Web Portal for Measurement Applications based on GRID Services

Leopoldo Angrisani(1), Luigi Battaglia(2), Aniello Napolitano(3), Rosario Schiano Lo Moriello(1)

(1) *Dipartimento di Informatica e Sistemistica,* (3) *Dipartimento di Ingegneria Elettrica*
Università di Napoli Federico II, via Claudio 21, 80125 Napoli, Italia.
Ph: +39-081-7683170, Fax: +39-081-2396897,
E-mail: {angrisan, aniello.napolitano, rschiano}@unina.it

(2) *Consorzio SESM, Via Circumvallazione Esterna di Napoli 116,*
80014, Giugliano in Campania, e-mail: lbattaglia@sesm.it.

Abstract. The paper presents a Web Portal for measurement application based on GRID Service as an innovative solution to provide researchers with a "collection" of measurement algorithms characterized by huge computational burden. In its current version, the portal allows the execution of two different algorithms (i.e. estimation of spectral density power and instantaneous frequency). Other functionalities will successively be added to the portal, thus offering the opportunity of providing a complete set of algorithms and method for measurement signal processing.

I. INTRODUCTION

Digital signal processing algorithms are nowadays playing a more and more important role in most measurement processes. Also in the presence of poor digitized data, reliable and repeatable results can often be assured by complementing the adopted procedure with a suitable processing stage. The realization of a "collection" of measurement algorithms to be shared among researchers and technicians working on similar topics should thus be advisable. Complex algorithms, however, suffer from critical drawbacks, in terms of computational burden and hardware resources, when applied to very long acquired records. Moreover, Web Service-based approaches, the most adopted for Web-oriented applications, have proved unsuitable and unfeasible for complex measurement algorithms, especially for those requiring parallel computing to optimize their computational burden. All parallel tasks peculiar to a complex measurement algorithm should, in fact, be created on demand and be alive only during their operating time; these requirements cannot be satisfied by any WS-based approach.

The authors have recently suggested [1] the adoption of GRID Service as an innovative solution to provide researchers with the "collection" of measurement algorithms and concurrently overcome the cited drawbacks. GRID Service is a new hardware/software approach capable of merging the higher computational performance of parallel GRID computing with key mechanisms and standards of Web Service. More specifically, a GRID Service approach conveniently applies to measurement methods requiring the execution of similar and repetitive computations on large data sets; these computations can, in fact, be distributed on several GRID nodes for their independent accomplishment. Obtained advantages in terms of processing time reduction become better upon the increasing of computational complexity of the algorithm and the bandwidth of the network linking all GRID nodes.

To highlight potentialities of GRID Services, the authors have also presented a successful example of their application to a power measurement algorithm for digital wireless communications [1]. Proposed GRID Service has been implemented on OGSA-compliant middleware GT4 [14] that provides the necessary hardware and software resources as well as security services to access the GRID Service, through one of its components, referred to as GSI [6] (GRID Security Infrastructure). Security services [13] turn out to be fundamental to assure (i) integrity and privacy of data exchanged between user and GRID Service, (ii) authenticity of the user, and (iii) correct use of GRID resources. GSI exploits, in particular, asymmetric cryptography, based on PKI (public key infrastructure), in order to assure secure authentication and communication. PKI is based on two keys, referred to as private and public, used to encrypt and decrypt messages, respectively. Each GRID entity, either user or host or service, is univocally authenticated through a certificate including primary information: (i) subject (user, host or GRID Service identity), (ii) public key belonging to the subject, (iii) identity of a CA (Certificate Authority), mandated to certify that either public key and identity belong to the subject, and (iv) CA digital signature. All GSI certificates are encoded in IETF (Internet Engineering Task Force) X.509 standard, and signed and trusted by at least one CA.

On one hand, the presence of authentication mechanism makes GRID environment very secure, but, on the other hand, involves a very complex and onerous procedure for client-server interactions, thus allowing only expert user to use GRID services and slowing down their diffusion and development.

To overcome this limitation, the authors suggest hereinafter a new strategy based on a Web Portal for sharing GRID Services among users connected to Internet. The Web Portal acts as an intermediate tier between the clients invoking GRID Services and hardware and software resources of GRID environment. This way, it is so possible to simultaneously take advantage of the easiness of client access (granted by standard procedure based on user ID and password) peculiar to Web sites with the astonishing performance granted by GRID Services. As an example, a first release of Web portal is presented in the following; a new GRID Service, mandated to instantaneous frequency estimation, has been implemented and made accessible in addition to the service already proposed in [7][8].

