

AUTOMATISIERTE GENERIERUNG VON XML-SCHEMATA AUS EN/ISO 13606 ARCHETYPEN

Janzek-Hawlat S¹, Kuttin O¹, Sabinovic S¹, Rinner C¹,
Duftschmid G¹

Kurzfassung

Ein semantisch interoperabler Austausch von EN/ISO 13606 EHR-Extracts setzt voraus, dass diese vom Empfangssystem sinnvoll interpretierbar sind. Dies kann durch eine Validierung der EHR-Extracts gegen das Referenzmodell und die zugrundeliegenden Archetypen geprüft werden. Wir präsentieren eine Methode zur automatischen Ableitung von XML-Schemata aus Archetypen, auf deren Basis EHR-Extracts validiert werden können.

1. Einleitung

Eine qualitativ hochwertige, integrierte Versorgung von Patienten erfordert einen semantisch interoperablen Gesundheitsdatenaustausch zwischen den Electronic Health Record (EHR)-Systemen der involvierten Gesundheitsdienstleistungsanbietern. Dabei wird unter semantischer Interoperabilität vor allem die korrekte inhaltliche Interpretation der Informationen, welche zwischen den verschiedenen Computersystemen ausgetauscht werden, auf Anwendungssystemebene verstanden.

Eine effiziente Vorgehensweise für semantische Interoperabilität beim Austausch von EHR-Daten bietet der Zweimodell-Ansatz: Dieser basiert auf dem *Referenzmodell* (RM), welches die Bausteine eines EHR beschreibt und dem *Archetype Object Model* (AOM), das zur Instanziierung von *Archetypen* dient. Letztere beschreiben, wie konkrete Inhalte eines EHRs aus Bausteinen des RM zusammengesetzt sind [1, 2]. Das AOM [3] ist das objektorientierte Modell-Äquivalent zur *Archetype Definition Language* (ADL) [4], mit deren Hilfe Archetypen syntaktisch beschrieben werden. Archetypen, wiederum, sind Instanzen des AOM (siehe *Abbildung 1* obere Hälfte). Im Gegensatz zu den im Bereich der *Health Level 7* (HL7) *Clinical Document Architecture* (CDA) zur semantischen Spezifikation individueller CDA-Ausprägungen eingesetzten Implementierungsleitfäden, stellen Archetypen eine rein formale, computerlesbare semantische Beschreibung von EHR-Inhalten dar. Seitens HL7 wird zwar an einer Spezifikation der mit Archetypen vergleichbaren Templates gearbeitet, die Archetypen-Spezifikation liegt jedoch bereits als fertiger Standard vor.

Als Grundlage dieser Arbeit sind die Ergebnisse von [7] anzusehen, wo beschrieben wird, wie durch den Einsatz von *Extensible Markup Language* (XML)-Technologien auf dem RM der *EN/ISO 13606* [5] basierende, archetypkonforme *EHR-Extracts* aus einem bestehenden EHR-System exportiert werden können (siehe *Abbildung 1* untere Hälfte). Die Abbildung vom Datenmo-

¹ Institut für medizinische Informations- und Auswertesysteme, Medizinische Universität Wien

dell des EHR-Systems auf das EN/ISO 13606 Datenmodell erfolgt dabei über ein visuell definierbares Mapping zwischen 2 XML-Schemata, wobei das „Ziel-Schema“ des Mappings das archetypkonforme EHR-Extract beschreibt. Das Ziel-Schema kann dabei neben dem angesprochenen Mapping auch für eine Validierung von EHR-Extracts herangezogen werden, die von anderen EHR-Systemen empfangen wurden. Damit kann sichergestellt werden, dass das EHR-Extract für den Empfänger korrekt interpretierbar ist und in das Datenmodell des eigenen EHR-Systems überführt werden kann. Die Verwendung des Ziel-Schemas zur Validierung ist insofern nützlich, als derzeit noch kein entsprechendes Tool für die Validierung eines EHR-Extracts gegenüber einem Archetyp existiert. Ein vor allem in der HL7-Welt häufig praktizierter Ansatz zur Validierung von Gesundheitsdaten liegt in der Verwendung von Schematron Regeln. Diese bieten zwar für den Zweck der Validierung umfassendere Möglichkeiten, ermöglichen allerdings nicht die in [7] angestrebte Abbildung vom Datenmodell des EHR-Systems auf das EN/ISO 13606 Datenmodell.

