eUPB: Towards an Integrated e-Service Platform in Large Scale Distributed Environments

Florin POP, Mugurel ANDREICA, Ciprian DOBRE, Razvan RUGHINIS, Alin MOLDOVEANU, Costin BOIANGIU, Mihai IONESCU

Faculty of Automatic Control and Computers, University POLITEHNICA of Bucharest, ROMANIA
E-mails: firstname.lastname@cs.pub.ro

Abstract: The main goal of this paper is to present the architecture and functionality of e-Service based platform. The project is structured along several dimensions that follow the development of complementary services, integrated to support everyday work experience, research and learning in the University POLITEHNICA of Bucharest (UPB). The platform support is represented by Internet as a large scale distributed environment. The current evolution of the Internet can be viewed from multiple perspectives: service oriented (Internet of Services), users centered (Internet of People), real-world integration over Internet (Internet of Things), production and use of multimedia content over Internet. The main services in eUPB[1] platform are: (i) data retrieval, aggregation and search service, (ii) communication service for heterogeneous networks, (iii) mobile services to support context-aware applications, (iv) secure data delivery infrastructure for wireless sensor networks, (v) 3DUPB - the 3D MMO virtual replica of UPB, (vi) analysis and content extraction of scanned documents, and (vii) collaboration service. This is a position paper presenting the general architecture of eUPB and a description of each services device and functionality.

Keywords: e-Services, distributed platform

1. INTRODUCTION

We live in a dynamic and changing world. In this context e-Services represent the transposition of the traditional services in the virtual world. The number of users that moved from physical world into digital world grows exponential in last decade. The e-Services are easy to use, permanent and they have continuous access, direct communication, timely and consistent information. e-Services are exposed to all kind of threads from the Internet, so when it comes about their security or other challenges, it should receive a special attention. In a digital world, when it comes about e-Services, there are some main characteristics that are common to all of them. One of them is scalability, which indicates its ability to handle growing amounts of work in a graceful manner or its ability to be enlarged. Security challenges are imposed at each step. Data encryption, password protection and account creation are other subjects discussed and applied during the development of the e-Service system. A large number of users characterize e-services and they must be able to respond to all their requests.

We proposed in this paper an integrated platform based on service oriented architecture being user-centered, offering data integration, production and use of multimedia content. The project aims to develop an integrated set of services designed to support the academic community in University POLITEHNICA of Bucharest (UPB). The system is called eUPB. eUPB will increase efficiency, collaboration and professional socialization, using the latest technologies in information processing and communication. eUPB project is structured along several dimensions that follow the development of complementary services, integrated to support everyday work experience, research and learning in the UPB.

Data retrieval, aggregation and search service aims to provide a set of features including: automatic collection of information, guided and focused retrieval of researcher profiles, aggregation and storage of structured data in time, aggregated and personalized view of collected information. The functionality of this service will be demonstrated in UPB for the results of research activities (projects, papers, books, monographs, patents, etc.). It will make such a researcher's profile, automatic collection of research results-level advanced search fields, emphasizing collaboration offering quick access to information.

Communication service for heterogeneous networks aims to offer two types of sub-services for data transfer: data transfer multipoint-to-multipoint and multicast data transfer. These two types of sub-communication services are required for transfers of audio / video streams (real time or on demand) and the transfer of data generated by the interactions between users in a real environment (video-conferencing) or virtual.

[1] The authors of this paper want to manifest their acknowledgements for Prof.dr.ing. Nicolae Tapus, Prof.dr.ing. Valentin Cristea and Prof.dr.ing. Florica Moldoveanu for all their support and coordination of research activities in the eUPB project.

Contextualization mobile service aims to provide a set of features that include automatic collection of contextual information from mobile devices and sensors, customized information retrieval based on contextual information, aggregating and data storage, providing an aggregated view of information on mobile devices. Service functions will be demonstrated at the UPB information applications, location and recommendation using mobile devices.

Monitoring and control service for environmental parameters follows the design and implementation of a heterogeneous wireless sensor network. The scope is to automatically locate people and objects inside buildings, environmental monitoring and control of parameters. Also, the service will allow predictions based on data collected on the network environment it is placed. Scientific challenge of the project is to develop optimized and secure communication protocols for heterogeneous WSN architecture, including specialized types of sensors for providing location services, monitoring and quality control to the specified level.

3DUPB Service - the 3D MMO virtual replica of UPB will foster mutual understanding and cooperation at university level, by having a unified collaborative virtual space. It will also promote and improve the image of UPB, the permanent presence online as a multiuser 3D virtual reality and collaborative. The service will provide an open platform that allows integration, visualization and 3D interaction for a wide range of applications practical purposes (the UPB services).

Analysis and content extraction of scanned documents service will enable the definition of independent modules for processing and analysis of images (digitization, segmentation, detection and compensation slope, layout analysis), interfacing between them (connectivity graph, testing multiple approaches for the same function, using voting machines between similar approaches), publication of results where the human operator intervention is required, import and export data. To apply the results obtained will be generated as PDF files, multi-layer hierarchical structure of the document containing the image postprocessor in the foreground, text and meta-information in the background. PDF files will be optimized for remote access, will contain significant text for indexing in search engines will keep the original appearance of documents and meta-information will allow recovery.

Collaboration service will be used for collaboration and creating virtual communities that collaborate to achieve a common goal. As in real life, there are certain rules (laws) to be observed in such collaborations. The main purpose of this work is finding a way to translate these informal rules in a formal model, in which we can say with certainty that all agencies involved in the community will know the laws for community interactions. The system will provide no way to violate these rules.

The rest of the paper is structured as follow: Section 2 presents related work in the field of e-Services in the context of eUPB system, Section 3 presents the general architecture of eUPB platform, and then in Section 4 each competent are described in details. The paper ends with conclusions and future developments.

2. RELATED WORK

Most European countries collect and store their research information in national repositories. This research information is often spread over several regional or local systems with heterogeneous structure. In order to get additional value out of heterogeneous information sources, their inherent structures have to be mapped into a specified format that allows for a quality integration of the individual sources within a target system (Ivanovic et al, 2011). The supports for these operations are electronic services.

The concept of e-Service is the short form for electronic service and represents one prominent application of utilizing the use of information and communication technologies in different areas. The role of technology in facilitating the delivery of services makes them very important for electronic services. (Rowley, 2009) defines e-Services as: "…deeds, efforts or performances whose delivery is mediated by information technology. Such e-Service includes the service element of e-tailing, customer support, and service delivery"

Regarding data retrieval and aggregation, CRIS (Current Research Information Systems) cover the research activity of an organization. CERIF (Common-European Research Information Format) is an EU recommendation to member states for CRIS. CERIF allows interoperability across CRIS. CERIF provides metadata describing publications with formal syntax and declared semantics. The CRIS provides the research context for the publication and links to associated research datasets and software (Keith et al, 2008).

