Media grids for scalable semantic computing and intelligence on distributed scenarios

Journal:	*IEEE MultiMedia*
Manuscript ID:	Draft
Manuscript Type:	Regular
Date Submitted by the Author:	n/a
Complete List of Authors:	Nesi, Paolo; University of Florence, Dept. of Systems and Informatics Bellini, Pierfrancesco; University of Florence, Dept. of Systems and Informatics Bruno, Ivan; University of Florence, Dept. of Systems and Informatics; University of Florence, Dept. of Systems and Informatics Cenni, Daniele; University of Florence,, Dept. of Systems and Informatics; University of Florence, Dept. of Systems and Informatics
Keywords:	M.8 Business Grid < M Services Computing, D.2.9 Management < D.2 Software Engineering < D Software/Software Engineering, D.3.2.r Scripting languages < D.3.2 Language Classifications < D.3 Programming Languages < D Software/Software Engineering

Media grids for scalable semantic computing and intelligence on distributed scenarios

Pierfrancesco Bellini, Ivan Bruno, Daniele Cenni, Paolo Nesi

Distributed Systems and Internet Technology Lab, Department of Systems and Informatics, University of Florence, Italy, http://www.disit.dsi.unifi.it/, nesi@dsi.unifi.it,
tel: 0039-0554796363, fax: 0039-055-4796363

Abstract

The micro grid technology is playing a central role in the semantic evolution of small and medium size services. The article presents the main requirements and architectures of micro grids for content and semantic computing. These flexible applications allows to enforce new functionalities and intelligence into complex and integrated solutions by supporting automated content processing and management back office, controlling content delivering networks, providing services via active and proactive applications on client devices. To this end, they have to provide high flexibility and scalability and a range of features possible to be provided in the past only by large integrated platforms, which were unaffordable for small and medium service portal providers. A solution can be to enforce semantic computing into micro grid by using media language for formalizing processes to produce, post-produce, licensing and delivering content by satisfying a large range of scenarios. Semantic computing primitives may be also enforced into intelligent tools and players to bring into the hands of the final users more features and capabilities, such as local semantic search, recommendations, and therefore, a real personal assistant. Those clients are not passive tools and play a role in the general grid architecture. The experience reported has been gained in several years of work in the sector.

Introduction

The recent challenges on media services have lead software engineers to identify and integrate in a scalable manner multiple technologies. Examples are large scale services such as social networks, content delivering networks, archives, encyclopedia, e-commerce portals and other service networks. In most cases, large-scale media services are managed by large companies and/or institutions allowing a number of simultaneous accesses and millions of delivered content items. Recently, on the spur of final users via high-speed networks, also many small and medium companies/entities are moving toward setting up similar services. This trend is also supported by the decrement of computational and storage costs, making possible the exploitation at reduced costs of semantic computing and grid-based multimedia computing tools and solutions. On such grounds, the augmentation of new and more intelligent services and the scale up of their size are becoming key issues for many services and content providers. Subsequently, the business of multimedia content management and distribution is becoming more accessible for a large number of actors.

The typical small-scale applications, where new technologies of scalable semantic computing are pervading, are public digital archives, best practice networks, service/content distribution, thematic social networks, e-commerce portals, distributed monitoring systems, content repurposing processes, etc. They are characterized by a limited number of users and of content items with respect to the global scale businesses.

On the other hand, the insertion of semantic computing can enforce new functionalities and business models, without the need to have a critical mass to perform large investments. The augmented services can become attractive to compete with huge service providers. Most of them, share common requirements for: media adaptation, repurposing, indexing; workflow management; semantic computing; content management, distribution, streaming, quality of services, control delivering networks; intelligence on client devices; integration with delivering networks, etc. Therefore, the insertion of semantic computing in an automated back office is becoming an urgent need for low-costs distributed multimedia solutions. They have to cope with several of the above-mentioned features with a certain degree of integration into the back office infrastructure model and language, and have to be capable to work with some degree of intelligence on client side so as to exploit the resources efficiently, and providing high quality services.

Most small and medium media services need to be continuously updated to follow the trends augmenting, changing and providing new functionalities/services for their user communities. Examples of this kind of service reshaping/tuning can be:

- production of additional kinds of recommendations (advertising, content to users, user to users, group to users, content to groups, etc.),
- insertion of additional reasoning on metadata, use data and/or profiles,
- changes in similarity distances and clustering algorithms,
- production of content on demand for new devices,
- insertion of full text indexing in several languages, reindexing of the database, etc.
- gradual insertion of semantic computing in running service infrastructures,
- insertion and/or tuning of user generated content processing workflow,
- changes in business and transaction models, licensing and trusting,
- integration of additional distributive channels, and thus changed into the content delivering network,
- review of datamodel to cope with additional metadata for additional services (insertion of GPS coordinates, insertion of barcode readers, insertion of RID readers, ..), etc.,
- addition of more detailed analysis on user behavior.

The areas of production and distribution are frequently kept separate in traditional content factories, whereas have to be integrated or even joint together in services including and processing User Generated Content (UGC), or in services making content enrichment; as well as where content repurposing/adaptation and delivering phases have to be performed automatically like it occurs with mobile devices [InsightReport]. In most cases, to enforce some of the new features a stricter integration of back office and distribution phases is needed; the latter is typically performed via some the Content Delivering Network, CDN [Vkali2003].

The exploitation of most of the above-mentioned functionalities both in single processes and/or in integrated environments is mandatory when phases of multichannel content reception, production and delivering have to be efficient and harmonized. In such events, it is very important to reduce communication costs, to remove bottlenecks and critical points, and to exploit functionalities without communication overhead to move content. In such cases, programming and coding are needed at system, business logic and process levels. To this end, old services have to be updated to cope with dynamic changes and allocations, or new services have to be designed; flexible procedures/processes have to be dynamically allocated on computational resources. In most cases, technical solutions are based on setting up custom processes on dedicated computer systems during restructuring and integration. This means, creating solutions that are difficult to be reconfigured to cope with further micro changes whenever requested. The business logic may be transferred into the processing units in a much faster manner than moving the huge data sized content. This may allow to reduce costs in small and medium size solutions, yet being not more difficult at all when it comes to large global scale systems since in those cases the movement of data has to be in any way performed [Li2009].

Among the technologies supporting flexible management of the media-based back office management we can see the grid [Foster2002], the Service Oriented Architectures (SOAs) with web services (W3C WSDL), media processing languages, parallel distributed processing, software agents, cloud computing, and obviously the semantic computing technologies. In order to cope with the automated content management on parallel and distributed architectures, the grid technologies can be the underlining solutions to manage distributed resources such as: industrial computers, dedicated servers, storage systems, playout servers, communication channels, etc. Grid solutions provide capabilities for both service negotiation and activation management via web services by using Service Level Agreements (SLAs) [Leff2003], as well as for resource discovery and control [Roure2005].

