The “Norme in Rete”- project: Standards and Tools for Italian Legislation

E. Francesconi
ITTIG - CNR – via de’ Barucci 20, Florence - Italy
http://www.ittig.cnr.it
francesconi@ittig.cnr.it

Abstract

The “Norme in Rete” (NIR) project aims at establishing standards for Italian legislation and tools to promote their adoption. In this paper an overview of these standards, including their latest version features, as well as a description of the tools developed by ITTIG-CNR for their implementation are presented.

1 Introduction

Fragmentation of legislative information and inconsistencies of different legal document formats represent historical obstacles to a systematic organization of a normative corpus. In Italy we faced a scenario developed from centralized architectures in the 80’s, when few content holders were able to manage and distribute legal documents by charge (Court of Cassation, the Official Journal and publishers), to the uncontrolled distribution of legal contents in the 90’s (public bodies (Central and Local PAs, Judicial offices) and private publishers used to distribute legal contents for free or by charge). The main drawback of such architectures was represented by the lack of a standard description of legal documents able to encourage the development of legal information systems with characteristics of interoperability and effective of use. A standard for legal documents, in fact, allows to improve their quality, as well as to define a common ground to build up legal documents access facilities for both citizens and legal experts.

In order to overcome such problems at national level, in Italy the “Norme in Rete” (NIR) project was launched in 2001. It was proposed by CNIPA [Italian National Center for Information Technology in the Public Administration] in conjunction with the Italian Ministry of Justice: it aims at defining standards for the Italian legislation, allowing the creation of a unique access point for legal documents in a distributed environment with search and retrieval facilities, as well as a mechanism of stable cross-references able to guide users towards relevant sites of public authorities adhering to the project. To achieve these purposes, the NIR project proposed the adoption of XML as a standard for representing legal documents, defined by using three DTDs with increasing degree of complexity (Megale and Vitali 2001): they aim at representing a legal text with respect to its structural or formal profile and to its semantic or functional profile using particular meta-information. Moreover a standard to univocally identify legal measures based on URN technique has been defined (Spinosa 2001): it is able to provide a stable system of cross-referencing within the NIR domain.

The Institute of Legal Theory and Techniques of the Italian National Research Council (ITTIG-CNR) in Florence has been having a relevant role in the national working groups which established NIR standards, along with the University of

Bologna and national institutions as the Italian Parliament (the Senate and the Deputies Chamber), the Ministry of Justice, the Court of Cassation.

In order to make the adoption of such standards easier, ITTIG-CNR developed a number of tools within the project. The main one is *NIREditor*, an authoring tool which includes facilities, based on previous studies on legislative drafting (Biagioli 1992), and modules which aim at managing new or legacy law documents according to the established standards.

This paper is organized as follows: in Section 2 the standards established within the NIR project are introduced; in Section 3 the NIR semantics description of legislative documents is discussed; in Section 4 the main features of the new versions of NIR standards are summarized; in Section 5 the main tools (particularly NIREditor) developed by ITTIG within the project are presented. Finally, in Section 6 some conclusions are discussed.

2 The NIR standards

The NIR project proposed the adoption of XML as a standard to represent legal documents. Two specific national work groups produced two main official standards.

A first group, coordinated by Spinosa of ITTIG-CNR in Florence, aimed at defining a standard for legal document identification, defined according with the uniform name (URN) technique: an unambiguous identifier, which allows the references to be expressed in a stable way, irrespective of their physical location (Spinosa 2001).

A second group, coordinated by Vitali of University of Bologna, aimed at defining a standard for legal documents, formulated by defining XML-DTDs (NIR-DTDs) of increasing degree of complexity in text hierarchy description for different kinds of legal documents (Megale and Vitali 2001).

2.1 The NIR-URN Standard

Within the NIR domain, documents are identified through uniform names. Uniform Resource Names (URNs) were conceived for providing unambiguous and lasting identifiers of network resources, independently of their physical locations. This technique appears extremely useful in domains, as the legal one, where references to other measures are particularly important. The use of hypertext links on the Web based on Uniform Resource Locations (URLs) in fact allows to express references, providing also an effective retrieval systems, but do not appear to be suitable for wide-scale use in the law. References based on physical locations, expressed through URLs, in fact presents the following well-known problems (Spinosa 2001):

- difficulty in knowing the location of the cited resource;
- loss of validity over time of the document locations;
- impossibility of describing references to the resources not published yet.