Techniques for interconnecting peers in order to provide efficient multicast communication services were described in (Andreica et al., 2010a) and (Andreica et al., 2010b). Similar communication services (e.g. publish-subscribe, durable storage of message) are provided by messaging providers implementing the JMS API (Richards et al., 2009).

For context-aware service, MobiPADS (Chan and Chuang, 2003) is a middleware system for mobile environments. Its main entities are the *Mobilets*, which are entities that provide a service. They can be migrated between different MobiPADS environments to provide a specific service. XMIDDLE (Zachariadis, *et al*, 2002) is a middleware that focuses on the need of applications for mobile devices to form ad-hoc peer-to-peer networks. The Context Toolkit (Salber, *et al*, 1999) offers facilities for context-aware applications: encapsulation of sensors, access to context data through a network API, abstraction, storing and sharing of context data, and the access control. Although many context-aware systems have been developed in the past few years, their functionality remains primitive because of problems such as (Chen, 2003): Lack of reusable context-aware mechanisms, Limited resources on devices, and/or Privacy issues in accessing user. Unlike the set of middleware services for mobile context-aware applications, these middlewares provide limited methods of adapting to changes in the context are too dependent on specific hardware components or are too centered on a specific functionality.

For services platform, sensor networks and types of sensors are presented in (Akyildiz, *et al*, 2002) and (Mainwaring, et al, 2002). Techniques for routing and dissemination protocols are described in (Al-Karaki and Kamal, 2004) and in (Akkaya and Younis, 2005). Security protocols and mechanisms are specified in (T. Kavitha and D. Sridharan, 2010) and in (Rughinis and Gheorghe, 2010).

The most well-known similar virtual space is Second Life, which actually already has a lot of education related functionality. There are several universities which already have established themselves a presence in the Second Life realm. The functionality of these virtual representations ranges from mere graphical representations to some rich in interactions and collaboration which explore the boundaries of online education. However Second Life is a proprietary system, having costs and limitations: it does not allow full control over the virtual space or good extensibility and integration with a real space.

Also we do not believe that the future of such spaces should and would be inside any proprietary system (Ariane's Life in the Metaverse, accessed on 15.02.2011) (OpenSim, accessed on 15.02.2011) (Multiverse Platform Architecture, accessed on 15.02.2011).

We consider that the 3D MMO web will evolve quite similar to the current, 2D web, as a decentralized network of distinct but interconnected virtual worlds.

Therefore we would develop 3DUPB in such a way that it should be totally customizable to our needs and open in what it regards connectivity and standards, so it can naturally evolve and connect to the 3D MMO web of the future.

The most well-known content conversion project is Google Books (Google, 2011). It allows searching the full text of books that Google has scanned, converted to text using optical character recognition, and stored in its digital database. Google Books currently converts books from public domain or copyrighted material from its Library Project participants around the world into digital text at a rate of around 1000 pages per hour (New York Times, 2006); offering unprecedented access to what may become the largest online corpus of human knowledge.

Some digitization initiatives refuse to take part in Google’s project due to copyright issues (New York Times, 2006) and turn to private content conversion solutions. The current business model for ACCSs consists of on-demand services offered to the client wishing to digitalize paper documents.

Such an example is DocWorks which is a fully-grown commercial product developed by Content Conversion Specialists GmbH. It offers solutions for all the previously discussed stages of (pre)processing but requires a high degree of operating knowledge and is generally used in large document repositories by specialized personnel focusing on speed of conversion. Being a commercial product (like many others) it offers no access to inside processing and thus any scientifically-valuable information remains hidden inside the software.

OCRopus is an open source document analysis and OCR system. It allows a certain degree of modularity and it was designed by some research-oriented minds. It lacks somehow the perfect interface of a commercial product but it is a fully usable one (OCRopus, 2011).

There is a wide body of research in the area of maintaining the consistency of a distributed system. Depending on the guarantees that the system needs to provide with regard to the consistency, we can distinguish two big approaches: pessimistic algorithms, usually providing strong consistency models (such as one-copy serializability) and optimistic algorithms, which are tailored for applications that can tolerate relaxed consistency.

Regarding all these remarks, the eUPB systems will enhance the existing results and experience for e-Services, creating a flexible, integrated and scalable platform consisting in multiple electronic services.

3. eUPB SYSTEM ARCHITECTURE

The integration of proposed services for eUPB platform is presented in Figure 1. All services are represented in integration with others and the auxiliary components useful for platform are presented.

It is necessary to go beyond the quality dimensions of e-Services and also take into account the inherent characteristics of e-Service delivery and the factors that differentiate one service experience from another. The dimensions that were discovered are: acceptance, accessibility, digital divide, benchmarking, e-readiness, efficiency, stakeholders, security, and usability.

The main advantages of eUPB services are: reducing public expenditure, combating bureaucracy within public institutions, increasing the transparency of the use and management, reduction and streamlining of direct contact, promoting the use of Internet and technology leadership in public environment. Public services come also with their disadvantages: inequality of opportunity for certain social groups; security and privacy risks, covering the road from sender to receiver, slow feedback. The eUPB platform tries to minimize all of these risks using research experience form distributed environments.

The Internet is an effective marketing and information channel that simplifies business operations and brings competitive advantage. e-Services are comfortable, fast and secure, and enable businesses and government institutions to reduce costs, improve their reputation and raise customer satisfaction. All eUPB services will be available over the Internet as an integrated platform.

The main paradigm used for eUPB systems is Service oriented architecture (SOA). Without SOA, it is hard to understand the code and to efficiently maintain the shared functionalities because of the repeats or variations. Those common functionalities are abstracted out as "services" and implemented as separate code units. These services have to be deployed into an SOA system and the SOA makes them available to the codes to the service consumers. The services are accessed via service interface.

If the services are implemented as Web Services, the services do not reside on the same system as the service users. The services are deployed on Web Service containers and are accessed across the networks via Web Services interface. If the user interface is exposed directly by the Services to the users, there is an additional problem with overall control and the QoS issues.

