REFERENCE MODEL

The *open*EHR Demographic Information Model

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

Revision: 2.0

Pages: 27

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

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

Copyright Notice

Amendment Record

Issue	Details	Who	Completed
RELEASE 1.0			
2.0	**CR-000189**. Add LOCATABLE.*parent*. New invariant in PARTY. **CR-000190**. Rename VERSION_REPOSITORY to VERSIONED_OBJECT. **CR-000194**: Correct anomalies with LOCATABLE.*uid*	S Heard T Beale H Frankel T Beale	25 Jan 2006
RELEASE 0.96			
RELEASE 0.95			
1.4.7	**CR-000133**. Remove details /= Void invariant from PARTY	R Chen	12 Mar 2005
1.4.6	**CR-000048**. Pre-release review of documents. Corrected STRUCTURE to be ITEM_STRUCTURE. Make ACTOR.*languages* a List not a Set.	D Lloyd	22 Feb 2005
1.4.5	**CR-000024**. Revert *meaning* to STRING and rename as *archetype_node_id*. **CR-000118**. Make package names lower case.	S Heard, T Beale T Beale	10 Jan 2005
RELEASE 0.9			
1.4.4	**CR-000041**. Visually differentiate primitive types in *open*EHR documents.	D Lloyd	04 Oct 2003
1.4.3	**CR-000013**. Rename key classes, according to CEN ENV 13606.	S Heard, D Kalra, T Beale	15 Sep 2003
1.4.2	**CR-000035**. Clarify circular relationships between PARTY and PARTY_REL.	Z Tun	14 Aug 2003
1.4.1	**CR-000003**. Removed ARCHETYPED and VERSIONABLE classes.	T Beale, Z Tun	18 Mar 2003
1.4	**Formally validated using ISE Eiffel 5.2**. Minor corrections to invariants.	T Beale	25 Feb 2003
1.3.1	**Review by H Frankel, MCA.** Corrections to diagrams and class texts. Improved PARTY_RELATIONSHIP semantics. Added Patient instance example. Made *time_validity* attributes optional.	T Beale	13 Feb 2003
1.3	Corrections to diagrams and class texts. Inheritance changed to ARCHETYPED for most key classes. Some instance examples added.	Z Tun, T Beale	08 Jan 2003
1.2	General modifications, addition of CAPABILITY class.	T Beale, D Lloyd	22 Oct 2002
1.1	Renamed CONTACT_DESCRIPTOR to CONTACT. Removed CONTACT.*role*. Renamed PARTY_ROLE to ROLE. Changed CONTACT.*address* to *addresses*. Renamed SPATIAL to STRUCTURE. Introduced PARTY and ACTOR classes.	T Beale	18 Sep 2002
1.0	Created from EHR RM.	T Beale	28 Aug 2002

Acknowledgements

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.

Table of Contents

1 Introduction

1.1 Purpose

This document describes the architecture of the *open*EHR Demographic Information Model. The semantics are drawn from previous work in GEHR, existing models in CEN 13606 and the HL7v3 RIM, and other work done in Australia.

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;
- Medical informaticians and clinicians intersted in health information;
- Health data managers.

1.2 Related Documents

Prerequisite documents for reading this document include:

- The *open*EHR Modelling Guide
- The *open*EHR Support Information Model
- The *open*EHR Data Types Information Model
- The *open*EHR Common Information Model

Other documents describing related models, include:

- The *open*EHR EHR Information Model
- The *open*EHR Demographic Model

1.3 Status

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

This document is available at http://svn.openehr.org/specification/TAGS/Release-1.0/publishing/architecture/rm/demographic_im.pdf.

The latest version of this document can be found at http://svn.openehr.org/specification/TRUNK/publishing/architecture/rm/demographic_im.pdf.

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 provided on the mailing list openehr-technical@openehr.org, or by private email.

1.5 Conformance

Conformance of a data or software artifact to an *open*EHR Reference Model specification is determined by a formal test of that artifact against the relevant *open*EHR Implementation Technology Specification(s) (ITSs), such as an IDL interface or an XML-schema. Since ITSs are formal, automated derivations from the Reference Model, ITS conformance indicates RM conformance.

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

Corbamed,

PIDS

Archetypes

3 Demographic Package

3.1 Overview

The demographic model illustrated in FIGURE 1 is a generalised model of the facts one might expect to see in a demographic server. The purpose of the model is as a specification of a demographic service, either standalone, or a "wrapper" service for an existing system such as a patient master index (PMI). In the latter situation, it is used to add the required *open*EHR semantics, particularly versioning, to an existing service.

