An object-oriented method to create an SGML DTD of an electronic patient record

E. Kuikka, A. Eerola, A. Miettinen,
J. Porrasmaa, M. Ek, J. Komulainen

Report A/1999/6

ISBN 951-781-614-6
ISSN 0787-6416

UNIVERSITY OF KUOPIO

Department of Computer Science
and Applied Mathematics

P.O.Box 1627, FIN-70211 Kuopio, FINLAND

An object-oriented method to create an SGML DTD of an electronic patient record

E. Kuikka, A. Eerola, A. Miettinen, J. Porrasmaa, and M. Ek
Department of Computer Science and Applied Mathematics
University of Kuopio

J. Komulainen
Department of Pediatrics
Kuopio University Hospital

Abstract

A patient record is typically a document which is updated by many users, required to be represented in many different layouts, transfered from a place to another, and archived for a long time. It is also an object for various types of queries. Thus, it is a very good candidate to be represented as a structured document and coded using SGML document standard.

An SGML based system requires that the structure of the document is defined in advance as a Data Type Definition (DTD). If the structure is defined based on the paper form, the structure does not necessary reflect what the doctors and nurses do in their everyday practise. The structure should reflect the use and reuse of the content elements. The object-oriented design methods produce this kind of information. The paper represents the method to generate the SGML DTD with the use of the UML diagrams.

Keywords: SGML, XML, object-oriented, UML, patient record, mapping rule

1 Introduction

The patient record contains the identification information for a patient as well as information of the closest relatives, medical facts, for example, allergies, blood group, etc. and actions and diagnoses in the various clinics. Traditionally, a patient record is a pile of paper forms filled by various clinics and laboratories of a hospital. The content consists of text, drawings, pictures, photos, films. The main part of the patient record, however, contains text dictated or written by doctors and nurses.

The textual content of the patient record consists of two types of data. The names of the patient, doctor, clinic, etc. are typical fixed-length data and suitable to be managed in the databases. For the other type of information, such as entrance reason, preliminary information, problem, diagnoses, statement, etc., it is not possible to fix any predefined length. This kind of information is usually written with the typewriter or a text processing system and represented in lines with variable lengths.

Many electronic patient record systems or prototypes have been implemented in particular environments using predefined techniques, for example, certain databases, certain programming languages, etc. [Bra94, BPD96, Car94, PSRL94]. This approach has produced new problems when additional features have been added into the system, when data has been transfered between various systems and when old data needs to be processed with new program versions.

An opposite approach to these specific implementations is taken in a set of systems or prototypes which represent the patient record as a structured document containing the standardized markup [Ben96, PGR+97, TMOK97, TMO+97, YOO+98, Sco98, JG99]. SGML (Standard Generalized Markup Language) [ISO86] is an international standard to define the markup notation used in structured documents.

An SGML document follows the structure which is defined in advance with the use of a Document Type Definition (DTD).

Very often the developing of the DTD is based on the paper form of a document. The designers use example documents or the knowledge of the user when they define the potential semantic elements and their hierarchical relationships [MeA96, TW96]. There is a possibility that, for example for the patient record, the derived structure may conflict with what the doctors and nurses actually require in their everyday practices. The problem here is that the necessity of information is not considered and new information requirements may not be found. Further, the possibilities to evaluate the medical care and nursing routines may not be discovered.

The aim of this paper is to develop a method for the derivation of an SGML DTD by starting from the description of the utilization of the patient record in medical care and nursing. Here we emphasize the improvement of the medical treatment, the rationalization of the information flow and the quality of the system. The method has the following steps:

1. The flow of the data in the clinic is considered from the patient record's point of view. The traditional data flow analysis [You89] is used to get the general description of actions around patient record.
2. The conceptual modeling is done using the Unified Modeling Language (UML) [Boo98]. Here it is important that the model describes the medical treatment in a natural way.
3. The elm (enables lucid models) tree diagram [MeA96] of the SGML DTD is created from the UML class diagram using modification rules derived in this research.

After this introduction, Section 2 defines what structured documents and SGML are. Section 3 describes the concepts of object oriented approach and the two first steps of the method, the analysis and design of the patient record conceptual structure producing a set of data flow diagrams and a set of UML diagrams. Section 4 represents the rules that are used to transform the UML diagrams to elm tree diagrams. Section 5 depicts the application of the rules to produce the elm trees phase by phase. The comparison to related work and the conclusion finish the paper.

2 Structured document and SGML

A structured document consists of named elements whose identifiers and hierarchical relationships are defined in advance. An example is represented in Figure 1. This document for the patient information contains the name of the patient and a sequence of visit data. Each visit information consists of the day, the names of the clinic and the doctor, and the problem of the patient.

SGML (Standard Generalized Markup Language) is an international standard [ISO86, Gol90, vH94] to define the notation which is used to represent hierarchically structured documents. The structure is defined using the DTD (Document Type Definition). The DTD is common for a set of documents (for example, a set of books, articles, patient records) and all the documents of the same type have to follow the DTD.

The DTD contains an *element rule* for each structure part of the document. The element rule has three parts: the name of the element, indicators for the existence of the markup, and the definition of the content model. The rule starts by the characters `<!ELEMENT` and ends with the character $>$. The *content model* specifies the sub-elements, their order and existence conditions. The DTD for the document in Figure 1 is as follows.

```
<!ELEMENT patientinfo                  - - (name, visit*)>
<!ELEMENT name                         - - (#PCDATA)>
<!ELEMENT visit                        - - (day,clinic,doctor?,problem)>
<!ELEMENT (day,clinic,doctor,problem) - - (#PCDATA)>
```

This DTD defines that the `patientinfo` element consists of the name of the patient (`name`) and zero or more visits (`visit*`) in the clinic. The patient name is a sequence of characters (`#PCDATA`). The `day`, `clinic`, `doctor?` and `problem` elements are contained in the `visit` element, in this order and by the

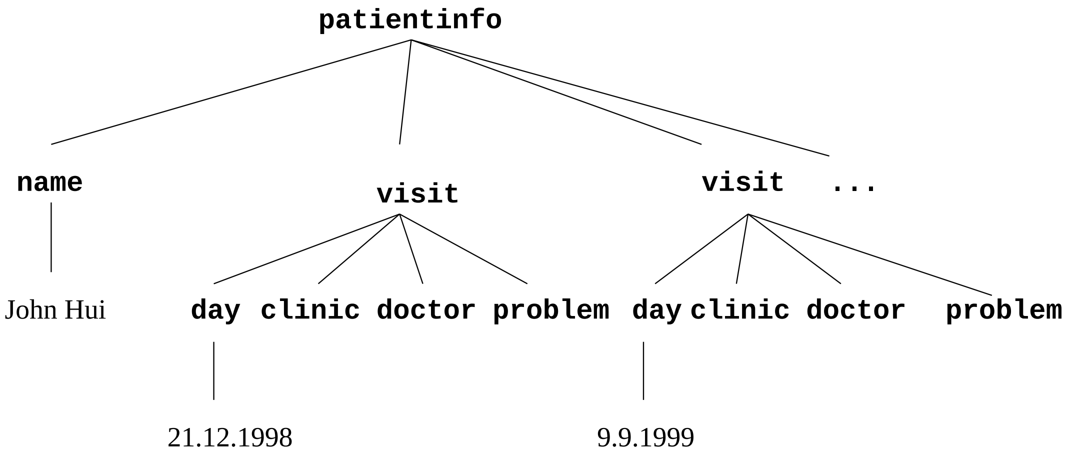


Figure 1: The structure of a patient record

condition, that the `doctor` element is optional. The characters $- -$ after the element identifier state that the element markup has to exist in the document instance.

If the same content model exists in many places of the DTD, it is possible to define a *parameter entity* to shorten the DTD in the following way.

```
<!ENTITY % timeandplace "day,clinic">
<!ELEMENT (day,clinic)     - - (#PCDATA)>
<!ELEMENT patientinfo      - - (name, visit*)>
<!ELEMENT name             - - (#PCDATA)>
<!ELEMENT visit            - - (%timeandplace;,doctor?,problem)>
<!ELEMENT (doctor,problem) - - (#PCDATA)>
```

The parameter entity is used like an element in the DTD, but the names of parameter entities are not used in the document instance.

The elm (enables lucid models) tree diagrams developed by Maler and el Andaloussi [MeA96] and represented in Appendix 1 define tree diagrams for operators of the content model. Figure 2 depicts the previous DTD as an elm tree diagram.

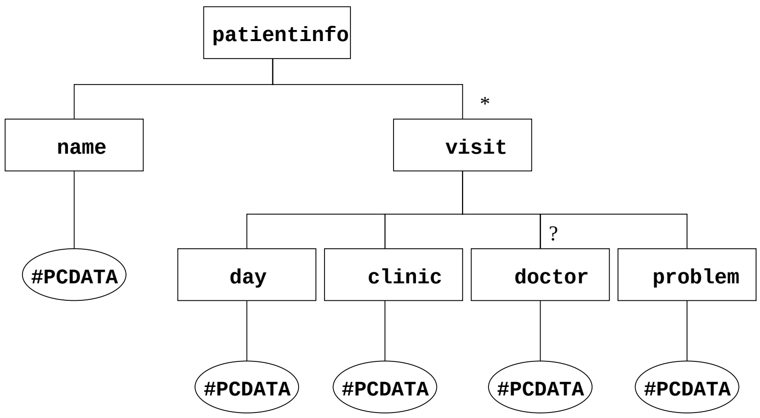


Figure 2: The elm tree diagram of the DTD

A document instance according to this DTD contains the markup which uses the start and end tags. For example, the start tag for the element patientinfo is <patientinfo> and the end tag < /patientinfo>. The instance would be as follows.

```
<patientinfo><name>John Hui</name><visit><day>21.12.1998</day><clinic>...
</clinic><doctor>...</doctor><problem>...</problem></visit><visit>...
</visit>...</patientinfo>
```

3 Analysis and design

Our target was to analyze carefully where in the organization the information of the patient record is utilized and in which order, what is the importance of the information and when, where and why the information about the patient is gathered to the patient record. In the definition of the structure of the electronic patient record we utilized Unified Modeling Language (UML) [Boo98]. In the UML, the analyst defines *use cases*, *class diagrams*, *state diagrams* and *interaction diagrams* [BJR99, JBR99]. We felt, however, that the UML:

- does not show the boundaries of the system,
- does not give an overview of the functionality of the system, and
- is insufficient for the description of the flow of the information in the organization.

The use case diagram gives an overview of the interaction with the system not between the actors. Each use case is described giving an exact information about the actors, pre- and post-conditions, special requirements, alternative scenarios (primary, secondary and exception handling), uses and extends relationships. From these data the analyst gets a lot of information for designing user interfaces and methods. However, the overview of the flow of information in the organization is missing. One possibility to describe this is business processes [Ber96]. This was not suitable, because the Kuopio University Hospital has an organization with departments. Hence, we utilized the traditional *data flows* [You89] in describing how the patient record is used and transfered in the hospital clinic. It was natural to ask from doctors, nurses and secretaries the following type of questions:

- What information do you need in the patient ward and in the developing of the action?
- What are the requirements to store the information?
- Why is the information important?
- From where does the information come?
- Where does the information go?
- What problems do you have?
- What development ideas do you have?

