RINGrid contract no. 031891

WP6: Prototype implementation and verification

Deliverable 6.2
Report on requirements verification

Document Filename:	RINGrid-WP6-D6_2-2008-04-20-1-GRN-Report_on_requirements_verification.doc
Work package:	Prototype implementation and verification
Partner(s):	PSNC, GUP, GRNET, CNIT, REUNA
Lead Partner:	GRNET
Document classification:	Public

Abstract: Deliverable 6.2 is using the results from the experimentation phase described in D6.1, alongside a use case analysis, in order to refine the outcome of D4.3 and come up with a final, complete and coherent set of requirements and recommendations for instrumentation grid infrastructures. This is being done through establishing a "model" workflow that can represent any of the experiments under consideration and then using it to define the capabilities required for remote instrumentation.

Delivery Slip

	Name	Partner	Date	Signature
From	Constantinos Kotsokalis	GRNET	20/04/2008	
Verified by				
Approved by				

Document Log

Version	Date	Summary of changes	Authors
0.1	27/02/2008	Skeleton and assignments	Constantinos Kotsokalis
0.2	25/03/2008	Draft questionnaire analysis; Model use case draft; First draft for section 5; Questionnaire responses in place;	Constantinos Kotsokalis, Anastasios Zafeiropoulos, Evi Giannakopoulou, Thomas Prokosch, Tomasz Rajtar
0.3	29/03/2008	Textual description and graphical illustrations of model use case; First draft of section 6;	Constantinos Kotsokalis, Anastasios Zafeiropoulos, Evi Giannakopoulou, Tomasz Rajtar, Franco Davoli
0.4	31/03/2008	Completed graphical representation of model use case; Completed questionnaire analysis; Next draft of section 6;	Constantinos Kotsokalis, Anastasios Zafeiropoulos, Evi Giannakopoulou, Tomasz Rajtar, Franco Davoli
0.5	01/04/2008	Introduction, executive summary and conclusions; Update on final requirements;	Constantinos Kotsokalis, Anastasios Zafeiropoulos, Evi Giannakopoulou, Franco Davoli, Thomas Prokosch
0.6	02/04/008	Update on flowchart; Update of final recommendations; Final draft of section 5; Update on executive summary; Bibliography and abbreviations in place;	Constantinos Kotsokalis, Anastasios Zafeiropoulos, Evi Giannakopoulou, Franco Davoli, Thomas Prokosch, Tomasz Rajtar
0.9-prefinal	05/04/2008	WP-Internal review by REUNA;	Marcela Larenas, Carlos Vogel, Alberto Montecinos Sandra Jaque
1.0-final	20/04/2008	Final version after addressing internal reviewers' comments.	Constantinos Kotsokalis, Franco Davoli

Table of Contents

List of Figures

List of Tables

1. Executive Summary

This document is the last deliverable of the RINGrid project, presenting the final outcome of the project's work. Effort within other work packages has produced a path of results which converged and were combined in this deliverable: Candidate use cases, existing technologies, emerging technologies, technology-driven recommendations, experimentation on prototypes.

It is useful to preface this Executive Summary with the most important reasons why Remote Instrumentation on the Grid is an advance from plain, standalone Remote Instrumentation. We will adopt the term Remote Instrumentation *Services* (RIS), to indicate instrumentation that is made accessible by exposing it in terms of Grid Services, as the Grid is a *Service Oriented Infrastructure* (SOI).

The most obvious advantage in this new approach is that we can exploit the storage and processing capabilities that the classical data/computing Grid offers. By representing the instrument as a service and integrating it with other services through well-understood protocols, it becomes (conceptually, but also technically) straightforward to directly store experimental data to arbitrary locations world-wide, replicate them in multiple locations, and perform post-processing that might previously take days or months in only a fraction of the time. The massive Grid capabilities, surpassing any kind of supercomputer performance for specific categories of scientific problems, are a perfect match for experimental and applied science.

However, the gains stemming from this integration extend far beyond this obvious advantage.

The very sharing of resources that lies in the heart of the Grid concept is of utmost importance in the case that we are studying. Securing the sharing of the instruments through industry-standard methodologies (PKI, Shibboleth, etc.) offers an unprecedented way for scientists to cooperate efficiently through standardized security means, without the hassle of implementing interoperability layers for each different experiment and site combination. Additionally, the Virtual Organization (VO) concept accompanied by implementations such as the VO Management Service (VOMS), allows easy and fast creation of scientific groups that have direct access to instruments allocated to them. Fine-grained authorization methods, currently being researched on the Grid and elsewhere, can be applied to define policies of great granularity, about what one may or may not do on specific instances of equipment. Problems such as the formation of virtual laboratories now find grounding that solves almost automatically, simply through proper configuration, some of the most basic problems that had to be faced: Remote user authentication, access policy definition, implementation of interoperable access mechanisms, transport-layer security for data confidentiality.

Instrumentation as a service, an analogy to *software as a service*, allows composing atomic (in the software sense) experimental actions into measurement chains or long-standing experimental processes, irrelevant of the location of the cooperating instruments. Workflow execution mechanisms bringing together multiple different instruments, possibly geographically dispersed, can raise experimental science to new levels. The hard-coding of previous-generation scientific applications meant that either someone had to do all the experiments manually, one by one, or that an experiment involved a single instrument on a single site. Using RIS, the composition of multiple different experiments into a single application in the form of a workflow sets a new paradigm.

This list of RIS advantages neither is nor is meant to be exhaustive; one could elaborate further on the reasons that this integration is a good idea. Scalability, dynamic nature of workflow-based applications, resiliency through replication of data, standardized representation of resource information, and many others. This short text is meant only to summarize the most important *whys*, before elaborating on the *what* and the *how* of RIS.

We start the document off by presenting the questionnaire that we built, soliciting input from various diverse user groups regarding the experiments that they typically execute as part of their work. The questionnaire is formatted in a domain-agnostic way, looking at the various use cases in an abstract manner and at the instruments as “systems” with input and output. Thus, scientists are able to provide accurate feedback using their own terminology, which we then decipher into IT terms and requirements.

This interpretation is taking place later in the document, where we analyze the results of our use case survey through this questionnaire. A “model use case”, also termed “model workflow” due to its representation as such, is then built to express the steps that a remote experiment will typically go through. We identify two indisputable, necessary such steps in the workflow, namely the execution of an experimental operation (measurement, observation, etc.) and the post-processing of each such operation. These two steps are complemented by a number of optional (in a “least common denominator” sense) steps, which may or may not be obligatory depending on the specifics of the experiment. For instance, preparation of input, setting of access policies, etc., are such “optional” steps for the model use case. After conceptually describing it, the model use case is normatively defined through a textual representation, UML diagrams, and a flowchart. Based on this, we describe the requirements that are implied or directly suggested. Security and policies, instrument virtualization, monitoring and accounting, workflow and execution management, but even the existence of a local collaborator, are elaborated on at this point.

Following, the document summarizes the findings of deliverable 4.3, which forms the basis of our list of recommendations for RIS. A list of requirements from the technological analysis previously conducted, as well as the corresponding recommendations, is briefly presented to connect with previous work. Based on this list, and the requirements from the previous chapter, we conclude the document with a final, coherent, self-contained list of recommendations. These recommendations build on the Open Grid Services Architecture (OGSA), to enhance it and adapt it to the requirements of Remote Instrumentation. This “conceptual design”, as previously termed, includes not only recommendations which are a result of direct and obvious technical requirements of the instruments, but also those that are based on user comments and must be considered as development guidelines for interfaces.

2. Introduction

The RINGrid project has started 18 months ago on a roadmap to study the requirements for Remote Instrumentation Services in a Grid computing context. This document concludes the effort, offering a final, self-contained, comprehensive list of requirements that must be fulfilled for efficient use of instruments on the Grid. Figure 1 shows the information flow between project work packages.

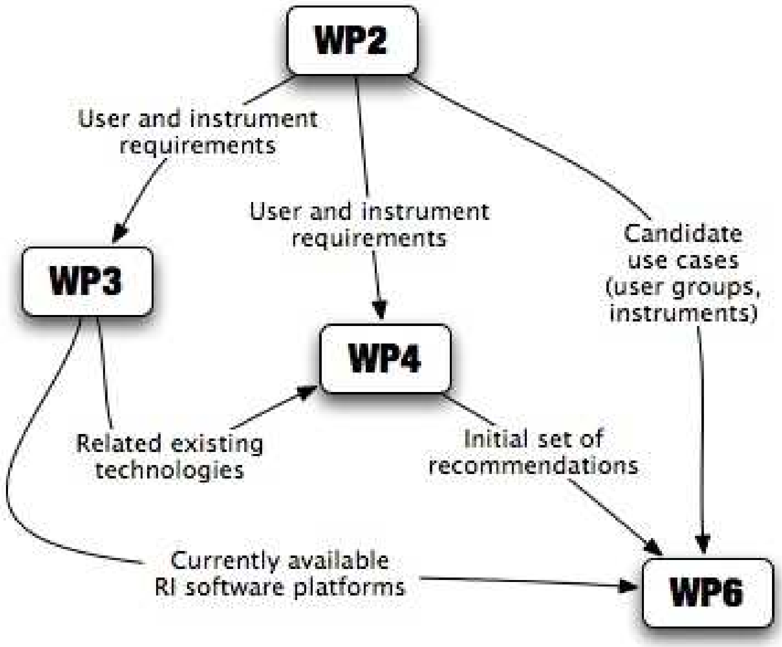

Figure 1: Information flow between work packages

Work Package 2 (WP2) made a survey and identified user groups that can benefit from the project's work, the respective instrument installations and their instrument-oriented requirements. Based on this survey, a database was built including information on contacts and instruments. Work Package 3 (WP3) started with this input and performed an extensive technology study for all currently existing technologies, which are relevant to the requirements mentioned in WP2. This research extended both to the networking and to the software (RI platform) worlds, touching on subjects as diverse as optical networking and Web Services Architecture [WSARCH]. Work Package 4 (WP4) continued from that point, looking at emerging technologies in the same areas that WP3 studied. With the results of the various parts of these studies, WP4 concluded with a list of recommendations for RIS, looking at the problem from a technological point of view. Figure 2 illustrates how the various pieces of information come together in order to construct this list of requirements - what has also been previously referred to as a "conceptual design".

Devising a detailed architecture is out of the scope of this work; therefore, we rely on the Open Grid Services Architecture [OGSA], as also mentioned in deliverable 4.3 [D4.3]. D4.3 identified the essential and relevant OGSA parts based on a study of emerging technologies that seem to be relevant. At the same time, it took into account the work of WP3 on existing technologies that address user and instrument requirements, as recorded in WP2.

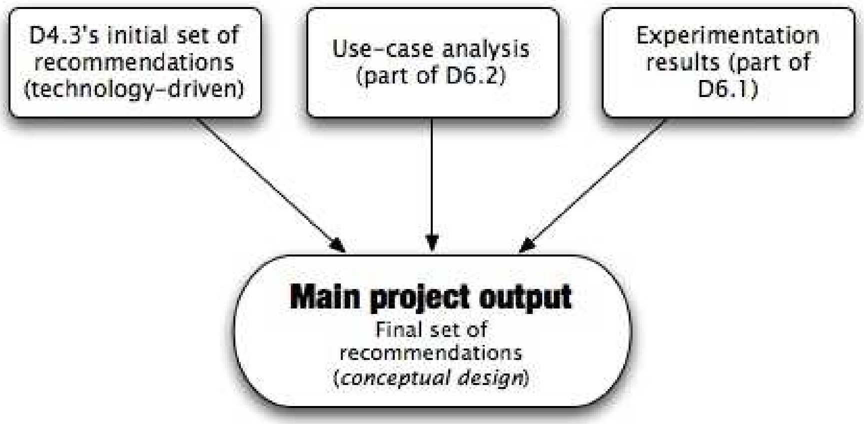

Figure 2: Specific input for final outcome

3. Use case questionnaire

In order to devise the model use-case for remote instrumentation, we built a questionnaire, which solicited descriptions of experiments from the scientists who perform them. Being aware of the difficulty of experimental scientists to familiarize themselves with the idea of a "use case", from the systems analysis point of view, we decided to make the questionnaire short and not to include terminology such as *exceptions*, *actors*, *scope*, etc. Then, based on the replies we received, we construct the model use case with a formal (normative) systems-based description.

Table 1: Questionnaire

Name:	Name of the person filling in the questionnaire. Used for contact purposes.
Email address:	Email address of the person filling in the questionnaire. Used for contact purposes.
Education:	This was a choice between BSc, MSc, PhD/Dr. Eng. It is being used to provide an understanding of the user's expertise (additional to the "Expertise" question later in the document).
Discipline:	Details about the area of expertise of the user. Used to see whether the user describes an experiment that he/she is very familiar with, and as such it is possible to solicit further/specialized information.
Level of expertise:	The user's level of expertise, according to one's own subjective estimation. Complemented by the "Education" and "Discipline" fields.
Type of instrument:	The type of the instrument being used for the experiment. Used for classification purposes.
Summary of experiment:	The summary of the experiment, providing a high-level view of the experimental steps described later on.
Experiment duration:	Used to acquire an understanding of the experiment's scale, with regard to the time it takes to complete.
"Will exclusive access to the instrument be needed?":	The reply to this question implies a need (or not) for reservation of the instrument, either in advance or immediate. As it was shown in section 6.2 of [D6.1], overall there seems to be a need for fine-grained reservation functionality when it comes to remote instruments.
Most important technical limitations:	This was the only "technical" question to the users. It was included in the questionnaire for two reasons: The first one was to see whether the user has a technical understanding of the experiment (i.e., an understanding of the overall technical requirements), in order to ask for more information, if needed. The second reason was to find out if there are any major issues to look at; usually, scientists do know about the most important infrastructure requirements (e.g., storage for large volume of produced data), even if they miss the details.
Steps of the experiment:	This was the very description of the experiment. For each

step, the following information was asked to be provided:

1. Step description
2. Input or input action
3. Expected output / tangible result
4. Is this an optional step? (Yes/No)
5. Known technical requirements
6. Would this step be impossible without physical access to the instrument?
7. Would this step be more difficult to perform without physical access to the instrument? Why?
8. If there is a collaborator on the instrument site, does your reply to the previous two questions change? How/why?

