# A Personalization Environment
# for Multi-Version Clinical Guidelines

Fabio Grandi

Dipartimento di Elettronica, Informatica e Sistemistica
Alma Mater Studiorum – Università di Bologna
Viale Risorgimento 2, I-40136 Bologna, Italy
email: `fabio.grandi@unibo.it`

**Abstract.** In this work, we introduce a personalization environment for the representation and efficient management of multi-version clinical guidelines. The environment is composed of an XML repository accessed through a personalization engine, which uses temporal perspective, patient profile and context information to reconstruct a guideline version tailored to a specific use case. In particular, we apply and extend to clinical guidelines solutions we previously developed for norm texts in the legal domain, and show how multi-version representation capabilities and personalization query facilities can be added to their management.

**Keywords:** clinical guidelines, document retrieval, temporal database, ontologies, personalization, versioning, XML

## 1 Introduction

Clinical guidelines are definitions of "best practices" encoding and standardizing clinical procedures for a given disease [18]. The advantages of adopting computer-based guidelines as a support for improving the work of physicians and optimizing hospital activities have been acknowledged by many authors and several computer systems have been developed (see e.g. [8, 11]). Clinical guidelines are subject to continuous development and revision by committees of expert physicians and health authorities and, thus, multiple versions coexist as a consequence of the clinical and healthcare activity.

In this paper, we propose to apply to the management of clinical guidelines some techniques we previously developed for norm documents in the legal domain [13, 14], which present strong similarities. Hence, we will introduce solutions to model and to provide personalized access to multi-version clinical guidelines, which can be stored both in textual and in executable format in an XML repository [30]. The XML language has already been proposed by many authors and adopted in several research projects (e.g. [3, 6, 26]) as a suitable means to encode clinical guidelines. Hence, our approach can be considered as a compatible extension of such proposals, to which we aim at adding multi-version representation capabilities and personalization query facilities.

To this end, we will describe how a multi-version XML data model and the prototype system we developed for e-Government applications can be applied to the representation and management of multi-version clinical guidelines. In this way, multiple
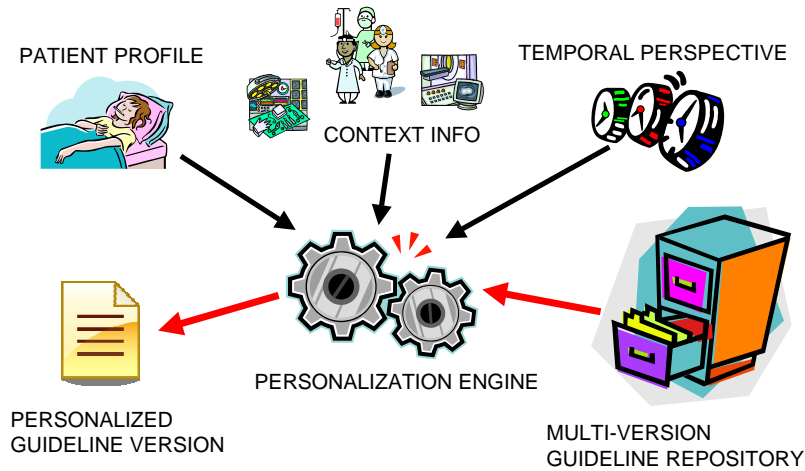
Fig. 1: Personalized access to multi-version guidelines.

temporal perspectives, patient profile and context information can be used by an automated personalization service to build a guideline version tailored to a specific use case (see Fig. 1).

The paper is organized as follows. In Section 2, temporal and semantic versioning of clinical guidelines is introduced with reference to advanced application requirements. Section 3 is devoted to the description of a multidimensional XML data model supporting temporal and semantic versioning of guidelines. In Section 4, a prototype system efficiently implementing the personalization engine sketched in Fig. 1 is briefly described. Conclusions and description of future work will finally be found in Section 5.