3.1 Data flow in the clinic

The work started by making a careful analysis with the staff of the outpatient ward of the Department of Pediatrics in Kuopio University Hospital. The structured data flow analysis [You89] describes using several diagrams in different levels how the patient record is used in a hospital clinic and how it is transported from one hospital clinic to another.

The context diagram is a good tool for describing the boundaries of the information system. What belongs to the target, for example the clinic considered, and which are the external targets that send information to the target. Figure 3 describes the persons or organizations which have connections to a hospital clinic when patient comes to the clinic and when the patient record is processed. It also shows how the patient record and other involved documents are handled.

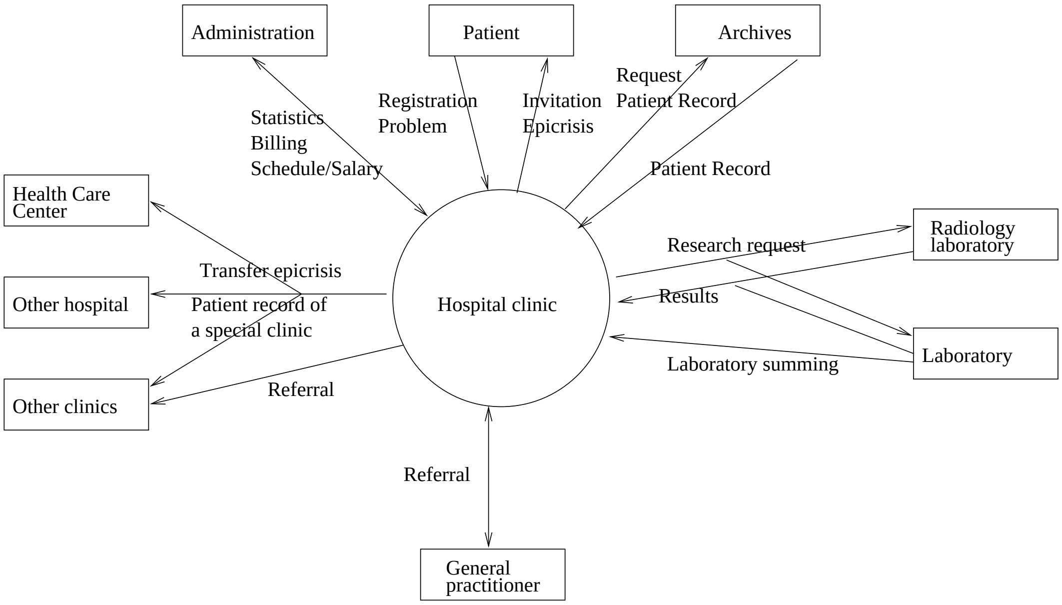


Figure 3: The context diagram of the hospital clinic

A more detailed description was made for the hospital clinic and represented in Figure 4 as an overview data flow diagram. The clinic gets the patient record and some involved documents from outside organization. Thereafter, the patient record is utilized and information is inserted into it in the work posts inside the clinic. The problem here is that the patient record is needed, perhaps even at the same time, in many places inside the clinic, inside the organization or inside the health care area. The secretaries do not always know where the patient records are. They cannot find them sufficiently fast. This causes that, for example, laboratory tests are performed many times increasing the costs.

Our analysis continued by describing in more detail the actions involved the patient record in various working posts of the clinic which are mainly responsible on the processing of the patient record.

We noticed that the doctors and nurses felt the data flow analysis as a natural way to describe their work. They felt it easy and natural to describe what information they need in taking care of the patients. They also felt that the diagrams were sufficiently easy to read and understand and gave them a lot of information about their actions.

3.2 Conceptual modeling of the patient record

We modeled the conceptual content of the patient record with the use of UML class diagrams [BJR99]. Here we depicted the facts collected while interviewing the nurses, doctors and secretaries. We also used all of the paper forms from the clinics. Our purpos was to increase uniformity, reusability and comprehensibility of the system. At the same time we took into the consideration the queries that the doctors and nurses desired to ask from the electronic patient record data. In the following subsections we first define the concepts used in UML diagrams and then give our analysis results.

3.2.1 Object-oriented approach and UML diagram

According to the object-oriented approach the concepts of the real world are *objects*. The patient record, clinic, blood test, etc. are examples about objects. Each object possesses an *identity*, a *state* (defined by the values of its *attributes*), and a *behavior* (methods). An attribute of an object can be an object. This leads to the *aggregation relationship* between objects. The relationship of the aggregation may be one-to-one, one-to-many and many-to-many (the last one is changed into two one-to-many relationships).

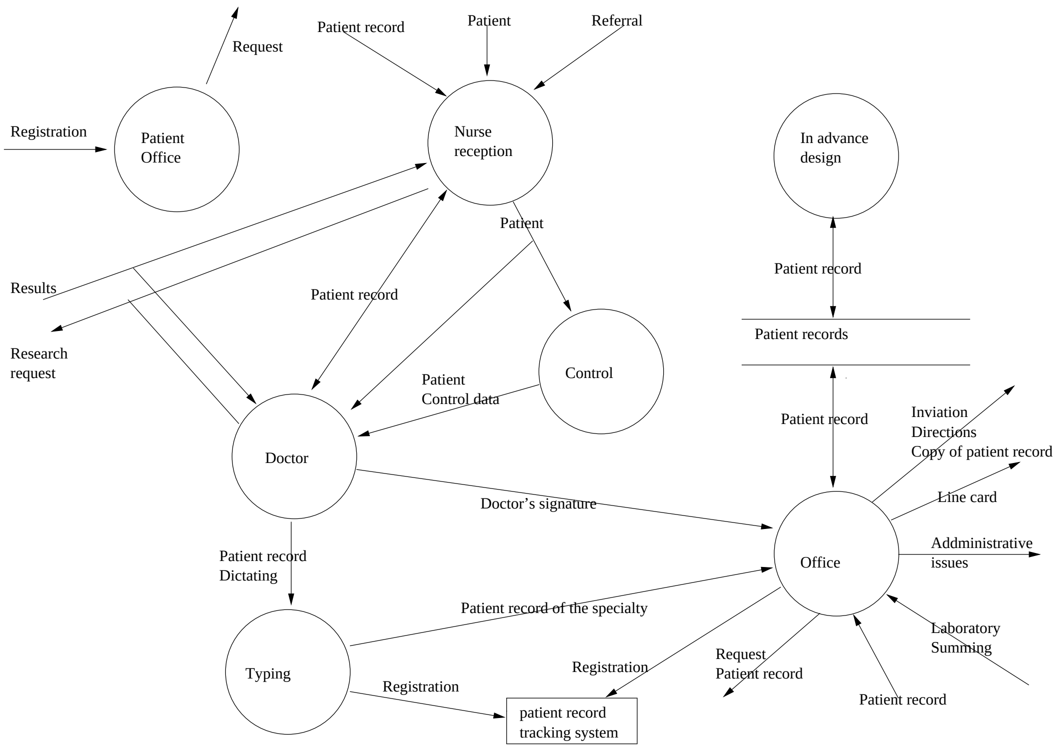


Figure 4: The overview data flow diagram in the hospital clinic

The objects are defined in *classes* which form a class hierarchy. The objects sharing a common definition belong to the same class, while objects resembling each other belong to the same class hierarchy. The class hierarchy allows subclasses to inherit properties, i.e. attributes and methods, from super classes. The super class is a *generalization* of its subclasses. *Concrete classes* define objects. Other classes are *abstract classes* and they are used for the definition of the properties that subclasses inherit. The class hierarchy enables the analysis of the similarity of the objects.

The concepts are usually described using UML class diagram [BJR99]. Figure 5 presents as an example a part of the UML class diagram of the simplified patient record with the most important concepts and notations of the object-oriented approach.

3.2.2 UML diagrams for a patient record

In the conceptual modeling phase we first concluded that the patient record class has the structure represented in Figure 6. According to this class diagram, each patient record object consists of the identification data, the medical record and the nursing record. Additionally, the patient class is associated to the patient class as a purpose to define that each patient has a patient record.

Our analysis continued by defining the three subclasses of the patient record class. The class diagram of the identification data is represented in Figure 7. To register the personal information of a patient there exists a set of identification data forms. The forms collect information about the patient, her or his relatives as well as about the possible accident. These are defined as subclasses for a patient, relative and accident information. Patient and relative classes are specifications for the abstract person class and inherit its attribute information.

The class diagram of the medical record class is represented in Appendix 2.1. According to the diagram, the medical record consists of continuing medical records of various special clinics as well as medical certificates, blood group and blood transfusion information and a summary form with a set of summary markings. Summary markings are derived from the visit notes of various clinics. The blood

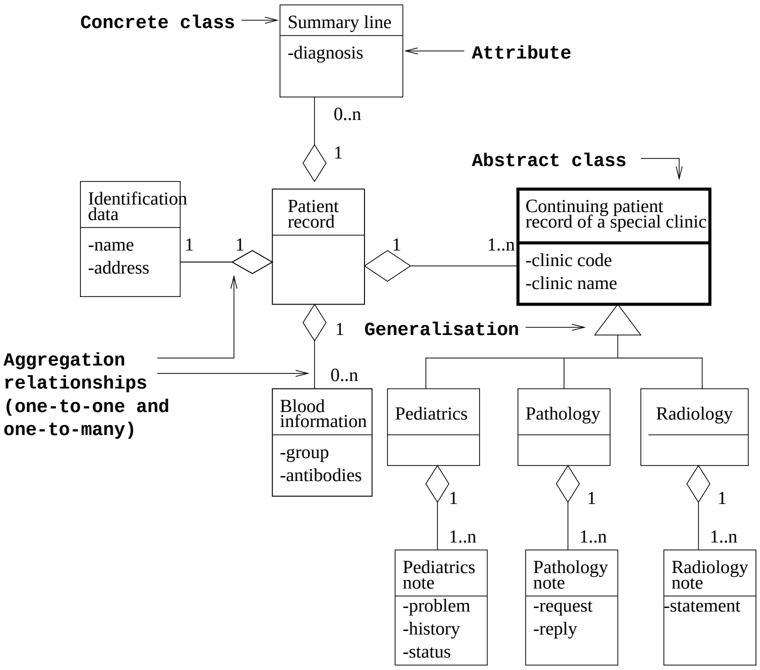


Figure 5: An UML class diagram

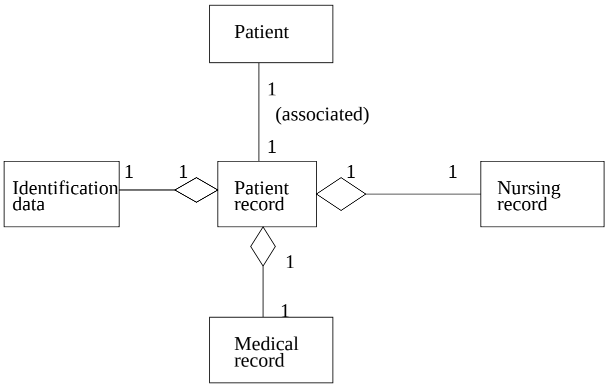


Figure 6: Class diagram of the patient record

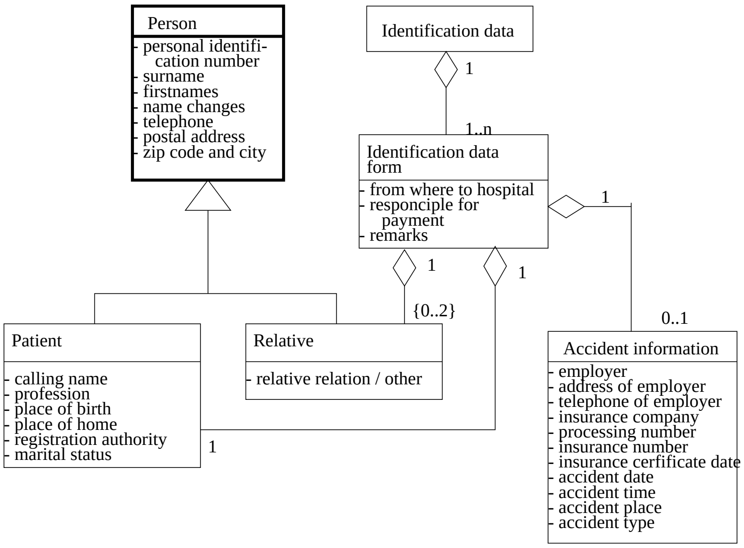


Figure 7: The class diagram of the identification data.

information consists of laboratory, subscriber and newborn's information.

