

## MAPPING FACULTY CURRICULUM INTO XML SCHEMA<sup>1</sup>

Kosa Nenadić<sup>2</sup>, Dušan Surla<sup>2</sup>

**Abstract.** Managing student services at a university requires maintaining a diverse number of documents, such as a curriculum, exam application form, entry form, student booklet code, etc. This paper describes the curriculum document based on XML Schema with instance documents of the proposed schema as curriculum XML documents. Remaining student service documents can be easily described in much the same way.

*AMS Mathematics Subject Classification (2000):*

*Key words and phrases:* XML schema, curriculum

### 1. Introduction

An Internet application for managing university student services has been developed at the Faculty of Sciences in Novi Sad, and has been in use for the past two years. Papers [1-7] show parts of modeling and implementation results related to this application. The new version of this system, whose development is under way, is based on XML documents (papers [8-9]). Monographs [10, 11] are used as the basis for this paper. These monographs are related to Bibliographic Software System development based on XML files. The search through the available literature showed that this topic has not been treated up to now.

Basic university student service documents include the curriculum, exam application form, student's booklet code, entry form, etc. All these documents can be represented as XML documents. This paper presents an analysis of the curriculum structure and proposes the way of their XML Schema description. Using this schema, it is possible to generate the entire curriculum offered by the Faculty of Sciences Novi Sad.

### 2. The curriculum

The structure of curriculum document contains heading and courses. Heading contains the enacting date, expiry date, department, majoring profile and dates of some minor document alterations. Part of the document that refers

---

<sup>1</sup>This paper is part of the research project 'XML Technologies and Cooperative Information Systems', supported by the Ministry of Science and Environmental Protection of the Republic of Serbia (Project No. 1875).

<sup>2</sup>Department of Mathematics and Informatics, Faculty of Science, University of Novi Sad, Trg D. Obradovica 4, Novi Sad 21000, Serbia and Montenegro, E-mail: [surla@uns.ns.ac.yu](mailto:surla@uns.ns.ac.yu)

to courses consists of mandatory courses and elective groups / elective courses. Each of these has information about course title, course semesters, number of lessons per week and examination type.

We distinguish two different types of curriculum documents according to the structure of their courses. The first type refers to curriculum documents whose list of courses contains mandatory courses and elective groups. The elective group is consisted of courses and/or groups of courses. Each student chooses the prescribed number of courses from the elective group and its subgroups. The second type refers to curriculum documents whose list of courses contains mandatory courses plus a number of elective courses that are selected from of the list of elective courses. Each elective course has a list of courses, and a student chooses one course from this list. Additionally, the course list can be divided into groups. If this is the case, these groups affect student's choice.

XML curriculum document schema modeling is realized using XML Spy editor. Of different XML views in this editor, design view and text view are used most frequently. Graphical elements represent different schema concepts, making thus the work much more convenient. The hierarchical structure of XML document is represented in a tree form. The sections to follow describe curriculum modeling in Spy editor.

### 3. Curriculum element

The curriculum consists of the main element, *Curriculum*, and the subelements *Heading* and *Courses* (Figure 1). These elements in turn contain other subelements.

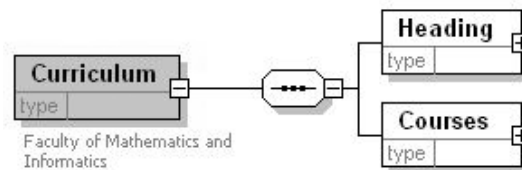
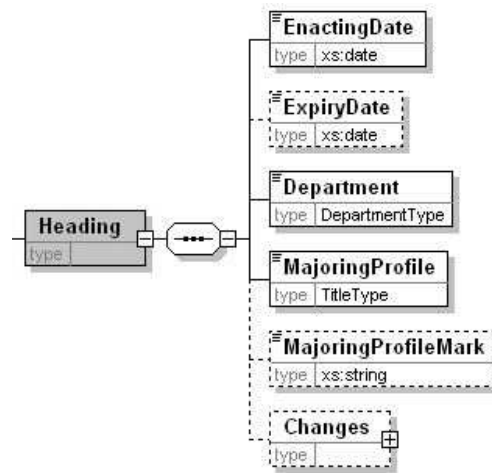


Figure 1. The main element *Curriculum*

#### 3.1 Heading element

The *Heading* element has an anonymous complex type consisting of the subelements *EnactingDate*, *ExpiryDate*, *Department*, *MajoringProfile*, *MajoringProfileMark* and *Changes* (Figure 2).