In order to avoid these problems, a system of references based on assigning a uniform name (URN) to each legal resource and on a resolution mechanism (RDS: Resolver Discovery Service) able to retrieve the corresponding object has been chosen. The URN-NIR standard has been established in conformity with those defined within IETF (Internet Engineering Task Force) by the URN Working Group.

The uniform name system of the domain of interest must include:

- a schema for assigning names capable of representing unambiguously any legal measure, issued by any authority at any time (past, present and future);
- a resolution mechanism from uniform name to on-line location of the corresponding resources.

For NIR documents the URN syntax has been defined according to RFC 2141 *URN Syntax* (Moats and Sollins 1997): it includes a name-space identified by “nir” (this identifies the context in which the names are valid) and a syntax to represent measure details:

```
<URN> ::= "urn:nir:" <NSS-nir>
```

The specific name `<NSS-nir>` must contain information to unambiguously identify a document. In legislative environment, it is also necessary to distinguish among any later versions of a document and among its amendments. In this case information regarding a specific version of a document is added. The general structure of a specific URN-NIR is therefore:

```
<NSS-nir> ::= <document> ["@" <version>]
```

The `<document>` part is composed by document information related to the enacting authority, the type of measure, the details and any annex:

```
<document> ::= <authority> ":" <measure> ":" <details>
                        [":" <annex>]
```

The complete syntax specification of the uniform names belonging to the “nir” name-space can be seen in (Spinosa 2001); examples of uniform names of legal documents are here below reported:

Act no. 675 of 31 December 1996
`urn:nir:stato:legge:1996-12-31;675`

Decree of Ministry of Finance of 20.12.99
`urn:nir:ministero.finanze:decreto:1999-12-20;nir-3`

AIPA circular of 21 June 2001, No. 31
`urn:nir:autorita.informatica.pubblica.amministrazione:circolare:2001-06-21;31`

Decision of the Italian Constitutional Court No.7 of 23 January 1995
`urn:nir:corte.costituzionale:sentenza:1995-01-23;7`

As discussed, the NIR uniform name assigned to legal document depends on the characteristics of the document itself, therefore it is independent from the availability of the document, from its physical location and the access mode. Only the significant details of the document and the knowledge of the URN syntax are necessary to its identification.

Within the NIR domain, URN technique is used to:

1. represent cross-references to other legal measures;
2. navigate through references and retrieve legal measures of the NIR domain;
3. represent relationships among legal documents.

The association between uniform names and documents can be obtained by using meta-information inserted in the document itself (ex: in HTML files using a META element, in XML files using a suitable element) or maintained outside the documents but strictly related to it (by specific attribute in a database, or using growing methods as RDF technology).

In order to make effective the use of URNs in hyperlinks, a database associating the document URNs to one or more physical location of them is necessary. This process is known in literature as "resolution mechanism": it can be centralized or, hopefully, distributed according to the DNS resolution system over the Internet. Such database can also contain other meta-information (details, title, subject, relations among the acts, etc.) which enrich the system response. Besides acting as a resolution system, this database gives also the possibility to to provide an answer to the user, even in case of uncompleted or uncorrected uniform names, derived from wrong citations (for example the resolution service gives back the list of the documents whose URNs partially match the provided URN, or it attempts to correct automatically the URN itself).

The URN technique of representing references allows the development of parsing tools (see Section 5.1.1) able to automatically detect references and construct the related URNs on the basis of the significant details of the referred document (expressed in natural language by the citations) and the knowledge of the URN syntax only. Moreover, it allows to construct a knowledge base storing the relations among legal documents.

2.2 The XML-NIR Standards

As well as the URN-NIR standard, the NIR project has defined a standard based on XML, aimed at describing the content of legislative documents. For this purpose three DTDs with increasing degree of complexity have been established:

- the "DTD flessibile" (niloose.dtd) contains about 180 elements: it does not establish any mandatory rules (unless in a very small quantity) and it is used for legacy legal documents not following drafting rules;
- the "DTD base" (nirlight.dtd) contains about 100 elements: it represents a subset of the "DTD completo": it is useful to train users in adopting the DTD standards;
- the "DTD completo" (nirstrict.dtd) contains about 180 elements: it follows legislative drafting rules and it is used to write new legal texts.

