

Chapter 3

Technological Perspective: Architecture, Integration, and Standards



3.1 Introduction

From the previous chapter, we already know that a health information system is the socio-technical subsystem of a health care setting which comprises all data, information, and knowledge processing as well as the associated human and technical actors in their respective data, information, and knowledge processing roles. In this chapter, we first look at what these information systems look like, i.e., we take the technological perspective and examine the architecture of information systems. This perspective is then complemented by the management perspective in Chap. 4.

Section 2.11 taught us that health information systems are constructs built from a variety of components. We will go through the three layers of information systems: the domain layer, the logical tool layer, and the physical tool layer. For each layer, we will explain, step by step, the layer's components and how they are to be assembled and integrated to achieve what users experience as the health information system. Although we keep the non-computer-based part in mind, we will focus on the computer-based part.

In Sect. 3.2, we start at the domain layer and discuss the kind of data that must be processed in health care settings before, in Sect. 3.3, we present the functions interpreting or updating these data. From Sect. 3.4, we describe the tools for data, information, and knowledge processing to be used in health care settings. Starting at the logical tool layer and its application components, we explain how interoperable application components can be integrated, i.e., connected to work together seamlessly (Fig. 3.1) and how various approaches for integration lead to different architectural styles. At Sect. 3.10, we start describing tools at the physical tool layer. Again, we are dealing with integration. But we must be aware that at the physical tool layer, physical data processing systems and the challenges of integration are different from those at the logical tool layer.

In this chapter, we will explain the more generic technological concepts for health information systems which, in principle, are valid for most life situations and



Fig. 3.1 Health information systems constitute an essential part of providing good health care. Decisions are made by an interdisciplinary tumor board. (Courtesy of Karin Kaiser/MHH)

health care settings. However, in certain situations and settings, there are specific challenges and requirements leading to specific architectures for the respective health information systems. In Chap. 6, we will therefore discuss how information systems for certain life situations and in certain settings actually take up these challenges and fulfill these requirements.

After reading this chapter, you should be able to

- differentiate the most important entity types which are processed in health care,
- explain important information-processing functions of health care settings,
- name and describe types of application systems used in health care settings,
- explain typical architectural styles of health information systems and describe them by differentiating concepts and relationships at the domain layer and the logical tool layer,
- explain the different types of integrity, interoperability, and integration at the logical and physical tool layer,
- select suitable technologies and tools for achieving integration, and
- describe the available standards for different aspects of interoperability and select suitable standards for achieving specific types of integrity, interoperability, and integration.

For you as a reader, achieving these learning objectives is the prerequisite for being able to construct health information systems that are appropriate to people's life situations and that meet the stakeholders' needs. However, you must be aware

that these information systems will only be useful to people if they are systematically managed from the start and if their quality is systematically monitored. We will take a closer look at these aspects later on in Chaps. 4 and 5.

Please note that the terms highlighted in italics are terms from the glossary or represent functions or application system types (Sects. 3.3 and 3.4).

3.2 Domain Layer: Data to be Processed and Provided

We will now look at data that represent information and knowledge in both the health care sector and biomedical research. We need to be aware that data is not only stored and processed in one particular information system, but that it often also needs to be provided to or shared with the information system of another facility or setting. For example, data from health care should be provided for research so that medical progress is possible. And data from research should be provided for use in health care to apply new knowledge. This mutual relationship is often described with the concept of the “learning health care system,” in which data from everyday medical care is used to gain new insights in medical research and the research results are constantly fed back into practical care (translation) and medical education.

We can distinguish data according to different aspects:

- personal vs. non-personal data,
- standardized vs. non-standardized data,
- data on particular entity types (compare Sect. 2.8).

These distinctions are useful because

- personal data require special security measures and are subject to certain restrictions on their processing,
- non-standardized data are more common in the health care sector but much more difficult to process and to be used by machines than standardized data, and
- data on specific entity types require specific application systems and algorithms for processing.

After this section, you will understand what kinds of data are processed in and provided by most of the health care settings and in biomedical research.

3.2.1 *Personal vs. Non-personal Data*

According to the European General Data Protection Regulation (GDPR) “... ‘personal data’ means any information relating to an identified or identifiable natural person” [1]. Data on health and illness are mostly created as personal data. For example, a 12-lead electrocardiogram (ECG), data on physical/mental well-being and weight, the results of an echocardiography, or the activity data for a particular

person like Mr. Russo (Sect. 1.4) are personal data. Individuals' health data belong to the most sensitive personal data on humans.

By cumulating data, the relation to a single identifiable person can be reduced. Whether this really turns personal data into non-personal data, however, depends on the information represented. Look, for example, at the German city of Leipzig with around 500,000 inhabitants. Data describing that there are 50,000 people with elevated body temperature in Leipzig in December would certainly not be personal data. On the other hand, if there is a certain rare disease occurring only once among 1,000,000 people, and there is a record with data on a citizen of Leipzig suffering from this rare disease, then this is personal data, even if the data do not include a name, birthday, or other identifier. This is similarly true for human genetic data. Since the genetic code is individual for each person, such data must always be considered personal data.

