REFERENCE MODEL

The *open*EHR EHR Archetype Model

Editors:{T Beale, S Heard}[1]*, {D Kalra, D Lloyd}*[2]

Revision: 0.7

Pages: 23

1. Ocean Informatics Australia
2. Centre for Health Informatics and Multi-professional Education, University College London

The *open*EHR foundation

is an independent, non-profit community, facilitating the creation and sharing of health records by consumers and clinicians via open-source, standards-based implementations.

Founding Chairman	David Ingram, Professor of Health Informatics, CHIME, University College London
Founding Members	Dr P Schloeffel, Dr S Heard, Dr D Kalra, D Lloyd, T Beale
Patrons	To Be Announced

email: info@openEHR.org **web**: http://www.openEHR.org

Copyright Notice

Amendment Record

Issue	Details	Who	Completed
0.7	CR-000019. Add HISTORY & STRUCTURE supertype. CR-000028. Change name of STRUCTURE class to avoid clashes.	T Beale H Frankel	03 Mar 2004
0.6.3	CR-000021. Rename CLINICAL_CONTEXT.*practice_setting* to *setting*.	A Goodchild	10 Feb 2004
0.6.2	CR-000013. Rename key classes according to CEN ENV13606. CR-000041. Visually differentiate primitive types in openEHR documents.	D Lloyd, S Heard, T Beale	10 Oct 2003
0.6.1	CR-000005. Rename CLINICAL_CONTEXT to EVENT_CONTEXT. CR-000018. Add DIRECTORY class to AM.EHR Package.	A Goodchild	10 Jun 2003
0.6	CR-000007. Changed C_ENTRY.*c_subject, c_provider.* CR-000009. Merge C_ENTRY.*c_protocol* and *c_reasoning.*	S Heard, D Kalra	11 Apr 2003
0.5	CR-000003, CR-000004 changes. Changed package naming, improved heading structures. Removed ORGANISER_TREE, FOLDER_TREE. (Formally validated). Added C_ORGANISER_PROXY, C_ENTRY_PROXY.	T Beale	23 Mar 2003
0.4	Simplifications in line with EHR RM 4.1. Fixed multiplicity of C_OBSERVATION.*c_data*, C_EVALUATION.*c_data*. Merged C_ACTION_SPECIFICATION and C_INSTRUCTION. Added C_CLINICAL_CONTEXT class. Removed existential invariants. **Formally validated using ISE Eiffel 5.2**.	T Beale	25 Feb 2003
0.3	Removed C_PERSISTENT_TRANSACTION, C_EVENT_TRANSACTION.	T Beale	10 Jan 2003
0.2	Corrections	T Beale	10 Jan 2003
0.1	Initial Writing	T Beale	20 Dec 2002

Acknowledgements

Thanks to...

The work reported in this paper has been funded in by a number of organisations, including The University College, London; The Cooperative Research Centres Program through the Department of the Prime Minister and Cabinet of the Commonwealth Government of Australia; Ocean Informatics, Australia.

Andrew Goodchild, senior research scientist at the Distributed Systems Technology Centre, Brisbane provided valuable in-depth comments and insights on all aspects of the model during its early development.

Table of Contents

1 Introduction

1.1 Purpose

This document describes the architecture of the *open*EHR EHR Archetype Model. This model describes the semantics for all archetypes for the EHR, i.e. archetypes for EHR, Folders, Transactions, Organisers, and Entries.

The intended audience includes:

- Standards bodies producing health informatics standards
- Software development groups using *open*EHR
- Academic groups using *open*EHR
- The open source healthcare community

1.2 Related Documents

Prerequisite documents for reading this document include:

- The *open*EHR Modelling Guide
- The *open*EHR Data Types Archetype Model

Other documents describing related models, include:

- The *open*EHR EHR reference model

1.3 Status

This document is under development, and will be published as a proposal for input to standards processes and implementation works.

Future changes will include:

- Currently the UML diagrams are hand-produced. In the next version, the Rational Rose representation will replace these.
- Specific design principles will be referred to throughout the model text, so that readers can easily find the theoretical discussion on which any part of the model is based.

The latest version of this document can be found in PDF and HTML formats at http://www.openEHR.org/Doc_html/Model/Reference/ehr_am.htm. New versions are announced on openehr-announce@openehr.org.