## 2  Versioning of Clinical Guidelines

The fast evolution of medical knowledge and the dynamics involved in clinical practice imply the coexistence of multiple *temporal versions* of the clinical guideline documents stored in a repository, since guidelines are continually subject to amendments and modifications. In fact, it is crucial to reconstruct, borrowing the term from the legal field, the *consolidated version* of a guideline as produced by the application of all the modifications it underwent so far, that is the form in which it currently belongs to the state-of-the-art of clinical practice and, thus, must be applied to patients today. However, also past versions are still important, not only for historical reasons: for example, a physician might be called upon to justify his/her actions for a given patient P at a time T on the basis of the clinical guideline versions which were valid at time T and applicable to the pathology of patient P. In other words, temporal concerns are important in the medical domain as they are in the legal domain and, thus, a guideline management system should be able to retrieve or reconstruct on demand any temporal version of a given clinical guideline to meet advanced application requirements.

Moreover, another kind of versioning, which we will call *semantic versioning*, plays a fundamental role, because clinical guidelines or some of their parts have limited *applicability* with respect, for instance, to the population of patients. In fact, a given guideline (e.g. involving treatment of heart diseases) may contain different recommendations which are not uniformly applicable to the same classes of patients: one general therapy may be non applicable to persons who suffer from some metabolic disorders (e.g. diabetes mellitus) or chronic diseases (e.g. kidney failure) or present some addiction (e.g. cocaine); one first-choice drug may not be given to patients who are already under treatment with possibly interacting drugs (e.g. anticoagulants), or show genetic or acquired hypersensitivity or intolerance to some substances (e.g. patients with enzymatic defects or documented allergies), and so on. Hence, when dealing with a specific patient case, a physician may be interested in finding a *personalized version* of a clinical guideline, that is a version tailored to the patient's health state and anamnesis, only containing recommendations which are safely and effectively applicable to his/her personal case.

In addition to linking guidelines to classes of patients, semantic versioning can also involve more generic applicability contexts (e.g. hospitals without PET diagnostic equipment, or selected centers taking part to a clinical trial), which might require the application of a particular version of the general guideline, which may also no longer be part of the consolidated state-of-the-art guideline. For instance, consider version v1 of a clinical guideline G which prescribes a biopsy to confirm a cancer diagnosis but has been superseded by a new version v2 which introduces a PET scan for the same cancer diagnosis, making in most cases the biopsy unnecessary. However, in some hospital H which is not equipped with a PET scanner, the right version of G to be followed is v1, although no longer considered valid by the medical community. Therefore, the applicable version of the guideline for context H is G(v1), with biopsy as a mandatory diagnostic means. This example also shows how temporal and limited applicability aspects may also interplay in the production and management of versions.

## 2.1 Temporal Versioning

As far as temporal versioning is concerned, several independent time dimensions are involved in the representation and management of clinical guidelines, in particular when we consider an environment also supporting the guideline authoring and approval process. Relevant time dimensions include valid, event, availability, proposal and acceptance times [5, 27]. Even considering an environment where only approved guidelines are stored, and retrieved by final users to be consulted or followed, at least two time dimensions are relevant:

**Validity time.** It is the time the guideline is considered in force by the medical community and, thus, is applied to patients. It has the same semantics of valid time as in temporal databases [20], since it represents the time the guideline actually belongs to the state-of-the-art of clinical practice.

**Efficacy time.** Borrowing the term from the legal domain, it is the time the guideline can be applied to a concrete case. It usually corresponds to validity, but it might be the case that an obsolete, superseded guideline continues to be applicable to a limited number of cases. While such cases exist, the guideline continues its efficacy though no longer considered in force.
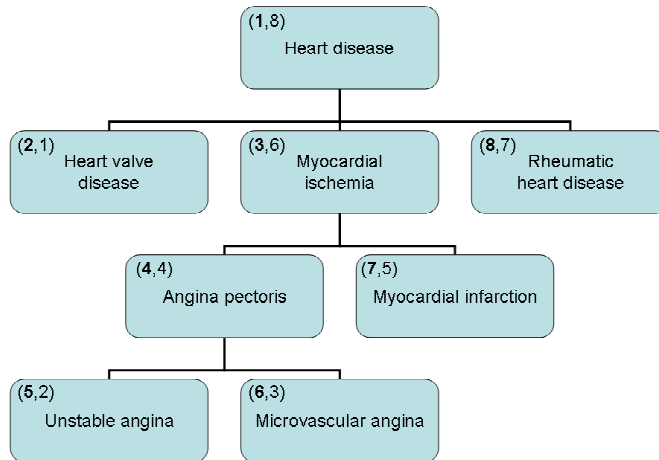
Fig. 2: A sample ontology involving a taxonomy of hearth diseases, where each class has a name and is associated to a (**pre-order**,post-order) pair.