II. WEB PORTAL FOR INVOKING GRID SERVICES

Stemming from what stated in section I, a fully-accessible Web portal (WP) for measurement applications based on GRID Services is presented in the following. The main advantage associated with the adoption of a WP is its widespread use throughout the network applications; a number of WP have, in fact, been implemented on different browser in order to provide an interface to access a broad variety of information sources. Despite of their common diffusion, WPs have not yet been adopted for use in GRID environment. A WP can be exploited as a graphic user interface (GUI) through which an external client can access and take advantage of resources and services of GRID environment without accounting for all the strategies and resources needed to attain the measurement result. From this point of view, the client interacts with the GRID in such a way to consider the GRID itself as a virtual computing resource. Moreover, the use of WP allows clients to use the implemented GRID Services without taking into account the underlying security services. The client can, in fact, access into the WP with a standard used ID/password mechanism. On the other hand, the WP run on an authenticated and certified GRID node that is, thus, admitted to exploit GRID resource and services. This way, the inherent high security level of GRID infrastructure is maintained, but the possibility of taking advantage of GRID Services is granted also to unskilled clients.
The open source integrated development environment (IDE) Eclipse has been adopted for implementing the whole services in Java and HTML language. Thanks to the software package GDTE [9], Eclipse, in fact, offers the opportunity of automatically fulfilling a number of steps needed to realize a Globus Toolkit 4 [11][12][13] (GT4)-based GRID Service. The authors have, in particular, implemented some specific-purpose Web Service (WS), deployed them by means of GT4 scripts, and made them run in the GT4 Web Services Container [5]. All fundamental steps and adopted software strategies can easily be tailored to any measurement service characterized by a certain degree of inherent parallelism. The middleware GT4, however, does not provide any service or tool to generate a WP for accessing the GRID Services. To overcome this lack, the WP has been implemented as java servlet integrated in a java server page (JSP); the JSP is a multiple platform solution to create dynamic HTML web page at server-side using the java language. The obtained Web page has, finally, been run in Tomcat Web Services Container (WSC).
The main functionalities required to access the GRID Services through WP, i.e. registration or login mechanisms and GRID service selection and submission, are provided by two different servlet, referred respectively to as login and application servlet, details of which are given in the following.

A. Registration or login mechanism.

An access procedure has been realized to protect and control the hardware and software resources involved into the GRID infrastructure.
When the client connects to the WP through a standard HTTP connection, the principal access page is presented. Two options are provided to the client:

1. If the client accesses to the WP for the first time, he has to register in the database of authenticated and admitted client. To this aim, the client must complete a registration form with typical data useful to his identification and authentication (such as name, e-mail address, phone number and so on) and submits the form to a login servlet. The servlet inserts the received data in a registration-table of SQL database, calculates the user ID and password and returns them to the client for successive accesses (Fig.1).

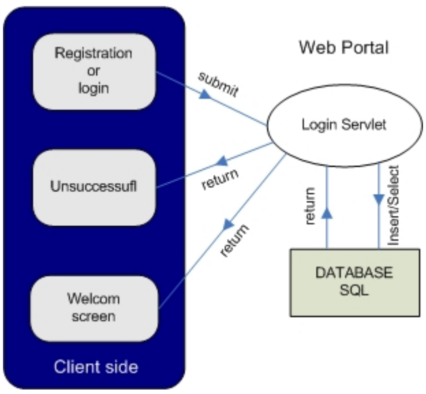

Figure 1 Registration and login mechanism.

2. If the client is already present in the SQL database, he can access to the implemented GRID Services through user ID and the password previously obtained. The login servlet compare this data with those contained in the registration-table of SQL database. If a complete matching occurs, the client in provided with a welcome page, otherwise, an unsuccessful login screen is given and the access to GRID Services is forbidden. An example of the implemented web page for login is given in Fig. 2.

