SIGMCC: a System for Sharing Meta Patient Records in a Peer-to-peer Environment

Mario Cannataro [a], Domenico Talia [b], Giuseppe Tradigo [a], Paolo Trunfio [b], and Pierangelo Veltri [a] *

[a] *Experimental and Clinical Medicine Department, Magna Græcia University, Catanzaro - Italy*

[b] *DEIS, University of Calabria, Rende (CS) - Italy*

Abstract

This paper considers the interoperability and information sharing between health care providers. It proposes a distributed Peer-to-Peer (P2P) based framework that enables health operators of different hospitals to share and aggregate clinical information about patients and therapy effects. Patient records are mapped into a simple XML-based *meta-Electronic Patient Record* (meta-EPR). The meta-EPR is not a standard EPR proposal, but it is a lightweight data structure defined to contain relevant and aggregate information extracted from the different EPRs adopted by each hospital. Hospital operators formulate queries against meta-EPR schema; queries are then distributed to the connected hospitals hosting meta- EPR instances, through a P2P infrastructure. The presented framework has been fully implemented in a system called SIGMCC, which offers an Application Programming Interface (API) for query formulation, data loading and updating. As a case study, an application of the proposed meta-EPR to the cancer medical domain has been developed. Finally, SIGMCC implements a view mechanism to allow personal (patient) information protection against unauthorized users.

* Corresponding Author

Email addresses: cannataro@unicz.it (Mario Cannataro [a]), talia@deis.unical.it (Domenico Talia [b]), gtradigo@si.deis.unical.it (Giuseppe Tradigo [a]), trunfio@deis.unical.it (Paolo Trunfio [b]), veltri@unicz.it (Pierangelo Veltri [a]).

1 Introduction

The Electronic Patient Record (EPR) has become an essential tool for accessing in an efficient manner information regarding patient health history and personal data, required by administration, medical doctors and researchers. The EPR contains data of different types, from alphanumerical ones to images, notes and data produced by instruments, e.g., for biochemical analysis. Of particular interest is data regarding patient health history and treatments, e.g., allergy or required drugs. Recent efforts for defining standard guidelines for EPR compilation, electronic representation, and accessing control [1,2] have been guided by the importance of tracing any information about history of patient, but also by the necessity of monitoring expenses supported by local and central governments. Indeed, an EPR contains data about the various hospitals the patient visited, the family genetic pathologies, clinical treatments, but also personal information regarding incomes necessary for bills. Moreover, EPR stores information about treatments and response to drug types, that can also be used by researchers to derive information on clinical effects (outcome research).

Large volume of information can be stored on each EPR, but as for any large data container, efficient access methods and protocols are required to allow medical doctors to access and read data in an efficient manner. Data aggregation mechanisms are also necessary to provide general information about therapy efficiency, or to obtain indicators about clinical improvements. Indeed, decisions in health care often need to be taken by integrating and comparing data coming from multiple data sources. For instance, information may be found in different data repositories such as family doctors records, hospitals databases, and the national health system repository, each one using its own data structures. Moreover, single patient records may be stored by using different identifiers in the different data sources.

Currently, EPRs are organized in a stand-alone way and decisions in a health center are taken by using only local data, resulting in incomplete information. To enable effective decision-making, clinical and administration units need to access distributed data, and have to deal with schema heterogeneity, and access policies. Heterogeneity can be faced by using a mediator-based approach as in [4], where a mediator defines a global schema, and queries against the global schema are mapped to queries defined on the local data. A problem with this approach is the global-to-local schema mappings.

Recently, the use of XML [26] as a standard language for data exchange has been proposed for EPR sharing. Health Level 7 (HL7) is an organization which develops extendible standards for structured health care information to support patient care and information exchange [12]. The HL7 Clinical Document

Architecture (CDA), Release 2, is an ANSI standard (code name ANSI/HL7 CDA, R2-2005) released on April 2005 . The CDA Release 2.0 provides an exchange model for clinical documents. CDA documents are based on XML, the HL7 Reference Information Model (RIM) and coded vocabularies, and they can be displayed using XML-aware Web browsers.

Decentralized and Peer-to-Peer (P2P) approaches have been recently proposed for data access and integration [1] for XML-based EPRs. P2P systems are largely used to share information among distributed data sources. Indeed, each data source is managed by a peer node that maintains its own autonomy and shares resources with other nodes, using Internet for communication. Nevertheless, sharing and accessing remote data requires that each peer must know the data model used by the others. The problem in using a P2P framework for sharing EPRs is that no EPR standard has been yet adopted.

This paper proposes a hybrid P2P architecture allowing different health centers (hospitals) and operators to share information about patient records. Due to the lack of common EPR structure, we propose to collect representative data extracted from the different EPRs adopted by each hospital into an XML-based data structure named meta-EPR. The meta-EPR is not a novel EPR but a container that allows to collect in a simple way clinical information stored into local EPRs. Examples of such information are clinical data, therapy outcomes, number of patients cured for a specific disease (e.g., lung cancer). Meta-EPRs are obtained by extracting data from local EPRs using extraction rules. For instance, Fig. 1 reports a set of meta-EPRs obtained by extracting data from a set of different patient records. Meta-EPRs are populated by using wrapper modules (defined in [22]) which use logical disjunctive rules to extract data from EPRs.