Personal health data must only be accessible to those persons the individual has authorized before. A health information system must guarantee this requirement of personal data privacy. Likewise, the health information system must ensure data security and data safety of personal health data. While data security ensures availability, confidentiality, integrity, and protection from unauthorized access, data safety concerns protecting personal data against loss.

Personal data on certain persons may only be processed by other persons or facilities if the person expressly consents to having their data processed. If the data are to be processed for another purpose at a later time, the person must give their consent again. In medicine, for example, this means that health data collected from a patient during treatment at a particular health care facility may only be transferred to another health care or research facility with the patient's consent. Thus, Mr. Russo must also first give his explicit and informed consent for his data to be used in the scientific study to investigate the effect of close-knit home monitoring on rehospitalization in patients with heart failure. This also applies to pseudonymized health data, i.e., data in which the directly identifying data on the person, for example, the name or a patient number, have been replaced by a number that cannot be directly assigned to a person.

National legislation may allow for different levels of consent. For example, it may be a requirement that individuals must give permission for their data to be used individually for each research project. However, it may also be possible for individuals to give permission for the use of their data for a broader research topic or for research in general (often called "broad consent"). It is important, therefore, to know exactly what options are available in your country.

The handling of personal data in the European Union is comprehensively regulated by the GDPR [1].

Not only personal data but also the processing of non-personal data may be subject to restrictions. For example, if the data are research data obtained by a scientist in the course of experiments at great expense, then this scientist has intellectual property (IP) rights. Such data may only be used if this scientist agrees to its use.

3.2.2 *Standardized vs. Non-standardized Data*

When a physician, for example Mr. Russo’s cardiologist, documents the medical treatment, the data used to describe the treatment, for example in the discharge letter, may be recorded as continuous narrative text without further structuring. We refer to such free text as non-standardized data. With free text, a situation can be described in detail and exactly using full linguistic expressiveness—if there is enough time. The disadvantage of such non-standardized data is, however, that the recorded data are hardly comparable, their completeness cannot be checked, and further processing, especially interpreting its semantics by a machine, is very difficult. However, there are promising approaches from artificial intelligence to use natural language processing (NLP) to tag free texts with terminologies (e.g., Systematized Nomenclature of Medicine Clinical Terms (SNOMED CT)) in such a way that further processing is possible. For example, the SNOMED CT code 414024009 may be used to tag the discharge letter describing Mr. Russo’s coronary artery disorder. The other two problems, however, cannot be solved with this approach.

If, prior to documentation, the entity types for which data are to be recorded, the properties (attributes) of the objects of these entity types that are to be documented, and the exact value set of these attributes are defined, then we speak of a *standardized documentation* or of standardized data (example in Fig. 3.2). In the case of standardized documentation, it is easy to see—and to validate by a machine—whether all the desired data have been captured and the data from different sources can be easily compared. Further processing by machine is also well-prepared. To support standardized documentation, certain terminologies exist and can be used. Depending on the purpose of the documentation, these may be terminologies such as the SNOMED CT mentioned above or classifications such as the International

Shoulder pain and Disability Index (SPADI)

Pain scale

How severe is your pain?

Circle the number that best describes your pain where: 0 = no pain and 10 = the worst pain imaginable.

At its worst?	0	1	2	3	4	5	6	7	8	9	10
When lying on the involved side?	0	1	2	3	4	5	6	7	8	9	10
Reaching for something on high shelf?	0	1	2	3	4	5	6	7	8	9	10
Touching the back of your neck?	0	1	2	3	4	5	6	7	8	9	10
Pushing with the involved arm?	0	1	2	3	4	5	6	7	8	9	10

Fig. 3.2 Excerpt from the English version of the Shoulder Pain and Disability Index; a standardized questionnaire for assessing shoulder function [2]

Classification of Diseases (ICD) or the nursing diagnosis classification of the North American Nursing Diagnosis Association (NANDA) International. NANDA may be used to assign the aforementioned case of Mr. Russo to the class 00146, for example, as Mr. Russo still suffers considerable anxiety from his heart problems. Thus, cases assigned to certain classes can be counted and statistically analyzed.

As standardized data, in contrast to free text data, have a certain structure, they are often also called structured data. Accordingly, free text is often referred to as unstructured data.

3.2.3 Entity Types

In health information systems, data on the following entity types are particularly important:

- entity types about individuals,
- entity types about patients' diseases and their treatment,
- entity types about organizing health care,
- entity types about knowledge.

In the previous two sections, we saw the difference between personal and non-personal data and between standardized and non-standardized data. Please note that a lot of data describing individuals are personal data. However, this is not always true. For example, diagnoses without reference to a specific person are non-personal. There are also differences in terms of standardization for data describing individuals. For example, findings can be standardized or non-standardized.