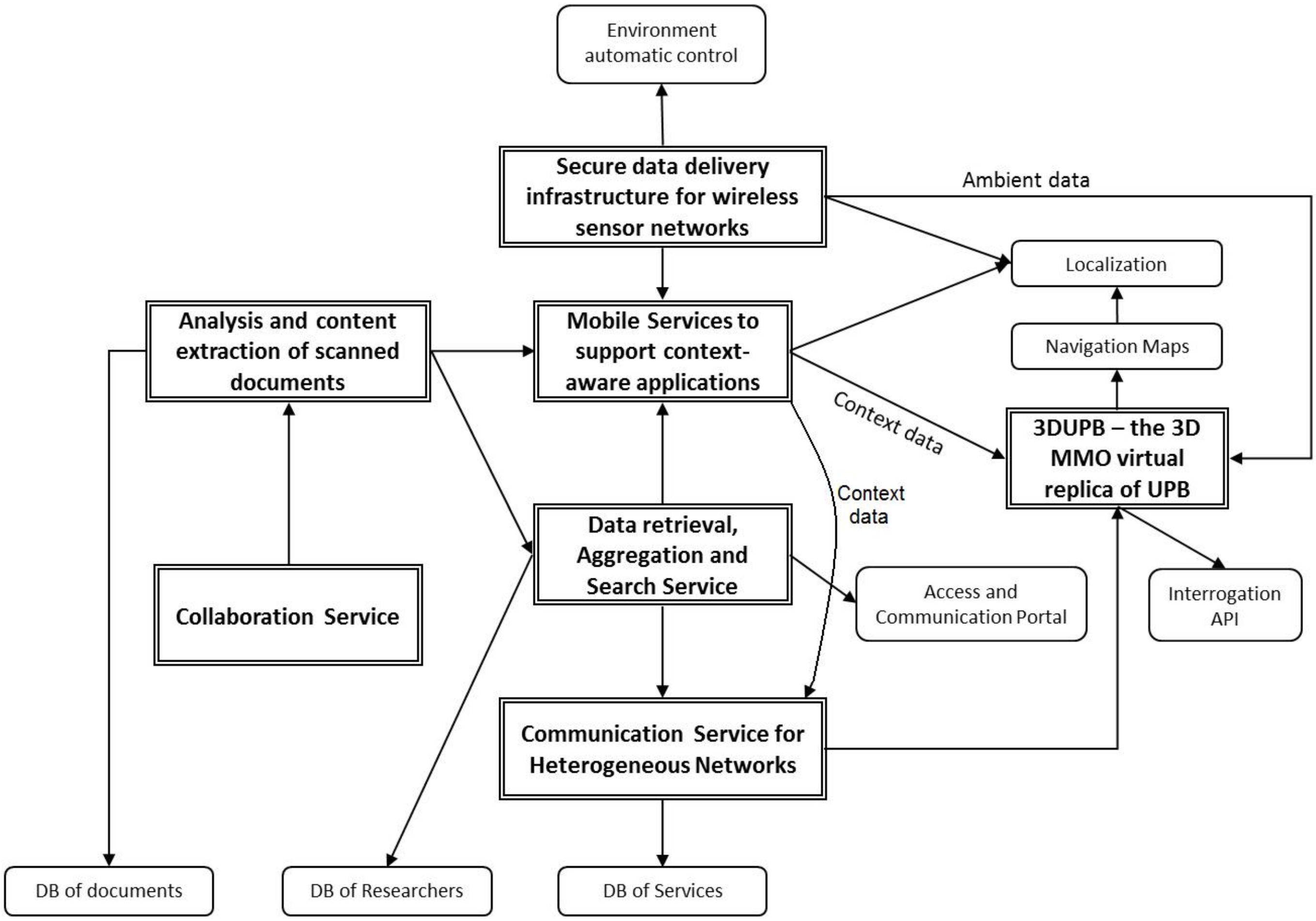

Figure 1. eUPB architecture

5. eUPB SERVICES AND FUNCTIONALITY

This section presents each component of eUPB platform highlighting the main functionality and integration with other components and services. It will also be provided the requirements and facility for some components. In the current development of eUPB platform, some services ware devised and the implementation stats, and for some other services the current status is the design and research.

5.1. DATA RETRIEVAL, AGGREGATION AND SEARCH SERVICE

The functionality of this service will be demonstrated in UPB for the results of research activities (projects, papers, books, monographs, patents, etc.). It will make such a researcher's profile, automatic collection of research results-level advanced search fields, emphasizing collaboration offering quick access to information.

There are numerous academic digital libraries and bibliographic databases online: ACM Digital Library, IEEE Xplore, CiteSeerx, etc. Navigating through these large collections is difficult and time consuming. For this service we propose to develop a system that will automatically find articles similar to a research topic, having specific authors, given its title and abstract. The system will return a list of articles or projects, ranked by their similarity to the research topic. We will focus on finding papers similar to the research projects proposed by the Distributed Systems Laboratory at UPB but the resulting system should be able to retrieve relevant results for any research topic in computer science. The results come from an index containing either research or diploma projects proposed by our University and the DBLP Computer Science Bibliography. We experiment with different settings for the system and evaluate the results and also try to propose a solution for the cross-language retrieval problem (most of the projects proposed by our University are defined in Romanian while DBLP is in English).

The service will follow different entities and activities (see Figure 2). The classification of all retrieved and aggregated data will be guided by semantic metadata.

Figure 2. Components for data retrieval, aggregation and search service

We will consider the following digital library:

- *ISI Thomson*: http://isiknowledge.com
- *ACM Digital Library*: http://portal.acm.org/
- *DBLP*: http://www.informatik.uni-trier.de/~ley/db/
- *ScienceDirect*: http://www.sciencedirect.com/
- *IEEE Xplore*: http://ieeexplore.ieee.org/
- *CiteSeer*: http://citeseerx.ist.psu.edu/
- *PubZone*: http://www.pubzone.org/
- *AcademicINFO*: http://www.academicinfo.net/
- *NSDL Library*: http://nsdl.org/collection/computer-science/

The standards uses for service development are: CERIF-2008, BibTex (http://en.wikipedia.org/wiki/BibTeX), APA Style: (http://owl.english.purdue.edu/owl/resource/560/01), OASIS (http://www.openoasis.org), Digital Object Identifier (DOI): http://dx.doi.org. The existing similar instruments that will be analyzed and considered is GogoleScholar and Arnetminer (http://arnetminer.org).

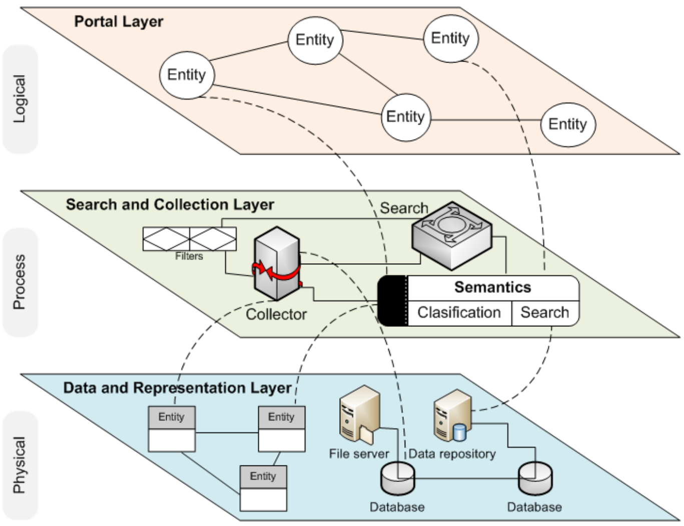

Figure 3. Architecture data retrieval, aggregation and search service

The architecture of service is presented in Figure 3. The service considers three different layers for modeling: physical layer (for data storage and interaction), process layer (for information retrieval and dynamic processing) and logical layer (for entity interaction model). There are three classical models in Information Retrieval Systems: Boolean, vector model and probabilistic model. We will consider all models to offer different facilities. In the implementation of the information retrieval component of our system we relied on the open source search platform Apache Solr.