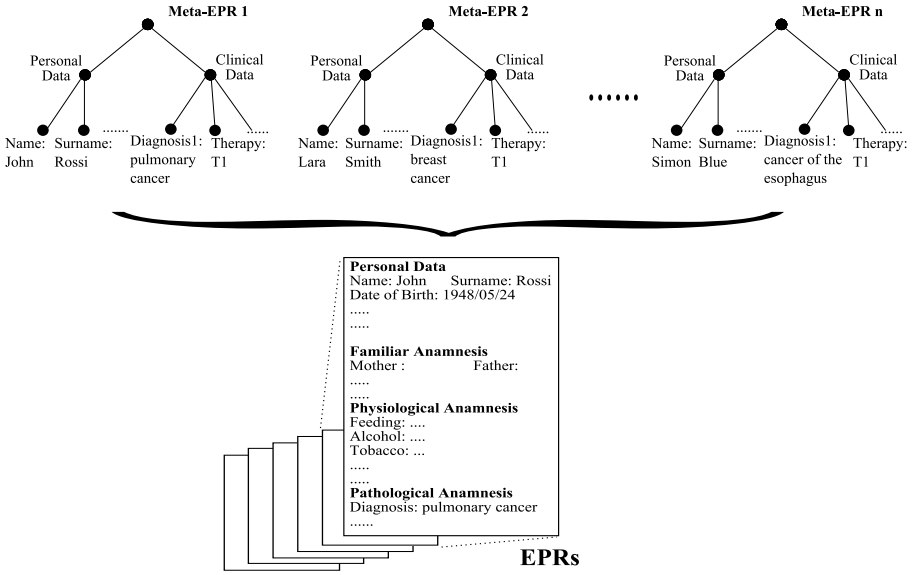

Fig. 1. An Example of EPRs mapped into a meta-EPR

Meta-EPRs are shared among health structures through a P2P infrastructure based on a super-peer architecture. In such an architecture each health structure provides one *super-peer* and one or more regular *peers* (e.g., one peer for each department or operator). Each super-peer hosts an XML native database, a Berkeley DB instance [23], containing the meta-EPRs of the health structure, while peer applications are used by medical doctors to submit queries against local or remote meta-EPRs. Queries are formulated on a peer and evaluated by its reference super-peer. The super-peer is then in charge of collecting local data and distributing the query across the other super-peers. Moreover, each database implements a view mechanism for securing data access. Indeed, queries coming from remote peers (i.e., remote databases) may select only anonymous information about therapy and clinical data, but none private information about patients may be treated by unauthorized users. In the proposed, simple security model, only local users of an hospital may access private data of local patients. The proposed framework has been fully implemented in a system prototype named SIGMCC, that has been tested at regional scale, proving an efficient and secure access to meta-EPRs. Moreover, the system includes administration functions for managing the addition of new meta-EPRs and updating existing ones with new data, improving the Berkeley database update function. Thanks to the P2P philosophy, the system scales with number of served hospitals. Finally, a preliminary beta version of SIGMCC has been presented at [7]. The contribution of this paper consists in defining a new and more realistic meta-EPR, the definition of update and security management modules, and the implementation of a new interface able to capture different EPR schema variations among super-peers.

The rest of the paper is organized as follows. Section 2 discusses related work. Section 3 describes the proposed meta-EPR and patient record data modeling and management. The section presents also the view mechanism used for privacy management. Section 4 describes the P2P infrastructure and communication strategy among health centers. Section 5 presents the SIGMCC prototype architecture and its implementation, while Section 6 reports the application experiences of SIGMCC as framework for health center cooperation. Finally, Section 7 concludes the paper.

2 Related Work

The American Health Level 7 (HL7) organization has proposed the Clinical Document Architecture as a standard for the exchange, management and integration of electronic health care information [12]. Such standard has been widely adopted by the American health providers. Software houses providing health information systems have developed electronic patient records that have been using especially by USA health providers. Patients may move through dif-

ferent hospitals by carrying on their electronic patient records, and databases may be shared among hospitals with appropriate access grants, to allow: (i) simple construction of patient's clinical history, and (ii) finding of relations (e.g. association rules) between diagnosis and therapy for studying new therapy protocols.

Following such an approach, the European Technical Normalization Committee, operating in 19 European member states, is trying to impose a preeminent health care information technology standard in Europe (CEN/TC 251) [8]. Nevertheless, no standard has been adopted by governments, leaving to health providers the choice of their own EPR management system. This is also due to the absence of legal value of the electronic information: in many countries legal valid EPRs have to be still paper format. On the other hand, the commonly adopted Digital Imaging and Communications in Medicine (DICOM) format is largely adopted for distributing and viewing medical image regardless of the origin instrument [19]. DICOM has been developed by a joint committee formed by the American College of Radiology (ACR) and the National Electrical Manufacturers Association (NEMA). It is developed in liaison with other Standardization Organizations including CEN/TC 251 in Europe and JIRA in Japan, with review also by other organizations including IEEE, HL7 and ANSI in the USA. Usually EPRs refer to images stored in separate specialized archiving systems by using DICOM.

This paper does not propose any new format, but it addresses the problem of managing different EPR formats by using the widely adopted standard model for data exchange, i.e. XML [26]. In our approach, EPR data is thus mapped into an XML document that is shared among peer nodes. The mapping between EPR data and meta-EPR field is obtained by specialized wrappers that are designed in a semi-autometic way.

Recently, XML-based EPRs have been proposed for data sharing among hospitals. For instance, [2] proposes a system that allows for querying XML data in a P2P environment. The user is able to add new XML data in the P2P environment and efficiently querying them by adding semantics to the document. The proposed meta-EPR shares common structure XML data, so that semantics is well known to the peer nodes. Similarly, [6] treats the problem of managing general purpose XML resources in a P2P environment. In [1] authors deal with the problem of managing privacy and data protection using XML Encryption [28] and XML Signature [29]. Differently by the cited works, the here proposed meta-EPR represents only reduced and (medical judged) relevant information to share.

Another XML-based EPRs for record exchange is proposed by the Synapses project [17]. The heart is the Federated Healthcare Record (FHCR) server which accepts requests for data (in the form of clinical objects) from clients,

decomposes them into queries against the connected "feeder" systems (where data is actually stored), and integrates the responses dynamically. The results of the Synapses project are the basis for the Healthcare Information Systems Architecture (HISA) [11]. This is *yet* another proposal of defining a commonly adopted data structure. The problem encountered during this project is that often operators need access to aggregate data in a simple way and in a distributed environment. The here presented framework showed that in case of oncology EPRs (used as data sets for our early experiments [20]), the meta-EPR structure and the P2P infrastructure are good candidates for simple data accessing and visualization for decision support systems and outcomes validations.

The use of ad hoc distributed infrastructures for managing and sharing medical information has been recently explored by some Grid-based projects such as Mammogrid [15] and ICGrid [16]. The goal of Mammogrid is to develop a European-wide database of mammograms that will be used to investigate a set of important healthcare applications, as well as to support effective co-working between healthcare professionals throughout the European Union. The ICGrid's aim is to create a distributed environment that enables the integration, correlation and retrieval of clinically interesting episodes across intensive care units.

Recently, the use of P2P platforms for sharing clinical information has been proposed. In [5], a P2P system that enables a community of radiologists to share radiological images and their associated diagnoses is presented. The use of P2P in such system allows to overcome one of the main limitations of centralized approaches, that is, the fact that data being shared can grow so much to make storing all the information on a single machine inefficient or infeasible. From an architectural point of view the main difference with our framework is that we adopt a super-peer model for the P2P infrastructure. The use of the super-peer architecture allows to achieve better scalability in real scenarios, where a large number of health structures are involved on a national or international scale.

Security data management for guaranteeing secure access to XML data has been studied and many approaches have been defined. The problem with XML is that control is based on rules that are defined on the server that hosts database, while due to the nature of XML the desiderata is that privacy management rules are defined in the XML document itself. There have been several proposals of XML access control methods such as [18], [10], while the W3C consortium has been defining an element encryption recommendation document for XML elements, allowing to encrypt only portion of the XML document [24], but the problem is still a discussion topic. SIGMCC implements an ad hoc filtering module allowing data server to send personal data (i.e. data protected by privacy laws) only to authorized clients, that are user

belonging to the the same health structure. In the P2P environment, the messages are passed as query answers to a peer sender. Nevertheless, the current version does not protect data from data interceptor, but this is acceptable due to the fact that only non-personal clinical data are distributed among hospital through the Internet.