Notice that validity and efficacy time both have the semantics of valid time but represent different and independent valid time notions. Both are necessary to correctly deal with cases as the one in the last described example: the guideline version G(v1) for the applicability context H can still be selected today as its efficacy includes current time, although its validity does not. Furthermore, in addition to the time dimensions which model the dynamics of guidelines in the real world, transaction time [20] plays an important role when automatic management of information through computer systems is involved and, thus, should never be neglected, since it allows to execute retro- or pro-active modifications and to keep track of their execution for audit purposes. For example, it might be the case that a physician makes a wrong decision in choosing a drug following the provisions of a guideline retrieved from the system when the returned consolidated version is actually out-of-date; the decision is taken while a modified version of the guideline (e.g. involving the adoption of some more effective and less potentially dangerous drug) is already available but has not been stored in the information system yet. Hence, transaction time is needed to ascertain *a posteriori* that the correct version was stored retroactively and, thus, the physician acted in good faith.

Temporal versioning along multiple time dimensions can be added to documents in an XML repository by making temporal the XML encoding [7], that is introducing timestamps as annotations in the XML document.

## 2.2   Semantic Versioning

Semantic applicability of multi-version resources can be defined with reference to domain ontologies. Ontologies [16, 17], which are conceptualizations of a domain into a machine-understandable format, have recently become quite popular with the advent of the semantic web [1], where the introduction of common reference ontologies is

necessary to allow information and its interpretation to be shared by both human and automatic agents.

Appropriate applicability of clinical guidelines to individual patients can be defined according to a consensual taxonomy of diseases, like the ICD-10 endorsed by the World Health Organization [19] or the MeSH Section C maintained by the US National Library of Medicine [22]. For instance, consider Fig. 2, which depicts a small portion of a medical ontology representing a classification of principal heart diseases. Notice that, at this stage of the research, we deal with "tree-like" ontologies defined as class taxonomies induced by the IS-A relationship. This will allows us to exploit during query processing the pre-order and post-order properties of trees in order to enumerate the nodes and check ancestor-descendant relationships between the classes; such codes are displayed in the upper left corner of the ontology classes in the Figure, in the form: (**pre-order**,post-order). For instance, the class "Myocardial ischemia" has pre-order "3" which is also its identifier, whereas its post-order is "6". Before the personalization engine can be used to build a guideline version tailored to a specific patient, the patient must be classified with respect to the disease ontology, on the basis of medical records by means of a suitable reasoning service [13], or through a profile explicitly supplied by the physician. Moreover, additional semantic versioning coordinates, referencing specific domain ontologies, can also be considered to model context-dependent applicability of guidelines.

Hence, in XML resource repositories, reference to ontology concepts (e.g. using class identifiers like those in Fig. 2) can be added to the resource representation and storage as a new versioning coordinate. In this way, applicability annotations can be embedded in the guideline documents to be used by automatic personalization tools. Obviously, also the annotation of clinical guidelines which defines their semantic versioning must be effected by medical domain experts, as part of the guideline drafting and approval process itself. Whenever an ontology definition is changed, temporal versions of the ontology also must be maintained, as the temporal perspectives for navigating the ontology and for searching the guideline repository must be same for consistency reasons. The ontology temporal versioning techniques introduced in [15] can be used to this purpose.

One of the global effects of versioning is an increase in the number or size of the documents to be stored, also depending on the fact that different versions of the same document are stored as separate XML files or are arranged into a single multi-version XML file, owing to a uniform encoding of variant parts within the document structure. The latter solution, which is our choice, is often unavoidable in order to keep the growth of the storage space under control, especially when different versions of the same document may differ by a few nodes only. Personalization, which has shown to be a powerful tool to cope with information overload on the internet [25], can also be particularly effective when used in the management of large XML repositories of versioned documents [13]. In this case, the adoption of personalization techniques can prevent in most cases users to have to go through a huge amount of irrelevant information to find out the right version(s) of the one of interest and, thus, might help to make their search faster and more accurate. Hence, personalization based on semantic versioning may improve the quality of the interaction with the user by further focus-

ing the search on really relevant versions only, which is a desirable feature for clinical guideline management. For example, one of the acknowledged most relevant obstacles in the use and dissemination of guidelines [4] is the need for adapting them to constraints in local settings (e.g. concerning available hospital resources and practitioners' skills). Management of multi-version guidelines with context-based semantic personalization might help to overcome this problem [9, 28]. Other use cases requiring a sort of location-based semantic personalization can also be found: for instance, consider a guideline involving the recommendation of a new drug non yet registered in a given country, or introducing a new protocol only available in selected medical centers participating to an experimental program: the actual contents of the guideline should be changed according to the place where the guideline is retrieved or executed.

