Mobile Medicine: semantic computing management for health care applications on desktop and mobile devices

Pierfrancesco Bellini · Ivan Bruno · Daniele Cenni · Alice Fuzier · Paolo Nesi · Michela Paolucci

Abstract In many health care situations, powerful mobile tools may help to make decisions and provide support for continuous education and training. They can be useful in emergency conditions and for the supervised application of protocols and procedures. To this end, content models and formats with semantic and intelligence have more flexibility to provide medical personnel (both in off-line and on-line conditions) with more powerful tools than those currently on the market. In this paper, we are presenting Mobile Medicine solution, which exploits a collection of semantic computing technologies together with intelligent content model and tools to provide innovative services for medical personnel. Most of the activities of semantic computing are performed on the back office on a cloud computing architecture for: clustering, recommendations, intelligent content production and adaptation. The mobile devices have been endowed with a content organizer to collect local data, provide local suggestions, while supporting taxonomical searches and local queries on PDA and iPhone. The proposed solution is under usage at the main hospital in Florence. The smart content has been produced by medical personnel, with the adoption of the new ADF-Design authoring tool, which produces content in MPEG-21 format. The mobile content distribution service is integrated with a collaborative networking portal, for

P. Bellini · I. Bruno · D. Cenni · A. Fuzier · P. Nesi (✉) · M. Paolucci
Distributed Systems and Internet Technology Lab, Department of Systems and Informatics,
University of Florence, Florence, Italy
e-mail: nesi@dsi.unifi.it
URL: http://www.disit.dsi.unifi.it
URL: http://mobmed.axmedis.org
URL: http://xmf.axmedis.org

P. Bellini
e-mail: pbellini@dsi.unifi.it

I. Bruno
e-mail: ivanb@dsi.unifi.it

D. Cenni
e-mail: cenni@dsi.unifi.it

M. Paolucci
e-mail: paolucci@dsi.unifi.it

🖄 Springer

discussion on procedures and content, thus suggestions are provided on both PC and Mobiles (PDA and iPhone).

Keywords Mobile Medicine · Content distribution · Cross media content · Automated production · mpeg-21 · pda · iphone · Semantic computing · DRM

1 Introduction

Medical personnel need to access fresh information and knowledge in emergency conditions, and during activities of continuous medical education and training. This knowledge supports medical/paramedical personnel in the adoption of continuously evolving standards, intervention protocols, complex dosages for pharmaceutical prescriptions depending on the context and patient status, etc. In hospital, the needed information and knowledge regarding these aspects is continuously updated and have to be propagated in short or real-time; for example, via desktop terminals and in some cases reprinted on paper. In a paperless hospital and in emergency/critical-conditions, mobile devices are mandatory tools for information access and therefore necessary to take important decisions; on such grounds, the solution has to guarantee the access to right and updated information in the needed time [21, 26].

Semantics computing technologies may be profitably used to enforce more intelligence and efficiency in some of the above-mentioned services in the medical area, thus, integrating technologies of content distribution with semantic processing and making decision support capabilities. In many applications, the semantic computing (as a support to the decisions, via the exploitation and processing of descriptors and semantic information) is confined on the server side to provide recommendations and reasoning about semantics, content and use data, users profiles, etc. Among these applications, collaborative solutions, social networks and recommendation systems are the most widespread cases where server side semantic computing is applied in several different extents, to provide a set of features and services to the users. Most of the widespread Social Networks, are mainly focused on simple content formats (e.g., YouTube, Flickr, LastFm) [19], while others are focused on: establishing social relationships among users (e.g., Facebook, MySpace, Orkut, Friendster) [23], searching users sharing same interests or people already acquainted with in the real world, and modeling groups [18]. The former Social Networks organize and classify content on the basis of simple direct keywords, so that users access, retrieve and share them. Despite massive success of Social Networks in terms of penetration, most of them have limited semantic computing capabilities and provide simple recommendations about possible friends and for marginally similar content items. Most of them can manage only simple audiovisual content (e.g., images or video, which are the simplest content items to be generated by users). On the other hand, the technologies of semantic computing may make reasoning about many other aspects such as: content descriptors, user profiles, device capabilities, use data, contextual data, etc. In Medical 2.0 (http://medical20index.com/), some examples about the applications of web 2.0 technologies on medical environment are presented, and provide limited capabilities.

In the hospital and emergency medical situations, there are many additional constraints. In general, connections can be discontinuous (even in the event of multiple protocols: WiFi, UMTS, GPRS, ..); for example, when patients are moved, along corridors, in the country side, in tunnels, on the street, on the ambulance, in critical conditions, etc. In this scenario, the server side management of information and semantic computing services is not enough since

the information and knowledge have to be accessible as real-time off-line services on mobile devices to provide a useful service (they may be not accessible or too slowly accessible). The mobile device has to be able to guess the user's intentions and wishes, and to provide suggestions/help, on the basis of the context and user profile. The information has to be smartly recovered and processed to provide suitable suggestions to medical personnel in real-time. ***In the context of this paper, this is the so-called Mobile Medicine scenario.***

In the Mobile Medicine scenario, there is the need of performing semantics computing on mobile devices and in general on client side. Semantic computing capabilities on the mobile device can be useful to provide support for making faster decisions also in off-line conditions. For example, to take into account the context, to provide local and personal recommendations, to select and use dosage tools, to recover health care procedures, to perform classification and assessment models, to identify and follow intervention models, to get access to suitable educational content, to access and exploit interactive technical manuals, etc. Therefore, in order to integrate autonomous capabilities and semantic computing on mobile devices, intelligent content models and packaging solutions for delivering them are needed. They may have capabilities of enforcing/modeling multiple paths or experiences; exploiting complex semantics and descriptors, creating interactive intelligent content with semantic computing capability, and when possible taking into account server side fresh information.

On the side of intelligent content, there is a number of multimedia and cross media content formats for content packaging such as MPEG-21 [13, 14], MXF (http://www.mxf.info/), SCORM/IMS [22], MPEG-4 [25], and proprietary formats such as Macromedia, Adobe, etc., that put together a set of multimedia content and other pieces of information. Most of these formats have been invented to offer advanced experiences to final users in terms of media usage or interoperability, whereas they do not support the exploitation of complete semantic information and intelligence. Most of them only wrap different kinds of digital resources/files in a container/package (e.g., content metadata and descriptors, relationships among resources, etc.). Examples of intelligent content are: X-MEDIA (http://www.x-media-project.org), EMMO [28], AXMEDIS (http://www.axmedis.org) [7, 8, 10, 11], and KCO [3]. X-MEDIA model presents semantic aspects that can be managed by ontologies and RDF. X-Media is focused on knowledge management for text and images with objects having a very limited autonomy of work, objects that are not proactive with the users. EMMO encapsulates relationships among multimedia objects and it maps them into a navigable structure. An EMMO object contains media objects, semantic aspect, associations, conceptual graphs, and functional aspect. KCO uses semantic to describe the properties of KCOs, including raw content or media items, metadata and knowledge specific to the content object and knowledge about the topics of the content (its meaning). MPEG-21 is focused on the standardization of the content description related to Digital Rights Management, DRM, aspects [20, 24]. AXMEDIS extended MPEG-21 is conceived to propose content packaging, and integrate presentation aspects, semantic computing, intelligence and behavioral capabilities [9].

The comparison of the state of the art of Mobile Medicine system can be performed only at level of single technology since there are no other similar integrated solutions available. Therefore, we decided to design and develop a Mobile Medicine solution to cope with the above-mentioned problems, while exploiting the possibilities of semantic computing, in an integrated solution for the medical area providing support for:

- delivering and exploiting on mobile devices intelligent content able to help making decisions, estimating dosages, performing assessments, collecting data, etc.,

- providing on-line suggestions via semantic computing on servers of the web portal,
- automated/assisted content production, collaborative content finalization and semantic enrichment,
- providing off-line suggestions via local semantic computing on the mobile devices.

The experience reported in this paper refers to the design and develop of the so-called Mobile Medicine service and tools. It is presently under trial at the largest Florence Hospital joined with University of Florence health care area (i.e., Careggi Hospital) and publicly accessible at http://mobmed.axmedis.org. It is functional for desktop computers (i.e., Microsoft Windows, Linux, MAC OS) and for mobile devices such as iPhone/iPod/ iPad, and PDA as Windows Mobile based phones. The most innovative aspects are related to the automated production, semantic enrichment, distribution and exploitation of complex content and therefore the exploitation of semantic computing capabilities on PC and mobile devices, thanks to the distribution of packaged content. Among the semantic computing aspects: indexing and querying, automated adaptation, user and content recommendations.

The paper is organized as follows. Section 2 reports the requirement and the architecture of Mobile Medicine. In the discussion, the semantic aspects are stressed to highlight the flow of the information and tools needed to exploit such aspects, thus showing the enabling technologies needed, such as: the scalable back office for semantic computing; the content format extended from MPEG-21 to enforce, migrate and execute semantic descriptors; the content organizer to allow providing local suggestion and support for intelligent objects on mobile devices. It allows to provide the final medical users with a tool able to perform semantic search and receive suggestions on the basis of private local information. Section 3 refers to the semantic computing algorithms enforced into the Mobile Medicine Collaborative Network Portal (CNP) for suggestions, grounded on semantic computing as to proximity and similarities among users, objects, groups, etc. Similar metrics have been also used on the mobile to provide local suggestions (see Section 2.4). In Section 4 and 5, experimental results on clustering and related comments, and content access are provided. Conclusions are drawn in Section 6. In Appendix, some details regarding similarity metrics are reported.

2 Mobile Medicine requirements and architecture

According to above described scenario, the most important requirements for the integrated Mobile Medicine solution have been identified. The most relevant capabilities deals with providing support to medical personnel:

A. **during emergency and critical conditions (on-line and off-line)**, to get right content that may be useful to take decisions such as: estimating dosages, estimating objective assessment models, guiding on choosing medicine and solutions, reminding precise procedures/protocols, reminding the usages of accessible instruments (for example in an ambulance), etc. This requirement may be satisfied by using a set of technologies: server side recommendations, local recommendations, intelligent content model, suitable servers exploiting semantic computing on content, off-line availability of these features, etc.

B. **for continuous medical education/formation via mobile devices** (CME means life-long education of medical personnel). This implies to define courses, assign them to personnel, and monitor progresses, etc., while using mobile devices. The adoption of mobile devices for medical education is still not very widespread, yet. Moreover, in

this context we mean education focused on the exploitation of the intelligent content and tools.

C. **To produce and refine intelligent content with simple assistive tools**. This feature is meant to (a) collect a range of possible media types and automatically make them suitable for a range of devices, (b) provide production tools for producing intelligent content as to decision support (as described above) including the production of semantic descriptors and packaged content for distribution. When it comes to point (b), production tools have to be intuitive and usable for non-ICT specialists such as medical personnel. This implies that the medical personnel using such tools may complete and refine content autonomously.

In Mobile Medicine, the capabilities of mobile devices are very important since they have to provide services in off-line conditions and have to allow local search, to produce local recommendations, to perform intelligent updates, and to provide support for taking decisions and assessment on the basis of context. The content itself may perform reasoning on semantics, thus providing different behavior on the basis of content descriptors, user preferences, contextual information, etc. The off-line activity implies to have on the mobile devices a local software application which may provide these features autonomously. Moreover, the same services have to be provided on-line from the web portal towards classical connected devices. Recently, small commercial applications for medical area are available on the market for smart phones, such as Epocrates (http://www.epocrates.com/). On the other hand, they are not supported by semantic modeling that may allow users to search them, to search into their content, to maintain them updated on the mobile, to take into account server's fresh information, to get suggestions and recommendations on the basis of the context and preferences, in the cases of emergency, during home visits, and/or for educational purposes. For these reasons, mobile devices and applications of Mobile Medicine have to provide a set of challenging features that cannot be met without the injection of a certain intelligence into the content itself.

Figure 1 reports the general architecture of Mobile Medicine solution. It consists of a four-layers architecture including: (1) a back office for semantic computing; (2) a front-end portal and services (called **Mobile Medicine Collaborative Network Portal**, CNP) which performs a part of semantic computing activities and provides on-line services to the client side layers; (3) the final user tools (Content Organizer)) for content management on mobile devices and content production on PC; (4) the final user players and browsing tools located on PC, PDA, iPhone devices. In the case of PDA and iPhone mobile devices, they are typically activated by the Content Organizer, while on PC, they are activated by the browser.

