Model-based Assessment of Data Availability in Health Information Systems

A. Winter, A. Strübing
Institute of Medical Informatics, Statistics and Epidemiology, University of Leipzig, Leipzig, Germany

Summary

Objectives: To introduce a formal definition of data availability as a contribution to trustworthiness of health information systems and to automatically detect respective weaknesses and propose solutions.
Method: Specifying an ontology, based on enterprise functions and application systems of (health) information systems and closely linked to the Three-Layer Graph-Based Meta Model (3LGM²). Deriving appropriate measures and algorithms.
Results: A formal definition for data availability is introduced and elucidated by an example. This concept is used e.g. to disclose missing communication links and to suggest solutions.
Conclusions: Data availability is a necessary but not sufficient condition for trustworthiness of health information systems. If information management has a thorough description of the information system at its disposal, e.g. by using 3LGM², the calculation of data availability does not need further efforts.

Keywords

Health information systems, hospital information systems, information management, architectural models, quality, data availability, trustworthiness

Methods Inf Med 2008; 47: 417–424

doi:10.3414/ME9123

1. Introduction

Since patient care is not only a health but also a life and death issue [1], patients have to trust on caregivers and both patients and caregivers depend on the trustworthiness of the information systems used. Trustworthiness covers aspects "ranging from privacy and security over systems' availability to the reliability of health data for care and research" [2]. For ensuring availability especially of healthcare information systems, we must consider their complexity and heterogeneity arising from bringing together more or less autonomous application systems in one information system [3]. Despite the engaged discussion between 'best of breed' solutions and 'holistic' or 'integrated' approaches ([4], p 277) this situation will remain, especially for regional health information systems.

These application systems support different (sets of) tasks such as clinical admission, radiotherapy, or care planning. As a prerequisite for performing a task, certain information is needed. For example, radiotherapy at least needs name, birthday and an identification number of a patient, which should have been the result of clinical admission. If clinical admission is supported by an ADT system (application system for admission, discharge and transfer of patients) and radiotherapy is supported by a RIS (radiological information system) the covering information system has to have means for transporting data representing name, birthday and identification number from the ADT system to the RIS. The more complex the information system, the more difficult it is to check it for having these means.

Hence, the availability of a whole health information system or parts of it highly depends on the availability of data representing information about patients as well as medical knowledge. This paper focuses on this aspect of trustworthiness and therefore deals with the following questions:

- How can information systems in health care be checked for their ability to provide all of its application systems with the data needed to support the tasks they have to support?
- How can managers of information systems benefit from such checks?

We propose a model-based approach to give an answer to these questions. Therefore we first introduce an ontology to support modeling information systems based on the Three-Layer Graph-Based Meta Model (3LGM²) for modeling health care information systems [3, 5]. In Chapter 3 we apply methods also used in compiler theory to define data availability as a logical predicate operating on 3LGM²-based models. Chapter 4 will demonstrate how this predicate can be used within information management not only to disclose problems but also to suggest suitable solutions.

2. Ontology for Modeling Information Systems in Health Care

Analyzing the ability of an information system to provide application systems with the data they need, first requires an ontology. This ontology has to contain precise descriptions of the relevant parts of an information system. We use 3LGM², which, similar to [6], bases especially upon the concepts of enterprise functions and application systems.

An enterprise function can be regarded as a directive for human or machine action.

Table 1 $ACC = \left(acc_{p,q}\right)_{p=1..6,q=1..6}$

$acc_{p,q}$		Entity types q = 1, …, 6					
		1	2	3	4	5	6
Enterprise functions p = 1, …, 6	1	0	0	u	0	0	0
	2	0	i	i	i	0	0
	3	u	0	0	0	i	i
	4	0	u	0	u	0	i
	5	0	0	i	0	u	0
	6	0	0	i	0	i	u

Table 2 $SUP := \left(sup_{p,n}\right)_{p=1..6,n=1..8}$

$sup_{p,n}$		Application systems n = 1, …, 8							
		1	2	3	4	5	6	7	8
Enterprise functions p = 1, …, 6	1	0	0	0	0	0	0	1	0
	2	0	0	0	0	0	0	1	0
	3	1	0	0	1	0	0	0	0
	4	1	0	0	0	0	0	0	1
	5	0	0	0	0	1	0	0	0
	6	0	0	0	0	0	0	0	1

The tasks mentioned in the introduction (e.g. "clinical admission") are examples for enterprise functions. Within the computer-supported part of an information system, we call the tools used to support the execution of enterprise functions (computer-based) application systems. Application systems may also store data. Interfaces enable communication among application systems.

Let $\underline{EF}$: = {ef_1, ..., ef_P} be a set of enterprise functions, $\underline{AS}$: = {as_1, ..., as_N} be a set of application systems, P >0, N >0. Let $SUP := \left(sup_{p,n}\right)_{p=1...P,n=1...N}$ be a two-dimensional matrix describing the relationship between tasks and tools mentioned above, such that $sup_{p,n} \in \{0,1\}$ and $sup_{p,n} := 1$ if the function ef_p is supported by the application system as_n and $sup_{p,n} := 0$ else.

Executing enterprise functions needs information of a certain type about physical or virtual things of the hospital. These types of information are represented by entity types. Data representing information about particular entities of the respective entity type are stored in application systems. The access of an enterprise function to an entity type can be in an interpreting or an updating manner.

