A Pervasive Grid, from the data side

Jean-Marc Pierson
LIRIS (UMR CNRS, INSA)
Institut National Sciences Appliquées
7 avenue Jean Capelle, 69621 Villeurbanne cedex, France
Jean-Marc.Pierson@liris.cnrs.fr

Abstract

In this article, we give our vision of a Pervasive Grid: A grid dealing with light, mobile and uncertain devices, using context-awareness for delivering the right information using the right infrastructure to final users. We focus on the data side of the problem, since it encompass a number of problems related to such an environment, from data access, query optimization, data placement, to security, adaptation,... We present on an use case how a proposed set of services can interact and help to solve a typical pervasive scenario.

Keywords: grid computing, pervasive systems, data management

1 Introduction

The last eight years have seen the wide development and adoption of Grid Computing [11, 18]. Its main purpose is to help for coordinating large-scale heterogeneous resources sharing (hardware, software) and problem solving in dynamic, multi-institutional virtual organizations. From its original purpose of distributed supercomputing and massive data processing, Grid computing moves more and more to become a common platform and way for utility and service computing. It allows the sharing and the coordination in heterogeneous environments (from clusters, main frames to personal computers) through relatively well developed middlewares architectures like OGSA [12] and one major implementation Globus [13].

On the other hand, mobile, small, wearable personal devices are entering our everyday life. People expect more and more information and services to be accessible or readable on their hand-held digital assistant, from cell phones to personal digital assistants. Pervasive Information Systems aims at offering the right information, to the right people, at the right time, on the right format. People are not aware of the transparent infrastructure that offers them the information or the service. Non intrusiveness, mobility, context-awareness and unpredictability characterize the key constraints of these environments.

The current trend is to interconnect the two domains, offering the enormous potential for resource sharing to hand-held devices. Grid services would then be accessible, controllable from pervasive devices, and results would be sent to these devices for on-site or remote analysis.

Our purpose is a little bit different, and go a step further : we envision a Pervasive Grid. In this Pervasive Grid, resource sharing is still the key concept, and we include the full heterogeneity of devices (even hand-helds) in this Grid. While I. Foster and al. denote in [11] "A computational grid is a hardware and software infrastructure that provides dependable, consistent, pervasive, and inexpensive access to high-end computational capabilities", the pervasive term has been foreseen here as a way to offer a transparent access for high computational resources for end-users. To paraphrase I. Foster, a Pervasive Grid is a hardware and software infrastructure that provides dependable, consistent and inexpensive access to pervasive resource sharing capabilities. Users are not aware of the set of available resources, they just use services without notice. Theses resources may be either computational cluster, mobile phones, or whatever can share something with the outside world. Conversely, in a proactive way, the pervasive environment may provide users with information according to their preferences.

Inventing such a Pervasive Grid is a large task, that can not be done in one article. We will focus our purpose in this article on the data side of the problem : data management in a Pervasive Grid. The first reason for this is that data is the core of the system: Most of the applications use data as input or produce data as output. The second reason is because the data management has long been forgotten in Grid Computing, and efforts has been made late in order to handle the available data. And finally, the data side exhibits a number of problems symptomatic of the Pervasive Grid environment.

The rest of the article is organized as follows : After motivating the lacks in existing systems (either Pervasive or Grid Computing) to achieve our vision in section 2, we detail in section 3 who are the actors of a Pervasive Grid in a global architecture view, and we identify a number of standalone services for data management one would like to integrate. We present in section 4 mutual links, interactions, and today advances in our group with the most promising steps in the direction of a Pervasive Grid. We then exhibit the mentioned services on a typical use case (section 5). Before concluding and presenting current trends, we discuss some implementation issues in section 6.

In the following, P-Grid will refer to Pervasive Grid.

2 Related Work

The two domains related to a Pervasive Grid are the Grid and the Pervasive systems, which are both very young research fields. Nevertheless, due to both very active communities, some works have already started in the direction of a P-Grid.

Davies, Storz and Friday [9, 34] were the first to introduce the concept of

"Ubiquitous Grid", that is close to our P-Grid. The purpose of their article is to compare the notion of Grid Computing (definition of I. Foster [12]) and the notion of Pervasive Systems (definition of M. Weiser [35]). They identify similar interests : heterogeneity, interoperability, scalability, adaptability and fault tolerance, resources management, services composition, discovery, security, communication, audit, payment. They present briefly an use case developed with Globus Toolkit 3 (GT3), for an ubiquitous grid and their solution. Lack of details makes it difficult to evaluate exactly what has been done to make GT3 behave like an ubiquitous grid, and what parts of ubiquity has been addressed. Moreover, the authors do not deal with data management.