Grid based solutions can be exploited for managing and harmonizing globally distributed services via internet, as well as via local back offices and distributed resources in some intranet. They can cope with dynamic allocations of services, thus exploiting flexibility in order to reduce costs and increase efficiency. Grid solutions may be combined with cloud computing and virtualization technologies to optimize the resource consumption, see for example with some distributed power management and scheduling. The grid and cloud computing can be solutions to make those services scalable, with the dynamic allocation of new or different processes, while they are not effective to cope with flexibility in functionalities and procedures. Cloud computing is not enough to cope with the reshape of business processes, models and procedures for the production and distribution workflows, since changes are at functional level.

To cope with these problems, some media and multimedia grid solutions try to provide solutions for a number of areas. For example: mmGrid [Basu2003] and MediaGrid.org [Walsh2005] have been developed to improve the interactivity on classical and/or 3D remote interfaces; MediaGrid [Huang2007], and AXP2P [Bellini2007] are representative solutions for P2P-based CDN; GridCast [Harmer2005] proposes a grid based solution supporting broadcasters on their distribution. In AXMEDIS (automating production of cross media content for multi-channel distribution) [Bellini2006], [AXMEDIS], an infrastructure for media grid has been designed and developed in the context of a large international integrated project of the European Commission. The framework has been designed with the aim of reducing costs by increasing efficiency and flexibility for content management and computing for multichannel CDN, exploiting DVB and IP distributions towards PC, STB and Mobiles, but also for traditional physical media production and distribution. Local computational media grids able to cope and optimize back office problems are typically called *micro grids*. In [Volckaert2008], concepts of macro and micro grid have been recalled in the context of a media grid architecture via a simulation. Therefore, *micro grid* solutions have to be capable to manage local resources such as computational power, storage and functionalities at service of front-end services and larger applications or global grids. In the context of this paper, as to media grid we intend a scalable grid architecture where processes may implement integrated functionalities coming from the areas reported in Table 1, which reports the main aspects of the solutions reviewed, as reported in the following box.

	AXMEDIS	MMGRID	MediaGrid	GridCast	MediaGrid.org	Omneon MediaGrid
Content Ingestion, UGC management	X					
Content Management	X		(X)	X	X	X
Content Computing	X		(x)	(x)	(x)	
Content Storage Management	(X)			(X)		X
Semantic Computing	X					
General Communication support	X	X	X	X	X	
Content Delivery Network Management	X		X	X	X	
User Interaction Support, rendering	X	X			X	
Global or Local grid	L/(G)	G	G	G	G/L	L

Table 1 – Media grids and coverage areas; (x) means partial support, G as global, L as local, X full support. All the mentioned solutions support the distribution processes.

Box on Media Grid

MmGrid [Basu2003] has been designed to support interactive applications with graphics, rendering, streaming, and tele-immersion. The aim of mmGrid is the organization of a resource pool for providing interactive services to users accessing the grid via an http-based interface. **MediaGrid.org** [Walsh2005] consortium proposed a grid solution to address computational power to support of 3D and virtual reality rendering and distribution. In some cases, the grid solution is also pervading client devices such as in IPTV clients, based on P2P or other. Thus, **MediaGrid** in [Huang2007] is a P2P solution to share videos for IPTV. The MediaGrid P2P network monitors the conditions and tunes the services maximizing network capacity. It is a solution to set up Content Delivering Networks according to what is commonly called P2P progressive streaming. In [Ishaq2009], the concept of grid is used to set up a network of nodes organizing services via DHT (Dynamic Hast Table), called the Semantic Grid Service Registry. Services are searched by using the classical DHT-based P2P algorithm in [Xia2005]. **GridCast** [Harmer2005] is a service-oriented architecture for broadcasting media via IP. It presents some content management capabilities including those related to scheduling broadcasting and subsequently controlling playout servers for broadcasting. Most of the high-level activities may be demanded to a workflow management system, which also plays the role of integrating other tools via programming. The GridCast solution is mainly tuned to cope with broadcasters' content management problems, to cope with MXF, AAF, etc. [MXF]. GridCast has been developed on the basis of Globus Toolkit [Globus], and it has some capability for setting up a geographical grid infrastructures rather than for local massive parallel architecture organization on cloud computing. **Omneon MediaGrid** [Omneon] proposes a grid powered scalable content storage. In [Roure2005], a review of semantic grid solutions has been provided taking into account general implications and historical evolution. Semantic grids are suitable solutions to cope with user communities, parallel processing, taking decisions and reasoning on descriptors and content usage; and in most cases, they are based on web services, software agents, descriptors, inferential engines for semantic computing activities. Geographical grid solutions based on OGSA (Open Grid Service Architecture), such as Globus [Globus], fit this model.

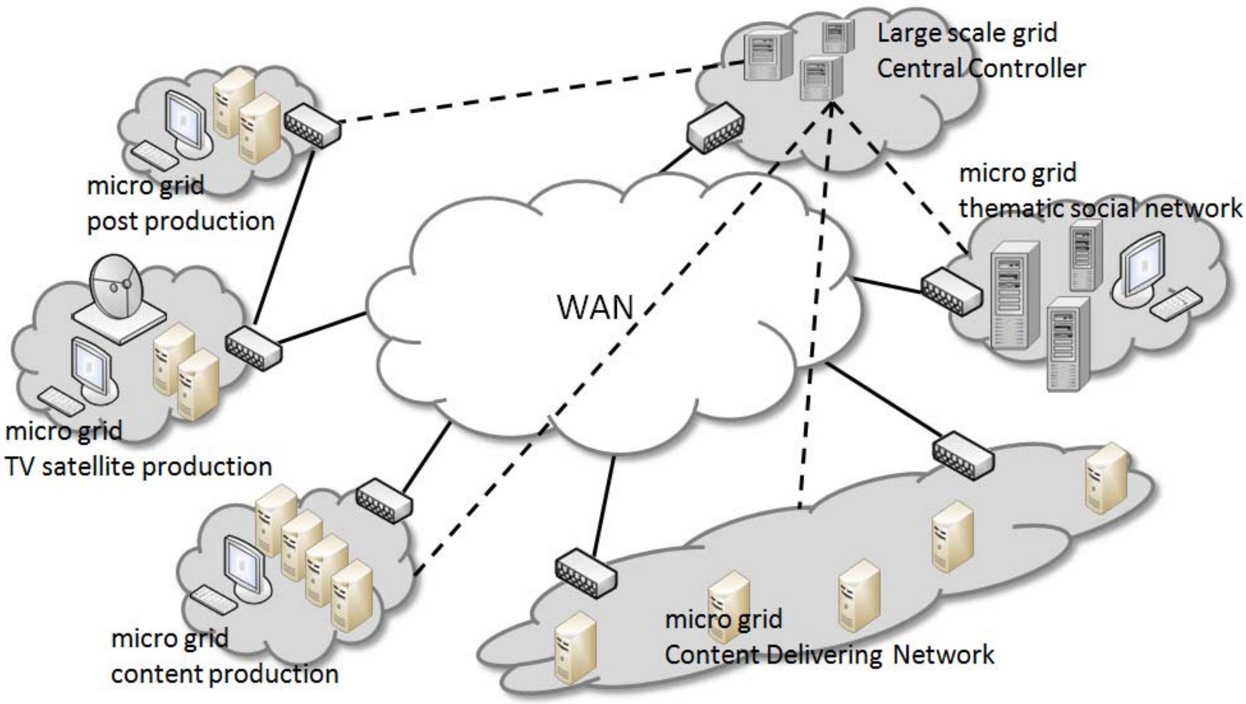

Figure 1 – A Large-scale grid and some related micro grids.
Dashed lines represent logical controls dependencies dynamically established.