SIGMCC also implements a module for updating XML documents that is able to add new subtree structure to XML documents in the Berkeley DB instance. XML document updating has been studied in [25], where update functions have been defined and integrated in XQuery [27] for an XML management system based on relational database. Such a problem has to be considered when using native XML databases that, although may support simple updating procedures, do not yet provide updating of portions of XML documents. In the here proposed framework, updates has to be performed on views [3], analyzing the meta-EPRs and identifying subtrees that have to be updated.

3 Modeling and Sharing EPR Data

The proposed framework is the result of a research project whose goal was sharing and accessing information stored in hospitals distributed in a geographical area [20]. The actors and users of the proposed framework are: (i) health providers; (ii) health care administrations managing funds for hospitals; and (iii) operators dealing with data in each structure. Health providers comprise hospitals, university medical centers, private health centers. The health administrations are orthogonal to the health providers and require indicators for rapidly monitoring expenses and efficiency of the health providers. Operators comprise medical doctors, nurses, medical researchers, administration operators.

The main problem for achieving the above discussed functionalities was data schema heterogeneity. This problem is addressed by the proposed meta-EPR, which contains few (but meaningful) information extracted from hospital and department repositories. The meta-EPR is authored having in mind the analysis goals and by considering the data fields available in local EPRs. After designing the meta-EPR schema, relevant EPR fields are mapped onto meta-EPR data and proper wrappers are generated (e.g. SQL-based wrappers for relational EPRs, or document-based wrappers for text EPRs). Data distribution and sharing are supported by using the P2P infrastructure described in Section 4. As an example, Fig. 2 reports the case of a medical doctor searching for 40 years old patients affected by lung cancer and cured with therapy protocol $p1$. The framework allows to define queries on a common data schema similarly to [3], and to retrieve information from local and remote hospital repositories. An application example of such a framework could be studying

therapy protocols effects on patients across several hospitals and having different clinical histories.

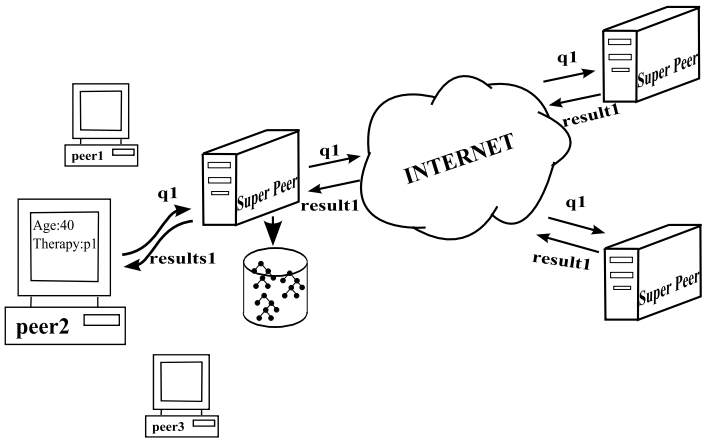

Fig. 2. An Example of Query Flow: an operator on peer2 issues the query q1, the local super-peer queries the local meta-EPR, sends q1 to others super-peers and collects results through the P2P infrastructure.

3.1 The Meta-EPR

The meta-EPR is a simplified patient record, able to contain relevant information extracted from real EPRs. The meta-EPR data model, defined using the XML language, contains both personal and clinical information. Personal information includes: *Name*, *Surname*, *Date of birth*, *Sex*, *Date of (possible) death*. Clinical information contains *Diagnosis*, *Year of the diagnosis*, *Phase of disease* and *Therapies*. The meta-EPR also contains (not mandatory) information on the particular domain of analysis. For example, information on *family clinical history*, *used therapy*, *cancer progression time*, and *notes* are included for the cancer domain. Currently, the meta-EPR is built in two steps: the meta-EPR schema is first defined taking into account generic and disease-specific information, then the relevant fields of each real EPR are identified and mapped on meta-EPR fields. In this phase some typical problems (e.g. synonyms) related to schema mapping need to be faced.

The meta-EPR data structure aims to simplify data accessing and sharing among clinical departments, where similar cases have been treated and result therapies have been collected, with progression/regression healthy results. Accessing to similar cases is useful for designing appropriate therapies. The medical domain here considered is related to cancer disease. The meta-EPR has been designed considering cooperation among oncology departments, and has been guided by using their requirements. Nevertheless, such meta-EPR can be used for any department with appropriate changing. In the following the proposed XML schema of the meta-EPR is reported.

```
<?xml version="1.0" encoding="UTF-8" ?>
<xs:schema targetNamespace="MEPR" ...>
<xs:element name="MEPR">
 <xs:annotation>
   <xs:documentation>attribute name="IdHospital" attribute
   name="DateOfSource"</xs:documentation>
 </xs:annotation>
 <xs:complexType>
  <xs:sequence>
    <xs:element name="PersonalData">
    <xs:complexType>
     <xs:sequence>
         <xs:element name="Surname" type="xs:string" />
         <xs:element name="Name" type="xs:string" />
         <xs:element name="Sex">
         <xs:simpleType>
             ...
             <xs:pattern value="[MmFf]" />
             ...
         </xs:element>
         <xs:element name="DateOfBirth" type="xs:date" />
             ...
         <xs:element name="FiscalCode">
             ...
         <xs:element name="ResidentialData">
             ...
         <xs:element name="ClinicalData">
             ...
         <xs:element name="Diagnosis" maxOccurs="unbounded">
            ...
         <xs:restriction base="xs:string">
         <xs:pattern value="[1234]"/>
           ...
         <xs:element name="Mutation" type="xs:string" minOccurs="0"/>
           ...
         <xs:element name="TimeToTheProgression" ...minOccurs="0"/>
         <xs:element name="Metastasis" type="xs:string" minOccurs="0"/>
           ...
         </xs:complexType>
         </xs:element>
         <xs:element name="PerformanceStatus" type="xs:string"/>
         <xs:element name="LifeQuality" type="xs:string" minOccurs="0"/>
         <xs:element name="ConcomitantPathologies" ... minOccurs="0" />
         <xs:element name="Allergies" type="xs:string" minOccurs="0" />
         <xs:element name="FamilialAnamnesis" minOccurs="0">
           ...
         </xs:element>
        </xs:sequence>
    <xs:attribute name="IdHospital" type="xs:string" use="required" />
    <xs:attribute name="DateOfSource" type="xs:date" use="required" />
  </xs:complexType>
 </xs:element>
</xs:schema>
```

Fig. 3. General schema of the meta-EPR for the cancer domain.