The "DTD flessibile" and "DTD completo" are composed by four common files:

1. global.dtd: containing general definitions;
2. norme.dtd: containing definitions of the division structures;
3. text.dtd: for text, table and form structure definitions;

4. meta.dtd: containing metadata schemes definitions.

Differences are present in the main files nirstrict.dtd and nirloose.dtd. The nirstrict.dtd establishes an order to the partitions of a law text. Collections of articles are still considered the basic elements of the norm (their numbering is independent from the hierarchical organization of the other elements). Numbering of the divisions is mandatory. Titles of the divisions are not provided, while they are optional for the other elements. The nirloose.dtd establishes only few constraints and it is used for legacy legal documents which usually do not follow particular legislative drafting rules.

Basically NIR-DTDs allow legal documents to be described using two main kinds of elements:

1. Structural elements;
2. Metadata.

Structural elements can be divided into:

- Generic document elements: references to other measures, formatted text-embedded relevant entities (tables, lists, etc.);
- Specific legal document elements: heading, preamble, sections, articles, paragraphs, etc.

Structural elements describe the form of a legal text (*formal profile*).

Similarly NIR-XML standards consider two kinds of metadata:

- General metadata: subject classification, publication date, relationships among acts;
- Analytical metadata (*analytical provisions*): they consist in *provisions* types (Amendments (Insertion, Abrogation, Substitution), and Rules (as Obligation, Definition, Penalties, etc.)) and by their *arguments* (for example the *addressee* of an Obligation);

Analytical metadata have been proposed in the '90 by Biagioli of ITTIG-CNR (Biagioli 1997) and a "provision-centric" semantic description of a legal text has been received as NIR standards.

While general metadata provide general information on the act, analytical metadata describe the semantics of the provisions (*functional profile*). The detection of the functional profile of a law text consists in describing the provision it contains through a model made of provision types (*regulative profile*) and their arguments (*thematic profile*). The first one reflects the lawmaker directions, the second one the peculiarities of the regulated field. The formal profile represents the traditional habit of organizing legal texts in chapters, articles, paragraphs, etc.; on the other hand the functional profile is related to how the semantics of the text is organized. The functional profile is traditionally described by the legislator by assigning titles to formal partitions: partition titles are nothing but *ante-litteram* metadata, therefore analytical provisions basically are a formalized version of these titles and their NIR-XML version is the way how they are adopted within the NIR project.

3 The Semantics of Provisions

Analytical provisions describe of the *functional profile* of a legislative text; they are formalized in a model (Biagioli 1997) whose structure has been included as NIR standard. As introduced in Section 2, provision types provided by NIR standards are divided into two main groups: Amendments and Rules.

Amendments can be:

- *content amendments*: they modify literally the content of a norm or as regards the meaning without literal changes;
- *temporal amendments*: they modify the times of a norm (come-into-force and efficacy time);
- *extension amendments*: they extend or reduce the cases on which the norm operates.

Amendments can have acts or norms as arguments.

Rules are provisions which aim at regulating the reality considered by the including act. Adopting a typical law theory distinction, well expressed by Rawls, they consist in:

- *constitutive rules*: they are mainly rules on entities of the regulated reality. They consist basically in those ones introducing entities and those ones which assign a juridical profile to the entities (“empowering norms”);
- *regulative rules*: they are mainly rules on actions. They consist in those ones disciplining actions and those ones which discipline the substantial and procedural defaults (“remedies”).

4 New versions of the NIR standards

A new phase of the project has been launched in 2004. It aims at defining new versions of both URN and DTDs in order to cope with new needs risen from the use of the first version of the standards and from discussions within the NIR standards working groups.

As regards URN standard, the URN-NIR group has worked on new versions of the uniform name grammar (v. 1.3) where an extended space, to be used only for cross-references, is proposed.

Basically the proposal aims at including the possibility to identify each single partition of a legal document, so that they can be referenced not only using the HTML anchor within a document (by the use of the # character, that cannot be transmitted to a Web server), but also referred independently from the document they belongs to (this is important when partition of a legal text are organized in different documentary units).

Further new features are represented by the inclusion of the possibility to express the characteristics of a reference, not explicitly expressed in the textual form of the citation (for example the possibility to distinguish *static* references (to historical texts) from *dynamic* ones (in-force texts) of the same act). Using this extended space (syntax) it is possible to refer a specific version of an act without knowing its version identifier, but simply expressing the interval time to which the referred document is to be considered.