The class diagram of continuing medical of the different special clinics is represented in Appendix 2.2. It specifies the structure of notes of various clinics. Certain clinics use the same structure which are defined by the general note type class. A negotiation question is, are the differences between clinics needed or could the clinics work in more standardized way using similar notes. Common information of all the clinics is defined in the general information of a special clinic class and the confirmed note class. The last step is to optimize the class hierarchy. This problem is considered, for example, in [PK97, CW98, Eer99].

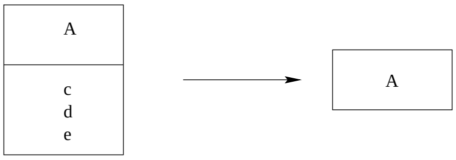
The class diagram of the nursing record is represented in Appendix 2.3. As this diagram depicts, the nursing record consists on the nursing plan of the first aid, the diet information, entrance interview made by the nurse, the table of nursing and the continuing nursing plan. The continuing nursing plan is made on a basic form and it contains a set of nursing plan notes. Each note consists of the nursing needs information and problems, nursing plan as well as the progress, control and rating of the nursing actions. The class has the measurements class as a subclass.

4 Transformation of UML diagrams to elm tree diagrams

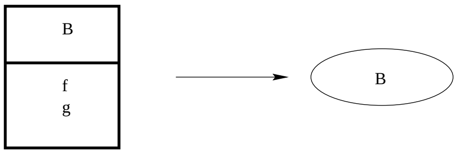
In the design of the SGML DTD, reference [MeA96] advises the designer to "classify the potential semantic components (of a document) according to your sense of their similarities". In our method these similarities can be found in UML diagrams. What we need are the generation rules from the UML diagrams to SGML DTD. For this process, we have defined three sets of rules, the *transformation rules*, *placement rules* and *processing rules*, to generate semiautomatically the elm tree diagrams of the DTD from the UML class diagrams. Transformation rules define how an elementary element (for example, class, attribute, etc.) of the UML diagram is transformed to a node of elm tree diagram. The placement rules define how attributes as well as aggregate and generalization relationships specify the location of generated nodes in the tree. The processing rules define how the hierarchies of classes are processed. The first two groups of rules we represented already in [KEP+99]; in this research we develop them further and add the rules to transform the class hierarchies.

TRANSFORMATION RULES:

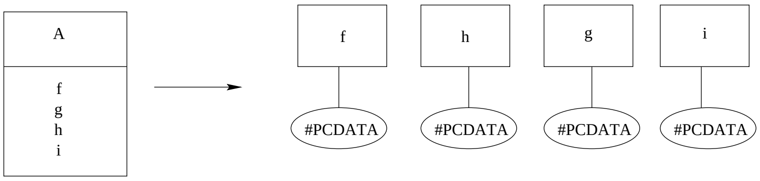
Rule 1: The concrete class generates an element.



Rule 2: The abstract class generates a parameter entity.

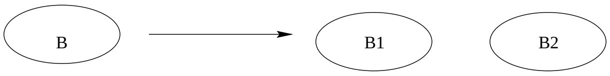


Rule 3: The attribute generates a content element.



The content of the attribute may be, in addition to the `#PCDATA` type, also `RCDATA`, `CDATA` or other data type as required.

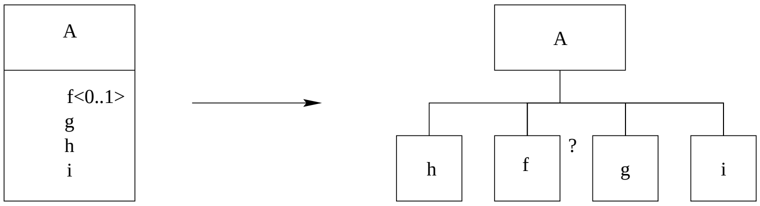
Rule 4: The parameter entity may be divided into many parameter entities.



If all of the attributes of an abstract class are used in all of the subclasses in the same order, the rule 4 is not needed. However, if the attributes of an abstract class are used in various sequences we need the rule 4. In practise, an abstract class having the attributes is first transformed into a parameter entity which is then divided into as many parameter entities as there are separate sequences of attributes. The names of new parameter entities are generated from the original parameter entity name.

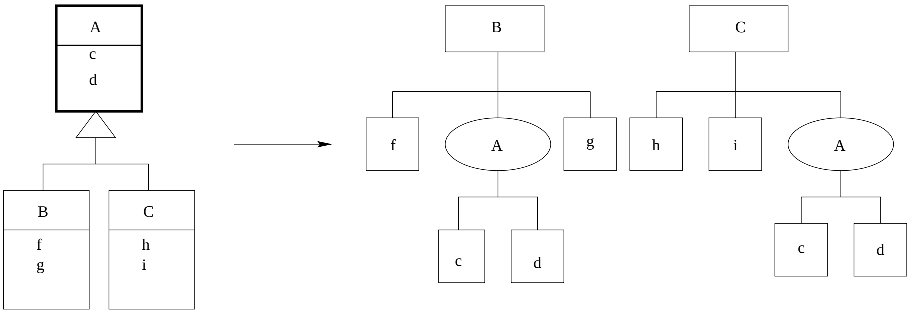
PLACEMENT RULES:

Rule 5: The element corresponding to the attribute is placed into the content model of the element or the parameter entity corresponding to its class. The order of the attribute among other attributes is selected. The occurrence of the attribute is indicated.



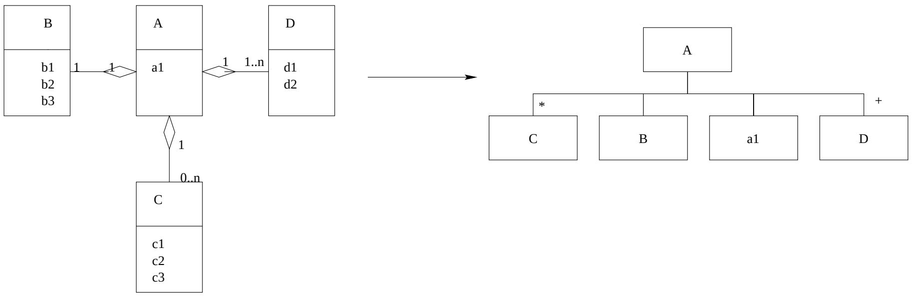
Since UML diagrams do not consider the order of the attributes of a class, this information is added to elm tree diagrams. The sequential order of subelements is from left to right in the tree diagram. In the UML diagram, after the name of the attribute the optionality of attributes is defined by the numbers `<0..1>` and the multiplicity of the attributes is defined by the numbers `<0..n>` and `<1..n>`. In the elm tree diagrams we use the notations represented in the explanation of Rule 7.

Rule 6: The parameter entity corresponding to the super class in the generalization relationship is placed into the content models of the elements or parameter entities corresponding to each of the subclasses. The place of the parameter entity among other elements or parameter entities is selected.



The rule is applied also in the case of the multiple inheritance.

Rule 7: The part in the aggregate relationship is placed as an element or a parameter entity into the content model of the element or parameter entity corresponding to the aggregate. The order of elements corresponding to the parts is selected.

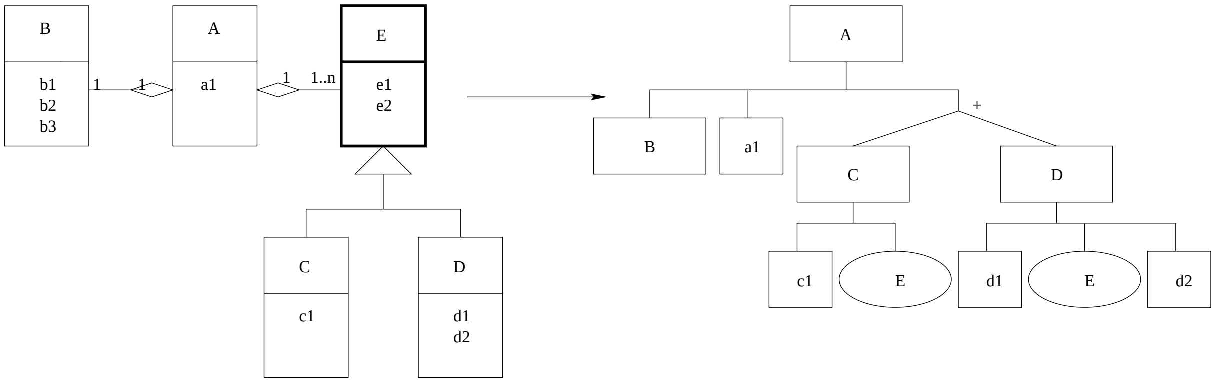


The aggregation relationship is defined in the UML diagrams using numbers in both ends of the relation notation. These numbers correspond to the notations of elm tree diagram as follows.

UML diagram	Elm tree diagram
1	no character
0..1	?
0..n	*
1..n	+

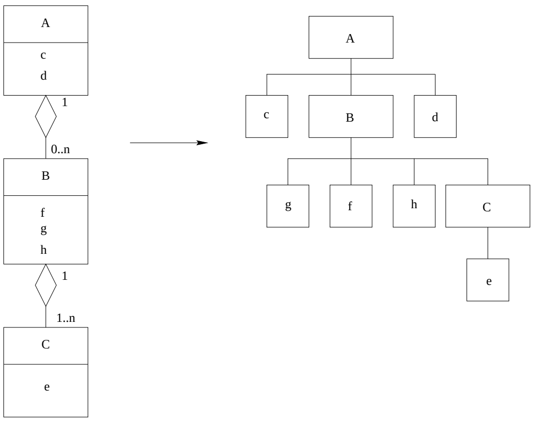
PROCESSING RULES:

Rule 8: The generalization rule 6 is applied before aggregation rule 7.