The proposed schema is in charge of hosting few but significant clinical data, to simplify the access to relevant information spread on different hospitals and for having rapid and efficient access to treatments, patients and therapies. Accessing and retrieving data coming from different hospitals, requires a data structure commonly adopted by different hospitals. Nevertheless, there is no commonly adopted solutions into hospitals thus that, the meta-EPR wants to cover the heterogeneity gap allowing users to formulate queries without worrying about different schemas. The XML language allows to support the

problem of taking in charge the possibility of finding possibly incomplete data in some hospital patient records. The XML-based meta-EPR instances allow to accept incomplete information where necessary, and, at the same time, offers a scalable solution in case of adding or removal of information. Any hospital has to associate to each patient record a meta-EPR instance. In the following we report an example of meta-EPR instances extracted from the oncology department of the University of Catanzaro Medical School [21] (name and surname do not indicate any real person for privacy reasons), used to populate the prototype database.

```
<?xml version="1.0" encoding="UTF-8" standalone="yes" ?>
<MEPR xmlns="MEPR" ... xsi:schemaLocation="MEPR MEPR.xsd"
IdHospital="Catanzaro Hospital" DateOfSource="2006-02-01">
<PersonalData>
  <Surname>Rossi</Surname>
  <Name>Paolo</Name>
  <Sex>m</Sex>
  <DateOfBirth>1954-02-02</DateOfBirth>
  <PlaceOfBirth />
  <Nationality />
  <FiscalCode>rsspl5402lcm1234</FiscalCode>
  </PersonalData>
  <ResidentialData>
  ...
  </ResidentialData>
  <ClinicalData>
  <Diagnosis Date="Feb 2002">
    <IstologicaDiagnosis>
      <Name>cancer pulmonary </Name>
      <Type>cancer type 1</Type>
      <Code />
      <Stadium>3</Stadium>
    </IstologicaDiagnosis>
    <TimeToTheProgression>7 months</TimeToTheProgression>
    <Metastasis>stomach </Metastasis>
        <Therapy>
      <Treatment>gemeitabin </Treatment>
    <Results />
    </Therapy>
    <Relapse />
  </Diagnosis>
  <Diagnosis Date="Gen 2004">
   ...
  </Diagnosis>
  <PerformanceStatus>3-4</PerformanceStatus>
    <LifeQuality />
    <ConcomitantPathologies>...</ConcomitantPathologies>
    <Allergies>...</Allergies>
     <FamilialAnamnesis>
        <Diagnosis>
       <Relative>grand father</Relative>
       <Name>cancer</Name>
       <Code />
      </Diagnosis>
   </FamilialAnamnesis>
    <DateOfDeath>2005-12-12</DateOfDeath>
    </ClinicalData>
  </MEPR>
```

Fig. 4. Example of meta-EPR instance for the cancer domain.

The meta-EPR defined above allows to support several instances of diagnosis for a patient, allowing to store historical information on treatments and

hilliness in an aggregate form, and using a single XML document. Meta-EPR instances may be queried for retrieving information that can be used for studying and mining clinical information. The meta-EPR reported here has been defined for oncology departments where obtaining relevant and representative information about clinical treatments and results, in a simple and representative way, is very important for defining therapy strategies.

3.2 Data Management

Meta-EPRs instances do not replace EPRs, but require their existence and their organization in electronic repositories. With the cooperative work as a target in mind, the meta-EPR instances are created by wrapping data from patient records of each hospital. Thus each health department that wants to participate to a cooperative net based on meta-EPR sharing, has to use a data wrapper that, knowing the local data organization, extracts information and maps them into meta-EPR instances. Thus, meta-EPRs are produced as XML instances, and are stored into a native XML database.

We have chosen the BerkeleyDB [23] native XML database, a research product freely available [1] . It supports the XQuery language and provides a set of basic functions for managing data. Each health structure that wants to share health information has to present a BerkeleyDB instance. For instance, in Fig. 5 two different wrappers extract EPR information from the oncology departments DEPT D1 and DEPT D2 local repositories. The wrappers take in charge of the different data organizations. Both wrappers export data in meta-EPR instances and they are loaded into the BerkeleyDB instance hosted into the hospital. Due to the lack of EPR standard, each EPR repository may present its own logical schema and data format (e.g., relational data, text documents, etc.), thus requiring ad-hoc wrapper modules to extract data. For the here presented framework, wrappers are defined by using a disjunctive logic program module, defined in [22]. Such a module allows the semiautomatic wrapping of documents and is not described in this paper.

The representation of patient records in the meta-EPR requires a up to date maintaining phase that takes in charge updates in the hospital patient records. XML documents representing meta-EPRs have to be updated with respect to the repository updates. We consider only insertion of new information into the meta-EPRs repository. One of the following cases is possible:

- a patient record is inserted into the hospital (or department) database and it refers to a patient that has never been hosted by the health structure. In

[1] At the time of submission, the company producing BerkeleyDB has been acquired by Oracle, but it is still an Open Source project

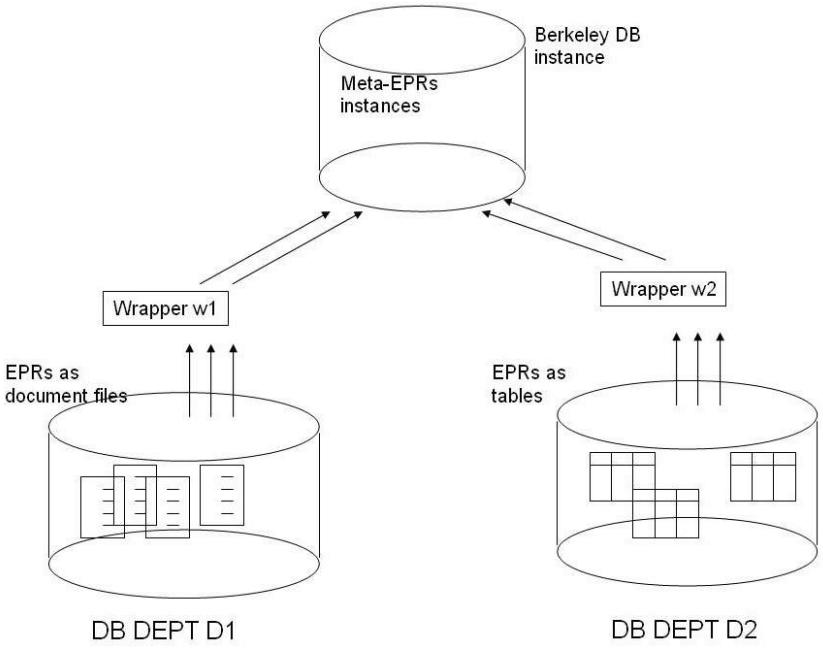

Fig. 5. Extracting information from two different EPR databases by using specialized wrappers

this case, a new meta-EPR has to be created and inserted in the BerkeleyDB instance.

- a patient record is inserted into the hospital (or department) and a meta-EPR instance relative to such a patient (and its personal data) is already present into the XML database instance. In this case, new clinical data has to be inserted into the existent meta-EPR that needs to be modified and enriched with new information.

The first case is simple and the insertion operation is easily supported by the BerkeleyDB module. The latter is not supported as update function in the BerkeleyDB due to the fact that only part of an existent XML document has to be modified. A simple update module has been defined to manage the update function, to take in charge new patient records.

Finally, schema updating is managed by single BerkeleyDB administrator. By the way, schema updating is only limited to adding information to the meta-EPRs, to guarantee that meta-EPRs instances are still valid with respect to the new schema and queries coming from remote users can still be supported.