Hingne et al. [17] propose a multi-agent approach to realize a P-Grid. They are interested in communication, heterogeneity, discovery and services composition, and scheduling of tasks between the different devices constituting the P-Grid. Almost nothing is said concerning the data management.

McKnight et al. [22] introduce the concept of Wireless Grid. Their interest is on the mobile and nomadic issues that they compare with traditional computing grids, P2P networks and web services. An interesting point of this article is to put in light the relationships between these actors. The authors focus their interest on services they identify as the most important: resources description and discovery, coordination, trust management and access control. These latter points are related to data management, together with the resources (data) description and discovery.

In [33], S.H. Srinivasan details a Wireless Pervasive Grid architecture. The author separates its grid in two parts: One is the "backbone grid", physically linked like the network backbones; the other one is wireless, the "access grid". Agents realize the proxy between the two grids, and behave on behalf of the mobile devices of the "access grid" on the "backbone grid". Interesting points in this proposal concern the pro-activity and the context-awareness of the presentation to the end-users.

Complementary to these, some other works exist to tackle some aspects of pervasive systems in the computing grids, mainly mobility and adaptation to light devices. Some works [27, 16, 14] have focused on the use of light devices to interact with computing grids, submitting jobs and visualizing results. Integrating more the mobile devices in the grids are [24, 23] that propose proxy services to distribute and organize the jobs among a pool of light devices. [20] solicits surrounding devices to participate in a problem solving environment. Others [26, 19] are interested in the advantages of mobility features of IPv6 in the notification and adaptation of grids. Mobile agents are used in [5, 2] to migrate objects between sites. Some researchers [7, 8] have investigated how a grid middleware (Legion, OGSI.NET) can be adapted to tackle mobility issues. Context-awareness are basis for the works of [36], while the authors in [10] include mobility and context-awareness in their approach.

Unfortunately, almost none of these works handle the complexity of a P-Grid as a whole, dealing with mobility, context-awareness and unpredictability, at the same time. And none are handling the data management itself. We thus propose in the following to have an integrated view of services related to data management in a Pervasive Grid.

3 Services for data management in a P-Grid

We have identified a minimal set of services related to the management of data, that must be present in a P-Grid. We expose in this section these services, while we will figure out the location of these services in the global architecture, as well as their mutual links in the section 4.

- storage: This service is responsible for the physical storage of the data on the P-Grid, regardless the reliability of the storage site. This service allows one user (or application) to store one piece of data on the system: The underlying storage solution may be a file system, a database or whatever can store data. The user is the source of authority (SOA) of this data, and controls the authorizations of access, replication, ... One can notice that all data may not be in the P-Grid, for performance or security reasons. For instance, it may become impossible to store all data issued by sensors since the throughput could be too large. Or some data, like medical data, may never be stored "as is" outside an hospital, for obvious confidentiality reasons but should be anonymized or encrypted beforehand.
- replication: Two main reasons push in favor of the replication service: performance and reliability. Adjusting the number of copies and their location to suit their usages (i.e. moving the replicas around) helps to increase the performance of the system and to decrease the user queries process time. The second motivation comes from the instability of the P-Grid storage sites. Increasing the number of replicas of a data allows to increase its availability, but decrease the control over its proliferation and its consistency. This is particularly important when dealing with volatile and mobile devices. Nevertheless, all data can not be replicated, and a judicious choice must be done according to the usage of the data, which must be monitored.
- cache: A cache service allows to keep in temporary storage space some accessed data or results of user queries. This reduces the cost of accessing data (it may be a network cost as well as a financial cost). The main difference with the previous replication service is the hidden property of a cache service, that acts transparently with respect to the users requests. It is a proxy between these and the data themselves. Collaborative caching must be deployed, so that several distributed cache services cooperate, using the semantics of the cached data. This collaboration increases the total space dedicated to the cache as well as the global hit-rate.
- access: This service has two tasks. First, it accesses the data previously stored by the storage service. Second, it realizes the access to data stored outside the P-Grid, in external databases, flat files or whatever. Doing

this, it must implement an integration mechanism between the different available data sources.