1.4 Peer review

Known omissions or questions are indicated in the text with a “to be determined” paragraph, as follows:

TBD_1: (example To Be Determined paragraph)

Areas where more analysis or explanation is required are indicated with “to be continued” paragraphs like the following:

To Be Continued: more work required

Reviewers are encouraged to comment on and/or advise on these paragraphs as well as the main content. Please send requests for information to info@openEHR.org. Feedback should preferably be discussed on one of the appropriate mailing lists, openehr-technical@openehr.org or openehr-clinical@openehr.org.

1.5 Document Structure

This document commences with a background section which briefly describes the influences behind the *open*EHR model. The work originates in the requirements analysis and proposals for an EHCR architecture developed during the Good European Health Record Project (1992-5; [14]). *Open*EHR has now integrated parallel R&D activities in Europe: the Synapses project (1996-8; [20]), EHCR SupA [12], [13], [15] and SynEx project (1998-2000; [10]); and the Australian GEHR project (1997-2001; [27], [28]). Since 2000 these empirical implementation and validation streams of work have collaborated to share experiences and progressively identify a pathway for a convergence of ideas and formalisms. This document represents the first fruits of this convergence: a common Reference Model drawing on the superset of implementation experience and lessons learned over a decade of R&D in this field.

This work uses the archetype approach [2], and is founded on two analytical characterisations of the clinical domain, namely an ontological analysis and a context analysis, both documented in [3].

The main part of this document describes the formal model.

2 Background

This section describes the inputs to the modelling process which created the *open*EHR Reference Model.

2.1 Requirements

2.2 Design Principles

3 Overview

3.1 Model Overview

FIGURE 1 illustrates the package structure of the *open*EHR EHR Archetype Model.

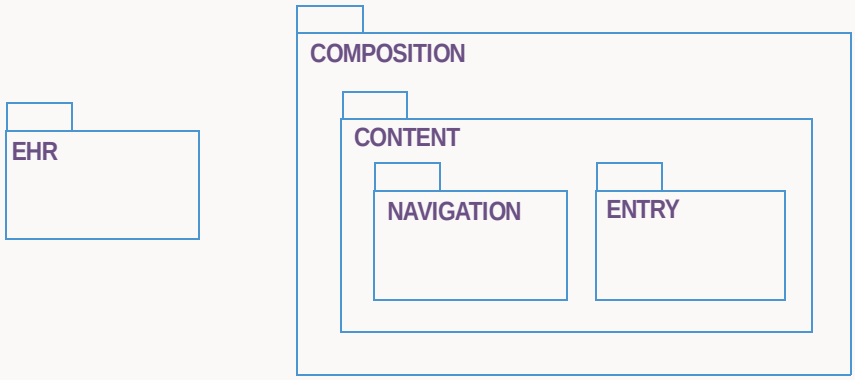

FIGURE 1 AM.EHR and AM.COMPOSITION Packages

4 The AM.EHR Package

4.1 Overview

The EHR archetype package is illustrated in FIGURE 2.

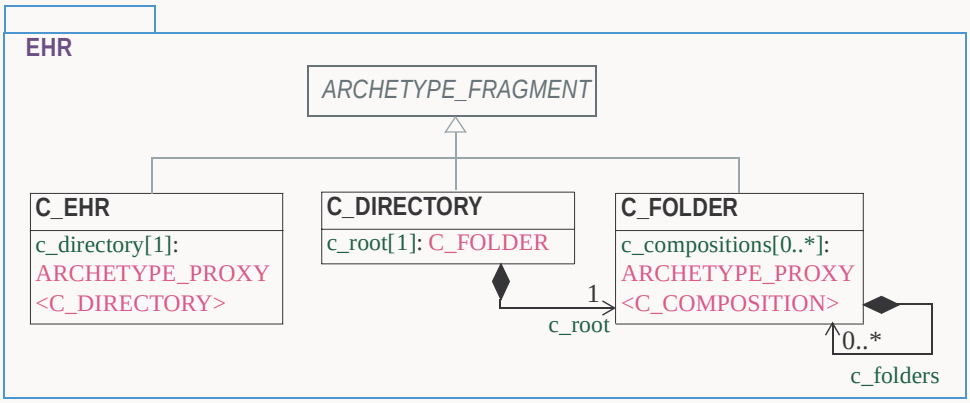

FIGURE 2 AM.EHR Package

4.2 Class Descriptions

4.2.1 C_EHR Class

CLASS	C_EHR	
Purpose	Constrainer class for instances of `EHR`.	
Inherit	`ARCHETYPE_FRAGMENT`	
Attributes	**Signature**	**Meaning**

	c_directory:
	`ARCHETYPE_PROXY`
	`<C_DIRECTORY>`
		Archetype for directory structures of this EHR