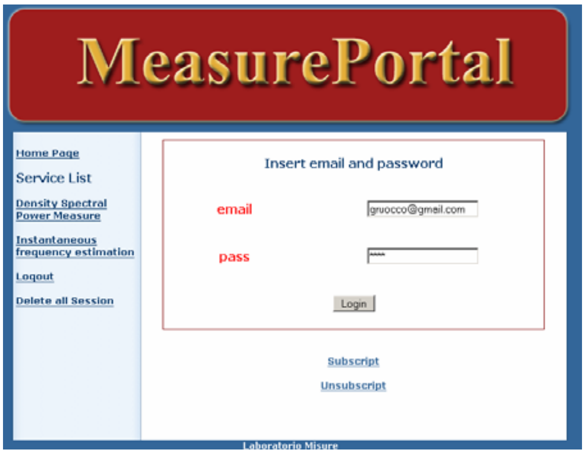

Fig.2 Web page generated for login mechanism

B. Selection and submission of GRID Service.

Once the client is logged into the WP, he can select a GRID Service from those available on welcome page and submit its execution into the GRID infrastructure. As stated above, the client cannot directly require the service submission, since he does not possess on his own terminal the required certificates. His submission request is thus redirected to the WP application servlet that executes the actual service submission encapsulating the client selection within a new request carried out by an account authenticated and certified in GRID infrastructure. Moreover, the implementation of the application servlet on Tomcat WSC allows multiple submission requests, sent from different clients, to be effortlessly managed, thus granting multithread services execution. Each GRID Service thread is invocated according to the software design pattern, referred to as factory/instance pattern [3],[4], recommended by WSRF (Web Service Resource Framework) [10]. The invocation has been, in particular, split into two main parts mandated respectively to service initialization and execution. With regard to the first part, it accounts for the initialization of GRID Service resources; for all implemented services, the associated resources are *measurement_result* (i.e. the result obtained from the measurement), and *address_ip* (i.e. the ip address of the client that has required the GRID Service). The initialization is carried out through three java classes, referred to as Factory Service, Resource Home and Resource, that interacts to one another as shown in Fig 4. In particular, to create new resources, application servlet invokes the ***createResource*** method on Factory Service, the location of which is known by means of the associated URI (universal reference identifier). This operation can be accomplished only executing the ***create*** method on Resource Home, i.e. the class mandated to manage all the needed resources. To make Factory Service capable of invoking the desired method, Resource Home reference must be obtained by means of a GRID helper class referred to as ***ResourceContext***. Once the Resource Home reference has been achieved, Factory Service can invoke the ***create*** method on Resource class, that create the instance of required resource returns an object of type ResourceKey, i.e. a resource identifier (Fig.3). Created resources and associated object are finally added in an internal list of Resource Home; the list allows client to access any available resource though the corresponding identifier. The obtained object along with the URI of the GRID Service is the service End Point Reference (EPR).

Thanks to the obtained EPR, the application servlet can now invoke the methods of the desired GRID Service, i.e. ***data-method*** and ***measurement-method***. The former method manages the whole measurement procedure, from data generation and data acquisition to results evaluation. To this aim, the authors have provided the method with a

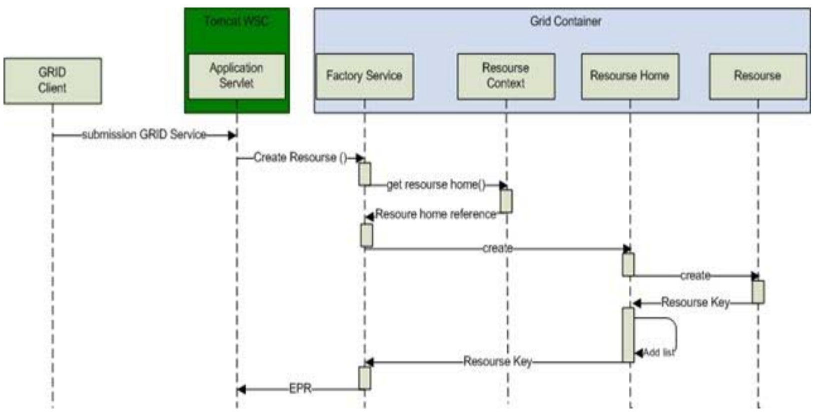

Fig.3 GRID Services initialization according to factory/instance pattern.

straightforward graphic user interface (GUI); the GUI allows the client to select the measurement station he needs and configure typical parameters of the adopted instrumentation through proper fields. It is worth noting that fields are initially completed with default values in order to allow also inexperienced clients to correctly exploit the selected GRID Service. Required configuration is, finally, sent to the instrument that returns the input signal samples. More specifically, the measurement station adopted for the realized GRID infrastructure consists of a vector signal generator, namely *ESG 4438C*^{TM} by Agilent Technologies (250-6GHz of output frequency, 80 MSample of memory depth and 16 bit of vertical resolution), and a digital storage oscilloscope, namely *SDA 6000A*^{TM} by LeCroy (6 GHz of channel bandwidth, 20 MS/s maximum sample rate, 20 MSample memory depth, and 8 bit of vertical resolution).