- indexation: The indexation service is responsible for maintaining a global link between a physical data and a logical representation of this data. This representation may be a logical name, or a more detailed semantic description contained in metadata. This latter, that is a semantic indexation of the data, allows to have a semantic view of the available data. This issue can help to group geographically data (or their replicas) which are dealing with the same themes. A second benefit of a semantic indexation is to optimize the replacement policies of the collaborative caches, keeping in those the hottest topics (the data related to the more accessed themes). Finally, it allows to group efficiently data to apply an adapted access control policy to a set of data.

- search: This service relies on the previous one to find data. But it must also allow for hybrid queries, that act on stored data as well as on computable data (i.e. data that can be extracted or generated from stored data, after processing).

- transport: Data transport service is managing the communication of the data in the P-Grid. It relies on the the local available infrastructure to deliver the data: It can be either FTP-like for classic data, streaming for multimedia data, it can use UDP or TCP over a UMTS, WiFi, Bluetooth underlying protocol. As multiple transport solutions may co-exist on one particular device, this service must choose which combination is the best choice for one communication.

- security: This service can be decomposed in several parts. Data security is fundamental in a P-Grid, given its high degree of dynamic and unreliability. Allowing one to access a piece of data is first to allow one to enter the device handling the data. This must be controlled via identification, then authentication. This is far from trivial when users can not be known everywhere, and when central CAs must not become single points of failure. Typically, a Single Sign On must be available in the infrastructure. When one is authenticated, then comes the problem of authorization. Data access must be controlled at a fine grain, at the closest to the data itself, to allow the SOA of the data to give minimal sufficient rights, and the sites holding the data must control the given permission on the fly. Finally, an encryption mechanism must be provided in order to increase the confidentiality and non disclosure of the data on the devices. The communication encryption is the problem of the transport service. *Related to security is also the history service, that will be detailed later, to allow for traceability in the users' usage of the data.*

- adaptation: The adaptation service is a key service for the data in a P-Grid. Before delivering the data to the final user, it must be adapted to the device of this users: First, it may not correspond to the physical characteristics of the device (for instance, the result is an audio file, but no audio output is available on the device; the result should be transformed to plain text). Second, the profile of the user must be taken in consideration (for example, the language of the result should be changed). Finally, the available transport service can not handle the result (if the result is a video, and Bluetooth is the only communication possibility, the bit-rate should be adapted accordingly). Adaptation services must be available on the P-Grid, or external and accessible through the P-Grid. Combining and chaining these services, using their operational semantic is the responsibility of the adaptation service.

- processing: The above adaptation service is a particular processing service. The user may initiate any available service on retrieved data, from visualization to anonymisation, for instance. This aspect is not directly related to data and will not be discussed further.

- computer-human interaction: The wills of the user, as well as his feedback regarding the usage of the system should be taken into account for presenting data and computation results. Non intrusiveness and intuition should drive the research in this field and interaction systems using all modalities of interaction (from visual to sound to tactile) be developed.

These different services must be developed as independent modules, if possible, and must not rely on a specific middleware : Indeed, a hand-held device will not have the same capabilities to embed a heavy middleware as compared to an office computer. Only their presentation (API, exchange protocols and messages format) must be compatible. We will discuss in section 6 which technologies can be used to achieve these constraints.

These modules must also be developed as distributed modules, if possible. Scalability issues lead to avoid single point of decision, that could become single point of failure. Providing distributed modules may increase the reliability of the system (if it exists a part of redundancy) and also increase the global performance of the system, balancing the load between several instances of one service.

To the minimal set of services dedicated to data management, expressed above, we express the need of underlying services, not specific to data management but useful for it.

- execution service: This service is responsible for the execution of services on the P-Grid. It realizes the interface with the scheduler, organizes and verifies the correct execution of the tasks.

- monitoring service: The monitoring service must be aware of the current state of the resources, from services to network to hosts. This service allows two benefits: to have a global view of the P-Grid, in terms of available services (and their state : working, idle, stopped, ...), hosts, network resources, data location, ... and to personalize this view according to the dynamic permissions and needs of the users.

- discovery service: Discovery is relatively important for mobile or transient devices. It permits these devices to know which are the available services in their neighborhood, so that interactions can be initiated.