3.2.3.1 Entity Types About Individuals

The following entity types describe individuals. They are listed in alphabetical order.

Entity type	Description
Health care professional	Health care professionals are persons who treat, according to their specialization (e.g., nephrology or pediatrics), patients with certain leading diagnoses. Examples of health care professionals include physicians and nurses
Human resource	Human resources are persons working in health care facilities, i.e., health care professionals, nurses, administrative staff members, and IT staff members
Informal caregiver	Informal caregivers are persons who are not trained health care professionals but who care for a patient, for example, a relative or close friend
Patient	Patients are persons who are the subject of health care. Information about a patient includes the patient identification number (PIN) and—in transinstitutional information systems—the transinstitutional PIN
Person	Persons are individuals who are dealt with in a certain health information system. This includes individuals who are taking care of their own health in any way or who are subjects of health care at a health care facility. Persons may also be health care professionals, informal caregivers, or participants in clinical trials

In health information systems, it is crucial to be able to identify persons unambiguously in order to avoid mix-ups. Each person must be assigned a unique patient identification number (PIN). This PIN should be valid and unchangeable for a lifetime (i.e., the PIN should not be based on changeable personal attributes such as the name). The PIN is the main prerequisite for being able to merge all of a person's health-related information. Before a PIN can be assigned, the person must be correctly identified (3.3.2.1).

3.2.3.2 Entity Types About Patients' Diseases and Their Treatment

If persons are ill and are the subject of care, we call them patients. Certain data on findings and about diseases of the patients and their therapies is required for health care.

Entity type	Description
Diagnosis	Diagnoses are the identified cause or nature of patients' diseases or medical conditions
Discharge summary	Discharge summaries briefly summarize diagnoses, treatment, and recommendations from the discharging health care facility. The discharge summary is necessary for the receiving health care facilities in order to be able to provide further treatment
Finding	Findings summarize the results of diagnostic procedures for patients such as lab and X-ray examinations. Laboratory findings may consist of the measured values of clinical chemical parameters. But they may also contain complex genetic data from sequencing, also called "omics" data. Each type of data requires specialized software to present these data to medical personnel in such a way that they can interpret them well for diagnostic purposes
Health record	Health records are descriptions about a person's past and present health conditions, for example, disease history, systems review, social history, past medical history, family history, or medication. The health record is necessary for health care professionals in order to make informed clinical decisions
Image	Images are rasterized representations of macro- or microscopic entities or processes within biological systems, generated by using different modalities (e.g., computed tomography, histologic slices, X-ray images) and used for diagnosis, <i>prevention</i> , therapy, and rehabilitation
Informed consent	Informed consent is a patient's consent to the proposed treatment
Medical history	Medical histories comprise all information collected for a patient which is needed as a basis for <i>medical care planning</i> . The documented medical history is the result of medical history taking between the health care professional and the patient
Medical procedure	Medical procedures are procedures performed by physicians for patients, for example, X-ray examinations or operations
Nursing history	Nursing histories comprise all information collected for a patient which is needed as a basis for <i>nursing care planning</i> . The documented nursing history is the result of the admission interview between the nurse and the patient
Nursing procedure	Nursing procedures are procedures performed by nurses for patients, for example, taking blood or taking the temperature

Entity type	Description
Order	Orders are requests made by health care professionals for diagnostic, therapeutic, or drug services, for example, laboratory orders or radiological orders. Orders are directed to health care facilities and other health care professionals
Patient record	Health care facilities have records for their patients, collecting data and documents related to health care in this facility. This part of the patient's health record is called the patient record. Persons thus have multiple patient records if they have been a patient in more than one health care facility. Patient records are those parts of health records which are related to health care in a certain health care facility
Patient record archive	Entities of the entity type "patient record archive" describe how and where the patient record can be found
Procedure	Procedures are "activities performed in the provision of health care (includes medical history taking, physical examination, diagnostic and therapeutic interventions, training and education, and counseling)" [3]
Sample	Samples are specimens taken from a patient or another person, for example, a blood sample or a urine sample
Self-diagnosis	A self-diagnosis is a diagnosis made by an individual for his or her own condition rather than by a health care professional
Self-gathered symptoms	Self-gathered symptoms are signs or supposed signs of a disease that have been noticed during observation of one's own body

3.2.3.3 Entity Types About Managing Health Care

Health care takes place at different places. Further data is needed to organize this care.

