



REFERENCE MODEL

The *openEHR* Support Archetype Model

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

Revision: 0.9.3

Pages: 15

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

© 2003 The *openEHR* Foundation

The *openEHR* 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

© Copyright *openEHR* Foundation 2001 - 2003

All Rights Reserved

1. This document is protected by copyright and/or database right throughout the world and is owned by the *openEHR* Foundation.
2. You may read and print the document for private, non-commercial use.
3. You may use this document (in whole or in part) for the purposes of making presentations and education, so long as such purposes are non-commercial and are designed to comment on, further the goals of, or inform third parties about, *openEHR*.
4. You must not alter, modify, add to or delete anything from the document you use (except as is permitted in paragraphs 2 and 3 above).
5. You shall, in any use of this document, include an acknowledgement in the form:

"© Copyright *openEHR* Foundation 2001-2003. All rights reserved.
www.openEHR.org"

6. This document is being provided as a service to the academic community and on a non-commercial basis. Accordingly, to the fullest extent permitted under applicable law, the *openEHR* Foundation accepts no liability and offers no warranties in relation to the materials and documentation and their content.
7. If you wish to commercialise, license, sell, distribute, use or otherwise copy the materials and documents on this site other than as provided for in paragraphs 1 to 6 above, you must comply with the terms and conditions of the *openEHR* Free Commercial Use Licence, or enter into a separate written agreement with *openEHR* Foundation covering such activities. The terms and conditions of the *openEHR* Free Commercial Use Licence can be found at http://www.openehr.org/free_commercial_use.htm

Amendment Record

Issue	Details	Who	Completed
0.9.3	CR-000043. Move External package to Common RM and rename to Identification (incorporates CR-000036 - Add HIER_OBJECT_ID class, make OBJECT_ID class abstract.)	D Lloyd	09 Oct 2003
0.9.2	Fixed typos in C_SET, C_LIST.	B Fowler	01 Apr 2003
0.9.1	CR-000003, CR-000004 changes. Changed package naming, improved heading structures.	T Beale	20 Mar 2003
0.9	Initial writing. C_External package taken from Common Archetype Model. C_Support and C_Relationship packages from Data Types AM. Made C_REL_MULTIPLE<T> abstract. Formally validated using ISE Eiffel 5.2.	T Beale	25 Feb 2003