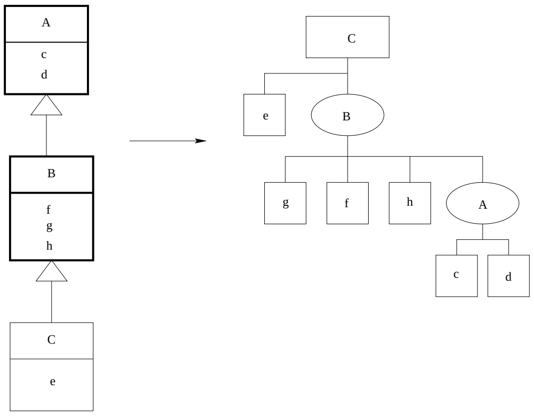


Rule 9: Processing of the aggregation hierarchies are made top-down and classification hierarchies are made bottom-up.

Aggregation hierarchy:



Classification hierarchy:

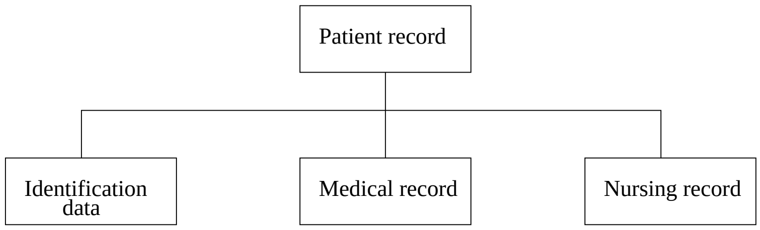


5 The DTD for the patient record

The application of the rules to the represented UML diagrams generates the elm tree diagrams represented in Figures 8-9 and in Appendix 3. The corresponding DTD rules for these trees are represented in Appendix 4.

However, two issues concerning the application of the rules should be mentioned. First, using the rules in Section 4, it is possible to generate the DTD which preserves relationships and naming of class and attribute names defined in UML diagrams. However, although it is natural to use explanatory, and usually long names in UML diagrams, the same names are not suitable in the final DTD. When selecting names for the elements we have used the naming rules represented in [ISI99]. Second, whenever several elm tree nodes corresponding to the attributes or subclasses are placed as subnodes, we have always used a sequential order for them. Other possibility would have been to connect the attributes using the set operator as a purpose to allow their free order.

Figure 8 represents the elm tree diagram and the corresponding DTD rule generated from the medical record class diagram in Figure 6.




```
<!ELEMENT PatientRec      - -  (IdentificationData,
                                MedicalRec,
                                NursingRec)>
```

Figure 8: The elm tree diagram and the DTD rule corresponding to the patient record class

Figure 9 depicts the elm tree diagrams generated from the identification data class diagram in Figure 7. The DTD rules which correspond to the elm tree diagram are represented in Appendix 4.

The Appendix 3 represents the the generated elm tree diagrams for the UML diagrams in Appendix 2. From a class diagram several tree diagrams are generated to prevent the tree diagrams to come too large.

6 Related work

Maler and el Andaloussi [MeA96] represent the DTD development as part of a global SGML project. The aim of the design project is the same as in our paper: to setup a complete SGML-based document production system. The analysis and modeling method consists of ten phases. First, during the *analysis phase* the designer identifies potential needs, classifies them into logical categories, validates them and selects the needs that the markup models should address. Second, in the *structure modeling phase* the designer builds models for the document hierarchy, information units and data-level elements. Third, during the *evaluation phase* she or he populates the models, makes connections within the models and from the outside world, and validates all the models for completeness. The designer uses her or his own knowledge or the knowledge of the user about the current application or paper-form documents created by the system for the decisions. The designer fills a *set of forms manually* as a purpose to find out the structure models. Various forms list all possible elements, all candidate elements, etc. The structure

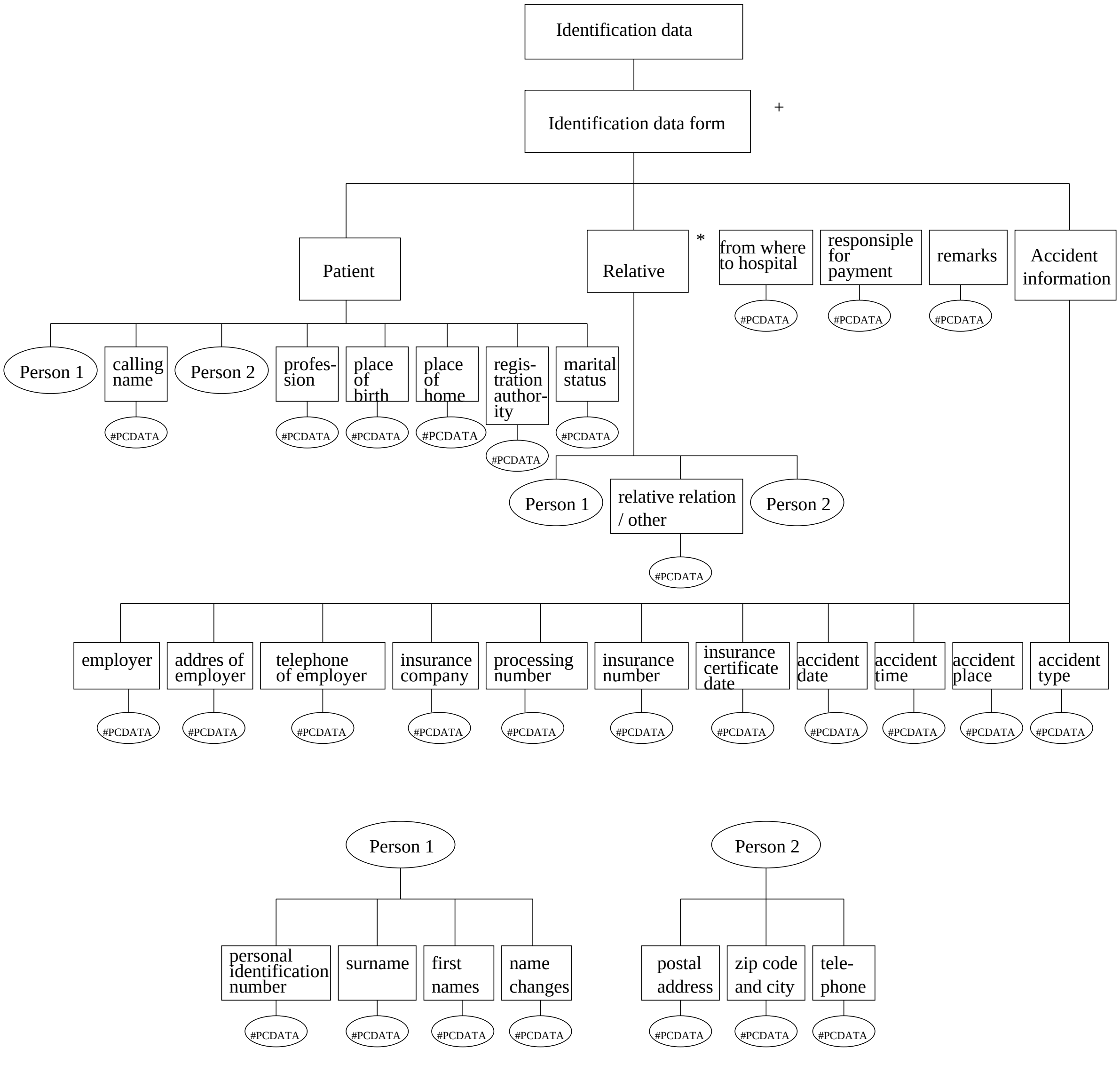


Figure 9: The elm tree diagram of the identification data element

models are depicted as elm tree diagrams to describe the elements and their relationships and to have a possibility to discuss with the users about the details of the structure.

Salminen, Kauppinen and Mehtovaara [SKL97, SLK96] have represented the structure definition method based on OOA (Object-Oriented Analysis) method of Shlaer and Mellor [SM92]. The method is especially developed into a sitation when several documents have relationships and all of them need to be structured. The design is divided into the following phases: the *domain definition* to choose the candidate documents, the *object modeling phase* to identify and describe objects, their attributes and relationships, the *state model generating phase* to represent a condition or situation of the object during its life cycle, and the *content modeling phase* to provide for each document a structure description, a reuse table, an acquisition and usage table. The document objects and their relationships are described in a document-relationship diagram. A state model is described by providing a state transition diagram for each document object having interesting dynamic behavior. Similarly as in our method, the object-oriented approach offers concepts and notations especially for conceptual modeling of the domain. The reusability and similarity of the concepts is carefully considered.

In the ISIS European XML/EDI Pilot Project [IXPP99] the aim is to apply XML technology [Wor98] into the electronic message interchange of business data in statistics, transport and health care sectors. The authors of the project paper [ISI99] have considered the definition of generalized message descriptions with the use of the UML diagrams. Parallel to our work, they have developed the mapping rules from the UML diagram to XML DTD declatations. The common part of their and our results are consistent. However, the viewpoints are different. When they concentrate on the description of a message using UML diagrams, we consider more carefully the analysis and design phases, semantics of electronic patient record. Although the mapping rules are very similar, we present also the total process description for changing UML class diagrams to DTD tree representation. Further, we consider the modeling of the alternative classification (even the multiple inheritance) and the definition of the order of components.

7 Conclusion

We have presented a method to generate an SGML DTD for a large structured document which is used by many persons in a big organization. The design phase represents the information flow using data flow diagrams and the conceptual modeling of the content of a document using object-oriented UML class diagrams. Using a set of developed rules it is possible to generate the elm tree diagrams corresponding to the UML class diagrams. The DTD rules are easily written from the elm tree diagrams. This approach produces a modular DTD by grouping rule definitions to a certain UML class diagram together. If we restrict the elm tree diagrams such which define the content models allowed according to the XML standard our method produces an XML DTD.

The method gives an example and a pragmatic new way of action process to design structured documents. The generation rules to define the DTD from UML diagrams can be used in any information systems. The results can be used for searching the possibilities to standardize the content of the electronic patient record.

References

[Ben96] T. Benson. SGML, HTML and EPR, application of the standard generalized markup language (SGML) in electronic patient records. Available http://www.mcis.duke.edu/standards/HL7/committees/sgml/references/sgmlepr.htm, 1998, 1996.

[Ber96] A. Berztiss. *Software methods for Business Reengineering*. Springer-Verlag, Berlin, 1996.

[BJR99] G. Booch, I. Jacobson, and J. Rumbaugh. *The Unified Modeling Language Users Guide*. Addison Wesley, Reading, Massachusetts, 1999.

[Boo98] G. Booch. *UML Users Guide*. Addison Wesley, Chicago, Il, 1998.

[BPD96] T. Bürkle, H.-U. Prokosch, and J. Dodeck. Steps towards integration of knowledge based functions into a hospital system. In J. Brender, J.P. Christensen, J.-R. Scherrer, and P. McNair, editors, *Medical Informatics Europe '96*, pages 286–290. IOS Press, 1996.