Das Ziel-Schema wurde bisher manuell aus dem Archetyp abgeleitet, was für einen „Proof of concept“ vertretbar ist, für einen breiteren Einsatz der Methode aber insofern keine praktikable Lösung darstellt, als (1) pro Archetyp ein eigenes XML-Schema erstellt werden muss, was bei einer großen Anzahl von Archetypen aufwendig wird, und (2) die Erstellung eines validierfähigen XML-Schemas eine Klonung von Schema-Elementen erfordert. Dies hat den bekannten Nachteil, dass Adaptierungen in der Schema-Repräsentation einer *RM*-Klasse an mehreren Stellen nachgezogen werden müssen. Eine Verwendung des Vererbungsmechanismus zur Lösung des Problems ist nicht möglich, da dies innerhalb des Schemas komplexe Typen erfordern würde, die gleichzeitig eine Erweiterung `<xsd:extension>` und Einschränkung `<xsd:restriction>` ihres übergeordneten komplexen Typs sein müssten. Beispielsweise müsste ein komplexer Typ für einen systolischen Blutdruck, der laut Archetyp eine Instanz der *Reference Model* (RM)-Klasse *ELEMENT* ist, eine Erweiterung des komplexen Typs für die RM-Klasse *RECORD_COMPONENT* sein, da letztere laut RM von der Klasse *ELEMENT* um einige Attribute erweitert wird. Gleichzeitig müsste er jedoch auch eine Einschränkung des komplexen Typs für *RECORD_COMPONENT* sein, da dessen Attribut *name* auf den Text „systolic pressure“ einzuschränken wäre.

Ziel der hier beschriebenen Arbeit ist es daher, die Ableitung eines XML-Schemas aus einem Archetyp zu automatisieren. Wie bereits erwähnt ergänzt die Arbeit Vorarbeiten aus [7] und fokussiert analog zu [7] auf die im EHR-Standard EN/ISO 13606 spezifizierten Konzepte. Ergänzende Konzepte anderer Organisationen, wie z.B. openEHR-Templates, werden im Rahmen dieser Arbeit nicht berücksichtigt.

Die automatische Ableitung eines XML-Schemas aus einem Archetyp hat im Vergleich zu einer manuellen Erstellung neben der offensichtlichen Aufwandsersparnis den Vorteil, dass die oben angesprochene, klonungsbedingte Redundanz im XML-Schema unproblematisch wird. Adaptierungen in der Schema-Repräsentation einer RM-Klasse würden bei einer automatischen Generierung des Schemas ja im Algorithmus programmatisch realisiert und müssten nicht manuell an mehreren Stellen des Schemas nachgezogen werden.