In addition to the mentioned layers, there is an external service for intellectually property management, IPR, which is based on AXMEDIS[1] Digital Rights Management, DRM, Servers. This facility has been used to protect and distribute content, while enabling the exploitation of content rights to users only according to the produced licenses [20]. AXMEDIS framework is a set of free basic tools for DRM and scalable processing, integrating functionalities for massive content production and management, and for multichannel distribution [5]. This paper is presenting Mobile Medicine solution, and not

---

[1] AXMEDIS (automated production of cross media content for multichannel distribution) (http://www. axmedis.org) is a content media framework developed by an European Commission IST IP Research project [6, 7], with the support of more than 40 partners, among them: University of Florence, HP, EUTELSAT, BBC, TISCALI, TEO, ELION, Telecom Italia, RAI, SIAE, SDAE, FHGIGD, AFI, University Pompeo Fabra, University of Leeds, EPFL, University of Reading, etc..
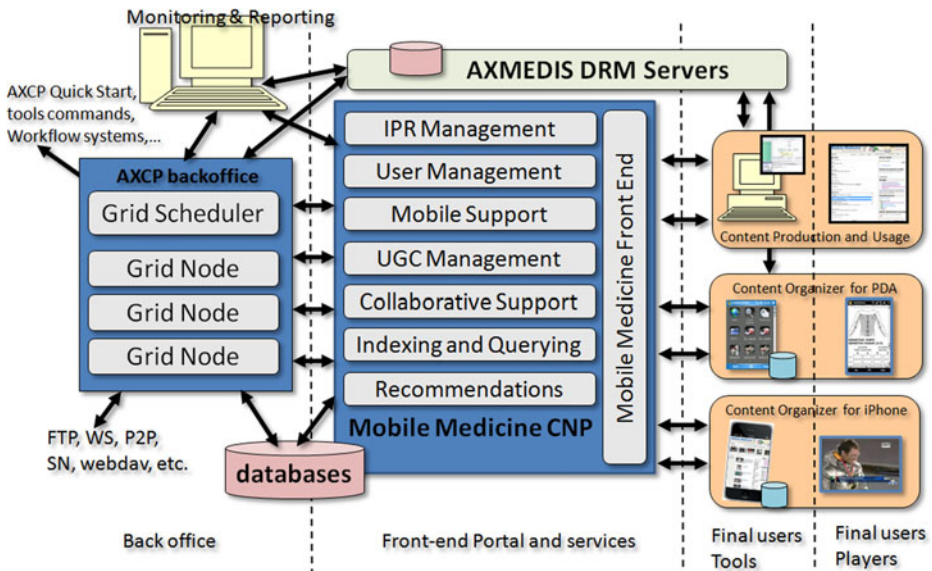
Fig. 1 Mobile Medicine architecture

AXMEDIS tools. Mobile Medicine is grounded on some few AXMEDIS tools and it is hosted in its domain and facilities.

The deployed architecture is scalable and consists of a number of servers allocated as virtual machines on the DISIT datacenter: a DRM server (with licensing and certification authority), a Mobile Medicine front-end server with Apache web server for PHP aspects and Tomcat web application server plus caching tools, a cluster of two database servers, a cluster for the AXCP back office that allocates on demand from 1 to 30 nodes of the media computing grid. Some of the mentioned main services of the front-portal are implemented as PHP modules of Drupal, while others are JSP Web Applications or stand alone applications, which may be put in execution on the back office grid nodes. This mix of technologies on the front-end is kept together by AJAX. The production side is into the hands of the many users of the large test bet in Careggi Hospital, which is the largest hospital in the Florence area. On the client sides, users get connected with their PC via browser obtaining the needed tools for playing and work. On PDA and iPhone, users have to install on their devices a Content Organizer tool for local semantic assistance as described in the following. Moreover, external or sporadic users may access on Mobile Medicine with their mobile without installing the Content Organizer. Some statistical data on portal activity have been reported at the end of the paper.

2.1 Mobile Medicine collaborative network portal and semantic data collection

According to Fig. 1, **Mobile Medicine CNP** manages the WEB portal/pages for multichannel distribution. A number of services are provided via web interface front-end of the CNP, while they are enabled by the back office. Therefore , the most important CNP services and modules are:

- **User Management** module to allow registration of new users, registration and authentication, user profile collection, use data collection about the activities carried

out by users. Each user may be registered to multiple working groups on the CNP, provide content, play content, provide comments, make queries, send messages and votes, and it may establish connection with colleagues, etc. All these activities are logged to collect use data needed for providing recommendations as described in the rest of the paper.

- **IPR Management**: a module for the intellectual property management of content. It allows the definition of rules to access and exploit content on the portal and on mobiles devices. The IPR rules are formalized in MPEG-REL (Rights Expression Language) and posted on the DRM licensing server.

- **Mobile Medicine Front-end**: is the web-based interface providing the collaborative environment user interface and the content access according to the several protocols: http, http download, progressive download for audio visual, and P2P towards PC and mobile devices. It recognizes PC and mobile devices distributing the right content format and web pages in real time.

- **Mobile Support**: a module to provide web interface to mobile devices such as: iPhone/ iPad/iPod, Windows Mobile devices and other smart phones. It provides support for mobile devices accessing the other modules of Mobile Medicine CNP such as: Collaborative Support, User Management, Grouping, content, etc.

- **UGC Management** (User Generated Content Management): a module to manage workflow activities of content upload, enrichment, review, acceptance, publication. Once the content is uploaded, it may be inspected or directly moved to the back office for the automated transcoding and formatting needed for the publication towards final users' devices. During the content upload, a set of metadata, the group assignment, and the medical taxonomical classifications are requested. The users' activities on UGC and enrichment are tracked to collect pieces of information to produce recommendations.

- **Collaborative Support**: this module manages CNP activities such as: web page production, discussion forums, content enrichment and discussions, messaging, news, multilingual translations, etc. Most activities can be performed on both PC and mobile devices. The module records these activities for the production of recommendations as described in the rest of the paper.

- **Querying and Indexing Data Collection**: this module is charged with collecting data for indexing the different content kinds managed on the portal: content object for pc and mobile, web pages, forum topics, comments on any kind of content, etc. Queries can be performed on the portal by using the above-mentioned modalities as simple text queries or as advanced queries specifying the modalities. A complete substring match and advanced detailed search are provided for specialists.

- **Recommendations**: the proposed solution collects a large number of pieces of information to produce suggestions / recommendations. Suggestions are used to simplify the content search and to push users to get access to new content, and to read news. Due to computational complexity, the estimation is demanded to the back office as described in Section 2.2, while the presentation of proposed lists or random selections from these lists is performed in real-time according to the user's accesses to the portal.

The Mobile Medicine CNP manages and collects information to perform semantic computing and reasoning on the backoffice and on the client side Content Organizer which indexes information and performs intelligent queries based on similarity distances, fuzzy search, and it provides computing suggestions/recommendations on the basis of static and

dynamically collected information. Among the main semantic pieces of information managed, there are:

- **user profile descriptions** collected via user registration and dynamically on the basis of user actions, migrated also on the mobile: selected content, performed queries, preferred content, suggested content, etc.;
- **relationships among users/colleagues** (connection among users similarly to friendships, group joining) that impact on the user profile and are created via registration, by inviting colleagues to join, by performing registration to groups, etc.;
- **user groups descriptors** and their related discussion forums, multilingual web pages, with the possibilities of providing comments and votes (with taxonomical descriptors and full text in multiple languages for web pages and groups);
- **content descriptors** for simple and complex multilingual content (intelligent mini-applications), web pages, forums, comments, technical descriptors, etc.;
- **device capabilities** for formal description of any acceptable content format and parameters, CPU capabilities, memory space, SSD space;
- **votes and comments on contents, forums, web pages**, etc., which are dynamic pieces of information related to users, content, forum, groups;
- **lists of elements marked as preferred by users,** which are dynamic pieces of information related to users;
- **detailed descriptors about downloads and play/executions** of simple and/or complex content on PC and mobiles, to record user actions as references to played content, which are dynamic pieces of information related to users' preferences such as: date and time, GPS position, object ID, etc.;
- **descriptors about content uploaded and published** by users (only for registered users, and supervised by the group administrator). Each Content element has its own static multilingual metadata, descriptors and taxonomy; while the related action of upload is a dynamic piece of information associated with the user who performs it. In addition, content elements can be associated with groups.

In Mobile Medicine (see Fig. 2), the semantic information is collected on server portal and on mobile devices for both on-line and off-line activities. The information mainly flows from server to client devices, while use data flows in the other direction. Therefore, in order to propagate the content descriptors towards mobile devices, they are packed together into the content package or provided with one or more XML description files (namely performed by the AXCP and the Content Organizer on smartphone, the ObjectFinder for both PDA and iPhone). The user benefits from semantic computing results from both server and mobile device sides. For example, fuzzy query results, suggestions, taxonomy based results at their query, participating to social activities of groups, etc. In addition, on the mobile side, a local semantic engine and a player are used to collect local use data, to exploit them locally in providing query results on the basis of taxonomy and suggestions, recommendations. In the respect of privacy, locally collected use data and profile may be sent back to the server only having the user authorization. In most cases, to provide local suggestions (i.e., those that are based on local private data, use data and profiles) is the only solution to help users finding content in a faster manner, especially when the mobile device is off-line.

2.2 Back office activities and tools

In the back office, the Mobile Medicine solution performs a set of activities, to produce and/or to exploit the above-mentioned semantic information. Some of these activities
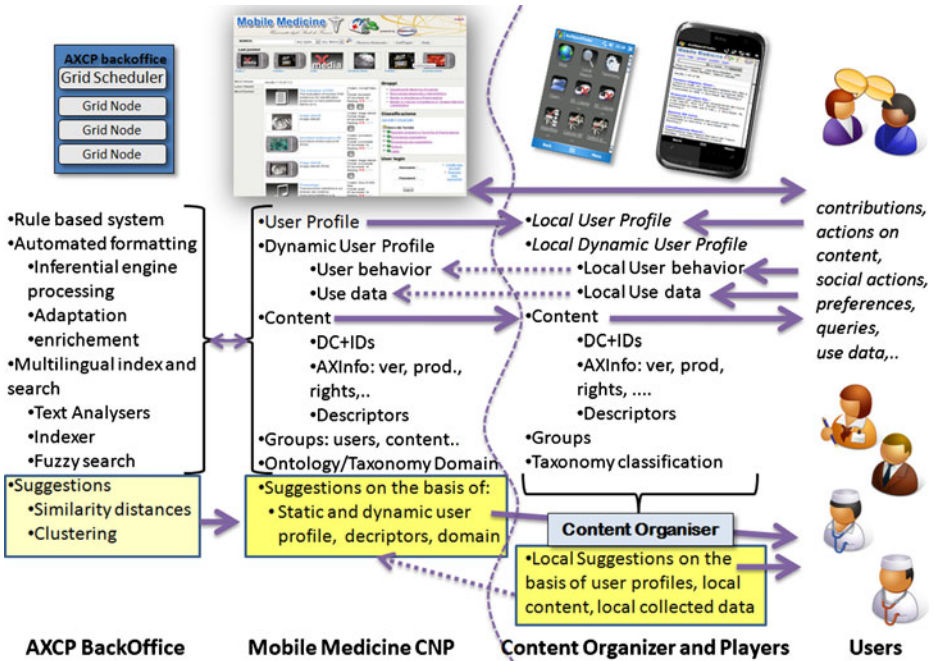
Fig. 2 Flow of semantic information on Mobile Medicine

produce data exploited by the front-end in real time (i.e., on-demand) on the basis of the users' requests. For example: queries on the multilingual indexes and semantic classifications and comments, the search of similar users on the basis of their profile, the identification and re-production of suitable content on the basis of device profile (web, mobile, PDA,..). On the other hand, many back office activities are computational intensive, and take hours and days to be completed and, thus, they have to be performed off-line, periodically, incrementally and sporadically in the presence of some specific conditions/events. In general, the computational complexity may be due to one or more of the following causes, for example: content size (e.g., adaptation of real video files), number of users to be processed to estimate similarities (for example for clustering and recommendations), number of content items to be ingested and/or transcoded. Among the several complex back office activities, those related to semantic computing are put in evidence in the following, while other activities are not detailed due to lack of space: content ingestion, multilingual indexing, content production, adaptation, packing and publication.

Content enrichment The several kinds of content in the portal provide a limited number of pieces of information for their indexing, so that they need to be enriched in order to provide enough information. Content items in the portal can be audio, video, applications, dosages, web pages, comments, forum topics, etc., and they can provide descriptors, multilingual information, taxonomical classification, association to groups, user ranking, comments, preferences, etc. In most cases, the content has no technical description and very limited multilingual data, etc. Some of this information has to be produced in multiple languages.

Therefore , the content enrichment may be strongly computationally intensive and may include:

- **extraction of technical descriptors** for any digital resources which are needed for content adaptation and repurposing. The information is different according to either simple or complex content resources. Basic information may be: duration, resolution, size, dimension, video rate, sample rate and size, file format, MIME type, number of included files, file extension, etc. Different libraries or tools have been used to extract this information: FFMPEG for video and audio, ImageMagik for images, etc.
- **extraction of semantic descriptors.** The content can be also processed to extract information from the content data, for example: by processing images to understand the represented content (what is included into the image); documents to get a summary and extract keywords; video files to segment major scenes and understand them; audio to extract tonality, rhythm, etc. These activities are strongly computational intensive. In this case, content semantic descriptors may be formalized in some XML, RDF and/or MPEG-7, and may be used to estimate similarity among content items.
- **multilingual translation** of textual content and/or descriptors. The content typically reaches the archive and portal in one or a few more languages. An automated draft translation can be useful to make the users work faster and simpler, while limited to validation.