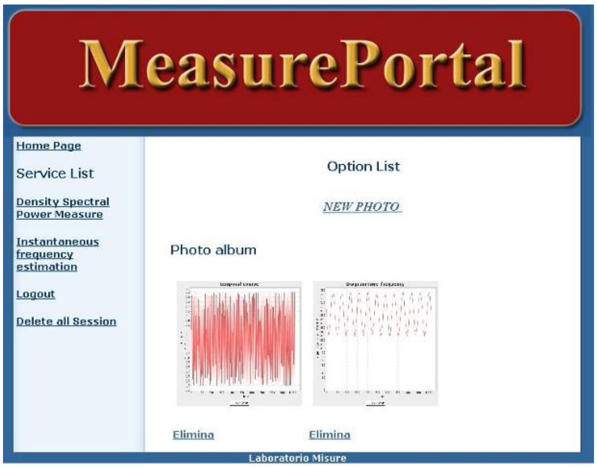

Fig. 4 Example of the web page for displaying the results

According to the specific required GRID Service, ***data-method*** invokes the ***measurement-method*** necessary to gain the desired measurement results. To this purpose, ***measurement-method*** distributes the processing stage in parallel tasks that are handled through a PBS scheduler. More specifically, the PBS [2] collects the tasks in a proper queue managed according to a first-in-first-out strategy and schedules their parallel execution on the available GRID nodes (cluster PBS). The PBS scheduler uses two files, referred respectively to as *prologue* and *epilogue* files, in order to allow all GRID nodes to share data among themselves. The former file is used to store the input signal samples on a GRID node, while the latter sends the data resulting from the processing stage to the application servlet that updates the resource *measurement_result* and delivers its value to the client. The presentation stage is inquired to the WP, which stores the results in a particular record in order to provide them to the client, when they are required. As an example, Fig. 4 shows the web page generated when the client requires graphical results. For the sake of clarify, Fig. 5 provides the interaction steps between client, application servlet and GRID Services.

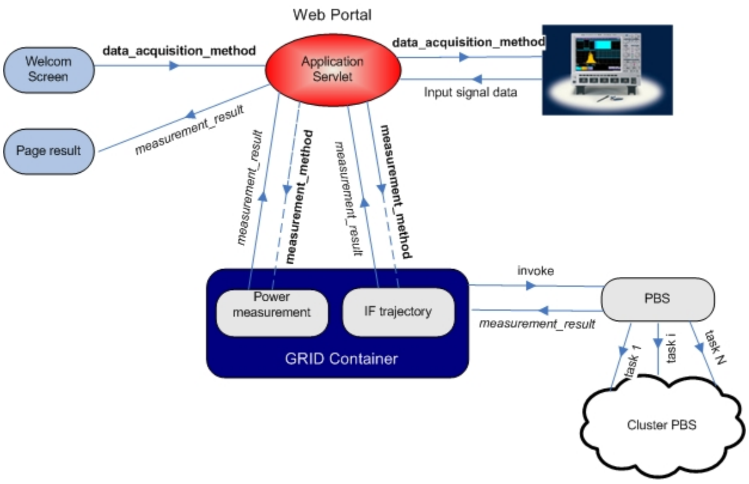

Fig. 5 Fundamental steps for submission of the GRID Services

III. IMPLEMENTED MEASUREMENT APPLICATIONS

A brief description of the measurement applications implemented as GRID Services is presented in the following; particular attention is focused on the portions of associated measurement algorithms that can be broken up in similar tasks and distributed on different GRID nodes for parallel computation.

A. Power measurement based on input signal tapering

The method, firstly proposed in [7] and based on multitaper approach, has proven very effective for power measurement in digital wireless communications. The basic idea underlying the multitaper approach is the calculation of a certain number, H, of PSD (power spectral density) estimates, each of which evaluated with a different window function, also called data taper and applied to the whole acquired signal, and then to average them together. If all H data tapers are pairwise orthogonal, the resulting multitaper estimator can exhibit attractive performance, in terms of reduced bias and variance, particularly for spectra with high dynamic range and/or rapid variations like those characterizing most signals in digital wireless telecommunication system. The multitaper estimator can thus be expressed as

$$S_x(f) = \frac{1}{H}\sum_{i=0}^{H-1} S_x^i(f) \tag{1}$$

where the terms $S_x^i(f)$, referred to as eigenspectra, are given by

$$S_x^i(f) = \left|\sum_{n=0}^{N-1} x(n) h_i(n) e^{-j2\pi f n T_c}\right|^2 \tag{2}$$