As regards XML standards, the XML-NIR group has worked on a new version of the NIR-DTDs (v. 2.0). In this second version, the joint work of ITTIG - CNR and CIRSFID department of the University of Bologna introduced main news on metadata. The new version of general metadata includes the possibility to insert some relevant information which is not included in the original text, but which can be derived from different fonts (as the nature of the act, the measure rank: it can be of constitutional rank, of different level (primary, secondary, etc.) or derived from praxis), the title of the measure in case it is not included in the original documents, and so on.

Important news are also proposed by Palmirani of CIRSFID to describe the dynamics of legal texts (Palmirani 2005) ("in force" and "efficacy" times to be applied to provisions and the related norms respectively).

5 Tools for NIR standard implementation

In order to make easier the adoption of NIR standards, a number of tools have been developed for their automatic or semi-automatic implementation. They are aimed to handle legacy contents and to produce new legal documents according to the NIR standards. Legacy contents modules (Section 5.1) are able to manage the formal structure of legal texts and to extract their semantics. To allow the production of new legal texts according to the NIR standards, a specific editor (*NIREditor*) has been developed. It includes the modules able to manage legacy contents and it is able to work on native XML-NIR and URN-NIR formats.

5.1 Legacy content handling

A particular attention has been addressed to design automatisms for legacy content handling, since they represent key-factors for promoting the adoption of the standards. Four modules have been implemented:

1. the *Cross-Reference Parser*, designed to detect cross-references and to construct the related URNs;
2. the *Structure Parser*, designed to automate the XML-NIR conversion of legacy contents;
3. the *Provision Automatic Classifier*, which automatically classifies paragraphs into provisions according to the NIR provision scheme (Biagioli et al. 2005a);
4. the *Provision Argument Extractor*, which automatically identifies the arguments of the provisions (Bartolini et al. 2004).

The first two modules are able to detect the formal profile of a legal text, producing its XML-NIR description. The last two modules are able to detect the functional profile of a legal text, producing its consequent XML-NIR semantic annotation.

5.1.1 The Cross-Reference Parser

A legal text may contain lots of cross-references to other measures that have to be described using the related URN, so that references can be transformed in effective links when documents are published on the Web. Information to build URNs are usually contained in the citation (for example the citation: "Act 24 November 1999, No. 468" generates the following URN-NIR "`urn:nir:stato:legge:1999-11-24;468`").

Especially in the phase of legacy content conversion, the manual construction of a URN for each reference can be a time-consuming work. For this reason a module able

to automatically parse a legal document, detecting cross-references and assigning them the related URNs has been developed.
The parser is generated using LEX and YACC technologies (Lesk 1975), (Johnson 1975), on the basis of the vocabulary of the citations and the URN grammar expressed in EBNF (see Section 2.1).

Using the LEX technology a lexical analyzer is generated able to detect *tokens*, namely symbols (words, numbers and punctation marks) belonging to the citation vocabulary. Then using the YACC technology, a syntactical analyzer is generated able to recognize a sequence of tokens, generated by LEX, as representing a reference, and to construct the related URN.

5.1.2 The Structure Parser

As previously introduced, the structure parser is able to transform a legacy content into its XML-NIR representation. So far the expected document native formats are HTML and plain text; other proprietary formats are being considered.

To obtain the automatic conversion of legacy legal documents from their native format to XML-NIR, two parsing strategies have been adopted for different portions of a legal text.

For the body of the text a non-deterministic finite-state automaton (NFA) has been implemented. The NFA defined to parse the body of a legal text is a 5-tupla (VT, VN, N, R, I, Z) where:

- VT is the vocabulary, namely the set of symbols to be considered (in our case words, numbers and punctation marks);
- VN is a set of states, representing the NIR elements to be described by the XML-NIR mark-ups;
- $R = VN \times (VT \cup \{\varepsilon\}) \rightarrow 2^{VN}$ is the set of transitions among the states, which correspond to formal rules of document partition separation (ε is an empty symbol representing the possibility to walk from a state to another without scanning any symbol; 2^{VN} formally represents the set of all the subsets of VN: in fact, considering a non-deterministic automaton, starting from one state you can reach more than one destination states);
- $I \in VN$ is the initial state;
- $Z \subseteq 2^{VN}$ is the set of final states.

Parsing a document according to the automaton model, different formal portions of a legal text can be mapped to the related NIR-DTDs elements.

