

TECNOLOGIAS XML PARA A INTEROPERABILIDADE SEMÂNTICA EM SAÚDE: REFERENCIAL TEÓRICO E IMPLEMENTAÇÃO

XML Technologies for Semantic Interoperability in Healthcare: Technical Background and Implementation

Luciana Tricai Cavalini¹; Arôvel Oliveira Moura Júnior²; Pedro Monteiro Menezes³; Timothy Wayne Cook⁴

Resumo **Objetivos:** Descrever a implementação do Modelo de Referência das especificações *Multilevel Healthcare Information Modeling* (MLHIM) em tecnologias XML, bem como um conjunto de exemplos de conceitos de saúde gerados a partir do repositório do *National Cancer Institute – Common Data Elements*. **Materiais e Métodos:** As especificações MLHIM adotaram *XML Schema Definition 1.1* como base para a sua implementação de referência, uma vez que as tecnologias XML são consistentes em todas as plataformas e sistemas operacionais, apresentando ferramentas disponíveis para todas as linguagens de programação convencionais. **Resultados:** Nas especificações MLHIM, a representação do conhecimento de saúde é definida pelo modelo de domínio, expressa em *Concept Constraint Definitions* (CCDs), que fornecem a interpretação semântica dos objetos persistidos de acordo com os tipos genéricos do modelo de referência. O conjunto de CCDs foi a base para a simulação de instâncias de dados semanticamente coerentes, de acordo com validadores XML independentes, persistidos em um banco de dados XML. **Conclusão:** Este trabalho mostra a viabilidade da adoção de tecnologias XML para a realização da interoperabilidade semântica em cenários reais de saúde, provendo os desenvolvedores de aplicativos com uma quantidade significativa de experiência acumulada e um vasto leque de ferramentas disponíveis. **Palavras-chave:** Troca de Informações, Registros Eletrônicos de Saúde, Design de Software

Abstract **Aims:** To report the implementation of the MLHIM Reference Model in XML Schema Definition language version 1.1 as well as a set of examples of healthcare concepts generated from the National Cancer Institute – Common Data Elements (NCI CDE) repository. **Materials and Methods:** The Multilevel Healthcare Information Modeling (MLHIM) specifications have adopted the XML Schema Definition 1.1 as the basis for its reference implementation, since XML technologies are consistent across all platforms and operating systems, with tools available for all mainstream programming languages. **Results:** In the MLHIM specifications, the healthcare knowledge representation is defined by the Domain Model, expressed as Concept Constraint Definitions (CCDs), which provide the semantic interpretation of the objects persisted according to the generic Reference Model classes. The set of CCDs was the base for the simulation of semantically coherent data instances, according to independent XML validators, persisted on an XML database. **Conclusion:** This paper shows the feasibility of adopting XML technologies for the achievement of semantic interoperability in real healthcare scenarios, by providing application developers with a significant amount of industry experience and a wide array of tools through XML technologies. **Keywords:** Information Exchange, Electronic Health Records, Software Design

1. Professora Adjunta do Departamento de Tecnologias da Informação e Educação em Saúde - Faculdade de Ciências Médicas - Universidade do Estado do Rio de Janeiro - Doutorado em Saúde Pública; 2. Médico do Instituto Nacional do Câncer - Mestre em Saúde Materno-Infantil; 3. Analista de Sistemas do Hospital Sarah-Rio - Tecnólogo em Sistemas de Informação; 4. Chief Technological Officer da MedWeb 3.0 - MSc in Health Informatics

Introdução

A implementação de Registros Eletrônicos de Saúde (RES) tem sido proposta para aumentar a eficácia do sistema de saúde, mas estas expectativas ainda não foram cumpridas. Desde 1961, quando o primeiro RES foi instalado no *Akron General Hospital*¹, e durante os mais de 50 anos desde então, várias empresas de software têm tentado obter a capacidade de tornar sistemas distribuídos semanticamente interoperáveis, a fim de fornecer uma plataforma de informação em saúde coerente².

Os desafios relacionados com ao registro computadorizado de informação clínica estão principalmente associadas ao fato de que a saúde é um sistema complexo e dinâmico. Em relação à complexidade, sabe-se, por exemplo, que a *Systematized Nomenclature of Medicine – Clinical Terms (SNOMED-CT)*, a terminologia mais abrangente na área da saúde, tem mais de 311 mil termos, conectados em mais de 1.360.000 formas³. Assim, o dinamismo observado nas informações de saúde está essencialmente relacionado à velocidade da incorporação tecnológica, que é a principal característica do setor⁴.

Além disso, o sistema de saúde é, por definição, hierárquico e descentralizado; assim, espera-se que os pacientes obtenham acesso ao sistema através da atenção primária, sendo, em seguida, referenciados para níveis de maior complexidade de atendimento. Por razões históricas e econômicas, os serviços de atenção básica estão localizados próximos dos domicílios do usuário, enquanto as instituições de saúde mais complexas (como hospitais) são geralmente construídas em áreas centrais. As funções da atenção básica e de média e alta complexidade são claramente diferentes, o que determina uma alta variabilidade em sua estrutura arquitetônica e, em consequência, cada instituição de saúde adotará fluxos de trabalho específicos que se

adaptam à sua forma e função. Este processo irá refletir sobre a especificidade das informações coletadas, armazenadas e processadas dentro de um determinado serviço de saúde⁵.

No entanto, nenhuma instituição de saúde é isolada das demais. Por causa da configuração do sistema de saúde, os pacientes circulam em mais de um serviço. Isto é particularmente verdadeiro para pacientes com condições crônicas que são atendidos em média por mais de 80 profissionais de saúde diferentes no curso de sua doença. Assim, idealmente, o registro de cada paciente deveria ser mantido de forma longitudinal, uma vez que qualquer parte da informação pode ser importante em qualquer momento da vida do paciente⁶.

A realização de tais níveis de interoperabilidade entre RES ainda permanece como um desafio. Atualmente, há uma multiplicidade de empresas e instituições governamentais cuja tarefa é desenvolver sistemas de informação em saúde, cada um deles executando o seu próprio modelo de dados, que é específico para cada sistema. Tais modelos de dados não são apenas diferentes de sistema para sistema, mas eles também estão sempre mudando em compasso com a mudança dos requisitos dos sistemas, o que inclui as contínuas evoluções científicas em medicina, as políticas governamentais e as normas das operadoras de planos e seguros de saúde⁷.

Esta mudança constante é um componente caro da gestão da informação em saúde e cria uma situação complexa no contexto semântico dos dados de saúde, que é incorporado tanto na estrutura da base de dados quanto no código-fonte do aplicativo. Assim, quando se tenta realizar o compartilhamento de dados entre sistemas de informação de saúde diferentes, mesmo na situação mais simples (quando os tipos de dados são os mesmos), o contexto completo em que os dados foram registrados permanece desconhecido para o sistema receptor⁸.