General requirements for micro grid infrastructures for media computing

Micro grids are the essential operative arms of large-scale grid solutions (see Figure 1). Micro grids typically manage the local computational power and receive instructions and commands from the higher-level grid services negotiating the service by means of some SLA with the requesters. Micro grid can be specialized architectures to serve one or more large-scale grids. Connections are dynamically performed according to temporary conditions: the addition of new services, the adoption of a new CDN, etc.

In small and medium sized service applications local and ad-hoc micro grids can be solutions for leveraging the functionalities of present solutions, without the need to be at the service of large scale grids, but totally dedicated to server local applications/portals and services. Their flexible exploitation of computational power and micro functionalities may allow coping with the computational needs provoked by the insertion of new general functionalities, with respect to the computational costs of the operative activities. For example, the whole re-indexing of large database (or a global clustering, or the global re-classification, or massive video transcoding of the original formats) may be very computationally intensive. On the contrary, the day by day indexing/clustering process can be much lighter since incrementally applied only to daily coming content and information, and thus, the grid may be scaled down dedicating resources to other services. Thus, for small medium enterprises, the usage of physical hosts for specific actions is typically much more expensive than using multipurpose grid nodes cloned from the same virtual machine which can be used to allocate a large range of different kind of processes.

In order to contains the costs, the micro-grids infrastructures for media content management, production and multichannel distribution have to provide the needed flexibility to integrate a large set of functionalities, tools and technologies. Moreover, they have to provide basic structural facilities and support for the following main features.

- **description of grid processes –** Each single grid process has to provide a formal description of the resources needed to execute successfully the process and scheduling conditions. For example, the list of needed functionalities (via plugins or other means) and tools, the CPU capabilities, the HD space, the network connections, the description of scheduling parameters (e.g., period, deadline), etc. This information is used by the micro grid scheduler to find a suitable node for the process execution [Foster2000].
- **discovering and describing resources –** The grid scheduler has to be capable to discover resources in local or global networks, for example via different protocols. in turn, resources have to respond providing their descriptors for grid, grid nodes and micro grids. Once discovered, the resources (computational and/or data storages) can be centrally managed by the grid scheduler. Resources may be located in industrial computers, virtual clusters, dedicated servers, SAN/NAS, as well as portions of desktops. In fact, in small medium enterprises, also portions of personal computers are relevant resources that can be reserved for the micro grid. For example, according to the typical usage and assigned hours based on each day of the week. Thus, for a desktop, it could mean to reserve to the grid the 30% of the CPU during the working hours, and the 100% else ware. Resource management, also means to estimate, monitoring and taking into account the status of grid resources in terms of free HD space, CPU speed, spare CPU clocks, available plug ins and tools, available connections and their throughput, etc.
- **allocation/activation of processes** on physical and/or virtual resources -- The allocations and schedule of activities on the grid have to provide buffering support to collect execution requests, and reservations. Processes may be programmed as periodic or sporadic to be dynamically scheduled by the grid scheduler according to the resources available on the grid, and according to their deadline and execution time. The activation of periodic processes on the grid is typically performed from the grid user interface, while sporadic processes are typically invoked by external communication channels via web services. Both models can be adopted by workflows management systems, web portals, and/or other grids. The grid processes may be allocated on other grids or micro grids; thus supporting hierarchical organization of grid processing where a grid at higher level may ask other micro grids to execute some tasks, and allocating them on several nodes.

The solution has to be scalable, it has to be work in a reasonable manner in terms of reaction and processing time, even when the accesses number (e.g., requests performed per day, per minute), and the number of items are in the range of millions. In this context, "reasonable" means at limited costs and with costs proportioned to be business. Facilities related to these aspects are those connected to the exploitation of physical hosts used by t he grid nodes and their resource optimization such as power, HD space, CPU, network.

Micro grid has also to support **safeness and reliability** with the possibility of setting up fault tolerant solutions and services: for example defining replica of processes, data and/or of the whole micro grid at the service of some higher-level grid. Safeness and reliability have to include the errors and exceptions monitoring and management, which may occur at all levels: on grid nodes, in the network, in the scheduler process execution. The reliability also implies to continuously control the micro grid behavior by verifying the status and liveness of micro grid resources. Grid resources may also change their profile/capability over time and thus the allocations may need to be adjusted in real time to new conditions. On this regard, in order to set up fault tolerant solutions, the single nodes can be quite easily used to set up redundant solutions; for example, by executing the same process at the same time on different grid nodes. On the other hand, the Scheduler is a critical point being a unique center of control for the whole micro grid. To this end, fail over solutions can be set up among a set of Schedulers, for example, by keeping active only one at a time while the others are ready to take the role of the Scheduler in the case of failure.

In terms of **security and trust,** the micro grid has to guarantee the security of processes and data on the grid infrastructure, as well as of those distributed and communicated in/out of the grid. This can be performed via standard protocols based on authentication and/or certificates with credential verification. Security aspects have also to be addressed at process programming level since in most cases the single process may cope with content and data protection and certification, and also with protected content that may be accessed only according to some specific authenticated and certified tools on the basis of formal licenses, such as in Digital Rights Management, DRM, models [Rosenblatt2003].

The Micro Grid Architecture

The architecture for media micro grid may presents five main areas (see Figure 2): resources, grid nodes pool, grid scheduler, request manager and the integrated development environment.

