

# XML Documents Production For An Electronic Platform of Requests For Proposals

Frédéric Bapst and Christine Vanoirbeek  
Swiss Federal Institute of Technology – LITH MEDIA group  
IN-F Ecublens, 1015 Lausanne, Switzerland  
frederic.bapst@epfl.ch

## Abstract

*This paper addresses the problem of document production in the context of an electronic market application, namely a platform dealing with requests for proposals (RFP). Our project relies on XML as the leading technology, in accordance with a general trend in electronic commerce. The consequences of the RFP workflow on the document publishing requirements are motivated. Various aspects of the XML technology are discussed, with respect to both the present tools and the expected soon developments. We describe our structuring approach and the two environments that we used as testbeds. The most challenging issues concern collaborative authoring, WYSIWYG software, document life cycle, and linking expressiveness.*

## 1. Introduction

The request for proposals (RFP) process has proved to be an efficient business model, and tends to meet a growing success [8]. Nowadays, the management of RFPs still relies much on paper documents. We are working on a project which addresses the challenge [6] of porting the whole process on a fully electronic platform<sup>1</sup>. This ambition is related to the general revolution of electronic commerce, more specifically to the reorganization of business-to-business relations on top of the modern networks technology.

For a couple of years, the XML technologies [5] have evolved into a major field of interest in the computer science community. With its ultra-rapid development and propagation, XML already represents both a success story and a milestone in the Internet evolution. In general, XML can be used to represent any kind of data. More specifically, here are some families of domains that XML meets:

<sup>1</sup>the project is supported by the Swiss National Fund for Scientific Research (5003-054533). It involves the following consortium: EPFL, HEC Lausanne, University of Geneva, and Pioneers Research Consulting Group.

- *databases*: data collections that match the concept of a set of records, with homogeneous fields (e.g. dictionaries, price lists, etc.);
- *protocols*: description of the behavior of a platform, specification of event messages used by the internal machinery of a distributed application (see e.g. the ICE norm [11]);
- *publishing*: production of documents (e.g. reports or books), with a strong accent on authoring and typography (XML was derived from SGML, which was originally designed by the document publishing community).

In the field of electronic commerce, a lot has already been done around the use of XML for databases or protocols purposes. This meets the problems of data exchange and task automation in many business processes.

In this paper, we focus on the document publishing aspects. Indeed, business also occasionally involves the production of some complex documentations, which let the authors present their thoughts in the most effective way. The RFP process is one such business situation, where documents are mainly designed *by* and *for* human brains. We show that XML may yet play a role as decisive as in automation situations. Our electronic RFP platform is accessed through a Web server (<http://rfp.unil.ch>) acting as the core engine, but this topic will not be developed in this paper. Our focus is on the way end users process the documents, including the authoring issues.

## 2. Workflow Overview

### 2.1. Basic RFP Process

The process of RFP is a kind of demand-driven business. Roughly, the buyer publishes the project specification as an RFP, then the potential suppliers submit their offers, and finally the buyer chooses the best candidate. Inbetween, there is usually the possibility to discuss ambiguities in a forum-like way.

An emitted RFP contributes (i) to specify the prescriber's needs, (ii) to officially and widely announce the buying intention, and (iii) to prepare the comparison step by constraining the contents of the proposals.

**Legal aspects.** Most often, the whole process is subject to legal rules, and the participants have to respect their obligations. The legal aspects play a crucial role in practice. Nevertheless, we claim that a research project can be an opportunity to anticipate some modifications, because the technology often tends to precede the laws.

**Collaborative work.** The RFP process nearly always involves a group of collaborators, at the buyer side as well as at the supplier side. Different services take part in the management of the RFP, because such a project has commercial, technical and organizational implications. Both the authoring and the evaluation processes reveal this groupware dimension.

**Constrained document contents.** In the RFP process, constraining the document structure is a means to anticipate the evaluation step: the constraints put on the proposals serve as a guaranty that every decisive information will be mentioned, and that every point will be found in the same place across the different proposals.

**Evaluation step.** Basically, the central step in the whole RFP process is the evaluation of the proposals. Everything should be done to prepare this phase, and to guide the staff in the comparison of the proposals.

## 2.2. Benefits Expected From Electronic Support

The traditional RFP process was based on paper documents. Our project proposes the shift into an electronic system. We discuss here the major benefits expected from this important reorientation.

**Dynamic views.** The electronic world allows multiple ways of visualizing the information. It encourages the management of dynamic views, structural transformations for presentation purposes, and on-the-fly data assembly. This feature is also related to the Decision Support Systems principles, where the access to the information is organized in a set of indicators that summarizes the situation. This is interesting in the case of RFP, because people need hints to feel the market trends.