[Bra94] V. Bradley. Innovative informatics CPR: Computerized patient record. *Journal of Emergency Nursing*, 20:230–232, 1994.

[Car94] P.C. Carpenter. The electronic medical record: perspective from Mayo Clinic. *International Journal of Bio-Medical Computing*, 34:159–171, 1994.

[CW98] J.-L. Chen and F.-J. Wang. An inheritance flow model for class hierarchy analysis. *Information Processing Letters*, 66:309–315, 1998.

[Eer99] A. Eerola. A disciplined approach to the maintenance of the class hierarchy. Technical Report Raport A/1999/2, University of Kuopio, Department of Computer Science and Applied Mathematics, 1999.

[Gol90] C.F. Goldfarb. *The SGML Handbook*. Oxford University Press, Oxford, UK, 1990.

[ISI99] ISIS European XML/EDI Pilot Project. *Mapping from UML Generalised Message Descriptions to XML DTDs*, 1999. Available http://www.tieke.fi/isis-xmledi/D2/D22.htm.

[ISO86] ISO 8879, ISO, Geneva. *Information Processing – Text and Office Systems – Standard Generalized Markup Language (SGML)*, 1986.

[IXPP99] ISIS European XML/EDI Pilot Project. ISIS XML/EDI Project Website. Available at http://www.tieke.fi/ovt/, 1999.

[JBR99] I. Jacobson, G. Booch, and J. Rumbaough. *The Unified Software Development Process*. Addison Wesley, Reading, Massachusetts, 1999.

[JG99] B. Jung and J. Grimson. Synapses/SynEx goes XML. In P. Kokol et al., editor, *Medical Informatics Europe '99*, pages 906–911. IOS Press, 1999.

[KEP+99] E. Kuikka, A. Eerola, J. Porrasmaa, A. Miettinen, and J. Komulainen. Design of the SGML-based electronic patient record system with the use of object-oriented analysis methods. In P. Kokol et al., editor, *Medical Informatics Europe '99*, pages 838–841. IOS Press, 1999.

[MeA96] E. Maler and J. el Andaloussi. *Developing SGML DTDs from Text to Model to Markup*. Prentice Hall, Upper Saddle River, New Jersey, 1996.

[PGR+97] D. Pitty, C. Gordon, P. Reeves, A. Capey, P. Vieyra, and T. Rickards. The place of SGML and HTML in building electronic patient records. In C. Pappas, N. Maglasveras, and J.-R. Scherrer, editors, *Medical Informatics Europe '97*, pages 329–333. IOS Press, 1997.

[PK97] A Putkonen and M. Kiekara. A case-tool for supporting navigation in the class hierarchy. *Software Engineering Notes*, 22(1):77–84, 1997.

[PSRL94] D. Paty, D. Studney, K. Redekop, and F. Lublin. MS COSTAR. A computerized patient record adapted for clinical research purposes. *Annals of Neurology*, 36:S134–S135, 1994.

[Sco98] Scott & White Hospital and Clinic. *Scott and White Electronic Medical Record Project*, 1998. Available http://swinfo.tamu.edu/news/ options/july98/empx_his.htm, copied January 26, 1999.

[SKL97] A. Salminen, K. Kauppinen, and M. Lahtovaara. Towards a methodology for document analysis. *Journal of the American Society for Information Science*, 48(7):644–655, 1997.

[SLK96] A. Salminen, M. Lehtovaara, and K. Kauppinen. Standardization of digital legistive documents. In M.S. Lynn, editor, *Proceedings of the 29th Annual Hawaii International Conference on System Sciences*, pages 72–81, Hawaii, USA, 1996. IEEE Computer Society Press.

[SM92] S. Shlaer and S.J. Mellor. *Object Lifestyles – Modeling the World in States*. Yourdon Press, Englewood Cliffs, NJ, 1992.

[TMO+97] H. Tadeka, T. Matsumura, T. Okada, S. Kuwata, and M. Ioue. Approach to establish an electronic patient record system in an intelligent hospital. In *Proceedings of IMIA-WG4: Common Security Solutions for Commmunication Patient Data*, 1997. Available http://dchps.hops.med.osaka-u.ac.jp/home/hp-info/paper/epr2/body.htm, copied January 4, 1999.

[TMOK97] H. Tadeka, T. Matsumura, T. Okada, and S. Kuwata. Dynamic template driven data entry system for an EPR system. In *Proceedings of Toward an Electronic Patient Record'97, Vol. 2*, pages 139–143, London, 1997.

[TW96] B. Travis and D. Waldt. *The SGML Implementation Guide*. Springer-Verlag, Berlin, 1996.

[vH94] E. van Herwijnen. *Practical SGML*. Kluwer Academic Publisher, Dordrecht, 1994.

[Wor98] World Wide Web Consortium (W3C). *Extensible Markup Language (XML) 1.0*, 1998. Available at URL: http://www.w3.org/TR/1998/REC-xml-19980210.

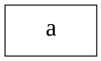
[YOO+98] H. Yoshihara, K. Obe, K. Ohashi, R. Yamamoto, S. Yamazaki, Y. Hirose, K. Matsui, T. Hishiki, Y. Yamashita, M. Kimura, K. Minagawa, and H. Oyama. Standardization of exchange procedures of clinical information using Medical Markup Language (MML). In B. Cesnik, A.T. McCray, and J.-R. Scherrer, editors, *MEDINFO 98 Proceedings*, page 129, 1998.

[You89] E. Yourdon. *Modern Structured Analysis*. Yourdon Press, Englewood Cliffs, NJ, 1989.

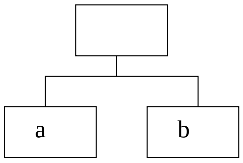
Appendix 1: The elm tree diagrams for SGML content models

CONTENT MODEL — TREE DIAGRAM

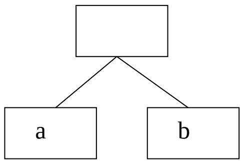
(a)



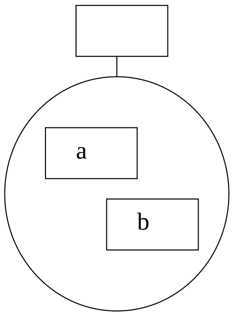
(a,b)



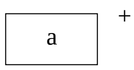
(a|b)



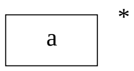
(a&b)



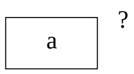
(a+)



(a*)

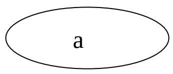


(a?)



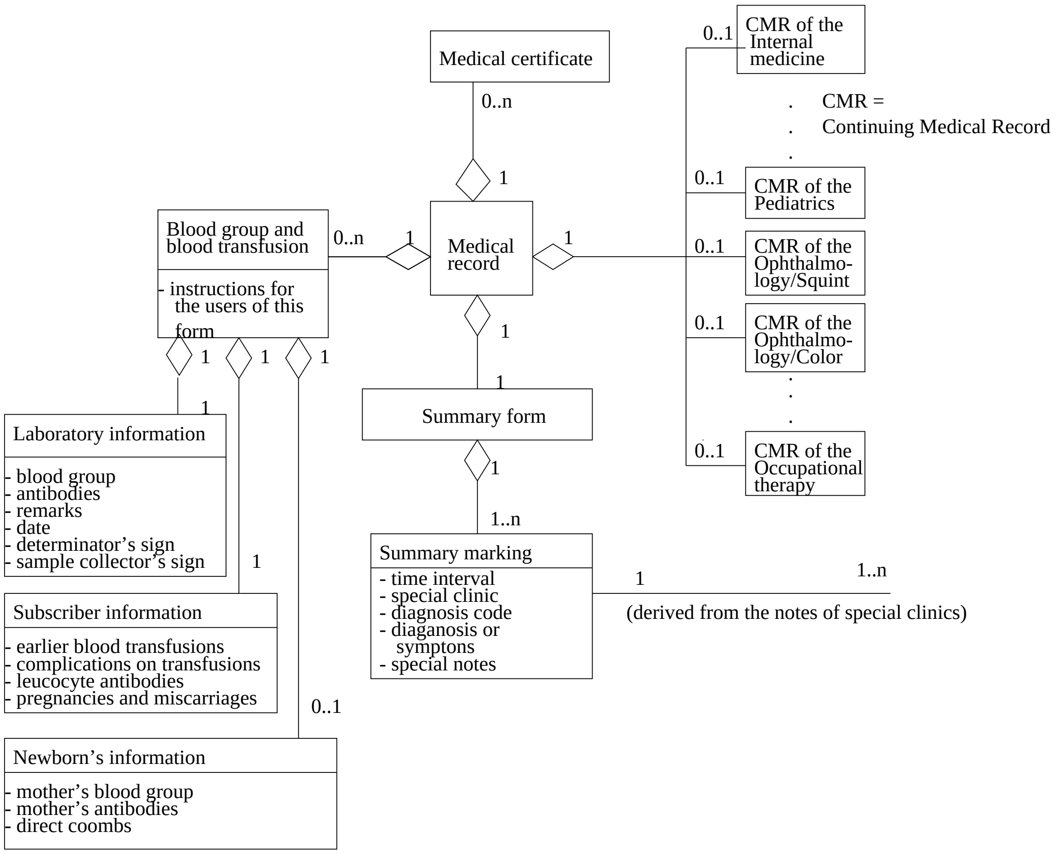
(#PCDATA)

(%a;)