- history service: This service is a memory of what occurred on the P-Grid. It has two main roles : First, it keeps a trace of the access to the services and the data, for traceability or financial reasons. Second, it allows to reconstruct a composition of services for one user (or a kind of user), in a given context, saving some time discovering the services or the data available. Techniques such a data mining in this history may be used for this issue.

The presented architecture is quite different from a Computing Grid, taking into account at each stage of the services development the instability, the mobility and the context-awareness of some members of the P-Grid. Conversely, it differs from a pure pervasive systems since it can use some stable infrastructure to store data or execute some compute-intensive services. It is not either a Peer2Peer architecture, since each node does not have the same role, and roles may be distributed dynamically.

4 Links, interactions, and today first steps

In a P-Grid, several infrastructures may co-exist, some being stable, some being unstable. It is thus necessary to identify which services must be deployed on the stable part, and which services may be deployed on the unstable part. Moreover, all services do not need to be embedded everywhere: Indeed, some devices can not afford to have too much services on them. The placement of the services in the architecture depends on a number of factors: its importance in the global architecture, the importance of its responsiveness, of the quality of the response... Moreover, the properties of these factors evolve with time and the available infrastructure. As an example, if a device is connected to the Internet, it can use distant web services while he can only use its surroundings in the other case.

For designing the global architecture, we start from the data end. It seems important to describe the data, and at two different levels: First, the data structure must be available so that the data access service can use mediators to find appropriate mappings between data sources. Second, data itself must be described, for instance for the semantic indexation service or the adaptation service. Semantic data description is maintained by metadata associated to data (in [25], we proposed to index the documents according to their content). We propose to keep these metadata in a database, one database by data source, being on the hosting machine of the data themselves. The accesses on these metadata (availability and access permissions) are those of the data. Moreover, the metadata are generally small compared to the original data, they thus do not increase the load of the data server. The metadata are extracted either manually (given at the same time than the data), or automatically using services available on the P-Grid. If the storage site do not have enough space or the user does not want to store these metadata for any reason, this latter solution allows to reconstruct dynamically the metadata.

One of the consumers of these metadata is the indexation service. This service is replicated on the P-Grid, to balance the load, to make it closer to the data and to increase the robustness of the system. One indexation service is responsible for a set of data sources, each service collaborating with others to have a larger view of the available data, wrt the limit of local index size. The indexing of data in the related indexation service can be either automatic a priori (when the data enters the data source, they register with the indexation service), or a posteriori. In this latter approach, developed in [21], the indexing is done when data is transmitted on the network: Indeed, active routers can examine passing packets and decide on their indexation. Unfortunately, this solution only indexes data being requested, and is complimentary to the a priori solution.

Once the data indexed, the search is facilitated since one must only contact the indexation services. A collaboration between them must be constructed so that they can exchange their content. We have shown in [25, 6] that a two-levels collaboration is suitable and is enough, allowing to exchange only a small amount of the data while still being efficient during the search.

The next step is to effectively access the data, and thus to secure this access. We must handle AAA (Authentication, Authorization and Accounting). First, we secure the access to the site itself. We use an authentication mechanism based on a distrust certification model [28]. The idea is, for a given user identified on a home site H to gain access to other sites thanks to a trust chain between these and the original site H. The user gets a home certificate and gain trust certificates during his visits. Each certificate embeds a profile, which is correlated with local permissions. Each device that may accept incoming visitors must host the authentication service of its site. Without any certificate, visitors access limited resources (default profile). Second, data access must be controlled at a fine grain, as close as possible to the data, in order for the SOA of the data to give minimal sufficient rights; the sites holding the data must control the given permission on the fly. We propose to use Sygn [30, 32] for this purpose. Sygn allows for a role based access control, and is based on certificate given by the SOA of the data to the potential user of this data. The process involves only these two users and do not need any stable infrastructure. Only an access control service must be deployed where the data are held, which controls the validity of the presented certificates according to the action being requested on the data. This control is only based, from the storage point of view, on the association between the identifier of the SOA of the data and the identifier of the data. One important characteristic is the necessity to have unique identifier of the data on the P-Grid: This can be achieved through the use of users public keys. These two mechanisms (authentication and authorization) use certificates:

These are generally small in size and can be embedded on mobile devices with the users (usb keys, ...). This is preferable to a central storage site, being a single point of failure and a privileged target for attacks. Nevertheless, we can imagine a proxy acting for the user as a cache when certificates are needed (for performance and ease of use reasons). Third, concerning the accounting, Sygn can be set so that every accesses to the data are stored together with the data. We believe the access to the data is the right location for this accounting, providing we store the identifier of the requester and his associated action. Finally, security is also achieved by encrypting data. An encryption service must be provided in order to increase the confidentiality and non disclosure of the data on the devices. This service can be embedded on devices, but some permanent services on the stable infrastructure must exist for those too short in computing resources. We propose to use CryptStore [31], which rely on the distribution of key shares on the network. This flexible solution allows the user to delegate the encryption/decryption service out of its device, while keeping a secure control of the encrypted data. The key shares to decrypt the data are spread on storage sites, where their access is controlled in the same way than the data. One advantage of this solution is the fact that the SOA does not need any key management procedure. Another advantage is the fact that only a parameterizable subset of all the key shares are necessary to decrypt the data, increasing the reliability of the system when some key servers are down or unreachable.

Once the data accessed, it can be interesting to create replicas of these, or more simply to cache them. The replication service presented in [15] needs monitoring services to optimize the placement of the replicas. This monitoring must give up-to-date information on the status of network resources, and aggregate these so that it reflects the cost to get a file. On the other side, the cache service [6] needs the semantic indexation service presented above to adapt the replacement policies to the data usage. It also handles metadata thus it needs a database to keep those.

Before delivering the data to the user, it can be adapted to his context. This adaptation (or any treatment on the data), will use services available on the P-Grid. It needs to interact with a middleware that will launch, execute and monitor the services. Moreover, the adaptation service [3] is based on the description of available services and on the adequate representation of the context. More generally, a service to discover and register available services is mandatory and should exist on every device. This local service interacts with other services on the stable infrastructure, and with its surroundings to discover the locally available services.

From the user point of view, only a discovery service and a set of certificates must be present with him. These will allow him (and, by delegation his device) to reconstruct part by part his (its) view of the P-Grid.

5 Use case

Mr Smith lives in Lyon, France, and leaves a few days to Vienna, Austria. Unfortunately, on the spot, a car runs up against Mr Smith, who remains on the road, unconscious. The helps arrive. Only one information is available on Mr Smith: a single numerical identifier (for example on a smart card), allowing the helps to quickly reach its medical file (of which at least a part is in France, perhaps in Lyon), to find there important information (in this situation, it is for example the allergies to the drugs, of its operational antecedents - was already anesthetized, with which product? did he have an operation, there exist an operational report, radiographies?) which will allow as well as possible to adapt the care to bring to him.

The simple case presented here remains however limited and does not approach all the facets of a medical P-Grid (like its use in epidemiology, biomedical research, genic medicine, etc...), but is enough to illustrate a certain number of research topics. We will now show how the P-Grid described before can help in this situation, and which services would be used here.

I first detail the conditions of the usecase. First of all, let suppose that Mr Smith has a health card, on whom is indicated his identifier. I adopt the approach unrealistic (idealistic?) for the moment that the problem of the single identifier is solved (this particular point is a subject political, ethical, and who is far from being solved, even at the European scale). His medical data are distributed, scattered on the territory in various care centers (a public care network made up of a certain number of hospitals), and in a private clinic not connected to this network. The contents of the medical file must be accessible in a protected way: Only the accredited people can reach certain parts of the file, and ideally the accreditations are given by Mr Smith himself. All the existing written health documents are in French. In the private clinic, the documents are stored in audio format (operational report of anesthesia...). The Austrian first-aid worker speaks German, has a Palm, with WiFi connection. The WiFi hot spot is in the ambulance and allows the first-aid worker to consult the medical file of the patients through the network of the hospitals.

The intervention of the first-aid worker begins on the spot of the accident and continues on the road towards the hospital. When the first-aid worker wants to recover allergy to certain products concerning the patient, he should initially know where these information are. From both the identifier of Mr Smith and the request itself (allergies?), the system seeks, thanks to the service of indexation, the storage centers likely to have these information.

The first-aid worker must then contact these centers. He should obtain authorizations to enter French information systems: He obtains them starting from his certificate of membership of the health Austrian system. Trust certificates are established for him to access the network of care, and for the access to the private clinic (by inevitably the same trust certificates, because there may exist different mistrust policies towards the Austrian care network).