From the above discussion and examples, it is evident that the selection and reconstruction of the version(s) of interest for a user become a new challenging problem to support personalization in multi-version guideline retrieval.

---

RECOMMENDATIONS
1. IDENTIFICATION OF PATIENTS
    WITH RISK OF UNSTABLE ANGINA

    ...
2. INITIAL EVALUATION AND MANAGEMENT

    ...
3. EARLY HOSPITAL CARE
3.1. **Initial Treatment Strategy**

    ...
3.2. **Drug Therapy**
3.2(v1). ***Anti-Ischemic and Analgesic Therapy***
3.2(v1).1. Therapy with nitrates

        ...
3.2(v1).2. Therapy with beta-blockers
3.2(v1).2(v1).  *...administration of drug D1...*
3.2(v1).2(v2).  *...administration of drug D2...*
3.2(v1).2(v3).  *...administration of drug D3...*

        ...
3.2(v1).3. Therapy with ACE inhibitors

        ...
3.2(v2). ***Antiplatelet/Anticoagulant Therapy***

      ...
4. CORONARY REVASCULARIZATION

    ...
5. LATE HOSPITAL CARE

    ...

---

Fig. 3: The structure of a fragment of a sample multi-version clinical guideline.

## 3   An XML Data Model for Multi-Version Guidelines

In this Section, we introduce a multi-version XML document model supporting multiple temporal and semantic versioning coordinates. In doing this, we do not refer to a specific document structure (e.g. defined via a DTD or XML Schema), but we rather introduce a versioning annotation scheme which can be applied to any generic XML resource. In particular, it can be easily adapted to available proposals for the XML encoding of clinical guidelines, including those described in [3, 6, 26].

We start by formally defining as **version** a piece of text within a guideline document, with a common temporal and semantic pertinence. Owing to the definition, a version can be assigned a timestamp and an applicability annotation to uniquely define its temporal and semantic pertinence. Obviously, different versions of the same object must differ in their temporal and/or semantic pertinence.

For the sake of simplicity, but without loss of generality, we only consider in the examples which follow one time dimension (i.e. validity) and one semantic dimension (i.e. reference to classes in an ontology of diseases like the one in Fig. 2). Let us consider as running example the clinical guideline fragment in Fig. 3, involving recommendations for the treatment of unstable angina patients. The figure displays the text organization, which has a three-level section structure, where section 3.2. has two different versions, namely 3.2(v1) and 3.2(v2), whereas section 3.2(v1).2 has three different versions, namely 3.2(v1).2(v1), 3.2(v1).2(v2) and 3.2(v1).2(v3). The multi-version XML encoding of such guideline fragment is shown in Fig. 4.

In the XML encoding, we use the <version> element to delimit the boundaries of a version within the document. The <valid> and <applies> elements are then used to assign the temporal and semantic pertinence, respectively, to the version which contains them. Validity and applicability properties are inherited by descendant nodes in the XML tree-structure unless locally redefined with a new version definition. Therefore, there is no reason to repeat the valid or applies annotation when the pertinence is not changed from the ancestor version in the XML tree-structure. In general, redefinition may involve only a subset of the versioning dimensions, while the others dimensions are inherited.

