. Wavelets and Multifractal formalism for singular signals: Application to turbulence data. Phys. Rev. Lett., 67(25): 3515–3518.

[11] MALLAT SG & HWANG WL. 1992. Singularity Detection and Processing with Wavelets. IEEE Trans. on Information Theory, 38: 617–643.

[12] BOLZAN MJA, ROSA RR & SAHAI Y. 2009. Multifractal analysis of low-latitude geomagnetic fluctuations. Annales Geophysicae, 27: 569–576.

[13] ROSA RR, SHARMA AS & VALDIVIA JA. 1999. Characterization of asymmetric fragmentation patterns in spatially extended systems. Int. J. Mod. Phys. C, 10: 147–163.

[14] ROSA RR, KARLICKY M, VERONESE TB, VIJAYKUMAR NL, SAWANT HS, BORGAZZI AI, DANTAS MS, BARBOSA EBM, SYCH RA & MENDES O. 2008. Gradient pattern analysis of short solar radio bursts. Adv. Space Res., 42(5): 844–851.

[15] DANTAS MS. 2010. Análise Espectral de Padrões-Gradiente de Séries Temporais Curtas, (INPE-15676-TDI/1450). Dissertation (MSc in Applied Computing) – National Institute for Space Research, São José dos Campos, 2009. Available at: <http://URLIB.NET/SID.INPE.BR/MTC-M18@80/2009/02.05.10.55>. Acessed: 20 MAR. 2010.

[16] FUJIMOTO Y & IOKIBE T. 1999. Measurement of determinism in time series by chaotic approach and its applications. Int. J. of Advanced Computational Intelligence, 3(1): 50–55.

[17] FUJIMOTO Y & IOKIBE T. 2000. Evaluation of deterministic property of time series by the method of surrogate data and the trajectory parallel measure method. IEICE Trans. Fundamentals, vol. E83-A, No 2: 343–349.