Let $\underline{ET}$: = {et_1, ..., et_Q} be a set of entity types Q >0 and $ACC := \left(acc_{p,q}\right)_{p=1..P,q=1..Q}$ be a two-dimensional matrix describing how enterprise functions access entity types. It shall hold that $acc_{p,q} \in \{0, i, u, iu\}$ and $acc_{p,q} := 0$ if enterprise function ef_p neither interprets nor updates entity type et_q, $acc_{p,q} := i$ if enterprise function ef_p interprets entity type et_q, $acc_{p,q} := u$ if enterprise function ef_p updates entity type et_q, and $acc_{p,q} := iu$ if enterprise function ef_p updates and interprets entity type et_q.

In health care information systems, data is usually stored redundantly in different application systems; for every entity type (e.g. "Patient identification and case (ADT information)") a particular application system (e.g. ADT-system) is defined as master system. Only the master system creates or updates data of the respective type. If a respective data set has been created or updated, a corresponding HL7 message [7] is sent to all those application systems storing this data redundantly. We can generalize this concept of master systems for any entity type.

The prerequisite are appropriate communication links between these systems. We can describe the communication links by a communication matrix for each entity type

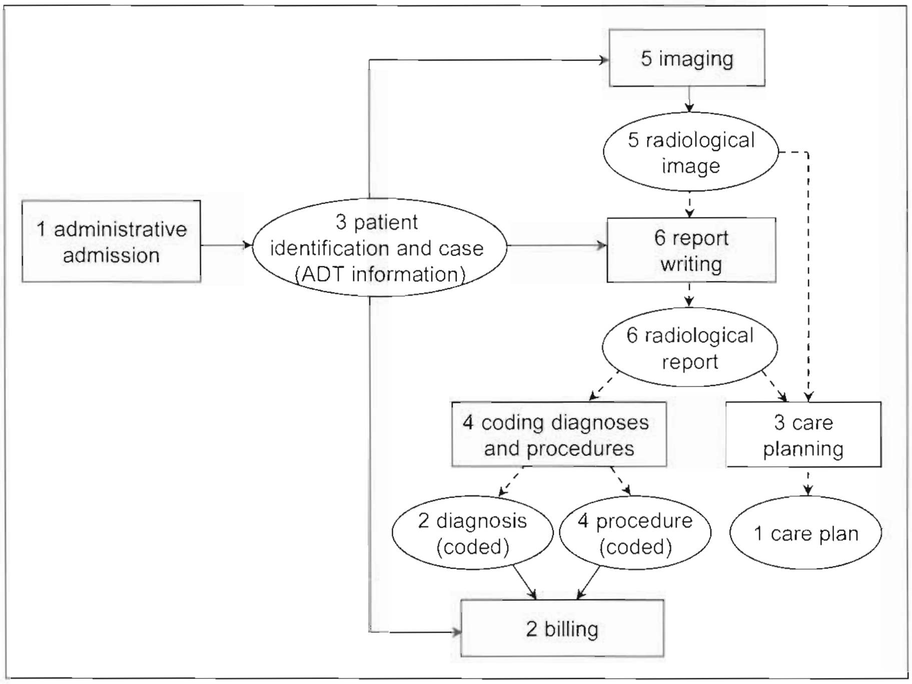

Fig. 1 Enterprise functions and the accessed entity types in a model of an information system as described before in this example. Entity types are depicted as ovals, enterprise functions as rectangles, interpretation as an arrow pointing to a function, update as an arrow pointing to an entity type. The meaning of dashed arrows is explained within the text.

$et_q \in ET$. Let $R_q := \left(r^q_{n,m}\right)_{n=1...N,m=1..N}$ be a communication matrix for et_q. We define $r^q_{n,m} := 1$ if as_n can send data representing information about et_q to as_m, and $r^q_{n,m} := 0$ else. Each communication matrix R_q is an adjacency matrix of a directed, labeled graph of application systems and communication links being able to exchange data about et_q.

We assume that once data reaches an application system it can be used within the application system for storing or processing deliberately as needed.

To model data storage we use a matrix $STORE := \left(store_{q,n}\right)_{q=1...Q,n=1...N}$, $store_{q,n} \in \{0, s, m\}$. For data concerning entity type et_q and an application system as_n we define $store_{q,n} := 0$ if the data is not stored in as_n, $store_{q,n} := s$ if the data is stored in as_n and $store_{q,n} := m$ if as_n is master for et_q.