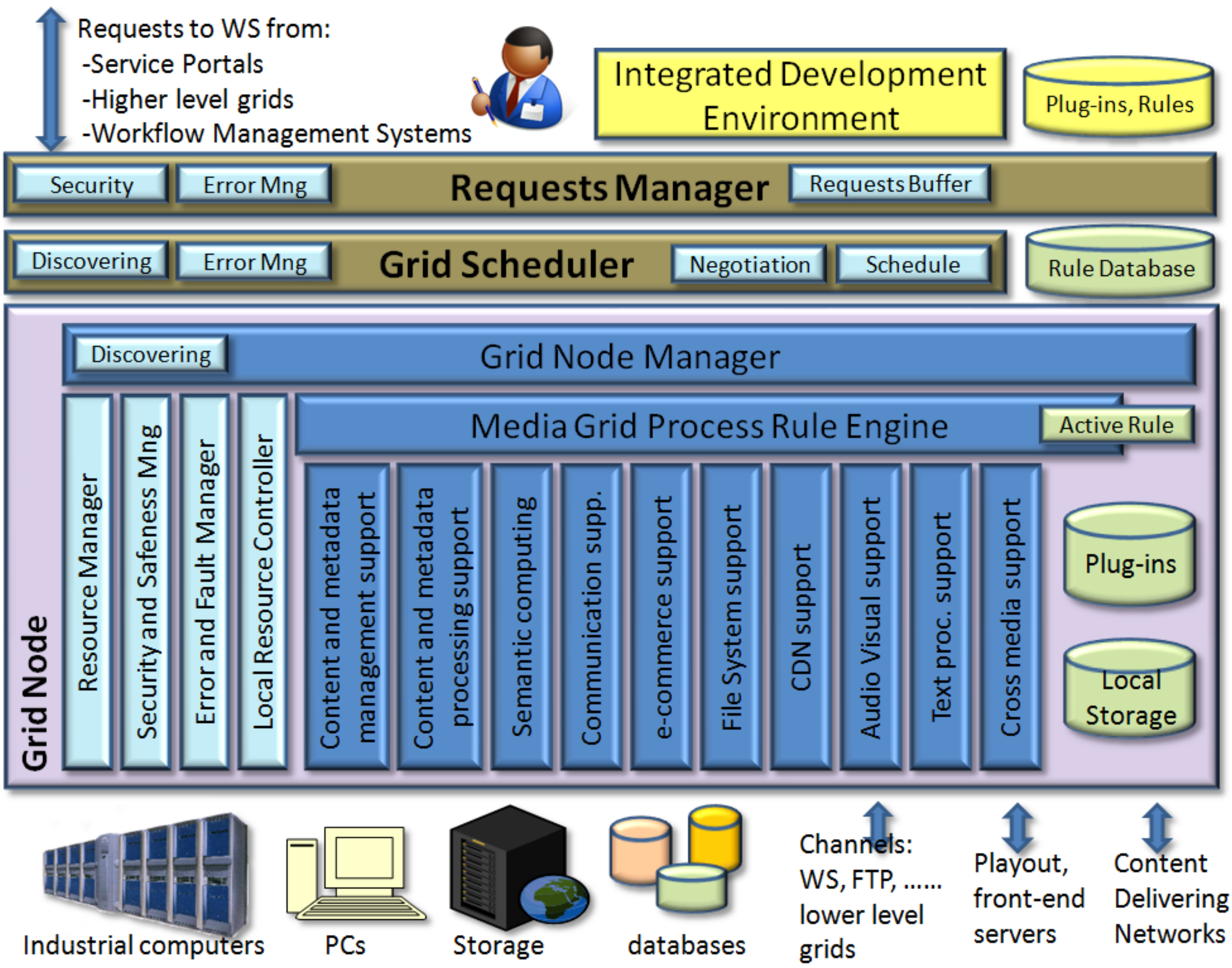

Figure 2 – Micro Grid architecture, five main areas: resources, Grid Node, Grid Scheduler, Request Manager and Integrated Development Environment.

The **Integrated Development Environment** consists of a set of tools to produce processes to be allocated on grid nodes. In many grids, executions are totally demanded to external executable processes developed for the hosting operating system. The adoption of compiled processes for nodes implies a lower flexibility since the business logic of the grid process is enforced into binary monolithic routines. The development environment may also be used to formalize the grid process descriptor to formalize the resource needed and the general behavior. In the AXMEDIS micro grid, processes can be external executable or directly formalized as Process Rules in the Media Computing Language, which is a JavaScript-based language extended with thousands of functionalities to cope with content, metadata, semantic computing, databases, distribution channels, CDN, etc. It provides a classical programming environment or a visual one where procedures and processes are represented via visual elements which can be combined [Bellini2009].

The micro grid programming facility has to provide a certain degree of flexibility. Therefore, the micro grid programming model and processes should have to:

- support a large range of functionalities for media processing and semantic computing. Media grid processes need to get access to functionalities coming from different areas: content and metadata ingestion, management, processing; semantic computing: taking decisions engines, inferential engine, descriptors processing; communication capabilities; database management, query activation; content delivering, managing CDN; workflow integration. These functionalities may be accessible as native API of a scripting language or libraries and/or as executable tools. In the latter case, a certain reduction of flexibility is obtained.
- be capable of interpreting a light processing language for scripting the grid process logic. Scripting languages based on simple batch programming are also quite diffuse while they present some limitations in terms of tools for code development and debug. Because scripting language has to integrate media computing functionalities by calling external processes and/or integrating plug ins.
- be capable to produce processes for the grid without re-compiling code to obtain executable tools with injected facilities for grid control and monitoring, see for example: Condor [Condor] or Globus grid [Globus]. The injection of grid control code into user processes may enforce a deeper control into the single processing nodes, while it is also a source of huge costs since each new functionality has to be accessible at code level and has to be recompiled with some grid library. It is not always an easy going process.
- be capable to execute new and non-planned procedures on the computational resources of the grid: without stopping and putting in maintenance the infrastructure. For example, by sending on the grid nodes procedures for installing new versions of software modules and plugins.
- support changes in processes at run time (easier by using scripting then native code), such as
    - adding/removing computational resources, storages, etc.;
    - changing the process code from one execution time to the next in periodic processes;
    - changing binary library and tools.
- be supported by an integrated development environment, with debug support and visual language for process and procedure modeling.

The **Request Manager** is the front-end of the micro grid and has the duty to control the authorization to the service usage and thus to accept them. The requests may be filtered by a business process to sign a SLA with the grid manager and organization. At technical level the Request Manager receives and collects the requests (in a Request Buffer) in order to activate processing rules on the micro grid according to their descriptors. They may arrive from several sources such as: front-end service portals, higher-level grids, workflow management systems, programmers on the integrated development environments, from the same micro-grid, etc. As to small/medium services the micro grid is totally at the service of the requesters sending the invocations according to process models with their formal description including: priority, period/sporadic, deadline, needed resources, functional part, etc. [Bellini2006]. On other hand, even if the micro grid is typically slaved to a factory or to larger grid or portals, the micro grid processes may also need the capability of activating rules/processes on the micro grid sending some invocation to the Grid Scheduler; this feature can be very important to implement recovery from failure procedures or for certain parallel algorithms in which the number of processes is not deterministic. In some cases, the requests to the grid may consist in: providing a new version of a processing rule/process, sending updated versions of plug ins/tools, making query on the **Rule Database** (containing the pool of processes that may be activated), posting a rule on the rule database, verification of the available resources and node descriptors, etc.

The **Grid Scheduler** discovers resources on the nodes, collects the answers and according to the selected optimization algorithm (e.g., taboo search, genetic algorithm, distributed deadline monotonic,..) finds the best matching to allocate processing rules on Grid Nodes. The Scheduler has the duty to control the Grid Nodes, to collect errors and set up fault tolerant invocations by duplicating calls with multiple or different parameters, to set up strategies of recovering from failure (e.g., reactivation of a failed rule), and to reconnect nodes that for some reasons lose the connection, to collect updates on their profile (that may change since resources are dynamically allocated, for example the HD space, the free memory, etc.). The Scheduler may be the single point of failure of the grid and thus failover/hot-spare solutions to set up a chain of multiple Schedulers is needed. In low cost solutions, the Scheduler can be allocated on the same virtual machine of a node and when needed the grid node may leave the node role to assume that of Scheduler in the case of failure of the main Scheduler. In those solutions, the presence of common NAS/SAN may help on setting up a repository for common status information of the grid. The allocation of multiple Schedulers and nodes can be a solution for a fault tolerant media micro grid infrastructure which can be easily allocated on a cloud computing infrastructure.