5.2. COMMUNICATION SERVICE FOR HETEROGENEOUS NETWORKS

The communication service for heterogeneous networks will be implemented as a flexible and extendible communication framework which will provide two types of data transfer sub-services: *multipoint-to-multipoint* and *multicast*.

The service makes use of the concept of *communication channel*. The messages sent on a communication channel are seen as being sorted in time (either according to their sending time or according to an explicitly set time stamp). The users of the service may create new communication channels and send and receive messages on them. Besides the actual data delivery, the service will also store the sent messages; according to a configurable policy (e.g. how long a message should be stored). The API provided by the communication service contains the following functions:

```
createCommunicationChannel(attributes,
metadata)
```

- *attributes*=attributes of the communication channel (e.g. who can access the channel for reading/writing, the priority of the messages sent on the channel etc.)
- *metadata*=information about the communication channel (e.g. description) the function returns the identifier of the newly created communication channel.

```
getCommChannelList(attributeFilter,
maxResults)
```

- the function returns a list of tuples *(commChannelID, attributes, metadata)* of all the communication channels (but not more than *maxResults* of them) whose attributes match the attribute filter *attributeFilter*.

```
sendMessage(commChannelID, message,
attributes)
```

- the function sends the message *message* (a sequence of bytes) on the channel *commChannelID*
- *attributes*=attributes associated to the message (one of the attributes may be a time stamp used for ordering the messages on the channel)

```
getMessages(ccID, attribFilter,
maxResults)
```

- the function returns a list of pairs *(message, attributes)* of all the messages (and their attributes) sent on the channel *ccID* whose attributes match the filter *attribFilter* (but not more than *maxResults* messages)

```
subscribeForMessages(ccID, attribFilter,
maxResults)
```

- the user subscribes to the service in order to automatically receive all the messages sent on the channel *ccID* whose attributes match the filter *attribFilter*
- the function returns an identifier *subscriptionID*
- when such messages arrive, a callback function *receivedMessages(subscriptionID, msgList)* is

called: *msgList* is a list with at most *maxResults* pairs *(message, atributes)* of the messages and their attributes which match the restrictions of the subscription *subscriptionID*

`cancelSubscription(subscriptionID)`

- cancels the subscription *subscriptionID*

Most of the service's API functions may be accessed through standard technologies (e.g. web services), except for the functions *subscribeForMessages* and *cancelSubscription*. In order to access these two functions, a dedicated client implemented in the same programming language as the service (Java) must be used (or any Java program may be used, as long as an instance of a special type of Java object is created and used).

When a user wants to access the service, it must find out the network address of one of the active peers. This is the service discovery problem, which we currently solve through a centralized (though replicated) lookup server.

The service will be run on a set of dedicated machines (the core peers). However, any user with external IP and enough resources (CPU, RAM, storage space, etc.) and credentials may instantiate its own peer(s), if it so wishes. The API also contains a function for the dynamic creation of service instances (i.e. of peers). The newly started instance will connect and communicate to the other existing instances (peers) and may be used by the clients. The peers interconnect in an overlay, based on local decisions. Some of the information used for constructing the overlay consists of context information (e.g. place-related context or user-related context) provided by other services of eUPB.

The communication service will also consider the QoS requirements of the communication channels (specified, for instance, through their priority). Currently, the way QoS can be ensured is through the properties of the peer-to-peer overlay of the service instances and by redirecting clients to appropriate service instances to interact with. We are also considering using QoS in order to provide incentives to users to instantiate service instances (e.g. you get better QoS if you contribute your resources to the system by instantiating a peer).

5.3. MOBILE SERVICES TO SUPPORT CONTEXT-AWARE APPLICATIONS

The mobile services for contextualization provide a set of features that include automatic collection of contextual information from modern mobile devices, customized information retrieval based on contextual information, aggregating and storing contextual data, and providing an aggregated view of contextual information to third-party application. These services will be demonstrated through applications for orientation, information and recommendation, using mainly the capabilities of modern mobile devices. They provide a unitary API for retrieving and requesting contextual data, as well as developing actions to be triggered when specific contextual conditions are met, directly on the user's mobile phone.

Figure 4 presents the main layers of the proposed architecture. The first layer includes modules responsible with the monitoring of context data. On this layer a mobile phone runs a service that is able to dynamically instantiate monitoring modules. These modules are able to collect context data, and can be loaded or unloaded dynamically by the user. In addition, their management is automated through a remote repository.

The context is composed of *location* (out-door we use GPS sensors, for in-door location we combine information received from several sensors, such as GSM cells, WiFi access points, etc), user's *identity* (if available, provided using certificates installed on the mobile device), *time*, and current *capabilities* of the mobile phone (battery level, light intensity, accelerometer, etc.). If the user's identity is found, it is augmented with additional information, such as the user's *profile* (it obtained from an external service, it refers to its research profile, or/and academic profile, such as user part of a course class), and user's *activity* (obtained from the internal agenda for example).

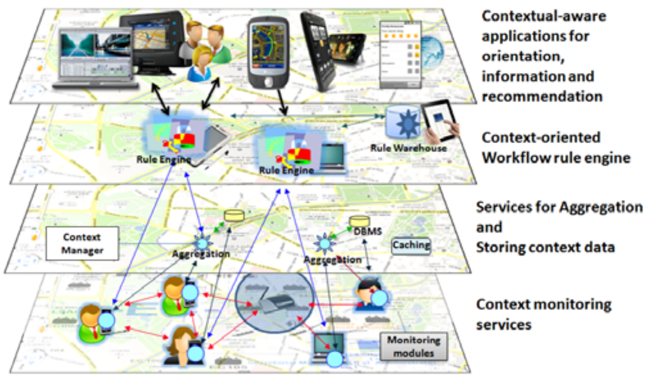

Figure 4. The proposed architecture

The next layer deals with the aggregation and long-term storing of context data. Information from context sources is gathered by the context manager and organized based on concepts from a predefined model. This actually represents an abstraction layer (middleware) which can be used by all sorts of applications to access context information. The domain described by the model will act as a contract between the middleware system and consumer applications.

The information and services offered by the contextualization services will be consumed by two sorts of applications. Autonomous applications will use the services directly to access context information. They will control entirely the way they will react to context changes. In addition, we define a third layer, which uses context Rule actions. They are started when specific conditions are evaluated by the Rule Engine. Changes in the context may trigger different actions on the mobile phone according to a predefined rule set. The rules are expressed in an XML-based format and are stored in a remote repository. The user is therefore able to dynamically load and execute on the local mobile phone specific rules, depending on his/her own preferences.