The integration service transforms the request of the first-aid worker towards the schema of the databases in the centers of care, and negotiates, according to the profile and of the presented request, the parts of the database accessible to him. The request of the first-aid worker is expressed by using a common vocabulary (ontology of the medical field) gumming the problems of language. This work has already been completed in the network of care, which already did this in-house work at the time of the creation of the network and which thus presents a coherent schema. The link between the network of care and the private clinic allows for an automatic integration between the different schema (if the automatic integration is not possible, the first-aid worker could not be of a great help, the schema of the bases being probably expressed in French).

To reach the data themselves, the first-aid worker must present to the access control (Sygn) the mandatory certificates to read the file. However, this information is controlled by Mr Smith, who must issue an access certificate himself (it is the principle of Sygn). Here, he is unconscious, therefore he cannot. Mr Smith must have previously created certificates for standard accesses to some of his data: For example, the people being able to endorse the first-aid worker role can read the allergies to the drugs. A repertory of the standard certificates of Mr Smith must be accessible on line. This one is retrieved by the first-aid worker from the identifier of the patient to find appropriate certificates. The first-aid worker presents these access certificates to Sygn which authorizes the access and returns the data.

The metadata of the documents are analyzed to know their nature and to see how they can be exploited by the Palm. An adaptation service is probably required, to create a chain of transformation from the original documents (in French, writing and spoken) to exploitable documents for the first-aid worker in a disturbed environment where one can only read and not listen: a service for audio-to-text transformation, a French-German translation service. Finally, the first-aid worker reaches the relevant data and gives the first drugs to the patient.

During the transport of the patient, the system transmits to the hospital collected information (drugs, known allergies, identifier of the patient). In the hospital, even before the arrival of the ambulance, a surgeon can recover, with the same idea but with different conditions (less constrained terminal of consultation, higher role in the care network...), more complete information (operational antecedents, scanner...) in order to be able to intervene judiciously as fast as possible.

In the scenario, the first-aid worker is very active, interacting with the global infrastructure. One should understand that much of the tasks should be automated, delegated and performed transparently by his device.

6 Implementation issues

The architecture described in the previous sections is not yet working as a whole set. Nevertheless, efforts in our group for implementing parts of it have already been done and could be successfully reutilized and integrated in a P-Grid: Some works [3, 25, 28, 15] have been initiated in the context of pervasive systems, others in the context of data grids [31, 30, 32, 6].

In the case of the data grids, we have developed our contributions using standard middlewares like Globus [13], conforming to the OGSA architecture [1]. OGSA offers the desired functionalities of the components of the architecture, the infrastructure being assured by WSRF (Web Service resource Framework). On the data side of the architecture, OGSA does not offer an integrated view by rather a set of tools and services to access the data (OGSA-DAI-Data Access and Integration), to handle the replicas (RMS - Replica Management Service), to transfer the data (GridFTP). Moreover, some activities that are directly related to data management are studied as different fields, like for instance the information service (that can provide information on data sources, storage resources, ...). Some problems we face using OGSA is the heaviness of the protocols and the architecture. As an example, in WSRF, the service calls are encapsulated in SOAP, that adds a large amount of payload to the service. In a light device infrastructure, the cost of this protocol can be seen as a potential bottleneck. Moreover, it is difficult to separate efficiently the services in OGSA to manage a light architecture, embeddable on light devices with poor memory and network connections. Finally, OGSA does not deal with mobility, locality, context-awareness in its development. But it can be a relatively good starting point as [7] shown for mobility, and since it is widely adopted and standardized. But one would have to adapt the services to pervasive environment, and to work on the effective modularity of the architecture.

On the pervasive systems side, we have developed a software platform called PerSE [29, 4] that can host pervasive services. It is user oriented, and is based on the principle of service composition to handle user wills (the users describe partial actions that are transformed in fully described actions based on the context of use). Available services can be constructed outside PerSE, only their interface (messages, protocols) have to be known in PerSE for the composition. The core PerSE base is small, and can be extended using security or log services, independently developed and attached on demand to PerSE. Efficiency and development reactivity is a key concept in the construction of PerSE, thus it does not rely on well established standards. The drawback is the limited easiness to distribute and extend the PerSE base as is.