$\{h_i(n): n = 0,..,N-1; i = 1,...,H\}$ denotes a set of orthonormal data tapers, and $x(n)$ stands for the i^{th} input signal sample. Due to the simplicity of their computation a set of sine tapers is commonly adopted, the i^{th} of which is:

$$h_i(n) = \left[\frac{2}{N+1}\right]^{\frac{1}{2}} \sin\left[\frac{(i+1)\pi n}{N+1}\right] \tag{3}$$

A standard FFT algorithm proves appropriate to evaluate the eigenspectra over a grid of equally spaced frequencies. Computations of the various eigenspectra are totally independent of one another and can concurrently be conducted Then whole previous steps are divided into parallel tasks each of which involves one or more taper and obtain the own eigenspectrum. This stage of the method turns out to be the most feasible for a GRID implementation. After that the averages of the obtained eigenspectra is calculated, the wavelet thresholding procedure is applied to it to achieve the final PSD and provided the desired measurement results. More details can be found in [7].

B. Instantaneous frequency estimation by means of Warblet Transform

A digital signal processing method has been presented in [8] for instantaneous frequency (IF) estimation of IF trajectories exhibiting periodic evolution versus time. The IF estimation is a very helpful analysis and diagnostic tool in a wide range of research and industrial fields, from seismic and oceanic investigations to biomedical applications, from telecommunication apparatus test and measurement to power system quality assessment. The operating steps of the method are sketched in Fig.7. After the input signal $s(t)$ has been digitized by means of a proper data acquisition system, the short time Fourier transform (STFT) is applied on the acquired samples, according to the following expression, in discreet time:

$$STFT_{\varpi}(n,i) = \sum_{i=1}^{M}\sum_{n=1}^{M} s(n) * \varpi_i(n) e^{-j2\pi\frac{n}{M}} \qquad i,n = 1,..,M \tag{4}$$

where $\omega_i(n)$ stand for the gaussian window delayed by i samples and expressed as

$$\omega_i(n) = \sqrt{\frac{1}{\sqrt{2\pi}\sigma} e^{-\frac{1}{2}\left(\frac{n-i}{\sigma}\right)^2}} \qquad i = 1,...,M \tag{5}$$

M stands for the number of acquired samples.

According to what stated in [8], STFT coefficients can be achieved by means of the evaluation of M independent fast Fourier transform (referred to as FFT_i, i=1,..,M) each of which applied to the result of the multiplication of input signal by a Gaussian window characterized by delay equal to i. The computation of FFT_i results can, thus, be concurrently conducted into GRID environment by means of the distribution of the whole STFT in M different, parallel tasks. Moreover, each task is mandated to apply on the obtained FFT coefficients a standard peak location algorithm to single out the coefficient c_j corresponding to the significant components of signal spectrum (Fig.2). When the M tasks have been accomplished, all the coefficients c_j are collected in a record to attain a rough IF trajectory estimation. A windowed fast Fourier transform (FFT) of the rough IF trajectory, depurated of its mean value, is, then, carried out in order to evaluate frequency, amplitude, and phase constant of the significant components its spectrum consists of. For the sake of clarity, an IF trajectory consisting only of one spectral component has been considered in the following; details for the application of the method to periodic IF are given in [8]. The Warblet transform (WT) is, finally, applied on FFT result in order to give a good readability and capacity resolving on time/frequency plane to the IF trajectory. The discrete-time WT can be expressed as:

$$WT_i(n, f, a, \beta_m, f_m, \rho_m) = \sum_{i=1}^{M}\sum_{n=1}^{M} s(n) * h_i(n, f, a, \beta_m, f_m, \rho_m) \tag{6}$$

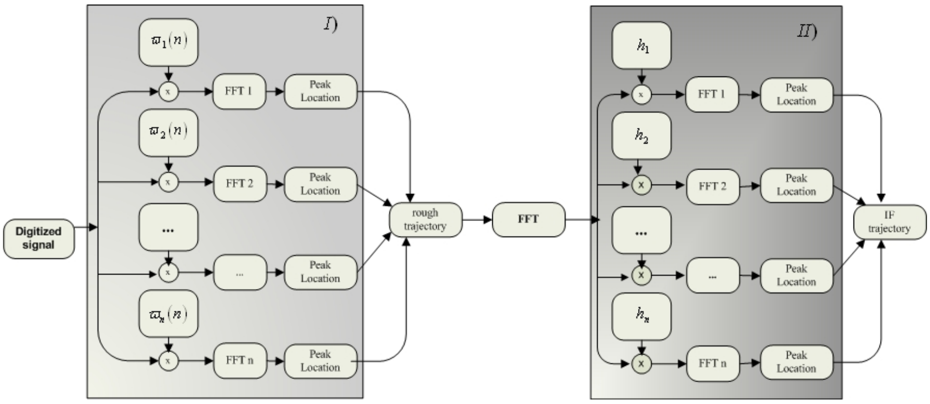

Fig 7 Block diagram of the algorithm for IF trajectory estimation.