An example of such rule can, for example, detect that it is lunch time by consulting the user's internal agenda from the mobile device. It then determines the user's current location and can use Internet services to determine restaurants nearby. A notification is then brought up. If the user is interested he can access more details about the suggested nearby restaurants. In another situation, the application observes that the user is in a free time interval according to his/her agenda and place (location), and also that the weather is sunny (using weather Internet services). According to the user's settings it can suggest parks nearby, or other similar outdoor activities close to the user's current location.

Finally, the fourth layer is responsible with the applications, expressed as rules and actions, which can be used for orientation, information and recommendation purposes. At this layer there are local utilities that can help with context-triggered actions. There are also the applications that use the context data to improve response to stimulus (an interior or exterior request). An application can react to changes in the current context and take specific actions depending on some predefined rules. For this, conditions are evaluated period as the data is retrieved.

Third party applications and services, such as the ones presented in this paper, can use the API provided by the context-aware services. They can use functions for obtaining particular context data, using filters, or can subscribe for context data. They can also declare new execution rules for users to install on their mobile devices.

5.4. SECURE DATA DELIVERY INFRASTRUCTURE FOR WIRELESS SENSOR NETWORKS

Wireless Sensor Networks are composed of hundreds, even thousands of small and cheap devices, that auto-organize themselves into a network with the purpose of monitoring and event detection in the environment in which they are placed.

The *Secure Data Delivery Infrastructure* will provide complete services such as: sensing, routing and security services. These services are organized into an extensible framework that is able to accommodate new services very easy.

The *Sensing Service* enables the collection of environmental parameters. Each sensor device has a limited set of sensors; therefore sensor nodes are heterogeneous from the sensing point of view.

The network will contain sensor boards with a large number of *sensors*: temperature, humidity, luminosity, pressure, presence, oxygen, carbon dioxide, liquid presence, liquid level, current, and ultrasound. These sensors will be used to collect data that will be used for monitoring the environment.

Monitoring the environment can be used for providing an appropriate level of comfort for the humans that are working of studying in the environment, such as the appropriate room temperature and luminosity, or for detecting critical events, such as flooding, fire, lack of oxygen.

Through the *SenseSelection service*, the base stations are able to request data from certain sensors only. The base stations are remotely managed by the administrator and can be commanded to send *SenseSelection* packets that request only certain data readings from the network. Each sensor type has an associated identifier and a selection packet contains a number of sensor identifiers. These packets must reach every sensor device in the network. This is performed by the routing service.

The *SenseSelection layer* on the sensor nodes will perform proper data collection only from the specified sensors and delivers it to the upper layer that puts information into data packets and sends it towards the base station.

The *Routing Service* is responsible with routing packets from source to destination. The traffic pattern in sensor networks has only two types: one-to-one when a sensor node is sending the collected data to one of the base stations and one-to-many when the base station sends some information to some or all sensor nodes.

We consider a multi-hop architecture for the network, meaning that the base station is accessible through one or multiple intermediary nodes called hops. The *Routing protocol* is able to organize the network in a hierarchical logical topology. Sensor nodes will organize themselves in clusters of nodes, each containing a cluster head. Each cluster member will communicate with the base station through the associated cluster head.

The *Routing layer* is placed between the medium access layer and the security layer in the communication stack. Each sensor node stores its cluster head identifier in the Routing layer and will use this node as next hop when routing data towards the base station. The cluster heads are chosen by the base station using a linear or genetic algorithm and taking into consideration the energy level of each node and its relative location from the base station.

For the one-to-many communication, the Routing layer will have implemented an *energy-efficient gossip protocol*. Gossiping is an enhanced method of flooding in which the node that receives a packet, will further send that packet to a set of selected neighbors that have a low probability to have already received that message. Gossiping avoid the problem of implosion by using unicast messages instead of broadcast messages.

The *Security Service* provides a set of lightweight security protocols that are able to meet all requirements necessary for sensor networks. Each sensor network application requires a different level of security. This level of security can be assured by specifying the security requirements and enabling/disabling specific security protocols and mechanisms.

Authentication and Anti-replay Security Protocol (AASP) is a lightweight and energy-efficient security protocol that provides authentication, anti-replay, integrity and reliability. It establishes an authentication connection between source and destination before sending any data packets. Anti-replay and anti-injection protection is assured by computing a Message Authentication Code (MAC) using the payload of the previous packet and the sequence number of the current packet, thus providing a strict context to that specific packet. Integrity is assured by using the current payload when computing the MAC and reliability is assured using acknowledgements.

Storm Control Mechanism (SCM) assures protection against flooding and Denial of Service attacks. Flooding attacks are used by malicious nodes with the purpose to consume resources on authorized sensor nodes, such as energy, bandwidth, processing power and memory. SCM keeps track of the global and per conversation packet rate and detects high rates of packets that are similar to flooding attacks. Upon detection, it is able to block flooding streams of packets and permit legitimate ones.

Hop-by-hop Inspection Mechanism (HIM) provides protection against energy-targeted attacks, in which an attacker sends incorrect and large packets from a long distance in hops from the base station and consumes large amounts of energy on the intermediary nodes. HIM stops such an attack by inspecting packets on the intermediary nodes and rejecting invalid packets.

Lightweight Encryption Module (LEM) enables energy-efficient encryption for data packets exchanged by resource-constrained devices. The encryption method is Advanced Encryption Standard (AES) and it is used with a 128 bits secret key. This module assures data confidentiality.

All these modules are integrated together in the Security Service. The base stations can be remotely configured to send *SecuritySelection packets* that are used to configure the sensor devices with the right level of security. For example, the base station disseminates a message in which it specifies only AASP and HIM modules. Therefore, sensor nodes will enable only these modules from the Security Service.

The specified set of services consisting in the Sensing, Routing and Security services, are able to assure a reliable and secure data delivery towards centralized locations, where data can be further processed. The set of services can be easily extended in order to support new functionalities, such as localization or QoS services.

5.5. 3DUPB - THE 3D MMO VIRTUAL REPLICA OF UNIVERSITY "POLITEHNICA" OF BUCHAREST

3D MMO (massively multiplayer/multiuser online) virtual spaces are part of the culture of the young generation. Impressive figures from probably the most "digitally" advanced country - South Korea - show 90% of households having broadband internet and 40% of South Koreans using a digital representation of themselves (so called avatar in Western digital culture) (Ahonen and O'Reilly, 2007). Figures are lower in other countries, but they are catching up rapidly on an ever ascending trend.

Chat services, social networks and the most advanced form on online interaction - 3D MMO virtual spaces - are making the turn from just entertainment to an important part of human society, affecting business, science and education (Castranova, 2006).

We conceived the presentation layer of eUPB based on emergent ideas from virtual reality and online worlds, mainly because this familiarity the new generation has with them but also for their sheer power of representation and interaction (Aldrich, 2009).