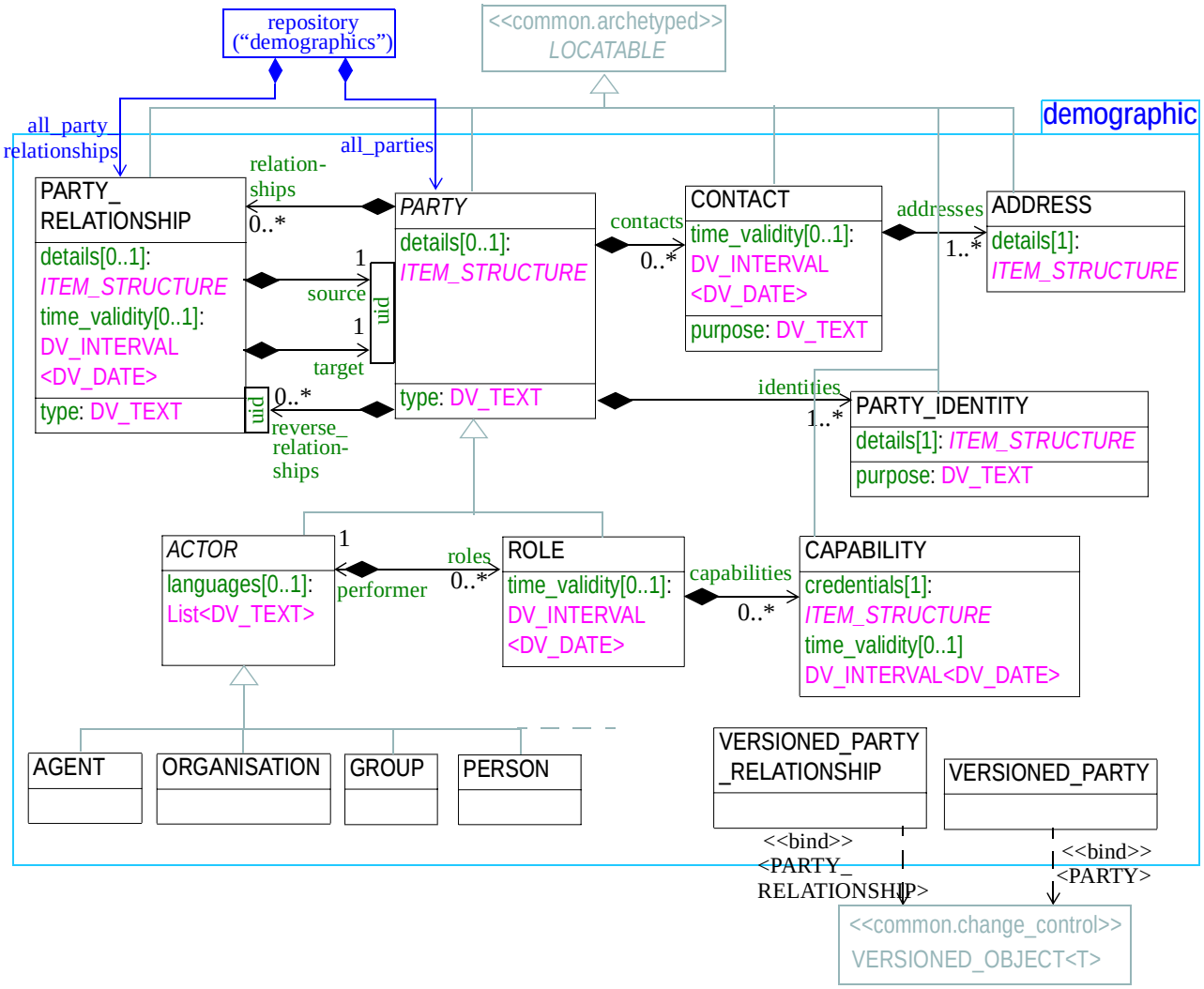

FIGURE 1 rm.demographic Package

The general design is based on the scheme of party and accountability described by Fowler [19], and is influenced by clinical adaptations including work done in Australia [11] and the HL7v3 RIM [15] (itself influenced by the Fowler models).

One of the main design criteria of the model is that it expresses attributes and relationships of demographic entities which exist *regardless* of particular clinical involvements or participations in particular events. Participations are meaningful only within the context of the health record or other relevant model where they record context-specific relationships between demographic entities and events in the real world.

Another criterion is that instances of the classes in the model must be serialisable into an EHR Extract in an unambgiuous way. This requires that each PARTY be a self-contained hierarchy of data, in the same way as distinct COMPOSITIONs in the EHR model are distinct hierarchies in an Extract. In order to ensure this condition, PARTY_RELATIONSHIPs must be implemented correctly, so as to prevent endless traversal of all PARTY objects through their relationships, when serialising. See Party Relationships below for details.

3.1.1 Archetyping

The model is designed to be used with archetypes, hence the generic nature of all entities. Every class containing an attribute of the form *details*:STRUCTURE is a completely archetypable structure. As a result, archetypes can be defined for concepts such as particular kinds of PERSON, ORGANISATION; for actual ROLEs such as “health care practitioner”, and for party identities and addresses.