Muitas soluções têm sido propostas para o problema da interoperabilidade entre sistemas de informação de saúde, o que inclui um conjunto vasto e variável de modelos de representação do conhecimento, especialmente terminologias e ontologias⁹. No entanto, os altos custos de implementação e manutenção dos RES disponíveis têm impedido sua adoção generalizada; até mesmo alguns retrocessos têm sido observados nos últimos anos¹⁰. Atualmente, o único método de desenvolvimento de *software* que permite a interoperabilidade semântica entre sistemas de informação em saúde é a modelagem multinível (ou dual) originalmente proposta pela Fundação *openEHR*¹¹ e desenvolvida por dois projetos baseados nos mesmos princípios: a família de Normas ISO 13606¹² e as especificações *Multilevel Healthcare Information Modeling (MLHIM)*¹³.

Embora a capacidade de obter interoperabilidade semântica entre RES com o uso da modelagem multinível já tenha sido comprovada em *software*, há relativamente poucas implementações comprovadas das especificações *openEHR* ou da Norma ISO 13606. Isto pode ser atribuído à complexidade das especificações *openEHR* ou pelo fato do padrão ISO 13606 não permitir a persistência dos dados, mas apenas a troca de mensagens entre os sistemas. Outra barreira importante para a adoção mais ampla da modelagem multinível de acordo com os modelos *openEHR* e ISO 13606, é o uso de uma linguagem específica de domínio, a *Archetype Definition Language (ADL)*, para a definição dos modelos de dados. Em ambos os casos, a ADL foi adotada para a definição de restrições às classes do modelo de informação (conhecido como Modelo de Referência), para cada conceito de saúde. Alguns autores têm manifestado as suas preocupações sobre as barreiras técnicas do uso de ADL para o desenvolvimento de aplicações que possam ser instaladas em serviços de saúde reais, em que os conceitos terão um alto nível de complexidade¹⁴.

Objetivo

Dado o fato de que a interoperabilidade semântica é uma questão tão fundamental para o sucesso na adoção de tecnologias de informação na área da saúde, e a modelagem multinível é uma solução para isso, há uma necessidade de tornar tais princípios implementáveis em sistemas de informação em saúde reais. Isto foi obtido nas especificações MLHIM adotando tecnologias XML para a sua implementação, que são um padrão da indústria para o desenvolvimento de *software* e para a troca de informações. Este trabalho apresenta o desenvolvimento de um aplicativo de demonstração baseado na versão 2.4.3 das especificações MLHIM.

Metodologia

A abordagem metodológica adotada neste estudo incluiu: (a) a implementação dos componentes básicos das especificações MLHIM (o modelo de referência e os modelos de domínio) em XML *Schema* 1.1 e (b) a geração de dados simulados com base em um conjunto de modelos de domínio selecionados para conceitos demográficos e clínicos e (c) a demonstração da persistência e os procedimentos consulta ao banco de dados implementadas em duas aplicações de demonstração, usando os dados simulados produzidos.

Visão Geral das Especificações MLHIM

As especificações MLHIM¹³ estão publicadas (<https://github.com/mlhim>) como um conjunto de ferramentas de código aberto para o desenvolvimento de RES e outros tipos de aplicativos médicos, de acordo com os princípios da modelagem multinível. As especificações estão estruturadas em dois modelos: o Modelo de Referência e o Modelo de Domínio.

O Modelo de Referência MLHIM conceitual é composto por um conjunto de classes (e seus respectivos atributos) que permitem o desenvolvimento de qualquer tipo de aplicativo para a área da saúde, desde RES hospitalares até pequenos aplicativos que coletem dados em dispositivos móveis. Isto foi obtido através da minimização do número e do resíduo semântico das classes do Modelo de Referência, quando comparados com as especificações *openEHR* originais. As classes e a semântica remanescentes foram consideradas como necessárias e suficientes para permitir que qualquer modelo de persistência de dados estruturados em saúde. Portanto, a abordagem do Modelo de Referência MLHIM é minimalista¹⁵, mas não tão abstrata como uma linguagem de programação.

O Modelo de Referência MLHIM é expresso em um documento em XML *Schema Definition* 1.1. Cada uma das classes do Modelo de Referência são expressas como uma definição de um *complexType*, dispostos como 'xs: extension'. Para cada *complexType* há também uma definição de 'element'. Esses 'element' são organizados em grupos de substituição, a fim de se adequar ao conceito de herança de classe definido no Modelo de Referência conceitual.

O Modelo de Domínio MLHIM é expresso em *Concept Constraint Definitions* (CCDs), implementadas em XML *Schema Definition* 1.1, sendo conceptualmente equivalentes aos arquétipos *openEHR* e ISO 13606. Cada CCD define as combinações e restrições de classes e atributos de classe (que são genéricos e estáveis) do Modelo de Referência MLHIM, que são consideradas necessárias e suficientes para representar adequadamente um determinado conceito em saúde. Em geral, as CCDs são definidas para permitir ampla reutilização, mas não há qualquer limitação quanto ao número de CCDs permitidos para um mesmo conceito no ecossistema MLHIM. Cada CCD é identificada por um *Universal Unique Identifier* Tipo 4 (UUID). Isso proporciona a permanência da

definição do conceito para sempre, criando assim uma base estável para as instâncias de dados definidas de acordo com os contextos temporais, espaciais e ontológicos existentes no momento de captura do dado. Este é um requisito muito importante, para que se possa preservar a semântica original do momento da captura dos dados, de modo que quaisquer análises futuras dos dados não sejam desviadas de forma imprevisível. Este é um problema comum quando os dados são migrados de um formato de banco de dados para outro e de código-fonte da aplicação é modificado¹⁶. Uma vez que estes são os lugares onde a semântica está contida em sistemas de informações de saúde convencionais, após a migração, os dados já não mais representam sua semântica original.

A principal inovação nas especificações MLHIM é o uso de definições dos *complexTypees* na CCD com base nas restrições nos tipos do Modelo de Referência. Dado o fato de que a maioria dos conceitos médicos são multivariados, para a maioria das CCDs, $n (n > 0)$ *complexTypees* serão incluídos. Assim, como é provável que haja uma CCD com mais de um *complexType*, a cada um deles também será associado um UUID tipo 4, que é semelhante ao processo de identificação da CCD completa acima descrito. Isto permite a existência de múltiplos *complexTypees* da mesma natureza (por exemplo, uma CCD pode ter mais do que um *ClusterType* ou mais do que um *DvStringType*) na mesma CCD sem conflito das restrições. Essa abordagem também facilita a consulta à base de dados, uma vez que cria uma instrução de caminho única e universal para qualquer dado específico produzido de acordo com as especificações MLHIM.

As CCDs têm capacidade para acomodar qualquer número de ontologias e terminologias médicas. Todos os *complexTypees* podem incluir links como informação computável pelo aplicativo ('xs: appinfo'), que pode ser usado para incluir toda a semântica necessária, ligada ou não a quaisquer ontologias ou terminologias. Estes

links são criados como parte da CCD em um elemento 'annotation' e permitem a inserção de conteúdo no formato *Resource Description Framework* (RDF) para maior aprimoramento da semântica relacionada ao conceito, com base em qualquer ontologia relevante ou outras fontes de definição para o conceito.