3.3 Querying and Security

The BerkeleyDB supports XQuery [27] queries formulated on XML document paths. Queries may be formulated against the meta-EPR schema associating

search parameters to the attribute values. Knowing the XML schema, search parameters are defined as search condition on element nodes. In a tree document representation, as the one showed in Fig. 6, an operator may formulate a query to find information contained in the meta-EPRs, as for instance, *find patients cured with protocol P1 and affected by lung cancer*. This will avoid user to query the whole patient records and allows to collect data from different hospitals. Since meta-EPRs databases are shared among different hospitals, a doctor may be able in finding information from local and remote databases, avoiding reading many (and often heterogeneous) patient record instances adopted in different hospitals (or in different department). The operator interested in collecting aggregate information from remote EPRs may thus collect data rapidly into a uniform data schema, enabling the implementation of clinical decision support systems. Note that meta-EPRs allow a cooperation among departments that use (even quite) different EPR structures belonging to the same hospital. Query results are presented as a stream of XML documents, i.e. copies of meta-EPRs that satisfy search constraints. Aggregation functions may be performed directly on the XML document stream.

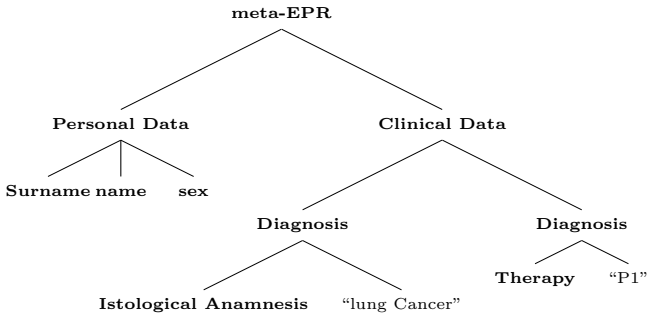

Fig. 6. Query Defined on Meta-EPR

For instance, the query defined on the meta-EPR schema reported in Fig. 6 searches for patients affected by lung cancer and cured with protocol P1.

While querying meta patient records, we have to take into account that meta-EPRs may contain sensitive private data, i.e. data that refers to personal information that can be accessed and manipulated only by authorized operators. We implemented a simple access control method that is based on a view mechanism defined on server side, i.e. on the machine hosting the meta-EPR database. When query results must be returned to non-authorized operators, as for instance in case of (remote) queries coming from different structures, the query tree results are yet generated as XML documents conforming to the meta-EPR XML schema, but they have empty values for personal data fields. Fig. 7 shows how data results are presented and filtered of personal data.

Currently, there is a lot of interest for securing XML data by using cryptography. We avoid to send personal information through users. However, the proposed module may always be updated with criptography module if it is

required to dispacth messages containing personal information.

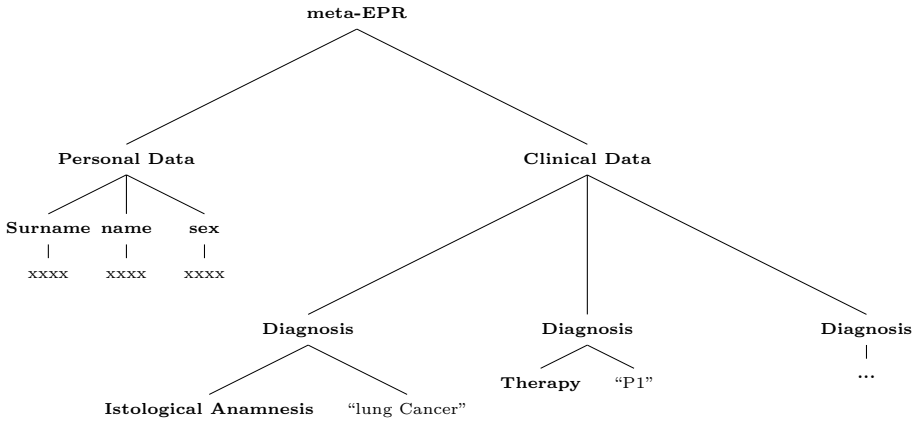

Fig. 7. Results on Meta-EPR instances, with Personal data Filtering

4 Peer-To-Peer Infrastructure

The P2P infrastructure adopted as the cooperation framework of this system is based on a *super-peer* architecture. Super-peer networks include two kinds of nodes: *super-peers* and *peers*. A super-peer node acts as a centralized server for a number of regular peers, while super-peers connect to each other to form an overlay network that exploits P2P mechanisms at a higher level. In the following, architecture, implementation, and functioning of the such an infrastructure are described.

4.1 Architecture

Fig. 8 shows the architecture of the P2P infrastructure.

From an administrative point of view, the system is composed of a set of autonomous health structures, each one including an arbitrary number of hospitals and operators. According to the super-peer approach, two kinds of applications are defined in this scenario:

- A *peer* is an application through which health operators can search and query the clinical data shared among the participant structures.
- A *super-peer* is an application that supports the operations of a group of peers, also through the interaction with other super-peers in the network.

Within each hospital one or more peer applications can be executed. For instance, peer applications can be installed on all the machines used by operators. On the contrary, there is exactly one super-peer application per health

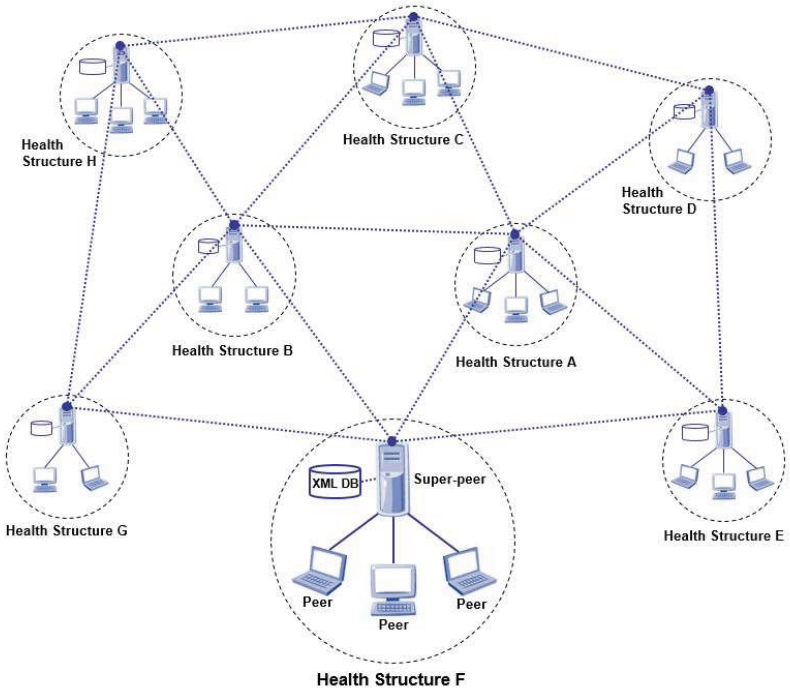

Fig. 8. Architecture of the P2P infrastructure

structure. Since the super-peer acts as server for a set of peers, it must be installed on a machine satisfying appropriate requirements in terms of reliability, availability and efficiency.