The 3D representation and interaction service (acronym 3DUPB) has several important objectives:

- Foster socialization and collaboration throughout all the university: as a single virtual space, will break the current barriers between the faculties or our university.
- Create transparency and strong online presence, as the system will be accessible also from outside the university, by any online visitor.
- Offer a powerful presentation level (integration, visualization and 3D MMO interaction) for a large variety of services and applications (some from the eUPB system, while others will be developed later).
- Implement, test and tune some new concepts in online and interactive education and collaboration; based on this experiment, the platform would be extended to allow interconnectivity of several universities.

3DUPB will be a dynamic and extensible virtual replica of University "Politehnica" of Bucharest. From the functionality point of view, the key aspects to note are:

- Accurate representation of the real space of the university, based on high-quality calibrated 3D models and textures.
- The representation of each user through a customizable avatar.
- The use of massively multiuser online technologies allows simultaneous access and complex interactions (travel, actions, gestures, voice and text communication) for thousands of users, inside the same virtual space.
- Users will be: students, professors, administrators, quests and. intelligent agents (bots).
- A plethora of functionality to foster socialization: user profiles creation and discovery, common interests groups, games, online events, etc.
- Virtual laboratories.
- Dynamic updating of the virtual space based on the content of the real space: basically all aspects, starting from weather condition to location of the users are supposed to be updated automatically.
- Large scale use of video broadcasting from the real space to the virtual space.
- Extensibility in terms of data content: users can define their personal 3D virtual spaces in the UPB, to present their interests, host private events for friends, etc.
- Extensibility in terms of functionality: 3DUPB will offer an API for the creation of new modules and applications. One basic idea is be reflectivity: the possibility for an application to dynamically inspect the content of the virtual space surrounding a user avatar. Also the new applications will be given, under strict security rights, the possibility to alter the virtual space according to their needs. Advances functions would include automatic avatar control or aspect, hidden communication channels for intelligent agents, etc.

For the implementation of 3DUPB, main technical challenges are related to the scalability of the MMO server.

Unlike other application types, where the work flow is initialized and directed by a single user, virtual spaces support complex interactions, involving many users at the same time. The actions of an individual user and their effects, calculated by the server, must be propagated in real time to all those affected by them. For example, even a simple movement of a user avatar should be broadcasted immediately to the other users from his vicinity.

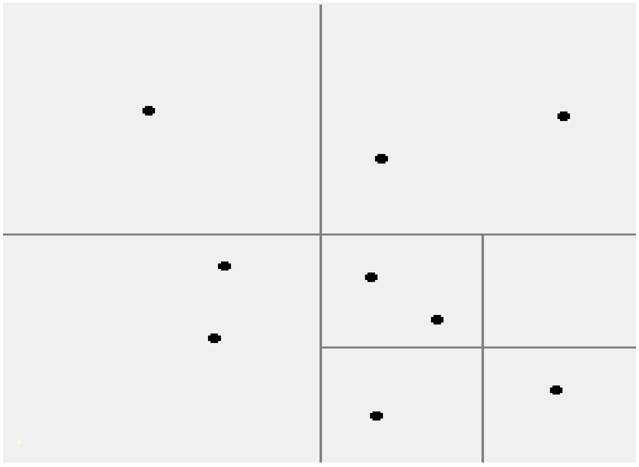

Figure 5. Quad-tree spatial decomposition

All the complex interactions from such virtual spaces translate into huge computational demand, growing with the number of users. The challenge is to perform all these operations in real-time (Waldo, 2008): any delay of more than hundreds of seconds between a user action and the visibility of its effects in the virtual space creates "lag" - a phenomenon which dramatically reduces user’s immersion and participation.

There is yet no perfectly satisfactory solution to solve these contradictory aspects: the need of scalability and the need of real time interactions.

We handle this challenge with the idea of spatial decomposition, which aims to reduce and also divide the computational effort by assigning users to various spatial regions, each of them handled by distinct computational elements (computers or processors).

So, the decomposition allows:

- reducing total workload, as each users actions are performed only on users in its region (or neighboring, if user is on close to border)
- division of workload, each region being assigned to different processing unit

There are many ways to perform decomposition: static or dynamic, handmade or automatic, etc. Figure 5 shows using a quad tree for decomposition, with a threshold set to 2 users / region, just for example.

Of course, decomposition itself introduces architectural and computation threshold. The processing units must keep track of users' movements and share border data and many others (Moldoveanu, 2008).

Figure 6 shows the high level architecture for 3DUPB server:

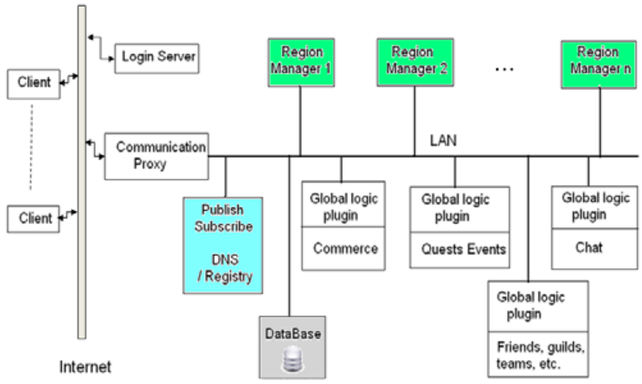

Figure 6. System Overview

Besides the technical challenges regarding the software architecture, we consider that the biggest challenge is the overall design and implementation (Craig et al., 2009) of this mixed reality virtual environment of an unprecedented scale. It might be that our university would be the first in the world with such a complex and original virtual campus.

And the plans go far beyond. Once the concepts and implementation will be tuned, we plan to enlarge the system to link any number of virtual replicas of universities and other types of institutions in a single, interconnected network - which would open the field for new infrastructures and paradigms in national and international education.

5.6. ANALYSIS AND CONTENT EXTRACTION OF SCANNED DOCUMENTS

Paper documents such as newspapers, books and other prints suffer in time of various forms of autonomous decay that can affect paper, among which are paper acidification, and ink and copper corrosion. In order to overcome these drawbacks of printed documents, a direction pursued in recent years is converting scanned documents into electronic files, especially in the case of large electronic libraries, for easier access to documents.

An Automatic Content Conversion System (ACCS), based on optical character recognition (OCR), enables operations such as editing, word searching, easy document storing and multiplication, and the application of a large set of text techniques including text-to-speech and text mining to be performed on the digitalized document.

In addition, this ensures both a better preservation of original documents, due to minimizing the need for physical use and makes it suitable for automatic data processing or usage under a large spectrum of devices including mobile phones or other mobile devices.

ACCSs normally consist of the processing stages presented in Figure 7. The stages are arranged in a pipeline and convert the document image acquired from optical scanners into digital text which can be stored using machine readable file formats such as Portable Document Format (PDF). Note that actual OCR is only a small step in the process, as most of the work is done on image preprocessing to ensure optimum results after OCR (Man, 2002) (Niblack, 1986).