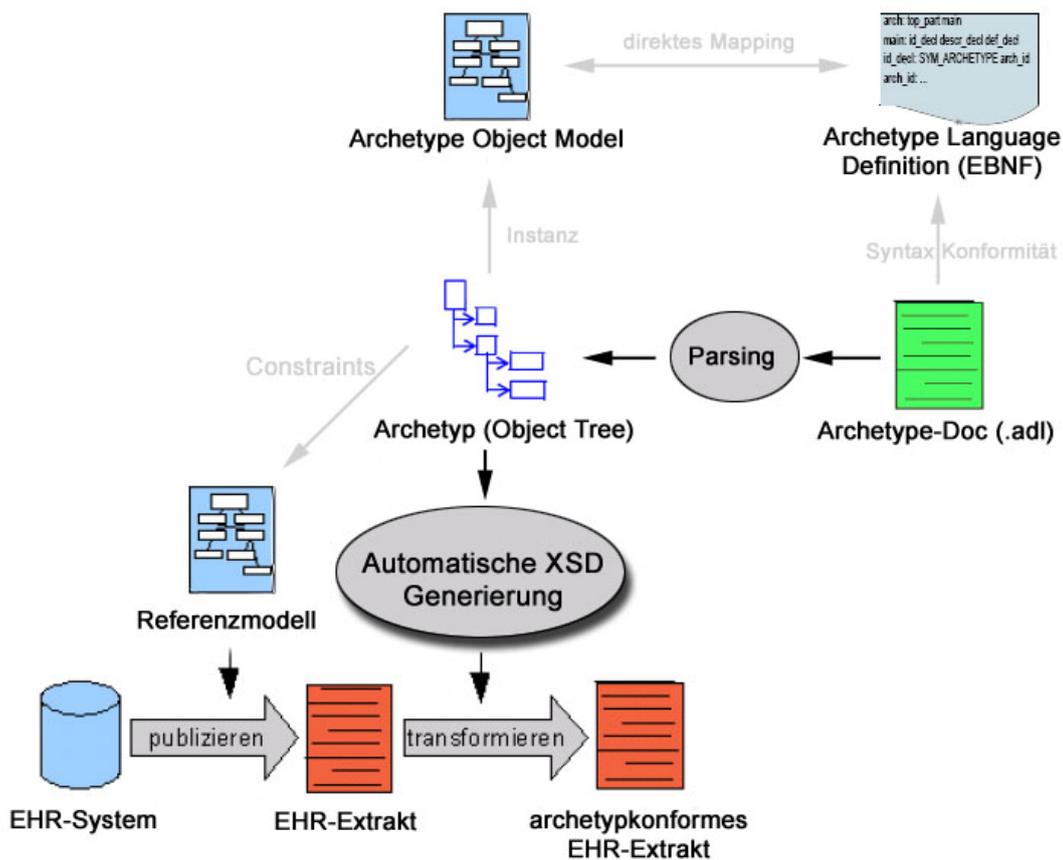


Abbildung 1: Grobe Übersicht an welcher Stelle die Resultate dieser Arbeit greifen werden, samt den Komponenten und deren Beziehungen (graue Pfeile) zueinander.

2. Methode

Zur Beschreibung und Validierung von XML-Dateien hat sich *W3C XML-Schema Definition 1.0 (XSD)* als formal ausdrucksstärkste Definition durchgesetzt. XSD kommt auch in der EHR-Modellierung zum Einsatz, unter anderem wurden für EN/ISO 13606 XSD-Implementierungen des RM (siehe www.linkehr.com) sowie des AOM (siehe www.openehr.org) öffentlich zur Verfügung gestellt.

Diese Arbeit widmet sich dem Ansatz einer automatisierten Transformation von ADL-Dateien zu XSD-Dateien (siehe *Abbildung 2*). Aus einer erfolgreichen Transformation resultiert ein XML-Schema, das eine Hierarchie gültiger RM-Instanzen sowie deren Constraints gemäß eines (oder mehrerer) Archetypen abbildet und in weiterer Folge eine Validierung von Archetyp-konformen EHR-Extracts, die als XML-Datei repräsentiert sind, ermöglicht.

```

CLUSTER[at0021] occurrences matches {0..1} matches { -- dimensions CLUSTER <xs:element name="CLUSTER_Dimensions">
ELEMENT[at1002] occurrences matches {0..1} matches { -- Length                                     <xs:complexType>
  value matches {                                                                              <xs:complexContent>
    PQ matches {                                                                               <xs:complexType>
      property matches {"openehr::122"}                                                       <xs:extension base="CLUSTER">
      unit matches {"mm", "cm", "m"}                                                           <xs:sequence>
    }                                                                                           <xs:element name="parts" minOccurs="0">
  }                                                                                             <xs:complexType>
}                                                                                               <xs:sequence>
ELEMENT[at1003] occurrences matches {0..1} matches { -- Breadth                             <xs:element ref="ELEMENT_Length" minOccurs="0"/>
  value matches {                                                                              <xs:element ref="ELEMENT_Breadth" minOccurs="0"/>
    PQ matches {                                                                               </xs:sequence>
      property matches {"openehr::122"}                                                       </xs:complexType>
      unit matches {"mm", "cm", "m"}                                                           </xs:element>
    }                                                                                           </xs:sequence>
  }                                                                                             </xs:extension>
}                                                                                               </xs:complexContent>
}                                                                                               </xs:complexType>
}                                                                                               </xs:element>

```

Abbildung 2: Transformation eines Archetyps von ADL (links) auf XSD (rechts). Man erkennt, dass der „Cluster Dimension“ mit den beiden Elementen „Length“ und „Breadth“ auf beiden Seiten vorhanden ist.