Estimating suggestions implies to perform a number of semantic computing activities that range from: (a) estimating similarities among non-homogenous entities which are described by a mix of symbols and numbers, such as users' descriptors, user and content descriptors, content and content, taxonomy terms, annotations, etc.; (b) producing suggestions and recommendations by using techniques based on queries and/or clustering. The formal similarity among non-homogenous data can be used for direct estimation of distances and as a basis of clustering techniques as described in the following.

The above-described semantic computing activities are performed in the back office realized by using AXCP grid scalable architecture [6]. The AXCP grid can manage parallel executions of processes (called rules) allocated on one or more grid nodes. Rules are managed by a grid Scheduler and they are formalized in extended JavaScript. The Scheduler performs the rule firing, resource and node discovering, error report and management, fail over, etc. It puts rules in execution (with parameters) periodically or when other applications ask for. With the extensions developed in Mobile Medicine, a Rule may perform activities of semantic computing, extraction of descriptors, indexing, in addition to those of content and metadata ingestion, query and retrieval, storage, content formatting and adaptation, transcoding, synchronisation, estimation of fingerprint, watermarking, summarization, metadata manipulation and mapping, packaging, protection and licensing, publication and distribution.

2.3 Mobile Medicine final users tools

In the Mobile Medicine, final users are both producers and final users (i.e., prosumer) of the intelligent content for both PC and mobile devices. In this section, a short overview of the format for intelligent content is provided, while the other sections report the description of the semantic computing provided by the ObjectFinder tool hosted on the Smartphone (Windows Mobile PDA and iPhone); and the authoring tool for producing the content, while automatically producing the related semantic descriptors.

In the Mobile Medicine scenario, the content types range from *single files*: audio, video, images, documents, slides and animations; to *cross media* containing supportive tools such as a sort of mini-application (internally consisting of a set of files and descriptors). As depicted in Fig. 3, examples of cross media content can be:

- **interactive guidelines/procedures** to help users to remind of the correct procedures and help them in taking decisions on the specific cases and pathologies. They are internally built in a way similar to flow charts describing what has to be done from general to specific cases (for example in emergency conditions). The medical personnel is guided to answer to a set of questions leading the system to identify precisely the context and thus the actions to be performed on the patient and his general conditions. The so called triage model typically underlies the whole process with several other procedures connected to it to address specific cases;

- **calculators and tools** which are interactive applications where the user may insert data/ info collected out of the patient/context (such as: age, weight, pressure, temperature, habits, symptoms, conditions, reactions, etc.) to obtain from the device/content some estimations/suggestions to be used to take decisions. For example, the estimation about the probability of pulmonary emboli, the estimation of a dosage on the basis of patient weight, the estimation of fat percentage, the suggestion of a prescription about what you could eat, the assessment of neurological conditions on the basis of standard quantitative models, etc.

The above-described content model has to provide a certain degree of autonomy to react to the user's stimulus and to provide support to take decisions. The digital content items may be composed or linked to create more complex content solutions and paths for both fixed and mobile devices and accessible off-line. To this end, the cross media content has be

Fig. 3 Cross media intelligent content of Mobile Medicine: video, audio, sliding, assessment, data collection, dosages, decision support

indexed in local database to be searched and thus activated in short time at disposal of any medical personnel using the mobile device, with no need to wait for the web page, nor for the content download. The cross media content may take advantage of the communication with the server if any, and may provide hints and suggestions to the user in any case, even when the mobile device is off-line. For the above-mentioned complex content, an authoring tool is available to allow medical personnel to create their own procedures and tools, see Mobile Medicine Portal.

At level of content format, the content/applications are modelled as AXMEDIS packages in binary/xml file format [6, 7], enforcing in addition Mobile Medicine semantic descriptors and capabilities. The format structure supports the navigation, the creation of nesting levels and the direct access to resources via links and references. The interaction aspects in the model are delegated to the presentation layer. The binary version of the format includes a table for immediate access to digital essences, even when they are encrypted and it is based on ISO Media file format [9, 11]. This allows content to be downloaded, streamed and/or progressively downloaded and played in real time, even when essences are protected.

In the Mobile Medicine format **multiple descriptors**, identification and classification info are hosted: Dublin Core, extended metadata and descriptors, taxonomical association and RDFS formalizations, plus AXInfo (a set of metadata used to manage the content life-cycle), workflow info; plus any other identification and/or descriptors in XML, and/or RDF. Descriptors included in the content package may refer to internal and/or external digital essences. The format supports **multiple presentations** in the same package. For example, it may provide and alternate HTML/CSS/JS, XML, SMIL, ePub, NewsML, MPEG-4 and FLASH in dominating the main canvas of the player. HTML and other XML formats may use style sheets and digital essences (text, video, audio, image, etc.) hosted into the content package. On AXMEDIS players, presentation layers (such as SMIL, HTML, and FLASH) and also internal events may put in execution AXMethods written in extended JavaScript for activating **behavioural actions** which allow to perform a range of functionalities of the intelligent content including: inspect and modify the content structure (e.g., add new resources, internal search, reflection), control the resource rendering, perform calls to web services, take decisions (on the basis of user profile, context, actions performed, descriptors, rule based, etc.), activate semantic computing, communicate with servers, recover GPS position, recover data from web, etc.

On such grounds, the content behaviour is specified by coding the business logic intelligence with JavaScript. Semantic computing and processing capabilities are activated via JavaScript by using the same formalism of the AXCP back office language grid mentioned before. Each single content may implement different communication modalities with the portal server to get and post information. The content may take decisions locally or may ask for fresh server information according to the communication conditions, before deciding.

The production of Mobile Medicine content is performed by using a set of integrated tools (see Fig. 4):

- **automated production** exploiting AXCP extended rule functionalities. When the UGC is produced by the prosumers and uploaded on the portal, some metadata are requested, while other descriptors are automatically added when digital essences are adapted and the package produced for publication on CNP (as PC and mobiles, this is carried out according to the IPR). This modality is used for single files: video, audio, images, documents, etc., in any format. During the upload, the prosumer provides some metadata, grouping and taxonomical classification which are used by the portal and on
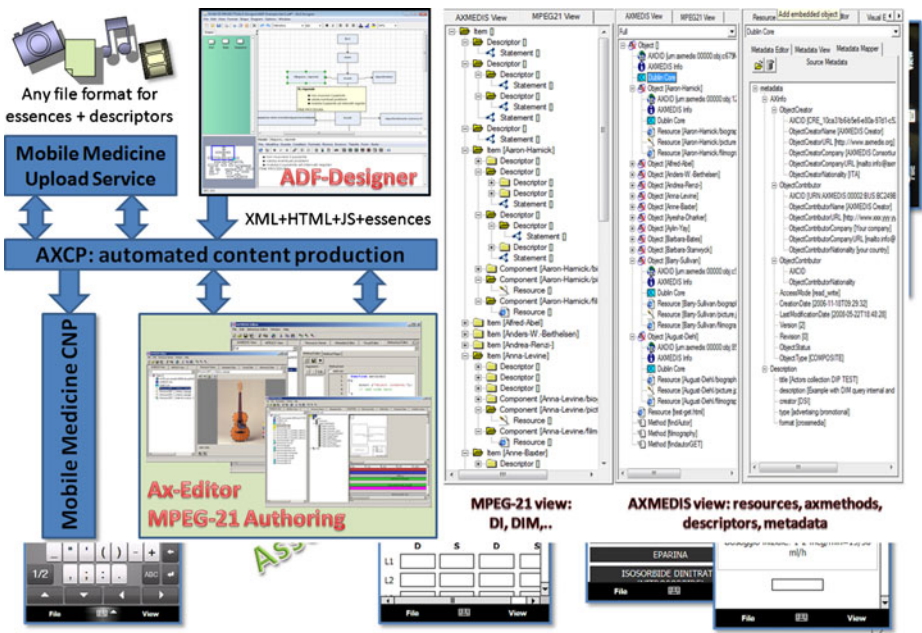
Fig. 4 Content production flow and format structure in MPEG-21 and package views

the smart phone for semantic computing. Technical and other semantic descriptors are automatically added by the AXCP and included into the content package.

- **assisted production** of complex intelligent content (emergency procedures, questionnaires, decision taking supports, etc.) by using a visual language as depicted in Fig. 4, which shows the ADF-Designer tool. The ADF visual formalism has been specifically designed to formalize Medical procedures and it is simpler than the flow chart (screen pages connected to possible other successive pages). Details about the single screen are provided with a HTML graphic editor tool. The interaction capabilities are confined in HTML controls. The ADF tool generates automatically the integration and business logic code in a set of files, which is used to produce automatically the Mobile Medicine content package in MPEG-21 and such package can be uploaded on the portal. During the content ingestion a specific descriptor to propagate the semantics into the iPhone Mobile Medicine application is produced.

- **manual authoring.** The AXMEDIS Editor is a graphic authoring tool for MPEG-21, including: structure editor, presentation tool editor (HTML and SMIL), metadata and descriptor editor, behavior editor and simulator, workflow editor, IPR licensing tools, and protection tool on packager.

The Mobile Medicine content package is the vehicle to move semantic descriptors from portal to mobile devices and it is the container where semantic computing capabilities find a context in which medical personnel can take decisions and interact with the general distribution portal. In fact, cross media content on mobile players executes content where the content itself changes behavior depending on user profile and actions, or on basis of local context (GPS, accelerometers, device status, communication status, ..) and server context (server files and info, server accessibility); it may communicate use data or other pieces of information regarding the mobile status to the server. The Mobile Medicine

content may be used to actively and proactively: (a) collect patient data, (b) remind medical personnel or patients of performing certain actions or of checklists to be followed, (c) collect annotations proactively, (d) create multimedia scrapbooks, (e) collect user generated content, etc.

2.4 Mobile device content organizer for local semantic computing

In the context of the Mobile Medicine, it is very important to provide a certain number of services in off-line conditions. The solution has to be user friendly, so that medical personnel do not need to take care about the content/application life cycle. To this end, a Content Organizer (technically called ObjectFinder) for smartphone has been designed for the Mobile Medicine (it is freely distributed for PDA Windows Mobile, and for iPhone) (see Fig. 5, for the architecture of the PDA version, a similar structure is also available in the iPhone version which can be downloaded from Apple Store free of charge).

The ObjectFinder has a set of essential semantic computing capabilities; it processes the content items to index them according to the descriptors, taxonomical classification of medicine, and it provides support for querying and organizing content according to the user data and requests. It has been designed to support the users to collect, index, organize, update, search and retrieve content items on the basis of their semantic descriptors, user profile (static and dynamic, including use data), user preferences, etc. It enforces semantic computing and intelligence at level of mobile device and it benefits from the management of intelligent content it can download and collect into the smart phone. The ObjectFinder provides a direct usable interface based on icons for main functionalities: portal access, local search, taxonomy based content browsing, suggestions, access files, etc.; and for content play with a single finger click. The ObjectFinder **Presentation Engine** is based on HTML with a protocol to access objects stored into the mobile devices (e.g., those with icons and query results).

The Content Organizer **ObjectFinder** can

- **collect and index cross media content** coming from several channels: from the distribution portal, from the connected computers (e.g., USB, Bluetooth, IRDA), network connection (e.g., HTTP from a web page download). The indexing is
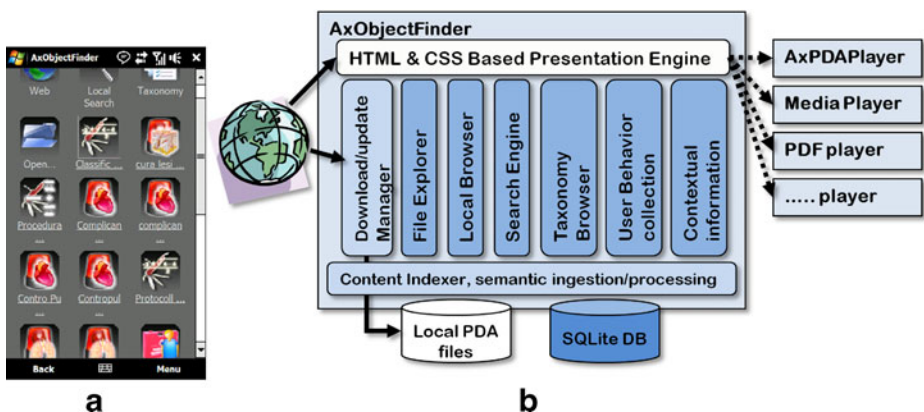

Fig. 5 ObjectFinder for PDA: **a** user interface, **b** architecture

performed by exploiting MPEG-21 package metadata and semantic information into the local database of the mobile so as to exploit search/query/reasoning facilities for the final user. The Web icon allows users to get access to the Mobile Medicine portal for direct download, publication and discussion about mobile medical content. The **Local Files Explorer & Indexer** module explores regularly the smartphone PDA memory storage to find new media files, it extracts metadata/descriptors and indexes them in the local database by the Content Indexer that extracts the semantic information from the content and performs the minimal content processing to process descriptors and simple files.