**Automation and reuse.** The electronic media can help to reduce the efforts (manual work) in the production or processing of the documents. It should encourage the reuse of pieces of information (hence the current hot topic of so-called *components management systems*). An electronic environment also tries to automate all repetitive tasks.

**Standardization.** An electronic support *should* (!) encourage a kind of standardization. Thus we could expect

to reduce the overhead of each new project by using standard RFP procedures. The firms should take advantage of the experience that standards do convey, in order to avoid to most prominent pitfalls in the RFP process. A certain degree of standardization would help firms to speak the same language, thus reducing the risks of misunderstandings.

**Workflow assistance.** Another goal of an electronic environment is to smooth the progression of the work across the steps. The platform should reflect the natural evolution of the process, to offer the most relevant functionality from one phase to another. This also means insuring the integrity of the information from one collaborator to the other.

**Integration.** An electronic environment gives the opportunity to link together all the information used inside the business. The network may help to suppress some practical barriers to information access. This is a precondition to the tasks of information gathering and retrieval, archiving, or data mining.

## 2.3. Overview of our Electronic Market Platform

The core of our RFP project is the electronic platform itself, implemented as a Web site (<http://rfp.unil.ch>). Here is a summary of the supported features.

**Official documents submission.** The basic role of the platform is to let the firms emit their official RFPs or proposals, and to manage the underlying exchange of messages. We also offer several information broadcasting and filtering protocols.

**Market intelligence.** The platform offers two means of market intelligence. First we support a search engine mechanism, that relies on the way the firms describe themselves. Second we synthesize the market trends into a set of indicators.

**Access rights.** For every firm that wants to join the market, the platform manages a set of end users, with their own intervention rights. This ensures that only the project leader can finally decide the official emission of a document.

**Project follow up.** The support does not stop at the emission of the RFP. Our platform lets the firms annotate the subsequent steps of the projects, so that there is a trace of the whole project (evaluation, contracting, delivery, etc.). This is also a basis to define a reputation mechanism, that will help to fight the non-cooperative actors.

**Document publishing.** Our platform also addresses the document publishing requirements, e.g. by putting document templates at disposal, or even by providing an environment with specific editing commands, that the user would not find in a general editor. This topic is developed later in this paper.

### 3. Requirements on Structured Documents

This section first presents the classes of documents that are considered in our RFP platform. Then we state our objectives as to the document structures, sketching what kind of functionality they should enable.

#### 3.1. Document Set

We consider four main document classes in our RFP project. In fact, we restrict the discussion to the documents that request a high degree of authoring effort. For instance, the exchange of questions and answers that may follow the publication of an RFP is likely to be much simpler. Although the structured approach also applies on such small documents, it is more representative to illustrate it on bigger documents.

**Request for proposals.** The request for proposal itself is the official document through which the prescripator expresses his intention. It is a strategic business document. Its design is dependent on the firm internal organization. The authors do want to use all the richness of the natural language to compose them, and every word is weighted. Thus most of the information will remain out of the scope of a simplified formalism. The RFP contents covers two complementary issues:

- *Context:* every piece of information that help the suppliers to understand the needs and to invent a relevant solution. This includes technical specifications as well as some more informal material about the project context.
- *Procedure:* every piece of information that helps the suppliers to make a valid proposal, i.e. to respect the participation rules. This includes e.g. the deadline, possible fees, the legal constraints, but also the requested structure of the proposal document.

**Proposal.** The proposal is a document which reveals some structuring challenges:

- the prescripator tries to impose his unique structure, to help the evaluation phase where the proposals must be compared;
- the supplier often tries to reuse some existing documents, and thus is not very enthusiast to make every-time a restructuring effort;
- in general, it is a kind of art to find the compromise between a too loose and a too strict canvas.

**Evaluation report.** We already mentioned that the evaluation of the proposals is a central step. In our project, we suppose that the result of this phase will be delivered as an evaluation report. But the authoring of this document is likely to be a distributed process: different member of the project staff will begin to enumerate theirs remarks while

they are reading the proposals. These notes will be updated because a further reading can make the situation clearer. Finally the diverse opinions will be consolidated after a discussion phase, which will lead to the final decision. During all the evaluation, the staff is guided by a set of criteria. Some of these criteria were probably announced in the RFP. Perhaps they have already been split into a bunch of precise points, that is internal to the evaluation staff. A few criteria often have to be added during the evaluation, because the proposals may raise some unexpected discriminating issues.