Ausgangspunkt des Lösungsansatzes ist ein in der ADL spezifizierter Archetyp. Dabei ist zu beachten, dass ein Archetyp lediglich dazu dient, bestehende Constraints auf Klassen, Attribute und Beziehungen, die anhand des RM definiert sind, zu verschärfen. Der Prozess der Transformation gliedert sich in die folgenden drei Teilprozesse:

1. Übersetzung der ADL-Spezifikation in einen Object Tree: Das Konvertieren von ADL-Dateien in einen Java-Object-Tree wird mittels des ADL-Parsers der openEHR-Foundation (siehe http://svn.openehr.org/ref_impl_java/TRUNK/docs/download.htm) ermöglicht.
2. Semantische Interpretation des Object-Trees: Der Object-Tree stellt sich als Instanz des AOM dar. In Abhängigkeit der Klassen des AOM müssen semantisch äquivalente XSD-Fragmente identifiziert werden. Die XSD-Fragmente liegen in einer parametrierbaren, textbasierten Form vor welche die variablen Teile des RMs darstellen. Dabei ist zu beachten, dass aufgrund der „unique particle attribution constraint rule“ zwei XSD-Elemente auf derselben Hierarchieebene hinsichtlich ihres Tagnamens nicht identisch sein dürfen [8]. Eine in [7] vorgeschlagene Möglichkeit dieses Problem zu umgehen, ist die Verwendung von Tagnamen, die sich aus dem Namen der betreffenden RM-Klasse und der inhaltlichen Bedeutung des XSD-Elements gemäß des Attributs node_id der AOM-Klasse C_OBJECT zusammensetzen (siehe z.B. Tags <ELEMENT_Length> und <ELEMENT_Breadth> in *Abbildung 2* rechts).
3. Generierung eines XML-Schemas basierend auf dem Object-Tree: Ausgangspunkt ist ein Grundgerüst welches die obligatorischen (minoccur > 0) Instanzen, Attribute und Beziehungen des RM enthält. Diese müssen im EHR-Extract enthalten sein, auch wenn sie im Archetyp nicht explizit angeführt, d.h. eingeschränkt sind. Das Grundgerüst wird sequenziell um XSD-Fragmente erweitert indem der Object-Tree traversiert wird. Die XSD-Fragmente werden entsprechend der in den AOM-Instanzen enthaltenen Constraints (übernommene Constraints sind Datentypen, Multiplizitäten, Optionalitäten und Wertebereiche) parametriert.

3. Aktueller Status und Ausblick

Bezugnehmend auf die in Kapitel 2 beschriebenen Schritte stellt sich der aktuelle Status unserer Arbeit wie folgt dar: Im Rahmen von Schritt 1 wurden die EN/ISO 13606 Archetypen Blood pres-

sure und Dimensions [6] mittels des openEHR ADL-Parser erfolgreich als Object-Tree dargestellt und analysiert. Dieser Schritt kann als abgeschlossen betrachtet werden. Die Schritte 2 und 3 zu denen bereits das theoretische Konzept besteht, befinden sich am Beginn der Umsetzungsphase. In einem iterativen Prozess werden sämtliche XSD-Fragmente identifiziert und implementiert. Im Anschluss wird nach einem adäquaten Algorithmus gesucht der mittels Traversierung des Object-Trees die XSD-Fragmente parametrisiert.

4. Referenzen

- [1] BEALE T. Archetypes and the EHR, *Stud Health Technol Inform.* 96:238-44. 2003.
- [2] BEALE T. Archetypes: Constraint-based Domain Models for Future-proof Information Systems, <http://citeseerx.ist.psu.edu/viewdoc/download?doi=10.1.1.21.1158&rep=rep1&type=pdf>. 2001.
- [3] BEALE T. Archetype Object Model, <http://www.openehr.org/releases/1.0.2/architecture/am/aom.pdf>. 2008
- [4] BEALE T., HEARD S. Archetype Definition Language, <http://www.openehr.org/releases/1.0.2/architecture/am/adl.pdf>. 2008
- [5] EUROPEAN COMMITTEE FOR STANDARDIZATION, INTERNATIONAL ORGANIZATION FOR STANDARDIZATION. EN/ISO 13606 Electronic health record communication. 2006.
- [6] PROCHAZKOVA D. Architekturen der elektronischen, lebensbegleitenden Gesundheitsakte: Ein Vergleich der Archetypansätze von CEN und HL7 am Beispiel des Konzeptes 'Diabetes'. Master Thesis, Medical University of Vienna, 2006.
- [7] RINNER C, DUFTSCHMID G, WRBA T. CEN prEN 13606 konformer Export von medizinischen Daten aus einem Entity-Attribute-Value basierten Informationssystem, *Telemedizinführer Deutschland 2008 (Sonderkapitel "Best of Telemed Berlin")*: 34-8. 2008.
- [8] TUN Z, BIRD LJ, GOODCHILD A. Validating Electronic Health Records Using Archetypes and XML: CRC for Enterprise Distributed Systems: University of Queensland; 2002.