Table 3 $STORE := \left(store_{q,n}\right)_{q=1..6,n=1..8}$

$store_{q,n}$		Application systems m = 1, …, 8							
		1	2	3	4	5	6	7	8
Entity types $q = 1, \ldots, 6$	1	m	0	0	0	0	0	0	0
	2	0	0	0	0	0	0	m	s
	3	0	0	0	0	s	s	m	s
	4	0	0	0	0	0	0	m	s
	5	0	0	0	s	m	s	0	0
	6	0	0	0	s	0	0	s	m

Table 4 $R_3 := \left(r^3_{n,m}\right)_{n=1..8,m=1..8}$

$r^3_{n,m}$		Application systems m = 1, …, 8							
		1	2	3	4	5	6	7	8
Application systems $n = 1, \ldots, 8$	1	0	0	0	0	0	0	0	0
	2	0	0	0	0	0	0	0	1
	3	0	0	0	0	0	0	0	0
	4	0	0	0	0	0	0	0	0
	5	0	0	0	0	0	0	0	0
	6	0	0	0	0	0	0	0	0
	7	0	1	0	0	0	0	0	0
	8	0	0	0	0	1	1	0	0

Example (Part 1)

Suppose:

- A set of enterprise functions $\underline{EF} := \{ef_1, ..., ef_6\}$ as follows: ef_1 = "administrative admission", ef_2 = "billing", ef_3 = "care planning", ef_4 = "coding diagnoses and procedures", ef_5 = "imaging", and ef_6 = "report writing".
- A set of entity types $\underline{ET} := \{et_1, ..., et_6\}$ as follows: et_1 = "care plan", et_2 = "diagnosis (coded)", et_3 = "patient identification and case (ADT information)", et_4 = "procedure (coded)", et_5 = "radiological image", and et_6 = "radiological report";
- A matrix $ACC := \left(acc_{p,q}\right)_{p=1..6,q=1..6}$

  as can be seen in Table 1. The matrix $ACC := \left(acc_{p,q}\right)_{p=1..6,q=1..6}$ is illustrated in Figure 1.
- A set of application systems $AS := \{as_1, ..., as_8\}$ as follows: as_1 = "CIS clinical information system", as_2 = "communication server", as_3 = "controlling system", as_4 = "image serving system", as_5 = "MRT scanner system", as_6 = "PACS", as_7 = "PMS patient management system", and as_8 = "RIS radiology information system". In Figure 2 the application systems are depicted as rounded rectangles.
- A support matrix

  $SUP := \left(sup_{p,n}\right)_{p=1..6,n=1..8}$

  describing the relationship between enterprise functions and application systems as can be seen in Table 2.
- A storage describing matrix

  $STORE := \left(store_{q,n}\right)_{q=1..6,n=1...8}$

  describing where entity types are stored can be seen in Table 3.
- A communication matrix

  $R_q := \left(r^q_{n,m}\right)_{n=1..8,m=1..8}$

  for every entity type $et_q \in \underline{ET}$, as for example $R_3 := \left(r^3_{n,m}\right)_{n=1..8,m=1...8}$

  for et_3 ("Patient identification and case (ADT information)") (see Table 4).

For et_1 it shall hold: $r^1_{n,m} = 0, n = 1...8, m = 1...8$, i.e. et_1 is not communicated. For the other entity types the following links shall be given: $r^2_{8,2} = r^2_{2,7} = 1$, $r^4_{8,2} = r^4_{2,7} = 1$, $r^5_{5,6} = r^5_{6,4} = 1$, $r^6_{2,1} = r^6_{8,4} = 1$. The respective links are illustrated and annotated by the indices of the transported entity types in Figure 2.

3. Describing Data Availability by a Logical Predicate

Data availability in an information system simply means that data which is needed in a particular application system can actually reach this application system. This is similar to the data flow problem in compiler theory [8]. These problems can be solved by applying methods from graph theory to find shortest paths in a graph. In compiler theory this graph would consist of blocks of code and the reach of data definitions to certain blocks would be analyzed.

But, those methods can only be applied if the underlying problem is formalized in a way that adequate graphs can be con-

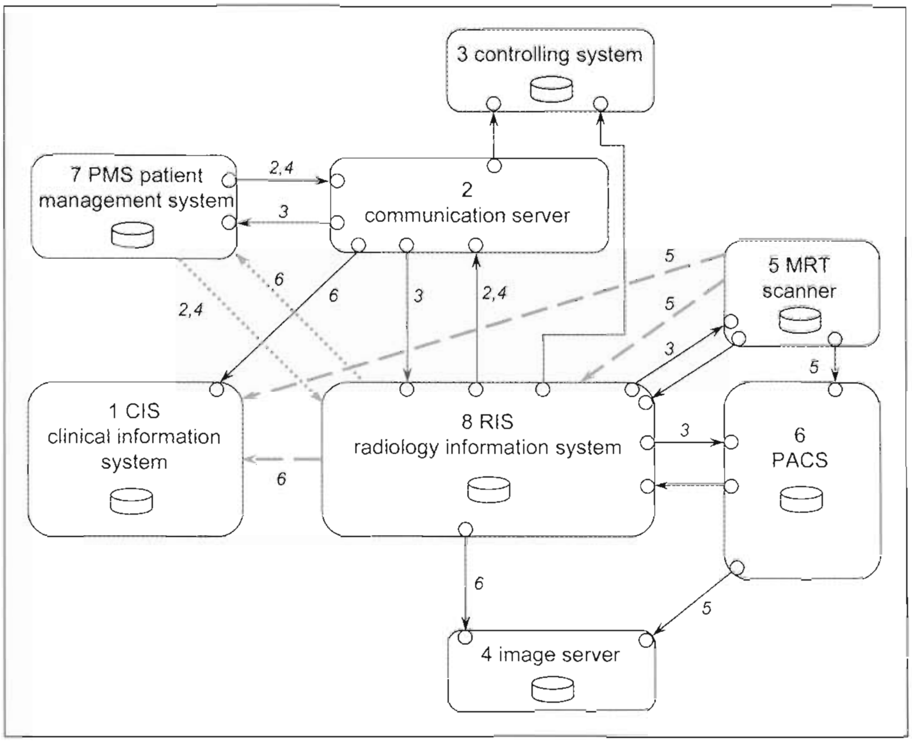

Fig. 2 Application systems and their communication links in a model of an information system as described before in this example. Communication links are illustrated by arrows showing the direction of transport of entity types; the annotated numbers q refer to the respective entity types $et_q \in \underline{ET}$ as being transported via the respective link and defined in S18.tif. The circles denote interfaces which are not important in this context. Note that some links are not annotated by indices of entity types; these links are not relevant for our example. Dashed and dotted arrows indicate needed but missing communication links.