**Business description.** Our last document class is not directly related to the management of an RFP. In our platform, we suppose that every participating firm publishes a business description, which summarizes its skills, the domain of activity, the catalog of services, etc. The idea is that the RFP platform should also work as a privileged place to look for some new partnership or to know more about the market actors. So we suggest that each firm will have the opportunity to introduce itself via one Web page. The firm is free to organize and format its business description as desired. For instance, it could mimic its site homepage. On the other hand, we will maintain a hierarchical collection of predefined skills and activities, and we strongly encourage the firm to link them to its business description. This will allow to design a search engine which will find all firms concerned with a certain profile. The skill could be linked to a particular portion of the business description, so that the search results could directly lead to the paragraphs explaining that activity.

#### 3.2. Structuring Goals

The document structures are highly dependent on the underlying objectives. Here we discuss three families of goals that lead to the structuring requirements for the RFP documents.

**Intrinsic information exploitation.** Obviously the most natural reason to structure documents is that we want to process its contents in a more assisted way. Here are some examples:

- the RFP platform could detect the key information in an RFP (deadline, emitter) to maintain a list of all currently active RFPs;
- the evaluation report could be instantiated with the criteria announced in the RFP;
- the notes agglomerated in the evaluation report could be filtered to focus on the comparison of two particular proposal;
- the association between the predefined skills and the business description could be initially assisted through a simple automatic keyword search.

**Conformity verification.** One of the functionality that is expected from the structured approach is the task of verifying that the document instances respect the constraints of a model. In our RFP project, we do not force the firms to use any standard model [8], because the table of contents of an RFP will depend on the project's context, the domain of activity, and the firm's own habits. But there is one situation where we face a conformity verification problem, namely between the proposals and the model suggested in the RFP. Here the use of an electronic environment can ensure that the prescriber's intentions are respected. Note that in this case, one model is defined for each RFP and thus has only this limited scope.

**Groupware support.** The document structures may not only rest on the conveyed information, but also on the general process of reading, writing or commenting it throughout the document evolution. The focus moves from the document contents to the environment used to edit it. This aspect is particularly important in our RFP project, because we insist on the integration of the document management within a workflow involving several end users. This can be achieved partially with some structuring guidelines. Here are some examples:

- we can model the concept of an *annotation* (who writes it, about what document parts, at what time, etc.), and encourage to insert comments in every document;
- the electronic support can simulate a *marker* functionality, to highlight some portion of a document, as we are used to do with paper documents;
- we can define different *access rights* according to the user and the edited portions (collaborative and concurrent edition is in fact a hard problem [3]);
- the environment should offer an *integrated* tool to visualize the document, edit it, and navigate like in any browser; the separation of these tasks in different softwares would seriously jeopardize the homogeneity of the process.

## 4. XML Means of Information Structuring

### 4.1. Structuring Methodology

The XML technology offers a framework to define structured documents. Here we describe two extreme "flavors" of structuring methodology.

**Heavy-weight structuring.** It consists in defining a complete model (DTD) of the documents, with a comprehensive and rigid set of tags, attributes, syntax rules, constraints, and associated semantics. This methodology applies in the same way as you design a data structure with a programming language. The structure of the information then conveys most of its interpretation. The contents play the role of particular values, without the power to express some unexpected variation.

**Light-weight structuring.** Instead of defining a complete model, the light-weight flavor consists in designing mechanisms to spot (i.e. to name) some elements, to semantically paint (i.e. to attach a property) some portions, to add some sentinel marks, or to express semantic links within the document. This methodology applies among others when the information is best expressed in natural text, i.e. when the domain requires a high degree of subtlety. Every single word does not need to convey a rich meta-information, but we still want some structuring facilities, e.g. to detect the presence of a particular key-sentence.

### 4.2. XML Document Production

The current technology offers different means to produce XML-structured data. In the context of the RFP project, the information is mainly supposed to be input (or at least assembled) by an end-user. Here we describe four typical families of approaches of interactive XML encoding.

**Native XML editors.** A first way to generate XML documents is to rely on a general-purposed XML editor. This approach lies in the document publishing tradition. When the editor is guided by the document model (the DTD or another schema mechanism), the process can guarantee the conformity of the edited instances. An editor offers a full access to every detail in the document.

**Filters and converters.** Instead of using a native XML editor, we can also let the end-user work with a non-XML application, and add a conversion step that will transform the data into an XML format. This approach can be used when the current software environment has proved to be appropriate and already produces structured information, though not in the desired XML format. But it is worth noting that the information should better be authored in a system that encodes the logical structures and not only the contents.