- **search and retrieve complex cross media content**

  ○ **making queries** on the mobile device searching for locally collected and directly accessible intelligent content. These queries may be full text or advanced, taking into account content classifications, file naming, grouping for types, taxonomical classification, etc., and general semantic descriptors and organizations;

  ○ **browsing on medical taxonomy**, **navigating into the content collection** organized for arguments or for intervention type to be performed (for example the pathology), or on the basis of other models and structure, etc.; commonly accepted medical ontologies are accessible;

- **execute complex cross media content** to get support for taking decision consulting content on the mobile. Content is executed with suitable players: PDF player, media player, etc. Cross media intelligent content items with internal semantic computing reach the PDA phones as MPEG-21 (for PDA) and as XML for iPhone. The different content formats are played by using their corresponding players. The Mobile Medicine content players may enforce interactivity and intelligence in the content depending on the user profile, user actions, context, GPS, communication with server, etc. iPhone version provides less behavioral capabilities and no DRM. The descriptor is used for indexing the content and collecting the semantic information. At each execution, some use data are collected to be exploited later on to provide recommendations.

- **receive updates automatically** on the mobiles with no user intervention (via the **Download/Update Manager**). This module collects content into the local storage of the mobile that is easily larger than 16 GB. It also records last date and/or obtained version, and it connects with the server to verify the availability of new versions, it downloads them and replaces versions by eliminating older ones;

- **receive personal suggestions** (local recommendations) which are computed on the basis of personal information collected on both server and mobile devices and privately offered on the mobile device: similar content, most used content, suggested content, etc. Similar content takes into account distance among the user profile as collected and/or declared and the descriptors of the objects.

Local recommendations are provided off-line and are based on: collected content semantics, user profile, collected use data from executions, navigations and queries. This information is used to make local suggestions such as the presentation of content according to the most played, less played, most recently played, less recently played, alphabetic order, taxonomical order, recently updated, etc. In this case, suggestions are produced on the basis of specific queries and not on clustering, as it is described in the rest of the paper as to server side suggestions. Each video of 45 min may be about 35 MB, while the single procedure is in the order of 20 kB. Therefore , thousands of objects may be collected into the smartphone. On the other hand, according to our experience 200 are more than enough

to cover the great majority of emergency conditions and deal with educational purposes. Personal recommendations are computed and provided in real time to the users on PDA and iPhones. In order to reduce the query time, the local database modeling the medical classification structure is provided precompiled into the installable file of the Content Organizers for both PDA and iPhone.

3 Suggestions via semantic computing

In this section, the details regarding the production of suggestions among *elements* such as: users, content, ads, and user groups, are reported. Among the possible combinations only some of them are viable as described in Table 1. Suggestions have to be computed on the basis of relationships U→U, G→C, C→U, etc. where U means User, G: Group and C: Content, thus C→U means proposing Content suggestions to Users. Similarly, other suggestions can be managed for other elements such as: mailing lists, play lists, etc. We prefer to call suggestions those computationally provided by the system; while, we refer to recommendations for those produced by the single users, when they recommend a content/ user to another friend, colleague.

The earliest solutions for guessing users' intentions have been based on keyword-based queries (i.e., sponsored search, or paid listing), which places ads and/or suggestions in the search results; and content match, also called content-targeted advertising, or contextual advertising, which places ads on the basis of the Web page content and content similarity [1, 15]. The latter is a harder problem with respect to the former.

Contextual suggestions are widespread and many systems can extract keywords from web pages to produce suggestions [30, 31], sometimes using semantic approaches [12]. In order to predict which terms are more relevant to a set of seed terms describing a product or service, models based on clustering, collaborative filtering, logistic regression, etc., are used, as in [2].

Table 1 Possible suggestions among elements

	Recipient of the suggestions		
	User	Content (played by a user)	Group (leader or members)
Suggested elements			
Users	Proposing to a user possible colleagues / friends	-no sense-	Proposing to a group responsible possible interested colleagues to be invited
Contents	Proposing to a user possible interesting contents	Proposing, when a content is played, similar content items	Proposing to a group member possible interesting content (not much different with respect to C–C combination)
Groups	Proposing to a user possible interesting groups	Proposing, when a content is played, possible interesting groups where similar contents are discussed	-no sense-
Ads	Proposing to a user possible interesting ads	Proposing, when a content is played, possible interesting ads	Proposing to a/all group member/s possible interesting ads

Suggestions can be computed through several different techniques by estimating similarities, making queries on keywords, adopting clustering or grid clustering, etc. In most cases, the elementary operation to estimate suggestions is the similarity distance among elements. When similarity has to be estimated on the basis of simple profiles/ descriptors, such estimation may have a limited complexity. On the other hand, the estimation of all the distances or similarities among elements can be computationally expensive in the presence of complex descriptors and/or millions of items, since it can be quadratic, an $O(N^2)$ (where N is the number of elements) or much more, depending on the complexity of the descriptors that may be based on several fields.

Therefore, the problem of providing suggestions is reduced to the estimation of similarities among elements, which in the Mobile Medicine are as follows:

- **Users** of the portal with information collected from users:

  ○ **static** aspects generically provided during registration. The static part is frequently not so much detailed in generic Social Networks, since users do not like filling in 'useless' forms and/or they tend to provide false data. In small thematic and business oriented Social Networks such kind of information is much more reliable. Among them:

    ▪ general information (name, surname, nickname, gender, age, location, foreign language skills, mother tongue, nationality, etc);
    ▪ instant messaging contacts (skype, messenger, ICQ, etc.);
    ▪ education and job, interests (content *Type* and *Format, or taxonomy*);

  ○ **dynamic** information collected on the basis of the activities the users perform on the portal elements, such as those on content, or on other users:

    ▪ votes and comments/annotations on: contents, forums, web pages, etc.;
    ▪ downloads and play/view/executions of content, web pages, etc.;
    ▪ uploads and publication of user provided content;
    ▪ mark content as preferred/favourite;
    ▪ recommend content/groups or users to other users;
    ▪ chat with other users, publish on groups;
    ▪ queries performed on the portal, etc.;
    ▪ create a topic in a forum or contribute to a discussion;
    ▪ relationships/connections with other users or groups;

- **Contents** can be files (audiovisual, document, images, audio, etc.) web pages, forum comments, advertising, professionally generated or not; In any case provided by some user or by an organization. The content may have:

  ○ **static** descriptors: more relevant since the content description is typically not changing over time. They are:

    ▪ metadata, keywords extracted from description, comments, etc.;
    ▪ technical description (as the Format in the following): audio, video, document, cross media, image,..;
    ▪ content semantic descriptors such as: rhythm, color, etc.; genre, called Type in the following;
    ▪ groups which the content has been associated with;
    ▪ taxonomical classification, ontological description, which the content has been associated with, taking into account also the general model domain;

- ○ **dynamic** aspects are marginally changed and may be related to:
  - ▪ user's votes, user's comments, user's profile limited to users playing that content;
  - ▪ number of votes, comments, download/play, direct recommendations, etc.;

- **Groups** of users that may have specific descriptors and those inherited by the users:

  - ○ **static** description of the groups such as:
    - ▪ objectives, topics, web pages, keywords, taxonomy, etc.;

  - ○ **dynamic** aspects related to:
    - ▪ users belonging to the group; users may: join and leave the group, be more or less active over time;
    - ▪ content associated with the group: files, comments, etc., with their taxonomical descriptor, metadata and descriptors.

On such grounds, most descriptors may change over time (see for example how they may depend on the user's actions: votes, play, download, comments, user joining a group, etc.) and therefore, distances and similarities should be updated at each change, even in real time. On the other hand, when it comes to cross media content, it can be very complex since you could shift from a simple classification into a multidimensional semantic classification. As to cross media content, what is meant is content which may have multiple format (audio, video, image, document and crossmedia). The crossmedia format files may include a collection of other contents glued together with a presentation layer based on SMIL, HTML, FLASH, ePub, and/ or MPEG-4 BIFS and all the related descriptors.

Moreover, when a new element is added (a new user, or group, or content, etc.), the estimation of a significant number of distances could be needed to both: provide suggestions and to consider it for the suggestions to be provided to other elements. The estimations of the new distances can be limited to the new added elements, G, with respect to those which are already included, M, so that the computational complexity can be limited to an $O(GM)$, when $M >> G$.

On such grounds, it is self-evident how high can be the costs for the semantic computing of suggestions. Solutions to reduce complexities are based on clustering techniques, grid and progressive clustering, and incremental estimations of similarity distances among elements. When the complexity of element descriptors becomes higher, as it occurs with crossmedia content or in the presence of complex descriptors, the problem of complexity management gets more evident. The basic problem is to model the similarity estimation among heterogeneous elements forming the description of an element, so as to guarantee any possible estimation, even among elements of different kinds and in the presence of uncertain and/or incomplete data. One solution to reduce the problem's complexity is to identify the minimal number of features (descriptor aspect) which are significant. This can be performed by using principal component analysis, PCA, or in any case by means of statistic analysis.

3.1 Clustering techniques

A solution to reduce the computational complexity of suggestions is the adoption of clustering techniques, which allow to group elements in families, for example by using k-means, k-medoids, hierarchical clustering, etc. This allows to reduce the complexity from N elements to K clusters, where K is typically much smaller than N. The suggestions are thus provided by estimating the similarity between the recipient of the suggestion and the closest

cluster descriptor. Thus, the effective suggestions may be directly provided by randomly picking elements from the most similar cluster, or by estimating specific distances/similarities on a smaller set.

Once the clusters are defined, they depend on similarity distances, which have static and dynamic parameters. It means that at each change of dynamic parameter or when a new element is added, a new clustering estimation would be needed. This can be avoided by applying the estimations only periodically and limiting the estimations to the new distances and to those that have been invalidated by changes. The new elements can be initially joined in a cluster according to its proximity with the descriptor representing the cluster (for example the cluster center).

Most of the clustering techniques can be applied only on numerically coded sets of values. For example, the k-means clustering unsupervised method is in most cases based on the possibility of estimating direct distance among elements. It provides quite good performance in terms of scalability; discovery of clusters with arbitrary shape; minimal requirements for domain knowledge to determine input parameters; ability to deal with noise and outliers; insensitivity to order of input records; and support for high dimensionality, it has typically a complexity of an $O(NKI)$, where N is the number of elements, K the number of clusters and I the number of iterations. Many other clustering algorithms exist, while the k-means has demonstrated the best performances when N is largely bigger than K and I [16]. The k-means assigns each point to the cluster, the centre of which is called centroid and it is the average of all the points in the cluster (it is not a real one).

The k-means algorithm starts by choosing the number of clusters, K (which can be determined by using statistical analysis or imposed); randomly generates K clusters centres; assigns each point to the nearest possible centroid by means of distance computation (see those presented in the following); computes again the cluster centres that minimizes the sum of the squared error in associating the points to clusters. The convergence is achieved by iterating the last two steps until no or minimal changes are performed on clusters. The k-means has been integrated into the AXCP tools and Mobile Medicine by using the Weka implementation [4].

One of the problems of k-means and other clustering solutions is their dependency on the availability of a numerical absolute distance measure between two numerical values, while the elements descriptors are mainly symbolic and in some cases with multiple values, coming from both the semantics and concepts they describe. This means that other solutions or numerical mapping or concepts are needed. Therefore, for some descriptors or metadata, the distance is quite simple to be estimated, for example as to the age; while in other cases it is very complex and may be arbitrary. Criteria to estimate the distance among element descriptors can be integrated by creating a weighted distance of factors. For example by taking into account also the number of matched or similar: *keywords* (using direct match or distance estimation which considers relationships in an ontology or taxonomy), *metadata information* (author name, creator name, registration date, location, etc.), see all the possible Dublin Core[2] terms.

In [4], the elements in the clusters were mainly content and users for entertainment application. Thus, descriptors were mainly on content *Format* and *Type* (genre). A model to map the distances among the concepts has been conceived to consider the affinity among them and common sense. *Format* would be image, document, audio, video, and crossmedia. For each *Format,* the *Type* may take about 15 different values. For example, for *Format*=audio, a *Type* would be jazz, pop, rock, folk, funk, disco, etc.; if *Format*=video,

---

[2] http://dublincore.org/

Type could be comedy, scifi, documentary, etc. Thus, a numerical mapping of descriptors for *Format* and *Type* has been presented based on cognitive aspects (see Fig. 6).