Trying to get an understanding of the various experimental steps from a systemic point of view (i.e., as a black box functionality-wise, with initial conditions, input and output) we asked for the input, either in form of materials or actions, the output - whether it is definitely included in the workflow - and about what are the preconditions that must be fulfilled before this step takes place. We saw that this phrasing and approach did not cause any significant problems to the users, who were able to provide the solicited information after minimal explanations. Question no. 6 (of the "*Steps of the experiment*" part) tries to find out whether this specific experiment cannot be implemented remotely at all. It is complemented by the following two questions: Question no. 7 tries to find out if access to the instrument is only *desired*, i.e., the results might be of inferior quality, but the experiment *can* be executed remotely; Question no. 8 explores the possibility of having this experiment run with a local collaborator. This is typical, for example, when some sample needs to be analyzed, in chemistry-related disciplines. In essence, questions 6 and 8 together help to separate the *social* from the *technical* reasons for the infeasibility of executing the experiment remotely. An example of this is the calibration of instruments: someone may refuse to allow others to calibrate the instruments before the experiment, so that one is certain that everything related to this process was done correctly; this is a "social" reason. However, for someone else this might not be an issue, so a local collaborator, who would do the calibration instead, would be enough. An additional example is the one mentioned before, with the sample analysis. Someone must insert the sample in the analyzer; so, if it is not the scientist herself, it must be some collaborator on the site of the instrument.

All questionnaire responses that were taken into consideration are available in Appendix I. An analysis follows in section 4.1.

4. Model use case

4.1. Questionnaire responses' analysis

The questionnaire was addressed to many different communities, including the people who participated in the WP2 study, the participants of the Open Grid Forum's "Remote Instrumentation Services in Grid Environment" Research Group [RISGE], the external collaborators of the consortium, etc. After filtering the responses we received to exclude those that were either irrelevant or of less significant use (e.g., duplicates), we were left with the following group of experiments:

- Evaluating the effects of noise and fading over a video transmission performed on a wireless channel
- Measurement of ultra high energetic cosmic rays with fluorescence detector telescopes and a Cherenkov detector-based ground array
- Running and processing 1D Nuclear Magnetic Resonance (NMR) spectra
- 8 bit Analog-to-Digital/Digital-to-Analog Experiment (HF)
- Acid-Base Valuation
- Determination of freezing temperature for specific types of meat
- Wireless sensor networks in indoor environment
- Global network of robotic telescopes
- Detection experiment of the type Ic supernova 2007gr
- Moessbauer spectra measurement

Thus, the distribution of disciplines is as shown in Figure 3.

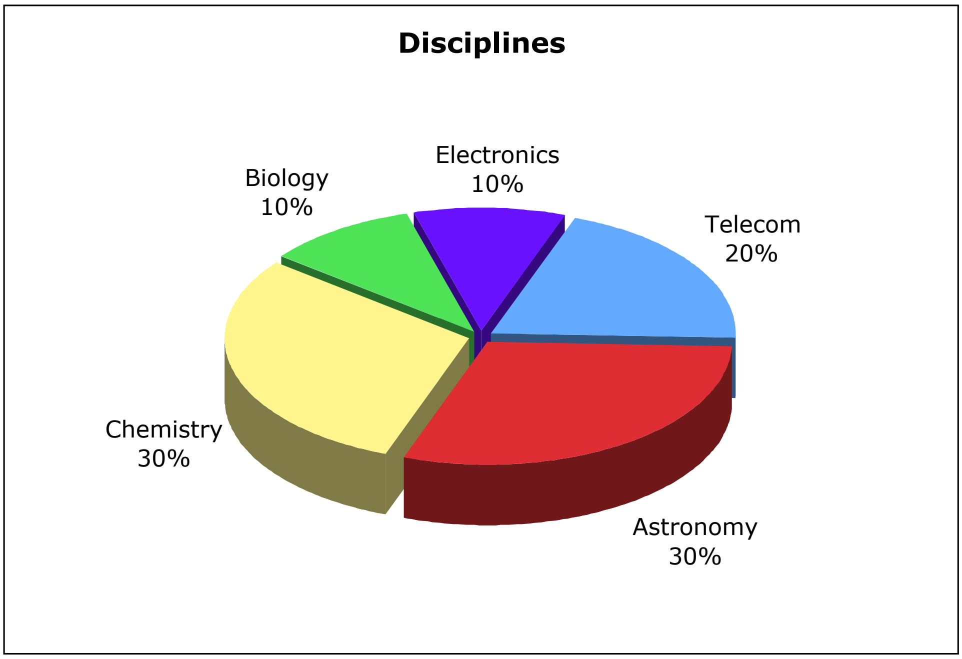

Figure 3: Distribution of disciplines for the use cases received

We performed an analysis of the responses received, trying to identify patterns in conducting (local) experiments. Then, based on these patterns, we defined a *model use case*, which is equivalent to a typical workflow for remote experiment execution; this will be normatively defined in sections 4.3 and 4.4. A less formal description follows.

In designing this model use case, we had two options as regards the approach to follow: Coming up with a "least common denominator" of the responses we received, i.e., to include only the compulsory steps which are necessary and at the very core of the experiment, or to also include optional steps. The optional steps may be completely irrelevant to a specific use case, or may be part of the experiment workflow in one instance but not in another, or they may even take place in one cycle of an experiment but not in others (within the same experiment instance). Therefore, we ended up with a *superset* of actions, relatively to the ones of the experiments described in the responses we received. This superset defines a workflow with a possible loop and plenty of optional nodes:

1. Schedule the experiment;
2. Enable the experiment (preparatory actions);
3. Define access policies;
4. Provide the input;
5. Calibrate the instrument;
6. Execute the experiment;
7. Evaluate experimental output;
8. If needed, loop to step 3;
9. Clean-up activities.

The 1st step in this workflow is to schedule the experiment. This implies advance reservation of the instrument, which is most of the times used by one user at a time. As also seen in section 6.2 of [D6.1], users do appreciate the availability of advance reservation functionality, and especially when it offers a fine-grained method for selecting timeslots for the reservations. This is an optional step and may be omitted, risking unavailability of the instrument to be used.

The 2nd step is to enable the experiment. By "enabling the experiment", we refer to preparatory actions which may have to take place before using the instrument for the experiment. Such actions are to retrieve and prepare the input for the experiment (digital data, samples, etc.), to switch on the instrument(s), to perform infrastructure checks, such as computing systems and software availability, and to configure this infrastructure appropriately in order to accept the data produced. This is an optional step, as it may be the case that such actions are not needed at all, or that they are being taken care of by specialized technicians, so that scientists do not need to mind that.

The 3rd step is the definition of access policies. This is to allow specific people to attend the experiment, either on-site, or through some video-conferencing facility transmitting still/moving images. When equipment control is relevant, access policies refer to granting permission to specific people to control the experiment. This is an optional step.

Providing the necessary input to the instrument is the next (4th) step. This is an optional step, as in certain cases no input is required for a specific experimental step. The input may be a real substance (as in the chemistry and biology cases of our responses), a definition file (as in the astronomy/astrophysics use cases), or some other kind of digital information.

After providing the input, a calibration of the instrument may follow as the 5th step. By calibration of the instrument, we mean the adjustment of its properties in such a way to minimize the expected error margin and eventually retrieve as accurate results as possible. The calibration is not necessarily a single-step process, but may require consecutive runs before the desired setting is achieved. This is handled by the loop that will be discussed further on.

The following (6th) step is the actual execution of the experiment, and this is the first compulsory step in this workflow. Controlling the instrument to perform measurements and monitor the relevant parameters is part of this activity; an implicit parallel activity is the possible storage of the output to the pre-defined data sinks. Such a definition may have taken place in the 2nd step (enabling the experiment), or may be hard-coded in the application and the equipment.

The 7th step, again an obligatory one, is to evaluate the output. Evaluation may have many different facets, including visualization, some scientific calculation leading to assertions, further simulations and analysis of the results, macro- and microscopic examination of samples when that applies, etc.

After completing the above optional and obligatory steps, an optional loop to step 3 (definition of access policies) may take place. This is useful for the calibration phase, or in case the experiment failed for any reason. Additionally, certain experiments are inherently multi-run ones, so multiple cycles are required anyway.

Finally, a cleanup may be necessary, according to site policies and rules. This may include switching off instruments, some kind of other maintenance, deletion of input and output, or other activities, which prepare the ground for the next person to perform an experiment. This is, again, an optional step.

In the case of multiple runs (active loop) that have produced data to be correlated and post-processed in some lengthy process later on, we do not consider this to be part of the experiment itself. As such, we have included only the evaluation of each run, which is taking place during the experiment lifecycle and before cleanup and release of the instrument to other users.

With the description elaborated so far, we have managed to decouple the experiments from their domain. This is a very important result in the process of understanding the requirements for remote instrumentation services. When transferring these domain-unspecific activities to the remote-instrumentation scope, a large set of requirements can be deduced. One or more of the following must be available, depending on the specific use case:

- *Selection services* (scheduling and reservation, also including computational, storage and networking resources) – Optional;
- *Input management* (identification, preparation processes) – Optional;
- *Instrument virtualization / Service provider* (representation as manageable resource) – Obligatory;
- *Policy decision and enforcement* – Optional;
- *Data management of input* – Optional, applies to digital input only;
- *Local operator* – Optional if remote control of instruments is possible. Can be assistive to input management as well;

- *Visualization devices* – Optional;
- *Software for post-processing* – Optional;
- *Workflow management* – Optional;
- *Accounting and Monitoring* – Obligatory;

Subsection 4.2 elaborates a little further on the concepts represented by the above requirements. Section 5 will summarize results from WP4; then section 6 will combine the two, along with results from prototype experiments described in section 6.2 of [D6.1], to provide the final list of recommendations.

4.2. Requirements resulting from the use case analysis

4.2.1. Selection services

Selection services refer to those services that are responsible for discovering candidate resources to execute a task on, allocate them (and reserve them in advance, if necessary), and then schedule jobs on them. The selection services are at the core of the Grid, offering the "resource-agnostic" part of it and the abstraction that the Grid represents. Resources can be any manageable entity, such as computing, storage, networking, data, and of course instruments.

4.2.2. Input management (identification, preparation processes)

As we saw, most instruments (and, as such, remote experiments) will need some kind of input in order to be invoked. Depending on the kind of the discipline and the experiment, the input can be either a substance or of digital nature. Chemistry will (by nature) use some sort of a sample that must be inserted into a microscope, spectrometer, or other such instrument. This sample will have to be prepared accordingly for the experiment, probably through pre-processing. Conversely, astronomy experiments may use a Remote Telescope Markup Language (RTML) input file that describes what to observe and for how long.
The input to be used must be first identified (i.e., selecting suitable input for the experiment), then built if needed, and finally prepared through appropriate pre-processing.

4.2.3. Instrument virtualization / Service provider

Instruments must be monitored and controlled, just like any other resource on the Grid. Similarly to other resources, there is the need for a service provider, which will abstract the instrument in such a way that any and all kinds of devices offer at least some common functionality for control and monitoring. The domain-specific steering of an instrument must be somehow implemented through the relevant Application Programming Interfaces (APIs), but the basic functionality that allows invoke this very domain-specific functions must be made available. In this regard, we refer to making the instruments a first-class Grid (and Service Oriented Architecture (SOA), more generally) citizen through such a service provider, which virtualizes them and allows their use from a distance.

4.2.4. Policy decision and enforcement

The issue of *policies* is central to services and SOAs. By "policy" here we refer to anything regarding access to resources and services. As such, *authentication* and *authorization* are part of this functionality. By "authentication" we refer to the process, which can confirm the identity of a user or an agent acting on the user's behalf. By "authorization" we refer to the process, which can confirm whether the user/agent is allowed or not to access certain functionality. Authentication is quite clear as a concept and is already widely deployed on the Grid as a basic, essential service, with the most common mechanism being X.509 certificates [PKIX]. However, authorization is a more complex issue and applies to many different facets of a Remote Instrumentation system. Access to any of the site resources may or may not be granted to someone based on system load, business rules, information confidentiality, site security, licensing issues, or other.

4.2.5. Data management of input

In section 4.2.2. we discussed the issue of experiment's input and input preparation. We also referred to those cases where the input is of digital nature. When this is the case, the management and handling of this input becomes relevant. This has to do with questions like:

- Where is the data available?
- How will it be fed to the instrument?
- Which are the possible privacy constraints related to this input? On which level should the related policy choices be applied?
- Is this data static, or should it be dynamically updated during the experiment?

This is a non-exhaustive list. In short, one can say that "data management of input" refers to all those aspects of digital input that do not refer directly to its creation and preparation.

4.2.6. Local operator

A local operator may or may not be needed, depending on the domain and the experiment itself. When there is a sample to be analyzed somehow, there must be someone who will insert the sample into the instrument. That is to say, under certain circumstances, physical access to the instrument is a strict requirement, and therefore a local collaborator/operator must be present and available. It may also be the case that the local contact is responsible for ensuring operational correctness by switching on and off the instrument(s), if that cannot be performed programmatically.

4.2.7. Visualization devices

Visualization is typically a very important part of experimental science. In a widely cited paper from 1988, the author writes: "The list of research opportunities for visualization in scientific computing is long and spans all of contemporary scientific endeavor" [MCCORMICK], then moving on to mention "a select sampling of advanced scientific and engineering applications", which have advanced visualization requirements: molecular modelling, medical imaging, brain structure and function, mathematics, geosciences, space exploration, astrophysics, computational fluid dynamics and finite element analysis. The visualization requirements of an application may vary widely, from simple histograms to 3D representations of extreme resolution needs (which also implies very demanding network requirements). For instance, in [LASZEWSKI], the authors refer to "three-dimensional (3-D) raw data with spatial resolution of as little as 1 μm" (the authors' list of this paper includes Ian Foster and Carl Kesselman, the founders of the Grid concept).

4.2.8. Software for post-processing

As will also be shown later in this document, certain post-processing of experimental results is almost always expected to take place. This post-processing may take many different forms, but in all cases (especially when *remote* instrumentation is involved), it will be based on software. As such, appropriate software to perform this post-processing is needed. At this point, we do not concern ourselves with licensing issues. The technical correctness and completeness of the software must be confirmed, through proper installation procedures and the site functional tests.