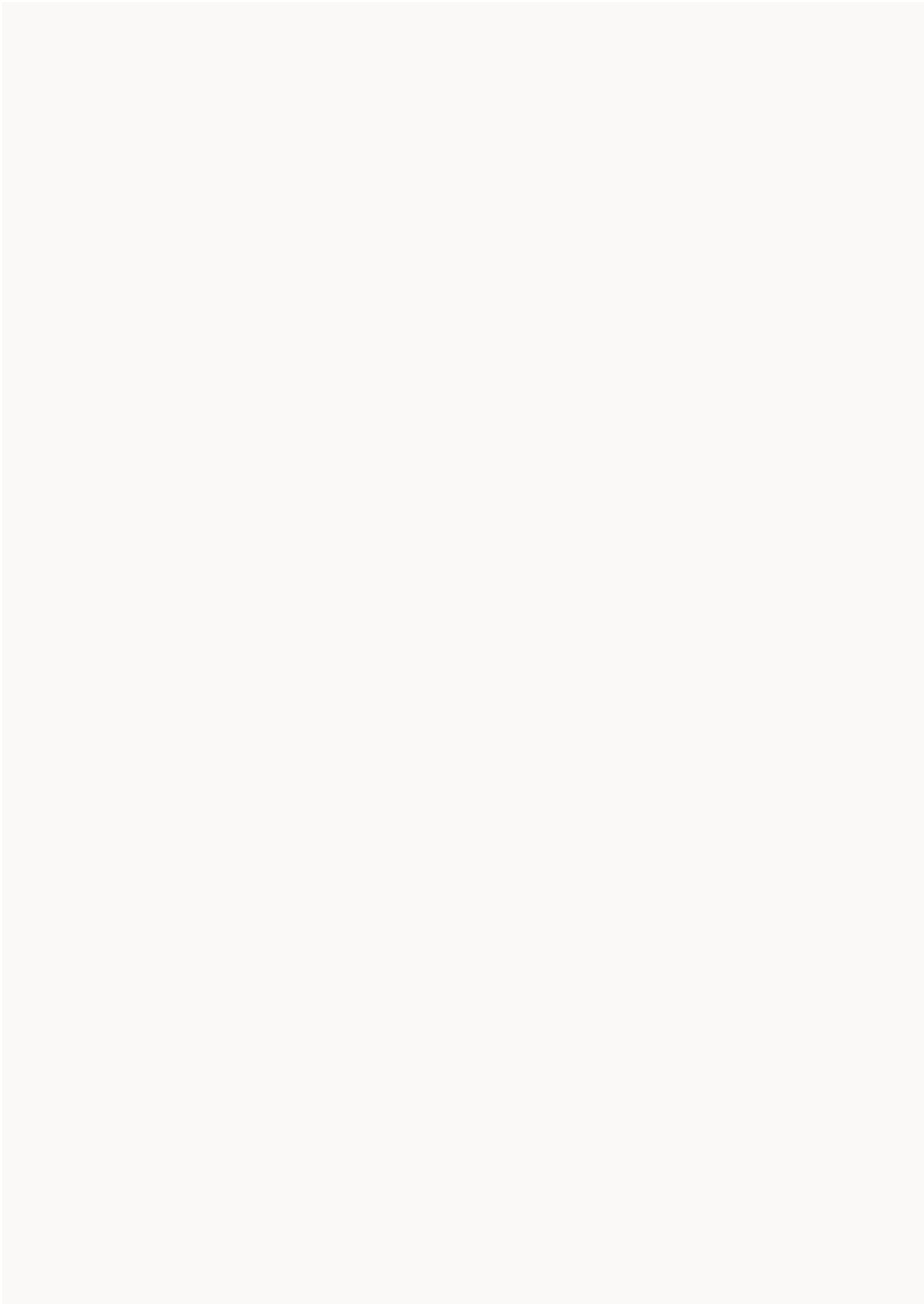
Acknowledgements

The work reported in this paper has been funded by a number of organisations, including The University College, London; Ocean Informatics Pty Ltd, Australia.



Table of Contents

1	Introduction.....	7
1.1	Purpose.....	7
1.2	Related Documents	7
1.3	Status.....	7
1.4	Peer review	7
2	Overview	9
3	AM.SUPPORT.BASIC Package	10
3.1	Overview.....	10
3.2	Class Descriptions.....	10
3.2.1	C_BOOLEAN Class.....	10
3.2.2	C_STRING Class.....	10
4	AM.SUPPORT.RELATIONSHIP Package	12
4.1	Introduction.....	12
4.2	Class Definitions.....	12
4.2.1	C_RELATIONSHIP Class.....	12
4.2.2	C_REL_SINGLE<T> Class	13
4.2.3	C_REL_MULTIPLE<T> Class	13
4.2.4	C_LIST<T> Class.....	14
4.2.5	C_SET<T> Class.....	14



1 Introduction

1.1 Purpose

This document describes the *openEHR* Support Archetype Model, which describes basic concepts used in archetype models. The intended audience includes:

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

1.2 Related Documents

Prerequisite documents for reading this document include:

- The *openEHR* Modelling Guide
- The *openEHR* Support Reference Model

1.3 Status

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

Currently the UML diagrams are hand-produced. Various tool versions exist (Rose, Objecteering), but the visual quality is still being improved; when this is complete, the tool-generated images will be used.

Also in the future, 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/doculist.htm>. New versions are announced on openehr-announce@openehr.org.