The approach has been based on producing a different tree when considering resource context: artistic (if it causes an emotional reaction) or informative; feeling: dark or positive (it is only related to the artistic resources); energy degree: energetic or calm (it is only related to the artistic resources); application: entertainment or mind work. It is referred to the resource's context of use; activity: active or passive (it is only related to the entertainment resources); kind: business or generic (it is only related to the mind work resources). Thus, a classification was defined for each element and hierarchy labels. The association of a numeric weight to each branch has been used to determine a numeric value for distances among attributes (if the attribute belongs to more than one branch, the average can be taken).

When the content or elements cannot be simply classified by genre (Type) and, thus, when the classification is more complex, for example involving taxonomy as in medicine, it is not easy to define absolute weights that create a precise ordering among all factors, along each dimension. This is the case of Mobile Medicine where medical classification taxonomy is accessible (see the Mobile Medicine portal). In that case, the distances between couples of elements have to produce coherent values depending on the semantics meaning of the medical terms. Each Element may have multiple associations into the medical taxonomy and each intermediate node may have multiple children.

Thus, for Mobile Medicine different semantic modeling and clustering techniques have been adopted, the *k*-medoids and/or hierarchical methods [16, 29], and finally a hierarchical evolution of the *k*-medoids. *K*-medoids is a partitioning technique which is based on the fact that one of the elements (called *medoid*) would be centrally located for each cluster. *K*-medoids adopts as a center of the cluster the element which has the minimal average (or the median) distance among the others involved in the cluster. This means that the complexity is grounded on $O(K(N-K)^2)$, that for $N>>K$ is an $O(N^2)$. This also means that initially the clusters' centers are some selected elements [29]. The algorithm is mainly implemented as follows:

- random selection of K points among the N, the medoids;
- associate each point of the N to the closest medoid (among the K) by using some distance metric, may be the Euclidean or others as described in the rest of the paper;
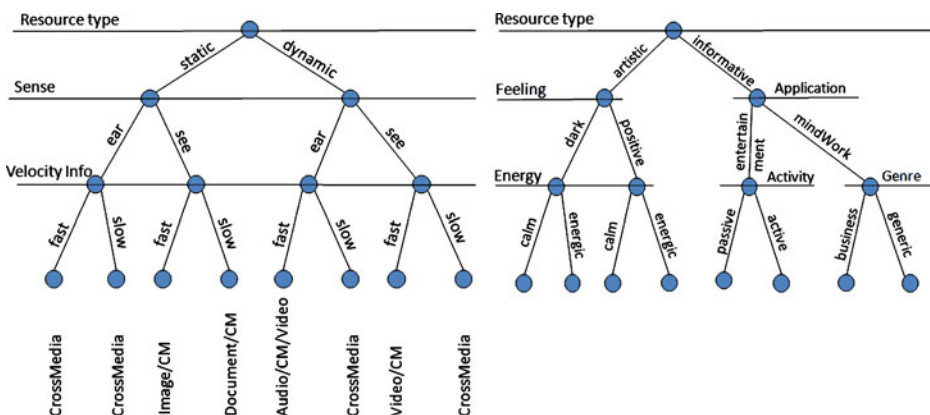

Fig. 6 Example of semantic distances among formats and types

- For each medoid m among the K

    ○ For each non-medoid n:

        ▪ Swap m and n and compute the global cost of the configuration
- Select the configuration with the lowest cost in terms of averaged distance of all the elements in the cluster;
- The last two steps have to be continued until no changes in *medoids* are accepted.

Hierarchical clustering [29] creates clusters on the basis of the distance among the single elements. The process starts by aggregating the closest elements to create smaller clusters of two elements and then aggregating these small clusters with other by following a sort a merging algorithm. The aggregation is based on the distance metric such as those discussed in the sequel of the paper. Hierarchical algorithms may differ for the mathematical model used for the merging of subclusters: complete linkage, single linkage and averaged.

3.2 Similarity distances

According to Table 1 and the descriptors adopted on Mobile Medicine, Table 2 reports the similarity distances used in the clustering model and to estimate suggestions. Please note that, despite the fact that Users, Content and Groups may all have both static and dynamic aspects, the dynamic aspects have been taken into account only for similarities involving Users. This has limited the complexity, while it should be noted that the metrics reported in this paper can be used to extend the estimations to consider dynamic aspects of Contents and Groups which are related to the activities by Users on them.

For example, the similarity distance $D(C,U)$ depends on both static and dynamic aspects: $D(C(s); U(s,d))$. It may be used to generate suggestion C→U taking into account the profiles of preferred contents of the Users, the dynamic information collected on the basis of the content play, favorite, highly ranked, etc., with respect to the content descriptors. In short, the similarity distance is practically reduced to estimate the distance between a content and the user represented by the descriptor of content he/she has: marked as preferred, positively ranked, uploaded, played, recommended, etc. Please note that preferred content description is a dynamic aspect of the user that presents information that can be compared with the static descriptions of the Content, for example in terms of taxonomical multilingual classification.

In general, the above-mentioned similarity distances can be scalar or vector data. For example, a scalar model may be obtained for the $D(U,U)$ taking into account both dynamic and static information:

$$D(U1; U2) = k_s \sum_{i=1}^{T_s} x_i Sd_i(U1, U2) + k_d \sum_{i=1}^{T_d} y_i Dd_i(U1, U2), \tag{1}$$

Table 2 Elementary similarity distances

	Recipient of the suggestions		
	User	Content (played by a user)	Group (leader or members)
Suggested elements			
Users	D(U(s, d); U(s, d))	-no sense-	D(U(s, d); G(s, d))
Contents	D(C(s); U(s, d))	D(C(s); C(s))	D(C(s); G(s, d))
Groups	D(G(s, d); U(s, d))	D(G(s, d); C(s))	-no sense-

Where:

- Sd$_i$ and Dd$_i$ are the distance metrics for a static and dynamic feature, respectively (static distances can be computed once);
- k_s and are weighting the static aspects with respect to the dynamic aspects and to adjust the scale factor among them according to the number of metrics and their ranges (they have to be fixed on the basis of the portal intention, on the other hand the dynamic aspects are much more reliable than static as previously commented);
- T_s, T_d are the number of static and dynamic features to estimate the similarity distance, respectively.
- x_i and y_i are the weights to give different relevance to the corresponding feature metrics.

The vector model leads to keep separate the single metrics:

$$D(U1;U2) = \left\{ \begin{array}{l} K_s(x_1 Sd_1(U1,U2), x_2 Sd_2(U1,U2), \ldots, x_n Sd_{Ts}(U1,U2)), \\ K_d(y_1 Dd_1(U1,U2), y_2 Dd_2(U1,U2), \ldots, y_n D_{Td}(U1,U2)) \end{array} \right\} \quad (2)$$

In this case, the distance can be Euclidean, for instance.

Both approaches can be used in the clustering techniques presented above. For example, in the case of $D(U,U)$ distances, a reduction of dimension could increase the computational cost, since users have a large set of metadata. In both cases, the single metric can be separately weighted more with respect to the others or they can be weighted per groups, for example, to give more relevance to dynamic aspects rather than to static. The weights of the above formulas (1) and (2) can be estimated by using a multilinear regression by considering the goals of the portal organiser or of the community of users [17, 27]. These techniques are used in the area of empirical assessment and evaluation.

In both cases, the problem is the definition of the single metrics. Thus, in the next sections, some of the Static (Sd$_i$) and Dynamic (Dd$_i$) metrics are described. They constitute the elementary blocks for the estimation for the similarity distances reported in Table 2.

3.3 Some static similarities in user profiles

Some static similarity distances can be very easily estimated as differences among numerical values, such as for ages, weights, height, etc.

For example, the similarity distance between two users on the basis of age can be estimated by using: Sda(User1, User2) = |Age(User1) − Age(User2)|/MAXdelta; where: MAXdelta is a value to normalize the age distance. If the different is greater, 1 is assigned.

Some of the distances between user features may not be significant for the identification of similarities among colleagues or friends in a certain context. Moreover, the most significant features for the estimation of conceptual similarity are frequently symbolic and may present multiple values (for example, a set of nationalities, a set of languages, a set of jobs, a set of taxonomy fields, a number of preferred localities, travels, etc.). Most of them are also dynamic and will be discussed later in the paper. Thus, similarity metrics are not simple to be determined and defined as described in the next sections.

In Appendix A, details about a selection of specific static metrics are reported, such as:

- Static metric on user's languages,
- Static metric on user's continent and nationalities,
- Static metric on user's medical/technical specializations,
- Static metric on user's Groups,
- Static metric on user's interested Taxonomy topics.

3.4 Some dynamic similarities in user profiles

The dynamic aspects of the user profile are collected on the basis of the activities the User performs on the portal and thus on other:

- **Contents**: supposing that acting on similar content may create a similarity between users or from users with content:

  ○ **positive**: downloads, set as preferred, recommend, publish, play, positive comments, high votes, ..;
  ○ **negative**: low votes, negative comments, ..;

- **Users**: supposing that acting on other Users is motivated by a similarity with them:

  ○ **positive**: set as friend, recommend, ...;
  ○ **negative**: negative comment, unconnect,... ;

- **Groups**: supposing that acting on some Group is due to a similarity between them:

  ○ **positive**: subscribe, contribute, associate a content with, etc.;
  ○ **negative**: leave, negative comment ...

It should be noted that, some of them may be based on either positive or negative impact on the similarity. For example, a vote could be positive over a given threshold while a comment can be positive or negative according to the meaning and context. In order to estimate the similarity we limited the assessment to the positive impacts and values assuming that a user would be better defined by means of its positive activities rather than via negative.

The major sources of similarity may be determined by the analysis of thematic classification for positive actions on Users, Contents and Groups. In Mobile Medicine, the thematic classification is modeled via medical taxonomy (for any element: User, Content, Group) and in addition using Format for content files. Other measures of User behavior aspects can be taken into account as dynamic metrics such as: duration of connection, number of sent emails/ messages, number of comments, number of connections, number of friends, etc. Most of them are marginally relevant in terms of similarity among elements; while some of them have been used as a basis of the estimation of other more relevant metrics as discussed in the sequel.

In Appendix B, details about a selection of specific dynamic metrics are reported, such as:

- dynamic metric on user's interested taxonomy topics,
- dynamic metric on user's interested formats,
- dynamic metric on user's preferred content items and colleagues.

3.5 Summary of similarity distances

Therefore, the above mentioned similarity distances can be used in the functions described in Table 2 and according to some model (vectorial or scalar) as described in Section 3.2. Thus, the resulting similarity distances can be defined as:

D(U,U)	*Function of (Sda(), Sdl(), Sdn(), Sds(), Sdg(), Sdt(), Sdf(), Ddt(), Ddf(), Ddp(), Ddc())*
D(C,U)	*Function of (Sdl(), Ddt'(), Ddf'())*
D(G,U)	*Function of (Sdl(), Sdt(), Ddt(), Ddf())*
D(C,C)	*Function of (Sdm(), Sdl(), Sdt(), Sdf())*

D(G,C)	*Function of (Sdl(), Sdt(), Ddt'(), Ddf'())*
D(U,G)	*Function of (Sdl(), Sdt(), Ddt(), Ddf(), Ddp())*
D(C,G)	*Function of (Sdl(), Sdt(), Ddt'(), Ddf'(), Ddp())*

Where, in the above definitions, only *Sdm()* has not been defined in the paper and represents a static similarity distance between content elements on the basis of their metadata, which can be based on a keyword distance models or on simple metadata matching or counting solution. On the other hand, this aspect is not presently been taken into account yet since it is marginal with respect to taxonomical and format classification models. Other functions are defined in Appendices A and B.

4 Experimental results on similarity estimations

In this paper, a solution to produce suggestions in the presence of complex descriptors has been proposed. The presented solution can be extended to a large range of cases as those of thematic social networks. The approach has generated a model to produce metrics able to estimate similarities among different data types. The metrics described have been adopted on Mobile Medicine social networks. According to a first validation, users did find the recommendations interesting. The proposed metrics have some weights (*Ki, Xi, Yi,..*) that have been tuned according to the social networks goals, and in general aiming to give more relevance to dynamic information rather than to the static information (since the latter are often incomplete and false). The Mobile Medicine portal is accessible on http://mobmed. axmedis.org from PC, PDA, iPhone and mobiles.

In this section, some experimental results are reported. They refer to the adoption of clustering techniques on Content elements, which is used to provide suggestions C→U, C→G, and C→C. This means that, according to definition of *D(C,C)* the clustering has been applied to content static aspects, while the suggestions have been performed on the basis of static-dynamic metrics such as *D(C,G)* or *D(C,U)* with respect to the clusters features.