4.2.9. Workflow management

When moving experimental science to a service-oriented context, the capability to compose services into new applications is directly implied. This composition suggests the use of a workflow management system, where a workflow (also termed "*process*" sometimes), models a scientific application. There are many different such workflow management systems nowadays, but the Grid poses a new requirement, which is not satisfied in full yet, and as such it is still a research topic: that is the need for QoS-aware workflows. In other words, it must be possible to provide QoS guarantees for complete workflows, through the composition of the QoS of services that make it up.

4.2.10. Accounting and monitoring

Monitoring is at the core of Grid computing, along with security and execution management. These two capabilities allow the Grid to become production-level, by enabling the tracking of problems and the logging of usage. Additionally, monitoring/accounting information allows choosing resources based on their utilization, which makes the Grid more efficient. Thus, it is necessary to have a monitoring framework that produces *alarms* to send to the operators and *operational data* to be sent to the accounting framework.

When dealing with instruments, especially with ones that are rare/expensive, their operational safety is of utmost importance. Therefore, correct operation must be monitored at all times, and all errors must be logged. Instrument operators will then be able to use relevant data, possibly correlate it with other monitoring information, and identify the source of any such problems. Additionally to the above, error monitoring and logging is very important in order to ensure proper execution and accurate results for the experiments; an error going unnoticed may cause large deviations and produce wrong output that should not be used academically or otherwise.

Site functional tests are a part of the monitoring infrastructure and refer to those facilities that allow someone ensure operational completeness and correctness for a complete experimental site. When computing, storage, or other resources are part of an experiment, as well as additional software, it may be necessary to check them before the experiment to be certain that there will not be a problem *during* the experimentation phase. Typically, these tests will use simple but proven methods to verify that networks are connected, batch schedulers are available and accept tasks, storage is also available, and that of course instruments are operational and can be used at the time. Software existence checks are also in scope, to verify that post-processing will be possible. Data issues are not to be handled by site functional tests, as they fall under input preparation and management.

With regard to accounting itself, it is needed to be able to evaluate the use of shared instruments (which is of interest to the stakeholders and the funding entities), but also to be able to establish SLAs, provided there exists a framework for that. Metrics such as availability of instruments and their performance for reference operations, can be used for this purpose.

4.3. Textual description

The use case format adopted below is derived from Alistair Cockburn's "Writing Effective Use Cases" [COCKBURN]. In this description, we do NOT repeat the explanations provided in section 4.1 for each consecutive step of this experimental workflow. Additionally, the workflow described here does not take into account issues relevant to post-processing of output data after multiple loops and end of the experiment, as mentioned before, as well as policy (which is not access-related, e.g., Acceptable Usage Policy), or financial/logistical issues.

Primary Actor:
The scientist performing the remote experiment.

Secondary actors:
- On-site equipment operator;
- Infrastructure and instrument technicians/maintenance engineer;
- Site security manager;
- Input provider (if any);
- Instrument/infrastructure owner.

Scope:
The Remote Instrumentation Infrastructure (RII), as composed by:
- The instrument(s)
- The network
- The computing and storage resources
- The enabling middleware and the post-processing software

Stakeholders and interests:
- The scientist performing the experiment has an interest in concluding it successfully and retrieve reliable measurements (or other results). This execution of the experiment must be as economic and technically complete as possible. Additionally, the scientist has an interest performing the experiment in a secure environment, which guarantees that access to results (but possibly also the input) is restrained to well-specified entities.
- Owner of the equipment and infrastructure. The owner has an interest in ensuring that the equipment is working properly and is not damaged as a consequence of the experiment. If the owner is also the manufacturer (possible in the case that vendors are leasing their equipment or providing it for testing purposes), the owner needs to ensure that the performance of the equipment is appropriate, in order to improve the equipment’s reputation and therefore gain a competitive advantage. Finally, if the equipment is rented and there are such financial earnings, the owner has an interest in maximizing use of the equipment through efficient scheduling of the experiments.
- Infrastructure and instrument technicians/maintenance engineer. The engineers of the infrastructure and of the instrument equipment have an interest in ensuring the proper operation of both, for reasons of professional reputation.
- Site security manager. The site security manager has an interest in ensuring that any access policy changes on the network and the software infrastructure will not affect

negatively the overall site security, and will not increase an intruder's chances to access the site.

- Input provider (if any). The quality of the input may affect an experiment, when input is relevant. If the provider of the input is an external entity that provides the input in a commercial context, they have an interest in improving their reputation through quick delivery of quality input and successful completion of experiments.

Preconditions:

- The RII (as defined in *Scope*) must be operational.
- The user has appropriate access rights to the infrastructure.
- When exclusive access is required, or there is a restriction in the number of concurrent users, the infrastructure must offer this functionality.
- It must be possible to verify at all times that the RII is functioning properly, either remotely or with the assistance of a local collaborator.
- It must be possible to define access policies that restrict access to some or all of the experimental process and results.
- If remote control of the instrument is not possible, a local collaborator must be on site.

Minimal guarantee:

- The RII must remain operational, without the security measures being affected in any way.

Success guarantee:

- The experiment was conducted in full; partial instrument output has been obtained and post-processed to extract the final output data.

Main success scenario:

1. If needed, schedule the experiment for a specific timeslot, indicating exclusive access or concurrent access limitations.
2. If needed, prepare the input.
3. If needed, switch on instruments, check and prepare processing infrastructure.
4. If needed, define access policies for the access of additional entities to the experiment and the output produced.
5. If needed, provide the input to the instrument (directly or through a local assistant).
6. If needed, calibrate the instrument, by modifying its settings and parameters.
7. Execute the experiment performing measurements and control of the instrument.
8. Evaluate the output of the specific measurement.
9. If additional runs are required, jump to step 4 and repeat onwards.
10. If needed, perform cleanup activities, such as deleting intermediate data and switching off the instrument and other relevant equipment.

Extensions:

1a. The instrument is not available for exclusive access at that time and day, or maximum number of concurrent users has been reached.
    1a1. Try another day and/or time.
2a. Input is not of acceptable quality or has been destroyed.
    2a1. Contact input provider and receive new input.
    2a2. Reschedule experiment.
3a. Instrument or other part of the infrastructure is not available for technical reasons.

3a1. Contact corresponding engineers to fix the problem(s).
3a2. If fixing the problem is a lengthy process, reschedule experiment.
4a. It is not possible for the user to define access policies.
4a1. Contact the site security managers and ask them to append new policies.
4a2. If addition of new policies is a lengthy process, reschedule experiment.
5a. It is not possible to provide the (digital) input to the instrument.
5a1. Contact the site infrastructure engineers.
5a2. If fixing the problem is a lengthy process, reschedule experiment.
5b. It is not possible to collaborate efficiently with the local assistant.
5b1. Contact instrument owner to arrange for another local collaborator.
5b2. Reschedule experiment according to availability of alternative collaborator.
6a. Calibration of the instrument fails.
6a1. Contact the site infrastructure engineers.
6a2. If fixing the problem is a lengthy process, reschedule experiment.
7a. The instrument or other part of the infrastructure is malfunctioning during experimentation.
7a1. Contact the site infrastructure engineers.
7a2. If fixing the problem is a lengthy process, reschedule experiment.
9a. The instrument cannot be switched off, or other technical problem has occurred.
9a1. Contact site infrastructure engineers.

4.4. UML and Flowchart representation

The model use case (workflow) defined above is represented below schematically. We are using Unified Modelling Language (UML) Use Case Diagram notation in Figure 4, UML Activity Diagram notation in Figure 5, and Flowchart notation in Figure 6.