Invariants	***c_directory_valid***: c_directory /= Void ***is_archetype_root***: is_archetype_root	

4.2.2 C_DIRECTORY Class

CLASS	C_DIRECTORY	
Purpose	Constrainer class for instances of `DIRECTORY`.	
Inherit	`ARCHETYPE_FRAGMENT`	
Attributes	**Signature**	**Meaning**

CLASS	C_DIRECTORY	
	c_root: C_FOLDER	Root folder of this directory.
Invariants	***c_root_valid***: c_root /= Void ***is_archetype_root***: is_archetype_root	

4.2.3 C_FOLDER Class

CLASS	C_FOLDER	
Purpose	Constrainer class for instances of FOLDER.	
Inherit	ARCHETYPE_FRAGMENT	
Attributes	**Signature**	**Meaning**
	c_folders: C_LIST<C_FOLDER>	Allowed sub-folders of this folder.
	c_compositions: C_LIST <ARCHETYPE_PROXY <C_COMPOSITION>>	Archetypes of composition types allowed under this folder.
Invariants		

4.2.3.1 EHR Path

To Be Continued:

4.2.3.2 Folder Path

To Be Continued:

4.3 AM.COMPOSITION Package

4.3.1 Overview

FIGURE 3 illustrates the *open*EHR COMPOSITION package.

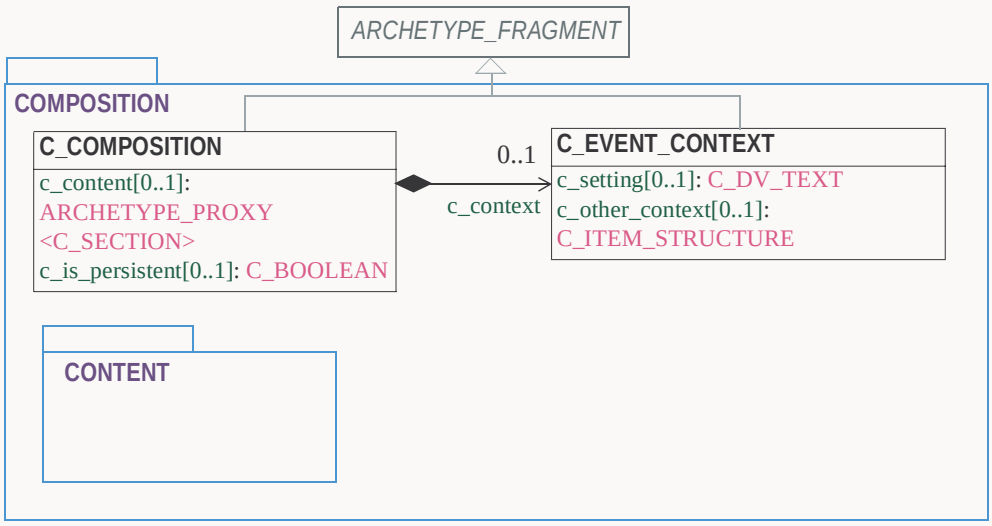

FIGURE 3 AM.COMPOSITION Package

4.4 Class Descriptions

4.4.1 C_COMPOSITION Class

CLASS	C_COMPOSITION	
Purpose	Constrainer class for instances of `COMPOSITION`.	
Inherit	`ARCHETYPE_FRAGMENT`	
Attributes	**Signature**	**Meaning**
	c_content: `C_SECTION_PROXY`	Link to other archetypes which describe the content of this Transaction.
	c_context: `C_EVENT_CONTEXT`	Constrainer for instances of `EVENT_CONTEXT`.
	c_is_persistent: `C_BOOLEAN`	Indicates which statuses of persistent and non-persistent this Transaction can have.
Invariants	***is_archetype_root***: is_archetype_root	