A segunda inovação chave diz respeito à abordagem adotada em MLHIM para o tratamento de dados faltantes ou dados que estão fora dos valores ou dos tipos esperados (denominados "valores excepcionais"), o que não é uma ocorrência incomum em sistemas de informação em saúde. Todos os tipos de dados em MLHIM (descendentes de *DvAny*) apresentam um elemento 'ev' para o registro da semântica de valores excepcionais¹⁷. Esta abordagem é semelhante ao que a Norma ISO 21090 denomina "Null flavors". No entanto, o modelo conceitual da Norma ISO 21090 é frágil e não permite expansão, criando a possibilidade de registro semântico faltante, incompleto ou incorreto para os dados faltantes. MLHIM resolve este problema, fornecendo uma árvore com base nos elementos 'ev-meaning' e 'ev-name' do *complexType ExceptionalValue*, sendo os valores para estes elementos fixos para cada *complexType*.

Por exemplo, no *INVType*, o 'ev-name' é "inválido" e o 'ev-meaning' é "O valor, conforme representado na instância de dados, não é um membro do conjunto de valores permitidos na amplitude de valores definidos para esta variável", conforme texto adaptado da Norma ISO 21090. Um exemplo de uma extensão para a Norma ISO 21090 é a *ASKRType*, representando a ocorrência comum (mas ainda subnotificada) referente a "Recusou-se a responder". Assim, além das extensões dos valores excepcionais no Modelo de Referência, qualquer CCD pode estender o *complexType ExceptionalValue* para criar contexto semântico para dados faltantes ou excepcionais específicos, sem nenhuma perda de interoperabilidade.

É importante notar que as especificações MLHIM estão preocupadas com a interoperabilidade semântica

de todos os sistemas de informação em saúde com todos os aplicativos do campo das ciências biomédicas. Tal fato significa que muitos dos requisitos de desenvolvimento específicos para um determinado tipo de sistema ou aplicativo não estão incluídos nas especificações. Isto compreende conceitos muito importantes, tais como, como persistir as CCDs de formas significativas e úteis, autenticação e autorização, *Application Programming Interfaces* (APIs) e processamento de consultas às bases de dados. Tudo isto está fora do âmbito das especificações MLHIM. Estes outros requisitos são bem definidos em outras especificações e padrões da indústria, e as tentativas para incluí-los dentro das especificações MLHIM serviria apenas para confundir a questão central da interoperabilidade semântica.

Descrição do Modelo de Referência MLHIM

A implementação do Modelo de Referência MLHIM versão 2.4.3 foi executada mediante a produção de um único arquivo no formato XML *Schema Definition* (XSD), de acordo com o padrão W3C XML versão 1.1 (código fonte disponível em <https://github.com/mlhim/specs>). A abordagem de implementação em XML foi baseada em extensões e substituições, de modo a manter a estrutura hierárquica do modelo conceitual.

Os tipos de dados do Modelo de Referência MLHIM são definidos de acordo com o pacote *Datatypes* e originalmente baseados na Norma ISO 21090 com modificações para reduzir a complexidade desnecessária e a dependência semântica. Para qualquer elemento de uma CCD, o elemento 'DvAdapter-dv' deve ser restrinvido por um dos *complexTypes* concretos deste pacote.

Os tipos de dados ordenados das especificações MLHIM compreendem qualquer tipo de dados cujas instâncias podem ser ordenadas; assim são todos os *complexTypes* sob o *complexType* abstrato *DvOrdered*.

Os *complexType*s que herdam de *DvOrdered* permitem a persistência de dados ordinais, como *ranks* e *scores* (*DvOrdinal*), datas e horas (*DvTemporal*) e números reais ou contagens (todos os *complexType*s que herdam de *DvQuantified*) (Tabela 1).

Os tipos de dados não ordenados das especificações MLHIM compreendem qualquer tipo de dados no formato caractere, Booleano ou encapsulados. Alguns desses dados os *complexType*s herdam diretamente do *complexType* abstrato *DvAny* e não têm qualquer outro relacionamento de herança (*DvBoolean* e *DvURI*). Por outro lado, os *complexType*s *DvString* e *DvCodedString* definem um conjunto de tipos de dados que podem conter caracteres (como, por exemplo, *DvIdentifier*), linhas ou parágrafos de texto, e *complexType*s que herdam de *DvEncapsulated*, definem os metadados comuns e permitem a persistência de todos os tipos de dados analisáveis ou multimídia (Tabela 1). Um diagrama UML do pacote *Datatypes* é mostrado na Figura 1.

O pacote *Structures* do Modelo de Referência MLHIM contém o *complexType* abstrato *Item* e os *complexType*s que herdam de *Item*; *DvAdapter* e *Cluster*. Os *Clusters*

são recipientes estruturados, qualquer *complexType* que herda de *Item* (inclusive outros *Clusters*), permitindo a definição de um conceito em saúde com qualquer tamanho ou formato. Os *DvAdapters* são a granularidade mais fina da estrutura do MLHIM Modelo de Referência, onde são atribuídos os tipos de dados para cada variável relacionada a um conceito de saúde.

Uma *Entry* é a raiz de um conjunto lógico de itens de dados. É também a unidade mínima de informação qualquer consulta ao banco de dados deve retornar, uma vez que uma *Entry* inteira (incluindo as subpartes) registra um grande conjunto de metadados relacionados a estrutura espacial, informações de data e hora, definição da trilha para a auditoria dos dados e outras informações contextuais, bem como o sujeito e o produtor da informação, que são requisitos para uma completa interoperabilidade semântica.

Cada *complexType* que herda de *Entry* tem elementos idênticos. A subtipagem é utilizada para permitir aos mecanismos de persistência dos dados a separação dos tipos de *Entry*. Assim, o *complexType* *CareEntry* define os elementos de estrutura de dados, de protocolo e de diretrizes clínicas para todas as entradas de dados clínicos.

	complexType raiz	complexType	Uso
Tipos de Dados Ordenados	<i>DvAny</i>	<i>DvInterval</i> <i>ReferenceRange</i>	Intervalos de tipos de dados que herdam de <i>DvQuantified</i> Intervalos normais ou anormais
	<i>DvOrdered</i> ^a	<i>DvOrdinal</i>	<i>Ranks</i> ou <i>escores</i>
	<i>DvQuantified</i> ^b	<i>DvQuantity</i> <i>DvCount</i> <i>DvRatio</i>	Quantidades (numeros reais) com unidades Dados de contagem Razões, taxas e proporções
	<i>DvAny</i>	<i>DvTemporal</i>	Datas ou horas completas ou incompletas Durações
Tipos de Dados Não Ordenados	<i>DvAny</i>	<i>DvBoolean</i> <i>DvURI</i> <i>DvString</i>	Dados Booleanos verdadeiros (e.g. TRUE/FALSE) <i>Uniform Resource Identifiers</i> (URIs) Caracteres alfanuméricos
	<i>DvString</i>	<i>DvCodedString</i> <i>DvIdentifier</i>	Termos provenientes de vocabulários controlados Identificadores
	<i>Dv Encapsulated</i> ^a	<i>DvMedia</i> <i>DvParsable</i>	Dados multimídia e seus metadados Caracteres encapsulados analisáveis

Tabela 1. Modelo de Referência MLHIM: pacote *Datatypes*. ^a*complexType* que herda de *DvAny*; ^b*complexType* que herda de *DvOrdered*.