$$\underline{UnavailableET}_p^{int} := \left\{ et_q \in \underline{ET}_p^{int} \,\middle|\, \neg DATA\_AVAILABILITY\left(\underline{AS}_q^{store}, \underline{AS}_p^{sup}, et_q\right)\right\}$$

Fig. 3 Set of interpreted entity types being actually not available at the application systems where they are needed

structed. Using the ontology defined before we can do so.

On this basis we can now define a logical predicate checking even complex health information systems for being able to make data of certain entity types (et_q), which are actually held by some application systems ($\underline{AS}^{source}$), available for certain other application systems ($\underline{AS}^{target}$). The following predicate shall check the graph given by the adjacency matrix R_q for appropriate paths:

$DATA\_AVAILABILITY(\underline{AS}^{source}, \underline{AS}^{target}, et_q) :\Leftrightarrow$
$\forall as_t \in \underline{AS}^{target} : \exists as_s \in \underline{AS}^{source}$:
$\exists$ path from as_s to as_t in the graph related to adjacency matrix R_q

This predicate is true if and only if data of the entity type can reach all target application systems from at least one of the source systems.

Given a graph, checking for those paths is no new problem and efficient solutions like the Floyd-Warshall algorithm [9] can be found in graph theory.

Example (Part 2)

Let us consider et_5 ("radiological image") and take its storing application systems $\{as_4, as_5, as_6\}$ as $\underline{AS}^{source}$. Radiological images are needed at the "CIS clinical information system" as_1, because as_1 supports report writing, which of course needs the images. Therefore we take $\{as_1\}$ as $\underline{AS}^{target}$. In this example we can easily find out by looking at Figure 2, that there is no appropriate path to deliver radiological images from one of the storing systems to the "CIS clinical information system". Hence, the predicate is false:

$DATA\_AVAILABILITY$
$(\{as_4, as_5, as_6\}, \{as_1\}, et_5) \Leftrightarrow$ false.

4. Applying the Predicate in Information Management

The predicate *DATA_AVAILABILITY* can support information management in controlling even complex information systems and helps discovering problems, which otherwise would be covered by the lots of details within an information system's architecture.

4.1 Unavailability of Interpreted Entity Types

Let

$$\underline{ET}_p^{int} := \left\{ et_q \in \underline{ET} \,\middle|\, acc_{p,q} = i \vee acc_{p,q} = iu \right\}$$

be the set of entity types interpreted by an enterprise function

$$\underline{AS}_p^{sup} := \left\{ as_n \in \underline{AS} \,\middle|\, sup_{p,n} = 1 \right\}$$

the set of application systems supporting ef_p and

$$\underline{AS}_q^{store} := \left\{ as_n \in \underline{AS} \,\middle|\, store_{q,n} \neq 0 \right\}$$

the set of application systems storing entity type et_q.

The following set $\underline{UnavailableET}_p^{int}$ contains the entity types being actually not available, because there are no corresponding communication links to deliver the respective data from the storing application systems to application systems supporting the execution of enterprise function ef_p (see Fig. 3).

Example (Part 3)

The sets of

- interpreted entity types are:
  $\underline{ET}_1^{\text{int}} = \emptyset$, $\underline{ET}_2^{\text{int}} = \{\text{et}_2,\text{et}_3,\text{et}_4\}$, $\underline{ET}_3^{\text{int}} = \{\text{et}_5,\text{et}_6\}$, $\underline{ET}_4^{\text{int}} = \{\text{et}_6\}$, $\underline{ET}_5^{\text{int}} = \{\text{et}_3\}$, $\underline{ET}_6^{\text{int}} = \{\text{et}_3,\text{et}_5\}$,
- supporting application systems are:
  $\underline{AS}_1^{\text{sup}} = \{\text{as}_7\}$, $\underline{AS}_2^{\text{sup}} = \{\text{as}_7\}$, $\underline{AS}_3^{\text{sup}} = \{\text{as}_1,\text{as}_4\}$, $\underline{AS}_4^{\text{sup}} = \{\text{as}_1,\text{as}_8\}$, $\underline{AS}_5^{\text{sup}} = \{\text{as}_5\}$, $\underline{AS}_6^{\text{sup}} = \{\text{as}_8\}$,
- storing application systems are:
  $\underline{AS}_1^{\text{store}} = \{\text{as}_1\}$, $\underline{AS}_2^{\text{store}} = \{\text{as}_7,\text{as}_8\}$, $\underline{AS}_3^{\text{store}} = \{\text{as}_5,\text{as}_6,\text{as}_7,\text{as}_8\}$, $\underline{AS}_4^{\text{store}} = \{\text{as}_7,\text{as}_8\}$, $\underline{AS}_5^{\text{store}} = \{\text{as}_4,\text{as}_5,\text{as}_6\}$, $\underline{AS}_6^{\text{store}} = \{\text{as}_4,\text{as}_7,\text{as}_8\}$;

Let us first check the entity types being interpreted by enterprise function ("care planning") for unavailability (see Fig. 4).