3.1.2 Names and Addresses

Classes have been included for PARTY_IDENTITY and ADDRESS, even though they contain only a link to details, in the form of the generic STRUCTURE class. This is not strictly necessary - it could have been done simply using appropriately named attributes in the classes PARTY and CONTACT - but is done to provide a place to add specific semantics in future releases of the model. It is also expected to make software development easier, since it provides explicit classes to which behaviour and other implementation attributes can be added. Lastly, it allows the notions of PARTY_IDENTITY and ADDRESS to be explicitly used in archetype-authoring tools.

Instances of PARTY_IDENTITY, linked to PARTY by the attribute *identities* are intended to express the names of people, organisations, and other actors - that is names which are “owned” by the party, e.g. self-declared (in the case of institutions and companies) or by virtue of social relations (names given by parents, tribes etc). Identifiers of Parties given by other organisations, or the state are not represented in this way, and should be recorded in the PARTY.*details* structure instead (see below).

3.1.3 Unique Identification

Identifiers of Parties given by organisations or the state are treated as any other attribute of a Party, i.e. recorded as part of the data in the PARTY.*details* structure.

To Be Continued: id used by the demographic system itself - probably an OID inherited from LOCATABLE

3.1.4 Party Relationships

Relationships between parties in the real world may be expressed using PARTY_RELATIONSHIP objects, as illustrated in FIGURE 2.

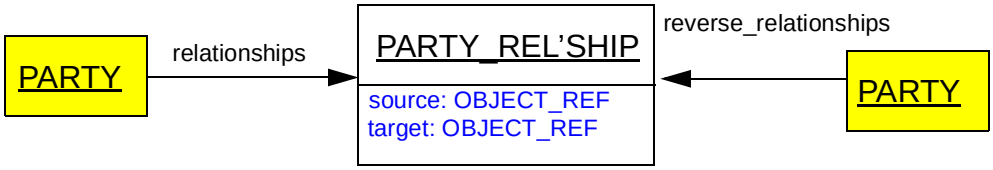

FIGURE 2 General Relationship Model

Relationships are considered *directional*, hence the use of the attribute names *source* and *target*, however, these names are otherwise neutral, and give no indication as to the meaning of the relationships, such as which party is responsible and which accountable (for comparison, see the demographic mod-

els of Fowler [19]). Accordingly, each party involved in a relationship includes it in its *relationships* list, if it is at the source end, or in the *reverse_relationships* list, if at the target end.

The usual way to determine which ends of a relationship the two parties should go is usually by which party's actions caused the relationship to come into being. For example, a relationship representing the concept "patient", between a health consumer and a health care organisation would have the consumer as source and the organisation as target.

Relationships are also *shared objects* whose existence is dependent upon both source and target PARTYs being present. There is no official way to show this in UML, but it is approximated by using white diamonds from the PARTY to the PARTY_RELATIONSHIP class. The semantics are such that if either party is deleted, the relationship must also be deleted.

Serialising PARTY_RELATIONSHIPs requires that they function like LINKs in the EHR model, i.e. express symbolic references, not physical ones.

To Be Continued:

3.1.5 Versioning Semantics

The class PARTY and its descendants ACTOR and ROLE are all potentially versioned in a demographic system. A version of a PARTY takes in all the compositional parts, such as identities, contacts etc.

To Be Continued: versioning semantics of relationships

3.2 Class Definitions

3.2.1 PARTY Class

CLASS	*PARTY (abstract)*	
Purpose	Ancestor of all party types, including real world entities and their roles. A party is any entity which can participate in an activity. The *name* attribute inherited from LOCATABLE is used to indicate the actual type of party (note that the actual names, i.e. identities of parties are indicated in the *identities* attribute, not the *name* attribute).	
CEN	healthcare agent	
HL7	Entity	
Inherit	LOCATABLE	
Attributes	**Signature**	**Meaning**
	identities: Set<PARTY_IDENTITY>	Identities used by the party to identify itself, such as legal name, stage names, aliases, nicknames and so on.
	contacts: Set<CONTACT>	Contacts for this party.
	relationships: Set<PARTY_RELATIONSHIP>	Relationships in which this role takes part as source.

CLASS	*PARTY (abstract)*	
	reverse_relationships: `Set<OBJECT_REF>`	Relationships in which this role takes part as target.
	details: `ITEM_STRUCTURE`	All other details for this party.
Functions	**Signature**	**Meaning**
	type: `DV_TEXT`	Type of party, such as “PERSON”, “ORGANISATION”, etc. Role name, e.g. “general practitioner”, “nurse”, “private citizen”. Taken from inherited *name* attribute.
Invariants	***Uid_exists***: uid /= Void ***Type_valid***: type = name ***Identities_valid***: identities /= Void ***and then not*** identities.empty ***Contacts_valid***: contacts /= Void ***implies not*** contacts.empty ***Relationships_validity***: relationships /= Void implies (***not*** relationships.empty ***and then*** relationships.for_all({PARTY_RELATIONSHIP}.source = Current)) ***Reverse_relationships_validity***: reverse_relationships /= Void implies (***not*** reverse_relationships.empty ***and then*** reverse_relationships.for_all(repositories(“demographics”).all_party_relationships.has_object(item) **and then** repositories(“demographics”).all_party_relationships.object(item).target = Current)) ***Is_archetype_root***: is_archetype_root ***No_parent***: parent = Void	