As shown in Fig. 8, each peer can communicate only with a reference super-peer, while super-peers communicate among them in a P2P fashion through the Internet. Each super-peer also hosts the XML database containing data visible and accessible to the other health structures in the system.

4.2 Implementation

The peer application is composed of two software modules: *User Interface*, that manages the interaction with the local user, and *Communication Manager*, which is responsible for managing the communication with the local super-peer, including search/response and connection/disconnection messages.

Similarly, the super-peer application includes two components: *Communication Manager* and *Data Manager*. The Communication Manager manages the communication with the other super-peers and the set of local peers it is responsible for. The Data Manager is in charge of accessing and querying the local XML database.

Communications among peer and super-peer applications are implemented

using the *JXTA* framework [9]. JXTA provides a set of XML-based protocols that allow computers and other devices to communicate and collaborate in a P2P fashion. A key concept in JXTA is that of *peer group*. Basically, a peer group is a collection of peers that have a common set of interests.

The peer group concept has been used to create different levels of aggregations among peers and super-peers. A *local peer group* includes all the nodes (peers and super-peer) in a given health structure. This group provides mechanisms for discovering the active nodes within the health structure and for supporting communication between peer nodes and the local super-peer.

Since peers of different hospitals belong to different local peer groups, they cannot directly communicate among them. To allow the sharing of meta-EPRs among different structures, a second level of aggregation, defined as *super-peer group*, has been introduced. This group includes all the super-peers in the network, and provides mechanisms for discovering the active super-peers and sending search/response messages among them.

4.3 Search mechanism

Searching data of interest using the P2P infrastructure is a multi-step task. In particular, the following steps are executed when a health operator wants to perform a distributed search over this infrastructure:

(1) The health operator specifies the search parameters and submits the request using his/her peer application.
(2) The peer application submits the search request to the local super-peer.
(3) The local super-peer performs the search query on its database and forwards the request to the remote super-peers in a peer-to-peer fashion.
(4) The remote super-peers perform the search queries on their databases and return back the results to the local super-peer.
(5) The local super-peer returns local and remote results to the peer application, which then presents the results to the operator.

As mentioned before, the User Interface of the peer application manages the interaction with the local user, allowing to formulate queries, submit requests, and visualizes results. Details about this interface and its use are given in Section 5.

4.4 Scalability remarks

Super-peer networks have been originally proposed to achieve a balance between the inherent efficiency of centralized search, and the autonomy, load balancing and fault-tolerant features offered by distributed search. Currently, the super-peer model is adopted by a number of widely-used P2P file sharing systems and protocols, such as KaZaA [14] and Gnutella [13].

Several studies have been conducted to evaluate the scalability of the super-peer model. For instance, in [30] the performances of super-peer networks are evaluated, and rules of thumb are given for an efficient design of such networks. This study demonstrates that super-peer architectures enhance the performance of search operations with respect to flat P2P networks, limiting in a significant way bandwidth consumption and processing load.

Since in a super-peer network communications take place only among super-peers (whose number is proportional to the number of organizations), this kind of architecture is the most appropriate to ensure extensibility and scalability in our system both at a national and international level. Another important benefit arising from the super-peer architecture is the autonomy of the single health structures. This is achieved because clinical data is managed only in local databases, and - at the same time - each health structure can set up an arbitrary number of local peers without requiring global coordination.

5 SIGMCC System Architecture and Prototype

The described system has been fully implemented [20] and has been tested with regional hospitals each one hosting meta-EPR and a super-peer node. The typical organizational structure of an health center, which comprises different departments and possibly different EPRs, is reflected in the architecture of SIGMCC. As is explained below, a peer represents a department or a user, while a super-peer represents the entire health center and its connection to the others centers.

The SIGMCC architecture comprises the following components:

- EPR wrappers, which extract data from EPRs belonging to an health center and feed the meta-EPR managed by a super-peer;
- meta-EPR, it collects relevant data managed by EPRs local to an health centers, and can be queried by the super-peers;
- super-peer nodes, which host the meta-EPR of an health center and support local/remote querying, query forwarding and result collection;

- peer nodes, which offers a Graphical User Interface through which the user is able to issue queries and to analyze results.

Both super-peers and peers communicate using the P2P infrastructure described in the previous section, by using a JXTA-API module.

5.1 EPR Wrapper

Each health center usually hosts a set of EPRs (e.g. departmental EPRs): an EPR wrapper module extracts relevant data from EPRs and copies them into the meta-EPR that is managed by the hospital super-peer. In case of relational EPR, the wrapper is similar to an ETL (Extract, Transform, Load) datawarehouse tool, while in the case of text-based EPR, the wrapper has to select specific portions of the text and map them into the meta-EPR (see Fig. 5. We currently use the approach described in [22]. A more simple solution could be used if EPRs are developed by using the XML-based Clinical Document Architecture [12].

5.2 Super-Peer Node

The super-peer node (see Fig. 9) contains: the XML meta-EPR repository, the JXTA-based module implementing the P2P mechanisms, and a Querying/Updating API bridging the database with the JXTA module, and implementing both data loading, for populating the BerkeleyDB instance, and data querying, for answering queryies coming from the local or remote peers. The Querying/Updating API, together with the Wrapper module, implements the update philosophy described in the Section 3.2 and allows to maintain updated the BerkeleyDB instance with respect to inserted patient records in the hospital repository. Moreover, the architecture implements secure module for identifying the user that wants to access to data, filtering out subtree of the meta-EPRs obtained by implementing a query, that cannot be sent to the user.

The super-peer repository is an instance of the BerkeleyDB [23] used to store the XML meta-EPRs. The Querying/Updating module is in charge of managing queries received from peers located in the same local network of the super-peer and/or from remote super-peers. Such module uses the open source API provided with BerkeleyDB. It is thus possible to load meta-EPRs and to formulate XQueries [27] against the database. Each super-peer may access both the local network for managing local peers, and the Internet for distributing queries formulated by some of its peers to the other super-peer nodes, and also to answer remote queries. Query management is provided by the API