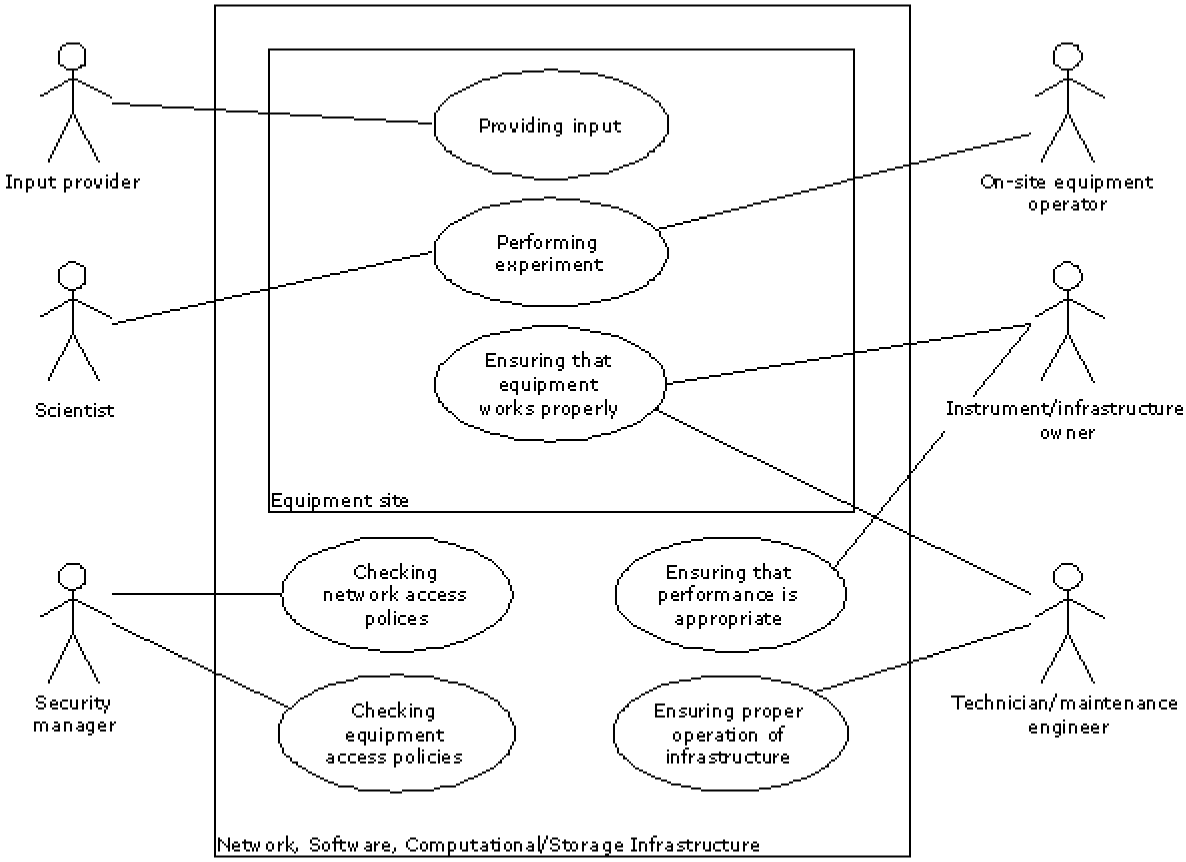

Figure 4: Use Case diagram of model workflow

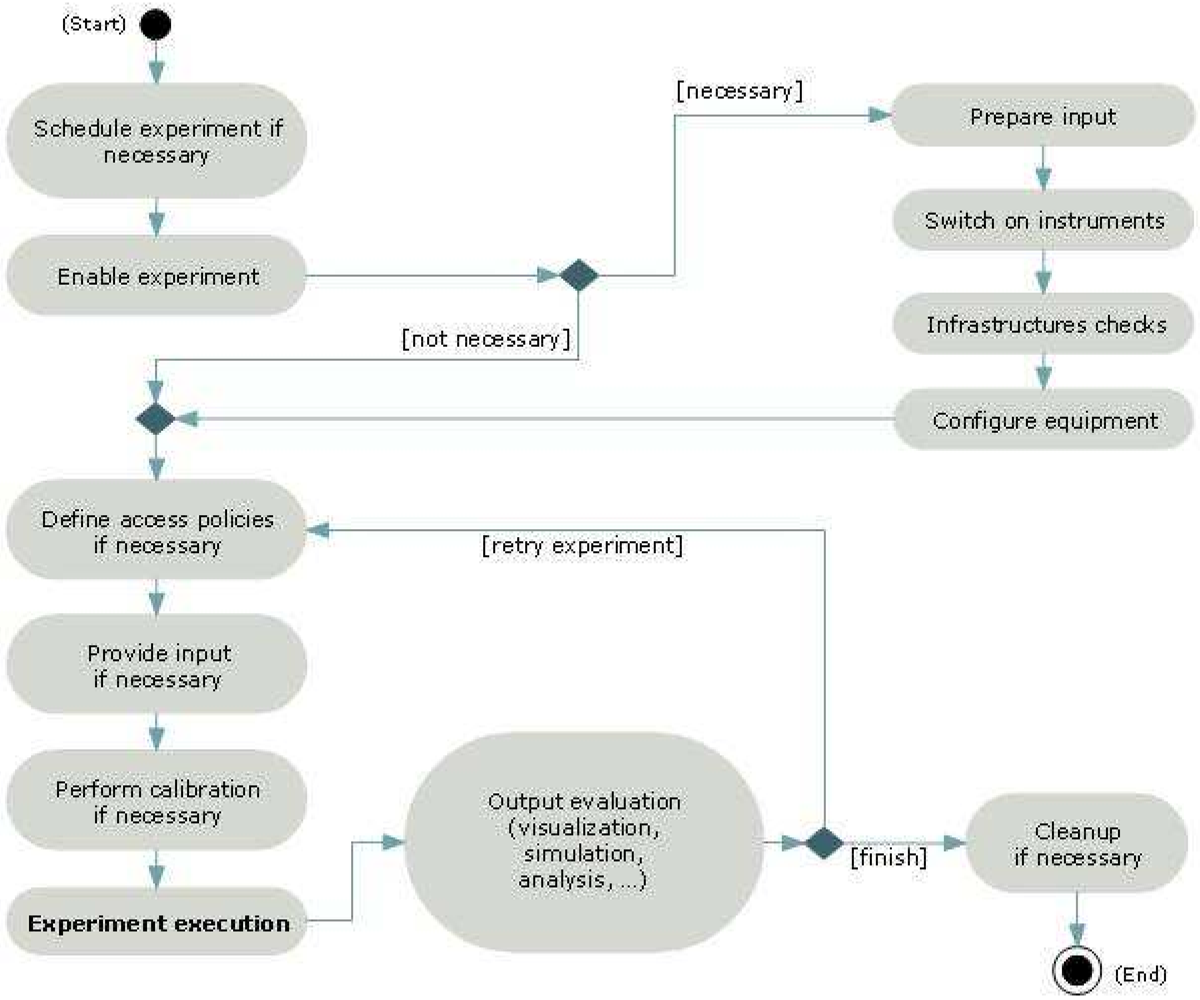

Figure 5: Activity Diagram for the model use case

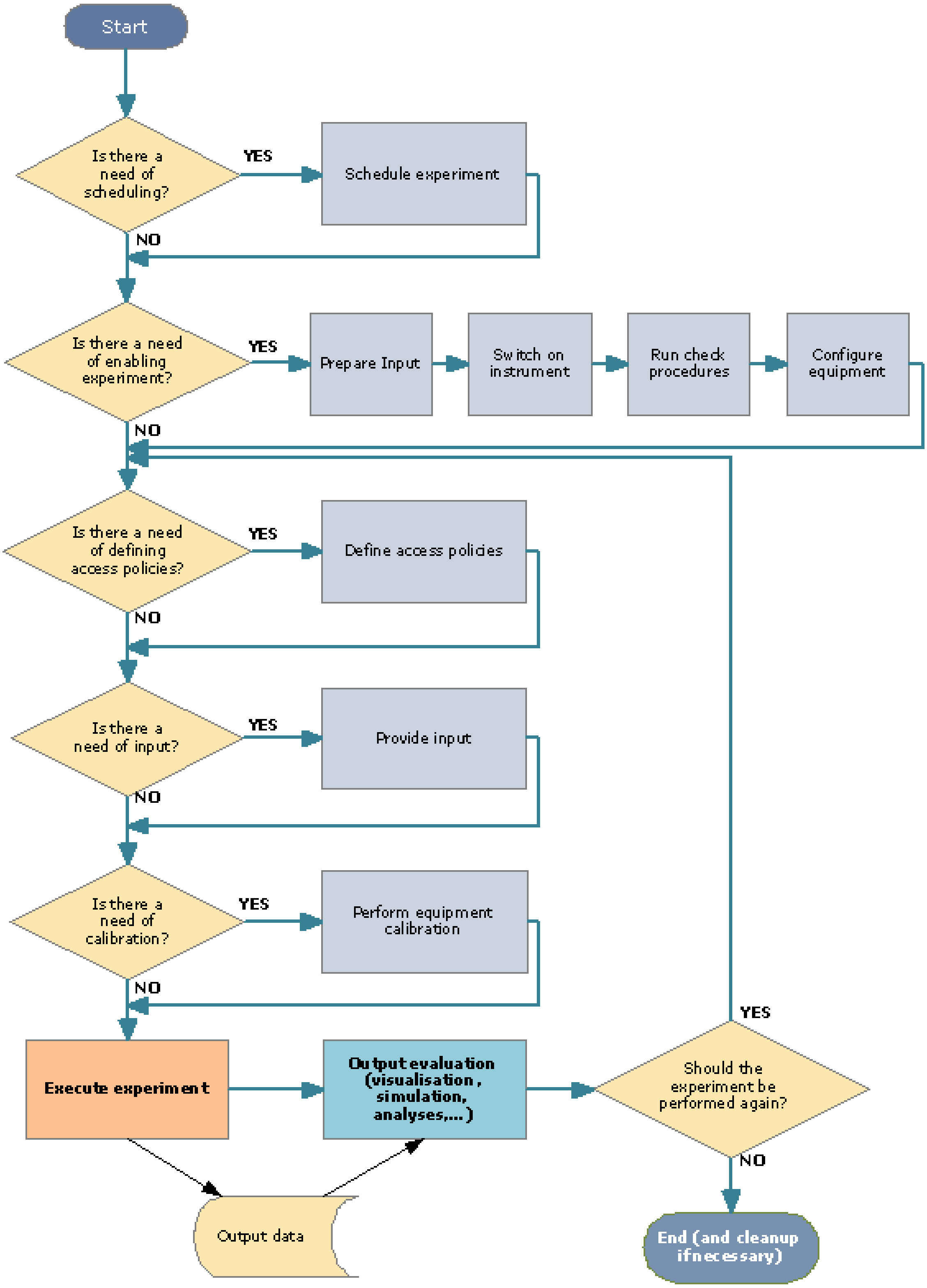

Figure 6: Flowchart of model use case

5. Recommendations from D4.3

D4.3 of the RINGrid project presents the work carried out in the WP4 workpackage. It starts out with a first set of conclusions, by examining technology as a driving element, and then deriving a set of recommendations from them. These recommendations will then be extended in the following chapter.

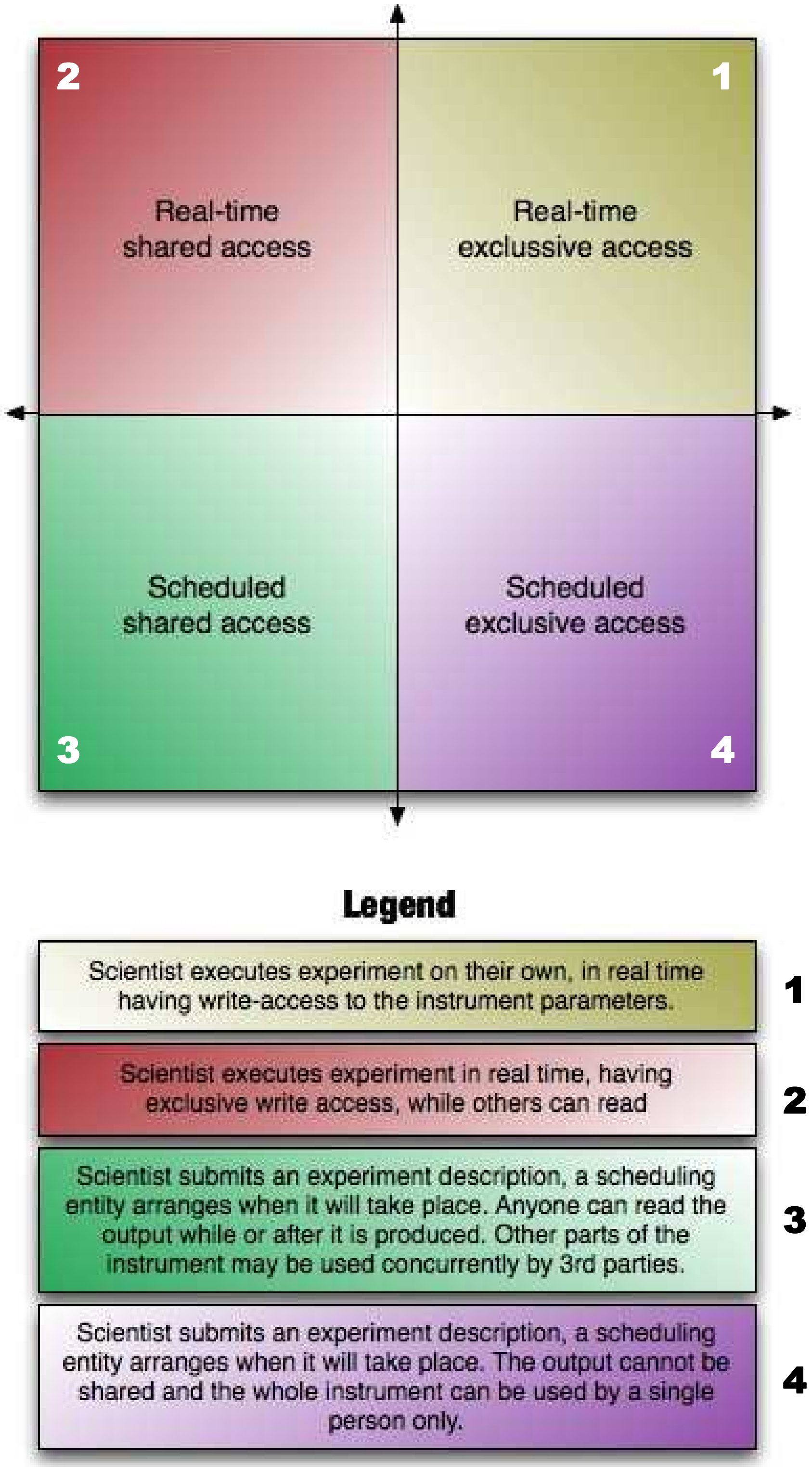

Figure 7: Categorization of experiments

This set of recommendations is based on a three-tier model, dividing the architectural elements into a *physical instrumentation domain*, an *abstract instrumentation domain* and an *experiment execution domain*. This division is similar to what the OGSA model architecture suggests. But, more important, it takes into account the requirements of remote experiments.

Experiments can be divided into four categories, as shown in Figure 7. Apart from this classification, there is also a difference between self-contained experiments (experiments with a clearly detectable start and end time) and always-on experiments, which will produce data regardless whether this data will be monitored or not.

From these classifications and from the status of the technology, recommendations are derived in D4.3. We start by describing the recommendations for the physical instrumentation domain, followed by the other two domains, as outlined above. For all points made in the rest of this section, the reader is encouraged to go to [D4.3] for more details and an elaboration of each requirement.

5.1. Recommendations for the Physical and the Abstract Instrumentation Domains

5.1.1. Networking recommendations

On the network level, there are a number of ground-level requirements for which the technology is already widespread:

- High throughput: bulk data transfer sometimes needs to be as fast as 1 Gbps. QoS guarantees need to be considered;
- Performance controllability: access to VPN facilities needs to be ensured;
- Network resource reservation capability;
- Security controllability;
- High availability;
- Multicast.

Overall, network-level QoS support offers functionality that is necessary to RIS and we believe the community should prioritize its further research. QoS support includes support for delay guarantees, adaptability, dynamicity, scalability, heterogeneity, as well as the ability to span different administrative domains while retaining complete support for different QoS levels. One relevant technology that still has not taken off, but may offer significant advances, is Bandwidth on Demand (BoD). Another technology, whose application must be investigated in this context, is IPv6; IPv6 offers significant advantages, also to overcome old solutions (such as NAT) to large numbers of manageable resources with assigned IP addresses.

Finally, attention should also be paid to Access Networks, both in respect to mobility as well as to scalable terminal access to final users. Although we have not explicitly looked into this subject, it is the case that in poorer areas of the world, satellite links could offer an alternative to the expensive ground connections, and thus offer the means to overcome the digital divide in this respect. Through carefully engineered links, to overcome the typical problems of satellite connections, it will be possible for scientists in these areas of the world to take advantage of RIS for their experiments.

5.1.2. Information exchange and representation

Producer/consumer models are important in distributing real-time experimental data. These models can be of great usefulness when asynchronous information exchange is needed, for instance when dealing with multiple measurements that have to be transmitted to the user in real-time for visualization purposes.

Data visualization is one more topic to be considered in the physical instrumentation domain. For data visualization there already exists a viable and satisfying approach covering most data types. We only see problems with interactivity, with the visualization of widgets, and with the need for much higher resolution visualization.

5.1.3. Instruments as Grid resources

Instruments need interoperability among diverse, heterogeneous and distributed resources and services provided by all kinds of scientific equipment. As a consequence, the design of the grid architecture needs to be extended so that instruments become a first-class member of the grid infrastructure. Therefore, the following requirements of remote experiments need to be addressed for instruments, just like it is possible for computing and storage:

- Resource discovery, query and scheduling (through Information Services, Execution Management Services, and Selection Services);
- Resource virtualization using the appropriate Service Providers;
- Resource sharing (similarly to the sharing of computational and storage resources, based on the VO concept).

The points above are direct analogies from the traditional Grid concepts. Additionally for instruments, however, licensing issues (for instrument drivers and support software) must be resolved. It is prohibitively expensive to buy a separate license for every user who wants to access an instrument remotely. However, most licensing models for software controlling expensive instruments are based on a specific workstation with a specific network card. So, some form of proxy is needed to relay the commands from the user's workstation to the workstation directly connected to the instrument(s). The instrument’s Service Provider can play the role of this proxy, which is the module to abstract the instrument appropriately and offer its functionality for consumption over the Grid. The basic question about this module is whether it should offer a uniform interface for any instrument, or if it should modify its interface depending on the instrument at hand. The first approach is better from a standardization and interoperability point of view, but the latter may offer more expressive constructs and provide actual *meaning* to instrument representations.

Additionally to the points made above, QoS provisioning from the resources themselves is of great importance to RIS. Establishment and monitoring of SLAs with resource brokers through corresponding reservation languages is necessary, not only for instruments but also for other kinds of resources. Bandwidth on Demand is one such case being heavily researched in recent years, but without substantial uptake. Such QoS guarantees from the resources themselves will create the foundation on which QoS-enabled workflows will be built.

5.2. Recommendations for the Experiment Execution Domain

This is the domain that the user of an instrument deals with. The main problem in the abstract instrumentation domain is that no standardized user interface exists. Currently, all experiments conducted remotely use an interface that is tailored towards a specific instrument or use case. So, a necessary task would be the creation of a standardized interface. An important recommendation for such a user interface would be to require it to be self-documenting and easily accessible even for untrained personnel. The user interface needs to provide the following monitoring and healing functionality:

- Satisfy the need to monitor the nodes and detect failures, so that the user can react properly;
- Recognize that data and code synchronization after a failure has occurred and has been dealt with;
- Automatically renegotiate client-server connections after a failure.

Therefore, the interface should mask the complexity of the whole system in order not to overburden the user.

This also applies to the user being able to easily define a workflow for his/her experiment, independently of the eScience area and the specific discipline involved. Thus, the workflow description methodology is one of the key aspects in this domain. We believe that semantic descriptions of instrument resources can help in this regard, through the definition of suitable ontologies and hierarchical classifications, in terms of successive aggregations of related objects. Workflows should, additionally, be extensible by adding new functionality, for example to choose automatically the best flow routing based on resource monitoring and discovery. This and other performance-related terms can be universally addressed by workflow management systems that can understand and use non-functional QoS terms, as defined by the user. Intuitive GUIs for workflow execution monitoring is just as important.

6. Final list of recommendations for Remote Instrumentation Services

The purpose of this section is to elaborate and further refine the main guidelines that were outlined in section 5 of [D4.3].