The **Grid Node** is the main controlled resource elements of the grid, and may be allocated (put in execution) as single or multiple instances on general purpose or specifically dedicated machines (virtual or physical computer). The grid nodes are the point of access to computational, communication, and/or storages resources, other resources can be SAN/NAS, databases, CDN, and networks. The node is the minimal grid service element controlled by the Scheduler, which can kill, pause, and resume the allocated processes on the node. The Grid Node may get access to local and remote resources according to its allocated processing rule/procedure, which defines its activity according to the general schedule of the Grid Scheduler. The Grid Node, at the level of the execution process, has to be capable to manage errors and faults in order to recover and/or report back to the Scheduler the general control. The Grid Node has to respond to the Scheduler discovering processes with general status information regarding its profiles and status. In the case of the AXMEDIS solution, the Grid Node processing logic is coded via a Media Computing Language which extended the classical JavaScript. Thus, a large set of functionalities is made accessible via core integrated modules and/or dynamic plug-ins, which are dynamic libraries with associated XML manifests for the dynamic loading of functional parameters and support help in the integrated development environment and debugger. According to the architecture reported in Figure 2, the rule/processes are executed on the grid node by the Media Grid Process Rule Engine which provides support for the exploitation of the local machine resources usage (e.g., CPU clock) according to the planned day by day profile. This capability is enforced via the Local Resource Controller, which allows employing office desktop computers as grid nodes. The Local Resource Controller allows keeping the control of CPU exploitation even when the execution procedure is demanded to some plug-in code.

Requirements for micro grid Media Computing Language

The AXMEDIS micro grid solution provides an open infrastructure compliant with the architecture reported in Figure 2. In AXMEDIS, additional features and/or installations can be easily deployed at run time via plug-ins and tools. Moreover, in order to perform integrated and optimized media-based services, a large number of functionalities are made accessible in each single node at level of Media Computing Language functionalities. Typical functionalities can be grouped into a set of categories, which are reported in the following to give the evidence of what is needed into a media grid:

- **content and metadata management.** It implies the access and retrieval from any kind of databases and storage area networks, as well as the access and processing of XML files and schemas. Moreover, metadata access implies also reach them via communication channels and harvesting via archive protocols such as OAI PMH, Z39.50;
- **content processing** for text, audio, video, image, multimedia, etc., via specific procedures and tools, such as adaptation and transcoding (e.g., from WMV to MPEG, resize, rescale, resample, ..); file processing and interpretation (XML, HTML, RDF, RDFS, etc.); formatting/layouting of presentation layer models (e.g., HTML/CSS, SMIL, MPEG-4); cross media format conversions (among them: MXF, MPEG-21, NewsML, SMIL, HTML/CSS, ePub, XML,..); fingerprinting and watermarking estimation and recognition/extraction. Among the content processing capabilities play a relevant role: the possibilities of processing XML via DOM, the application of regular expressions, the modeling of complex data structures via associative arrays, etc. More complex algorithms may be realized as native code and called via plug-ins;
- **semantic computing** is grounded on the automated production and exploitation of semantic information. Semantic descriptors may be directly accessible as collected information and/or provided by the user data: user profiles and preferences, network capabilities, device capabilities, context descriptor, use data, advertising, resources, .. Semantic extraction implies the application of algorithms for producing descriptors -- e.g., video and image analysis, audio analysis, doc analysis, for multilingual text processing for indexing (taking into account semantics, ontologies, dictionaries and metadata) obtaining integrated indexes where one may perform general and semantic queries in any language and form. Semantic processing can be used to exploit and perform reasoning on descriptors and logic to: take decisions on content layout formatting, provide recommendations, perform media file transcoding (see for example standards such as MEPG-DIA, CCPP), offering/proposing advertising, provide support to intelligent fuzzy search, etc. On the grounds of semantic computing, a number of low-level algorithms and tools are needed. For example, to estimate similarity distances among heterogeneous symbolic descriptors, so as to use them in direct matching and in clustering (e.g., K-Means, K-Medoids, hierarchical clustering); and to take decisions such as logic and inferential engines based on rules, Horn clauses, and First Order Logic, FOL;
- **communication** via several different channels (FTP/SFTP, HTTP/HTTPS, WSDL, ODBC, WebDav, etc.) and for content and information movement and distribution on operating systems, databases and SAN/NAS. The communication may lead to perform the integration with social networks, by migrating content, users and user profiles, automated publication for dissemination and promotion; and also by accessing the content of those social networks for repurposing and/or serving the portal users propagating queries in those large social networks;
- **content protection:** functionalities to protect content and licensing users which are needed to enable e-commerce and IPR management. Content protection may be implemented by content packaging and encryption (e.g., according to some standards such as MPEG-21, DVB, OMA). In order to set up an

effective e-commerce solution as DRM (Digital Rights Management) and/or CAS (Conditional Access Systems) many other aspects have to be managed. For example, the user registration with certification production (e.g., via a CA, Certification Authority), the unique ID assignment, the verification and validation of devices and players, the digital signature of content, etc. Once the protected content is produced on the micro grid, the content can be delivered to final users, who can be authorized by means of a licensing mechanism. In some cases, the licensing may involve massive production of codes and licenses for each content and user. As a limit case, millions of users with millions of licenses have to be produced and made accessible in quasi real time towards licensing servers and/or muxed into broadcasting channels. Conditional Access Solutions, such as those adopted by DVB, are typically more manageable to provide access to content via broadcasting. Micro grid may also be used to manage the customer relationships in the e-commerce applications and portals for billing, advertising, reporting, profiling services, etc. Among these services, the user profiling for supporting the on-demand business with personalized advertising is computationally intensive.

- **managing protected content.** The AXCP grid node may be used to manage content that has been already protected, for example when it is coming from content providers and it has to be repurposed and/or redistributed (for example in MPEG-21). These are the so-called content processing activities on protected content. Once the protected content has to be processed, the node itself has to be authorized/licensed and the technical platform certified to guarantee that the rights are correctly enforced according to their semantics, and thus the IPR respected. For example, a video adaptation can be performed, only if the grid itself is authorized via a corresponding license, the same for other rights such as those for extraction, packaging, synchronization, etc. For each of the supported rights on the grid, a formal validation is needed to guarantee that the grid node with its processing language does not allows via its programmers to violate any right in content access and manipulation;
- **content distribution**. Media and content distribution is realized by preparing the selected content and data from the back office to the distribution servers and channels. Presently, the distribution is typically multichannel and thus a media factory may have to provide content towards a plethora of front-ends: playout streamers, web servers, mobile servers, P2P networks, social networks, satellite carousel, DVB Mux, SMS/MMS services, etc. For each of them, specific formats and information packages have to be produced and updated. Moreover, the media grid may also be needed in the management of the distribution channel controlling and programming distribution servers to optimize: P2P networks, CDN, playout streamers, customer relationship management systems, etc. and for managing Quality of services, QoS. QoS control server may be focused on monitoring the content distribution on the clients, while collecting their feedback and tuning the services on the network. To this end, specific actions are performed, such as activating repeaters, proxies, changing the resolution according to streaming protocols.