Entity type	Description
Appointment	Appointments determine what persons have to be at a certain place at a given time. Examples are appointments for <i>patient admission</i> , examination, or surgery
Bed	In health care facilities with inpatient care (e.g., hospitals and rehabilitation centers), beds must be managed according to their occupation
Case	The cases mostly comprise a patient's stay in a hospital from <i>patient admission</i> to patient discharge or several outpatient treatments related to one disease. Information about a case includes the case identification number (CIN)
Cost unit	Cost units are organizational units of health care facilities responsible for bearing the costs or a part of the costs for the services to be provided for a patient
Drug	Drugs are substances administered to a certain patient for treatment, diagnosis, <i>prevention</i> , or rehabilitation
Food	A certain food must be provided according to different nutritional needs of patients, for example, normal diet and light diet food
Material	Materials such as medical strips, bandages, or needles are needed for <i>patient care</i>

Entity type	Description
Means of transport	Patients may have to be transported. Certain means of transport may be, for example, stretchers, ambulances, or flying ambulances
Medical device	Medical devices are technical or mechanical devices used for diagnostics and treatment. Software supporting diagnostics and treatment may also be considered medical devices. Medical devices are subject to strict legal regulations (medical device regulation (MDR))
Patient transport	Certain patient transports describe how and to what place a certain patient has been or must be transported
Transfer	Certain transfers describe the transfer of a certain patient, for example, by naming the referring physician or reasons for referring
Room	Rooms in the building of any health care setting (e.g., operating room, physician's office, waiting room, or bedroom in patient's home). Rooms have to be managed as a resource for <i>patient care</i>
Service	Health care facilities may provide non-medical services to their patients (e.g., internet access, transportation service)

3.2.3.4 Entity Types About Knowledge

It is not only information about patients that health care professionals need for care. Medical and nursing knowledge is also required.

Entity type	Description
Available medical and nursing knowledge	Available medical and nursing knowledge is medical and nursing knowledge that is available in a wide variety of knowledge sources and can basically be acquired. Some available medical and nursing knowledge is evidence-based knowledge that is confirmed, for example, through clinical trials. Other available knowledge is experiential knowledge from health care professionals or from patients
Classification	Classifications are sets of classes summarizing concepts not to be distinguished during analysis. In particular, there are classifications of diagnoses and classifications of procedures. A classification of diagnoses (e.g., ICD-11 or NANDA) consists of diagnosis classes. A classification of medical procedures (e.g., International Classification of Health Interventions (ICHI)) consists of procedure classes
Clinical pathway	Clinical pathways are evidence-based approaches describing which procedures are to be performed for a specific group of patients when and by whom
Clinical trial	A clinical trial is a research study testing a new treatment, medication, or medical device on patients. Results obtained in a trial may be used for care, and data recorded during care are necessary input for trials
Nomenclature	Nomenclatures (e.g., SNOMED CT) provide terms with codes that can be used to tag (or index) objects in medicine and health care. Objects may be, for example, cases, patients, or documents

Entity type	Description
Medical and nursing knowledge	Medical and nursing knowledge comprises knowledge of various medical and nursing specialties, for example, knowledge about diseases, symptoms, side effects, treatments, and health risks. Knowledge is stored in physicians' and nurses' heads as well as in various media such as books, journals, guidelines, webpages, or databases. A very important source of medical and nursing knowledge is MEDLINE, a bibliographic database of medical publications. It takes a lot of effort to select and acquire the knowledge that a person needs for a specific task from the amount of available knowledge. We therefore want to distinguish available and selected knowledge. Furthermore, it must be taken into account that some knowledge is not directly available, for example, in the form of books and journals, but only in the form of references to such sources
Acquired personal medical and nursing knowledge	Acquired personal medical and nursing knowledge is medical and nursing knowledge that has been selected and acquired by a person according to his or her own needs in order to make decisions regarding his or her own health or the health of others. Here, too, some knowledge is not directly available but only in the form of references to the sources of the knowledge

3.3 Domain Layer: Functions to Be Supported

In the last section, we introduced the data typical for health care settings and described data by entity types. Now we will explain where and in what contexts data on these entity types are processed in health care settings. As explained in Sect. 2.8, we use information-processing functions—or short: functions—to group classes of information-processing activities.

You will remember that functions use input data on certain entity types. The used data is updated, which often results in data on other entity types. In this section, we will present a selection of functions typical for health care settings and will explain which entity types are used and which are updated. However, we will not (yet) consider how they are typically supported by different data, information, and knowledge processing tools. This will be done in Sect. 3.4 and the following sections.

In this section, you will learn about the most important functions to be performed by patients, informal caregivers, health care professionals, and management and administrative staff in health care settings. You will also learn about data that are used or updated by these functions.

To illustrate which entity types a particular function uses and which it updates, we will use diagrams corresponding to those we also used in Sect. 2.14.1. Please read there again what the symbols mean.

3.3.1 *Functions to Be Performed by Patients and Informal Caregivers*

In Sect. 1.2, we looked at the life situations in which people have to deal with health. Initially, such life situations take place in the home setting and require specific action by the individuals and patients concerned. Thus, in our example from Sect. 1.4, Mrs. And Mr. Russo have to cope with their lives despite Mrs. Russo having broken her leg in the bathroom and Mr. Russo having a heart condition. To do so, they must contact with their general practitioner (GP) and with specialists to obtain various information. Their daughters must also participate in the care of their parents. In this context, the daughters are called informal caregivers, just like other relatives or friends who participate in the care of the Russos. In Sects. 1.3.1 and 1.3.3, we had already noted what patients and informal caregivers are particularly concerned about in this context.