3.2.2 PARTY_IDENTITY Class

CLASS	PARTY_IDENTITY	
Purpose	An identity “owned” by a PARTY, such as a person name or company name, and which is used by the party to identify itself. Actual structure is archetyped.	
CEN	`Person Name` (data type).	
HL7	`Person Name (PN)` (data type).	
Inherit	`LOCATABLE`	
Attributes	**Signature**	**Meaning**
	details: `ITEM_STRUCTURE`	The value of the indentity. This will often taken the form of a parsable string or a small structure of strings.
Functions	**Signature**	**Meaning**
	purpose: `DV_TEXT`	Purpose of identity, e.g. “legal”, “stagename”, “nickname”, “tribal name”, “trading name”. Taken from value of inherited *name* attribute.
	as_string: `String`	Indentity in the form of a single string.
Invariants	***Purpose_valid***: purpose = name ***Details_exists***: details /= Void	

TBD_1: may define DV_RWE_ID type for use here e.g. as LIST_S<DV_RWE_ID>

3.2.3 CONTACT Class

CLASS	CONTACT	
Purpose	Description of a means of contact of a party. Actual structure is archetyped.	
Inherit	`LOCATABLE`	
Attributes	**Signature**	**Meaning**
	time_validity: `DV_INTERVAL<DV_DATE>`	Valid time interval for this contact descriptor.
	addresses: `List<ADDRESS>`	A set of address alternatives for this purpose and time validity.
Functions	**Signature**	**Meaning**

CLASS	CONTACT	
	purpose: DV_TEXT	Purpose for which this contact is used, e.g. “mail”, “daytime phone”, etc. Taken from value of inherited *name* attribute.
Invariants	***Purpose_valid***: purpose = name ***Addresses_exists***: addresses /= Void ***and then not*** addresses.empty	

3.2.4 ADDRESS Class

CLASS	ADDRESS	
Purpose	Address of contact, which may be electronic or geographic.	
CEN	Address (data type)	
HL7	Address (AD) (data type)	
Inherit	LOCATABLE	
Attributes	**Signature**	**Meaning**
0..1	**details**: ITEM_STRUCTURE	The details of the address, in the form of a STRUCTURE. This may take the form of a SINGLE_S, whose data item is a parsable string or a list or tree of many parts.
Functions	**Signature**	**Meaning**
	type: DV_TEXT	Type of address, e.g. “electronic”, “locality”. Taken from value of inherited *name* attribute.
	as_string: String	Address in the form of a single string.
Invariants	***Type_valid***: type = name ***Details_exists***: details /= Void	

3.2.5 ACTOR Class

CLASS	*ACTOR (abstract)*
Purpose	Ancestor of all real-world types, including people and organisations. An actor is any real-world entity capable of taking on a role.
CEN	healthcare party
HL7	Entity

CLASS	*ACTOR (abstract)*	
Inherit	PARTY	
Attributes	**Signature**	**Meaning**
	roles: Set<ROLE>	Roles played by this party.
	languages: List<DV_TEXT>	Languages which can be used to communicate with this actor, in preferred order of use (if known, else order irrelevant).
Functions	**Signature**	**Meaning**
	has_legal_identity: Boolean	True if one there is an identity with purpose "legal identity"
Invariants	***Roles_valid***: roles /= Void ***implies not*** roles.empty ***Languages_valid***: languages /= Void ***implies not*** languages.is_empty ***Legal_identity_exists***: has_legal_identity	

3.2.6 PERSON Class

CLASS	PERSON	
Purpose	Generic description of persons. Provides a dedicated type to which Person archetypes can be targeted.	
CEN	healthcare person	
GEHR	G1_PERSON	
HL7	Person	
Inherit	ACTOR	
Attributes	**Signature**	**Meaning**
Invariants		

3.2.7 ORGANISATION Class

CLASS	ORGANISATION
Purpose	Generic description of organisations. An organisation is a legally constituted body whose existence (in general) outlives the existence of parties considered to be part of it.