The micro grid usage scenarios

In this section, details regarding some scenarios implemented by using AXMEDIS media micro grid solution are reported. The AXMEDIS micro grid solution is distributed free of charge, and it has been used in several real applications and trials as listed on: http://www.axmedis.org. The micro grid engine is called AXCP (AXMEDIS Content Processing) [Bellini2006]. Among the trials: back office management for IPTV; multichannel management for mobiles with OMA and MPEG-21 DRM; automated production and posting of content for satellite carousel; automated production and distribution of content towards local area kiosks distributed on several locations. Some examples of micro grid usage as back office and/or CDN engines are reported in Figure 3 and described hereafter with some details.

Digichannel audio visual content portal (http://digichannel.net/) distributes audio, video, photo, books and much more in high quality level by distributing original sources. Large size content (and collections) are packed into MPEG-21 intelligent objects which are protected by AXMEDIS DRM and delivered via Bittorrent based P2P networks such as Pirate-bay, Mininova, etc. The MPEG-21 content package may contain multiple and different items such as a CD ISO image plus cover and other info; or the DVD ISO image plus a booklet and images. In this case, the AXCP micro grid supports both the content production and distribution. On the context of production, the grid is used to automatically ingests, packs, protects and publish the content, according to user and content descriptors. Once published on the portal and on one or more P2P network guaranteeing a fast seeding. The effective distribution and diffusion of the package on the P2P can be periodically monitored by using AXMEDIS AXEPTool P2P control nodes to control the P2P CDN (bittorrent). The control nodes can be activated on specific torrent files to measure the seeding capabilities and thus the distribution level on the CDN. Once the user has downloaded the content, the user may put it in execution the package by using the Digichannel P2P/player tool that brings the user to e-commerce portal to buy a license according to DRM and its profile. The latter is used by the grid to produce the license. The DRM tool includes support for the verification and validation of client tools, detecting any changes in the client device/descriptor (player+HW), and using blacklists in the case of infringement detection. The acquisition of the license allows the user to unprotect and unpack the downloaded object obtaining the content in the original media format. The described business model has been agreed with the content providers distributed by Digichannel. In this case, a reduced version of the AXCP Grid Node (including the Media Grid Processing Rule Engine and a collection of its main functionalities) has been also enforced into cross media content players (i.e., Digichannel P2P/player) to execute the MPEG-21/AXMEDIS content. The player presents an extended version of MPEG-21 DIP and DIM capabilities, thus allowing a certain degree of intelligence into the cross media content player; it also includes DRM/CAS supports. Therefore, in this case the semantic computing is enforced into the content production and distribution, but also in the P2P CDN control and user licensing.

The activity of the cross media player can be performed in collaboration of progressive and P2P CDN and servers such as in the above described solutions, and in a BBC show case called AX4HOME [AX4HOME]. In the **AX4HOME** case, the player is activated from clicking on an AnyTime EPG (Electronic Program Guide) to make the recording of a free on air transmission in DVB-T, at the same time it downloads some additional content from the P2P CDN to create an intelligent content in MPEG-21/AXMEDIS format with interactive and proactive parts and may be advertising. The produced content provides nice shape and presentation interface (similarly to DVD), for example, providing links to other related episodes and content, games, biographies, back stages. Moreover, the produced interactive content can be generated in way to be only used into the domain/home of the user who created it [AX4HOME].

Mobile Medicine social network [MobMed] uses standard multimedia players fitting each and every multimedia file format, whenever the browser with additional players (plugins and/or ActiveX) allows it, which means hundreds of different file formats. Digital content can be audio files, video, images, documents, etc., or they can be complex, which is to say cross media, collections of videos, sequences of images, animations, SMIL interactive presentations, HTML, a range of different formats and types. For example, a large range of video formats are adapted to Flash for PC, and to MPEG-4 for Windows Media and iPhone. Complex content such as medical tools for dosages and procedures, educational content, sliding, etc., are produced in HTML or MPEG-21 for PC, and XML for iPhone. In this case, micro grid allowed automating a range of semantic computing activities such as: (i) ingestion and management of user generated content, UGC, with the related repurposing, adaptation for PC, Windows Mobile, iPhone, iPad;

(ii) estimation of content and user recommendations, both based on estimating similarities among descriptors and both based on static and dynamic descriptors that have to be continuously updated with the new use data, with the new users and with the new content; (iii) automated translation of textual descriptors and metadata, and corresponding indexing for fuzzy and semantic queries; (iv) managing metadata enrichment/update and content versioning that may include reformat and re-adaptation. On the grid, other smaller and more sporadic activities are managed such as the publication on other social networks and portals, the verification of consistency on the whole database and front-end portal, etc.

The Mobile Medicine solution also integrates a Content Organizer for mobiles where a reduced version of the Grid Node allows providing local recommendations to users via local semantic computing and using data collection. The client, called AxObjectFinder, is a content organizer for Windows Mobile devices and on iPhone (see Mobile Medicine application on Apple Store). It is an assistant for medical personnel and is strictly connected to the grid to get descriptors and content. The content and semantic information associated is transmitted from server to client in terms of enriched MPEG-21 packages and/or XML files in the case of iPhone. The local content organizer is capable to collect local use data which are used to provide local recommendations to the users.