In this section, we will now describe the information-processing tasks that patients and informal caregivers have to perform and what entity types are needed. Even though we introduced the term “function” in Sect. 2.8 in the context of health care professionals, it is very well-suited to also describe the information-processing tasks of non-professionals. The corresponding functions include but are not limited to (Fig. 3.4):

- *medical knowledge management,*
- *self-diagnostics,*
- *self-treatment,*
- *arrange appointments,*
- *physician’s orders filling,*
- *prevention.*

3.3.1.1 **Medical Knowledge Management by Non-professionals**

Patients and informal caregivers need medical and nursing knowledge, for example, about specific health conditions and their treatment, about diseases, symptoms, side effects, and about healthy lifestyles and health risks. *Medical knowledge management* by non-professionals as an information-processing function includes gathering such knowledge by consulting health care professionals as well as peers such as fellow patients, relatives, and friends (Fig. 3.3). The corresponding knowledge is gathered in the form of brochures, information material on various media, or references to corresponding sources and media. Patients and informal caregivers select units of knowledge and references to knowledge which they consider helpful.

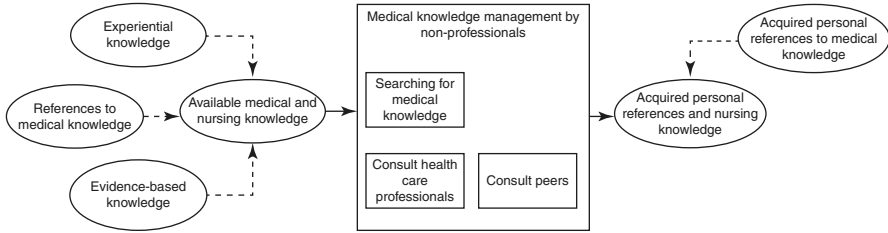


Fig. 3.3 *Medical knowledge management by non-professionals* (for the legend refer to Sect. 2.14.1)

As a result of *medical knowledge management*, patients and informal caregivers will have selected personal knowledge at hand—or in their mind. They must then organize this knowledge according to its perceived importance and relevance.

3.3.1.2 Self-Diagnostics

Individuals who feel ill or who are living a health-conscious lifestyle will carefully monitor and assess their own health conditions or the conditions of household members. By using the resulting self-gathered symptoms and acquired personal medical and nursing knowledge, they may try to identify their disease. This function, *self-diagnostics*, results in a self-diagnosis, which later may lead to *self-treatment*. This is illustrated in Fig. 3.4.

For monitoring, individuals may use health diaries and personal digital devices, for example, smartphone applications that may provide automatic monitoring of symptoms and conditions.

3.3.1.3 Self-Treatment

Self-treatment usually means treating one's own disease without direct medical supervision or intervention and relies on selected personal knowledge. We use this term here to indicate patient actions to treat their disease by themselves, for example, at home. Such *self-treatment* may be based on self-diagnoses or on diagnoses resulting from the execution of diagnostic procedures by health care professionals and filling the respective orders.

Self-treatment includes physical exercise, managing prescribed medication and self-medication, and using digital health applications (DiGAs) to support mental health, for example.

Self-treatment may also include *prevention*, which is better than cure. Many diseases result from unhealthy lifestyles and behavior. A change in behavior can therefore both prevent and, in many cases, make a significant contribution to curing

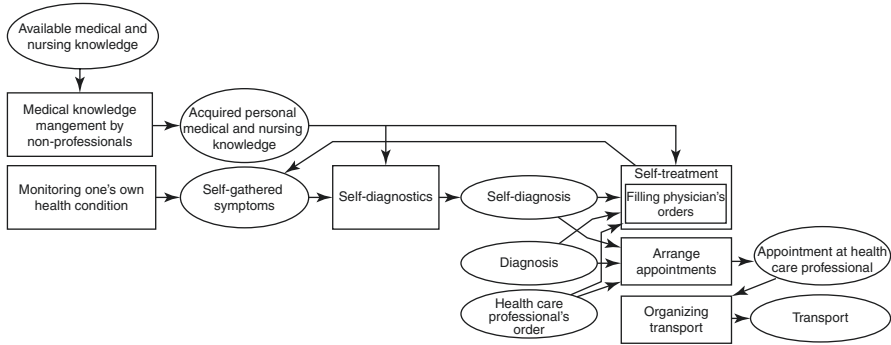


Fig. 3.4 Summary of functions to be performed by patients and informal caregivers (for the legend refer to Sect. 2.14.1)

diseases. *Prevention* through healthy behavior is an important contribution to one’s own health, which can be done very well in one’s own responsibility, especially in the home environment. *Prevention* includes using guidelines, reporting outcomes in diaries, and organizing appointments for training and advice as well as for transport to or from a health care facility providing support.