**Enacting date element.** The *EnactingDate* element refers to the date of curriculum enactment. The type of this element is declared to be a built-in simple type *date*.

Figure 2. The *Heading* element

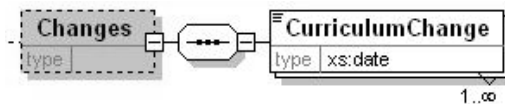
**Expiry date element.** The *ExpiryDate* element refers to the curriculum expiry date. This is an optional element whose type is declared to be a simple type *date*.

**Department element.** Each curriculum is referred to a certain department. The *Department* element carries this information, and value of this must be one of the atomic values of the enumerated type *DepartmentType*.

**Majoring profile element.** A majoring profile title is represented by *MajoringProfile* element. The element type is *TitleType* which is a restriction of built-in type *string*.

**Majoring profile mark element.** The *MajoringProfileMark* element is an optional element. The value of this element is the mark of the majoring profile.

**Changes element.** The Faculty Council is empowered to make decisions about curriculum changes. These decisions have to be documented and the date of the change recorded. The *Changes* element comprises *CurriculumChange* elements, which are of a simple type *date* (Figure 3). If the curriculum document has not been changed, the corresponding instance document of schema does not contain the *Changes* element. When the change is made a new *CurriculumChange* subelement is added along with appropriate date value.

Figure 3. The *Changes* element

### 3.2 Courses element

The *Courses* element includes information about curriculum courses. Courses are divided according to their nature. We distinguish mandatory courses (*Course*

element), elective groups (*ElectiveGroup* element) and elective courses (encapsulated in *ElectiveCourses* element). The XML Schema *choice* element enables multiple unordered choices. The *Courses* element is composed of the elements *Course*, *ElectiveGroup* and *ElectiveCourses* (Figure 4).

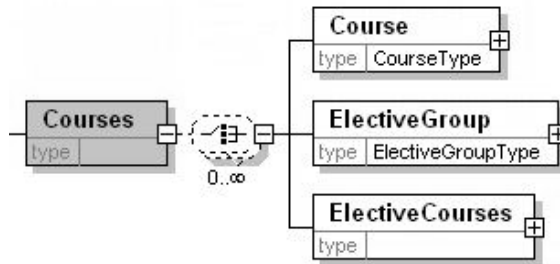
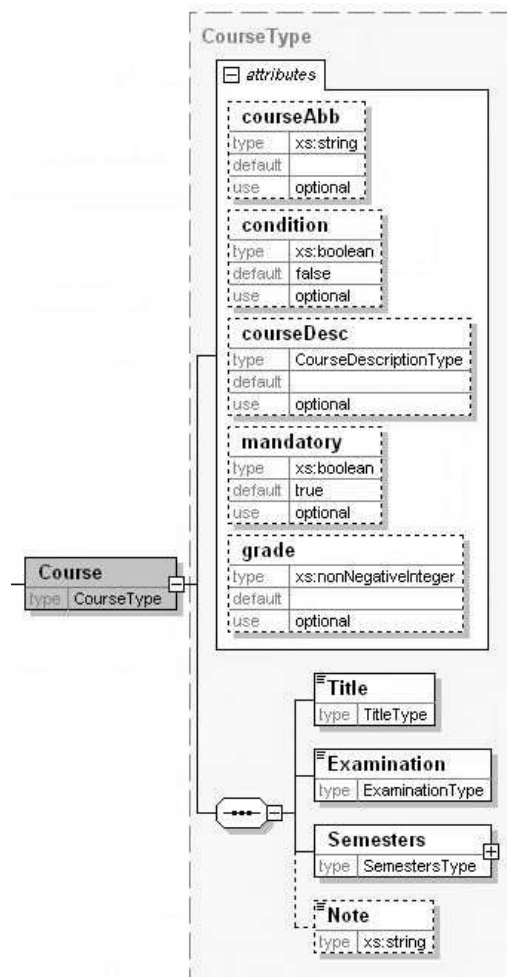


Figure 4. The *Courses* element

### 3.2.1 Course element

The *Course* element type is defined as a complex type *CourseType*. This type is a sequence of elements: *Title*, *Examination*, *Semesters* and *Note*. Type of the *Title* element is *TitleType*, and it is simple. The *Examination* element is described with enumerated type *ExaminationType*. The information about the number of lessons per week in each semester, represented by complex type *SemestersType*, is contained in the *Semesters* element. Also, there is an optional *Note* element.