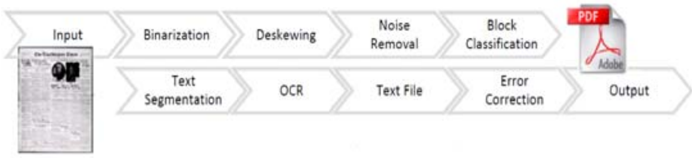

Figure 7. Content Conversion Pipeline

By using these techniques alongside with various storage data reduction methods some very large knowledge databases may be constructed, containing study manuals which may be used online by various content indexing, searching and presentation applications available on a practically unlimited number of devices.

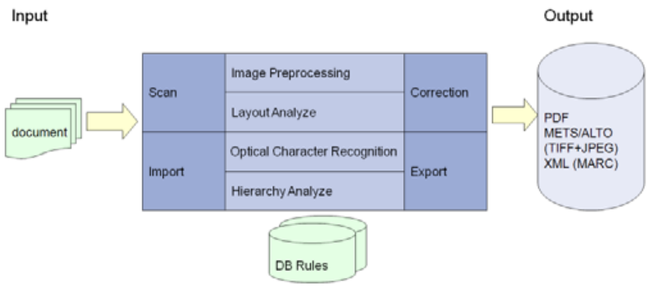

Figure 8. System Overview

The service of analysis and context extraction from scanned documents will allow:

- The construction of independent modules of analyze and image processing (image digitization, image segmentation, skew detection and correction, layout analyze)
- Interfacing the various modules to generate a full processing pipeline (edit interconnectivity graph, run multiple approaches for the same recognition problem and use voting to validate and improve the results)
- Editing of results for the various intermediate steps in layout and hierarchy detection, OCR; thus allowing human operators to verify and correct (if necessary) the output of different heuristically-based algorithms
- Import and export of document data

In order to effectively use the obtained results the files will be generated in PFD format, structured in a multi-layer manner: the original image stays in the front layer (to ensure that the original of the document is preserved and seen by the user) and the meta-information in a background layer (to ensure the usability of document in search, classification and text-retrieval operations).

Figure 8 presents the main stages of the processing, consisting in:

- **Input preparation** - scan/import of documents
- **Image Preprocessing** - skew detection and compensation, digitization and/or image segmentation (Srinivasan and Shobha, 2008)
- **Layout Analyze** - clustering of the image segments resulted in previous steps, according to the logical reading order of the document (Boiangiu, 2008)
- **Page Numbering** - markers detection and renumbering, if necessary
- **Hierarchy Analyze** - layout blocks are classified according to their importance and the logical table of content for the document is constructed
- **Optical Text Recognition** - text areas are processed to extract the valid words/sentences and their formatting
- **Output/Export of electronic document** in various formats

Furthermore, the proposed service wants to combine the best of research and industry worlds: assuring both the possibility to add components and user interfaces, specific to a commercial service which also considers usability and a research oriented open source product in a fully functional service.

5.7. COLLABORATION SERVICE

Law-Governed Interaction (LGI) is a message exchange mechanism that allows an open and heterogeneous group of distributed actors to engage in a mode of interaction governed by an explicitly specified and strictly enforced policy, called the “law” of this group. By “actor” we mean an arbitrary process, whose structure and behavior is left unspecified. An actor engaged in an LGI-regulated interaction, under a law L, is called an L-agent (or simply an “agent,” when the identity of the law does not matter); the messages exchanged under a given law L are called L-messages; and the group of agents interacting via L-messages is called an L-community.

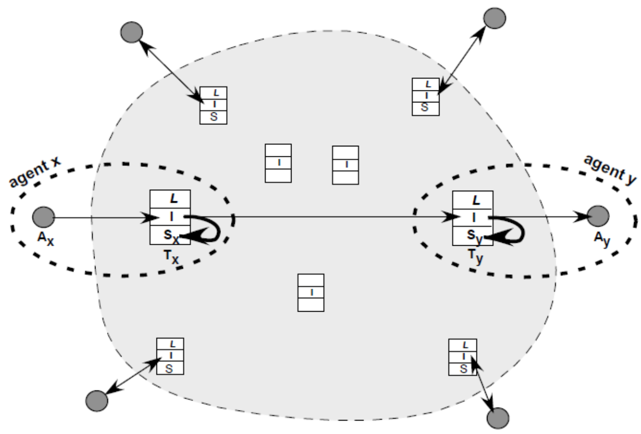

Figure 9. Interaction in LGI: Actors are depicted by circles, interacting across the Internet (lightly shaded cloud) via their private controllers (boxes) operating under law L.

LGI turns a set of disparate actors, which may not know or trust each other, into a community of agents that can rely on each other to comply with the given law L. This is done via a distributed collection of generic components called private controllers, one per L-agent, which need to be trusted to mediate all interactions between these agents, subject to a specified law L (as illustrated in Figure 9).

The Concept of Law Under LGI: LGI laws are formulated in terms of three elements, called: regulated events, control-state, and primitive operations-which are defined in the context of each agent operating under LGI. Only an abstract description of these elements is provided here.

Regulated Events (or, simply, events) constitute the domain of LGI laws. They are the local events that may occur at an individual agent (called the home of the event at hand), whose disposition is governed by the law under which this agent operates. All regulated events are related to inter-agent interactions. They include arrived events, which represent the arrival at the home agent of a message from the outside; and sent events, which represent the attempt by the home agent to send a message. There are additional regulated events whose relevance to interaction is less direct. One of them is the adopted event, which represents the birth of an LGI agent-more specifically; this event represents the point in time when an actor adopts a given law L to operate under, thus becoming an L-agent.

Control-State (or, simply, state) of a given LGI agent represents a function of the history of its interaction with other LGI agents. This function, mapping history of interaction to a state, is defined by a specific law For example, if the number of messages already sent by an agent is somehow relevant to the law under which it operates, then this law would have to mandate maintaining this number as part of its state. That is, the semantics of the control-state is not universal, but is defined by a specific law.

Primitive Operations (or, simply, operations) are the actions that can be mandated by a law, to be carried out in response to the occurrence of a given regulated event. These operations can be classified into two groups. First, there are communication-operations that affect message exchange between the home-actor and others. These include the forward operation that forwards a message to another agent, and the deliver operation that allows the home-actor to actually receive a message that arrived on its behalf. Second, there are the state operations that affect the state of the home-agent. These, and other operations to be introduced later, are called “primitive” because they are meant to be carried out if and only if they are mandated by the law.

5. CONCLUSIONS

The eUPB project presented in this papers is structured along several dimensions that follow the development of complementary services, integrated to support everyday work experience, research and learning in the University POLITEHNICA of Bucharest (UPB).

The paper highlights the support for all eUPB components considering the following aspects: service oriented, users centered, real-world integration over Internet, production and use of multimedia content over Internet.

The main services identified in eUPB system are: (i) data retrieval, aggregation and search service, (ii) communication service for heterogeneous networks, (iii) mobile services to support context-aware applications, (iv) secure data delivery infrastructure for wireless sensor networks, (v) 3DUPB - the 3D MMO virtual replica of UPB, (vi) analysis and content extraction of scanned documents, and (vii) collaboration service.