For the header and the footer of a legal text a different strategy has been adopted. Header and footer in fact are not usually characterized by particular typographical symbols separating formal partitions, corresponding to as many NIR elements. The identification of such elements can only be based on the sequence of words appearing with a probability that can be estimated and without knowing the states which produced such sequence. The aim of this approach is to uncover these hidden states. This is a typical problem that can be represented by Hidden Markov Models (HMMs), basically probabilistic automata where states are inaccessible.

According to (Rabiner 1989), each HMM designed to parse the header and the footer of a law text consists of:

- a set of states $S = \{s_1, s_2, ..., s_N\}$ in the model, corresponding to the logical meanings of each word in legal texts, and whose combinations represents the formal elements of the NIR-DTDs (for example the states s_i = `day`, s_j = `month`, and s_k = `year`, when combined, represent the element *date* of the NIR-DTDs);
- a set of observation symbols $V = \{v_1, v_2, ..., v_M\}$ per state, in our case corresponding to as many words that can be emitted at each state;
- $A = \{a_{ij}\}$, the state transition conditional probability distribution, which specifies, for each $s_i, s_j \in S$, the probability to go from state s_i to state s_j where:

$$a_{kl} = P[q_t = s_i \mid q_{t-1} = s_j] \tag{1}$$

  being q_t the state at time t;
- the observation symbol conditional probability distribution $B = \{b_j(k)\}$, specifying, for each $v_k \in V$ and $s_j \in S$ the probability to observe the symbol v_k being in state s_j (emission probability), where:

$$b_j(k) = P[v_k \text{ at } t \mid q_t = s_j],\ 1 \le j \le N,\ 1 \le k \le M \tag{2}$$

Once the model has been constructed, the *Viterbi Algorithm* (Viterbi 1967) allows to obtain the single best state sequence for a given observation sequence. Using this algorithm the observation of symbol emissions allows to walk forward on the probabilistic automaton, following all the available paths with a certain probability at each step. At the end of this process a backward procedure allows to identify the single path with the highest probability, so to identify the sequence of states producing the observed sequence (in our case to reconstruct the formal structure of a portion of a legal text, given a sequence of symbols (words, numbers and punctation marks).

5.1.3 The Automatic Provision Classifier

As regards the automatic detection of the semantics in a legal text the *Provision Automatic Classifier* is designed to automatically classify paragraphs into provision types. Two machine learning approaches of document classification have been tested: *Naïve Bayes* and *Multiclass Support Vector Machines*. For a detailed discussion of the methodologies we tried and of the experimental results see (Biagioli et al. 2005a). Currently within *NIREditor* the two approaches can be alternatively used, but the approach based on *Multiclass Support Vector Machines* is recommended since it reported better results.

5.1.4 The Provision Argument Extractor

The *Provision Argument Extractor* is designed to automatically detect the arguments of a provision. Knowing the provision type detected by the *Provision Automatic Classifier*, this module uses the specific grammar of the provision to extract the provision arguments using NLP techniques. Basically the purpose of this module is to select relevant text fragments corresponding to specific semantic roles that are relevant for the different types of provisions. It is realized as a suite of Natural Language Processing tools for the automatic analysis of Italian texts (see (Bartolini et al. 2002)), specialized to cope with the specific stylistic conventions of the legal parlance. For a deeper discussion of this module see (Bartolini et al. 2004), (Biagioli et al. 2005a).

5.2 NIREditor

The NIR-DTDs identify a wide and complex subset of documents: basically legal measures and regulative acts. The production of new documents, as well as the transformation of legacy contents according to the NIR standards, can be a hard problem to face without an editing system guiding and supporting the user.

Even though programs for XML drafting already exist, they have limits whether used for a specific class of documents, especially as concerns the generality and inadequacy of their editing functions with respect to the needs to implement the NIR-DTDs constraints. The more complex an XML standard is, the more general-purpose XML editors have to be personalized.

Other possible solutions, as adapting Microsoft Word or Open Office to adhere to a specific XML standard, suffer the same limitations. An advantage is that users are more familiar with such tools, however when they are used to produce documents according to a specific XML standard, they have to be personalized as well, and the more complex the standard is, the more users make use of the specific editing functions related to the standard, discarding generic editing functions, even advanced. Moreover, these solutions suffer of another limitation: they handle documents according to the application specific proprietary format, therefore the syntactic rules contained in the DTDs of the standard under consideration have to be mapped to the proprietary format. For these reasons we have decided to develop a specific environment to handle XML-NIR documents in their native format.