With reference also to Fig. 3, the XML fragment in Fig. 4 shows, within the outermost <recommendations> element, a hierarchical structure based on three levels of sections. The <recommendations> element is composed of one version, which defines its global semantic and temporal pertinence, that is applicable to class C3 in the ontology in Fig. 2 (patients with myocardial ischemia) and valid from 1980 on. It is made of several first-level sections (see also Fig. 3), of which only section 3 is evidenced in the Figure. Such a section, made of only one version to specify applicability to ontology class C4 (patients with angina pectoris), deals with Early Hospital Care. Its temporal pertinence is inherited from the container element.

```
...
<recommendations>
<version number="1">
    <applies to="C3"/>
    <valid from="1980-01-01" to="9999-99-99"/>
    ...
    <section number="3">
    <version number="1">
        <applies to="C4"/>
        <title>Early Hospital Care</title>
        ...
        <section number="2">
        <title>Drug Therapy</title>
        <version number="1">
            <applies to="C5"/>
            <title>Anti-ischemic and Analgesic Therapy</title>
            ...
            <section number="2">
            <title>Therapy with beta-blockers</title>
            <version number="1">
                <valid from="1980-01-01" to="1990-12-31"/>
                    ...administration of drug D1...
            </version>
            <version number="2">
                <valid from="1991-01-01" to="1998-12-31"/>
                <valid from="2001-01-01" to="2003-12-31"/>
                    ...administration of drug D2...
            </version>
            <version number="3">
                <valid from="1985-01-01" to="9999-99-99"/>
                    ...administration of drug D3...
            </version>
            ...
            </section>
            ...
        </version>
        <version number="2">
            <applies also="C7"/>
            <title>Antiplatelet/Anticoagulant Therapy</title>
            ...
        </version>
        ...
        </section>
        ...
    </version>
    ...
    </section>
    ...
</version>
</recommendations>
...
```

Fig. 4: An XML fragment showing the multi-version encoding of the guideline in Fig. 3.

In general, by means of redefinitions we can introduce, for each part of a document, complex validity and applicability properties including extensions or restrictions with respect to ancestors. For instance, the applicability assignment to section 3 which we just described is a restriction and the attribute *to* is used to this end. Actually, the applicability assigned to the version is the intersection of the *to* value and of the value inherited by the ancestor version (in this case C4∩C3, which equals C4 since it is a subclass of C3). The same applies to second-level section 3.2 (entitled "Drug Therapy"), whose first version (entitled "Anti-ischemic and Analgesic Therapy") applies to class C5 (unstable angina), which is also a restriction, whereas the second version (entitled "Antiplatelet/Anticoagulant Therapy") is also applicable to class C7 (myocardial infarction), which is an extension indeed. Attribute *also* is used in this case, and the applicability assigned to the version is the union of the *also* value and of the value inherited by the ancestor version (class C4∪C7). In other words, the contents of section 3.2(v2) both apply to angina pectoris and myocardial infarction patients.

The third-level section 3.2(v1).2 entitled "Therapy with Beta-blockers" is made of several versions, each one dealing with the administration of a specific drug and having its own temporal pertinence, whereas the (inherited) applicability is the same (namely C5, unstable angina). In order to derive the validities of the three drugs shown in the Figure, we assume the recommendations underwent the evolution which follows. Drug *D1* was introduced in 1980 and then replaced by the drug *D2* in 1991. However, the use of drug *D2* was suspended from 1999 to 2000, period during which it had been under investigation since suspected of causing adverse reactions. In 2004, due to evidence of long-term adverse effects, *D2* was definitely withdrawn. Drug *D3* has been introduced in 1985. Hence, the resulting history of recommended beta-blockers according to the guideline in Fig. 3 (which will in fact correspond to the answers to a sequence of snapshot queries issued on the multi-version document) is the following:

- from 1980 to 1984: drug *D1*
- from 1985 to 1990: drugs *D1* and *D3*
- from 1991 to 1998: drugs *D2* and *D3*
- from 1999 to 2000: drug *D3*
- from 2001 to 2003: drugs *D2* and *D3*
- from 2004 on: drug *D3*

As for 3.2(v1).2(v2) in the Figure, versions can be assigned multiple intervals as validity: this corresponds to adopt *temporal elements* [10, 20], that is disjoint union of intervals, as timestamps.