The *Course* element carries the following optional attributes: *courseAbb*, *condition*, *courseDesc*, *mandatory* and *grade*. They are shown in Figure 5. The *courseAbb* attribute represents abbreviated course title. If a course is obligatory to be passed for the next year's enrollment, a value of the *condition* attribute is true, while default value for this attribute is *false*. In some curriculum documents courses are categorized, e.g. humanities. The *courseDesc* attribute carries information about course category and it is of enumerated type *CourseDescriptionType*. Even though a course belongs to elective group it can be prescribed as mandatory within that group, the *mandatory* attribute denotes whether it is prescribed or not. The default value is *true*. There is also the *grade* attribute standing for the year of study.

Figure 5. The *Course* element

### 3.2.1.1 Semesters element

The *Semesters* element contains information about the number of lessons per week, in each semester. The element type is *SemestersType*. The element carries a required *semesterNo* attribute, which refers to the ordinal number of semester. This element also carries optional attributes: *lectures*, *exercises*, *laboratoryExercises*, *fieldExercises*, *practicalWork* and *fieldWork*, which refer to number of lectures, exercises, laboratory exercises, field exercises, practical work and field work, per week, respectively.

For each elective group and elective course (in the curriculum document), a total number of lessons per week in each semester is given, regarding the fact

that courses often have different number of lectures and exercises, while their total is equal. If that is not the case, the alternative total is given. Attributes *total* and *alternativeTotal* refer to this (Figure 6).

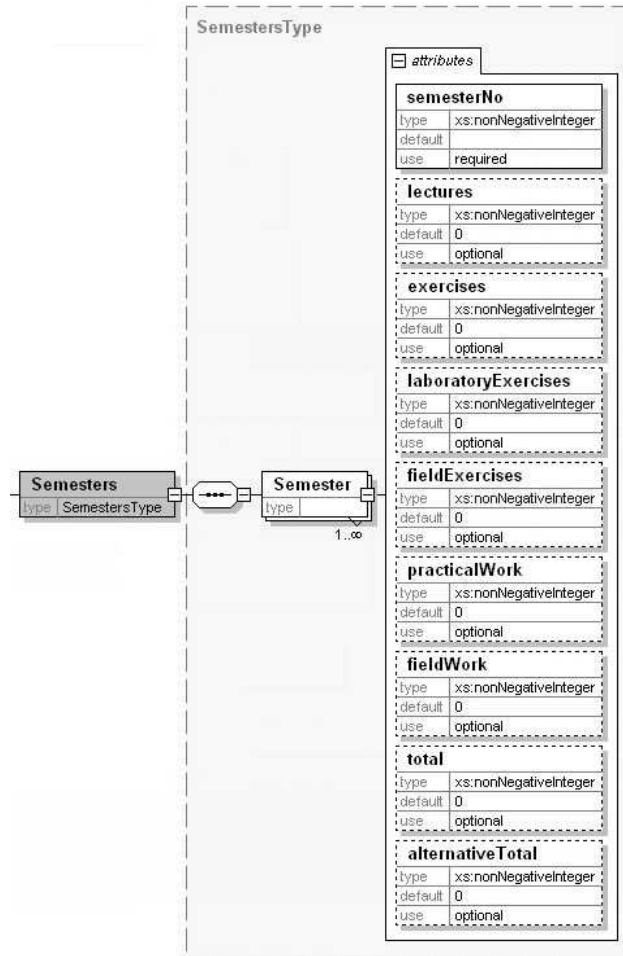


Figure 6. The *Semesters* element

### 3.2.2 Elective group element

The *ElectiveGroup* element is shown in Figure 7. The four subelements *Title*, *Examination*, *Semesters* and *ElectiveGroupCourses* are declared as part of *ElectiveGroupType* type. The *Title* and *Examination* elements are optional. And, the *ElectiveGroupCourses* element comprises all the information concerning the courses. The *Course* and *Subgroup* elements appear in arbitrary order, as child elements of the *ElectiveGroupCourses* element.

The *ElectiveGroup* element carries optional attributes. The *grade* attribute refers to the year of study. The *courseNo* attribute contains the number of the elective group courses. If necessary, it can be preset, although it is possible to count it using other XML technologies. The *chooseNo* attribute represents the number of courses that has to be chosen from the group; default value for this attribute is 1.