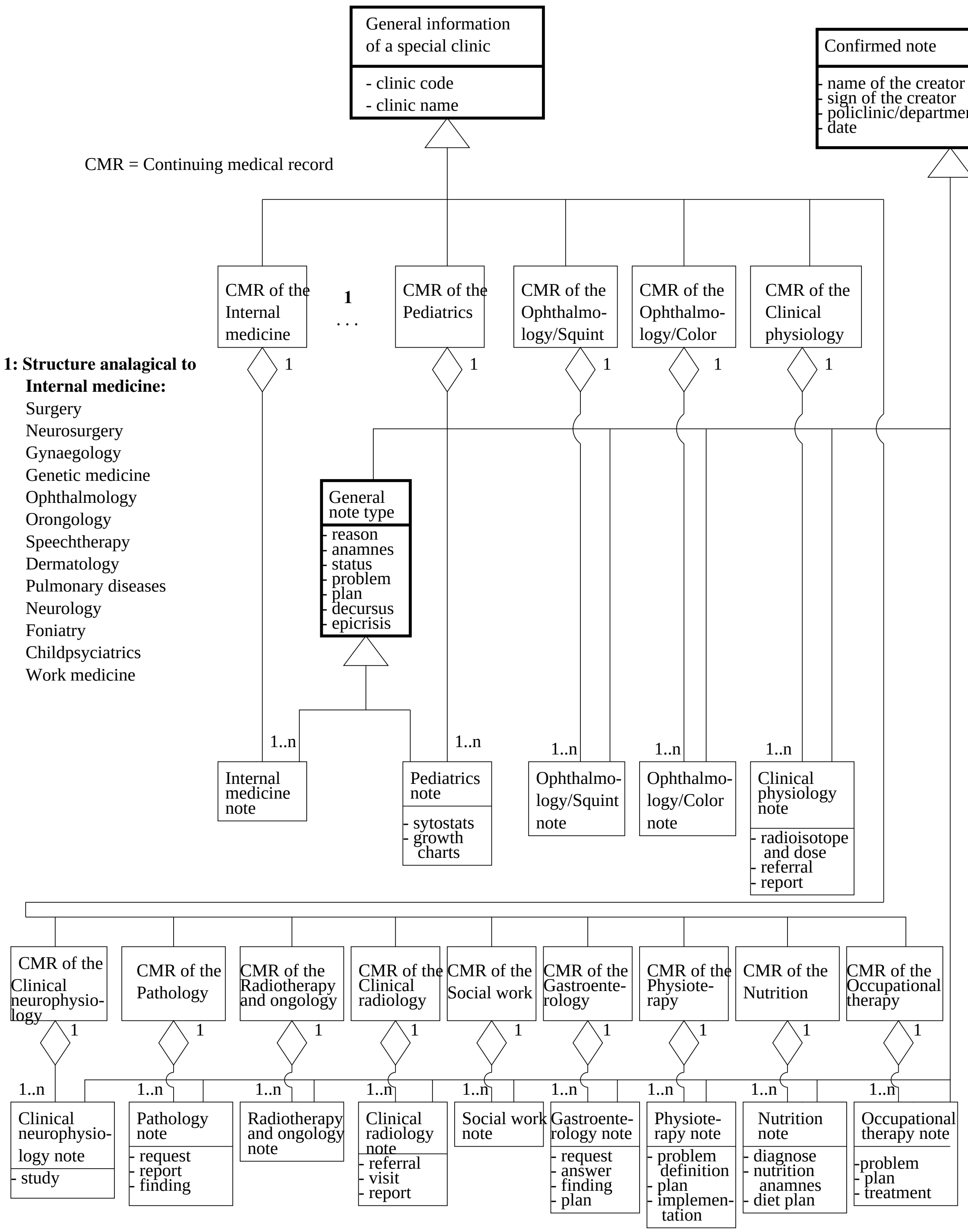


Appendix 2: Class diagrams

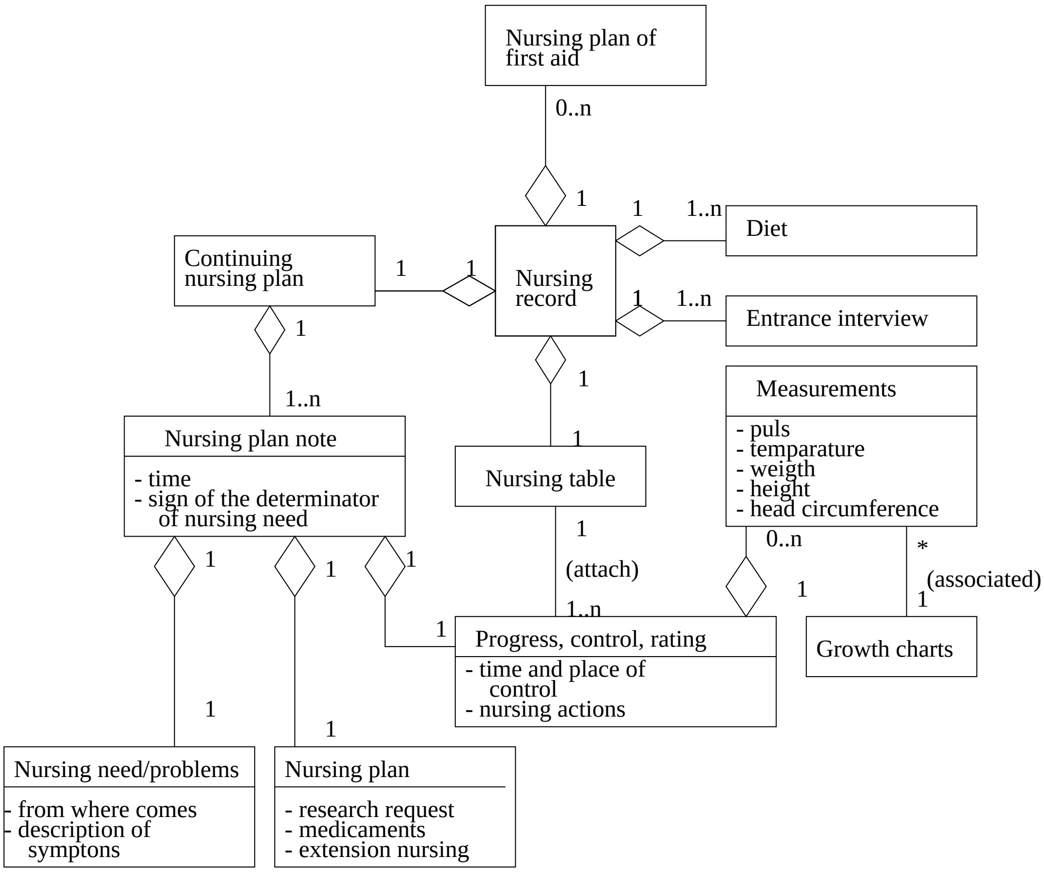
2.1 The class diagram of the medical record