where $h_i(n, f, a, \beta_m, f_m, \rho_m)$ is the mother warblet, given by:

$$h_i(n, f, a, \beta_m, f_m, \rho_m) = \varpi_i\left(\frac{n}{a}\right) e^{j\left[2\pi \frac{n}{M} + \frac{\beta_m}{f_m}\sin\left(2\pi f_{mt+\rho_m}\right)\right]} \quad i = 1,\ldots,M \tag{7}$$

which can be seen as the product of a time-limited function $\varpi_i(n)$, scaled by a, by a frequency modulated (FM) signal whose IF trajectory $f_{im}(t)$ is:

$$f_{im}(t) = f + \beta_m \cos(2\pi f_m t + \rho_m) \tag{8}$$

where f is the center (carrier) frequency of input signal. It is possible to demonstrate that, if the mother warblet is properly chosen, WT calculation can be carried out through the evaluation of more simple FFTs. More details can be found in [8]. Also in this case, to distribute the computational burden of WT, its elaboration has been realized through the parallel task (Fig.7).

REFERENCES

[1] L.Angrisani, L.Battaglia, R.Schiano Lo Moriello, “Design and implementation of a GRID-service for power measurement in digital wireless communication systems,” Proc. of IEEE IMTC/06, Sorrento (NA), Italia, 24-27 April 2006, pp.2300-2305.

[2] TORQUE v2.0 Admin Manual, Cluster official documentation, http://www.clusterresources.com/ torquedocs21

[3] B. Sotomayor, The Globus Toolkit 4 Programmer’s Tutorial, June 2005, http://gdp.globus.org/ gt4-tutorial/.

[4] Globus Toolkit 4 Administration Guide, Globus Toolkit official documentation, http://www.globus.org/toolkit/docs/4.0/, February 2004.

[5] Globus Toolkit 4 API documentation, Globus Toolkit official documentation, http://www-unix.globus.org/api/javadoc-4.0.0/.

[6] Grid Security Infrastructure (GSI), official documentation, http://www.globus.org/toolkit/docs/4.0/security/

[7] L.Angrisani, M.D’Apuzzo, M. D’Arco, ”A new method for power measurement in digital wireless communication systems,” IEEE Transactions on Instrumentation and Measurement, vol.52, No.4, August 2003, pp.1097-1106.

[8] L.Angrisani, M. D’Arco, R. Schiano Lo Moriello, and M. Vadursi, “On the Use of the Warblet Transform for Instantaneous Frequency Estimation,” IEEE Transactions on instrumentation and measurement, Vol. 54, 4, August, 2005, pp 1374-1380.

[9] Globus Development Tools for Eclipse (GDTE), official documentation, http://mage.uni-marburg.de/trac/gdt/wiki/TutorialsPage.

[10] Web Services Resource Framework committee-Draft-02-23 May 2006, official documentation, http://docs.oasis-open.org/wsrf/wsrf-primer-1.2-primer-cd-02.pdf.

[11] I. Foster, ”Globus Toolkit Version 4: Software for Service-Oriented Systems,” IFIP International Conference on Network and Parallel Computing, Springer-Verlag, 2005 LNCS 3779, pp 2-13.

[12] G. Fox, ”Grid Computing Environments,” IEEE Computational Science and Engineering, vol.5, No. 2, March-April 2003, pp.68 - 72.

[13] J. Jiang, D. Chen, T.Chen; X. Shen, “A Security Grid Portal Using PKI and Online Proxy Certificate Repository,” Computer and Computational Sciences, 2006. IMSCCS '06. First International Multi-Symposiums on Vol. 2, 20-24 April 2006, pp 93 – 96

[14] I. Foster, C. Kesselman, J. Nick, S. Tuecke, ”The Physiology of the Grid: An Open Grid Services Architecture for Distributed Systems Integration,” Open Grid Service Infrastructure WG, Global Grid Forum (GGF), June 2002.