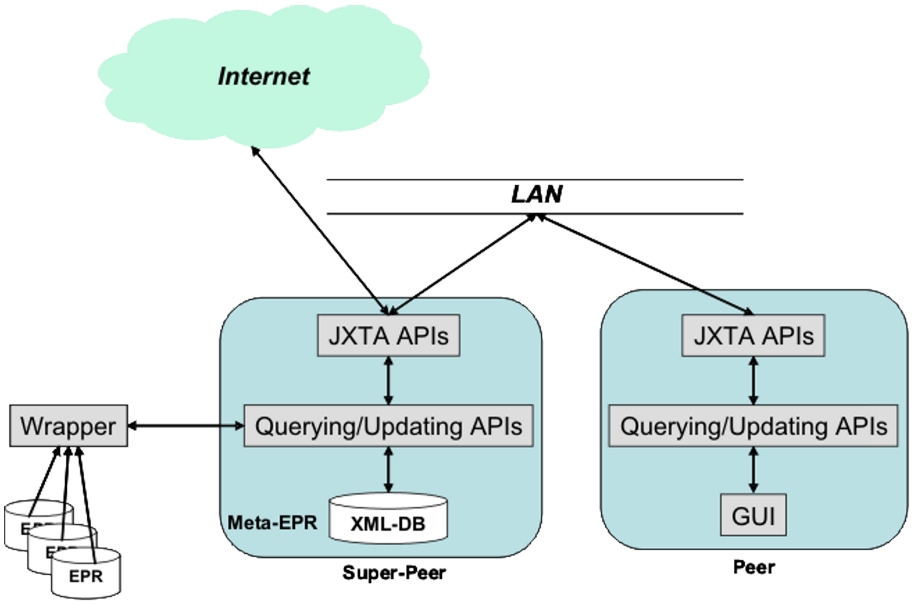

Fig. 9. Super-peer and Peer Modules

(see left part of Fig. 9), while P2P communications are provided by using the JXTA-based framework.

5.3 Peer Node

The querying functionality is offered to the user through a Graphical User Interface (GUI) installed on the peer node. The peer node also contains a JXTA-based module (JXTA API) that provides connection to the local area network and manages communication with the local super-peer.

The GUI allows to formulate queries, either using a query-by-example like structure, or by using an XQuery expression. In the first case the system is in charge of translating the query in an XQuery expression. The query is sent to the local super-peer that is in charge of formulating it to the local database and to dispatch it through the Internet to the other super-peers. For instance Fig. 10 reports the formulation of a query looking for patients affected by lung cancer, using the graphical interface. The GUI supports also wild card for composing queries, while experts may use XQuery expressions.

Results are obtained as (portions of) XML documents, and the GUI represents them reporting the name of the remote or local hospitals (i.e. super-peer) with the associated query (e.g. $q1$). Fig. 11 reports the results obtained from two hospitals (i.e. two different databases in two super-peers). The right part of the figure shows meta-EPRs elements retrieved from the Hospitals. User may navigate through the document hierarchy or may visualize the results in a table format using the visualize button reported in the bottom part of the figure.

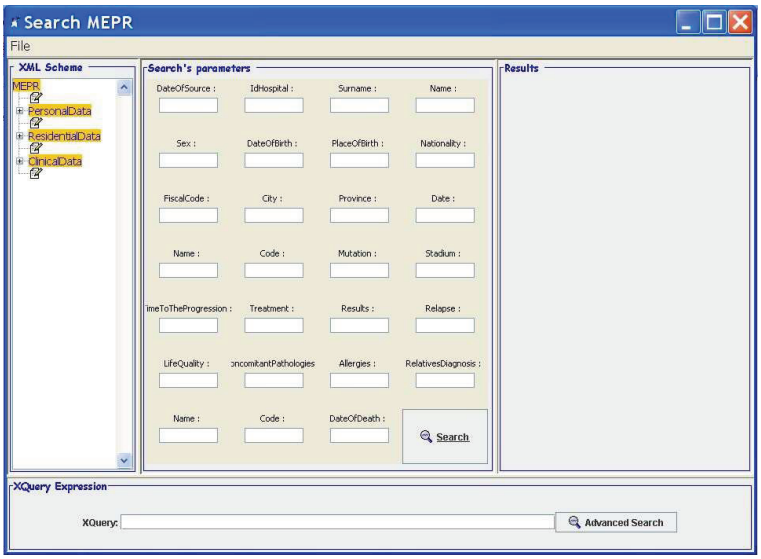

Fig. 10. Formulating Queries through the Peer GUI

Finally, the prototype supports join evaluation among meta-EPRs contained in different super-peer nodes. Note that the GUI blinds personal information about patient meta-EPRs returned to the requesting peer by remote hospitals (i.e. whose access is not authorized to the requesting peer).

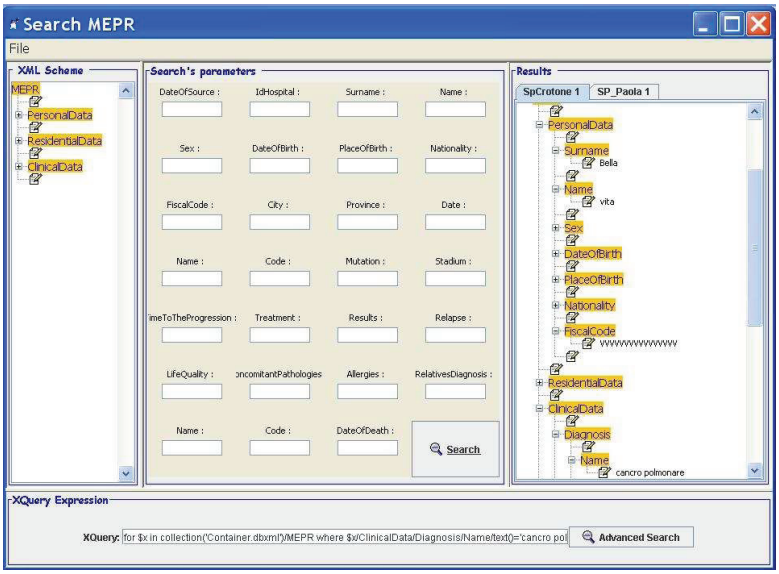

Fig. 11. Getting Query Results from Hospitals

Fig. 12 presents the graphical user interface for inserting new patient record into the meta-EPR repository. Note that such a module, stores the structure of the XML document. At the beginning, the system loads the schema structure of the meat-EPRs and then loads the meta-EPR instances validated with respect to this structure. Update is done automatically, depending on the new meta-EPRs.