2.2 The class diagram of the continuing medical records of various special clinics

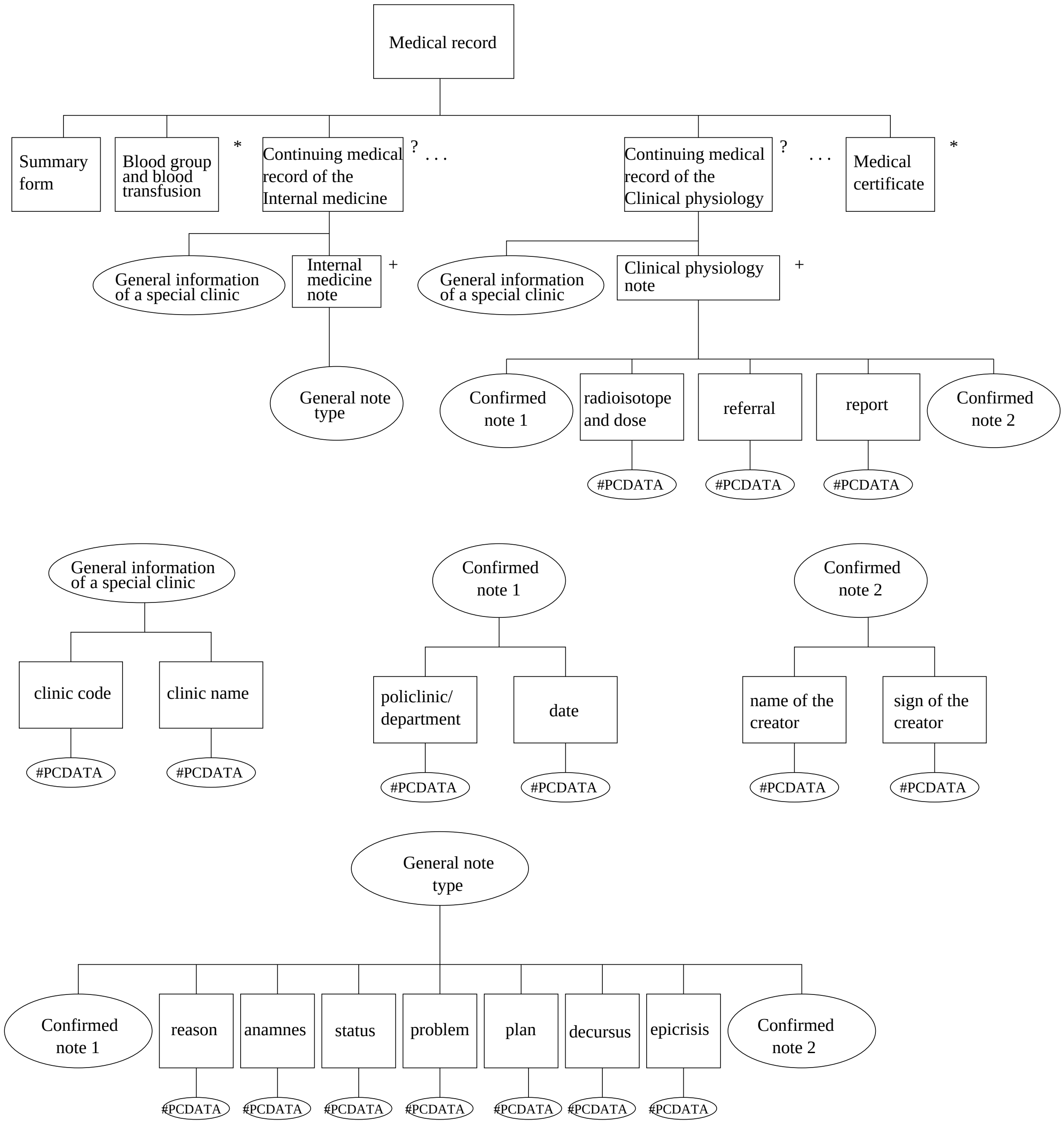


2.3 The class diagram of the nursing record

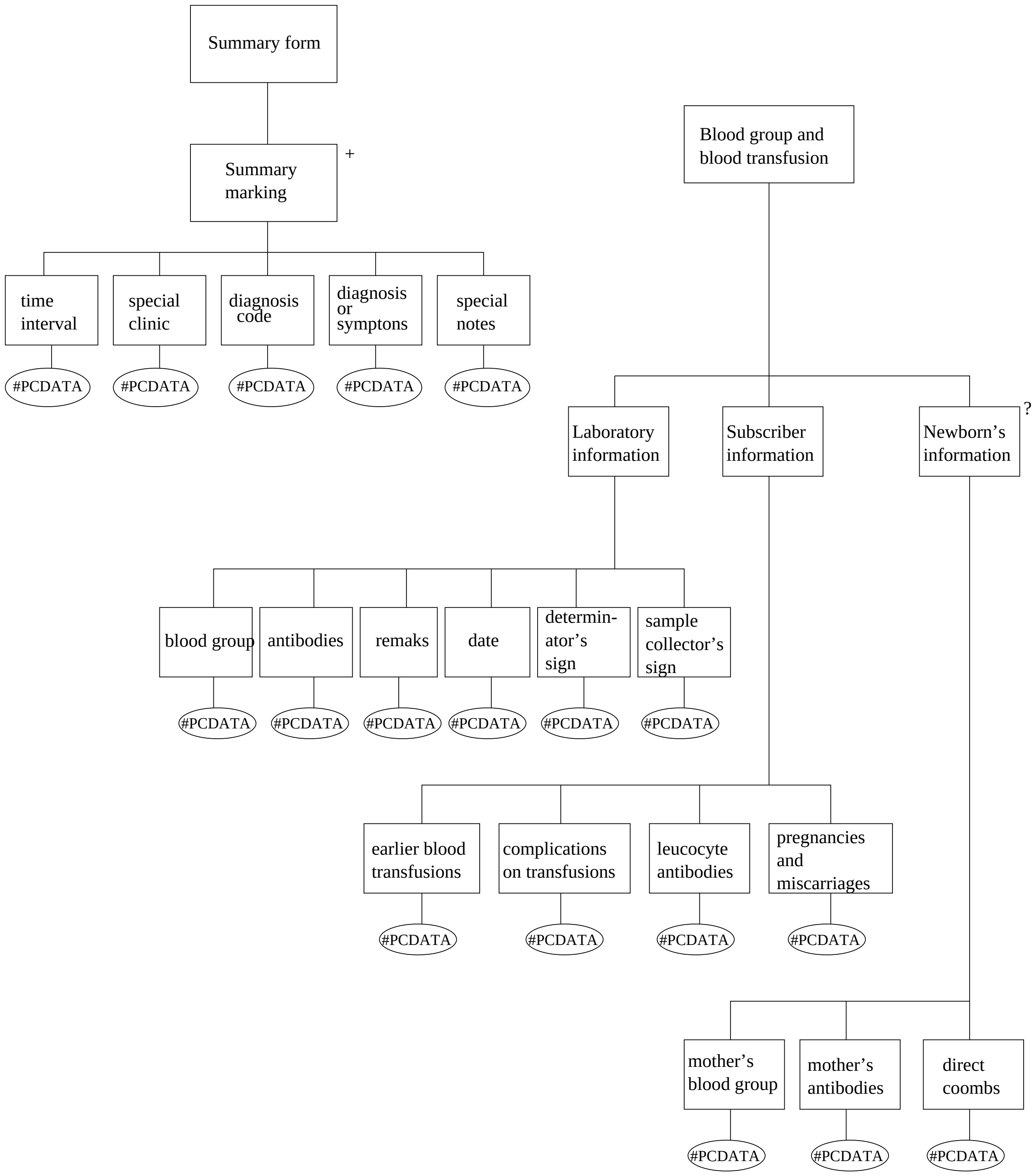


Appendix 3: Elm tree diagrams

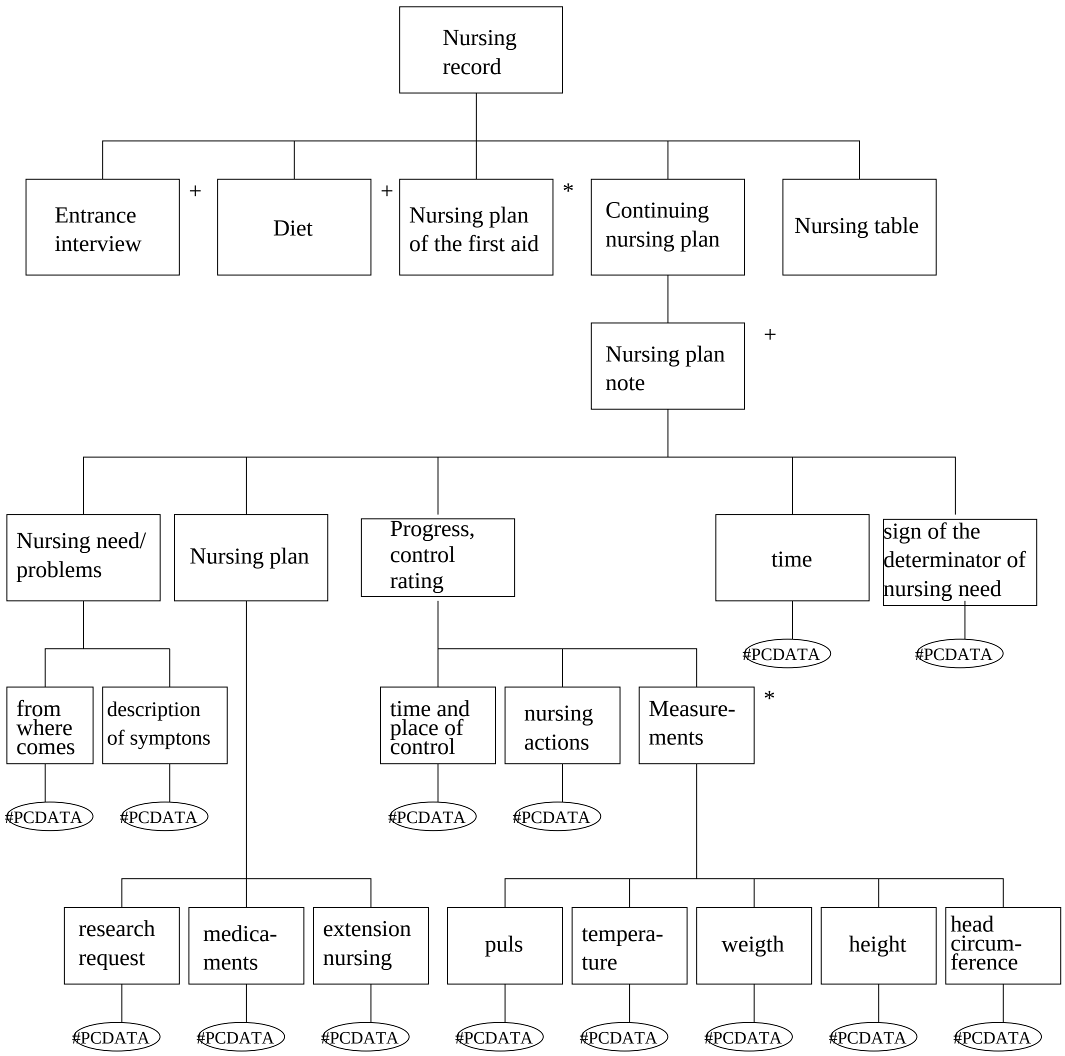
3.1 The elm tree diagram of the medical record