There are basically three sources of information at this point, upon which we can draw our recommendations for further development and possible standardization activities (e.g., within the RISGE group in OGF):

- Considerations about the current status and the foreseeable evolution of technology in middleware and networking, which constituted the basis for our analysis in section 5 of [D4.3];
- The general Remote Instrumentation Use Case, which was only sketched in D4.3, and formally defined in this document, stemming from both general considerations and the responses from direct user experience, which we collected and elaborated upon;
- The results of direct user experience with a diversified (though limited) number of Remote Instrumentation platforms, whose output was collected, and analyzed in section 6.2 of [D6.1].

In general, it can be noted that, as was indeed expected, our use case formalization and data collection from users' experience do not contradict the observations that could be made and the conclusions that could be drawn on a purely technological basis. Rather, we can now further refine and complement such conclusions. Table 2 consolidates the results of our study, into a single table of requirements for RIS. The next sections elaborate on these requirements, to end with an identification of OGSA gaps and suggestions for future directions.

Table 2: Consolidated view of final recommendations

Category	Requirement	D4.3 results	Experimentation	Use case analysis
Network	High throughput	x	x	x
	QoS guarantees (especially on access networks)	x	x	
	Performance monitoring and control	x	x	x
	Virtual Private Networks	x		
	Ability to reserve network resources / Bandwidth on Demand	x		x
	Control of security parameters (policy decision and enforcement)	x	x	x
	High availability	x		x
	Multicast	x		
	Use of IPv6	x		
	Network transparency of applications (e.g., firewall independence)		x	
Information management	Widespread use of publish/subscribe mechanisms	x		
	Visualization of complex data types	x		x
	Instrument resource discovery	x		x
	Instrument resource query	x	x	x
	Input management (data)			x
	Advanced accounting and monitoring facilities			x

Resource management and Scheduling	Virtualization of instrument resources	x	x	x
	(Fine-grained, user-controllable) Scheduling of instrument resources	x	x	x
	Sharing of instrument resources	x		x
	Service-related QoS provisioning	x		
	Establishment and monitoring of Service Level Agreements	x		x
	Checkpointing and recovery mechanisms	x		x
Interfaces and Visualization	Interactivity	x	x	x
	Variety of visualization widgets	x	x	x
	High resolution visualization	x		x
	Standardized interfaces for remote instruments and experiments	x	x	x
	Self-documenting, low learning curve interfaces	x	x	
	Node monitoring and failure detection	x		
	Localization of interfaces		x	
	Easy workflow definition	x	x	
	Intuitive workflow execution monitoring	x	x	
	Experiments based on templates, repository of experiments		x	
	Generic visual interfaces for instruments		x	
	Platform-independence of client-side interfaces		x	
System-side properties	Workflow management: Dynamic workflows with task re-routing and QoS provisioning	x		x
	Modular, flexible, standardized systems which adapt to the user and the application		x	x
Various issues	Licensing issues on driver and post-processing software	x	x	x
	Presence of on-site operator		x	x
	Input management (materials)		x	x

6.1. Requirements stemming from the Remote Instrumentation Use Case

All functionalities previously outlined in D4.3, organized in the light of the three-tier OGSA architectural model, are indeed required to address the multiplicity of scenarios stemming from the general Remote Instrumentation Use Case. Actually, most functional requirements in the list we have given in Section 4.1 further specify aspects that belong to the Abstract Instrumentation Domain. **The fact that most of them are optional should not appear as reductive**. In the attempt to include the generality of Remote Instrumentation instances in our use case, we have reduced the common denominator of all experimental activities characterized by the fact of being remotely executed to a very small subset. This is not surprising, since we are addressing all kinds of experimental activities, ranging over such different domains as large physics experiments, instrumentation and measurement in different fields of engineering, environmental monitoring, and so on. However, precisely for the same reason, requirements we have classified as *optional* may arise in a very large number of specific instances of Remote Instrumentation.

We recognize three functional requirements in the Abstract Instrumentation Domain as **common to all types of remote experiments**:

- Instrument virtualization
- Error monitoring and logging
- Accounting and monitoring/controlling experiment execution

They belong to the classes of Execution Management and Resource Management Services and are described in further detail in subsection 4.2. Among these, *Instrument Virtualization* or, in other words, the definition of appropriate *Service Providers* for Remote Instrumentation is of utmost importance and characterizes Remote Instrumentation Services with respect to other service types. This is an area where standardization work is certainly needed and where, most likely, a compromise between the generality of the interfaces and their specialization to instrumentation in different fields should be sought, in the light of both end-users' and application developers' needs.

Other functional requirements in the Abstract Instrumentation Domain that appear from our Use Case are:

- The presence of Selection services
- Site functional tests
- Policy decision and enforcement (including Security)
- Input and output (software post-processing) data management
- Workflow management

All these can be variously classified into the categories of Execution, Resource, Data and Information Management. As we have seen in D4.3, though all these functionalities exist in current Grid middleware, there is ample space for specialization to the Remote Instrumentation needs. In particular, we recall here the importance of QoS-aware workflows, producer/consumer models for efficient data transfer, cross-domain QoS-mapping and exchange of information for control purposes. The interaction with security policies has proved to be an important point in all of our service testing, in relation to ease of use and user's satisfaction, and it will be touched upon also in the next sub-section, dealing with these aspects.

Finally, there are some non-functional requirements that are of importance to Remote Instrumentation, as stemming from our use case description:

- Input management
- The presence of (and interaction with) local operator(s)
- The presence of specialized visualization devices (and the corresponding software for data representation – which pertains essentially to the local host, but may have the need to interact with data services).

In particular, the possible need of interaction with a local operator is peculiar to a number of instances of Remote Instrumentation Services (e.g., in material science), and calls for the increased integration between general collaborative tools and user interfaces for Remote Instrumentation (as, e.g., in the VCR concept described in previous deliverables – Section 4 of [D4.2], among others).

6.2. Requirements expressed by users' direct experience

In essence, our testing with user groups on specific platforms – though necessarily limited by the resources available to the present project – has confirmed some of our previous conclusions, regarding aspects that pertain to all our three domains.

We have already mentioned a point about security, which belongs to the Abstract Instrumentation Domain. In almost all cases, initial impairments caused by the interaction with local security policies had to be faced and solved before running the experiment. Thus, it is quite relevant for the real diffusion of Remote Instrumentation Services that Security Services be available to a large extent, in standard form, and easily implementable within the platform in use.

Almost all other conclusions that can be drawn from our limited user feedback pertain to Networking (Physical Instrumentation Domain) and Experiment Execution.

6.2.1. Networking

As regards Networking, it turns out that uninterrupted and reliable (in terms of service continuity and ease of access, not of data transfer reliability) network service is a strong and basic requirement, which is not yet fulfilled by the current Internet at large. All experiments we conducted involved interaction across continents, with the sole support of Internet access provided by NRENs, without any specialized reservation or agreement. Though this does not totally impede the remote experiment execution, it certainly hinders a proper diffusion of the practice. This is a primary need, which comes much earlier than other specialized networking services we have discussed, like Bandwidth on Demand and QoS based on specific SLAs.

The latter points, though, deserve some further comments based on our previous technology investigation, on the output of previous projects, and on our own networking experience. There are several approaches to providing QoS in IP networks:

- "Throwing bandwidth at the problem" (over-provisioning)
- QoS-IP (IntServ, DiffServ)
- MPLS
- GMPLS

Over-provisioning, which essentially guarantees a lightly loaded network even in the worst traffic conditions, is currently the most widely used solution by Internet Service Providers (ISPs), including National Research Networks (NRENs). There has indeed been a long debate in the scientific community on the real opportunity of QoS-IP. However, it should be noted that, though bandwidth may be abundant in the core network, there are still limitations in the access networks, be they cabled or wireless, that may call for QoS control approaches in order to guarantee low levels of latency and packet losses. Since the IntServ solution suffers from scalability problems, the DiffServ approach, which offers the possibility of guaranteeing resources to traffic aggregates, appears to be a more sensible way to deliver QoS. Indeed, the GEANT network provides such possibility to differentiate traffic, by offering Premium IP, traditional best-effort, and less-than-best-effort service classes. MPLS, through the concept of core-edge networks, gives one of the means of implementing differentiated services, by handling flows and offering the capability of switching them, while at the same time reserving the necessary resources. Rather than the generalized use of QoS-IP, an interesting concept to be fostered is that of the creation of Virtual Private Networks (VPNs), which can offer QoS support where and to the extent at which it is needed and where Traffic Engineering concepts can be applied on a smaller scale (see, e.g., [BBD06] and [BBDR08] and references therein for a discussion on planning and real-time control, respectively, of IP VPNs).

All previously mentioned technologies operate in the user plane of the network. GMPLS is a control-plane technology that allows generalizing the resource allocation concept, by extending manageable resources irrespectively of their nature (be they optical wavelengths, radio frequency bands, or time slots) and allowing further flexibility. It is very interesting, in this respect, to look at the dynamic provisioning of essentially circuit-switched high-speed connections over datagram networks, capable of providing a true BoD functionality (a recent overview of such implementations in support of Grid applications can be found in [KEJ06]).

The Remote Instrumentation instantiation of the SOA comprises such a wide range of cases and scenarios that it does not really make sense to strictly recommend specific networking requirements in support of Remote Instrumentation *tout court*. Rather, as in any approach that aims at being open, non-restrictive, and flexible, one should be prepared to scale the applications to the capabilities of the underlying networking platform, if needed, and to enhance them to exploit all the potentialities offered by the network, whenever present. This can be achieved through cross-domain optimized approaches, where QoS-aware workflows in the Abstract Instrumentation Domain match SLAs that translate the application requirements into QoS-networking; at the same time, they can receive feedback from the network in order to adapt to possible shortcomings, or signal the user that a certain quality level cannot be satisfied, so that the user can possibly choose a different course of action.

6.2.2. Experiment Execution

The most interesting conclusions, however, come from user observations that touch the sphere of Experiment Execution. It is apparent that the existing platforms we have tried all have some limitations in this respect. There is the need for:

- *Flexibility and adaptation to the user-specific experimental environment*. This point touches the Experiment Execution Domain, and pertains to the capability of customizing a Remote Instrumentation platform involving a certain type of devices to perform a specific experiment. There are at least two possible ways, differing in their level of complexity regarding their implementation, in which this task can be accomplished.
    - *Availability of experiment menus to choose from*. For instance, in our testing experience with the Device Farm of telecommunication measurement systems, a user clearly pointed out the opportunity to have a menu of available experiments that can be performed, from where to choose. This is one possibility, which was exploited in previous, non Grid-based projects (e.g., the LABNET project [DAV06]).
    - *Experiment configuration and composition*. Much more could be done by exploiting the composition and orchestration functionalities that would be offered by the Experiment Execution Domain. This is a subject that requires further investigation.
- *Flexibility and adaptation of the user interface to the user's skills and level of expertise*. This is another requirement that stemmed from user comments in our short experimentation phase. It would be highly desirable to have a Graphical User Interface

(GUI) capable of adapting to the level of user experience or to the needs of a particular experiment. For instance, only a subset of an instrument's sliders and buttons may be needed to perform the operations needed to carry out a specific task. In a learning session, a user may be guided progressively to the functionalities of the instrument front panel, through experimental phases that successively require more sophisticated manipulation or more keys to be available for operation.

- *Appropriate tools for self-training and learning-by-doing*. The usage of remote instrumentation to a large extent brings a novelty in teaching experimental sciences through distance learning sessions, as was noted in section 3.1.2 of [D4.1]. A topic to be investigated in the Experiment Execution Domain regards the development of tools by means of which users can perform supervised or even unsupervised training on the instruments, and gradually be acquainted with their usage. The relation of these tools with the world of distance learning standards should also be carefully explored.

All these capabilities should be embedded in business-value Remote Instrumentation Services. As a matter of fact, they are just a small instance of the flexible, user-friendly, workflow-oriented, orchestration services, a hint of which we have attempted to give in our discussion in section 5 of [D4.3].

6.3. Final recommendations for identified gaps

The recently established Remote Instrumentation in Grid Environment (RISGE) working group within OGF has undertaken the task of exploring "issues related to the exploitation of Grid technologies for conducting and monitoring measurement tasks and experiments on complex remote scientific equipment". As a final set of recommendations that attempt to summarize our investigation in this project and to provide input to the RISGE activity, we can list what follows. We do so in the light of the OGSA reference model, as adapted to the Remote Instrumentation environment. We start from the Physical Resources (including the network) and proceed upwards toward more sophisticated abstractions and services. It should be stressed, at this point, that it is out of the scope of the RINGrid project to provide solutions to the problems identified. Rather, the project performed this study in pursue of identifying the gaps in current approaches, and provide general recommendations about the directions that the RIS and the Grid communities must work towards.

6.3.1. Advanced networking services

The community should investigate the impact and the real needs in terms of advanced networking services. Among these:

- Mechanisms for QoS provisioning
- VPNs
- BoD and deployment of GMPLS functionalities for the dynamic creation of circuit-switched high-speed virtual connections on top of IP networks
- Traffic Engineering and cross-domain interaction

Networking services should provide flexible connectivity support in a heterogeneous environment of instrumentation and access network technologies, in order to fulfill the requirements of QoS-aware workflows in the abstract instrumentation domain. To this aim, a tighter interaction is required between the Grid and the network mangement and control functionalities, to translate QoS requirements across multiple domains and ensure rapid reconfiguration and adaptation.

6.3.2. IPv6

The impact and the benefits of the adoption of IPv6 should be investigated, in terms of transparency, enhanced flow handling, and mobility support. We believe that networking services in support of RIS can constitute a valid workbench for the introduction of widespread IPv6 connectivity, also in consideration of the large number of devices that should be made accessible over the network. These include both measurement instrumentation of different kinds and diffusion (i.e., not only large and expensive pieces of equipment, but also smaller and more common devices, which may be required to become part of a measurement chain), and sensor networks employed as large-scale data acquisition devices. The latter may be exposed as Grid resources through data sink devices, but even individual sensors may need to be addressable and configurable.