Datatypes Package

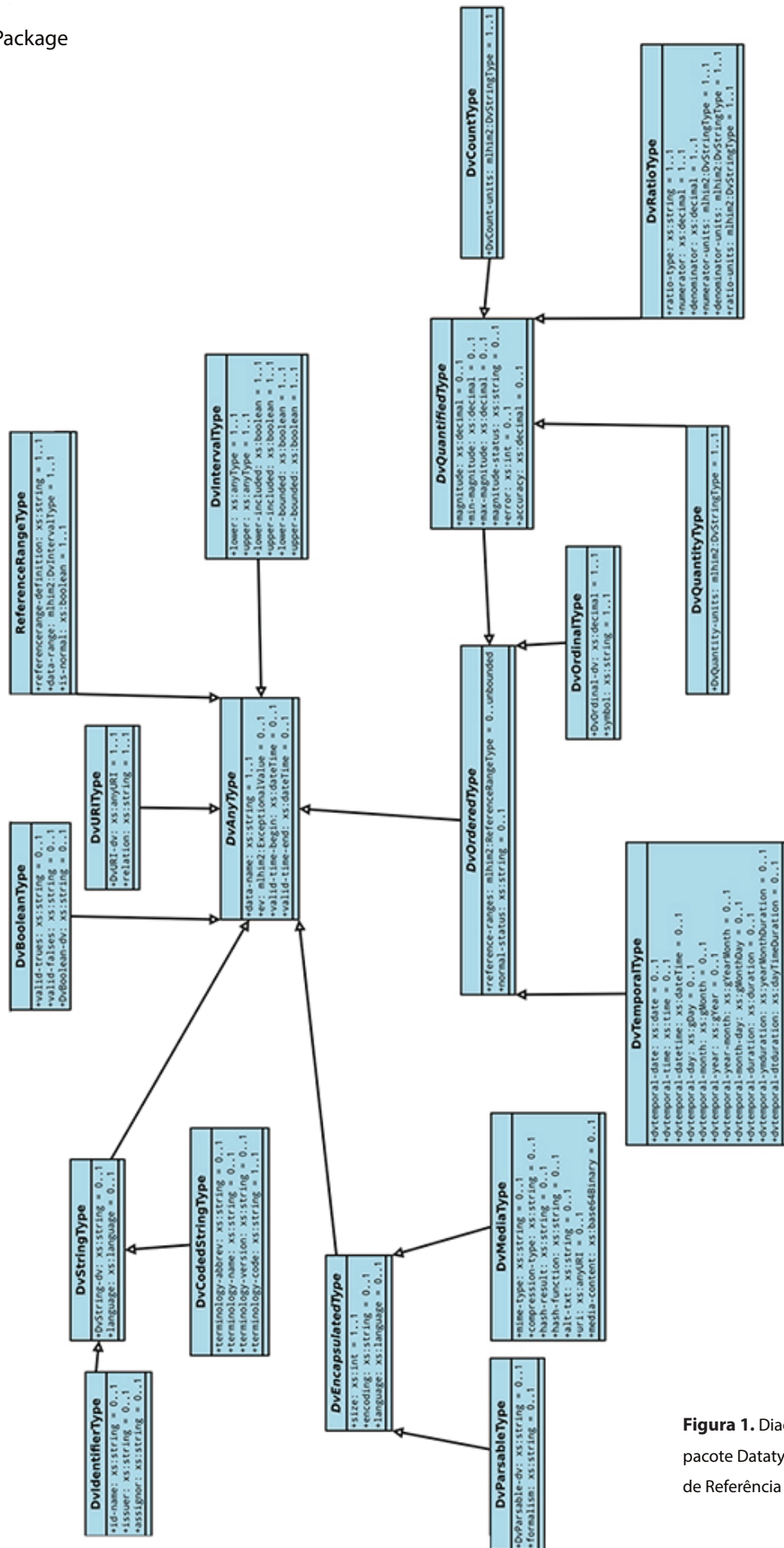


Figura 1. Diagrama UML do pacote Datatypes do Modelo de Referência MLHIM.

O *complexType AdminEntry* é utilizado para o registro de informações administrativas relacionadas ao processo clínico, mas que não são clinicamente relevantes por si próprias, tais como dados de admissão, episódio de atendimento, localização na enfermaria, alta e agendamentos. O *complexType DemographicEntry* é utilizado para registrar informações demográficas, como as estruturas relacionadas a nomes, funções e localizações. O *DemographicEntry* foi modelado nas especificações MLHIM como um *complexType* em separado de *CareEntry*

e *AdminEntry*, a fim de facilitar a separação entre as informações da clínicas e não-clínicas e, principalmente, para permitir a de-identificação de dados clínicos e administrativos.

Finalmente, o pacote *Constraint* é composto do *complexType CCD*, que apresenta um elemento chamado 'definition', o qual deve ser restrito por qualquer um dos *complexTypes* que herdam de *Entry* (Tabela 2). Um diagrama UML dos pacotes *Structures*, *Content* e *Constraint* é apresentado na Figura 2.

complexType raiz	complexType	Uso
Item (<i>Structures</i>)	<i>DvAdapter</i> <i>Cluster</i>	A variante elementar de <i>ItemType</i> , cuja função é ser um adaptador de <i>Data types</i> A variante de agrupamento de <i>ItemType</i> , que pode conter <i>DvAdapters</i> e/ou outros <i>Clusters</i> em uma lista ordenada
Entry (<i>Content</i>)	<i>CareEntry</i> <i>AdminEntry</i> <i>DemographicEntry</i>	Recipiente de dados de saúde Recipiente de dados administrativos Recipiente de dados demográficos
CCD (<i>Constraint</i>)	CCD	Define as restrições ao Modelo de Referência para um determinado conceito em saúde

Tabela 2. Modelo de Referência MLHIM: pacotes *Structures*, *Content* e *Constraint*