4.4.2 C_EVENT_CONTEXT Class

CLASS	C_EVENT_CONTEXT	
Purpose	Constrainer class for instances of `EVENT_CONTEXT`.	
Inherit	`ARCHETYPE_FRAGMENT`	
Attributes	**Signature**	**Meaning**
	c_setting: `C_DV_TEXT`	Constraint on setting
	c_other_context: `C_ITEM_STRUCTURE`	Constrainer on other context.
Invariants	***not_archetype_root***: ***not*** is_archetype_root	

5 AM.COMPOSITION.CONTENT.NAVIGATION Package

5.1 Overview

The NAVIGATION Package defines constraint structures for nested heading structures. It is illustrated in FIGURE 4.

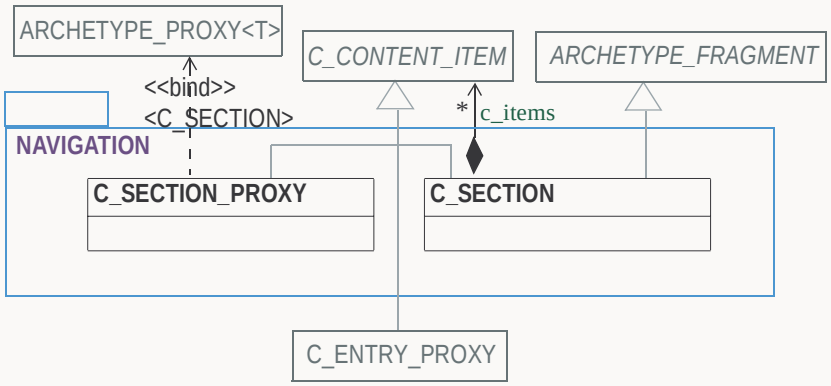

FIGURE 4 AM.COMPOSITION.CONTENT.NAVIGATION Package

5.2 Class Descriptions

5.2.1 C_SECTION Class

CLASS	C_SECTION	
Purpose	Constrainer class for instances of `SECTION`.	
Inherit	`C_CONTENT_ITEM, ARCHETYPE_FRAGMENT`	
Attributes	**Signature**	**Meaning**
	c_items : `C_LIST<C_CONTENT_ITEM>`	
Invariants		

5.2.1.1 C_Organiser Paths

Organiser paths are built from the values of the *meaning* attribute of each `SECTION` in a section structure, as a concatenation with separator characters, as follows:

- {"/" name }*

Examples include:

- "`/SOAP headings/problem/Plan`"

5.2.2 C_SECTION_PROXY Class

CLASS	C_SECTION_PROXY	
Purpose	Proxy for a new `C_SECTION` archetype.	
Inherit	`C_CONTENT_ITEM`, `ARCHETYPE_PROXY<C_SECTION>`	
Attributes	**Signature**	**Meaning**
Invariants		

6 AM.COMPOSITION.CONTENT.ENTRY Package

6.1 Overview

FIGURE 5 shows the ENTRY archetype package.