The paper si is a position one presenting the general architecture of eUPB. The future work it will be represents by implementation and testing all the presented components for eUPB. The QoS, interoperability and integration in a unified platform will be the keys concept for future development.

ACKNOWLEDGMENTS

The work presented in this paper was funded by the Sectorial Operational Programme Human Resources Development 2007-2013 of the Romanian Ministry of Labour, Family and Social Protection through the financial agreement POSDRU/89/1.5/S/62557.

REFERENCES

Ahonen T. and O'Reilly J. (2007), *Digital Korea. Convergence of Broadband Internet, 3G Cell Phones, Multiplayer Gaming, Digital TV, Virtual Reality, Electronic Cash, Telematics, Robotics, E-Government and the Intelligent Home*, Futuretext.

Akkaya, K., and M. Younis, “A survey on routing protocols for wireless sensor networks,” *Ad Hoc Networks*, vol. 3, 2005, p. 325–349.

Akyildiz, I., W. Su, Y. Sankarasubramaniam, and E. Cayirci, “Wireless sensor networks: a survey,” *Computer networks*, 2002, pp. 636-641.

Aldrich, C. (2009), *The Complete Guide to Simulations and Serious Games: How the Most Valuable Content Will be Created in the Age Beyond Gutenberg to Google*, HB Printing.

Al-Karaki, J., and A. Kamal, “Routing techniques in wireless sensor networks: a survey,” *Journal of IEEE wireless communications*, 2004, pp. 1-37.

Andreica, M. I., A. Dragus, A.-D. Sambotin, and N. Tapus (2010a). Towards a (Multi-)User-Centric Stack of (Multi-)Point-to-(Multi-)Point Communication Services. In: *Proceedings of the 5th Workshop on Enhanced Web Service Technologies (WEWST)*, pp. 36-45.

Andreica, M. I., A. Dragus, A.-D. Sambotin, and N. Tapus (2010b). Brief Announcement: Decentralized Construction of Multicast Trees Embedded into P2P Overlay Networks based on Virtual Geometric Coordinates. In: *Proc. 29th ACM Symp. on Principles of Distributed Computing (PODC)*, pp. 283-284.

Ariane's Life in the Metaverse, accessed on 15.02.2011.

Castranova, E. (2006), *Synthetic Worlds: The Business and Culture of Online Games*, University Of Chicago Press.

Chan, A.T.S, and S.-N. Chuang. (2003). MobiPADS: A Reflective Middleware for Context-Aware Mobile Computing. *IEEE Trans. Softw. Eng.* 29, 12, 1072-1085.

Chen, H. (2003). *An Intelligent Broker Architecture for Context-Aware Systems*, PhD proposal in Computer Science, University of Maryland, Baltimore, USA.

Content Conversion Specialists GmbH. *DocWorks*. http://www.content-conversion.com, Accessed on 15.02.2011

Costin-Anton Boiangiu, Andrei Spataru, Dan-Cristian Cananau, *Automatic text clustering and classification based on font geometrical characteristics*, Proc. 9th WSEAS International Conference on Automation and Information, 2008.

Craig A., Sherman W., Will J. (2009), *Developing Virtual Reality Applications: Foundations of Effective Design*, Morgan Kaufmann.

Kavitha. T., and D. Sridharan, "Security Vulnerabilities In Wireless Sensor Networks: A Survey," *J. of Information Assurance and Security*, vol. 5, 2010, p. 31–44.

Mainwaring, A., D. Culler, J. Polastre, R. Szewczyk, and J. Anderson, "Wireless sensor networks for habitat monitoring," *Proceedings of the 1st ACM international workshop on Wireless sensor networks and applications*, ACM, 2002, p. 88–97.

Moldoveanu A. (2008), *Highly-scalable server for massive multi-player 3D virtual spaces based on multi-processor graphics cards*, Annals of DAAAM for 2008 & Proceedings of the 19th International DAAAM Symposium, Published by DAAAM International, Vienna, Austria.

Multiverse Platform Architecture, accessed on 15.02.2011. http://update.multiverse.net/wiki/index.php/Multiverse_Platform_Architecture

New York Times. *Scan This Book! Article*. http://www.nytimes.com/2006/05/14/magazine/14publishing.html?_r=1. Accessed on 15.02.2011.

OCRopus. 2011, http://code.google.com/p/ocropus, Accessed on 15.02.2011

OpenSim. 2011, http://opensimulator.org, Accessed on 15.02.2011.

Richards, M., R. Monson-Haefel, and D. A. Chappell (2009). Java Message Service 2nd Edition. O'Reilly.

Rughinis, R., and L. Gheorghe, "Storm Control Mechanism in Wireless Sensor Networks," *2010 9th Roedunet Int. Conference (RoEduNet)*, 2010, pp. 430-435.

G. N. Srinivasan, G. Shobha, *An Overview of Segmentation Techniques for Target Detection in Visual Images*, Proceedings of the 9th WSEAS International Conference on Automation and Information (ICAI '08), WSEAS Press, June 2008.

Google Books. *About Section*. Accessed on 15.02.2011. http://books.google.com/googlebooks/about.html
http://arianeb.wordpress.com/2010/04/25/why-open-sim-is-the-future-metaverse-and-why-it-is-not-the-present/
http://opensimulator.org

Salber,D., A.K. Dey, and G.D. Abowd. (1999). The context toolkit: aiding the development of context-enabled applications. In *Proceedings of the SIGCHI conference on Human factors in computing systems: the CHI is the limit* (CHI '99). ACM, New York, NY, USA, 434-441.

Schmidt, A. (2002). *Ubiquitous Computing- Computing in Context*. Ph.D. Thesis, Lancaster University, UK.

Steve Man, *Intelligent Image Processing*, John Wiley & Sons, 2002.

W. Niblack, *An introduction to digital image processing*. Englewood Cliffs, NJ: Prentice-Hall, 1986, pp. 115–116.

Waldo, J. (2008), *Scaling in Games and Virtual Worlds*. Communications of the ACM, 51(8), 2008, pp 38-44.

Zachariadis, S., L. Capra, C. Mascolo, W. Emmerich. (2002). XMIDDLE: information sharing middleware for a mobile environment. In *Proceedings of the 24rd International Conference on Software Engineering*, ICSE 2002, ACM, New York, NY, USA, 712-712.

Keith Jeffery and Anne Asserson. 2008. Institutional repositories and current research information systems. *New Rev. Inf. Netw.* 14, 2 (Nov. 2008), 71-83.

Dragan Ivanovic, Dusan Surla, and Milos Rackovic. 2011. A CERIF data model extension for evaluation and quantitative expression of scientific research results. *Scientometrics* 86, 1 (January 2011), 155-172.

Jennifer Rowley. 2009. Digital Consumers: Reshaping the Information Professions. *J. Am. Soc. Inf. Sci. Technol.* 60, 11 (November 2009), 2383-2385.