1.4 Peer review

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.



2 Overview

The Support Archetype Model comprises basic types which are used in the other *openEHR* archetype models. The package structure is illustrated in FIGURE 1.

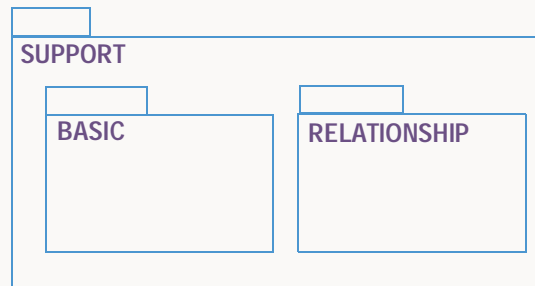


FIGURE 1 AM.SUPPORT Package

3AM.SUPPORT.BASIC Package

3.1Overview

The BASIC package is illustrated in FIGURE 2.

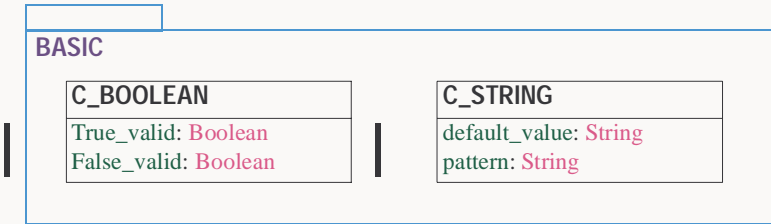


FIGURE 2 AM.SUPPORT.BASIC Package

3.2Class Descriptions

3.2.1C_BOOLEAN Class

CLASS	C_BOOLEAN	
Purpose	Constraint on instances of Boolean.	
Attributes	Signature	Meaning
	true_valid: Boolean	True if the value True is allowed
	false_valid: Boolean	True if the value False is allowed
Invariant	<i>Binary_consistency</i> : true_valid <i>or</i> false_valid <i>Default_value_consistency</i> : default_value.value <i>and</i> true_valid <i>or else not</i> default_value.value <i>and</i> false_valid	

3.2.2C_STRING Class

CLASS	C_STRING	
Purpose	Constraint on instances of STRING.	
Attributes	Signature	Meaning
	default_value: String	Default value for this String object. In theory this might be computable from the pattern, but there is no simple algorithm for generating a matching string from a pattern.
	pattern: String	Regular expression pattern for proposed instances of String to match.

CLASS	C_STRING
Invariant	<i>default_value_exists</i> : default_value /= Void <i>pattern_exists</i> : pattern /= Void and then not pattern.is_empty

4AM.SUPPORT.RELATIONSHIP Package

4.1 Introduction

The RELATIONSHIP package includes classes which express constraints on the relationships which occur in any reference model between archetype fragments. There are two kinds: single relationships - where an attribute of a class is of another (non-basic) class type, and multiple relationships such as attributes which are lists and sets of other types. The package is illustrated in FIGURE 3.