In our P-Grid approach, we plan to take some ideas from the two worlds, with the flexibility of PerSE to construct services and chain them, and the ambition of OGSA for its wide adoption and standardization process. The best overall approach is probably to start for OGSA, and adapt it with the PerSE ideas (and others related to pervasive systems) in mind.

7 Conclusion and Future works

In this article, we first gave our vision of a Pervasive Grid, from the data side point of view. We exhibit existing solutions and show their limitation, particularly for the data management side. We then identified a number of services useful in this context, and we explained their respective roles. We tried to organize these and to describe their mutual links and the links with an underlying infrastructure, as well as some already developed bricks. We illustrated our approach with an use-case before giving some trails for a real implementation.

As future works, we believe that such a P-Grid can be constructed out of the more detailed specification of the architecture, that should be the first step onward. More specific interests include the proactive aspect of the pervasive systems, and the semantic handling of all the parts of the proposal, from data to services.

References

[1] OGSA : Open Grid Services Architecture. www.ggf.org.

[2] Francoise Baude, Denis Caromel, Fabrice Huet, and Julien Vayssiere. Communicating mobile active objects in java. In Roy Williams Marian Bubak, Hamideh Afsarmanesh and Bob Hetrzberger, editors, *Proceedings of HPCN Europe 2000*, volume 1823 of *LNCS*, pages 633–643. Springer, May 2000.

[3] Girma Berhe, Lionel Brunie, and Jean-Marc Pierson. Content adaptation in distributed multimedia systems. *Journal of Digital Information Management, special issue on Distributed Data Management*, 3(2), June 2005.

[4] Pascal Bihler, Lionel Brunie, and Vasile-Marian Scuturici. Modeling user intention in pervasive service environments. In Laurence Tianruo Yang, Makoto Amamiya, Zhen Liu, Minyi Guo, and Franz J. Rammig, editors, *EUC*, volume 3824 of *Lecture Notes in Computer Science*, pages 977–986. Springer, 2005.

[5] Dario Bruneo, Marco Scarpa, Angelo Zaia, and Antonio Puliafito. Communication paradigms for mobile grid users. In *CCGrid 03*, page 669, 2003.

[6] Yonny Cardenas, Lionel Brunie, and Jean-Marc Pierson. Uniform distributed cache service for grid computing. In *Globe'05, Proceedings of DEXA'05*, Copenhagen, Danemark, August 2005.

[7] D. Chu and M. Humphrey. Bmobile ogsi.net: Grid computing on mobile devices. In *Grid Computing Workshop (associated with Supercomputing 2004)*, Pittsburgh, USA, November 2004.

[8] B. Clarke and M. Humphrey. Beyond the "device as portal": Meeting the requirements of wireless and mobile devices in the legion grid computing system. In *2nd International Workshop on Parallel and Distributed Computing Issues in Wireless Networks and Mobile Computing (associated with IPDPS 2002)*, Ft. Lauderdale, USA, April 2002.

[9] Nigel Davies, Adrian Friday, and Oliver Storz. Exploring the grid's potential for ubiquitous computing. *IEEE Pervasive Computing*, 3(2):74–75, 2004.

[10] Yamin et al. Towards merging context-aware, mobile and grid computing. *International Journal of High Performance Computing Applications*, 17(2):191–203, 2003.

[11] I. Foster and C. Kesselman, editors. *The Grid : Blueprint for a New Computing Infrastructure*. Morgan Kaufmann Publishers, Inc., 1999.

[12] I. Foster, C. Kesselman, J. Nick, and S. Tuecke. The physiology of the grid: An open grid services architecture for distributed systems integration, 2002.

[13] Globus Project. Globus toolkit. http://www.globus.org/, 1998.

[14] Francisco J. Gonzalez-Castano, Javier Vales-Alonso, Miron Livny, Enrique Costa-Montenegro, and Luis Anido-Rifo. Condor grid computing from mobile handheld devices. *SIGMOBILE Mob. Comput. Commun. Rev.*, 7(1):117–126, 2003.

[15] Julien Gossa, Jean-Marc Pierson, and Lionel Brunie. Fredi flexible replicas displacer. In *International Conference in Networking*, Mauritius, April 2006. IEEE CS Press.

[16] Piotr Graboswki, Bartosz Lewandowski, and Michael Russell. Access from j2me-enabled mobile devices to grid services. In *Proceedings of Mobility Conference 2oo4*, Singapore, 2004.