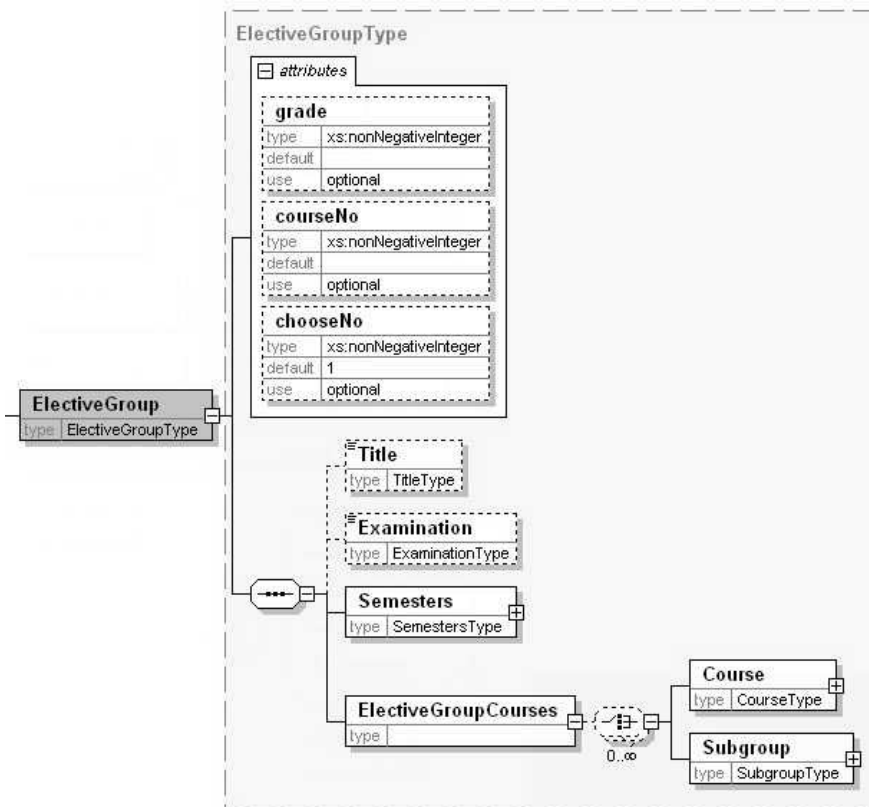
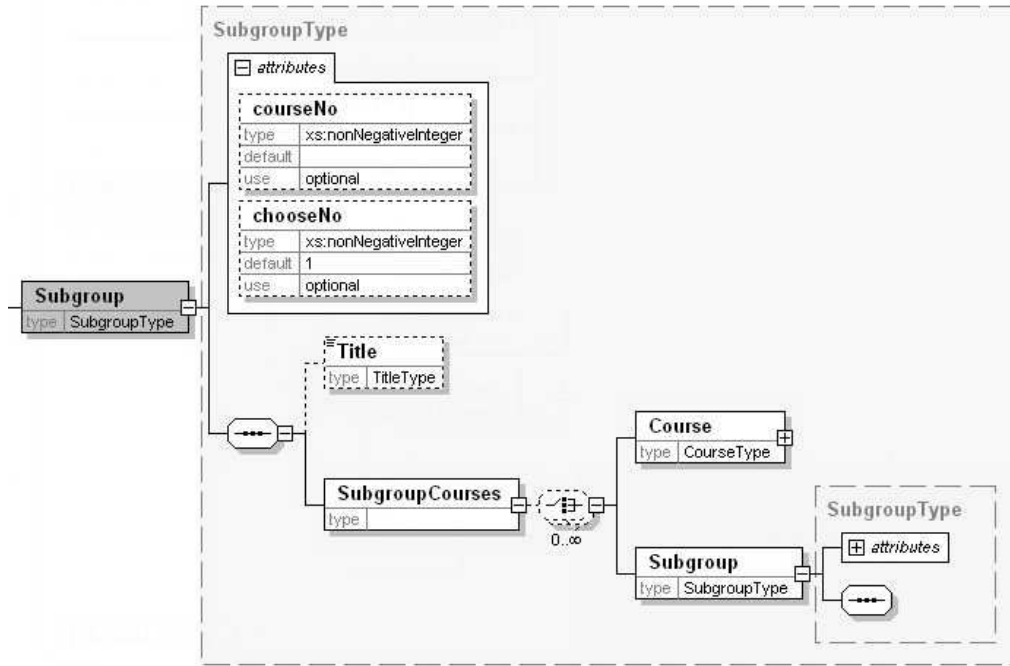


Figure 7. The *ElectiveGroup* element

### 3.2.2.1 Subgroup element

The *Subgroup* element is designed in order to realize operation of nesting subgroups of courses into the main elective group, as well as into the other subgroups (Figure 8). The sequence of subelements contains two elements: the *Title* and *SubgroupCourses*. The *Title* element is optional. The *SubgroupCourses* element represents arbitrary sequence of the *Course* and *Subgroup* elements.