As to produce HTML documents according to the HTML-DTD, specialized editors exist, similarly to help law texts drafting according to NIR-DTDs standards, a specialized visual editor (*NIREditor*) has been developed (Biagioli et al. 2003): it consists of a law drafting environment supporting specific Italian legislative technique functions. Similar initiatives exist at European level, as for example MetaLex (Boer et al. 2003). Metalex is a knowledge management system for legislative drafting: it aims at supporting users providing both content management and decision support components.

With respect to MetaLex, *NIREditor* is more focused at providing facilities for legal drafting with the aim of giving users a tool able to make the adoption of legal standards easier. The software architecture of *NIREditor* is represented by a kernel of Java specific functions library, fully integrated within the law drafting environment. they can also be integrated to the main XML general purpose editors supporting a Java API.

The NIREditor functional architecture has been designed by Biagioli, on the basis of previous studies on legal drafting (Biagioli 1992). The software architecture has been designed and developed by ITTIG computer science department (Biagioli et al. 2005b).

NIREditor operates within the URN-NIR and DTD-NIR frameworks in two working situations: it is designed to process legacy legal contents, as well as to assist the drafting of new texts. In both these two working situations *NIREditor* is designed to handle the formal as well as the functional profile of legal texts, using both manual and automatic facilities. In Sections 5.3 and 5.4 the main functions, proposed by Biagioli, dealing with the composition and the organization of new acts are described respectively.

5.3 The composition of new texts

For the composition of new texts, *NIREditor* is conceived as a visual editor, supporting the user in producing valid documents according to the NIR-DTDs. No XML validation is necessary within the editing environment (unlike general-purpose XML editors) since *NIREditor* allows the user to perform only valid actions. Moreover, it helps the user in composing particular sections of a new document using forms, and permits the introduction of the metadata provided by the NIR-DTDs.

The insertion of the XML formal partitions provided by the NIR-DTDs is guided by the editor which suggests the user the elements that can be introduced according to the context of the insertion point.

Particular facilities available within the drafting environment are the automatic numbering of the divisions and the updating of internal references in the event of text movements or variations. Automatisms are included as far as the construction of external and internal cross-references are concerned: using dialogue windows the construction of cross-references and the related URNs are possible, as well as the invocation of the *Cross-Reference Parser* on the whole document or on a selected text portion to automatically construct references and related URNs from their linguistic formulation.

Forms are widely used to guide drafter in composing specific portions of the document, providing also typical sentences depending on different type of acts, as for the header and the footer parts of a measure.

As with other word processors, it is possible to construct a new text by determining *a priori* the structure and insert the content afterwards (*top-down composition strategy*), or else text fragments can be inserted in no particular order, then organized and inserted into a suitable structure at a later time (*bottom-up composition strategy*). During the composition, a further step is represented by the application of the analytical metadata and their arguments to the divisions. This can be done by hand or using the *Provision Automatic Classifier* and the *Provision Argument Extractor* as a support. If metadata have been inserted the drafter can use them to be helped in determining the best structure of the text (further then being used to improve search and retrieval services in a legal document information system).

Moreover analytical metadata can be used to compose the titles (or headings) of the partitions since they usually are inserted according to the same criteria of analytical metadata, that is summing up the content of the partitions using meaningful terms.

5.4 The organization of new texts

For the organization of new texts, two alternative strategies can be followed, according to the formal or functional profiles of a legal text: the *formal organization strategy* and *the functional organization strategy* (Biagioli 1997), (Valente and Breuker 1997), (van Kralingen 1997).

The *formal organization strategy* considers the text according to the formal profile: the text is considered as made up of divisions (collection of articles). Using the formal strategy the partitions of similar rank to be grouped in a new partition are chosen explicitly by the draftsman. The editor will create a new partition of immediately higher rank, thus applying rules of formal text organization.