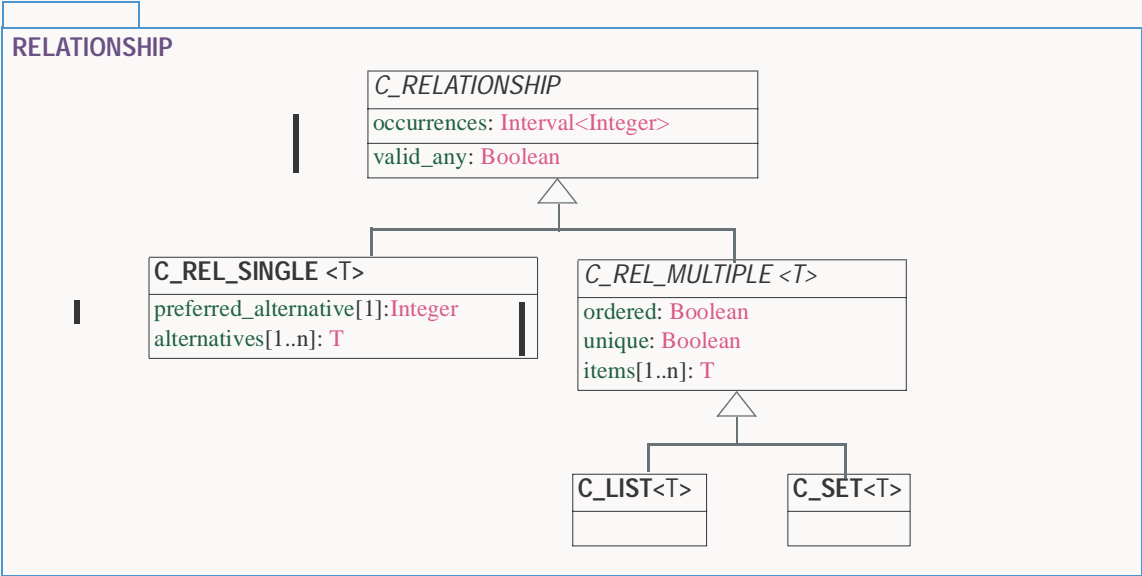


FIGURE 3 AM.SUPPORT.RELATIONSHIP Package

4.2 Class Definitions

4.2.1 C_RELATIONSHIP Class

CLASS	C_RELATIONSHIP (abstract)	
Purpose	Abstract parent of relationship constrainer types.	
Attributes	Signature	Meaning
	occurrences: Interval<Integer>	Constraint on the multiplicity of the frag- ment.
Abstract	Signature	Meaning
	valid_any: Boolean	True if anything is allowed as target of rela- tionship (i.e. within typing constraints of ref- erence model).
Invariant	occurrences_exists: occurrences /= Void and then occurrences.lower >= 0	

4.2.2 C_REL_SINGLE<T> Class

CLASS	C_REL_SINGLE <T>	
Purpose	Constrainer for single relationships.	
Inherit	C_RELATIONSHIP	
Attributes	Signature	Meaning
	alternatives: List<T>	Alternative items
	preferred_alternative: Integer	Preferred alternative.
Functions	Signature	Meaning
	valid_any: Boolean	True if no alternatives are supplied
Invariant	<i>occurrences:</i> occurrences.upper <= 1 <i>alternatives_valid:</i> (alternatives = Void and valid_any) or else alternatives /= Void and (not alternatives.empty and preferred_alternative > 0 and preferred_alternative <= alternatives.count)	

4.2.3 C_REL_MULTIPLE<T> Class

CLASS	C_REL_MULTIPLE <T> (abstract)	
Purpose	Constrainer for multiple relationships such as lists, sets etc.	
Inherit	C_RELATIONSHIP	
Attributes	Signature	Meaning
	is_ordered: Boolean	Indicates an ordered list
	is_unique: Boolean	Indicates unique membership
	c_items: List<T>	Items in list.
Functions	Signature	Meaning
	valid_any: Boolean	True if no items are supplied
	is_empty: Boolean	True if c_items is empty
Invariant	<i>valid_any:</i> c_items = Void implies valid_any <i>c_items_validity:</i> valid_any or else (c_items /= Void and then not c_items.empty)	

4.2.4 C_LIST<T> Class

CLASS	C_LIST<T>
Purpose	Constrainer for list relationships, in which multiple membership and ordering occur.
Inherit	C_REL_MULTIPLE<T>
Invariant	<i>Ordering</i> : ordered <i>Uniqueness</i> : not unique

4.2.5 C_SET<T> Class

CLASS	C_SET<T>
Purpose	Constrainer for set relationships, in which unique membership occurs.
Inherit	C_REL_MULTIPLE<T>
Invariant	<i>Ordering</i> : not ordered <i>Uniqueness</i> : unique

END OF DOCUMENT