Figure 8. The *Subgroup* element

### 3.2.3 Elective courses element

The *ElectiveCourses* element represents elective courses which belong to the same year of study (Figure 9). The element has an anonymous complex type formed from a sequence of *ElectiveCourse* elements followed by either *ElectiveCourseList1* or *ElectiveCourseList2* element; both of these elements describe all possible values for elective courses.

Also, this element carries only one attribute called *grade* and defines three identity constraints: key – *idKey1* and keyref – *idKeyRef1*, *idKeyRef2*.



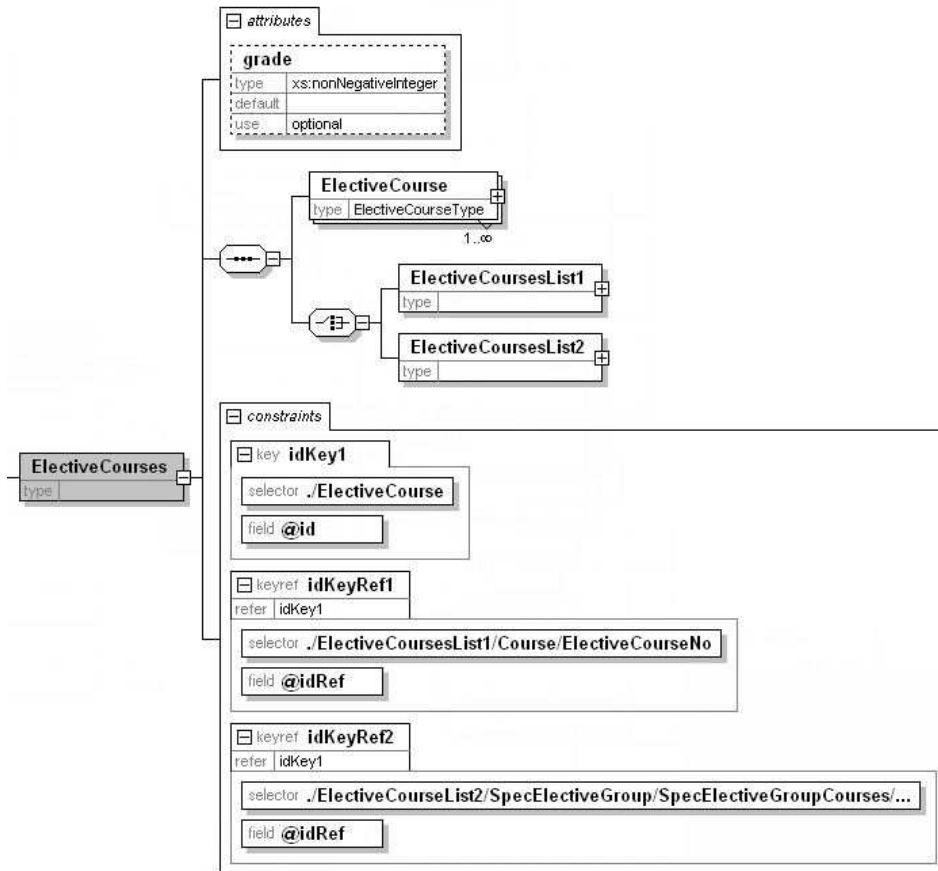
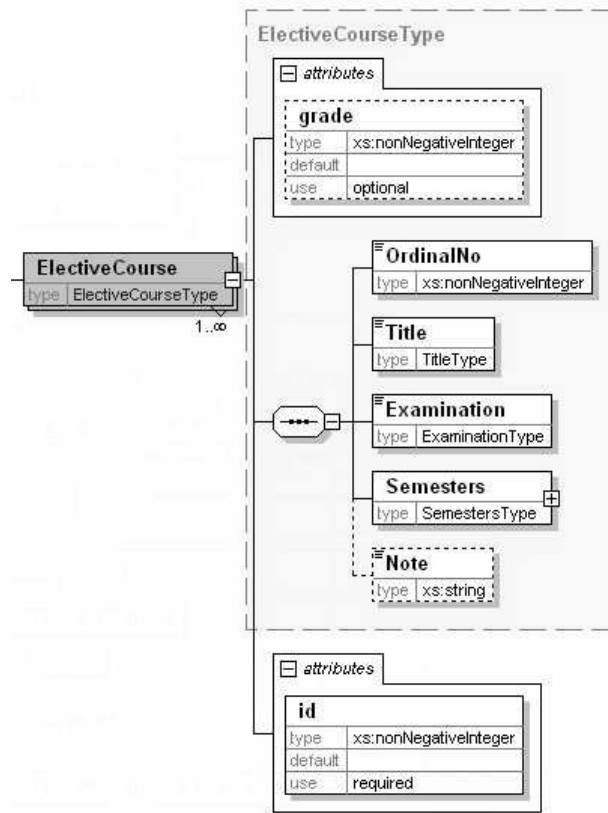