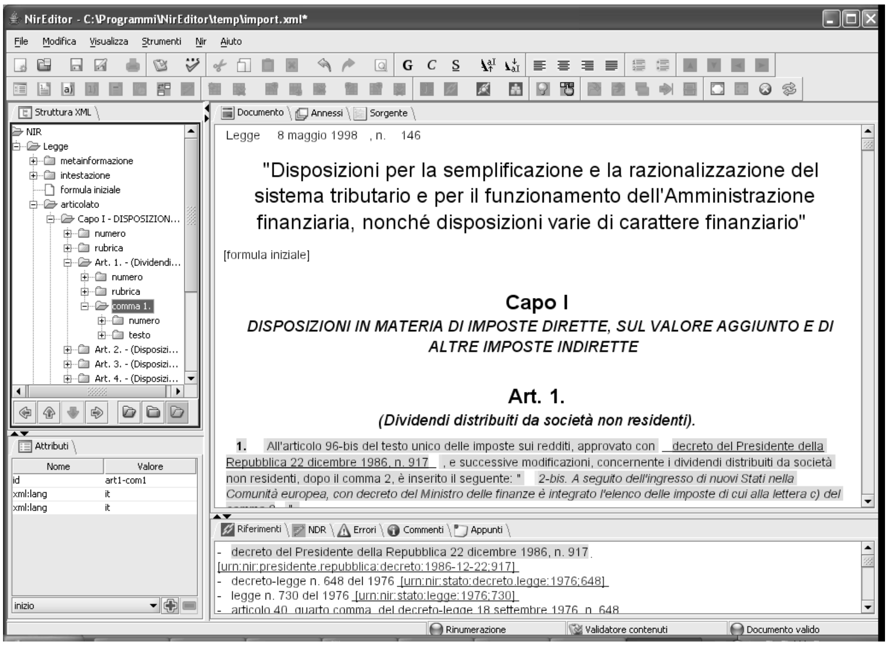
The *functional organization strategy* considers the text according to the functional profile, where the elementary component of a text is a *provision*. The draftsman carries out the same operations as in the formal strategy, but in an indirect way: the partitions to be grouped in a new one are chosen according to their content, affinities, etc. as well as it is decided where they should be placed in the text, according to the preferences of the drafter and the customary procedure of presentation (Biagioli 2000).

Basically, in this text organization strategy, the drafter chooses the partitions to be organized making queries on the analytical provisions (provision types, arguments and argument contents) associated to the partitions themselves.

5.5 xmLegesEditor

A new version of legislative editor implementing NIR standards has been released in 2005 with the name of xmLegesEditor. It has been developed on the basis of the experience matured on NIREditor, by ITTIG-CNR computer science department as well. The software architecture has been revised, aiming at stressing components modularity. xmLegesEditor includes the same functionalities of NIREditor as far as the legacy contents handling and the production of new documents are concerned. However new functions have been introduced: in particular the editing environment gives the user the possibility to manage particular document elements (as cross-references and notes) or sections (as attachments) in focused panes. New typologies of documents can also be managed, in particular it is possible to draft new bills according to the related NIR-DTD, released in a preliminary version, to handle multi-version bills including chambers amendments, as well as producing the working version of the text for parliamentary procedures, including a comparison between two different versions of a document before and after chambers amendments. As a prototype function the automatic production of the text of the amendments can also be obtained.

In Figure 1 the x*mLegesEditor* drafting environment is shown.



Nome	Valore
id	art1-com1
xml:lang	it
xml:lang	it

Legge 8 maggio 1998 , n. 146

"Disposizioni per la semplificazione e la razionalizzazione del sistema tributario e per il funzionamento dell'Amministrazione finanziaria, nonché disposizioni varie di carattere finanziario"

[formula iniziale]

Capo I

DISPOSIZIONI IN MATERIA DI IMPOSTE DIRETTE, SUL VALORE AGGIUNTO E DI ALTRE IMPOSTE INDIRETTE

Art. 1.

(Dividendi distribuiti da società non residenti).

1. All'articolo 96-bis del testo unico delle imposte sui redditi, approvato con decreto del Presidente della Repubblica 22 dicembre 1986, n. 917 , e successive modificazioni, concernente i dividendi distribuiti da società non residenti, dopo il comma 2, è inserito il seguente: " *2-bis. A seguito dell'ingresso di nuovi Stati nella Comunità europea, con decreto del Ministro delle finanze è integrato l'elenco delle imposte di cui alla lettera c) del*

- decreto del Presidente della Repubblica 22 dicembre 1986, n. 917 [urn:nir:presidente.repubblica:decreto:1986-12-22;917]
- decreto-legge n. 648 del 1976 [urn:nir:stato:decreto.legge:1976;648]
- legge n. 730 del 1976 [urn:nir:stato:legge:1976;730]
- articolo 40 quarto comma del decreto-legge 18 settembre 1976 n 648