CLASS	ORGANISATION	
CEN	healthcare organisation	
GEHR	G1_HCF	
HL7	ORGANIZATION	
Inherit	ACTOR	
Attributes	Signature	Meaning
Invariants		

3.2.8 GROUP Class

CLASS	GROUP	
Purpose	A group is a real world group of parties which is created by another party, usually an organisation, for some specific purpose. A typical clinical example is that of the specialist care team, e.g. “cardiology team”. The members of the group usually work together.	
Inherit	ACTOR	
Attributes	Signature	Meaning
Invariants		

3.2.9 AGENT Class

CLASS	AGENT	
Purpose	Generic concept of any kind of agent, including devices, software systems, but not humans or organisations.	
CEN	healthcare software, healthcare device	
HL7	DEVICE	
Inherit	ACTOR	
Attributes	Signature	Meaning

CLASS	AGENT
Invariants	

3.2.10 ROLE Class

CLASS	ROLE	
Purpose	Generic description of a role performed by an actor. The role corresponds to a competency of the party. Roles are used to define the responsibilities undertaken by a party for a purpose. Roles should have credentials qualifying the performer to perform the role.	
Use	Roles correspond to concepts like “general practitioner”, “nurse” and so on.	
CEN	`healthcare agent in context`	
HL7	`ROLE`	
Inherit	`PARTY`	
Attributes	Signature	Meaning
	capabilities: `List <CAPABILITY>`	The capabilities of this role.
	time_validity: `DV_INTERVAL <DV_DATE>`	Valid time interval for this role.
	performer: `ACTOR`	Actor playing the role.
Invariants	***Capabilities_valid***: capabilities /= Void ***implies not*** capabilities.empty ***Performer_exists***: performer /= Void	

3.2.11 CAPABILITY Class

CLASS	CAPABILITY	
Purpose	Capability of a role, such as “ehr modifier”, “health care provider”. Capability should be backed up by credentials.	
Use		
Inherit	`LOCATABLE`	
Attributes	Signature	Meaning

CLASS	CAPABILITY	
	credentials: ITEM_STRUCTURE	The qualifications of the performer of the role for this capability. This might include professional qualifications and official identifications such as provider numbers etc.
	time_validity: DV_INTERVAL <DV_DATE>	Valid time interval for the credentials of this capability.
Invariants	***Credentials_exists***: credentials /= Void	

To Be Continued: to be considered - accreditation, credentialling

3.2.12 PARTY_RELATIONSHIP Class

CLASS	PARTY_RELATIONSHIP	
Purpose	Generic description of a relationship between parties.	
HL7	RELATIONSHIP_LINK	
Inherit	LOCATABLE	
Attributes	**Signature**	**Meaning**
	details: ITEM_STRUCTURE	The detailed description of the relationship
	time_validity: DV_INTERVAL <DV_DATE>	Valid time interval for this relationship.
	source: OBJECT_REF	Source of relationship.
	target: OBJECT_REF	Target of relationship.
Functions	**Signature**	**Meaning**
	type: DV_TEXT	Type of relationship, such as “employment”, “authority”, “health provision”
Invariants	***Uid_exists***: uid /= Void ***Type_validity***: type = name ***Source_valid***: source /= Void ***and then*** source.relationships.has(Current) ***Target_valid***: target /= Void ***and then not*** target.reverse_relationships.has(Current)	

3.3 Instance Examples

In the following instance examples, the values of the attributes *uid*, *source*, *target*, and *reverse_relationships* are not meant to be taken as literally valid OBJECT_IDs - for the purposes of clarity, simple integers have been used.

3.3.1 Parties

3.3.1.1 Person

FIGURE 3 illustrates a possible set of instances for a PERSON, with home and work contact information. There are separate archetypes for the PERSON, each ADDRESS, and each PARTY_IDENTITY. In the following figure, "meaning" is the meaning from the value of the *archetype_node_id* attribute, functionally derived from the archetype local ontology.