3.3.1.4 Arrange Appointments

Either because of a self-made diagnosis or on the basis of a physician’s diagnosis and order, it may become necessary to visit a certain health care facility. This could be, for example, an initial consultation with the family physician, a referral to the radiologist for a radiological examination, or a visit to a physiotherapist. In any case, it is usually necessary to arrange an appointment beforehand. This can be very demanding for the patient if several facilities have to be contacted in order to obtain a timely appointment. Patients may use phones, physicians’ websites, or even specialized apps for this function.

3.3.1.5 Filling Physician’s Orders

A visit to a physician usually involves the physician recommending that the patient take certain measures. For example, they may propose that the patient see a specialist, receive a specific physiotherapeutic treatment, take a set of medicines in a specific order, or take preventive measures by changing unhealthy behavior. It is often a challenge for the patient to both remember these orders and organize their implementation. Compliance with the measures prescribed or recommended by a physician or therapist is referred to as adherence.

3.3.2 *Functions to Be Performed by Health care Professionals and Other Staff in Health care Facilities*

3.3.2.1 Patient Care

Patient care in a facility begins with the patients' admission and ends with their discharge and any necessary transfer to another facility. During the patients' stay at the facility, decisions must be made about the diagnostics and therapy to be executed, and appropriate procedures must be ordered. The data generated during treatment must be coded so that it can be further processed.

Patient Admission

The prerequisite for the treatment of a sick person by a health care professional in a health care facility is the admission of the person to that facility. *Patient admission* (short: admission) aims at recording and distributing the patient demographic and insurance data as well as data on the medical and nursing history (Sect. 3.2.3.2). In addition, each patient must be identified correctly and a unique patient and *case identification number* (CIN) must be assigned.

This function can be decomposed as follows (Fig. 2.4 in Sect. 2.14.1):

Appointment Scheduling

The health care facility must schedule an appointment for the patient's visit. Appointments must also be made in connection with *order entry* for diagnostic services in a radiology department, for example.

In addition, unplanned *patient admissions* must be possible (e.g., in case of emergencies).

Patient Identification

Before health care professionals treat a patient, they must be sure exactly who the person they are treating is. Patients will normally identify themselves or with the assistance of relatives through a health insurance card or identity card. Based on such documents, the health care professional or administrative staff of the respective health care facility assigns a unique PIN to the patient. A new PIN is assigned only if the patient is in the facility for the first time. If patients have already been in the facility, they must be identified as recurrent, and previously documented information must be made available (such as previous diagnoses and therapies). Hospitals, in particular, must be able to distinguish a patient's different cases or hospital stays. Therefore, in addition to the PIN, a *case identification number* (CIN) is usually assigned as part of *administrative admission*.

Merging all health-related data and documents of a particular patient can only succeed if all health care professionals and all health care facilities caring for the patient use the same PIN. If each facility assigns its own PIN to a patient, a tool is needed to map the different PINs of one patient to each other. Such an application system is called a *master patient index (MPI)* and will be discussed in Sect. 3.4.1. Some countries provide a single unique PIN for every citizen, for example, on the health insurance card.

Patient identification as a function therefore interprets the entity type “patient” by considering, for example, their name, birthday, and data from the ID card and updates the same entity type by updating the PIN.

Administrative Admission

Administrative admission starts following *patient identification*. It creates the so-called case, being the aggregation of several contacts clustered according to specific clinical and/or organizational purposes of the facility. In case of inpatient treatment, a case summarizes the stay at the facility from *patient admission* until discharge. Each case is uniquely identified by its CIN. Important administrative data, such as insurance data or details about booked services, patient’s relatives, admitting physician, and transfer diagnoses, must be recorded. The patient is assigned to a certain area, for example, a ward and a bed. Some of the administrative data must be available to other functions through the help of certain organizational media (such as adhesive labels and smart cards). Administrative data form the backbone of information processing in a health care facility. In case of changes, patient data must be maintained and communicated. If the referring physician, for example, the GP, has communicated relevant information (e.g., previous laboratory findings), this information must be sent to the responsible physician in the admitting facility. In hospitals, *administrative admission* is usually done either in a central patient admission area by administrative staff or directly on the ward (e.g., during emergencies or on the weekend) by health care professionals. In medical offices, there is usually a reception desk where this function is performed.

Even in emergencies, *patient admission* is necessary. At the very least, *patient identification* must be performed in order to assign a proper PIN and CIN. In emergencies, a short version of *administrative admission* may be applicable. If the patient is unconscious and does not have an identity card, a dummy name may be recorded to provide PIN and CIN. If using PIN and CIN properly, there will be no problem to replace the dummy name by the correct name later on.