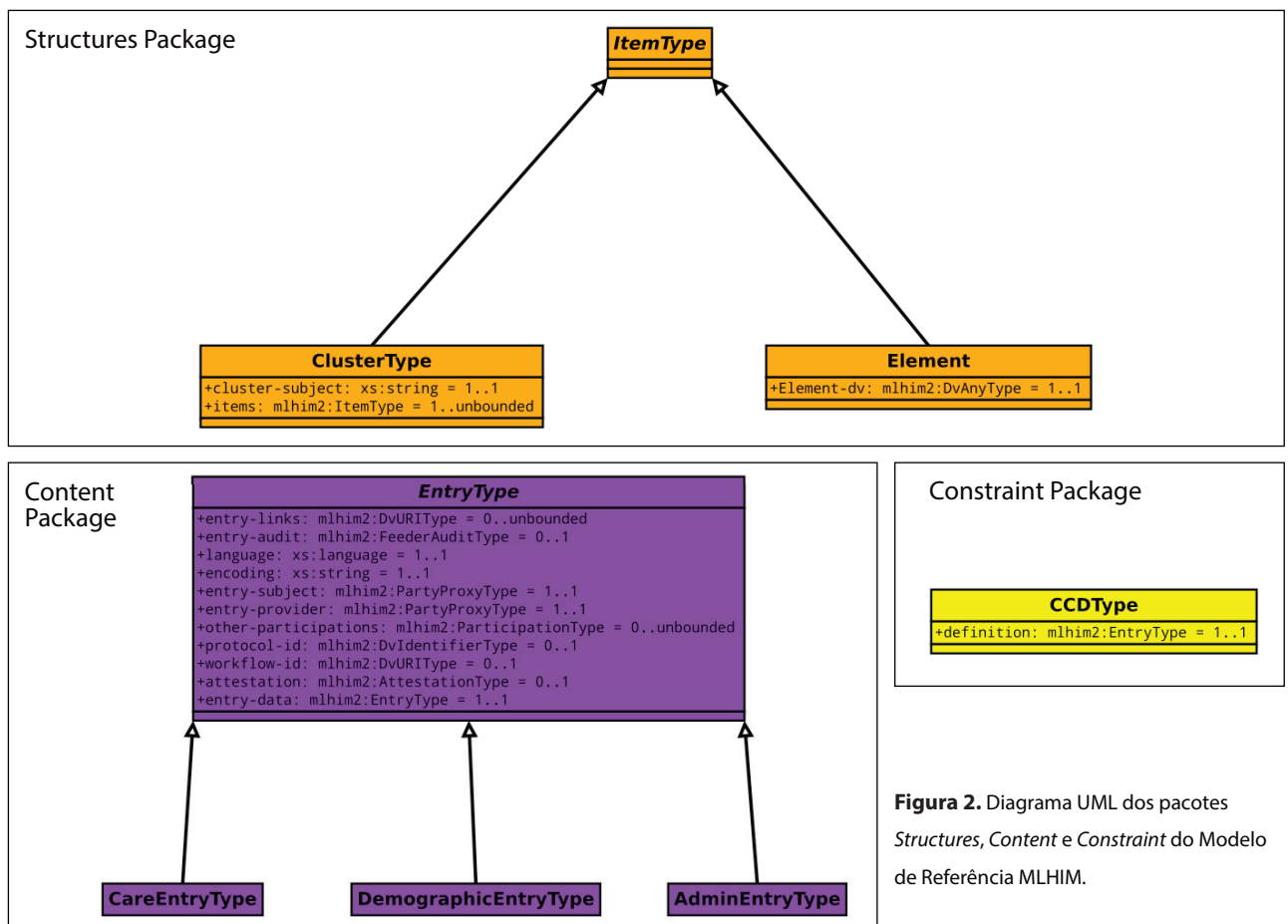


Figura 2. Diagrama UML dos pacotes *Structures*, *Content* e *Constraint* do Modelo de Referência MLHIM.

O pacote *Common* é composto por *complexType*s que herdam diretamente de 'xs: anyType' das especificações W3C XML Schema 1.1, contendo componentes de informação que são necessários para todos os tipos de CCDs, tais como a identificação do sujeito do cuidado, do profissional de saúde e de outros participantes, bem como da trilha de auditoria completa dos dados e de informações sobre os valores excepcionais (Tabela 3). Um diagrama UML do pacote *Common* é apresentado na Figura 3.

Desenvolvimento do Aplicativo de Demonstração

Dois aplicativos de demonstração foram desenvolvidos a partir do MLHIM *Demo* EMR, uma *framework* para o desenvolvimento de aplicativos para as especificações MLHIM com uma base de dados eXist-DB (código-fonte disponível em <https://github.com/mlhim/mlhim-emr>). Os modelos de dados dos aplicativos de demonstração

complexType raiz	complexType	Uso
<i>PartyProxy</i>	<i>PartySelf</i> <i>PartyIdentified</i>	Representação do sujeito a que se refere o registro. Dados de proxy para uma parte identificada que não seja o sujeito a que se refere o registro.
Todos	<i>Participation</i> <i>Attestation</i> <i>FeederAudit</i> <i>FeederAuditDetails</i>	Modela a participação de uma parte em uma atividade. Registra o atestado do conteúdo registrado por uma parte. Metadados de auditoria e outros referente à cadeia de alimentação dos dados nos sistemas. Detalhes de auditores para qualquer sistema participante da cadeia de alimentação dos dados.
<i>ExceptionalValue</i>	Ver [36]	Ver [36]