A different solution would be to apply the clustering to all the elements by considering a subset of common features and thus of reasonable distances, for example, those based on taxonomical classification and content formats (preferred or commonly used):

D(C,U)	*Function of (Ddt'(), Ddf'())*
D(C,C)	*Function of (Sdt(), Sdf())*
D(C,G)	*Function of (Sdt(), Ddt'(), Ddf'())*

On the other hand, this solution can be applied only if static, dynamic and hybrid distances produce comparable results in terms of scale and meaning.

In the experiments performed, before taking a final decision about which kind of clustering algorithm use, four different algorithms have been considered: *k*-medoids, hierarchical with complete linkage, hierarchical with single linkage and hierarchical with average linkage [16, 29]. As explained in Section 3.1 *k*-means was excluded since not suitable for clustering Mobile Medicine symbolic descriptors.

The following graph reports the clustering about Content elements into the Mobile Medicine portal. Among the several aspects used to assess the above mentioned solution the variance about the number of objects per cluster is a good measure (see Fig. 7). This metric allows to understand if the clustering model may converge to a limited variance thus to a good balance of distribution of elements among the clusters. More or less all the techniques lead to reduce the variance with the increment of the number of clusters.

Another parameter is the classification error (see Fig. 8). This metric allows to understand if the clustering model may reduce the error with the increment of clusters. Also in this case the k-medoids seems to be better ranked with respect to the other techniques.

The parameter depicted in Fig. 9 is the trend of the variance about the distance into the clusters. In this case, the difference between k-medoids and some of the hierarchical clustering solutions are not much relevant. Most of them present very low values for the Error.

The final metric for this assessment is the percentage of elements out of the cluster with respect to those of the cluster that have an averaged distance lower with respect to the latter (see Fig. 10). Thus, is a sort of estimation of *false positive* for the clustering algorithm.

On the basis of the above reported evaluation the k-medoids resulted to be the most precise clustering solution. On the other hand, its computational complexity is very high with respect to the hierarchical models. Moreover, when the number of items is high, a stratified hierarchical model can be adopted. The solution proposed has been tested with generated content up to millions of items. The hierarchical approach is based on segmenting the total number of elements/objects in chunks of 1,000 and estimating C clusters for each chunk. The resulting centers of clusters can also be segmented in chunks of 1,000 elements and clustered in C clusters. This process may continue until the resulting number of centers clusters is less than the chunk dimension (1,000). At level 0 of the hierarchy, the clustering has to be performed on N elements; at level i on $N*(C/1000)^i$. According to this model, every time a new element is added it can be temporary classified in the closest cluster according to its similarity with the last level C centers. Once 1,000 new elements are obtained, they are removed from temporary status and the incremental clustering can restart re-clustering from level 1 to last (since the selected elements for each level is random, this allow to regularize the new solution). In this manner, the clustering is incrementally performed. In Fig. 11, the comparison of execution time for total clustering of all the N elements/objects or for clustering only the last 1,000 and recreating incrementally the hierarchical clusters re-estimating only those at the higher levels above the 0.

From Fig. 11, it is evident that the adopted incremental clustering allowed us to reduce the computational complexity of the problem. To make the difference more evident, please note that with 1 millions of elements, the total clustering time is in order of 5.1 days (including data access), while the incremental clustering of the last 1,000 elements is performed in 4.49 h only (including data access), resulting time saving is of 27 times. With 2 millions of elements the resulting time saving is of about 61 times. The reported execution times have been provided referring to the execution on a single grid node. The
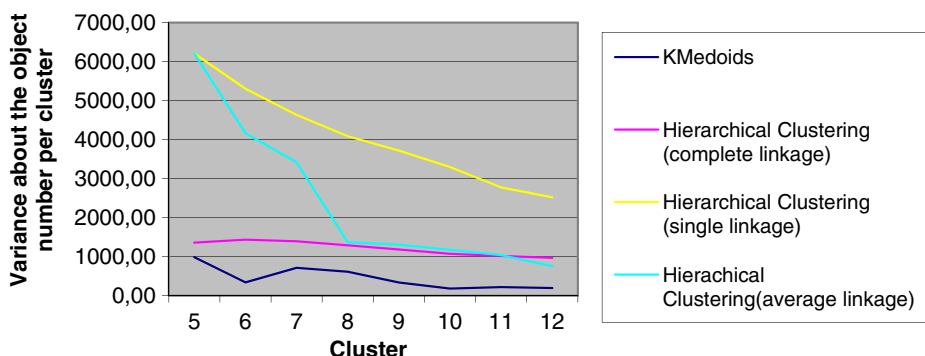

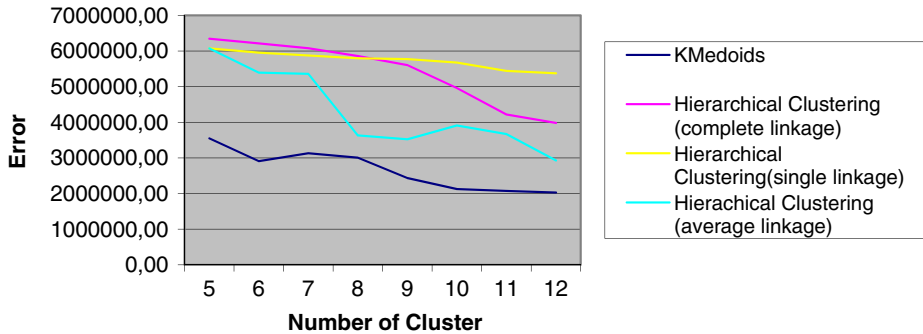Fig. 7 Trend of variance of the number of objects with respect to the number of clusters

Fig. 8 Classification error with respect to the number of clusters

adopted algorithm for hierarchical clustering is based on clustering chunks of 1,000 elements, the execution time depends on both the clustering time (almost constant for each chunk of 1,000 elements), and the data acquisition time, which depends on the database size. According to the total number of objects, the number of levels grows, and also the computational time to access at the needed symbolic data from the database exponentially increases. A relevant speed up can be obtained by increasing the number of nodes, since the clustering of the single chunks are quite independent each other.

5 Exploiting Mobile Medicine content

Mobile Medicine supports a large variety of content formats and distribute them towards a number of platforms. In Fig. 12, the distribution of content per (a) device/platform, and (b) per format types are reported. The figures reported the distribution measured from the portal in last 10 months on the portal. It can be noted that the most diffused access is PC, while among mobile the iPhone with Content Organizer (Mobile Medicine App) is preferred. As mentioned before, Mobile Medicine offers a range of content formats that may be accessible by different types of devices and platforms. The distribution highlights that cross media content is the most accessed format, followed by video and HTML content pages (still in this content, interactive elements similarly cross media). On mobiles, the most selected content formats are cross media with the 82% and video with 9%.
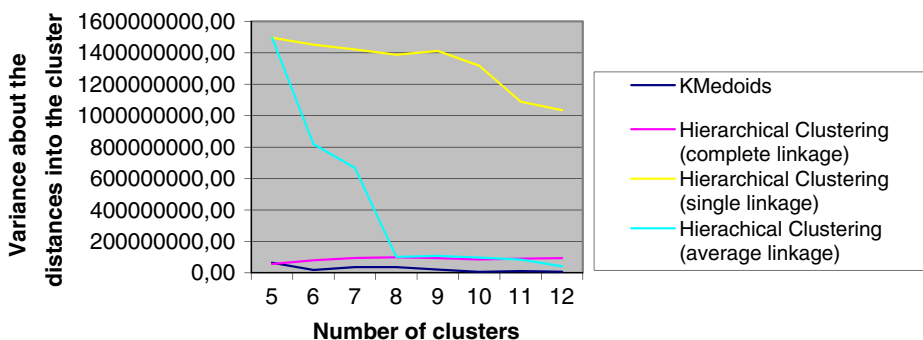

Fig. 9 Trend of variance about the distances into the cluster with respect to the number of clusters
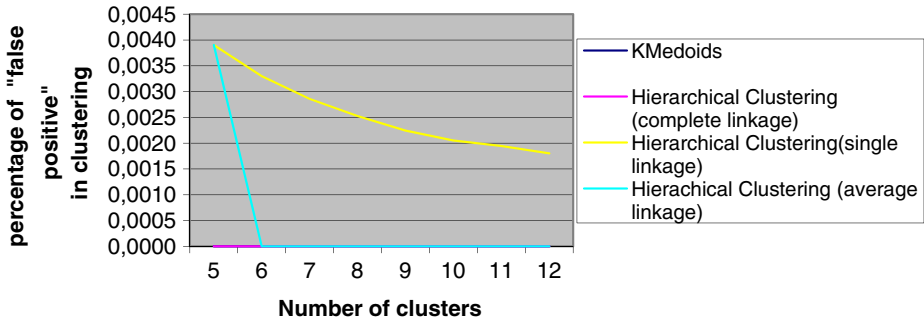
Fig. 10 Trend of a sort of *false positive* classifications with respect to the number of clusters

Moreover, a further analysis identified that in Mobile Medicine, the 63.8% of users accessed the portal to get/play a single content even if they returned back later in the next days. The 36.2% of them stay for two (14%) or more content items. Among them the 58% accepted suggestions, selecting content from the provided lists, others returned back of the main list of content and query results.

6 Conclusions and future tasks

In this paper, a model and solution to take advantage of semantic computing capabilities in the context of mobile medical applications, has been presented. A collection of semantic computing technologies together with innovative intelligent content model tools allowed providing an innovative integrated service for medical personnel, including personal content management and decision support on smarphones. The proposed solution has been developed by exploiting and expanding MPEG-21/AXMEDIS content format with semantic information and processing tools hosted on the back office and locally on mobile devices. A certain intelligence has been enforced into the mobiles by designing and developing a Content Organizer (called ObjectFinder) that can be installed in any Windows Mobile devices and on iPhone (via Apple Store), and iPad. Moreover, the intelligent cross media content presents a support for implementing complex medical procedures to provide a decision support to medical personnel in off-line and on-line conditions, and also for
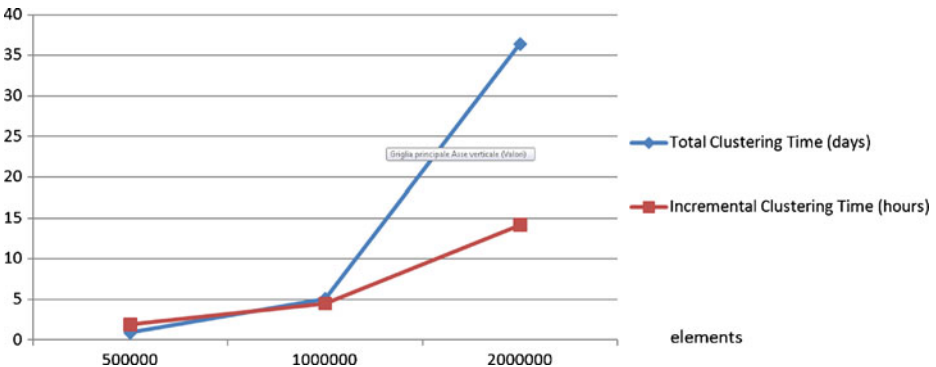

Fig. 11 Execution time for hierarchical clustering: total clustering vs incremental clustering, a scale factor of 24 h (a day), has been imposed to make them visible in the same graph

Fig. 12 Percentages of distribution of content accesses as preferred by the users: **a** per device/platform, and **b** per content format/type, where: PC-windows+ie means accesses by using PC with Windows operating systems and Internet Explorer; i-Phone-app means accesses by using Content Organizer Mobile Medicine application which can be downloaded via AppStore; PC-MacOSx means accesses via Mac OSx; PDA means accesses by using Windows Mobile PDAs with Content Organizer; i-phone means direct accesses by using iPhone integrated internet browser; PC-Others includes Unix like systems, etc
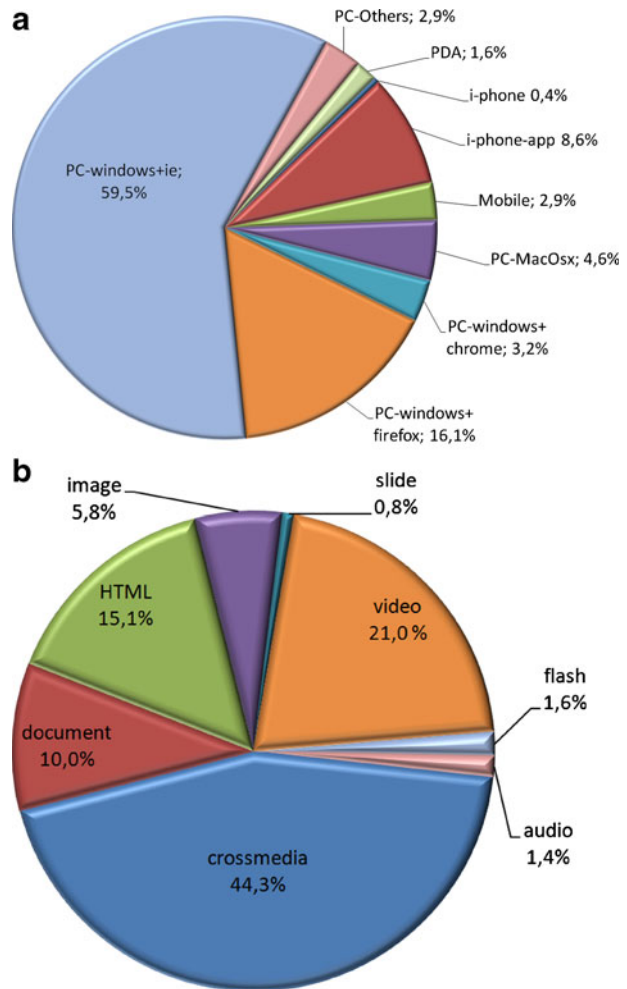
continuous medical education. The solution is under trial at the major Florence (Italy) medical center. The mobile content distribution service is integrated with a collaborative networking portal, where suggestions are provided from the server and computed by the AXCP grid. Thus, the proposed solution presents activities of semantic computing at both servers and client sides. For the suggestions on server side on the basis of use data and profiles, clustering techniques have been used. K-medoids and hierarchical clustering with averaged linkage have demonstrated to be the most interesting solutions, while a hierarchical incremental model has been used to cope with large populations. On this regard, some direct estimations measured on the platform have been provided regarding the execution time on clustering and the distribution about the preferred user content access.