Medical Admission

At the first contact with the patient, the physician will carry out the *medical admission*. This typically comprises recording the patients’ medical history (disease history, systems review, social history, past medical history, family history, medication). Some of this information may be collected from documents of the referring physician which are taken to the facility by the patients themselves.

As a result of *medical admission*, the admission diagnosis must be stated and coded using the mandatory classification, for example, ICD-11.

The medical history must be made available for other functions by including it with the patient's health record. However, some facilities only have access to patient-related documents and data that originated from the same facility, i.e., the facility's patient record.

Nursing Admission

Especially in the case of inpatient treatment in a hospital, nursing home, or rehabilitation center, the nurse in charge will proceed with the *nursing admission* at the ward. This typically comprises introducing the patient to the ward and recording the nursing history. Administrative data and the reason for hospitalization are already at the nurse's disposal. The nursing history contains information about the current diagnosis and therapy, orientation, communication ability, social contacts, nutrition, mobility, personal hygiene, and vital signs. Computer-based or department-specific, (semi-) standardized data entry forms may be available to collect the data. The collected data must be made available for the whole stay by including it in the patient's health record.

Visitor and Information Service

In any health care facility that provides inpatient care, administration must have a good overview of the recent bed occupation, i.e., about the patients' stay at the hospital. This is, for example, important for the clerks at the information desk, who must be able to inform relatives and visitors correctly, as well as for some general administration statistics.

Decision-Making, Planning, and Organization of Patient Treatment

The treatment of the patient requires the execution of a variety of diagnostic and therapeutic procedures. Health care professionals must decide with the patient which procedures to execute based on available data and knowledge. Then these procedures must be carefully planned and initiated. This process is repeated each time new information and knowledge is available.

This function can be decomposed as follows (Fig. 3.5):

Decision-Making and Patient Information

Health care professionals must decide on the next steps to take, for example, certain diagnostic or therapeutic procedures. Depending on the complexity of a diagnostic or therapeutic decision, they should be able to consult internal or external experts

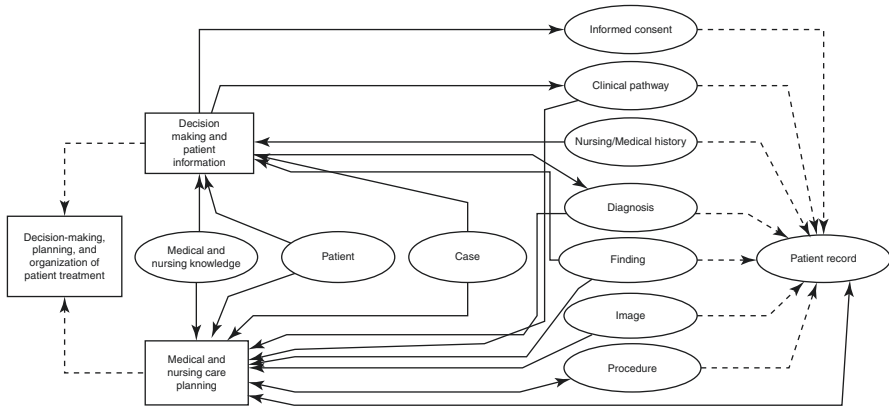


Fig. 3.5 Function *decision-making, planning, and organization of patient treatment*, its subfunctions and interpreted and updated entity types (for the legend refer to Sect. 2.14.1)

(e.g., in specialized hospitals) to obtain a second opinion. In this context, (tele)conferences may help. Health care professionals must be able to access all relevant patient data specific to a situation in addition to general medical and nursing knowledge (e.g., guidelines, recent study results and standards). Medication prescription may be supported by providing knowledge about adverse drug events. Decisions about clinical procedures must be documented. Patients must be involved in the decision-making process, the consequences of the planned diagnostic or therapeutic procedures should be explained, and their informed consent must be documented. Decision-making is a permanent function that is triggered by new information about the patient and the availability of new knowledge.

Medical and Nursing Care Planning

The next steps must be planned in detail. For each medical procedure (such as a radiological examination, an operation, or a chemotherapeutic treatment) as well as for each nursing procedure, the type, extent, duration, and responsible person and facility have to be determined. In nursing, care planning is documented in nursing care plans containing nursing problems, nursing goals, and planned nursing procedures. If necessary, other health care professionals are ordered to execute the planned procedures (e.g., a physician’s medical bandaging orders to be executed by a nurse or home care service at the patients’ home).

Care planning in cancer treatment is often performed by tumor board reviews. This means that a number of physicians who are experts in different specialties (disciplines) review and discuss the patient’s medical condition and treatment options.

Order Entry

Diagnostic and therapeutic procedures must often be ordered at specialized service units (e.g., laboratory, radiology, or pathology). These units may be an internal part of the health care facility of the treating health care professional or may be an external institution. The units execute the ordered procedures and communicate the findings or results back to the ordering facility. In order to avoid mix-ups, all units and facilities involved must use the same PIN. This is especially challenging when the service units do not belong to the same facility.