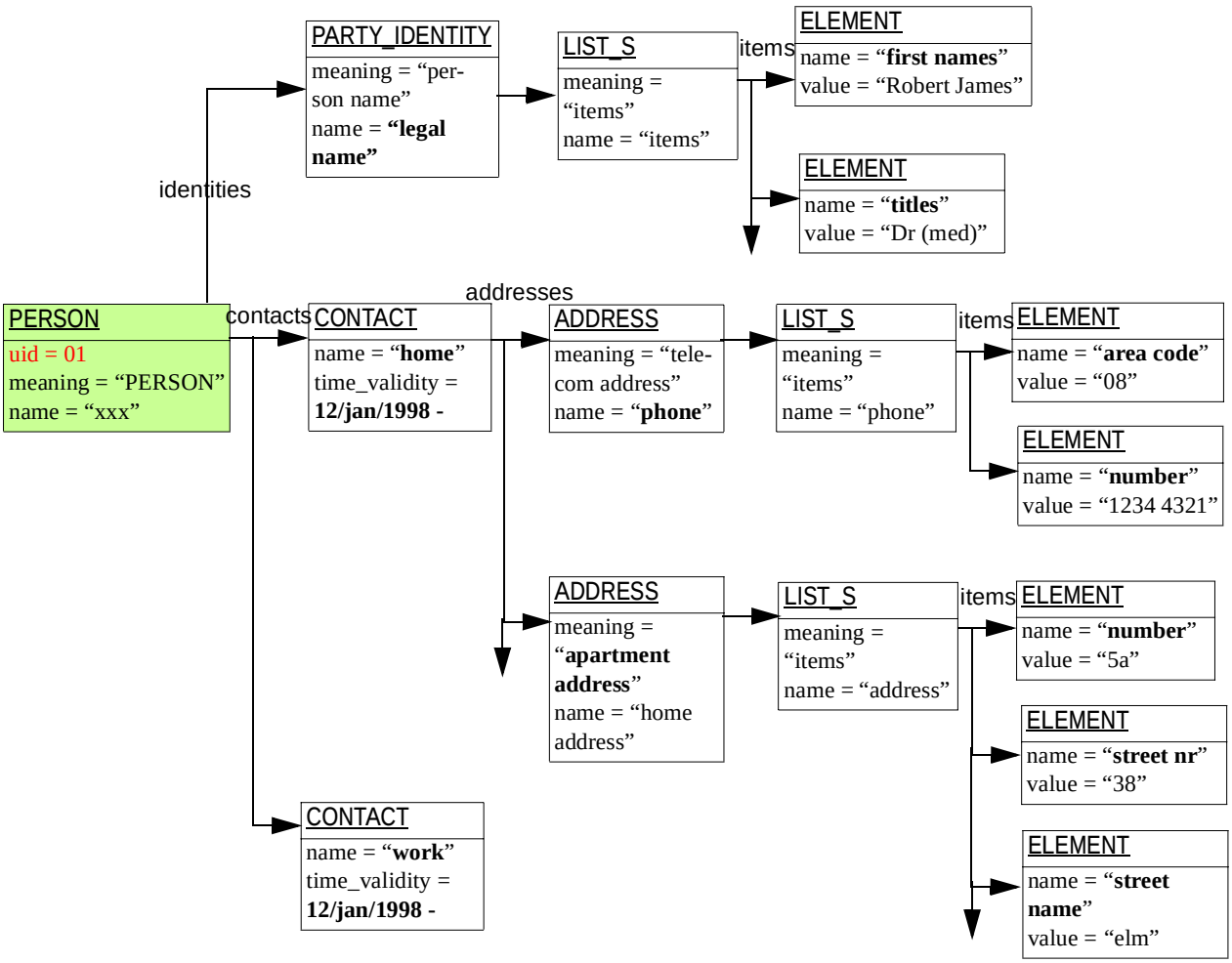

FIGURE 3 Person Demographic Information

3.3.1.2 Clinician

Credentials

3.3.1.3 Health Care Facility

3.3.1.4 Group

FIGURE 4 illustrates the demographic information for a cadiac surgery team in a hospital. The group includes a head surgeon, anaesthetist, assistant surgeon, and presumably others (not shown). Each of

these members of the team have an employment relationship with the hospital (shown only for one surgeon, in the interests of clarity).