We can find out further interpreted entity types being unavailable similarly:

$\underline{UnavailableET_1^{\text{int}}} = \emptyset$, $\underline{UnavailableET_2^{\text{int}}} = \emptyset$,
$\underline{UnavailableET_4^{\text{int}}} = \{\text{et}_6\}$,
$\underline{UnavailableET_5^{\text{int}}} = \emptyset$, $\underline{UnavailableET_6^{\text{int}}} = \{\text{et}_5\}$

Unavailability is marked in Figure 1 by dashed arrows starting at the unavailable entity type and pointing to the 'suffering' enterprise function.

$$\begin{aligned}
&\underline{UnavailableET_3^{\text{int}}}\\
&= \left\{\text{et}_q \in \{\text{et}_5,\text{et}_6\} \,\middle|\, \neg DATA_AVAILABILITY\left(\underline{AS}_q^{\text{store}},\underline{AS}_3^{\text{sup}},\text{et}_q\right)\right\}\\
&= \left\{\text{et}_5 \,\middle|\, \neg DATA_AVAILABILITY\left(\underline{AS}_5^{\text{store}},\underline{AS}_3^{\text{sup}},\text{et}_5\right)\right\}\\
&\quad \cup \left\{\text{et}_6 \,\middle|\, \neg DATA_AVAILABILITY\left(\underline{AS}_6^{\text{store}},\underline{AS}_3^{\text{sup}},\text{et}_6\right)\right\}\\
&= \left\{\text{et}_5 \,\middle|\, \neg DATA_AVAILABILITY\left(\{\text{as}_4,\text{as}_5,\text{as}_6\},\{\text{as}_1,\text{as}_4\},\text{et}_5\right)\right\}\\
&\quad \cup \left\{\text{et}_6 \,\middle|\, \neg DATA_AVAILABILITY\left(\{\text{as}_4,\text{as}_7,\text{as}_8\},\{\text{as}_1,\text{as}_4\},\text{et}_6\right)\right\}\\
&= \{\text{et}_5 \,|\, true\} \cup \{\text{et}_6 \,|\, true\} = \{\text{et}_5,\text{et}_6\}
\end{aligned}$$

Fig. 4 Set of interpreted entity types being actually not available at the application systems supporting enterprise function "care planning"

$$\underline{MissingLNK_p^{\text{int}}} := \left\{ (\text{as}_n,\text{as}_m,\text{et}_q) \in \underline{AS} \times \underline{AS}_p^{\text{sup}} \times \underline{ET}_p^{\text{int}} \,\middle|\, \begin{array}{l} \text{as}_n \in \underline{AS}_q^{\text{master}} \wedge \\ \neg DATA\_AVAILABILITY\left(\{\text{as}_n\},\{\text{as}_m\},\text{et}_q\right) \end{array} \right\}$$

Fig. 5 Set of communication links which have to be realized to overcome non-availability of interpreted entity types

4.2 Missing Communication Links for Interpreted Entity Types

Based on the detected entity types being unavailable for their interpreting functions, communication links should be identified which could help to overcome these problems. The following set indicates those communication links, which should be added to the information system to assure that all entity types of $\underline{ET}_p^{\text{int}}$ would become available for enterprise function ef_p. These communication links have to ensure, that the entity types can be transported from their master application systems to the application systems supporting the function ef_p. Let therefore be

$\underline{AS}_q^{\text{master}} := \{\text{as}_n \in \underline{AS} \,|\, \text{store}_{q,n} = m\}$ the set of application systems being master of an entity type ef_q. The set $\underline{MissingLNK_p^{\text{int}}}$ shall contain links described by their starting and endpoints and the entity type transported (as_n, as_m, et_q). It shall hold, that et_q is interpreted by ef_p ($\text{et}_q \in \underline{ET}_p^{\text{int}}$), that the starting application system is master of et_q ($\text{as}_n \in \underline{AS}_q^{\text{master}}$), that the endpoint supports executing function ef_p ($\text{as}_m \in \underline{AS}_p^{\text{sup}}$) and that there does not exist any path from as_n to as_m to transport et_q ($\neg DATA\_AVAILABILITY(\{\text{as}_n\},\{\text{as}_m\},\text{et}_q)$) yet (see Fig. 5).

Example (Part 4)

The sets of supporting application systems are:
$\underline{AS}_1^{\text{master}} = \{\text{as}_1\}$, $\underline{AS}_2^{\text{master}} = \{\text{as}_7\}$, $\underline{AS}_3^{\text{master}} = \{\text{as}_7\}$, $\underline{AS}_4^{\text{master}} = \{\text{as}_7\}$, $\underline{AS}_5^{\text{master}} = \{\text{as}_5\}$, $\underline{AS}_6^{\text{master}} = \{\text{as}_8\}$

Let us first determine the links to be added to overcome unavailability of entity types being interpreted by enterprise function ef_3 ("care planning") (see Fig. 6).