Tabela 3. Modelo de Referência MLHIM: pacote *Common*

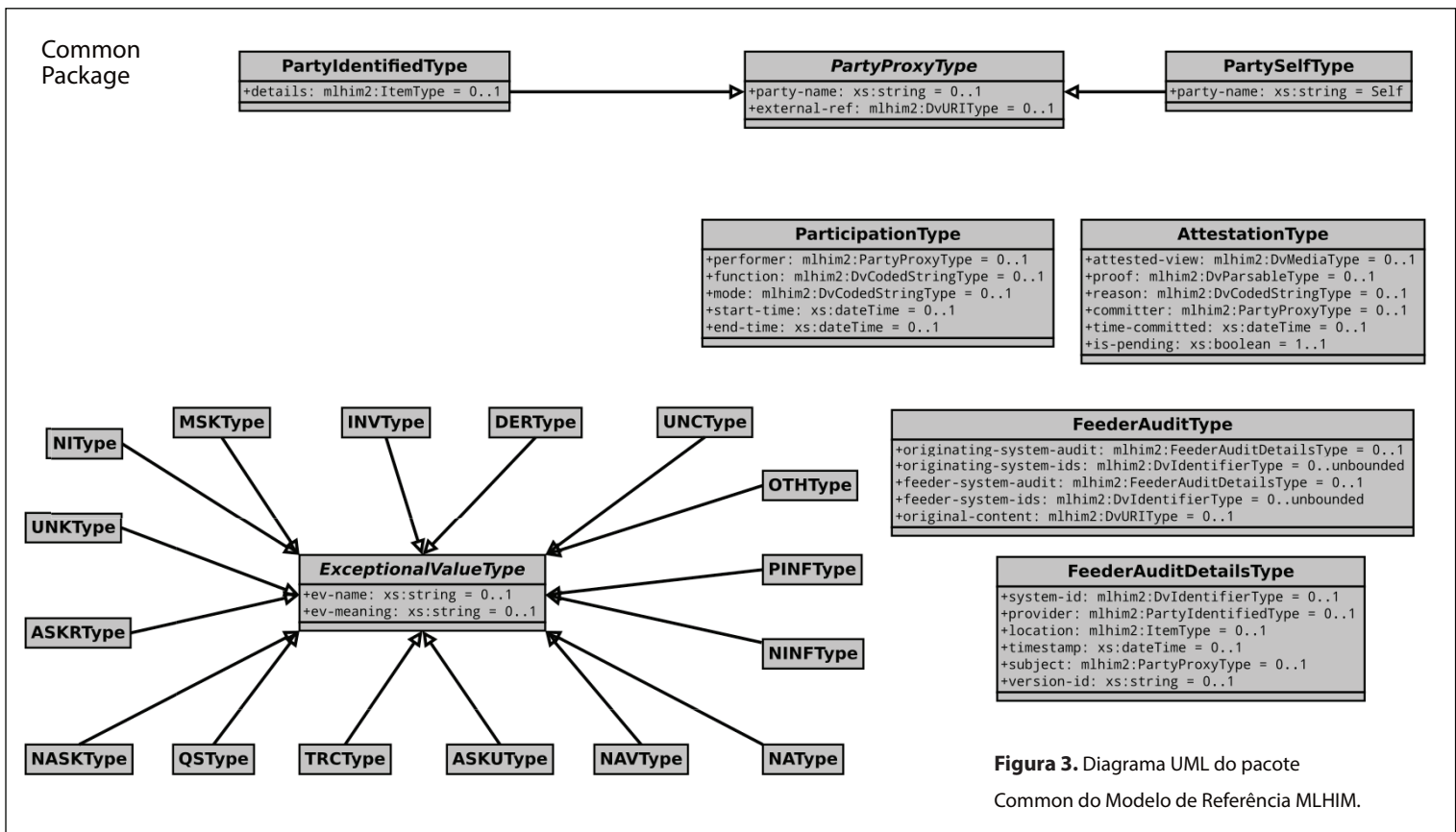


Figura 3. Diagrama UML do pacote *Common* do Modelo de Referência MLHIM.

foram baseados em uma seleção de um conjunto de *Common Data Elements* (CDE), desenvolvido pelo *National Cancer Institute*, disponível no endereço eletrônico do NCI CDE Browser (<https://cdebrowser.nci.nih.gov/CDEBrowser/>), referente aos Dados Demográficos e de Sinais Vitais (Demo 1) e Dados Demográficos e Bioquímicos (Demo 2). Os CDEs foram mapeados para CCDs com o uso do aplicativo *Concept Constraint Definition Generator* (CCD-Gen), um editor de CCDs MLHIM disponível na Internet (www.ccdgen.com), que gerou o código em XSD para os *complexType*s plugáveis (*Pluggable Complex Types – PCTs*¹⁸) referentes a para cada CDE. Uma mistura de CCDs baseados nas versões 2.4.2. e 2.4.3 do Modelo de Referência MLHIM foram usados para demonstrar a manutenção da validade das instâncias de dados baseadas nas especificações MLHIM ao longo do tempo, mesmo que o Modelo de Referência seja modificado em versões posteriores. As CCDs foram validadas e instâncias de dados XML simuladas foram geradas para cada CCD pelo uso da editor XML oXygen versão 14.2 e persistidas em um banco de dados eXist-DB.

Um design de aplicativos minimalista foi adotado para o desenvolvimento dos aplicativos de demonstração, que apresentam a finalidade exclusiva de demonstrar a interoperabilidade fornecida pelas especificações MLHIM e não representam uma implementação em escala industrial de um RES ou aplicativo médico totalmente funcional. Ambos os aplicativos foram implementados contendo algumas instâncias de dados do outro aplicativo de demonstração. O registro dos identificadores e dos dados demográficos dos pacientes simulados era idêntico em ambos os aplicativos, uma vez que não foi o objetivo deste trabalho abordar a questão da vinculação de registros (record linkage) dos dados, dado que esta questão está igualmente fora do alcance da interoperabilidade semântica. As CCDs utilizadas estão disponíveis na CCD Library do CCD-Gen.

Resultados

A obtenção da interoperabilidade semântica entre os dois aplicativos de demonstração foi baseada em dois elementos principais: as definições dos modelos de dados em CCDs, e da cadeia reversa de validação, a partir da instância de dados para o arquivo XSD da CCD, deste para o arquivo XSD do Modelo de Referência MLHIM e, finalmente, para as especificações W3C XML *Schema Definition* 1.1. e para as especificações W3C para a linguagem XML.

Modelagem de Dados

O CCDType “Dados Demográficos” foi restringido para um *complexType DemographicEntry*, que contém um *ClusterType* incluindo *DvAdapterTypes* para detalhes do paciente e para dados de endereço. Os CCDTypes “Sinais Vitais” e “Bioquímico” foram restringidos para um *complexType CareEntry* cada. A CCD “Sinais Vitais” inclui dados de pressão arterial, temperatura, frequência cardíaca e respiratória; A CCD “Bioquímico” define o modelo de dados para o registro dos valores dos exames laboratoriais de sódio, potássio, glicose, ureia e creatinina do paciente.

O processo de modelagem de dados definiu o tipo para cada elemento de dados, de acordo com o pacote *Datatypes* das especificações MLHIM, tal como definido na Tabelas 1 e na Figura 1. Por exemplo, para a definição do elemento de dados “Sexo”, escolheu-se *DvStringType*, e as restrições foram feitas para o seu elemento ‘enumeration’ correspondente, de forma a restringir os valores possíveis para “Masculino”, “Feminino”, “Desconhecido” e “Não Especificado”. O mesmo processo foi repetido para cada um dos elementos de dados incluídos nas CCDs, de acordo com os requisitos específicos de cada tipo de dados, tal como definido nas especificações MLHIM.

CCD	Elemento de Dados	Tipo de Dados MLHIM
Dados Demográficos	Gênero CEP Estado Cidade RG CPF Telefone Email Primeiro Nome Último Nome	<i>DvString</i> com enumeração <i>DvIdentifier</i> <i>DvCodedString</i> <i>DvCodedString</i> <i>DvIdentifier</i> <i>DvIdentifier</i> <i>DvString</i> <i>DvURI</i> <i>DvString</i> <i>DvString</i>
Sinais Vitais	Pressão Arterial Sistólica Pressão Arterial Diastólica Tipo de Esfigmomanômetro Localização do Manguito Posição do Paciente Frequência Cardíaca Frequência Respiratória Temperatura Corporal Local de Tomada da Temperatura Tipo de Termômetro	<i>DvQuantity</i> <i>DvQuantity</i> <i>DvString with enumeration</i> <i>DvString with enumeration</i> <i>DvString with enumeration</i> <i>DvCount</i> <i>DvCount</i> <i>DvQuantity</i> <i>DvString with enumeration</i> <i>DvString with enumeration</i>
Bioquímica	Sódio Potássio Glicose Ureia Creatinina	<i>DvQuantity</i> <i>DvQuantity</i> <i>DvQuantity</i> <i>DvQuantity</i> <i>DvQuantity</i>

Tabela 4. Resultados da modelagem de dados para os conceitos de Dados Demográficos, Sinais Vitais e Bioquímica como CCDs MLHIM.

Após a definição dos tipos de dados para todos os *DvAdapterTypes*, eles foram combinados em uma *ClusterType*. No CCD-Gen, este procedimento requer a seleção dos *DvAdapterTypes* que compõem um dado *ClusterType*. Para este aplicativo de demonstração, um único *ClusterType* foi definido para cada um das CCDs, incluindo todos os *DvAdapterTypes* correspondentes, como visto na Tabela 4.