## 3.1 Manipulation and Retrieval Operations

The multi-version XML data model can be equipped with two basic operators for the management of guideline authoring and maintenance: one devoted to change the textual content of a guideline portion and the other to allow modifications to the temporal and semantic pertinence of a given version. The former can be used for deletion of (a part of) the guideline (abrogation), or the introduction of a new part of the guideline (integration), or the replacement of (a part of) the guideline (substitution). The latter can be used to deal with the time/applicability extension or restriction of (part of) the guideline. Such operators, in order to preserve the well-formedness of the version structure

```
FOR    $a IN guidelines.xml
WHERE  textConstr ($a//section/title/text(), 'anticoagulant')
AND    tempConstr ('vTime OVERLAPS PERIOD('2007-01-01','2008-12-31')')
AND    applConstr ('C5')
RETURN $a
```

Fig. 5: An XQuery-equivalent query executable on a clinical guideline personalization system.

and the inheritance semantics, can be defined in a similar way as the ones defined for multi-temporal norm documents in [14].

Clinical guideline repositories, like the US National Guideline Clearinghouse [23] or the UK National Library of Guidelines [24], are usually managed by traditional information retrieval systems where users are allowed to access their contents by means of keyword-based queries expressing the subjects they are interested in. Adopting a system like the one described in [13] that we developed for norm documents, users are offered the possibility of expressing temporal and semantic specifications for the reconstruction of a consistent version of the retrieved guideline.

In particular, the queries can contain four types of constraints: temporal, structural, textual and applicability. Such constraints are completely orthogonal and allow the users to perform very accurate searches in the XML guideline repository. Let us focus first on the applicability constraint. Consider again the ontology in Fig. 2 and guideline fragment in Fig. 4: for the treatment of John Smith, an "infarctuated" patient (i.e. belonging to class C7), the sample recommendations in Fig. 3 will be selected as pertinent, but only the second version of Section 3.2 will be actually presented as applicable. Furthermore, the applicability constraint can be combined with the other three ones in order to fully support a multi-dimensional retrieval. For instance, a physician (or an health insurance officer) could be interested in all the guidelines ...

- ... which have a section whose title (*structural constraint*) contains the word anti-coagulant (*textual constraint*), ...
- ... which were valid between 2007 and 2008 (*temporal constraint*), ...
- ... and which are applicable to a patient suffering from unstable angina (*applicability constraint*).

More precisely, the system is able to answer queries having the XQuery [32] FLWR syntax in Fig. 5, where textConstr , tempConstr , and applConstr are suitable functions allowing the specification of the textual, temporal and applicability constraints, respectively (the structural constraint is implicit in the XPath expressions used in the XQuery statement).

## 4  Prototype Implementation

The personalization engine in Fig. 1, which is capable to execute queries like the one in Fig. 5, has been implemented as a prototype Multi-version XML Query Processor. The

prototype code is written in Java JDK 1.5 and employs ad-hoc data structures (relying on embedded "light" DBMS libraries) and algorithms which allow users to reconstruct on-the-fly the desired personalized version of the XML guideline, satisfying temporal, structural, textual and applicability constraints, by means of a multi-version extension of the *holistic twig join* approach [2]. Guidelines are stored in the XML repository using an indexing scheme based on *multi-version inverted indices*, that is an extension with timestamps and semantic annotations of the indexing solution proposed in [33]. In practice, the query processing algorithm implements the temporal slice operator proposed in [21], to which the processing of semantic constraints has been added, without an appreciable overhead. In fact, thanks to the properties of the adopted pre- and post-order encoding of the ontology classes, applicability constraints can be very efficiently tested during query processing by means of simple comparisons. A detailed presentation of the deployed data structures and holistic join techniques, together with a related work discussion on these topics, can be found in [12].

As a result, we obtain a high overall query processing efficiency mated with low memory requirements. In order to evaluate the performance of the prototype, a specific query benchmark was built and several exploratory experiments were conducted to test the personalization engine behavior under different workloads. The experiments have been effected on a Pentium 4 3Ghz Windows XP Professional workstation, equipped with 1GB RAM and a 160GB EIDE disk with NT file system (NTFS). Test were performed on three XML document collections of increasing size (namely 5,000, 10,000 and 20,000 guidelines, with a total size of 120MB, 240MB and 480MB, respectively). In all collections the guidelines were synthetically generated by means of a suitable tool, which is able to produce XML documents compliant to our multi-version model under different parameter configurations. For each collection, the average, minimum and maximum document size was 24KB, 2KB and 125KB, respectively. Experiments were conducted by submitting queries of five different types, mixing in various ways structural, textual, temporal and applicability constraints.