Figure 9. The *ElectiveCourses* element

### 3.2.3.1 Elective course element

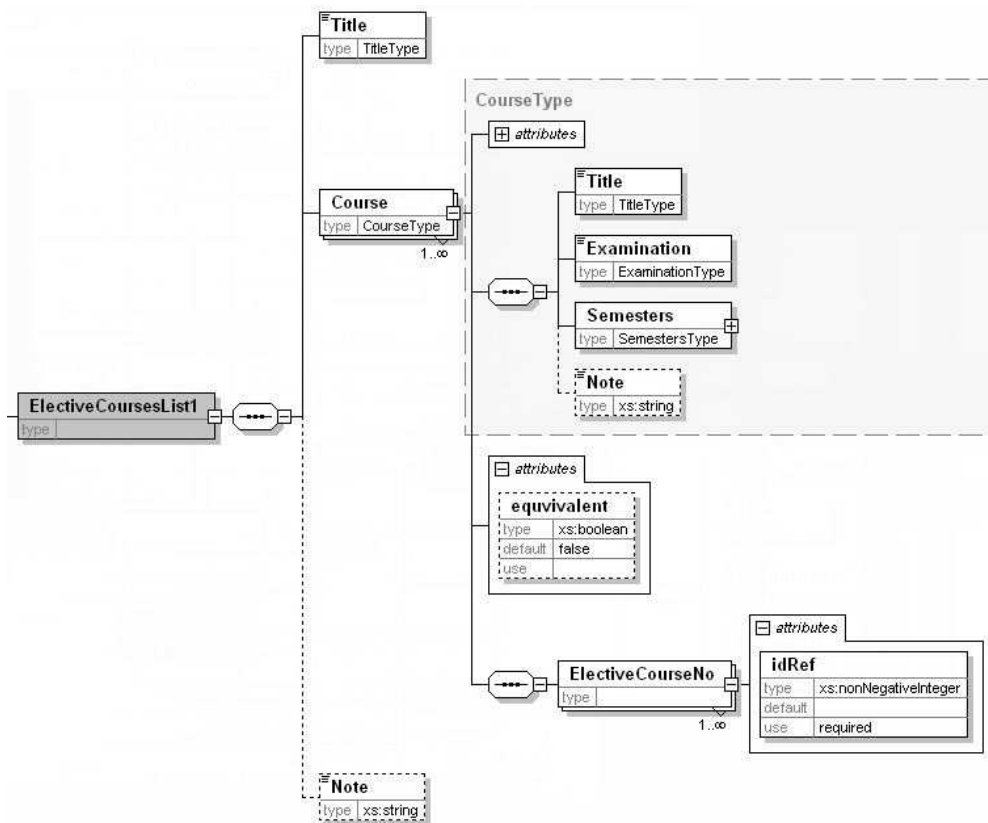
The *ElectiveCourse* element describes the elective course (Figure 10). Type of this element is the *ElectiveCourseType* type. Main difference between this type and the *CourseType* is the first sequence element. This element, called *OrdinalNo*, refers to the ordinal number of the elective course.

In addition to *ElectiveCourseType* type, the *ElectiveCourse* element has the *id* attribute. Value of this attribute has to be unique because identity constraint key is defined. The *ElectiveCourseNo* element's *idRef* attribute refers to *id* attribute value.

Figure 10. The *ElectiveCourse* element

### 3.2.3.2 First elective course list element

The *ElectiveCoursesList1* element is shown in Figure 11. The element has an anonymous type consisting of the following elements: *Title*, *Course* and optional *Note*. The *Course* element describes all courses that can be chosen as elective courses. In addition to standard attributes, the *Course* element also carries the *equivalent* attribute. Whether the choice of a particular course makes the same credit as all courses mentioned in the *ElectiveCourseNo* element sequence or not, it depends on logical value of the *equivalent* attribute. Default value of the *equivalent* attribute is *false*. In addition to child elements defined by *CourseType*, each *Course* element contains a sequence of the *ElectiveCourseNo* elements. The *idRef* attribute of the *ElectiveCourseNo* element is a reference to the *id* attribute of the *ElectiveCourse* element.