We can find out further links similarly:

$\underline{MissingLNK_1^{\text{int}}} = \emptyset$, $\underline{MissingLNK_2^{\text{int}}} = \emptyset$,
$\underline{MissingLNK_4^{\text{int}}} = (\text{as}_8,\text{as}_1,\text{et}_6)$, $\underline{MissingLNK_5^{\text{int}}} = \emptyset$,
$\underline{MissingLNK_6^{\text{int}}} = (\text{as}_5,\text{as}_8,\text{et}_5)$

The links which have to be added are marked by annotated dashed arrows in Figure 2. This means, that interfaces and communication links have to be implemented for transporting radiological images from MRT scanners to the RIS, and to the CIS in order to assure that images

$$
\begin{aligned}
&\underline{MissingLNK}^{\text{int}}_{3} \\
&= \left\{ (\text{as}_n, \text{as}_m, \text{et}_q) \in \underline{AS} \times \underline{AS}_3^{\text{sup}} \times \underline{ET}_3^{\text{int}} \middle| \begin{array}{l} \text{as}_n \in \underline{AS}_q^{\text{master}} \wedge \\ \neg DATA\_AVAILABILITY(\{\text{as}_n\},\{\text{as}_m\},\text{et}_q) \end{array} \right\} \\
&= \left\{ (\text{as}_n, \text{as}_m, \text{et}_q) \in \underline{AS} \times \{\text{as}_1,\text{as}_4\} \times \{\text{et}_5,\text{et}_6\} \middle| \begin{array}{l} \text{as}_n \in \underline{AS}_q^{\text{master}} \wedge \\ \neg DATA\_AVAILABILITY(\{\text{as}_n\},\{\text{as}_m\},\text{et}_q) \end{array} \right\} \\
&= \left\{ (\text{as}_n, \text{as}_m, \text{et}_5) \in \underline{AS} \times \{\text{as}_1,\text{as}_4\} \times \underline{ET} \middle| \begin{array}{l} \text{as}_n \in \underline{AS}_5^{\text{master}} \wedge \\ \neg DATA\_AVAILABILITY(\{\text{as}_n\},\{\text{as}_m\},\text{et}_5) \end{array} \right\} \\
&\quad \cup \left\{ (\text{as}_n, \text{as}_m, \text{et}_6) \in \underline{AS} \times \{\text{as}_1,\text{as}_4\} \times \underline{ET} \middle| \begin{array}{l} \text{as}_n \in \underline{AS}_6^{\text{master}} \wedge \\ \neg DATA\_AVAILABILITY(\{\text{as}_n\},\{\text{as}_m\},\text{et}_6) \end{array} \right\} \\
&= \left\{ (\text{as}_n, \text{as}_m, \text{et}_5) \in \underline{AS} \times \{\text{as}_1,\text{as}_4\} \times \underline{ET} \middle| \begin{array}{l} \text{as}_n \in \{\text{as}_5\} \wedge \\ \neg DATA\_AVAILABILITY(\{\text{as}_n\},\{\text{as}_m\},\text{et}_5) \end{array} \right\} \\
&\quad \cup \left\{ (\text{as}_n, \text{as}_m, \text{et}_6) \in \underline{AS} \times \{\text{as}_1,\text{as}_4\} \times \underline{ET} \middle| \begin{array}{l} \text{as}_n \in \{\text{as}_8\} \wedge \\ \neg DATA\_AVAILABILITY(\{\text{as}_n\},\{\text{as}_m\},\text{et}_6) \end{array} \right\} \\
&= \{ (\text{as}_5,\text{as}_1,\text{et}_5) | \neg DATA\_AVAILABILITY(\{\text{as}_5\},\{\text{as}_1\},\text{et}_5) \} \\
&\quad \cup \{ (\text{as}_5,\text{as}_4,\text{et}_5) | \neg DATA\_AVAILABILITY(\{\text{as}_5\},\{\text{as}_4\},\text{et}_5) \} \\
&\quad \cup \{ (\text{as}_8,\text{as}_1,\text{et}_6) | \neg DATA\_AVAILABILITY(\{\text{as}_8\},\{\text{as}_1\},\text{et}_6) \} \\
&\quad \cup \{ (\text{as}_8,\text{as}_4,\text{et}_6) | \neg DATA\_AVAILABILITY(\{\text{as}_8\},\{\text{as}_4\},\text{et}_6) \} \\
&= \{ (\text{as}_5,\text{as}_1,\text{et}_5) | \mathit{true} \} \cup \{ (\text{as}_5,\text{as}_4,\text{et}_5) | \mathit{false} \} \cup \{ (\text{as}_8,\text{as}_1,\text{et}_6) | \mathit{true} \} \cup \{ (\text{as}_8,\text{as}_4,\text{et}_6) | \mathit{false} \} \\
&= \{ (\text{as}_5,\text{as}_1,\text{et}_5), (\text{as}_8,\text{as}_1,\text{et}_6) \}
\end{aligned}
$$