The system behavior showed a good efficiency in every context, providing a response time (including query analysis, retrieval of the qualifying guideline parts and reconstruction of the result) of a few seconds for most of the queries. Moreover, the selectivity of the query predicates does not impair performances, even when large amounts of documents containing some (typically small) relevant portions have to be retrieved. The system is able to deliver a fast and reliable performance in all cases, since it practically avoids the retrieval of useless document parts. For the same reasons, the main memory requirements of the Multi-version XML Query Processor are quite limited, less than 5% with respect to an approach like the one adopted in [14], where complete documents are retrieved with a traditional XML engine working on structural and textual constraints, and then temporal and applicability constraints are applied using a DOM representation to prune out non-qualifying XML nodes. Notice that this property is very interesting for a system which is likely to run in a highly concurrent multi-user environment, since memory requirements are not crucial for performance. The prototype system also showed a good scalability behavior in every type of query setting, as the computing time for the same query always grows linearly with the number of documents. Full details on performance evaluation can be found in [12, 13].

## 5   Conclusions and Future Work

In this paper, we applied to the representation and management of clinical guidelines some techniques we previously developed for norm documents in the legal domain [12, 13]. In particular, we introduced solutions to model and to provide personalized access to multi-version guidelines, supporting multiple temporal and semantic versioning coordinates. The proposal involves the definition of a multi-version XML data model and the implementation of a prototype personalization engine.

Preliminary experimental work on query performance, with repositories of syntectic XML documents, showed encouraging results. In particular, the personalization engine proved to be very efficient in a large set of experimental situations and showed excellent scale-up figures with varying load configurations.

We underline that the very same techniques we presented for personalized access to multi-version textual guideline documents can also be applied to the enactment of workflows implementing multi-version clinical guidelines, provided that workflows are specified using an XML-based definition language, like BPEL [29] or XPDL [31], which can be enriched as well with temporal and semantic annotations in order to define versions [13].

Future work will consider the improvement of the approach to cope with more advanced application requirements (e.g. relaxing of constraint of tree-like ontologies) and the completion of the technological infrastructure required to set up the personalization platform with the design and implementation of auxiliary services (e.g. for automatic patient classification with respect to the disease ontology). Further work will also include the assessment of our developed system in a concrete working environment, with real users and in the presence of a repository of real clinical guidelines.

## References

1. T. Berners-Lee, J. Hendler, and O. Lassila. The Semantic Web. *Scientific American*, 284(5):34–43, 2001.
2. N. Bruno, N. Koudas, and D. Srivastava. Holistic Twig Joins: Optimal XML Pattern Matching. In *Proc. of SIGMOD 2001*, pages 310–321, 2001.
3. D. Buchtela, J. Peleška, A. Veselý, J. Zvárová, and M. Zvolský. An XML-based Format for Guideline Interchange and Execution. In *Proc. of MIE 2008*, pages 151–156, 2008.
4. M.D. Cabana, C.S.N.R. Rand, C.S. Powe, A.W. Wu, M.H. Wilson, P.C. Abboud, and H.R. Rubin. Why don't Physicians follow Clinical Practice Guidelines? A Framework for Improvement. *Journal of American Medical Association*, 282(15):1458–1465, 1999.
5. C. Combi and A. Montanari. Data Models with Multiple Temporal Dimensions: Completing the Picture. In *Proc. of CAiSE 2001*, pages 187–202, 2001.
6. A.K. Dubey and H.C. Chueh. An XML-based Format for Guideline Interchange and Execution. In *Proc. of AMIA 2000*, pages 205–209, 2000.
7. C.E. Dyreson and F. Grandi. Temporal XML. In M.T. Özsu and L. Liu, editors, *Encyclopedia of Database Systems*, pages 3032–3035. Springer-Verlag, 2010.
8. D.B. Fridsma. Special Issue on Workflow Management and Clinical Guidelines. *AIM Journal*, 22(1):1–80, 2001.
9. D.B. Fridsma, J.H. Gennari, and M.A. Musen. Making Generic Guidelines Site-specific. In *Proc. of AMIA 1996*, pages 597–601, 1996.