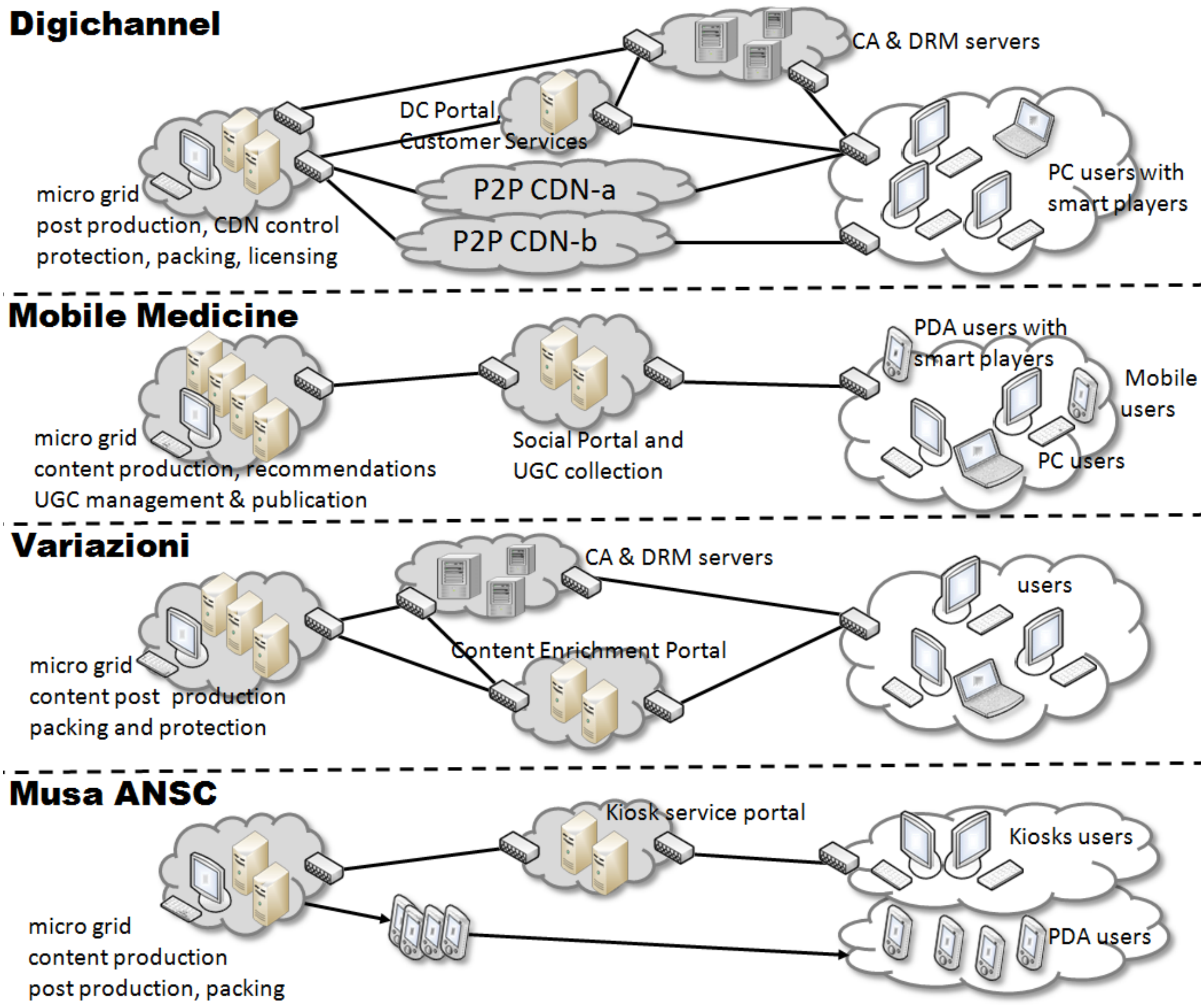

Figure 3 – Example of micro grid usage patterns.

Variazioni Content Enrichment Portal [Variazioni] performs enrichment on music educational content. Among the distributed content, some of them need to be adapted and protected according to some specific licensing conditions. Thus, a DRM solution has been tuned according to IPR of the institutions providing those content items. In this case, a media micro grid has been used as an external remote service for the Variazioni service front-end portal. The Variazioni portal invokes the AXCP microgrid every time a certain content has to be updated and re-published. The microgrid automatically perform some semantic computing activities: it gets the new content via SFTP, performs a suitable adaptation taking into account descriptors, packages and protects the content according to AXMEDIS MPEG-21, and then issues a set of licenses according to the user profiles. The micro grid is also used to produce client certificates (at the first user registration) and licenses on demand Variazioni when users accessing the protected content according to a pay per play model. The licenses may provide different kind of rights to play, enrich, print, and according to several business models. The protected content (audio, video and documents) in MPEG-21 encrypted format may be delivered via progressive download (if audio visual) or via P2P bittorrent. Users may have a limited number of terminals registered. Each action is tracked and reported to the Variazioni portal manager for reporting, accounting and statistical analysis.

Musical Instruments Museum of the Accademia Nazionale di Santa Cecilia (MUSA) adopted the micro grid for the automated production and update both Windows Mobile PDA and kiosks interactive content depicting one of the foremost collections of instruments in Italy [MUSA]. The AXCP micro grid automatically accesses to the MUSA archive and composes the audiovisual guides according to the current arrangement of the gallery and purposes, and to the device descriptors. Once the content is produced is automatically posted on the related computer for distribution and usage. The most valuable pieces are in the exhibition gallery, which can be visited with the support of PDAs. The advantage resides in the velocity in which the services can be updated with new content for the users.

Further Research and Conclusions

The media languages and micro grid technologies are suitable tools in the evolution of small and medium size services that are rapidly moving towards semantic computing. The usage of distributed systems and their pervasion are blurring the differences from what can be provided by servers and what can be privately computed on clients. In the above-described scenarios, the semantic computing enforced in the AXCP flexible micro grid via its media language for formalizing the processes allowed to automatically produce, post-produce, licensing content, delivering content by exploiting large range of features. Therefore, the proposed solution allow to set up flexible and scalable solutions to cope with a range of different applications and back office scenarios as reported in the article. Moreover, most of the solutions proposed enabled business models and are today accessible for small and medium enterprises at low or no cost.

In some cases, the semantic computing primitives have been also enforced into intelligent tools and players (see the cases of player for depacking content, and the content organizer and player). Thus, bringing into the hands of the final users more features and capabilities, such as local semantic search, recommendations, and therefore, a real personal assistant. In most cases, those clients are not only passive tools and they are playing a role in the general grid architecture. The described scenarios are representative of many other cases that may be realized by using micro grid models and tools. Our experience has been gained in several years of work in the sector and recently in a large integrated research and development project, AXMEDIS, of the European Commission with more than 40 partners, which has created a framework currently distributed free of charge.

Acknowledgements

The authors would like to express their thanks to all the AXMEDIS partners and affiliated members for their contribution and collaboration (namely: BBC, TISCALI, EUTELSAT, TIM, TEO, ELION, GIUNTI, HP, SIAE, SDAE, University of Leeds, FHG, University of Reading, UPC, EPFL, etc.), and to the European Commission. A warm thank-you to all the AXMEDIS people who have worked and sorry if they have not been mentioned by name. A specific acknowledgment to institutions and people behind the mentioned running services: Digichannel, ANSC MUSA, Mobile Medicine and Variazioni for their collaboration and stimulus.

Bibliography

[AX4HOME] AX4HOME DVB-T Recorder and Broadcast Enhancer, http://www.axmedis.org/documenti/view_documenti.php?doc_id=4005, http://www.axmedis.org/com/index.php?option=com_content&task=view&id=113&Itemid=80

[AXMEDIS] http://www.axmedis.org , AXMEDIS portal web site.

[Basu2003] S. Basu et al., ''mmGrid: Distributed Resource Management Infrastructure for Multimedia Applications,'' Proc. 17th Int'l Parallel and Distributed Processing Symp., IEEE CS Press, 2003, p.88.

[Bellini2006] P. Bellini, I. Bruno, P. Nesi, ``A GRID based Framework and tools for automating production of cross media content'', Proc. of the 11th IEEE International Conference on Engineering of Complex Computer Systems (IEEE ICECCS2006), California, USA, 14-18 August 2006.