Mobile Medicine portal and tools are all freely accessed, from public or via a free registration. All the mobile device applications and production tools can be freely downloaded from the help page of the Mobile Medicine portal: http://mobmed.axmedis. org and from the Apple Store, iPad version has been only locally provided. From the same page it is also possible to download other production tools for authoring complex intelligent content (namely: ADF Designer and AXMEDIS Editor).

Appendix A: main static metrics

Static metric on user's languages

The users may have one or more languages and they may be very useful for estimating the similarities among users in multilingual portals and communities. To this end, the following metric has been defined. In a multi-language and multicultural portal the matching on languages is very important to create friendship and communications.

As a first step, a matrix $m[i][j]$ is created where each element represents the similarity between the two languages, i,j. The matrix holds properties such as: $m[i][j]=m[j][i]$ and $m[i][j]=1$ if $i=j$. To generate matrix $m[i][j]$ a model has been used according to the above rationales and the following conditions:

- all languages are classified into families (Latin, Anglo-Saxon, Slovenian, Asian, etc.), groups and subgroups;
- to each leaf, one language or a set of similar languages is assigned;
- to each branch a numeric weight, $w(fi)_{treeLevel}$, is assigned where i is the number of families.

Thus for the hierarchy, the following property holds:

$$0 \leq \sum_{j=1}^{BL} w(fi)_j < 1; \text{ and } w(fi)_1 < w(fi)_2 < ... < w(fi)_{BL}.$$

Where: j is the tree level for language family fi; BL is the height to reach the leaf in the tree.

The numeric value of element $m[i][j]$ representing the similarity between two languages is calculated adding the weights of all the common branches. Therefore, the similarity $Sdl()$ is calculated taking into account all the languages chosen by the users involved and takes the maximum of similarity. For example, if User1 selects $(L1, L4)$ and User2 selects $(L2, L3, L5)$:

$$Sdl(User1, User2) = \max\{m_{12}; m_{13}; m_{15}; m_{42}; m_{43}; m_{45}\}$$

Example:
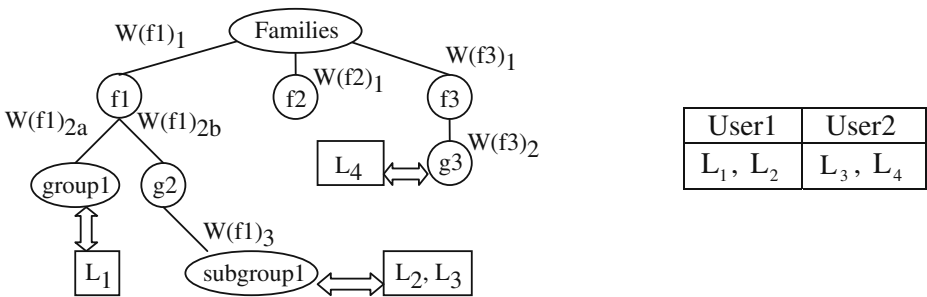
$$\begin{cases} Sdl_1(U1, U2) = \max\{m_{13}; m_{14}; m_{23}; m_{24}\} \\ where: m_{13} = W_{f1\_1}, m_{14} = m_{24} = 0, m_{23} = W(f1)_1 + W(f1)_{2b} + W(f1)_3 \end{cases}$$

Static metric on user's continent and nationalities

The users may have one or more Nationalities/Passports, city and location. This information may be very useful for estimating the similarity among users in multicultural portals and communities. To this end, the following metric has been defined. As a first step, matrix $n[i][j]$ is created where each element represents a similarity between two nations. The matrix has properties such as: $n[i][j]=n[j][i]$ and $n[i][j]=1$ if $i=j$. Thus, the similarity $Sdn()$ is calculated by taking into account the Nationalities and cities chosen by the users:

$$Sdn(U1,U2) = (n[\text{Nation}(U1)][\text{Nation}(U2)] + \lambda_1(U1, U2) + \lambda_2(U1, U2))/3$$

$$where\ \lambda_1(U1, U2) = \begin{cases} 1, & \text{if Region}(U1) = \text{Region}(U2) \\ 0, & \text{otherwise} \end{cases}, and :$$

$$\lambda_2(U1, U2) = \begin{cases} 1, & \text{if City}(U1) = \text{City}(U2) \\ 0, & \text{otherwise} \end{cases}$$

To generate the values of matrix $n[i][j]$ a decision tree is used, according to the following concepts. The world is divided into 5 continents, each of them contains several Nations (nearby nations are grouped under the same leaf). Therefore, to each branch a numeric weight, $w(ci)_{treeLevel}$, is assigned $0 \le \sum_j w(ci)_j < 1$; j is the tree level (the tree height is 2), and $w(c2)_1 < w(c2)_2$. The same model has been extended to regions and ethnical groups. The numeric value of element $n[i][j]$ representing the similarity is calculated adding the weights of all the common branches. Therefore, the similarity $Sdn()$ is calculated taking into account all the nationalities and regions chosen by the users and takes the maximum of similarity.

Example:
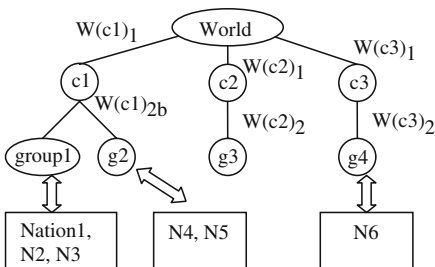
User1	User2	User3
N2	N4	N4
country1, city1	country2, city2	country2, city3

$$\begin{cases} Sd_2(U1, U2) = \left(\frac{n_{24}+\lambda_1+\lambda_2}{3}\right) = \left(\frac{W(c1)_1+0+0}{3}\right) \\ Sd_2(U2, U3) = \left(\frac{(W(c1)_1+W(c1)_{2b})+1+0}{3}\right) \end{cases}$$

Static metric on user's medical/technical specializations

In the medical area, the users may have collected during their studies one or more specializations, jobs and/or roles. Thus, they may be very useful for estimating the similarity among users in thematic portals and best practice networks. They can be presented as a predefined set of possibilities during the registration. To this end, the following metric for assessing similarity among users' has been defined.

As a first step, matrix *S[i][j]* with the similarities among specializations has been created. Each Element *S[i][j]* of the matrix represents the similarity between two specializations *i, j*; where: $0 \leq S[i][j] < 1$ if i≠j; while *S[i][j]* = 1 if they are identical, i = j. In order to build matrix *S[i][j]*, a decomposition in medical areas, subareas and specializations have been performed. Thus, a hierarchy has been created in which leafs correspond to the specialization classes. A set of weights $w_i(n)$ have been assigned to the branches of the tree, where *i* is the tree level in area n_i according to:

$$w_1(n) \geq 0, w_1(n) \leq w_2(n), 0 \leq \sum_{i-1}^{2} w_i(n) < 1.$$

The similarity between two specializations is the sum of the weights on the branches in common up to the root.

For example, if user *A* has set of specializations $P_A = \{p_{A1}, p_{A2}, \ldots, p_{AN}\}$, where $AN = Card(P_A)$, and user B has $P_B = \{p_{B1}, p_{B2}, \ldots, p_{BN}\}$, then, each couple of specializations taken from P_A, P_B determines the couple *i,j* and thus a value in matrix *S[i][j]*. Thus, the value of similarity *Sds()* for users *A,B* is estimated by using matrix *S[i][j]*, by means of the following model:

$$Sds(A, B) = \frac{\sum_{i \in P_A} \max_{j \in P_B}(S[i][j])}{AN}.$$

Therefore, each specialization of user *A* is compared with all the specializations of user *B*. The maximum value among them, for each specialization of *A,* is used to estimate an averaged value of similarity. Because each value of *S[i][j]* is limited to 1, also the averaged final value of *Sds()* is bounded from 0 to 1, and the metric is not symmetric: Thus, *Sds[A][B]* may be different with respect to *Sds[B][A]*. This is due to the fact that the sets of the specializations for the two Users may have different cardinality, while the metric is normalized with respect to the size of the reference User. A way to create a symmetrical metric could be to perform the estimation only on the basis of the parts in common between the two sets. On one hand, a lower precision is obtained, while a non symmetric metric is not creating any problems to semantic computing goals.

Static metric on user's groups

In collaborative portal and social networks, the users can create groups of discussion or thematic groups. These groups share commons goals and thus can be used to characterize the user profile. Typically, the users join and leave groups sporadically, thus this feature has been considered static even if it is dynamic. On the other hand, it is very probable that an user join a group during his/her life in the community, and not immediately at the registration time.

Given user *A*, its groups are G_A, and for user *B*, G_B. Thus, the similarity in this case can be directly estimated by using:

$$Sdg(A, B) = \frac{card(G_A \cap G_B)}{card(G_A)}.$$

For example: if user *A* is registered to 8 groups, user *B* to 4, and only 3 groups are in common, thus: *Sdg()=3/8*. The maximum value for *Sdg()* is obtained when both users are subscribed to the same groups, independently of their number. *Sdg()* is bounded from 0 to 1 and it is not symmetric.

Static metric on user's interested taxonomy topics

In Mobile Medicine CNP, each element, and thus also Users and Content items, may be classified with a set of terms taken from the chosen medical taxonomy. In the case of Users, the taxonomical classification is performed during registration and represents the areas of interest and/or of competences of the User. In this sense, the taxonomical classification is very important to estimate the similarity between users, and also for estimating suggestions such as: $C{\rightarrow}U$, $G{\rightarrow}U$, etc.

The taxonomy is a structure in which descriptive terms are present along the structure. Each term may have one or more children and one or more parents; children are typically specializations of their parents. The taxonomy can be represented as graph in which the braches can be weighted on the basis of their distance from the root. A maximum distance has been fixed to D, and a weight of $D/4$ has been assigned to the first level children of the root. The next branches have been weighed dividing the weight of the first level children for the distance of the root.

The similarity distance between a couple of taxonomy terms is reported in matrix $t[i][j]$, of TxT. The values of the matrix are generated on the basis of weights associated with branches along the minimal path (in term of the number of branches) between the two terms i,j:

$$t[i][j] = \frac{D - \sum_{k \in \min path\{i,...j\}} W_k}{D}$$

The estimation of this matrix accelerates the general computing of similarities since it avoids performing the single estimations for each comparison of Elements having the taxonomical classification: Users, Content, Groups.

Therefore, if element A has a taxonomical classification $T_A = \{r_{A1}, r_{A2}, \ldots, r_{AN}\}$, where AN is the number of terms, and element B has $T_B = \{r_{B1}, r_{B2}, \ldots, r_{BN}\}$, thus, each couple of taxonomical terms taken from T_A, T_B determines the couple i,j , and thus a value in matrix $t[i][j]$. Then, the value of similarity for elements A, B in terms of taxonomy is estimated by means of the following model:

$$Sdt(A, B) = \frac{\sum_{i \in T_A} \max_{j \in T_B}(t[i][j])}{AN}.$$

Each taxonomical term of element A is compared with respect to all the terms of element B. The maximum value among these similarities for each term of A is used to estimate an averaged value of similarity. Since each value of $t[i][j]$ is limited to 1, also the averaged final value of $Sdt()$ is bounded from 0 to 1, and it is not symmetric.

When content is uploaded and when a group is created a taxonomical classification is assigned. Thus, this static measure of similarity can be applied to estimate the similarity between Content and/or Group elements on the basis of their taxonomical classification.