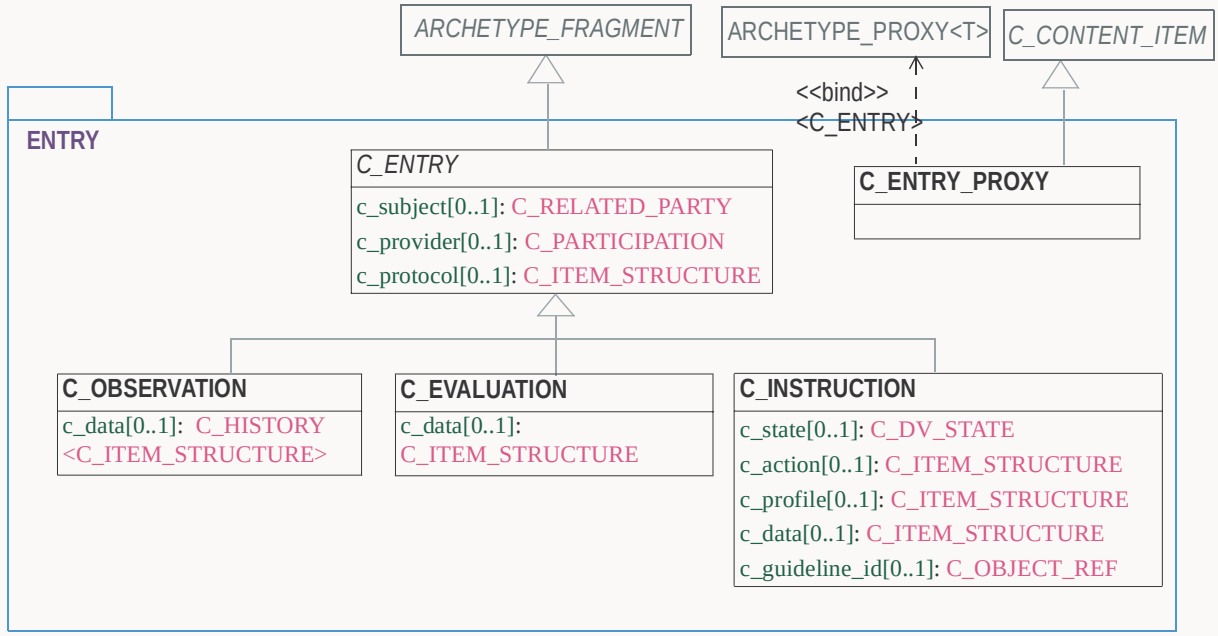

FIGURE 5 AM.COMPOSITION.CONTENT.ENTRY Package

6.2 Class Descriptions

6.2.1 C_ENTRY Class

CLASS	*C_ENTRY (abstract)*	
Purpose	Constrainer type for instances of `ENTRY`.	
Inherit	`ARCHETYPE_FRAGMENT`	
Attributes	**Signature**	**Meaning**
	c_subject: `C_RELATED_PARTY`	Constraint on valid subjects.
	c_provider: `C_PARTICIPATION`	Constraint on valid providers.
	c_protocol: `C_ITEM_STRUCTURE`	Constraint structure for protocol of `ENTRY`
Invariants	***is_archetype_root***: is_archetype_root	

6.2.2 C_OBSERVATION Class

CLASS	C_OBSERVATION	
Purpose	Constrainer type for instances of OBSERVATION.	
Inherit	C_ENTRY	
Attributes	Signature	Meaning
	c_data: C_HISTORY <C_ITEM_STRUCTURE>	Constraint on Observation data structure
Invariants		

6.2.3 C_EVALUATION Class

CLASS	C_EVALUATION	
Purpose	Constrainer type for instances of type EVALUATION.	
Inherit	C_ENTRY	
Attributes	Signature	Meaning
	c_data: C_ITEM_STRUCTURE	
Invariants		

6.2.4 C_INSTRUCTION Class

CLASS	C_INSTRUCTION	
Purpose	Constrainer type for instances of INSTRUCTION.	
Inherit	C_ENTRY	
Attributes	Signature	Meaning
	c_state: C_DV_STATE	State machine definition for this instruction
	c_action: C_ITEM_STRUCTURE	Structure definition for actions to be performed in this instruction
	c_profile: C_ITEM_STRUCTURE	Profile definition for this instruction
	c_data: C_ITEM_STRUCTURE	Data of this instruction

CLASS	C_INSTRUCTION	
	c_guideline_id : C_OBJECT_REF	Guidelines to use for this Instruction
Invariants		

6.2.4.1 C_ENTRY Archetype Paths

Entry archetype paths are based on the value of the *meaning* attribute:

- "|<C_ENTRY.meaning>"

Examples include:

- "|ECG results"
- "|blood pressure"

The paths to access the structures connected by the *item* and *protocol* attributes are simply the result of concatenating the ENTRY path, and the path of the relevant subpart, i.e.:

- "|<C_ENTRY.meaning>|<subpart-path>"

Examples include:

- "|ECG results|data|history|event|ECG_result|lead_3"
- "|blood pressure|data|history|offset|BP|systolic pressure"
- "|therapeutic_order|2nd course|action|generic name"

6.2.5 C_ENTRY_PROXY Class

CLASS	C_ENTRY_PROXY	
Purpose	Proxy for a new C_ENTRY archetype.	
Inherit	C_CONTENT_ITEM, ARCHETYPE_PROXY<C_ENTRY>	
Attributes	**Signature**	**Meaning**
Invariants		

END OF DOCUMENT