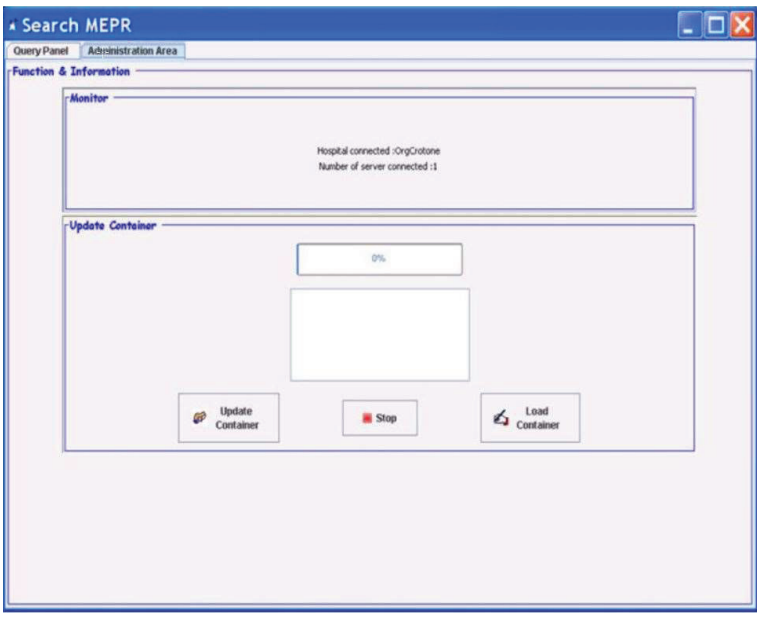

Fig. 12. Administrator updates the container with new EPRs

6 Application

The SIGMCC platform is the result of a research project [20]. In such a project the oncology domain was chosen and the previously presented meta-EPR was designed. Then, a set of valid EPR was extracted from the oncology department of University Magna Graecia Medical Hospital, and data were wrapped for populating meta-EPR instances. Four oncology departments of different health centers in Calabria region were interested in installing the SIGMCC platform, and local meta-EPR databases were defined with the common meta-EPR schema. Different data structures are currently used in such different structures (relational, plain text, and Microsoft Word formats). Simple queries set has been defined on the data set, and applied to the clinical data contained in the meta-EPR instances. Due to privacy procedures for personal data treatments, we were able to use only the real data set furnished by the University of Magna Graecia Hospital, that, as member of the project, was able to clean the data of personal information. On this data set we performed tests to study the medical doctor feedbacks in terms of clinical utility and in terms of performance indicators (e.g. percentage of survivals by years, percentage of relapses, etc.). The prototype and queries set have been presented to the medical community. Its application for use in different hospitals is part of a new project regarding tests and applications that is currently under reviewing process of the Regional Government.

7 Conclusions

The paper presented a framework for sharing and aggregating information provided by distributed health centers. The system introduces the concept of meta-EPR as a simple common adopted EPR containing relevant information extracted by different EPRs, while heterogeneity among different EPRs is faced through specialized wrappers. The framework uses a P2P infrastructure to support the communication and data transfer in a scalable way. Querying, updating and security have been realized on XML data, language used to describe and exchange meta-EPRs.

Acknowledgements

This work has been partially supported by the P2P-CKMSSSDM project (http://www.calio.it/p2p) funded by Regione Calabria (P.O.R. measure 3.16). Authors thank Domenico Conforti, Clara Pizzuti, Pierosandro Tagliaferri, and Antonio Volpentesta, for the discussions during the project meetings and Valeria Fionda for her contribution in the system implementation. They are also particularly grateful to D. Saccà for meta-EPR intuition.

References

[1] S. Abiteboul, O. Benjelloun, B. Cautis, I. Fundulaki, T. Milo, and A. Sahuguet. An electronic patient record "steroids": Distributed, peer-to-peer, secure and privacy-conscious. In *Very Large Data Base Conference (VLDB)*, 2004.

[2] S. Abiteboul, I. Manolescu, and N. Preda. Constructing and querying peer-to-peer warehouses of xml resources. In *International Conference on Data Engineering (ICDE'05)*, 2005.

[3] V. Aguilera, S. Cluet, T. Milo, P. Veltri, and D. Vodislav. Views in a Large Scale XML Repository. *VLDB journal*, 11(3), 2002.

[4] Eta S. Berner, editor. *Clinical Decision Support Systems Theory and Practice*. Health Informatics. Springer Verlag, 1998. ISBN: 0-387-98575-1.

[5] Ignacio Blanquer, Vicente Hernandez, and Ferran Mas. A p2p platform for sharing radiological images and diagnoses. In *Proc. DiDaMIC Workshop*, 2004.

[6] A. Bonifati, E. Q. Chang, L.V.S. Lakshmanan, T. Ho, and R. Pottinger. Heptox: Marrying xml and heterogeneity in your p2p databases. In *VLDB*, 2005.

[7] M. Cannataro, D. Talia, G. Tradigo, P. Trunfio, P. Veltri, and G. Zarola. A peer-to-peer infrastructure for sharing electronic patient records. In *UPGRADE-CDN Workshop, in conjunction with IEEE HPDC 2006 Conference, June 19-23, 2006, Paris, France.*, 2006.

[8] CEN/TC 251 European Standardization of Health Informatics - http://www.centc251.org/.

[9] CollabNet Inc. The JXTA Project. http://www.jxta.org.

[10] E. Damiani, S. De Capitani di Vimercati, S. Paraboschi, and P. Samarati. A fine-grained access control system for xml documents. *ACM Transactions on Information and System Security (TISSEC)*, 5(2):169–202, 2002.

[11] F. M. Ferrara. Tutorial on the cen/tc251 hisa standard: healthcare information systems architecture. *Studies in Health Technology and Informatics*, 45, 1997.

[12] Health Level 7 - http://www.hl7.org/.

[13] Gnutella http://rfc gnutella.sourceforge.net/.

[14] KaZaA http://www.kazaa.com/.

[15] Mammogrid http://www.mammogrid.com/.

[16] http://cygrid.org.cy/docs/HPCL-ICGrid-intro-3.pdf ICGrid Intensive Care Grid.

[17] Benjamin Jung and Jane Grimson. Synapses/synex goes xml. *Studies in Health Technology and Informatics*, 68:906–911, 1999.

[18] G. Miklau and D. Suciu. Cryptographically enforced conditional access for xml. In *Fifth Int. Workshop on the Web and Databases (WebDB 2002), June 6-7, 2002, Madison, Wisconsin*, 2002.

[19] Nema.org. Digital Imaging and Communications in Medicine - http://medical.nema.org/.

[20] P2P-CKMS-SSDM Project site - http://bioingegneria.unicz.it/veltri/projects.htm or www.calio.it/p2p.

[21] Policlinico Universitario Campus Magna Graecia. www.unicz.it.

[22] M. Ruffolo, N. Leone, M. Manna, D. Saccà, and A. Zavatto. Exploiting asp for semantic information extraction. In *International ASP'05 Workshop*, 2005.

[23] SleepyCat Software. The BerkeleyDB. http://www.sleepycat.com/.

[24] XML Encryption Syntax and http://www.w3.org/TR/xmlenc-core/ Processing W3C Recommendation 10 December 2002.

[25] Igor Tatarinov, Zachary G. Ives, Alon Y. Halevy, and Daniel S. Weld. Updating xml. In *SIGMOD '01: Proceedings of the 2001 ACM SIGMOD international conference on Management of data*, pages 413–424, New York, NY, USA, 2001. ACM Press.

[26] World Wide Web Consortium. XML Language. `http://www.w3.org/XML`.

[27] World Wide Web Consortium. XQuery 1.0 Query Language. `http://www.w3.org/TR/xquery/`.

[28] XML Encryption WG - http://www.w3.org/Encryption/2001/.

[29] XML Signature WG. http://www.w3.org/Signature/.

[30] B. Yang and H. Garcia-Molina. Designing a super-peer network. In *Proc. Int. Conference on Data Engineering (ICDE 2003)*, 2003.