6.3.3. Access networks issues

The transport capabilities of access networks, and the possible limitations connected with wireless access, in both cases of user access and distributed data acquisition, should be assessed. In particular, maintaining QoS requirements over access networks is an important issue that is not yet completely addressed, though various different technologies (e.g., IEEE 802.11e, IEEE 802.16, ETSI DVB/DVB-RCS, among others) offer possible mechanisms. Another important aspect regards the scalability of applications in relation with user access and equipment characteristics, an issue that relates signal processing and networking topics. Finally, the role of satellites – sometimes the sole means of accessing remote or secluded areas – should be investigated in relation to RIS requirements, both with respect to terrestrial access and to controlling instrumentation in space laboratories.

6.3.4. Grid-scope enhancements

OGSA services must be enhanced or complemented in the Abstract Instrumentation Domain, especially with regard to:

- *EMS Services*: Definition and implementation of appropriate Service Providers for instrument virtualization, taking into account the tradeoff between generality and ease of code development, understanding and maintenance. Through the definition of a standardized interface for accessing (control, monitor, query) instruments, they will become first-level citizens of the Grid. However, as also mentioned earlier in the document, there must be a design which facilitates a common understanding about the access modalities, and at the same time eases development without undermining service robustness. We envisage that *get* and *set* operations will be available for the instrument properties, alongside a generic method execution operation. Additional operations for instrument reservation and data staging may be needed.
- *Data management*: Investigation of the relations between Digital Libraries and Remote Instrumentation experiments' data collection and storage. It is envisaged that such integration will allow the efficient indexing and re-composition of the huge volumes of scientific data produced by instruments, thus facilitating their processing on the Grid. Investigation and assessment of producer/consumer models for data

dispatching and distribution to wide user populations must also take place, to enhance multi-party visualization capabilities (in collaboration with multicast technologies).

- *Information Services*: Management of lists of instruments and their properties. This is the essential step towards integrating instruments on the Grid. Information models such as the GLUE schema [GLUE] must include instruments just like they include computing and storage resources.
- *Resource Management*: Mechanisms for enhanced (cross-domain) service discovery: Semantic matchmaking makes much sense with instruments, and it is a direction that the community should investigate, as also mentioned in section 6.3.6. Additionally, service-level QoS must be further looked into, in order to make it possible to have an understanding of the QoS levels that a service can offer, before trying to contact it for direct (best-effort) use or the establishment of an SLA. For example, it is important to know the average availability of an instrument, before booking it for an experiment. The implementation of such techniques will also pave the way for QoS-enabled workflows, which is a much-needed development. With scientific applications being composed as workflows of orchestrated services, the performance predictability of such workflows becomes an important component of service economies.

6.3.5. User interfaces and visualization

A lot of requirements were set by the user themselves when it came to user interfaces and visualization issues. Interactivity, complex widgets and high-resolution devices are required for visualization, and a lot of work has already taken place in this regard. However, even more important to the users are the interfaces with which they need to work. Self-documenting interfaces that do *not* have a steep learning curve are of great importance. In the same way, the users seem to prefer generic interfaces for instruments, in contrast to a complex full-featured interface. It appears, in this regard, that the most interesting and useful compromise would be a system that adjusts to the user and the application, through the definition of expertise levels and corresponding levels of complexity for the interface, the functionality offered and the represented instruments. Additionally, users have repeatedly mentioned the idea of an experiment repository, where templates of experiments would be available to be customized by users depending on the specifics of the experiment at hand.

6.3.6. Semantic technologies

Investigate the use of ontologies, semantic web concepts, and description languages to enable true service orchestration and composition, oriented to the construction, on the fly, of Virtual Laboratories and the configuration of experiments on them. The many different possible types of instruments pose difficulties when using generic interfaces – as actually suggested in section 6.3.4. Therefore, it is proposed that as a later step, the possibility of using semantic concepts and technologies be further explored, in order to be able to define instruments and perform activities such as semantic matchmaking. This would be extremely useful in order to be able to express complex concepts that take into account the countless types and characteristics of instruments. For instance, it would enable performing queries such as *"give me a list of atmospheric pressure sensors with a vendor-defined error margin of X%"*. This is not currently possible, and it is envisaged that semantic technologies will enable us to construct this type of queries for use with resource discovery, towards building really dynamic, complex scientific applications.

7. Conclusions

In this deliverable we went through the process of combining various previous results into a coherent set of recommendations for OGSA and other technology developers, to cater for the needs of Remote Instrumentation. We verified the requirements from [D4.3] and extended them based on the feedback from experiments and use case analysis.

A model remote-experimentation use case was formed, as a superset of all the use cases we collected, in this document. The results of experiments on prototypes from D6.1 were also included, as a (limited, but we believe useful) way to acquire an understanding of the users' perception on the issue. Then, the above were combined with final outcomes of D4.3.

The main, absolutely necessary facilities identified are instrument virtualization, error monitoring and logging, accounting and monitoring/controlling of experiment execution. Additionally to these, some other functional requirements exist: the presence of selection services, the existence of site functional tests, policy decision and enforcement mechanisms, input and output data management, and workflow management. Further (non-functional) requirements involve input management (in the preparatory phase), the presence of (and interaction with) local operator(s), and the presence of specialized visualization devices.

The non-strictly-technological comments by users suggest some additional requirements, to be addressed mostly conceptually and algorithmically: flexibility and adaptation to the user-specific experimental environment, flexibility and adaptation of the user interface to the user's skills and level of expertise, and appropriate tools for self-training and learning-by-doing.

Definitions, abbreviations, acronyms

RIS	Remote Instrumentation Services
SOI	Service-Oriented Infrastructure
SOA	Service-Oriented Architecture
OGSA	Open Grid Services Architecture
OGF	Open Grid Forum
RISGE-RG	Remote Instrumentation Services in Grid Environment Research Group
RTML	Remote Telescope Markup Language
API	Application Programming Interface
UML	Unified Modelling Language
PKI	Public Key Infrastructure
QoS	Quality of Service
GUI	Graphical User Interface
SLA	Service Level Agreement
VO	Virtual Organization
MPLS	MultiProtocol Label Switching
GMPLS	Generalized MultiProtocol Label Switching
BoD	Bandwidth on Demand
EMS	Execution Management Services
VPN	Virtual Private Network
VOMS	Virtual Organization Management System

References

[RISGE] – Open Grid Forum / Remote Instrumentation Services in Grid Environment Research Group; https://forge.gridforum.org/sf/projects/risge-rg

[D6.1] The RINGrid Consortium; RINGrid Deliverable 6.1, "Report on Prototype Preparation"; http://www.ringrid.eu/public/deliverables/RINGrid-WP6-D6_1-2008-03-14-3-GRN-Report_on_prototype_preparation.pdf

[WSARCH] The W3C Working Group; "Web Services Architecture"; http://www.w3.org/TR/ws-arch/

[COCKBURN] Alistair Cockburn; "Writing Effective Use Cases"; Addison-Wesley Professional, Oct. 2000; ISBN 0201702258

[D4.3] The RINGrid Consortium; RINGrid Deliverable D4.3, "WP4 Final Report"; http://www.ringrid.eu/public/deliverables/RINGRID-WP4-D4_3-2008-02-13-Final.pdf

[D4.2] The RINGrid Consortium; RINGrid Deliverable D4.2, "Influence of the Remote Instrumentation Trends on the Development of the Network Infrastructure and Grid Services"; http://www.ringrid.eu/public/deliverables/D4-2_Final.pdf

[PKIX] International Telecommunication Union (ITU); "Recommendation x.509"; http://www.itu.int/rec/T-REC-X.509/en

[MCCORMICK] B.H. McCormick; "Visualization in scientific computing"; ACM SIGBIO, Vol.10, Issue 1; 1988

[LASZEWSKI] Gregor von Laszewski et. al.; "Real-Time Analysis, Visualization, and Steering of Microtornography Experiments at Photon Sources"; ANL/MCS/CP-101186; 9th SIAM Conference on Parallel Processing for Scientific Computing, San Antonio, TX; 2000

[OGSA] Ian Foster et. al. / The Open Grid Forum; "The Open Grid Services Architecture, version 1.5"; http://www.ogf.org/documents/GFD.80.pdf

[DAV06] F. Davoli, G. Spanò, S. Vignola, S. Zappatore, "LABNET:

towards remote laboratories with unified access", *IEEE Transactions on Instrumentation and Measurement*, vol. 55, no. 5, pp. 1551-1558, Oct. 2006.

[D4.1] The RINGrid Consortium; RINGrid Deliverable D4.1, "Virtual Research Labs in Europe and Third-Party Countries: From Current State of the Art Towards the Future Technologies According to Users' Expectations"; http://www.ringrid.eu/public/deliverables/RINGRID-WP4-D4_1-2007-07-13-Final-CNIT-Virtual_Research_Labs_in_Europe_and_3rd_party_countries.pdf

[BBD06] R. Bolla, R. Bruschi, F. Davoli, "Capacity planning in IP Virtual Private Networks under mixed traffic", Computer Networks, vol. 50, no. 8, pp. 1069-1085, June 2006.

[BBDR08] R. Bolla, R. Bruschi, F. Davoli, M. Repetto, "Hybrid optimization for QoS control in IP Virtual Private Networks", Computer Networks, vol. 52, no. 3, pp. 563-580, Feb. 2008.

[KEJ06] G. Karmous-Edwards, A. Jukan, Eds., "An optical control plane for the Grid community: Opportunities, challenges, and vision", Feature Topic, IEEE Communications Magazine, vol. 44, no. 3, March 2006.

[GLUE] Open Grid Forum; The GLUE Schema Working Group; https://forge.gridforum.org/sf/projects/glue-wg

Contact Information

Dr. Norbert Meyer:
Poznań Supercomputing and Networking Center
ul. Noskowskiego 10
61-704 Poznań, Poland

URL: http://www.man.poznan.pl
Tel. (+48 61) 858-20-00
Fax (+48 61) 852-59-54

Constantinos (Costas) Kotsokalis	ckotso@admin.grnet.gr
Anastasios (Tasos) Zafeiropoulos	tzafeir@admin.grnet.gr
Franco Davoli	franco.davoli@cnit.it
Tomasz Rajtar	tomasz.rajtar@man.poznan.pl
Thomas Prokosch	thomas.prokosch@gup.uni-linz.ac.at
Marcela Larenas	mlarenas@reuna.cl

Appendix I: Responses to questionnaire

Evaluating the effects of noise and fading over a video transmission performed on a wireless channel

Name:	Luca Berruti
Email address:	luca.berruti@cnit.it
Education:	Dr. Ing
Discipline:	Telecommunications
Level of expertise:	Senior professional
Type of instrument:	Telecommunication measurement instrumentation
Summary of experiment:	The experiment aims at evaluating the effects of noise and fading over a video transmission performed on a wireless channel. A video source signal is encoded and sent over a wireless channel, where a channel simulator injects noise and emulates signal multipath fading. A receiver gets the noisy data from the wireless channel, decodes the stream and displays the video. The effects of the noise can be observed by means of a webcam pointed to the display of the receiver station. Distortion in the power spectrum of the received signal can be visualized by means of a spectrum analyzer.
Experiment duration:	10 min
Will exclusive access to the instrument be needed:	Yes
Most important technical limitations:	The absence of a signal-switching matrix to automatically perform the interconnections among the instruments involved in the experiment.
Steps of the experiment:	1. Logging into client that transmits video *Input:* Username, password *Output:* Authorisation *Is this an optional step?:* No *Are there special technical requirements?:* No *Is physical presence at instrument site required?:* No *If not, is it desired?:* No *If physical presence is required, would a local collaborator help?:* 2. Setup video transmitter *Input:* Parameters modified using video transmitter GUI *Output:* Video transmitter GUI updates *Is this an optional step?:* No *Are there special technical requirements?:* No *Is physical presence at instrument site required?:* No *If not, is it desired?:* No *If physical presence is required, would a local collaborator help?:*

3. Logging into client that receives video
Input: Username, password
Output: Authorisation
Is this an optional step?: No
Are there special technical requirements?: No
Is there physical presence at instrument site required?: No
If not, is it desired?: No
If physical presence is required, would a local collaborator help?:

4. Setup video receiver
Input: Parameters modified using video receiver GUI
Output: Video receiver GUI updates
Is this an optional step?: No
Are there special technical requirements?: No
Is physical presence at instrument site required?: No
If not, is it desired?: No
If physical presence is required, would a local collaborator help?:

5. Connect the wireless Access Point output at the transmitter end to the input of the channel simulator (Elektrobit PropSim C2).
Input:
Output:
Is this an optional step?: No
Are there special technical requirements?: No
Is physical presence at instrument site required?: Yes
If not, is it desired?: No
If physical presence is required, would a local collaborator help?: Yes

6. Connect the output of the channel simulator to the input of the wireless Access Point at the receiver end
Input:
Output:
Is this an optional step?: No
Are there special technical requirements?: No
Is physical presence at instrument site required?: Yes
If not, is it desired?: No
If physical presence is required, would a local collaborator help?: Yes

7. By using a directional coupler, connect the output of the channel simulator to the input of the Spectrum Analyzer (Agilent E4404B).
Input:
Output:
Is this an optional step?: No
Are there special technical requirements?: No
Is physical presence at instrument site required?: Yes
If not, is it desired?: No
If physical presence is required, would a local collaborator help?: Yes

8. Start the receiver.
Input: Corresponding button on receiver GUI
Output: Receiver GUI updates
Is this an optional step?: No
Are there special technical requirements?: No
Is physical presence at instrument site required?: No
If not, is it desired?: No
If physical presence is required, would a local collaborator help?:

9. Start video transmission
Input: Corresponding button on transmitter GUI
Output: Transmitter GUI updates
Is this an optional step?: No
Are there special technical requirements?: No
Is physical presence at instrument site required?: No
If not, is it desired?: No
If physical presence is required, would a local collaborator help?:

10. Start Device Farm (3D representation)
Input: Browsing to specific URL
Output: Web browser display update
Is this an optional step?: No
Are there special technical requirements?: Standards-compliant web browser with necessary 3D plugins
Is physical presence at instrument site required?: No
If not, is it desired?: No
If physical presence is required, would a local collaborator help?:

11. Choose a fading profile, configure and switch on the channel simulator.
Input: Changing options on the device farm browser representation
Output: Browser window updates
Is this an optional step?: No
Are there special technical requirements?: No
Is physical presence at instrument site required?: No
If not, is it desired?: No
If physical presence is required, would a local collaborator help?:

12. Set noise level in channel simulator.
Input: Changing options on the device farm browser representation
Output: Browser window updates
Is this an optional step?: No
Are there special technical requirements?: No
Is physical presence at instrument site required?: No
If not, is it desired?: No
If physical presence is required, would a local collaborator help?:

13. Observe the effects of fading and noise on the decoded

video stream.
Input:
Output: Video receiver GUI updates
Is this an optional step?: No
Are there special technical requirements?: No
Is physical presence at instrument site required?: No
If not, is it desired?: No
If physical presence is required, would a local collaborator help?:

14. Observe the spectrum.

Input: Device farm on-screen representation.
Output:
Is this an optional step?: No
Are there special technical requirements?: No
Is physical presence at instrument site required?: No
If not, is it desired?: No
If physical presence is required, would a local collaborator help?:

15. Measure the packet loss and missing video frames.

Input: Video receiver GUI
Output:
Is this an optional step?: No
Are there special technical requirements?: No
Is physical presence at instrument site required?: No
If not, is it desired?: No
If physical presence is required, would a local collaborator help?:

16. Repeat the measurement after changing noise level and/or fading profile.

Input: Change parameters for video transmitter and device farm
Output:
Is this an optional step?: Yes
Are there special technical requirements?: No
Is there physical presence at instrument site required?: No
If not, is it desired?: No
If physical presence is required, would a local collaborator help?:

17. Switch-off device farm, transmitter and receiver

Input: Change parameters within the browser representations
Output:
Is this an optional step?: No
Are there special technical requirements?: No
Is physical presence at instrument site required?: No
If not, is it desired?: No
If physical presence is required, would a local collaborator help?:

Measurement of ultra high energetic cosmic rays with fluorescence detector telescopes and a Cherenkov detector-based ground array

Name:	Michael Sutter
Email address:	michael.sutter@ipe.fzk.de
Education:	MSc
Discipline:	Software Developer
Level of expertise:	Expert
Type of instrument:	Fluorescence telescopes and supply units
Summary of experiment:	The experiment's goal is to measure ultra high energetic cosmic rays with fluorescence detector telescopes and a Cherenkov detector based ground array. The experiment is located in Argentina (construction nearly finished) and in the US (planned). Energy spectrum, mass composition and source origin (anisotropy) are expected as major results. Fluorescence shifts last normally 14-16 hours each day over a period of two weeks of a month. Operation is planned for about 20 years.
Experiment duration:	16 h
Will exclusive access to the instrument be needed:	No
Most important technical limitations:	Limited network bandwidth to the experiment site and the experiments sub-sites (= within the 3000 sqrkm experiment site)

Steps of the experiment:

1. Switch on experiment
   - *Input:* Check that every computer system and service is in place for the execution of the experiment
   - *Output:* Possible error condition: In case of hardware problems, someone must be on-site
   - *Is this an optional step?:* No
   - *Are there special technical requirements?:*
   - *Is physical presence at instrument site required?:* Yes
   - *If not, is it desired?:*
   - *If physical presence is required, would a local collaborator help?:* Yes (for the repair of damaged hardware if needed)
2. Pre-RUN calibration
   - *Input:* Select experiment configuration, type of calibration
   - *Output:* Possible error condition: Bad photo multipliers. In this case, a log of the event must be kept.
   - *Is this an optional step?:* No
   - *Are there special technical requirements?:* -
   - *Is physical presence at instrument site required?:* No

- *If not, is it desired?:* No
- *If physical presence is required, would a local collaborator help?:*

3. Data acquisition: monitor data, rates, configuration, alarms.
   - *Input:* Experiment configuration, weather conditions
   - *Output:* Event rates. In case the moon is over the telescope, a re-configuration must be made and the experiment can continue.
   - *Is this an optional step?:* No
   - *Are there special technical requirements?:* Web (or web-like) access with graphs and events presented over the network
   - *Is physical presence at instrument site required?:* No
   - *If not, is it desired?:* No
   - *If physical presence is required, would a local collaborator help?:*
4. Post-RUN calibration
   - *Input:* Select experiment configuration, type of calibration
   - *Output:* Possible error condition: Bad photo multipliers. In this case, a log of the event must be kept, to be taken into account when analysing the data.
   - *Is this an optional step?:* No
   - *Are there special technical requirements?:*
   - *Is physical presence at instrument site required?:* No
   - *If not, is it desired?:* No
   - *If physical presence is required, would a local collaborator help?:*
5. Set to stand-by for next shift
   - *Input:* Switch off through corresponding software option and verify standby/shutdown state
   - *Output:* In case of error?
   - *Is this an optional step?:* No
   - *Are there special technical requirements?:*
   - *Is physical presence at instrument site required?:* No
   - *If not, is it desired?:* No
   - *If physical presence is required, would a local collaborator help?:*

Running and processing 1D NMR spectra

Name:	Zofia Gdaniec
Email address:	zgdan@ibch.poznan.pl
Education:	PhD
Discipline:	NMR spectroscopy
Level of expertise:	Expert
Type of instrument:	NMR spectrometer
Summary of experiment:	Running and processing 1D NMR spectra
Experiment duration:	1 h
Will exclusive access to the instrument be needed:	Yes
Most important technical limitations:	Necessity of delivering a sample to a distant location. Requirement of the physical presence of a collaborator at the beginning of the session in order to put a sample into the magnet and tune a sample.

Steps of the experiment:

1. Prepare a sample
   - *Input:*
   - *Output:*
   - *Is this an optional step?:* No
   - *Are there special technical requirements?:*
   - *Is physical presence at instrument site required?:* Yes
   - *If not, is it desired?:*
   - *If physical presence is required, would a local collaborator help?:* Yes (the collaborator can prepare it for us)
2. Insert the sample into the magnet
   - *Input:*
   - *Output:*
   - *Is this an optional step?:* No
   - *Are there special technical requirements?:*
   - *Is physical presence at instrument site required?:* Yes
   - *If not, is it desired?:*
   - *If physical presence is required, would a local collaborator help?:* Yes (the collaborator can insert it for us)
3. Logging in to the computer that controls the spectrometer and start the VNMR software
   - *Input:* User name, password
   - *Output:* Desktop interface, VNMR windows
   - *Is this an optional step?:* No
   - *Are there special technical requirements?:*
   - *Is physical presence at instrument site required?:* No

  - *If not, is it desired?:* No
  - *If physical presence is required, would a local collaborator help?:*
4. Creating parameter sets
  - *Input:* Parameters
  - *Output:*
  - *Is this an optional step?:* No
  - *Are there special technical requirements?:*
  - *Is physical presence at instrument site required?:* No
  - *If not, is it desired?:* No
  - *If physical presence is required, would a local collaborator help?:*
5. Setting frequency-related parameters
  - *Input:* Putting frequencies of 1H and/or heteronucleus
  - *Output:*
  - *Is this an optional step?:* No
  - *Are there special technical requirements?:*
  - *Is physical presence at instrument site required?:* No
  - *If not, is it desired?:* No
  - *If physical presence is required, would a local collaborator help?:*
6. Tuning a probe
  - *Input:* Adjusting the hardware
  - *Output:*
  - *Is this an optional step?:* Yes
  - *Are there special technical requirements?:*
  - *Is there physical presence at instrument site required?:* Yes
  - *If not, is it desired?:*
  - *If physical presence is required, would a local collaborator help?:* Yes
7. Locking sample, adjusting shims
  - *Input:* Optimising the parameters using acquisition window
  - *Output:*
  - *Is this an optional step?:* No
  - *Are there special technical requirements?:*
  - *Is physical presence at instrument site required?:* No
  - *If not, is it desired?:* No
  - *If physical presence is required, would a local collaborator help?:*
8. Setting pulse-sequence related parameters
  - *Input:* VNMR software commands
  - *Output:*
  - *Is this an optional step?:* No
  - *Are there special technical requirements?:*

- *Is physical presence at instrument site required?:* No
- *If not, is it desired?:* No
- *If physical presence is required, would a local collaborator help?:*

9. Initiating acquisition
   - *Input:* VNMR software commands
   - *Output:* Acquisition status window
   - *Is this an optional step?:* No
   - *Are there special technical requirements?:*
   - *Is physical presence at instrument site required?:* No
   - *If not, is it desired?:* No
   - *If physical presence is required, would a local collaborator help?:*
10. Queuing acquisitions
    - *Input:* VNMR software commands
    - *Output:* Acquisition status window update
    - *Is this an optional step?:* Yes
    - *Are there special technical requirements?:* -
    - *Is physical presence at instrument site required?:* No
    - *If not, is it desired?:* No
    - *If physical presence is required, would a local collaborator help?:*
11. Running the experiment and saving experimental data
    - *Input:* Command to start the experiment
    - *Output:* Stored experimental data
    - *Is this an optional step?:* No
    - *Are there special technical requirements?:*
    - *Is there physical presence at instrument site required?:* No
    - *If not, is it desired?:* No
    - *If physical presence is required, would a local collaborator help?:*
12. Processing experimental data
    - *Input:* VNMR software commands and input data set
    - *Output:* Processed data
    - *Is this an optional step?:* No
    - *Are there special technical requirements?:*
    - *Is physical presence at instrument site required?:* No
    - *If not, is it desired?:* No
    - *If physical presence is required, would a local collaborator help?:*
13. Plotting spectra
    - *Input:* VNMR software commands and processed data set
    - *Output:* Spectra visualisation

- *Is this an optional step?:* No
- *Are there special technical requirements?:*
- *Is physical presence at instrument site required?:* No
- *If not, is it desired?:* No
- *If physical presence is required, would a local collaborator help?:*

14. Finishing the session
    - *Input:* VNMR software commands
    - *Output:*
    - *Is this an optional step?:* No
    - *Are there special technical requirements?:*
    - *Is there physical presence at instrument site required?:* No
    - *If not, is it desired?:* No
    - *If physical presence is required, would a local collaborator help?:*

8 bit AD/DA Experiment (HF)

Name:	Giancarlo Parodi
Email address:	giancarlo.parodi@unige.it
Education:	Dr. Ing
Discipline:	Electronics
Level of expertise:	Expert
Type of instrument:	Waveform generators, oscilloscopes
Summary of experiment:	8 bit AD/DA Experiment (HF): The AD/DA converter receives an analog input, then converts it into a digital signal using 8 bits and finally converts it into the output analog waveform. This experiment works at HF (1000 Hz). The sampling rate is 3 kHz. The remote user is expected to modify the signal input and to observe the results.
Experiment duration:	10 min
Will exclusive access to the instrument be needed:	No
Most important technical limitations:	
Steps of the experiment:	1. Switch on the instruments o *Input:* Click on corresponding button o *Output:* o *Is this an optional step?:* No o *Are there special technical requirements?:* - o *Is physical presence at instrument site required?:* Yes o *If not, is it desired?:*

- *If physical presence is required, would a local collaborator help?:* Yes

2. Set instrument configuration
    - *Input:* Test points, input waveform (type, frequency, amplitude), oscilloscope parameters (ch1 amplitude, ch1 offset, ch1 coupling, ch2 amplitude, ch2 offset, ch2 coupling, time base)
    - *Output:*
    - *Is this an optional step?:* No
    - *Are there special technical requirements?:* -
    - *Is physical presence at instrument site required?:* Yes
    - *If not, is it desired?:*
    - *If physical presence is required, would a local collaborator help?:* Yes
3. Execute measurements
    - *Input:* Click on corresponding button
    - *Output:* Graphical display of waveform
    - *Is this an optional step?:* No
    - *Are there special technical requirements?:* -
    - *Is physical presence at instrument site required?:* Yes
    - *If not, is it desired?:*
    - *If physical presence is required, would a local collaborator help?:* Yes
4. Check results
    - *Input:* Waveform
    - *Output:*
    - *Is this an optional step?:* No
    - *Are there special technical requirements?:*
    - *Is physical presence at instrument site required?:* No
    - *If not, is it desired?:* No
    - *If physical presence is required, would a local collaborator help?:*
5. Loop to step 2
    - *Input:*
    - *Output:*
    - *Is this an optional step?:* Yes
    - *Are there special technical requirements?:*
    - *Is physical presence at instrument site required?:* No
    - *If not, is it desired?:* No
    - *If physical presence is required, would a local collaborator help?:*
6. Close experiment
    - *Input:* Click on button to switch off equipment
    - *Output:*
    - *Is this an optional step?:* No
    - *Are there special technical requirements?:*

- *Is physical presence at instrument site required?:* Yes
- *If not, is it desired?:*
- *If physical presence is required, would a local collaborator help?:* Yes

Acid-Base Valuation

Name:	Miguel Alvarez Pasaye
Email address:	malvarez@servidor.unam.mx
Education:	MSc
Discipline:	Chemistry
Level of expertise:	Senior professional
Type of instrument:	pH sensor
Summary of experiment:	This experiment is to develop a methodology to undertake the valuation of an acid with a base and receiving the pH data obtained. Acid-base valuations are the most common. When a standard solution of sodium hydroxide is added to hydrochloric acid, there is neutralization. The detection of the point of equivalence is measured with a pH sensor, which is connected to a computer through a serial port interface; data can be transmitted to a monitor either locally or remotely or stored in a file for subsequent analysis. This procedure may be observed by other users through a video camera.
Experiment duration:	30 min
Will exclusive access to the instrument be needed:	No
Most important technical limitations:	The software used to collect the data is proprietary, and must be installed on the computer connected to the instrument.