Appendix B: main dynamic metrics

Dynamic metric on user's interested taxonomy topics

In most cases, the taxonomical classification of positive actions is much more relevant than the static expression of interest in terms of taxonomy provided by the users, since they are in the 85% of cases not provided with the needed level of attention.

The taxonomical classification of Users, Contents, Groups acted by each specific User is a dynamic additional information representing its interested/preferred topics (played, marked, positively commented, recommend, etc.). Thus, a dynamic measure of similarity can be estimated keeping trace of taxonomy of all the elements touched during dynamic activities. To this end, in Mobile Medicine, a dynamic profile for the taxonomical classification is associated with each User (and Group) on the basis of positive actions dynamically performed on Content, and several counters for the different action types mentioned above. The distinct actions could be weighted in different manner or could create separate dynamic profiles without reducing the validity of the general model presented.

The dynamic taxonomical classification profile of an User (or Group) consists of vector $dt[i]$ modeling each term i of the taxonomy with a function $F()$: $dt[i] = F(taxonomy, i)/NOA$, where NOA is the Number Of Actions performed on an element (e.g., Content), and $T=Card(dt)$. Each element (e.g., Content) presents a static set of taxonomical terms identified by their corresponding indexes: $te = \{\alpha, \beta,...\}$, they have been determined during its creation.

When an action is performed by a User (or Group) on an element (e.g., Content) the corresponding counter NOA is increased, and the vector of taxonomical terms $dt[i]$ is updated according to the terms of taxonomy of the latter element $te\{\}$ by using:

$$dt_{(n+1)}[i] = \frac{(dt_n[i] \cdot NOA) + 1}{(NOA + 1)}, \text{ if } i \in te \text{ otherwise by using}: dt_{(n+1)}[i] = \frac{(dt_n[i] \cdot NOA)}{(NOA + 1)}$$

So that the new version of vector $dt[i]$ is produced as: $dt_{(n+1)}[i]$, where $0 \leq dt[i] \leq 1$. $dt[i]$ is the percentage of times the user has chosen an object associated with that taxonomy node.

Therefore, it is possible to estimate a similarity value between two elements (Users, Contents, Groups) on the basis of the above described dynamic taxonomical classification profile and the similarity distance matrix defined for $Sdt()$.

If element A has a taxonomical classification $dt_A[i]$, and B has $dt_B[j]$; thus, each couple of taxonomical terms determines the couple i,j (when $dt_A[i] \neq 0$ and $dt_B[j] \neq 0$), and thus a reference into matrix $t[i][j]$, of TxT. The value of similarity for elements A, B in terms of taxonomy is estimated by means of the following model:

$$Ddt(A, B) = \frac{\sum_{i=1}^{T} \max_{j=1...T}(t[i][j] \cdot (1 - |dt_A[i] - dt_B[j]|) \cdot dt_A[i] \cdot dt_B[j])}{NDTANN}.$$

Where $NDTANN$ is the number of non-null items of dt_A. Each taxonomical term of element A is compared with respect to all the terms of element B. The maximum value among these similarities for each term of A, weighted for the distance in the profile vectors, is used to estimate an averaged value of similarity. The value of $Ddt()$ is bounded from 0 to 1 and it is not symmetric.

This model can be used to estimate the similarity distance between a static taxonomy profile and a dynamic taxonomic profile as needed in comparing terms of taxonomy of Content and Users, or of Groups and Content. Thus, in these hybrid (static dynamic) cases, the above reported distance takes the form as:

$$Ddt'(C, U) = \frac{\sum_{j=1}^{T} \max_{i=1...T}(t[i][j] \cdot dt_U[j])}{NDTUNN}$$

Where $NDTUNN$ is the number of non-null items of dt_U.

Dynamic metric on user's interested formats

In Mobile Medicine, or in others portals with communities, several symbolic descriptors may exist such as the content Format that may assume enumerate values. For example for the format: image, document, images, video, audio, etc. To know the preferences of the users on these formats may be very important to present ads or to estimate similarities and thus suggest such as U→U, C→U, and C→G.

Also in this case, a dynamic vector of preferences can be created, *df[i]*, associated with the counting of the actions related to content having those formats. This allows understanding which are the preferred Formats for a certain User or Group. The vector *df[i]* is kept updated at each action with similar equations adopted for vector *dt[i]*. So that the new version of vector *df[i]* is produced as: $df_{(n+1)}[i]$, where $0 \leq df[i] \leq 1$, and $F=Card(df)$.

In order to estimate the similarity distance been formats a similarity model has been defined among the different values of *Format*. In this case, the model proposed in [4] and presented in Fig. 7 has been used. Thus, a similarity matrix *f[i][j]* of *FxF* has been produced reporting all distances among *Formats* values.

Therefore, similarly to the dynamic similarity between two elements, *A,B,* based on terms of taxonomy we have for the dynamic similarity between two elements (e.g., U→U) based on formats:

$$Ddf(A,B) = \frac{\sum_{i=1}^{F} \max_{j=1...F}(f[i][j] \cdot (1 - |df_A[i] - df_B[j]|) \cdot df_A[i] \cdot df_B[j])}{NDFANN}$$

Where *NDFANN* is the number of non-null items of df_A. This model can be used to estimate the similarity distance between a static format and a dynamic format profile as occur in comparing in terms of formats for Content and Users, or Groups and Content. Thus, the above reported distance is transformed as:

$$Ddf'(C,U) = \frac{\sum_{i=1}^{F} (f[i][Format(C)] \cdot df_U[i])}{NDFUNN}$$

Where *NDFUNN* is the number of non-null items of df_U.

Dynamic metric on user's preferred content items and colleagues

The Users may typically marks as preferred some elements (Content, Users). In the case of Users they are referring to friends or colleagues. A similarity metric can be defined in order to weight the proximity of two Users/elements on the basis of their preferred elements. Thus, metrics similar to *Sdg()* (defined in section "Static metric on user's groups") have been adopted for the preferred content, *Ddp(),* and preferred/linked colleagues, *Ddc(),* by using the ratio between the number of elements they have in common and those of the reference User. Also in this case, the metric is bounded from 0 to 1 and it is not symmetric. Please note that metric *Ddp()* can be also applied to similarities between Users and Groups, and to Content and Groups.

- *Ddp(U,G):* the ratio between the number of elements a User has in common with the Group and those of the reference User (the elements in commons are typically Contents).

- *Ddp(C,G):* the ratio between the number of elements a Content has in common with the Group and those of the reference Content (the elements in commons are typically Users).

References

1. Anagnostopoulos A, Broder AZ, Gabrilovich E, Josifovski V, Riedel L (2007) Just-in-time contextual advertising. CIKM'07, November 6–8, Lisboa, Portugal
2. Bartz K, Murthi V, Sebastian S (2006) Logistic regression and collaborative filtering for sponsored search term recommendation. EC'06, June 11–15. Ann Arbor, Michigan
3. Behrendt W, Gangemi A, Maass W, Westenthaler R (2005) Towards an ontology-based distributed architecture for paid content. In: ESWC 2005, LNCS 3532. Springer-Verlag Berlin Heidelberg, pp 257–271
4. Bellini P, Bruno I, Fuzier A, Nesi P, Paolucci M (2009) Semantic processing and management for multichannel cross media social networking. Proceedings of 2009 NEM Summit "Towards Future Media Internet", 28–30 September. Saint Malo France
5. Bellini P, Bruno I, Nesi P (2005) A distributed environment for automatic multimedia content production based on GRID. In: Proc. AXMEDIS 2005, Italy, 30 Nov. 2 Dec., pp 134–142, IEEE Press
6. Bellini P, Bruno I, Nesi P (2006) A language and architecture for automating multimedia content production on grid. Proc of the IEEE Int Conference on Multimedia & Expo (ICME 2006). IEEE Press, Canada, 9–12 July
7. Bellini P, Bruno I, Nesi P, Rogai D (2007) MPEG-21 for the multi-format and multi-channel content production, protection and distribution. Proc Int Broadcasting Conf, IBC 2007, Amsterdam, Sept
8. Bellini P, Bruno I, Nesi P, Spighi M (2008) Taxonomy and review of Complex Content Models. In: Proceedings of the 14th Int. Conf. on Distributed Multimedia Systems (DMS'2008). Boston, USA, 4–6 September, pp 71–76
9. Bellini P, Bruno I, Nesi P, Spighi M (2008) Intelligent content model based on MPEG-21. AXMEDIS '08. International Conference on IEEE press Automated solutions for Cross Media Content and Multichannel Distribution, 17–19 Nov, pp 41–48
10. Bellini P, Nesi P, Ortimini L, Rogai D, Vallotti A (2006) Model and usage of a core module for AXMEDIS/MPEG-21 content manipulation tools. IEEE Int Conf Multimedia & Expo (ICME 2006). Toronto, Canada, 9–12 July, pp 577–580
11. Bellini P, Nesi P, Rogai D (2007) Exploiting MPEG-21 file format for cross media content. 13th Int. Conf. on Distributed Multimedia Systems (DMS), San Francisco Bay, USA, 6–8 September
12. Broder A, Fontoura M, Josifovski V, Riedel L (2007) A semantic approach to contextual advertising. Proc. of 30th annual int. ACM SIGIR conf. on Research and development in information retrieval, SIGIR '07
13. Burnett IS, Davis SJ, Drury GM (2005) MPEG-21 digital item declaration and identification-principles and compression. IEEE Transactions on Multimedia 7(3):400–407
14. Burnett I, Van de Walle R, Hill K, Bormans J, Pereira F (2003) MPEG-21: goals and achievements. IEEE Multimedia 10(4):60–70
15. Ciaramita M, Murdock V, Plachouras V (2008) Semantic associations for contextual advertising. Journal of Electronic Commerce Research 9(1):1–15
16. Everitt B, Landau S, Leese M et al (2001) Cluster analysis. Arnold, London
17. Fioravanti F, Nesi P (2001) Estimation and prediction metrics for adaptive maintenance effort of object-oriented systems. IEEE Transactions on Software Engineering, IEEE Press, USA
18. Hart J, Ridley C, Taher F, Sas C, Dix A (2008) Exploring the Facebook experience: a new approach to usability. NordiCHI 2008: Using Bridges, 18–22 October, Lund, Sweden
19. Kruitbosch G, Nack F (2008) Broadcast yourself on YouTube—really? HCC'08, October 31, 2008, Vancouver, British Columbia, Canada
20. Lee J, Hwang SO, Jeong S-W, Yoon KS, Park CS, Ryou J-C (2003) A DRM framework for distributing digital contents through the internet. ETRI Journal 25(6):423–435
21. Michalowski W, Rubin S, Slowinski R, Wilk S (2003) Mobile clinical support system for pediatric emergencies. Source, Decision Support Systems, Volume 36, Issue 2. Elsevier, Amsterdam (October), pp 161–176

22. Mourad M, Hnaley GL, Sperling BB, Gunther J (2005) Toward an electronic marketplace for higher education. Computer of IEEE, pp 58–67, June
23. Nazir A, Raza S, Chuah C (2008) Unveiling Facebook: a measurement study of social network based applications. IMC'08, October 20–22, Vouliagmeni, Greece
24. Nesi P, Rogai D, Vallotti A (2006) A protection processor for MPEG-21 players. In Multimedia and Expo, 2006 IEEE International Conference. 9–12 July, Toronto, Canada, pp 1357–1360
25. Pereira F, Ebrahimi T (eds) (2002) The MPEG-4 book. IMSC Press, Prentice Hall
26. Protogerakis M, Gramatke A, Henning K (2009) A system architecture for a telematic support system in emergency medical services, ICBBE 2009. 3rd International Conference on Bioinformatics and Biomedical Engineering, Issue, 11–13 June 2009:1–4
27. Rousseeuw PJ, Leroy AM (1987) Robust regression and outlier detection. Wiley, New York
28. Schellner K, Westermann GU, Zillner S, Klas W (2003) CULTOS: towards a world-wide digital collection of exchangeable units of multimedia content for intertextual studies. In DMS 2003
29. Xui R, Wunsch DCII (2009) Clustering. Wiley, USA
30. Yih W, Goodman J, Carvalho VR (2006) Finding advertising keywords on web pages. WWW 2006, May 23–26. Edinburgh, Scotland
31. Zhang Y, Surendran AC, Platt JC, Narasimhan M (2008) Learning from multitopic web documents for contextual advertisement. KDD'08, August 24–27. Las Vegas, Nevada, USA

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.

Alice Fuzier is an Engineer and contractor at the University of Florence. Her research interests include social networking and clustering. She has worked on worked on projects: ECLAP and mobile medicine.

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.

Michela Paolucci is an Engineer and Ph.D. student at the University of Florence. Her research interests include social networking and collaborative work. She has worked on worked on projects funded by the European Commission such as: ECLAP, IMAESTRO, mobile medicine and AXMEDIS.