3.2 The elm tree diagrams of the summary form and the blood group and blood transfusion



3.4 The elm tree diagram of the nursing record



Appendix 4: The DTD of a patient record

```
<!DOCTYPE PatientRec [
<!ENTITY amp CDATA "&">
<!ENTITY auml CDATA "&auml;">
<!ENTITY ouml CDATA "&ouml;">
<!ENTITY aring CDATA "&aring;">
<!ENTITY lt CDATA "&lt;">
<!ENTITY gt CDATA "&gt;">
<!ENTITY uuml CDATA "&uuml;">
<!ENTITY eacute CDATA "&eacute;">

<!ELEMENT PatientRec    - -  (IdentificationData,
                                 MedicalRec,
                                 NursingRec)>


<!-- structure Definitions of Identification Data -->
<!ELEMENT IdentificationData
                        - -  (IdentificationDataForm+)>
<!ELEMENT IdentificationDataForm
                        - -  (Patient,Relative*, FromWhereHospital,
                      PaymentResponsible, Remarks,
                              AccidentInfo?)>

<!ELEMENT (FromWhereHospital,
           PaymentResponsible,
           Remarks)     - -  (#PCDATA)>

<!ENTITY % Person1 "PersonalIdentificationNumber,
                   Surname, FirstNames, NameChanges">
<!ENTITY % Person2 "PostalAddress, ZipCodeCity, Telephone">
<!ELEMENT (PersonalIdentificationNumber,
           Surname,
           FirstNames,
           NameChanges,
           Telephone,
           PostalAddress,
           ZipCodeCity) - -  (#PCDATA)>

<!ELEMENT Patient       - -  (%Person1,CallingName,
                              %Person2,
                              Profession,BirthPlace,HomePlace,
                              RegistrationAuthority,
                              MaritalStatus)>
<!ELEMENT (CallingName, Profession,
           BirthPlace,
           HomePlace,
           RegistrationAuthority,
           MaritalStatus)
                        - -  (#PCDATA)>

<!ELEMENT Relative      - -  (%Person1,
```

```
                              RelativeRelationOther,
                              %Person2)>
<!ELEMENT RelativeRelationOther
                        - -  (#PCDATA)>

<!ELEMENT AccidentInfo  - -  (Employer,
                              EmployerAddress,
                              EmployerTelephone,
                              InsuranceCompany,
                              ProcessingNumber,
                              InsuranceNumber,
                              InsuranceCertificateDate,
                              AccidentDate,
                              AccidentTime,
                              AccidentPlace,
                              AccidentType)>
<!ELEMENT (Employer,EmployerAddress,
           EmployerTelephone,
           InsuranceCompany,
           ProcessingNumber,
           InsuranceNumber,
           InsuranceCertificateDate,
           AccidentDate,AccidentTime,
           AccidentPlace,
           AccidentType) - -  (#PCDATA)>

<!-- structure Definitions of the Medical record -->
<!ELEMENT MedicalRec     - - (SummaryForm,
                              BloodGroupTransfusion*,
                              InternalMedicineContinuingMedicalRec?,
                              SurgeryContinuingMedicalRec?,
                              NeuroSurgeryContinuingMedicalRec?,
                              GynaegologyContinuingMedicalRec?,
                              GeneticMedicineContinuingMedicalRec?,
                              PediatricsContinuingMedicalRec?,
                              OphthalmologyContinuingMedicalRec?,
                              OphthalmologySquintContinuingMedicalRec?,
                              OphthalmologyColorContinuingMedicalRec?,
                              OrongologyContinuingMedicalRec?,
                              FoniatryContinuingMedicalRec?,
                              SpeechTherapyContinuingMedicalRec?,
                              DermatologyContinuingMedicalRec?,
                              RadioTherapyOngologyContinuingMedicalRec?,
                              PsychiatryContinuingMedicalRec?,
                              ChildrenPsychiatryContinuingMedicalRec?,
                              NeurologyContinuingMedicalRec?,
                              PulmonaryDiseasesContinuingMedicalRec?,
                              ClinicalPhysiologyContinuingMedicalRec?,
                              ClinicalNeuroPhysiologyContinuingMedicalRec?,
                              ClinicalRadiologyContinuingMedicalRec?,
                              GastroenterologyContinuingMedicalRec?,
                              PathologyContinuingMedicalRec?,
                              SocialWorkContinuingMedicalRec?,
```

```
                              PhysiatricsContinuingMedicalRec?,
                              WorkMedicineContinuingMedicalRec?,
                              MedicalRehabilitationContinuingMedicalRec?,
                              NutritionContinuingMedicalRec?,
                              OccupationalTherapyContinuingMedicalRec?,
                              MedicalCertificate*)>

<!ENTITY % SpecialClinicGeneralInfo
                              "ClinicCode,ClinicName">
<!ELEMENT (ClinicCode,
          ClinicName)    - - (#PCDATA)>

<!ELEMENT InternalMedicineContinuingMedicalRec
                         - - (%SpecialClinicGeneralInfo;,
                              InternalMedicineNote+)>
<!ELEMENT SurgeryContinuingMedicalRec
                         - - (%SpecialClinicGeneralInfo;,
                              SurgeryNote+)>
<!ELEMENT NeuroSurgeryContinuingMedicalRec
                         - - (%SpecialClinicGeneralInfo;,
                              NeuroSurgeryNote+)>
<!ELEMENT MedicalRehabilitationContinuingMedicalRec
                         - - (%SpecialClinicGeneralInfo;,
                              MedicalRehabilitationNote+)>
<!ELEMENT GynaegologyContinuingMedicalRec
                         - - (%SpecialClinicGeneralInfo;,
                              GynaegologyNote+)>
<!ELEMENT GeneticMedicineContinuingMedicalRec
                         - - (%SpecialClinicGeneralInfo;,
                              GeneticMedicineNote+)>
<!ELEMENT PediatricsContinuingMedicalRec
                         - - (%SpecialClinicGeneralInfo;,
                              PediatricsNote+, Sytostats, GrowthCharts)>
<!ELEMENT PsychiatryContinuingMedicalRec
                         - - (%SpecialClinicGeneralInfo;,
                              PsychiatryNote+)>
<!ELEMENT ChildrenPsychiatryContinuingMedicalRec
                         - - (%SpecialClinicGeneralInfo;,
                              ChildrenPsychiatryNote+)>
<!ELEMENT OphthalmologyContinuingMedicalRec
                         - - (%SpecialClinicGeneralInfo;,
                              OphthalmologyNote+)>
<!ELEMENT OrongologyContinuingMedicalRec
                         - - (%SpecialClinicGeneralInfo;,
                              OrongologyNote+)>
<!ELEMENT SpeechTherapyContinuingMedicalRec
                         - - (%SpecialClinicGeneralInfo;,
                              SpeechTherapyNote+)>
<!ELEMENT DermatologyContinuingMedicalRec
                         - - (%SpecialClinicGeneralInfo;,
                              DermatologyNote+)>
<!ELEMENT PulmonaryDiseasesContinuingMedicalRec
                         - - (%SpecialClinicGeneralInfo;,
```

```
                              PulmonaryDiseasesNote+)>
<!ELEMENT NeurologyContinuingMedicalRec
                         - - (%SpecialClinicGeneralInfo;,
                              NeurologyNote+)>
<!ELEMENT FoniatryContinuingMedicalRec
                         - - (%SpecialClinicGeneralInfo;,
                              FoniatryNote+)>
<!ELEMENT WorkMedicineContinuingMedicalRec
                         - - (%SpecialClinicGeneralInfo;,
                              WorkMedicineNote+)>
<!ELEMENT OphthalmologySquintContinuingMedicalRec
                         - - (%SpecialClinicGeneralInfo;,
                              OphthalmologySquintNote+)>
<!ELEMENT OphthalmologyColorContinuingMedicalRec
                         - - (%SpecialClinicGeneralInfo;,
                              OphthalmologyColorNote+)>
<!ELEMENT ClinicalPhysiologyContinuingMedicalRec
                         - - (%SpecialClinicGeneralInfo;,
                              ClinicalPhysiologyNote+)>
<!ELEMENT ClinicalNeuroPhysiologyContinuingMedicalRec
                         - - (%SpecialClinicGeneralInfo;,
                              ClinicalNeuroPhysiologyNote+)>
<!ELEMENT PathologyContinuingMedicalRec
                         - - (%SpecialClinicGeneralInfo;,
                              PathologyNote+)>
<!ELEMENT RadioTherapyOngologyContinuingMedicalRec
                         - - (%SpecialClinicGeneralInfo;,
                              RadioTherapyOngologyNote+)>
<!ELEMENT ClinicalRadiologyContinuingMedicalRec
                         - - (%SpecialClinicGeneralInfo;,
                              ClinicalRadiologyNote+)>
<!ELEMENT SocialWorkContinuingMedicalRec
                         - - (%SpecialClinicGeneralInfo;,
                              SocialWorkNote+)>
<!ELEMENT GastroenterologyContinuingMedicalRec
                         - - (%SpecialClinicGeneralInfo;,
                              GastroenterologyNote+)>
<!ELEMENT PhysiatricsContinuingMedicalRec
                         - - (%SpecialClinicGeneralInfo;,
                              PhysiatricsNote+)>
<!ELEMENT NutritionContinuingMedicalRec
                         - - (%SpecialClinicGeneralInfo;,
                              NutritionNote+)>
<!ELEMENT OccupationalTherapyContinuingMedicalRec
                         - - (%SpecialClinicGeneralInfo;,
                              OccupationalTherapyNote+)>
<!ENTITY % ConfirmedNote1 "PoliclinicDepartment,
                           Date">
<!ENTITY % ConfirmedNote2 "CreatorName,
                           CreatorSign">
<!ELEMENT (PoliclinicDepartment,
           CreatorName,
           CreatorSign)  - - (#PCDATA)>
```

```
<!-- structure Definitions of the general Types of clinical Notes -->

<!ENTITY % GeneralNoteType    "%ConfirmedNote1;,
                              Reason,Anamnes,Status,Problem,
                              Plan,Decursus,Epicrisis,
                              %ConfirmedNote2">
<!ELEMENT (Reason,Anamnes,Status,
          Problem,Plan,Epicrisis,
          Decursus,DiseaseProgress,
          FinalStatement,Sytostats,
          GrowthCharts)  - - (#PCDATA)>



<!ELEMENT (InternalMedicineNote,
           SurgeryNote,
           NeuroSurgeryNote,
           PsychiatryNote,
           ChildrenPsychiatryNote,
           GynaegologyNote,
           GeneticMedicineNote,
           OphthalmologyNote,
           OrongologyNote,
           SpeechTherapyNote,
           DermatologyNote,
           PulmonaryDiseasesNote,
           NeurologyNote,
           FoniatryNote,
           PediatricsPsychiatryNote,
           WorkMedicineNote)
                           - - (%GeneralNoteType)>
<!ELEMENT PediatricsNote   - - (%GeneralNoteType,Sytostats,
                                GrowthCharts)>
<!ELEMENT OphthalmologySquintNote
                           - - (%ConfirmedNote1;,
                                %ConfirmedNote2;)>
<!ELEMENT OphthalmologyColorNote
                           - - (%ConfirmedNote1;,
                                %ConfirmedNote2;)>
<!ELEMENT ClinicalPhysiologyNote
                           - - (%ConfirmedNote1;,
                                RadioIsotopeDose,
                                Referral,Report,
                                %ConfirmedNote2;)>
<!ELEMENT ClinicalNeuroPhysiologyNote
                           - - (%ConfirmedNote1;,
                                Study,
                                %ConfirmedNote2;)>
<!ELEMENT PathologyNote    - - (%ConfirmedNote1;,
                                Request,Report,Finding,
                                %ConfirmedNote2;)>
<!ELEMENT RadioTherapyOngologyNote
```

```
                          - - (%ConfirmedNote1,
                               %ConfirmedNote2)>
<!ELEMENT ClinicalRadiologyNote
                          - - (%ConfirmedNote1;,
                               Referral,Visit,Report,
                               %ConfirmedNote2;)>
<!ELEMENT SocialWorkNote   - - (%ConfirmedNote1,
                               %ConfirmedNote2;)>
<!ELEMENT MedicalRehabilitationNote
                          - - (%ConfirmedNote1,
                               %ConfirmedNote2;)>
<!ELEMENT GastroenterologyNote
                          - - (%ConfirmedNote1;,
                               Request,Answer,Finding,Plan,
                               %ConfirmedNote2;)>
<!ELEMENT PhysiatricsNote  - - (%ConfirmedNote1;,
                               ProblemDefinition,Plan,
                               Implementation,
                               %ConfirmedNote2;)>
<!ELEMENT NutritionNote    - - (%ConfirmedNote1;,
                               Diagnose,NutritionAnamnes,
                               DietPlan,
                               %ConfirmedNote2;)>
<!ELEMENT OccupationalTherapyNote
                          - - (%ConfirmedNote1;,
                               Problem, Plan, Treatment,
                               %ConfirmedNote2;)>
<!ELEMENT (Sytostats,
           GrowthCharts    - - (EMPTY)>
<!ELEMENT (RadioIsotopeDose,
           Referral,
           Request,Report,
           Visit,Answer,Finding,
           ProblemDefinition,
           Implementation,Treatment,
           Diagnose,NutritionAnamnes,
           DietPlan)       - - (#PCDATA)>


<!-- structure Definitions of the summary form -->
<!ELEMENT SummaryForm      - - (SummaryMarking+)>
<!ELEMENT SummaryMarking   - - (TimeInterval,ClinicSpeciality,
                               DiagnosisCode,
                               DiagnosisSymptoms,SpecialNotes)>
<!ELEMENT (TimeInterval,
       ClinicSpeciality,
       DiagnosisCode,
           DiagnosisSymptoms,
           SpecialNotes)   - - (#PCDATA)>


<!-- structure Definitions of the blood group and
     blood transfusion Info -->
```

```
<!ELEMENT BloodGroupTransfusion
                          - - (LaboratoryInfo,
                               SubscriberInfo,
                               NewbornsInfo?)>
<!ELEMENT LaboratoryInfo   - - (BloodGroup,Antibodies,
                               Remarks, Date,
                               DeterminatorsSign,
                               SampleCollectorsSign)>
<!ELEMENT (BloodGroup,Antibodies,
           DeterminatorsSign,Date,
           SampleCollectorsSign)
                          - - (#PCDATA)>
<!ELEMENT SubscriberInfo   - - (EarlierBloodTransfusions,
                               TransfusionsComplications,
                               LeucosyteAntibodies,
                               PregnanciesMiscarriages)>
<!ELEMENT (EarlierBloodTransfusions,
           TransfusionsComplications,
           LeucosyteAntibodies,
           PregnanciesMisscarriages)
                          - - (#PCDATA)>
<!ELEMENT NewbornsInfo     - - (MothersBloodGroup,
                               MothersAntibodies,
                               DirectCoombs)>
<!ELEMENT (MothersBloodGroup,
           MothersAntibodies,
           DirectCoombs)    - - (#PCDATA)>


<!-- structure Definition of the Medical Certificate -->
<!ELEMENT MedicalCertificate
                          - - (EMPTY)>


<!-- structure Definitions of the Nursing record -->
<!ELEMENT NursingRec       - - (EntranceInterview+,Diet+,
                               FirstAidNursingPlan*,
                               ContinuingNursingPlan,
                               NursingTable)>
<!ELEMENT (EntranceInterview,
           Diet, FirstAidNursingPlan,
           NursingTable)   - - (#PCDATA)>
<!ELEMENT ContinuingNursingPlan
                          - - (NursingPlanNote+)>
<!ELEMENT NursingPlanNote  - - (NursingNeedProblems,
                               NursingPlan,
                               ProgressControlRating,
                               Time,
                               NursingNeedDeterminatorsSign)>
<!ELEMENT NursingNeedProblems
                          - - (ComesFrom,
                               SymptomsDescription)>
<!ELEMENT NursingPlan      - - (ResearchRequest,
```

```
                                Medicaments,
                                ExtensionNursing)>
<!ELEMENT ProgressControlRating
                            - - (ControlTimePlace,
                                NursingActions,
                                Measurements*)>
<!ELEMENT (ExtensionNursing,
           ControlTimePlace)         - - (#PCDATA)>

<!ELEMENT Measurements      - - (Puls,Temparature,Weigth,Height,
                                HeadCircumference)>
<!ELEMENT (Time,
           NursingNeedDeterminatorsSign,
           ComesFrom,
           SymptomsDescription,
           ResearchRequest,
           Medicaments,
           NursingActions,
           Puls,Temparature,Weigth,Height,
           HeadCircumference) - - (#PCDATA)>
]>
```