Estes *ClusterTypes*, contendo todos os *DvAdapterTypes* de Cada CCD, foram associados a um *ComplexType* abaixo de *EntryType* (Tabela 3), ou seja, *DemographicEntryType*, *AdminEntryType* ou *CareEntryType*. No CCD-Gen, esta associação é feita pela seleção do Cluster que contém os dados correspondentes a cada CCD, que são incluídos no *complexType* derivado de *EntryType* como

valor para o elemento "entry-data". Neste exemplo, a CCD "Dados Demográficos" foi modelada como um *DemographicEntryType*, e as CCDs "Sinais Vitais" e "Bioquímico" foram modeladas como *CareEntryTypes*. Para completar a geração da CCD, metadados no formato *Dublin Core Metadata Initiative* (DCMI) foram incluídos na secção correspondente das CCDs.

Prova de Conceito

As CCDs "Dados Demográficos", "Sinais Vitais" e "Bioquímico" definiram as instâncias de dados simulados em XML para 130 pacientes fictícios, cada um deles com uma instância da CCD "Dados Demográficos" e n ($n = 1, 2, 3 \dots$) instâncias de dados para as CCDs "Sinais vitais"

e “Bioquímico”, resultando, por exemplo, em 1.531 instâncias de dados de pressão arterial diastólica para a CCD “Sinais Vitais”. Todas as instâncias de dados foram válidas de acordo com as CCDs correspondentes, que eram válidas de acordo com o arquivo XSD do Modelo de Referência MLHIM correspondente (2.4.1 ou 2.4.2), que é válido de acordo com as especificações W3C XML *Schema Definition* 1.1 e com as especificações W3C para a linguagem XML. Portanto, as especificações MLHIM obtiveram uma cadeia completa de validação reversa, a partir da instância de dados até as especificações W3C XML. Isso foi repetido para todas as instâncias de dados, com uma taxa de sucesso de 100%. A Figura 4 mostra uma XQuery realizada no banco de dados usando *Exide*, um editor online de XQuery.

aplicativos de demonstração pequenos, de modo que todo o processo possa ser visualizado de uma só vez, sem tornar a análise demasiado difícil.

Uma vez que a unidade de troca é o conceito, tal como definido por uma CCD, e a representação de dados em XML está disponível para todas as plataformas computacionais, fica, portanto, comprovado que qualquer tipo de sistema de informações em saúde ou aplicativos médicos pode ser acomodado no ecossistema MLHIM.

Discussão

Este estudo apresentou o processo de desenvolvimento, em código-fonte aberto, de uma especificação baseada nos princípios de modelagem multinível e em padrões da indústria. Os resultados mostraram que a adoção de tecnologias XML, implementadas de acordo com abordagem multinível para o desenvolvimento de software, permitiu o estabelecimento de uma cadeia de validação reversa a partir da instância de dados até as especificações originais W3C XML.

A real vantagem da adoção de tecnologias XML para o desenvolvimento das especificações MLHIM foi o potencial de desenvolvimento de aplicativos semanticamente interoperáveis para serviços de saúde reais, independentemente do tamanho do aplicativo ou de como ele será utilizado. Dado que XML é um padrão da indústria ubíquo e todas as principais linguagens de programação têm ferramentas de associação com XML *Schemas*, isso permite que os desenvolvedores de aplicativos trabalhem na sua linguagem preferida, usando os modelos de persistência de sua preferência, e mesmo assim não constroem “silos de dados”¹⁸. Nas aplicações baseadas em MLHIM podem persistir dados em XML nativo e outros tipos de bancos de dados NoSQL, bem como bancos de dados SQL, ferramentas para geração de interfaces de usuário a partir de XSD, seja no formato

The screenshot shows the Exide XQuery editor interface. The main window displays an XQuery script named 'BMP_Average_Values.xql'. The query uses XQuery 3.0 syntax to calculate average values for various medical parameters from a collection of XML documents. The results pane at the bottom shows the output of the query, which is an XML document containing the calculated averages for parameters like sodium, glucose, potassium, urea, and creatinine.

```

1 xquery version "3.0";
2 declare namespace mlhlm2="http://www.mlhlm.org/xmlns/mlhlm2/2_4_1";
3 let $title := "BMP Average Values";
4 let $bmp := collection("/db/apps/MLHIM/MLHIM-EMR")/mlhlm2:el-8c71fe4c-8dd2-4d0f-af05-15b5b7b9de24;
5 let $total := count($bmp);
6 let $savg := avg($bmp/mlhlm2:el-f6c5eade-6458-4799-874d-7f3d365d268d/mlhlm2:magnitude);
7 let $psavg := avg($bmp/mlhlm2:el-287fec54-254b-4b66-9c42-3b275fcd1f38/mlhlm2:magnitude);
8 let $pavg := avg($bmp/mlhlm2:el-781e9dda-855a-4a95-bee3-d482c44d1186/mlhlm2:magnitude);
9 let $uavg := avg($bmp/mlhlm2:el-866a221b-9c42-48cc-9c18-be99866222/mlhlm2:magnitude);
10 let $cavg := avg($bmp/mlhlm2:el-5186df95-13b3-4e25-9c8b-5e6d43eab79/mlhlm2:magnitude);
11
12 return
13
14 <results>
15 <ReportTitle>{$title}</ReportTitle>
16 <sodium>{$savg}</sodium>
17 <glucose>{$psavg}</glucose>
18 <potassium>{$pavg}</potassium>
19 <urea-bun>{$uavg}</urea-bun>
20 <creatinine>{$cavg}</creatinine>
21
22 <Total>{$total}</Total>
23 </results>

```

Figura 4. XQuery realizada no banco de dados usando Exide.

Assim, a prova de conceito da obtenção da interoperabilidade semântica entre os aplicativos de demonstração baseados em MLHIM e desenvolvidos especificamente para este estudo é demonstrada na capacidade de embaralhar as instâncias de dados entre as aplicações e, mesmo assim, as instâncias de dados mantêm a capacidade de manter o registro da semântica específica de cada conceito, assim como mantêm as restrições sintáticas exatamente como foram projetados para esta mesma semântica no momento de modelagem da CCD. Optou-se por realizar esta prova de conceito com

XForms ou mediante a adoção de outras *frameworks* específicas para outras linguagens de programação, conforme exigido para cada aplicativo.

O uso de tecnologias XML também permite o uso de ferramentas e tecnologias da emergente *Web Semântica*, permitindo que as CCDs sejam marcadas pelo prevalente RDF e outras marcações que permitam a busca semântica através de instâncias de dados em conformidade com as especificações¹⁹. A singularidade na abordagem das especificações MLHIM é que não se faz marcação nas instâncias de dados, mas nas CCDs a que as instâncias de dados se referem através de suas restrições sintáticas e semânticas. Esta abordagem reduz o tamanho e a sobrecarga globais, nos processos de consulta e intercâmbio referentes aos bancos de dados.

O processo de modelagem de conhecimento adotado neste estudo foi baseado nas especificações MLHIM. O processo de modelagem das CCDs foi uma tarefa relativamente simples para os especialistas do domínio (dois médicos), que ficaram responsáveis apenas por selecionar os conceitos no NCI CDE *Browser*, definir seus tipos de dados de acordo com o pacote MLHIM *Datatypes* e por fim definir as restrições para cada PcT, usando a interface de usuário do CCD-Gen. É importante notar que há relatos na literatura considerando a elaboração de arquétipos *openEHR* bastante complexa²⁰, o que não foi o caso para as CCDs MLHIM.