10. S.K. Gadia. A Homogeneous Relational Model and Query Languages for Temporal Databases. *ACM Trans. on Database Systems*, 13(3):418–448, 1988.
11. C. Gordon and J.P. Christensen. *Health Telematics for Clinical Guidelines and Protocols*. IOS Press, 1995.
12. F. Grandi, F. Mandreoli, and R. Martoglia. Issues in Personalized Access to Multi-version XML Documents. In E. Pardede, editor, *Open and Novel Issues in XML Database Applications*, pages 199–230. Information Science Reference, 2009.
13. F. Grandi, F. Mandreoli, R. Martoglia, E. Ronchetti, M.R. Scalas, and P. Tiberio. Ontology-based Personalization of e-Government Services. In C. Mourlas and P. Germanakos, editors, *Intelligent User Interfaces*, pages 167–203. Information Science Reference, 2009.
14. F. Grandi, F. Mandreoli, and P. Tiberio. Temporal Modelling and Management of Normative Documents in XML Format. *Data & Knowledge Engineering*, 54(3):327–354, 2005.
15. F. Grandi and M.R. Scalas. The Valid Ontology: A Simple OWL Temporal Versioning Framework. In *Proc. of SEMAPRO 2009*, pages 98–102, 2009.
16. T. Gruber. Ontology. In M.T. Özsu and L. Liu, editors, *Encyclopedia of Database Systems*. Springer-Verlag, pages 1963–1965, 2010.
17. N. Guarino, editor. *Formal Ontology in Information Systems*. IOS Press, 1998.
18. E.A. Hutchinson, and R. Baker, editors. *Making Use of Guidelines in Clinical Practice*. Radcliffe Medical Press, 1999.
19. International Statistical Classification of Diseases and Related Health Problems. World Health Organization, `http://www.who.int/classifications/icd/en/`, 2010.
20. C.S. Jensen and C.E. Dyreson et al. The Consensus Glossary of Temporal Database Concepts - February 1998 Version. In O. Etzion, S. Jajodia, and S. Sripada, editors, *Temporal Databases — Research and Practice*, pages 367–405. Springer-Verlag, 1998.
21. F. Mandreoli, R. Martoglia, and E. Ronchetti. Supporting Temporal Slicing in XML Databases. In *Proc. of EDBT 2006*, pages 295–312, 2006.
22. Medical Subject Headings - Section c: Diseases. US National Library of Medicine, `http://www.nlm.nih.gov/mesh/2009/mesh_browser/MeSHtree.C.html`, 2010.
23. National Guideline Clearinghouse. US Agency for Healthcare Research and Quality, `http://www.guideline.gov/`, 2010.
24. National Library of Guidelines. UK National Institute for Health and Clinical Excellence, `http://www.library.nhs.uk/GUIDELINESFINDER/`, 2010.
25. D. Riecken. Personalized Views of Personalization. *Communications of the ACM*, 43(8):27–28, 2000.
26. R.N. Shiffman, B.T. Karras, A. Agrawal, R. Chen, L. Marenco, and S. Nath. GEM a Proposal for a More Comprehensive Guideline Document Model using XML. *Journal of AMIA*, 7(5):488–497, 2000.
27. P. Terenziani, S. Montani, A. Bottrighi, G. Molino, and M. Torchio. Clinical Guidelines Adaptation: Managing Authoring and Versioning Issues. In *Proc. of AIME 2005*, pages 151–155, 2005.
28. P. Terenziani, S. Montani, A. Bottrighi, M. Torchio, G. Molino, and G. Correndo. A Context-adaptable Approach to Clinical Guidelines. In *Proc. of MEDINFO 2004*, pages 169–173, 2004.
29. The Web Services Business Process Execution Language. WfMC Coalition, `http://www.wfmc.org/standards/docs.htm`, 2010.
30. The eXtensible Markup Language home page. W3C Consortium, `http://www.w3c.org/XML/`, 2010.
31. The XML Process Definition Language. OASIS Organization, `http://www.oasis-open.org/committees/tc_home.php?wg_abbrev=wsbpel`, 2010.
32. The XML Query home page. W3C Consortium, `http://www.w3c.org/XML/Query`, 2010.

33. C. Zhang, J. Naughton, D. DeWitt, Q. Luo, and G. Lohman. On Supporting Containment Queries in Relational Database Management Systems. In *Proc. of SIGMOD 2001*, pages 425–426, 2001.