[17] Vipul Hingne, Anupam Joshi, Tim Finin, Hillol Kargupta, and Elias Houstis. Towards a pervasive grid. In *International Parallel and Distributed Processing Symposium (IPDPS'03)*, page 207, 2003.

[18] Steve Tueke Ian Foster, Carl Kesselman. The anatomy of the grid: Enabling scalable virtual organizations. In *International Journal of Supercomputing Applications*, 2001.

[19] Sheng Jiang, Piers O'Hanlon, and Peter Kirstein. Moving grid systems into the ipv6 era. In Lecture Notes in Computer Science (LNCS) 3033, editor, *Proceeding of Grid and Cooperative Computing 2003*, pages 490–499. Springer-Verlag Heidelberg, 2004.

[20] Stanislav Kurkovsky, Bhagyavati, Arris Ray, and Mei Yang. Modeling a grid-based problem solving environment for mobile devices. In *ITCC (2)*, pages 135–. IEEE Computer Society, 2004.

[21] Laurent Lefevre, Jean-Marc Pierson, and Sid-Ali Guebli. Deployment of collaborative web caching with active networks. In *International Workshop on Active Networks, IWAN 2003*, pages 80–91, Kyoto, Japon, December 2003. Springer Verlag, LNCS 2982.

[22] Lee McKnight, James Howison, and Scott Bradner. Wireless grids, distributed resource sharing by mobile, nomadic and fixed devices. *IEEE Internet Computing*, 8(4):24–31, July 2004.

[23] Sang-Min Park, Young-Bae Ko, and Jai-Hoon Kim. Disconnected operation service in mobile grid computing. In *First International Conference on Service Oriented Computing(ICSOC'2003)*, Trento, Italy, December 2003.

[24] Thomas Phan, Lloyd Huang, and Chris Dulan. Challenge:: integrating mobile wireless devices into the computational grid. In *MobiCom '02: Proceedings of the 8th annual international conference on Mobile computing and networking*, pages 271–278, New York, NY, USA, 2002. ACM Press.

[25] Jean-Marc Pierson, Lionel Brunie, and David Coquil. Semantic collaborative web caching. In *Web Information Systems Engineering 2002 (ACM/IEEE WISE 2002)*, pages 30,39, Singapore, dec 2002. IEEE CS Press.

[26] Akogrimo Project. www.akogrimo.org.

[27] GridLab Project. www.gridlab.org.

[28] Rachid Saadi, Jean-Marc Pierson, and Lionel Brunie. Apc : Access pass certificate, distrust certification model for large access in pervasive environement. In *Proceedings of International Conference on Pervasive Services IPCS'05*, Santorini, Greece, July 2005.

[29] Vasile-Marian Scuturici, Jean-Marc Pierson, and Lionel Brunie. Perse : Pervasive services environment, research report, 2005.

[30] Ludwig Seitz. *Conception et mise en oeuvre de mcanismes scuriss de partage de donnes confidentielles : application la gestion de donnes biomdicales dans les grilles de calcul*. PhD thesis, INSA de Lyon, July 2005.

[31] Ludwig Seitz, Jean-Marc Pierson, and Lionel Brunie. Encrypted storage of medical data on a grid. *Journal Methods of Information in Medicine*, 44(2):198–202, 2005.

[32] Ludwig Seitz, Jean-Marc Pierson, and Lionel Brunie. Sygn: A certificate based access control in grid environments. Technical Report 2005-07, LIRIS, July 2005.

[33] S.H. Srinavisan. Pervasive wireless grid architecture. In *Second Annual Conference on Wireless On-demand Network Systems and Services (WONS'05)*, 2005.

[34] Oliver Storz, Adrian Friday, and Nigel Davies. Towards 'ubiquitous' ubiquitous computing: an alliance with 'the grid'. In *First Workshop on System Support for Ubiquitous Computing Workshop (Ubisys 2003) in association with Fifth International Conference on Ubiquitous Computing*, Seattle, Washington, U.S., October 2003.

[35] Mark Weiser. The computer for the 21st century. *Scientific American*, 265(3):66–75, February 1991.

[36] G. Zhang and M. Parashar. Dynamic context-aware access control for grid applications. In *4th International Workshop on Grid Computing (Grid 2003)*, pages 101 – 108, Phoenix, AZ, USA, 2003. IEEE Computer Society Press.