Figure 11. The *ElectiveCoursesList1* element

### 3.2.3.3 Second elective course list element

In special cases of grouping, when the courses belonging to elective courses list are divided in special elective groups, the *ElectiveCoursesList2* element is used (Figure 12). The sequence of elements contains the *Title* element, *SpecElectiveGroup* elements, and optional *Note* element.

#### 3.2.3.3.1 Special elective group element

The *SpecElectiveGroup* element contains the *Title* and *SpecElectiveGroupCourses* element defined by *SpecElectiveGroupType* (Figure 13). The value of the *Title* element is the title of special elective group. The *SpecElectiveGroupCourses* element encompasses all courses that belong to that particular group. The attributes *minChooseNo* and *maxChooseNo* represent minimal and maximal number of courses that can be chosen from the group.

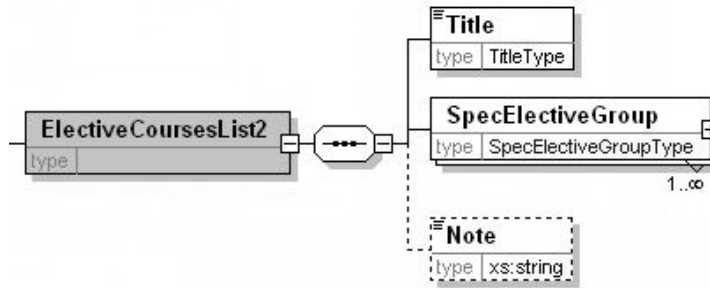


Figure 12. The *ElectiveCoursesList2* element

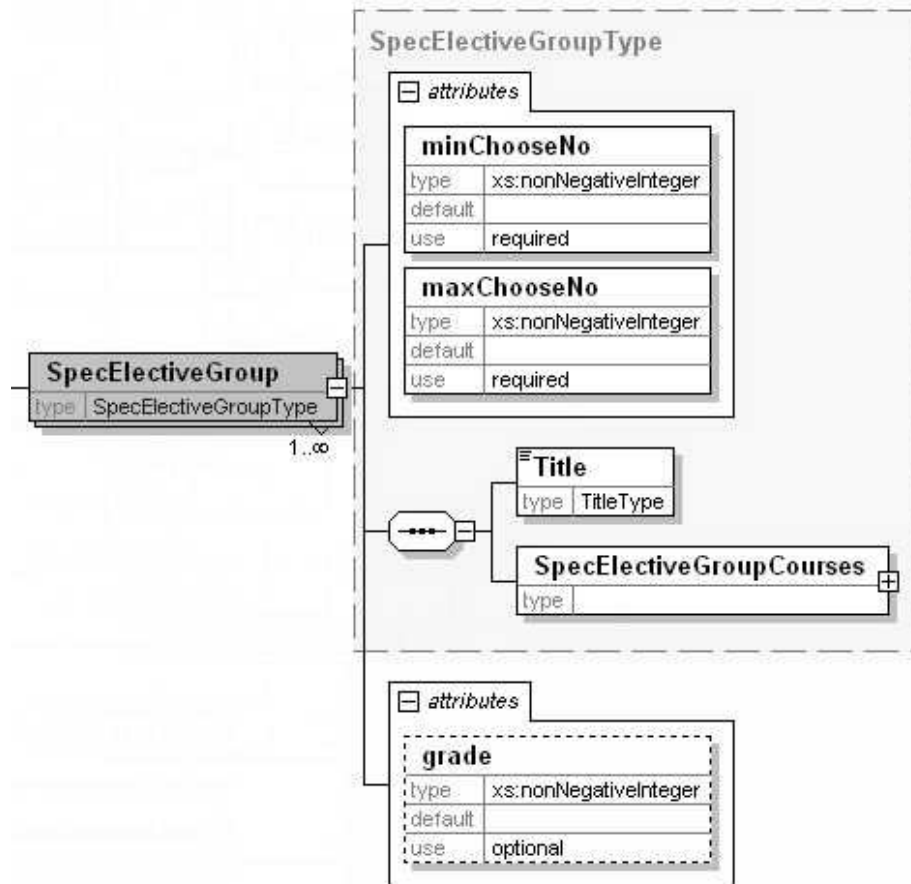


Figure 13. The *SpecElectiveGroup* element

### 3.2.3.3.2 Special elective group courses element

The structure of the *SpecElectiveGroupCourses* element is similar to the *ElectiveCoursesList1* element, in a sense that it represents a sequence of *Course* elements, which can be chosen and which are restricted to belong to some particular group of courses (Figure 14).