Figure 1: **The xmLegesEditor environment**

6 Conclusions and future developments

In this paper an overview of the "Norme in Rete" (NIR) project has been presented. It aimed at defining standards for Italian legislation and tools to promote their adoption. To identify and describe legal documents URN and XML standards have been respectively defined. To promote the adoption of such standards and to help users in implementing them ITTIG-CNR has developed some tools. In particular *NIREditor*, a specific editor for legal drafting dealing with NIR standards, has been developed. It also includes tools to handle legacy content and to extract document semantics. Version 2.0 of NIR-DTDs has been recently released, as well as the XMLSchema version of the NIR standards. Similarly, the official DTD for representing a new bill is going to be shortly defined and released. Finally a new version of the legislative editor, *xmLegesEditor*, implementing NIR standards has been presented. Is has been developed by ITTIG-CNR on the basis of the experience mature within *NIREditor* project.

References

Bartolini, R., A. Lenci, S. Montemagni, and V. Pirrelli. 2002. The lexicon-grammar balance in robust parsing of Italian. Proceedings of 3rd International Conference on Language Resources and Evaluation.

Bartolini, R., A. Lenci, S. Montemagni, V. Pirrelli, and C. Soria. 2004. Automatic classification and analysis of provisions in Italian legal texts: a case study. Proceedings of the Second International Workshop on Regulatory Ontologies.

Biagioli, C.. 1992. Law Making Environment. Proceedings of Workshop on Legal Knowledge and Legal Reasoning Systems, Tokyo.

Biagioli, C.. 1997. Towards a legal rules functional micro-ontology. Proceedings of workshop LEGONT '97.

Biagioli, C.. 2000. Ipotesi di modello descrittivo del testo legislativo per l'accesso in rete a informazioni giuridiche. Informatica e Diritto 2:90.

Biagioli, C., E. Francesconi, A. Passerini, S. Montemagni, and C. Soria. 2005a. Automatic semantics extraction in law documents. Proceedings of International Conference on Artificial Intelligence and Law. 133–139.

Biagioli, C., E. Francesconi, P. Spinosa, and M. Taddei. 2003. The NIR Project: Standards and Tools for Legislative Drafting and Legal Document Web Publication. Proceedings of ICAIL Workshop on e-Government: Modelling Norms and Concepts as Key Issues. 69–78.

Biagioli, C., E. Francesconi, P. Spinosa, and M. Taddei. 2005b. A legal drafting environment based on formal and semantic XML standards. Proceedings of International Conference on Artificial Intelligence and Law. 244–245.

Boer, A., R. Winkels, R. Hoekstra, and T.M. van Engers. 2003. Knowledge Management for Legislative Drafting in an International Setting. Proceedings of JURIX 2003: Legal Knowledge and Information System. 91–100.

Johnson, S.C. 1975. Yacc - Yet another compiler compiler. Technical Report CSTR 32, Bell Laboritories, Murray Hill, N.J.

Lesk, M.E. 1975. Lex - A lexical analyzer generator. Technical Report CSTR 39, Bell Laboritories, Murray Hill, N.J.

Megale, F., and F. Vitali. 2001. I DTD dei documenti di Norme in Rete. Informatica e Diritto 1:167–231.

Moats, R., and K. R. Sollins. 1997. URN Syntax. Technical Report RFC 2141, Internet Engineering Task Force (IETF).

Palmirani, M. 2005. Time Model in Normative Information System. Post-proceedings of the ICAIL Workshop on the Role of Legal Knowledge in e-Government.

Rabiner, L.R. 1989. A Tutorial on Hidden Markov Models and Selected Applications in Speech Recognition. Proceedings of the IEEE 77 (2): 81–106.

Spinosa, P. 2001. Identification of Legal Documents through URNs (Uniform Resource Names). Proceedings of the EuroWeb 2001, The Web in Public Administration.

Valente, A., and J. Breuker. 1997. A Functional Ontology of Law. C. Ciampi, F. Socci Natali, G. Taddei Elmi (eds), Verso un sistema esperto giuridico integrale, CEDAM.

van Kralingen, R.W. 1997. Frame-based Conceptual Models of Statute Law. Kluwer Law International.

Viterbi, A.J. 1967. Error bounds for convolutional codes and an asymptotically optimal decoding alghorithm. IEEE Transactions on Information Theory IT-13:260–269.