Fig. 6 Set of communication links which have to be realized to overcome non-availability of interpreted entity types at the application systems supporting enterprise function "care planning"

$$
(\exists \text{as}_n \in \underline{AS}_p^{\text{sup}} \cdot \neg DATA\_AVAILABILITY(\{\text{as}_n\}, \underline{AS}_q^{\text{store}}, \text{et}_q)).
$$

Fig. 7 Specification of entity types being not transported properly to their storing places

$$
\underline{UnstoredET}_p^{upd} := \left\{ \text{et}_q \in \underline{ET}_p^{upd} \middle| \begin{array}{l} \neg(\underline{AS}_p^{\text{sup}} \subseteq \underline{AS}_q^{\text{master}}) \vee \\ \exists \text{as}_n \in \underline{AS}_p^{\text{sup}} \cdot \neg DATA\_AVAILABILITY(\{\text{as}_n\}, \underline{AS}_q^{\text{store}}, \text{et}_q) \end{array} \right\}
$$

Fig. 8 Set of updated but unproperly stored entity types

are available when reports have to be written. In addition, links are needed to transport radiological reports from RIS to CIS.

But note that not always the direct links are best suited. In the setting of this example it may be e.g. appropriate to send reports to the communication server, which in turn could use the existing direct link to CIS to send reports to this system.

4.3 Improper Storage of Updated Entity Types

Availability of data for functions interpreting them requires proper storage of this data before. Data is stored in the context of executing functions updating entity types. Since we postulated a concept of master application systems and redundant storage of data in different application systems, we have to check our information systems for related problems. Given

$$\underline{ET}_p^{\text{upd}} := \{ \text{et}_q \in \underline{ET} | \text{acc}_{p,q} = \boldsymbol{u} \vee \text{acc}_{p,q} = \boldsymbol{iu} \}$$

as the set of entity types updated by function ef_p. For every function ef_p the set $\underline{UnstoredET}_p^{upd}$ will indicate the entity types of $\underline{ET}_p^{upd}$, which actually are

- updated by an application system being not master

  $(\neg(\underline{AS}_p^{\text{sup}} \subseteq \underline{AS}_q^{\text{master}}))$

  (not allowed by the concept of master systems),
- or not transported properly to their storing places (see Fig. 7).

This leads to the combined formula in Figure 8.

Example (Part 5)

The sets of updated entity types are:
$\underline{ET}_1^{\text{upd}} = \{\text{et}_3\}$, $\underline{ET}_2^{\text{upd}} = \varnothing$, $\underline{ET}_3^{\text{upd}} = \{\text{et}_1\}$,
$\underline{ET}_4^{\text{upd}} = \{\text{et}_2,\text{et}_4\}$, $\underline{ET}_5^{\text{upd}} = \{\text{et}_5\}$, $\underline{ET}_6^{\text{upd}} = \{\text{et}_6\}$

The entity types being not updated and stored properly, are as follows:

$\underline{UnstoredET}_1^{upd} := \varnothing$, $\underline{UnstoredET}_2^{upd} := \varnothing$, $\underline{UnstoredET}_3^{upd} := \{et_1\}$, $\underline{UnstoredET}_4^{upd} = \{et_2, et_4\}$, $\underline{UnstoredET}_5^{upd} := \varnothing$, $\underline{UnstoredET}_6^{upd} = \{et_6\}$

This means, that care plans, diagnoses, procedures and reports are not stored or updated properly in this information system and appropriate measures have to be taken.

Improper storage is marked in Figure 1 by dashed arrows pointing to the entity type which is stored improperly and starting at the enterprise functions causing the problem.

4.4 Missing Communication Links for Updated Entity Types

Based on the detected entity types being stored or updated improperly by their updating functions, communication links should be indentified which could help to overcome these problems. The following set indicates those communication links, which have to be added to the information system to assure that all entity types of $\underline{ET}_p^{upd}$ can be stored and updated properly by enterprise function ef_p (see Fig. 9).

$$\underline{MissingLNK}_p^{upd} := \left\{ (as_n, as_m, et_q) \in \underline{AS} \times \underline{AS} \times \underline{ET}_p^{upd} \;\middle|\; \begin{array}{l} as_n \in \underline{AS}_q^{master} \wedge as_m \in \underline{AS}_q^{store} \wedge \\ \neg DATA\_AVAILABILITY(\{as_n\}, \{as_m\}, et_q) \end{array} \right\}$$

Fig. 9 Set of communication links which have to be realized to overcome unproper storage of updated entity types

Example (Part 6)

The links to be added to overcome improper storage of updated entity types can be computed similarly as illustrated in part 4 of the example. These computations result in the following sets; they are marked by annotated dotted arrows in Figure 2:

$\underline{MissingLNK}_1^{upd} = \varnothing$, $\underline{MissingLNK}_2^{upd} = \varnothing$,

$\underline{MissingLNK}_3^{upd} = \varnothing$,

$\underline{MissingLNK}_4^{upd} = \{(as_7, as_8, et_2), (as_7, as_8, et_4)\}$,

$\underline{MissingLNK}_5^{upd} = \varnothing$,

$\underline{MissingLNK}_6^{upd} = \{(as_8, as_7, et_6)\}$

This means for example, that interfaces and communication links have to be implemented for transporting coded diagnoses and procedures from the "PMS patient management system" to the RIS. This is necessary because the PMS had been denoted to be master of coded diagnoses and procedures; and therefore updated data of these entity types have to be transported to the RIS. Perhaps the origin of the problem is not the missing link, but the declaration of PMS as master for coded diagnoses and procedures.