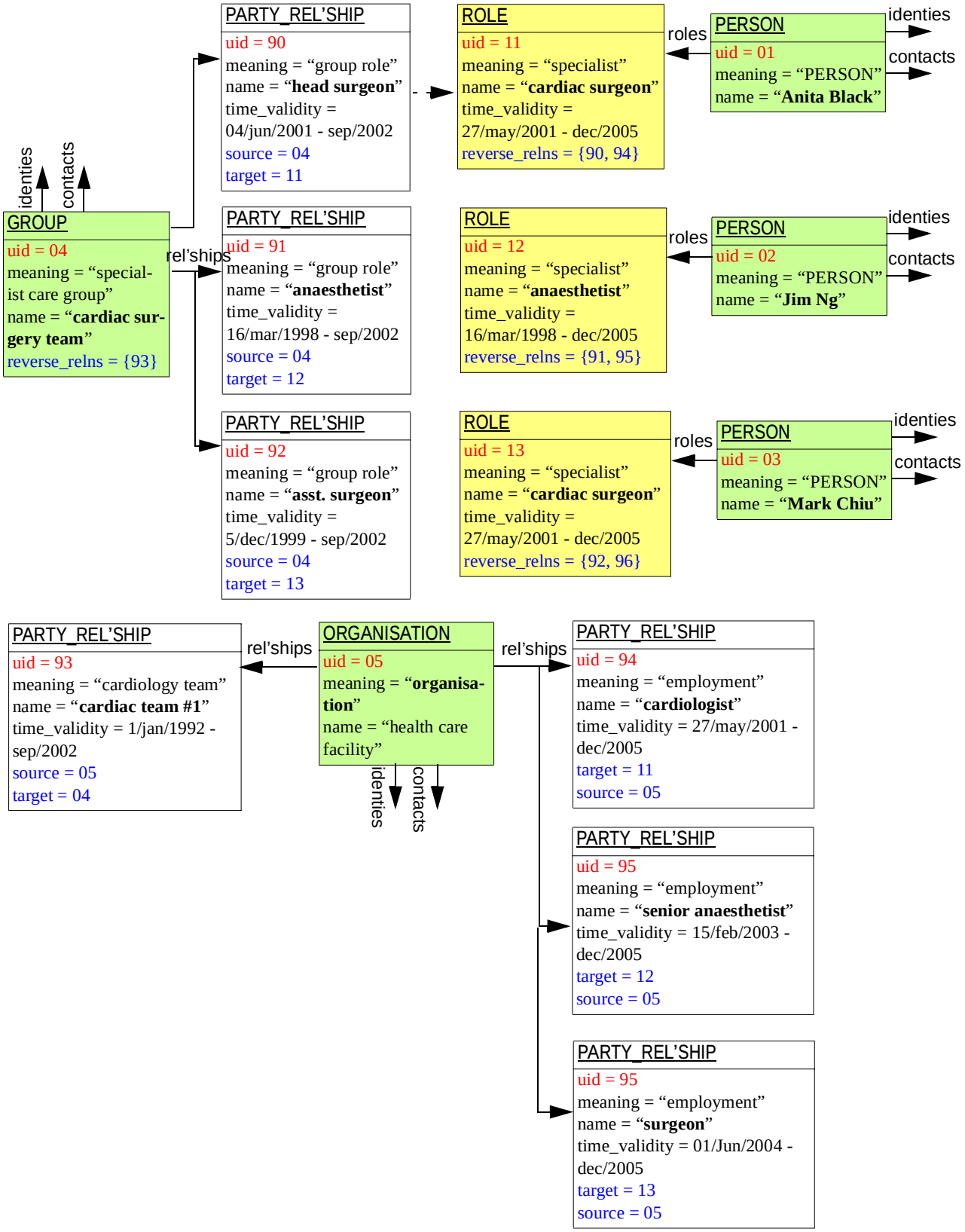

FIGURE 4 Group Demographics

3.3.2 Relationships

3.3.2.1 Familial Relationship

3.3.2.2 Employment Relationship

3.3.2.3 Patient

FIGURE 5 shows a simple way of representing the patient relationship between a person and a health care organisation.

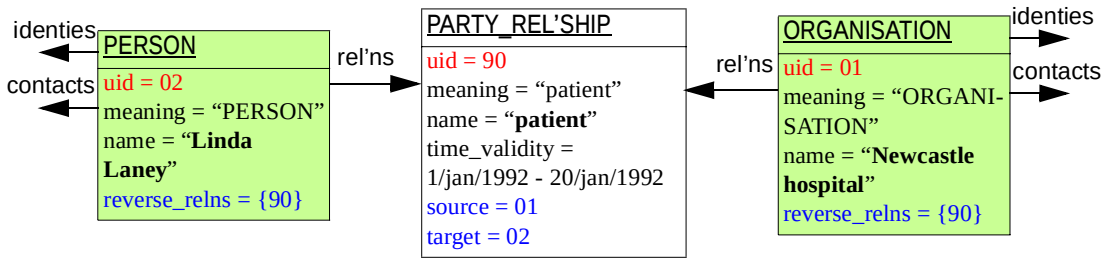

FIGURE 5 Simple Patient Relationship

FIGURE 6 shows the same logical relationship, but with both Parties acting through Roles, representing their status as healthcare consumer and healthcare provider respectively. Each of these Roles has associated credentials which document its official nature within the healthcare system.