[Bellini2007] P. Bellini, I. Bruno, D. Cenni, P. Nesi, D. Rogai, "P2P Architecture for Automated B2B Cross Media Content Distribution", Automated Production of Cross Media Content for Multi-Channel Distribution, 2007. AXMEDIS '07. Third International Conference on, AXMEDIS 2007, IEEE press, 28-30 Nov. 2007, pp.105-112.

[Bellini2009] P. Bellini, I. Bruno, P. Nesi, "Visual Programming of Content Processing Grid", The 15th International Conference on Distributed Multimedia Systems, DMS2009, Hotel Sofitel, Redwood City, San Francisco Bay, USA, 10-12 September, 2009.

[Condor] Condor home page: http://www.cs.wisc.edu/condor/

[Foster2000] I. Foster, A. Roy, and V. Sander, ''A Quality of Service Architecture that Combines Resource Reservation and Application Adaptation,'' Proc. IEEE/IFIP Int'l Workshop Quality of Service (IWQOS), IEEE CS Press, 2000, pp. 181-188.

[Foster2002] I. Foster, C. Kesselman, J. M. Nick, S. Tuecke, "Grid Services for Distributed System Integration", IEEE Computer, June 2002.

[Globus] Globus Toolkit: http://www.globus.org

[Harmer2005] T.J. Harmer, J. McCabe, P. Donachy, R.H. Perrott, C. Chambers, S. Craig, R. Lewis, B. Mallon, L. Sluman, ''Gridcast: A Grid and Web Service Broadcast Infrastructure,'' Proc. IEEE Int'l Conf. Services Computing, vol. 1, pp.35-42, IEEE CS Press, 2005.

[Huang2007] Y. Huang, Y.-F. Chen, R. Jana, H. Jiang, M. Rabinovich, A. Reibman, B. Wei, and Z. Xiao, “Capacity Analysis of MediaGrid: A P2P IPTV Platform for Fiber to the Node (FTTN) Networks”, IEEE JOURNAL ON SELECTED AREAS IN COMMUNICATIONS, vol.25, n.1, Jan 2007.

[InsightReport] The Insight Research Corporation Report “Content Management for Wireless Networks, 2008-2013”, August 2008, New Jersey, USA, http://www.insight-corp.com/reports/content08.asp

[Ishaq2009] A. Moiz Ishaq, S. Iftikhar, H. Farooq Ahmed, “Semantic Grid Service Registry”, 2nd International Conference on Computer, Control and Communication, pp.1-6, 17-18 Feb. 2009.

[Leff2003] A. Leff, J.T. Rayfield, and D.M. Dias, ‘‘Service-Level Agreement and Commercial Grids,’’ IEEE Internet Computing, vol.7, no.4, pp.44-50, 2003.

[Li2009] X. Li, B. Veeravalli, H. Li, “Multimedia Service Provisioning and Personalization on Grid-Based Infrastructures: Now and the Future”, IEEE Multimedia, pp.36-45, Vol.16, n.1, January-March 2009.

[MOBMED] Mobile Medicine, http://mobmed.axmedis.org

[MUSA] Musical Instruments Museum of the Accademia Nazionale di Santa Cecilia (MUSA), http://museo.santacecilia.it/museo/cms.view?munu_str=0_3_1_1&numDoc=148&l=EN .

[MXF] AAF/MXF: Multimedia Exchange Format, http://www.ist-nuggets.tv/mxf

[Omneon] Omneon MediaGrid http://www.omneon.com/mediagrid

[Rosenblatt2003] B. Rosenblatt, B. Trippe, S. Mooney, "Digital Rights Management - Business and Technology", M&T Books, 2003.

[Roure2005] D. De Roure, N.R. Jennings, N.R. Shadbolt, “The Semantic Grid: Past, Present, and Future”, PROCEEDINGS OF THE IEEE, vol.93, n.3, pp.669-681, March 2005.

[Variazioni] Variazioni Content Enrichment Portal, http://cep.variazioniproject.org/home

[Vkali2003] A. Vakali and G. Pallis, ‘‘Content Delivery Networks: Status and Trends,’’ IEEE Internet Computing, vol.7, n.6, pp.68-74, 2003.

[Volckaert2008] B. Volckaert, T. Wauters, M. De Leenheer, P. Thysebaert, F. De Turck, B. Dhoedt, P. Demeester, “Gridification of collaborative audiovisual organizations through the MediaGrid framework”, Future Generation Computer Systems, vol. 24, pp.371–389, Elsevier, 2008.

[Walsh2005] A. E. Walsh, ‘‘The Media Grid: A Public Utility for Digital Media,’’ Dr. Dobb’s J., Distributed Computing, pp.16-23. http://mediagrid.org/ , Nov. 2005.

[Xia2005] Q. Xia, R. Yang, W. Wang, “Fully Decentralized DHT based Approach to Grid Service Discovery using Overlay Networks”, Proceedings of the 2005 The Fifth International Conference on Computer and Information Technology, CIT’05, IEEE Press, 2005.

Biographies

Pierfrancesco Bellini *is a contract Professor at the University of Florence, Department of Systems and Informatics. His research interests include object-oriented technology, real-time systems, formal languages, computer music. Bellini received a PhD in electronic and informatics engineering from the University of Florence, and has worked on projects funded by the European Commission such as: AXMEDIS, MOODS, WEDELMUSIC, IMUTUS, MUSICNETWORK, VARIAZIONI and many others. He has been co-editor of MPEG SMR.*

Ivan Bruno *is a contract Professor at the University of Florence. His research interests include grid computing, image processing, semantic computing, and audio processing. Bruno received a PhD in electronic and informatics engineering from the University of Florence, and has worked on projects funded by the European Commission such as: MOODS, WEDELMUSIC, IMUTUS, MUSICNETWORK, AXMEDIS and many others.*

Daniele Cenni *is with Department of Systems and Informatics of the University of Florence. His research interests include social networks, semantic database, semantic computing, and clustering. Cenni received a degree in informatics engineering from the University of Florence, and has worked on projects funded by the European Commission such as: AXMEDIS and VARIAZIONI.*

Paolo Nesi *is a full professor at the University of Florence, Department of Systems and Informatics, chief of the Distributed Systems and Internet Technology lab and research group, and vice-director of the department. His research interests include massive parallel and distributed systems, physical models semantic computing, object-oriented, real-time systems, formal languages, and computer music. He has been the general Chair of IEEE ICSM, IEEE ICECCS, WEDELMUSIC, AXMEDIS international conferences and program chair of several others. He is at present co-program chair with Kia Ng (University of Leeds) of the international conference DMS 2010 of Software knowledge Institution. He has been the coordinator of several R&D multipartner international R&D projects of the European Commission such as AXMEDIS, MOODS, WEDELMUSIC, MUSICNETWORK and he has been involved in many other projects. He has been co-editor of MPEG SMR. Contact Paolo Nesi at nesi@dsi.unifi.it*