5. Discussion

This paper is another attempt to contribute to a theory of quality of information systems. Similar as key performance indicators for functional redundancy, informational redundancy, degree of heterogeneity and degree of computer-support in [10] it contributes to the aspect of structural quality [11] of information systems. Talking about structural quality makes clear, that data availability as described here may be necessary but by no means sufficient for the overall assessment of trustworthiness of health information systems.

Besides presenting one more criterion for structural and architectural quality of information systems, we wanted to show that formal methods used in graph theory and compiler theory can also be used in the field of health information systems, which is still dominated by pragmatic, straightforward approaches instead of systematic methods. The very prerequisite of such methods is an appropriate ontology for describing information systems. And if the methods shall be shared, a shared ontology is needed as well. We understand the ontology presented here as a proposal for a shared ontology for describing health information systems.

Moreover, this approach is not only a contribution to theoretical concepts, but intended to be as well an effective as an efficient support for information management to improve quality of information systems in practice. We hope that we could demonstrate in Chapter 4 the practical impact and therefore the effectiveness of our approach for its application in information management. We claim also for efficiency because information managers do not need to model their complex information systems especially for applying the predicates and algorithms proposed here. Of course, such efforts would not pay for only this end. Moreover, a model of an information system based on the meta model introduced here and perhaps by having used the 3LGM² tool [12] can also be used e.g. to analyze costs [13] or can serve as the basis for configuration management [14] and change management ([15], p 42) without alterations of the model.

Acknowledgments
Parts of this work have been supported by grants of the Deutsche Forschungsgemeinschaft (DFG). Thanks to Franziska Jahn and Ulrike Müller for improving the readability of the paper.

References

1. Lorenzi NM, Riley RT. Organizational Aspects of Health Informatics: Managing Technological Change. New York: Springer; 1995.
2. Breu R, Mitchell JC, Sztipanovits J, Winter A. Model-Based Design of Trustworthy Health Information Systems: http://www.dagstuhl.de/de/programm/kalender/semhp/?semnr=2009073, accessed 2008-06-10; 2008.
3. Winter A, Brigl B, Wendt T. Modeling Hospital Information Systems (Part 1): The Revised Three-Layer Graph-Based Meta Model 3LGM². Methods Inf Med 2003; 42 (5): 544-551
4. Kuhn KA, Giuse DA. From Hospital Information Systems to Health Information Systems. Methods Inf Med 2001; 40: 275-287.
5. Winter A, Brigl B, Funkat G, Häber A, Heller O, Wendt T. 3LGM²-Modeling to Support Management of Health Information Systems. International Journal of Medical Informatics 2007; 76 (2–3): 145-150. doi:http://dx.doi.org/10.1016/j.ijmedinf.2006.07.007
6. Chimiak-Opoka J, Giesinger G, Innerhofer-Oberperfler F, Tilg B. Tool-Supported Systematic Model Assessment. In: Mayr HC, Breu R, editors. Modellierung 2006, volume 82 of Lecture Notes

in Informatics (LNI). Bonn, Germany: Gesellschaft fuer Informatik; 2006. pp 183-192.
7. Alschuler L. First Do No Harm: A Standard for Electronic Communication in Healthcare. In: Dudeck J, Blobel B, Lordieck W, Bürkle T, editors. New Technologies in Hospital Information Systems. Amsterdam: IOS Press; 1997. pp 141-147.
8. Marlowe TJ, Ryder BG. Properties of Data Flow Frameworks. Acta Informatica 1990; 28 (2): 121-163.
9. Cormen TH, Leiserson C, Rivest RL, Stein C. Introduction to Algorithms: MIT Press; 2001.
10. Brigl B, Hübner-Bloder G, Wendt T, Haux R, Winter A. Architectural Quality Criteria for Hospital Information Systems. In: Friedman CP, Ash J, Tarczy-Hornoch P, editors. American Medical Informatics Association 2005 Proceedings: Biomedical and Health Informatics – From Foundations to Applications to Policy. Bethesda: American Medical Informatics Association; 2005. pp 81-85.
11. Donabedian A. Quality of care: problems of measurement. II. Some issues in evaluating the quality of nursing care. Am J Public Health Nations Health 1969; 59 (10): 1833-1836.
12. Wendt T, Häber A, Brigl B, Winter A. Modeling Hospital Information Systems (Part 2): Using the 3LGM² Tool for Modeling Patient Record Management. Methods Inf Med 2004; 43 (3): 256-267.
13. Kutscha A, Brigl B, Winter A. Modeling Economic Aspects of Hospital Information Systems to Give Decision Support for Strategic Information Management. Studies in Health Technology and Informatics 2006; 124: 672-678.
14. Office of Government Commerce (OGC). Service Support. Series: ITIL – The Key to Managing IT Services. London: TSO; 2000.
15. Office of Government Commerce (OGC). Service Transition. London: TSO; 2007.

Correspondence to:
Prof. Dr. Alfred Winter
Institute for Medical Informatics, Statistics and Epidemiology
University of Leipzig
Haertelstr. 16-18
04107 Leipzig
Germany
E-mail: alfred.winter@imise.uni-leipzig.de