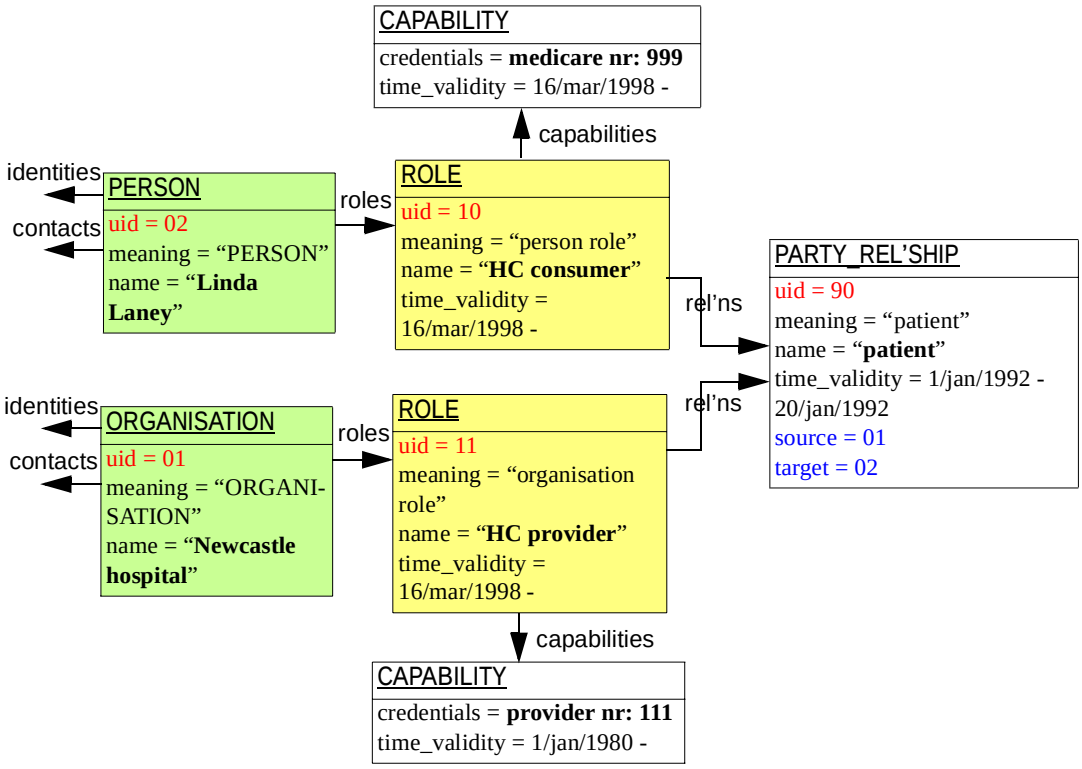

FIGURE 6 Patient Relationship with Roles and Credentials

A References

A.1 General

1 Beale T. *Archetypes: Constraint-based Domain Models for Future-proof Information Systems*. See http://www.deepthought.com.au/it/archetypes.html.

2 Beale T *et al*. *Design Principles for the EHR*. See http://www.deepthought.com.au/openEHR.

3 Gray J, Reuter A. *Transaction Processing Concepts and Techniques*. Morgan Kaufmann 1993.

4 Sowa J F. *Knowledge Representation: Logical, philosophical and Computational Foundations*. 2000, Brooks/Cole, California.

5 Schloeffel P. (Editor). Requirements for an Electronic Health Record Reference Architecture. International Standards Organisation, Australia; Feb 2002; ISO TC 215/SC N; ISO/WD 18308.

6 Sottile P.A., Ferrara F.M., Grimson W., Kalra D., and Scherrer J.R. The holistic healthcare information system. Toward an Electronic Health Record Europe 1999. Nov 1999; 259-266.

A.2 CEN

7 ENV 13606-1 - *Electronic healthcare record communication - Part 1: Extended architecture*. CEN/ TC 251 Health Informatics Technical Committee.

8 ENV 13606-2 - *Electronic healthcare record communication - Part 2: Domain term list*. CEN/ TC 251 Health Informatics Technical Committee.

9 ENV 13606-3 - *Electronic healthcare record communication - Part 3: Distribution rules*. CEN/ TC 251 Health Informatics Technical Committee.

10 ENV 13606-4 - *Electronic Healthcare Record Communication standard Part 4: Messages for the exchange of information*. CEN/ TC 251 Health Informatics Technical Committee.

A.3 GEHR Australia

11 Beale T. *Northern Territory Health Demographic Model*.

12 Heard S. *GEHR Project Australia, GPCG Trial*. Available at http://www.gehr.org/gpcg/ehra.htm.

13 Beale T, Heard S. *GEHR Technical Requirements*. See http://www.gehr.org/technical/requirements/gehr_requirements.html.

A.4 OMG

14 CORBAmed document: *Person Identification Service*. (March 1999).
(Authors?)

A.5 HL7

15 HL7 version 3 2nd Ballot specification. Available at http://www.hl7.org.

A.6 Software Engineering

16 Meyer B. *Object-oriented Software Construction*, 2nd Ed.
Prentice Hall 1997

17 Walden K, Nerson J. *Seamless Object-oriented Software Architecture*.
Prentice Hall 1994

18 Gamma E, Helm R, Johnson R, Vlissides J. *Design patterns of Reusable Object-oriented Software*
Addison-Wesley 1995

19 Fowler M. *Analysis Patterns: Reusable Object Models*
Addison Wesley 1997

20 Fowler M, Scott K. *UML Distilled (2nd Ed.)*
Addison Wesley Longman 2000

21 Booch G, Rumbaugh J, Jacobsen I. *The Unified Modelling Language User Guide*. Addison esley 1999.

A.7 Resources

22 Arden Syntax. http://www.cpmc.columbia.edu/arden/

23 Asbru / The Asgaard Project. http://smi-web.stanford.edu/projects/asgaard/

24 Digital Imaging ad Communications in Medicine (DICOM). http://medical.nema.org/dicom.html.

25 EON ref required

26 GLIF (Guideline Interchange Format). http://www.glif.org/.

27 IANA - http://www.iana.org/.

28 ProForma language for decision support. http://www.acl.icnet.uk/lab/proforma.html.

29 SynEx project, UCL. http://www.chime.ucl.ac.uk/HealthI/SynEx/.

END OF DOCUMENT