**Forms.** A popular way of soliciting user input is to design a set of forms. We consider here simple forms system, as specified for instance in HTML. Forms have the advantage to be a universal user-interface made of checkboxes, radio buttons, lists, text fields etc. But forms are best suited only when the structure of the user input match a predefined schema, namely a list of name-value couples.

**Dedicated user interface.** Finally, there is always the possibility to program a specific user interface that fits at best the application context. Traditionally, designing a complete interactive software application is expected to be by far the most expensive solution. But the XML technology offers several facilities that help to reduce the associated costs. For example, an XML application can be written in a very short time when relying on Web scripts and the Document Object Model (DOM).

Table 1 summarizes some of the pros and cons of these four approaches (which by the way can also be combined).

Approach	Features
XML Editor	document publishing tradition no other constraints than the DTD
Converter	supposes an adequate data source does not address user input
Forms	well-known interaction rigid edited structure
Dedicated UI	fine-tuned ergonomony development costs

**Table 1. XML production tools.**

In the context of the RFP project, we strongly feel the following dilemma. On the one hand, our documents require the power of a full publishing environment. On the other hand, we would like to guide the authors with special RFP-oriented commands. For instance, there are well-planned steps in the whole process, each with its own privileged editing requirements. We would like to take advantage of the facilities of a general editor, but with the possibility to adapt/constrain its behavior along the document life cycle. Consequently, we are studying and experimenting three issues, which are reported in the next sections:

- Going deeper into the use of the existing Web editors as a framework for editing RFPs, to exhibit some guidelines on retargetting them towards more structured editing.
- Developing in Javascript a small dedicated XML editor, to show the power of the present browsers, and to propose some commands typically related to RFPs.
- Sketching a general mechanism of declaratively adapting the behavior of an XML editor/browser, to show the need of an additional stone in the current XML technology.

## 5. Experiments with Amaya

Amaya [9] is the editor/browser prototype promoted by the W3C consortium as its testbed environment. It was developed on top of Thot [10], the academic leader in the field of structured documents editors. It offers a full support for CSS [2] style sheets, and perfectly integrates the editing and browsing dimensions. It also defines a very powerful document transformation utility, that can be used e.g. to assist the user in the composition of predefined complex structures from a simpler input.

Unfortunately the current version is restricted on HTML, although it fully supports XHTML [7] (a rewriting of HTML in XML), and MathML (an XML application for handling mathematics). As our project requires mainly a light-weight structuring approach, using XHTML is not a too restricting hypothesis, provided that we smartly use the HTML

structuring means (links, elements `div` and `span`, attributes `class` and `id`). An XHTML environment offers the advantages of being compatible with the current Web browsers, and of enabling all the XML panoply of tools (e.g. parsers).

In our experimental documents, we have included in the RFP the template of the proposals. Thus the suppliers start editing their proposal by tearing out a dedicated section from the RFP. This schema encourages the respect of the template. The subdivisions of the template are already named with identifiers (`id`), so that the proposal structure can be verified later. A *marker* facility has been implemented using CSS rules and the `class` attribute.

Our Amaya testbed also contains some pre-structured fragments, that the user can copy/paste from the RFP platform. This is used to help structuring annotations, evaluation notes, and the linking of skills to the business description.

Experimenting with Amaya is important because we are convinced that some of the leading XML editors of tomorrow will be based on the same principles. Moreover, the announced intentions for the upgrades of Amaya directly match our requirements, because it is planned (i) to support full XML, and (ii) to be oriented towards collaborative editing, as it is presently implemented in the Byzance editor [3].

## 6. Experiments with IE5

Internet Explorer 5 [4] is Microsoft latest web browser, which includes innovative XML features. To develop our IE5 testbed, we used the following supported functionalities:

- *native XML formatting*: IE5 accepts to browse any XML document; if the document is attached to a style sheet (either CSS or XSL), IE5 formats it accordingly, otherwise it shows a hierarchical view with collapse/expand facilities;
- *behaviors*: IE5 defines a new way to separate the HTML document contents from the behavior that the page should offer; as a side effect, this allows the CSS sheet to attach a particular reaction script to any XML elements, among a set of common events (e.g. a mouse click);
- *scripting library*: IE5 comes with a rich built-in library, which includes a standard (DOM) XML parser, as well as some routines that give access to the internals of the browser itself.

Combining these features gives the opportunity to develop our small XHTML editor, especially designed to match the RFP process. The goal was not to achieve the excellence of general editors, but to illustrate how to extend the browser with a couple of dedicated editing commands. The basic architecture consists in a Web page split into two

frames, one holding the document to edit, and the other giving access to a set of commands via some HTML form elements. The whole was programmed in Javascript, and is directly accessible through our global RFP platform.