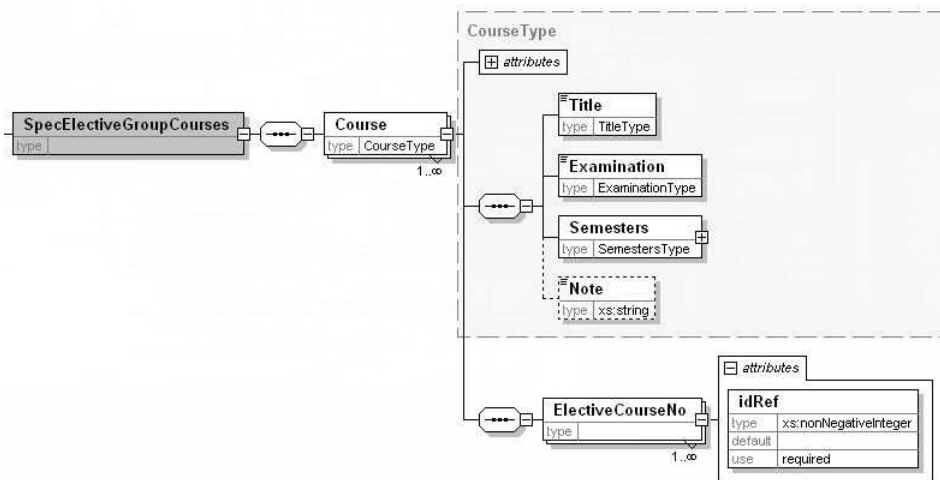


Figure 14. The *SpecElectiveGroupCourses* element

## 4. Conclusion

The curriculum document is described using XML schema language. The proposed schema enables instantiation of each curriculum document at the Faculty of Sciences Novi Sad. This faculty has about 150 curriculum documents characteristic of its departments: physics, chemistry, biology, mathematics and informatics. Minor changes in this schema could be made in order to apply it at other faculties.

Other student service documents, such as exam application form, entry form, student booklet code and semester form, can be described in XML schema language. Student service XML documents can be used for information system development, document storing and data migration.

## References

- [1] Sedej, M., Data entry and reporting for university curricula, BS thesis, Faculty of Sciences Novi Sad, 2002 (in Serbian).
- [2] Surla, V., Modeling, development and implementation of curriculum information subsystem, BS thesis, Faculty of Economics Subotica, 2002 (in Serbian).

- [3] Srećkov, V., Data migration in the student service information system, BS thesis, Faculty of Sciences Novi Sad, 2003 (in Serbian). <http://diglib.ns.ac.yu/ndltd/docs/set2/ndltd223/DiplomskiSreckov.pdf>
- [4] Gero, A., Modeling and implementation of the faculty curricula in the student service information system, BS thesis, Faculty of Sciences Novi Sad, 2003 (in Serbian). <http://diglib.ns.ac.yu/ndltd/docs/set1/ndltd86/diplomski.pdf>
- [5] Nenadić, K., Data structure of curriculums, BS thesis, Faculty of Sciences Novi Sad, 2003 (in Serbian). [http://diglib.ns.ac.yu/ndltd/docs/set2/ndltd333/Diplomski\\_rad.pdf](http://diglib.ns.ac.yu/ndltd/docs/set2/ndltd333/Diplomski_rad.pdf)
- [6] Mirković, M., Implementation of final exam registration in the student service information system, BS thesis, Faculty of Sciences Novi Sad, 2004 (in Serbian).
- [7] Racković, M., Surla, D., Data model of curriculums and higher education programs, Collection of papers, YU INFO'03, CD-ROM, Kopaonik (in Serbian).
- [8] Nenadić, K., Surla, D., Modeling of student service documents in XML schema language, YU INFO'04, CD-ROM, Kopaonik (in Serbian).
- [9] Nenadić, K., Surla, D., Visualization of XML curriculum document, Collection of papers, SYM- OP-IS'04, Iriski Venac, Fruska Gora (in Serbian).
- [10] Jakšić, M., Mijić, V., Surla, D., XML document of UNIMARC format, Faculty of Science, Novi Sad, 2004 (in Serbian).
- [11] Budimir, G., Surla, D., Quality Control System of XML Bibliographic Records, Faculty of Science, Novi Sad, 2004 (in Serbian).

*Received by the editors September 28, 2004*