Steps of the experiment:

1. Sensor connection to the PC
   - *Input:* Configuration parameters of operation of the universal lab interface.
   - *Output:*
   - *Is this an optional step?:* No
   - *Are there special technical requirements?:* Two options: Windows, the client program running at least win98 with 128 Mb RAM. In Linux, implementation of programs in C language so the relevant libraries are necessary.
   - *Is physical presence at instrument site required?:* Yes
   - *If not, is it desired?:*
   - *If physical presence is required, would a local collaborator help?:* No
2. Put Hydrochloric Acid

- *Input:* 50 ml of hydrochloric acid are added to the cell sensor.
- *Output:*
- *Is this an optional step?:* No
- *Are there special technical requirements?:* -
- *Is physical presence at instrument site required?:* Yes
- *If not, is it desired?:*
- *If physical presence is required, would a local collaborator help?:* No

3. Add sodium hydroxide
   - *Input:* Each 30 sec a drop of sodium hydroxide is added
   - *Output:* pH value through the program
   - *Is this an optional step?:* No
   - *Are there special technical requirements?:*
   - *Is physical presence at instrument site required?:* Yes
   - *If not, is it desired?:*
   - *If physical presence is required, would a local collaborator help?:* No
4. Repeat Step 3 until the end
   - *Input:* The experiment ends when the neutralization graph is complete (approx.30 min)
   - *Output:* Graphical variation of pH with the concentration of sodium hydroxide added.
   - *Is this an optional step?:* No
   - *Are there special technical requirements?:*
   - *Is physical presence at instrument site required?:* Yes
   - *If not, is it desired?:*
   - *If physical presence is required, would a local collaborator help?:* No

Determination of freezing temperature for specific types of meat

Name:	Moises Hernandez Duarte
Email address:	hduarte@servidor.unam.mx
Education:	MSc
Discipline:	Food engineering
Level of expertise:	Senior professional
Type of instrument:	Blast freezing tunnel
Summary of experiment:	The experiment consists of obtaining the freezing temperatures of a sample of meat that has been placed in a Blast Freezing Tunnel with several thermocouples inserted (from 1 to 12). The values of temperature are captured via the serial port of a computer, connected wirelessly to a local area network. For the experiment

	to be seen by several students simultaneously, a video camera transmits the experiment process. In addition to the video, the client computer receives temperature data obtained by the thermocouple inserted in the sample of meat.
Experiment duration:	40 m
Will exclusive access to the instrument be needed:	Yes
Most important technical limitations:	Parameters for the Blast Freezing Tunnel can only be modified from the panel. There is no way to interact remotely.
Steps of the experiment:	

1. Blast Freezing Tunnel connection to the PC
   - *Input:* Setup operational parameters at the instrument panel
   - *Output:*
   - *Is this an optional step?:* No
   - *Are there special technical requirements?:* In changing parameters for the Blast Freezing Tunnel
   - *Is physical presence at instrument site required?:* Yes
   - *If not, is it desired?:*
   - *If physical presence is required, would a local collaborator help?:* No
2. Put meat sample in the instrument
   - *Input:* Prepare and put the meat sample in the instrument
   - *Output:*
   - *Is this an optional step?:* No
   - *Are there special technical requirements?:*
   - *Is physical presence at instrument site required?:* Yes
   - *If not, is it desired?:*
   - *If physical presence is required, would a local collaborator help?:* No
3. Start collecting data
   - *Input:* Execute the program for collecting data
   - *Output:* Data set produced
   - *Is this an optional step?:* No
   - *Are there special technical requirements?:*
   - *Is physical presence at instrument site required?:* No
   - *If not, is it desired?:* No
   - *If physical presence is required, would a local collaborator help?:*
4. Finishing the experiment
   - *Input:*
   - *Output:* Final data collection
   - *Is this an optional step?:* No
   - *Are there special technical requirements?:*
   - *Is physical presence at instrument site required?:*

No
- *If not, is it desired?:* No
- *If physical presence is required, would a local collaborator help?:*

Wireless sensor networks in indoor environment

Name:	Paolo Medagliani
Email address:	paolo.medagliani@unipr.it
Education:	Dr. Ing
Discipline:	Wireless Sensor Networks
Level of expertise:	Senior professional
Type of instrument:	PICDEM Z wireless sensors
Summary of experiment:	We have placed a few nodes in our offices, in order to measure throughput, delay and connectivity in indoor environment. Our experiments have been performed both in the presence and in the absence of a relay node. The results effectively show both the bimodal behavior and the throughput and delay performance predicted by theoretical analysis
Experiment duration:	1 h
Will exclusive access to the instrument be needed:	Yes
Most important technical limitations:	The two most important limitations encountered in our experiments are: (i) limited number of wireless sensors used for our tests (due to the elevated price of each device); (ii) the source code provided by the Microchip presents some bugs, which sometimes have prevented the corrected execution of the experiments.

Steps of the experiment:

1. Generation of source code to upload to sensors
   - *Input:*
   - *Output:* C source code
   - *Is this an optional step?:* No
   - *Are there special technical requirements?:* Code generator
   - *Is physical presence at instrument site required?:* No
   - *If not, is it desired?:* No
   - *If physical presence is required, would a local collaborator help?:*
2. Uploading generated code to the sensors
   - *Input:* Generated code
   - *Output:*
   - *Is this an optional step?:* No
   - *Are there special technical requirements?:* Appropriate programming unit

- *Is physical presence at instrument site required?:* Yes
- *If not, is it desired?:*
- *If physical presence is required, would a local collaborator help?:* Yes

3. Place nodes in departmental offices in order to evaluate the impact of walls, furniture, people crossing the wireless transmission links.
   - *Input:*
   - *Output:*
   - *Is this an optional step?:* No
   - *Are there special technical requirements?:*
   - *Is physical presence at instrument site required?:* Yes
   - *If not, is it desired?:*
   - *If physical presence is required, would a local collaborator help?:* Yes
4. Connect collector node to PC
   - *Input:*
   - *Output:*
   - *Is this an optional step?:* No
   - *Are there special technical requirements?:* RS-232 ports and cable
   - *Is physical presence at instrument site required?:* Yes
   - *If not, is it desired?:*
   - *If physical presence is required, would a local collaborator help?:* Yes
5. Receive/record incoming packets and perform real-time statistical analysis
   - *Input:* Incoming wireless link packets
   - *Output:* Throughput, delay and connectivity performance estimations
   - *Is this an optional step?:* No
   - *Are there special technical requirements?:* Windows or Unix ad-hoc software for the analysis of raw data
   - *Is physical presence at instrument site required?:* No
   - *If not, is it desired?:* No
   - *If physical presence is required, would a local collaborator help?:*

Global network of robotic telescopes

Name:	Frank Breitling
Email address:	fbreitling@aip.de
Education:	MSc
Discipline:	Robotic astronomy
Level of expertise:	Senior professional
Type of instrument:	Telescopes
Summary of experiment:	OpenTel provides the means for interconnecting single robotic telescopes to a global network for sharing observation time, observation programs and data. OpenTel is an open network. Global networks of robotic telescopes provide important advantages over single telescopes. Independent of daytime and weather, they can more efficiently perform multi-wavelength observations and continuous long-term monitoring, as well as react rapidly to transient events such as GRBs and supernovas.
Experiment duration:	Seconds to years
Will exclusive access to the instrument be needed:	Yes
Most important technical limitations:	Bandwidth often is a problem for telescopes, which are located at remote locations.

Steps of the experiment:

1. Log into the system
   - *Input:* Credentials
   - *Output:* Authorised role
   - *Is this an optional step?:* No
   - *Are there special technical requirements?:*
   - *Is physical presence at instrument site required?:* No
   - *If not, is it desired?:* No
   - *If physical presence is required, would a local collaborator help?:*
2. Prepare an RTML request
   - *Input:* Experimental requirements
   - *Output:* RTML request
   - *Is this an optional step?:* No
   - *Are there special technical requirements?:*
   - *Is physical presence at instrument site required?:* No
   - *If not, is it desired?:* No
   - *If physical presence is required, would a local collaborator help?:*
3. Submit RTML request to the system
   - *Input:* RTML request
   - *Output:* Monitoring/observation data & metadata
   - *Is this an optional step?:* No
   - *Are there special technical requirements?:* The

system performs itself the (instrument) resource selection and does the scheduling of the observation. Additionally, the system is always online, so no calibration or switch-off is needed.
    - *Is physical presence at instrument site required?:* No
    - *If not, is it desired?:* No
    - *If physical presence is required, would a local collaborator help?:*
4. Execute application to process the resulting data
    - *Input:* Observation/monitoring data
    - *Output:* Experimental (application) results
    - *Is this an optional step?:* Yes
    - *Are there special technical requirements?:* An application to process the stored data is required.
    - *Is physical presence at instrument site required?:* No
    - *If not, is it desired?:* No
    - *If physical presence is required, would a local collaborator help?:*
5. Log off
    - *Input:* Appropriate portal option
    - *Output:*
    - *Is this an optional step?:* Yes
    - *Are there special technical requirements?:*
    - *Is physical presence at instrument site required?:* No
    - *If not, is it desired?:* No
    - *If physical presence is required, would a local collaborator help?:*

Detection experiment of the type Ic supernova 2007gr

Name:	Zsolt Paragi
Email address:	zparagi@jive.nl
Education:	PhD
Discipline:	Stellar and extragalactic VLBI, e-VLBI
Level of expertise:	Expert
Type of instrument:	e-EVN
Summary of experiment:	Detection experiment of the type Ic supernova 2007gr. Quick analysis of data providing accurate coordinates and check of flux density development, making follow-up VLBI observations with a global VLBI array possible. This was only the second source of this type ever detected with VLBI.
Experiment duration:	12 hours
Will exclusive access to the instrument be	Yes

needed:	
Most important technical limitations:	Resolution, limited by the short baselines in the current e-EVN array.
Steps of the experiment:	1. Define the observing setup and the exact timing of the observations of different target sources simultaneously with various remote radio telescopes. o *Input:* Text file defining the observations, to be processed by the Sched software. o *Output:* (Output of the Sched software) A vexfile, that can be used (after additional steps) to control the telescopes during the observations. o *Is this an optional step?:* No o *Are there special technical requirements?:* Availability of latest version of the Sched software o *Is physical presence at instrument site required?:* No o *If not, is it desired?:* No o *If physical presence is required, would a local collaborator help?:* 2. Carry out simultaneous observations on telescopes o *Input:* Vexfile (schedule) designed by the user o *Output:* Data towards the central data processor (correlator) in real time. There are various steps at the correlator before the final data reach the user. o *Is this an optional step?:* No o *Are there special technical requirements?:* 256 Mbit/s data rate from the telescopes for sufficient image sensitivity, and about 5 milliarcsecond image resolution. o *Is physical presence at instrument site required?:* Yes o *If not, is it desired?:* o *If physical presence is required, would a local collaborator help?:* Yes 3. Data processing: Scientific analysis of the data, model-fitting, imaging etc. o *Input:* Output from the correlator o *Output:* The scientific result, e.g., an image, and eventually the publication of the results. o *Is this an optional step?:* No o *Are there special technical requirements?:* Appropriate software, probably. o *Is physical presence at instrument site required?:* No o *If not, is it desired?:* No o *If physical presence is required, would a local collaborator help?:*

Moessbauer spectra measurement

Name:	Zara Cherkezova-Zheleva
Email address:	zzhel@ic.bas.bg
Education:	PhD
Discipline:	Moessbauer spectroscopy, XRD, catalysis, green chemistry
Level of expertise:	Senior professional
Type of instrument:	Moessbauer spectrometer
Summary of experiment:	The Moessbauer spectra measurement includes a number of steps: providing the required parameters; accumulation and fit of standard spectrum (a-Fe foil); putting the sample into the holder and then into the apparatus; getting and fitting the experimental spectrum of the sample; analysis and interpretation of the obtained hyperfine parameters using data base. The obtained results are phase identification, phase composition, oxidation degree of ions, ion coordination and magnetic behavior.
Experiment duration:	Few hours up to few days
Will exclusive access to the instrument be needed:	Yes
Most important technical limitations:	The samples should contain a minimum of 3% iron.

Steps of the experiment:

1. Providing the required parameters - frequency region, resolution, number of channels, type of memory, experiment geometry, aperture setting
   - *Input:* The parameters
   - *Output:* Optimal experiment parameters
   - *Is this an optional step?:* Yes
   - *Are there special technical requirements?:* Knowledge of the equipment use guide
   - *Is physical presence at instrument site required?:* Yes
   - *If not, is it desired?:*
   - *If physical presence is required, would a local collaborator help?:* No
2. Accumulation of spectrum of standard
   - *Input:* 10 mg/cm2, alpha-Fe foil
   - *Output:* Standard spectrum for the spectrum region calibration
   - *Is this an optional step?:* Yes
   - *Are there special technical requirements?:*
   - *Is physical presence at instrument site required?:* Yes
   - *If not, is it desired?:*
   - *If physical presence is required, would a local collaborator help?:* No
3. Fit of standard spectrum

- *Input:* Input the spectral data in the PC programme for calculations
- *Output:* Maximum and reference velocity, folding point
- *Is this an optional step?:* No
- *Are there special technical requirements?:* Fitting programme and experience
- *Is physical presence at instrument site required?:* No
- *If not, is it desired?:* No
- *If physical presence is required, would a local collaborator help?:*

4. Putting the sample into the holder and then into the apparatus
    - *Input:* Quantity of the sample, which contains 10 mg/cm^2 Fe
    - *Output:* Holder with sample
    - *Is this an optional step?:* No
    - *Are there special technical requirements?:*
    - *Is physical presence at instrument site required?:* No
    - *If not, is it desired?:* No
    - *If physical presence is required, would a local collaborator help?:*
5. Getting the experimental spectrum of sample
    - *Input:* Starting of spectra accumulation
    - *Output:* Spectrum of the sample
    - *Is this an optional step?:* No
    - *Are there special technical requirements?:*
    - *Is physical presence at instrument site required?:* Yes
    - *If not, is it desired?:*
    - *If physical presence is required, would a local collaborator help?:* Yes
6. Fit of spectrum of sample
    - *Input:* Input the spectral data in the PC programme for calculations
    - *Output:* Hyperfine parameters
    - *Is this an optional step?:* No
    - *Are there special technical requirements?:* Fitting programme and experience
    - *Is physical presence at instrument site required?:* No
    - *If not, is it desired?:* No
    - *If physical presence is required, would a local collaborator help?:*
7. Analysis and interpretation of the obtained hyperfine parameters
    - *Input:* Comparison with database
    - *Output:* Phase identification, phase composition,

oxidation degree of ions, ion coordination, magnetic behaviour

- *Is this an optional step?:* No
- *Are there special technical requirements?:* Database
- *Is physical presence at instrument site required?:* No
- *If not, is it desired?:* No
- *If physical presence is required, would a local collaborator help?:*