The processing of the business description is a typical situation to show the benefits of the IE5-based architecture. The linking between the predefined skills and the freely-composed activity summary can be made in the most natural way, through some simple mouse operations between two frames. The underlying scripts are able to transform these events into a well-structured document portion, that will then feed the platform search engine.

## 7. Interaction Sheet Mechanism

Our experience with Amaya and IE5 suggests that an interesting approach would be to define in the XML technology a new general mechanism, dedicated to the parameterization of the editor/browser behavior. The basic idea is to extend the CSS formalism with interactive properties, which would among others indicate which parts of the document are supposed to be edited. We named this innovative approach with the term *interaction sheet* [1]. We also suggest to refine the possible scope of a rule within a style sheet: instead of restricting the targets to the document elements, we propose to use full XPointers, so that any part of the contents could be accessed. This would enable a general marker facility, which semantically does not really change the documents contents, but rather only the presentation part.

The underlying idea of interaction sheets is that a document owns a certain life cycle. Its modifications often follows a planned workflow (or, alternatively, some business workflow contains steps that requires some known modification of that document). An interaction sheet mechanism would solve the dilemma of XML user input that we mentioned in section 4.2.

In this context of interactive properties, we need more a built-in set of editing commands than a new event model (like the one supported in HTML or in the IE5 behavior mechanism). One goal would be to define a way to attach in a declarative way the access rights to the document portions. This would guide the end user to choose one of the expected editing commands (e.g. modification of some elements or attributes), according to the workflow progression. We have sketched the basis of an adequate interaction sheet mechanism, but our detailed proposition falls outside the scope of this paper.

## 8. Conclusion and Perspectives

This paper discussed the question of the document publishing process in the context of an electronic RFP platform. The context of the application was described to exhibit the

requirements on the structured document environment. Our project is definitely based on XML, because it is likely to become a real standard in the Internet applications, including electronic commerce. But XML is a cutting-edge technology, and it is hard to predict all its implications. We claim that it is very useful to feed the discussion with various concrete experiments. In this context, our contribution is to take a document publishing approach, that is complementary with respect to the database point of view found in many other projects.

Our work mainly consisted in:

- a modeling effort, with the proposition of structuring guidelines for RFP documents;
- the evaluation of two software solutions, with experiments with the Amaya editor, and the IE5 browser;
- the proposition of a new mechanism in the XML technology, to help expressing the interventions of the end users on XML documents.

The document publishing problem is only one aspect of the global RFP process. Our project also addresses the running of a Web server (<http://rfp.unil.ch>) playing the role of the electronic market, which organizes business-to-business relations through the support of the full RFP process. We are now consolidating the whole platform, and plan to conduct real-size experiments involving our industry partners.

## References

- [1] F. Bapst and C. Vanoirbeek. Towards an Interaction Sheet Mechanism in XML Technology. Technical report, Swiss Federal Institute of Technology (EPFL), 1999. Submitted to Markup Technology'99.
- [2] B. Bos and C. Lilley. Cascading Style Sheets Level 2 (CSS2) Specification. Technical report, W3C, 1999. <http://www.w3c.org/TR/REC-CSS2>.
- [3] D. Decouchant, M. Romero-Salcedo, and M. Serrano. Principes de conception d'une application d'édition coopérative de documents sur internet. In *IHM'97*, 1997.
- [4] A. Homer. *XML IE5 Programmer's Reference*. Wrox, 1999.
- [5] M. Leventhal, D. Lewis, and M. Fuchs. *Designing XML Internet Applications*. Prentice Hall, 1998.
- [6] S. Messin. Vers un marché électronique d'appels d'offres en informatique. Master's thesis, University of Lausanne, 1998.
- [7] S. Pemberton and M. Altheim. XHTML 1.0: The Extensible HyperText Markup Language. Technical report, W3C, 1999. <http://www.w3c.org/WD-html-in-xml>.
- [8] F. Petit. *Comment établir un appel d'offres en informatique*. Masson, 1997.
- [9] V. Quint and I. Vatton. Using Amaya. Technical report, W3C, 1999. <http://www.w3c.org/Amaya>.
- [10] I. Vatton, C. Roisin, and V. Quint. That Reference Manual. Technical report, INRIA, 1998. <http://www.inrialpes.fr/opera>.
- [11] N. Webber and B. Hunt. The Information and Content Exchange (ICE) Protocol. Technical report, W3C, 1998. <http://www.w3c.org/TR/Note-ice>.