É importante notar que os sistemas que utilizam os conceitos expressos nas especificações MLHIM não precisam ter o Modelo de Referência nem mesmo as CCDs em seu código-fonte para serem semanticamente interoperáveis. É por isso que a cadeia de validação é importante. Desenvolver novos aplicativos com base no Modelo de Referência MLHIM é uma boa prática; no entanto, isso não é necessário para a eficácia da interoperabilidade semântica obtida através das especificações, o que é assegurado tão somente pela troca de documentos que contenham instâncias de

dados válidas e suas correspondentes CCDs. Visto que as especificações MLHIM são baseadas em um padrão da indústria sustentável e globalmente utilizado, a representação do Modelo de Referência e das CCDs pode ser incorporada a praticamente qualquer aplicativo e em qualquer linguagem de programação. A única exigência será que o aplicativo deve ser capaz de importar e exportar instâncias de dados válidas de acordo com a CCD correspondente. Isso fornece uma cadeia de validação completa (que nenhum outro padrão da área de informática em saúde fornece na atualidade), a partir da instância de dados para a CCD, e desta para o XSD do Modelo de Referência MLHIM, a seguir para as especificações W3C XML *Schema* 1.1 e, finalmente, para a especificação W3C da linguagem XML.

Entretanto, a adoção de qualquer solução tecnológica para um problema social requer acordos, e o domínio da saúde não é diferente. Assim, o primeiro grande obstáculo para a adoção de uma tecnologia como MLHIM é uma mudança na mentalidade de desenvolvimento de *software* convencional (de um nível) para a modelagem multinível. Há comentários anedóticos de que, dentro do domínio da informática de saúde, esta mudança de consciência é semelhante ao que foi exigido dos cosmólogos na transição do modelo geocêntrico para o heliocêntrico. Este é um desafio para muitos desenvolvedores de *software* que foram ensinados a desenvolver sistemas modelados em um nível. O mesmo pode se dizer em relação à mudança de paradigma entre os especialistas em XML, de não mais marcar semanticamente as instâncias de dados, mas sim os XSDs (no caso, as CCDs).

Porém, a questão certamente mais difícil é a necessidade de participação dos especialistas de domínio no processo de modelagem das CCDs. Embora já haja um consenso de que os profissionais de saúde devem ser incluídos no processo de design dos aplicativos que irão utilizar, isto ainda não é considerado como uma parte

formal do processo de trabalho em saúde. Para superar esse obstáculo, há a necessidade do surgimento de uma nova área de especialização em ciências biomédicas: a modelagem do conhecimento. Um especialista em modelagem do conhecimento em saúde deve ser especificamente treinado para obter o conhecimento do domínio dos profissionais de saúde e transformá-lo em modelos de conceito computáveis, tais como as CCDs MLHIM.

Conclusão

Os resultados deste estudo demonstraram que a interoperabilidade semântica em sistemas de informação de saúde pode ser obtida com a adoção da modelagem multinível, que é implementável em tecnologias XML pelas especificações MLHIM. Embora o objetivo geral das especificações MLHIM seja promover, em longo prazo, a interoperabilidade semântica e sintática em todos os sistemas de informação em saúde em escala global, mesmo aplicativos autocontidos podem se beneficiar das tecnologias MLHIM (por exemplo, uma empresa de *software*, uma localidade, um Estado). Mesmo nesses casos, é possível a criação de aplicativos que já são interoperáveis “de fábrica” e exigem menos sobrecarga de manutenção enquanto a ciência dos cuidados em saúde evolui em suas dimensões temporais, espaciais e ontológicas.

Referências

1. ACMI. 1961 electronic medical records. Disponível em: <http://www.youtube.com/watch?v=t-aiKllc6uk>. Acesso em 01/04/2013.
2. De Leon S, Connelly-Flores A, Mostashari F, Shih SC, The business end of health information technology. Can a fully integrated electronic health record increase provider productivity in a large community practice? *J Med Pract Manage* 2010; 25:342-9.
3. U.S. National Library of Medicine. 2011AA SNOMED CT Source Information. Disponível em: <http://www.nlm.nih.gov/research/umls/sourcereleasedocs/current/SNOMEDCT>. Acesso em 28/04/2013.
4. Maojo V, Kulikowski C. Medical informatics and bioinformatics: integration or evolution through scientific crises? *Methods Inf Med* 2006; 45:474-82.
5. Zusman EE. Form facilitates function: innovations in architecture and design drive quality and efficiency in healthcare. *Neurosurgery* 2010;66:N24.
6. Sittig DF, Singh H. A new sociotechnical model for studying health information technology in complex adaptive healthcare systems. *Qual Saf Health Care* 2010;Suppl 3:i68-74.
7. Raths D. Shifting away from silos. the interoperability challenges that hospitals face pale in comparison to the headaches plaguing State Departments. *Healthc Inform* 2010;27:32-
8. Blobel B, Pharos P. Analysis and evaluation of EHR approaches. *Stud Health Technol Inform* 2008;136:359-64.
9. Blobel B. Ontologies, knowledge representation, artificial intelligence: hype or prerequisites for international pHealth interoperability? *Stud Health Technol Inform* 2011; 165:11-20.
10. Saleem JJ, Russ AL, Neddo A, Blades PT, Doebbeling BN, Foresman BH. Paper persistence, workarounds, and communication breakdowns in computerized consultation management. *Int J Med Inform* 2011;80(7):466-79.
11. Kalra D, Beale T, Heard S. The openEHR Foundation. *Stud Health Technol Inform* 2005; 115:153-73.
12. Martinez-Costa C, Menarguez-Tortosa M, Fernandez-Breis JT. Towards ISO 13606 and openEHR archetype-based semantic interoperability. *Stud Health Technol Inform* 2009;150:260-4.
13. Cavalini LT, Cook TW. Use of XML Schema Definition for the development of semantically interoperable healthcare applications. *Lecture Notes in Computer Science* 2014; 8315:125-45.
14. Menezes AL, Cirilo CE, Moraes JLC, Souza WL, Prado AF. Using archetypes and domain specific languages on development of ubiquitous applications to pervasive healthcare. *Proc IEEE 23rd Int Symp Comput Bas Med Syst* 2010; 395-400.
15. Cavalini LT, Cook TW. Knowledge engineering of healthcare applications based on minimalist multilevel models. *IEEE 14th Int Conf e-Health Networ Appl Serv* 2012; 431-4.
16. Sanderson DB. Loss of data semantics in syntax directed translation. PhD Thesis in Computer Sciences. New York 1994; Rensselaer Polytechnic Institute.
17. Cook TW, Cavalini LT. Implementing a specification for exceptional data in multilevel modeling of healthcare applications. *ACM SIGHIT Rec* 2012; 2:11.
18. Lee T, Hon CT, Cheung D. XML Schema design and management for e-government data interoperability. *Electr J e-Gov* 2009; 381-391.
19. Daconta MC, Obrst LJ, Smith KT. *The Semantic Web*. Indianapolis 2003; Wiley.
20. Kashfi H, Torgerson O. A migration to an openEHR-based clinical application. *Stud Health Technol Inform* 2009; 150:152-6.