This function can be decomposed as follows (Fig. 3.6):

Preparation of an Order

If the order is to examine the patient's tissue or fluid samples, the specimen must be unambiguously assigned to a patient and then submitted (e.g., blood sample). Depending on the available service spectrum offered by a service unit, which may be presented in the form of service catalogs, the health care professional selects the appropriate service on an order entry form. Patient identification data (name, PIN) are documented together with relevant information such as recent diagnoses, relevant questions, service ordered (e.g., lab test), and other comments (e.g., on special risks).

If a medication is ordered by a physician and computer-based tools for *order entry* are used, computerized decision support systems could alert the physician, for example, in case of medication errors when a medication is ordered to which the patient is allergic.

The order may only be initiated by authorized persons. The order must be transmitted quickly and correctly to the service unit or to the person who is to execute the

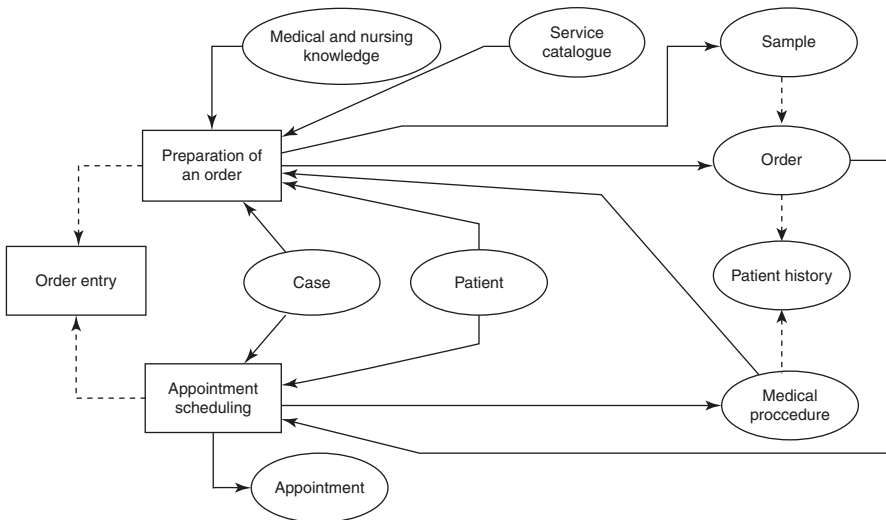


Fig. 3.6 Function “order entry,” its subfunctions and interpreted and updated entity types (for the legend refer to Sect. 2.14.1)

order. If a specimen is transferred, it must be guaranteed that the order and the specimen can be linked to each other at the service unit. It should be possible for the ordering health care professional to modify orders that have already been transferred, if necessary.

Appointment Scheduling

The patient's appointments must be scheduled (e.g., in radiological units) depending on the type of order. During scheduling, the demands of all parties (e.g., ordering physician, service unit, patient, transport unit, public transport) must be fairly balanced. This can be particularly challenging in the context of outpatient care if, for example, a suitable radiologist must first be found near the patient's place of residence and possible examination dates must be determined. It can also be complicated to have the patient transported to the radiologist. Depending on the health care system, patients may have to handle these tasks themselves and it becomes their function (Sect. 3.3.1).

Execution of Diagnostic, Therapeutic, and Nursing Procedures

The planned diagnostic, therapeutic, or nursing procedures (such as operations, radiotherapy, radiological examinations, medication) must be executed. Adequate tools and resources (e.g., staff, room, equipment) for the execution of the necessary procedures have to be available. This must be managed according to the special needs of the respective facility and health care setting (Chap. 6).

It is important that changes in care planning that may be due to new findings are promptly communicated to all involved units, facilities, and persons, enabling them to adapt to the new situation. All clinically relevant patient data (such as vital signs, orders, results, decisions) must be recorded as completely, correctly, and quickly as necessary. This supports the coordination of patient treatment among all persons and facilities involved and provides the legal justification for the actions taken. In inpatient care, a lot of these data may be recorded, for example, on the ward. In outpatient care, however, with the health care professionals caring for the patient at home, much of the data must also be collected and documented at the patient's home.

As far as possible and whenever sensible, data and metadata should be recorded and represented in a standardized manner (compare 3.2.2) to allow for seamless care across providers, for data aggregation and statistics, for computerized decision support, and for data retrieval. It is important that data can be linked by PIN and CIN even when data originate from different areas (such as ward, service unit, outpatient unit, home). The health care professionals and their facility must usually fulfill a lot of different legal reporting (such as for epidemiological registers) and documentation requirements. The items to be documented depend partly on the documenting unit and the documenting health care professional group (such as documentation by physicians or nurses, in outpatient units, in operation rooms, or in patients' homes). Clinical information should also be available for other functions such as *financial accounting, controlling, or quality management*.

This function can be decomposed as follows (Fig. 3.7):

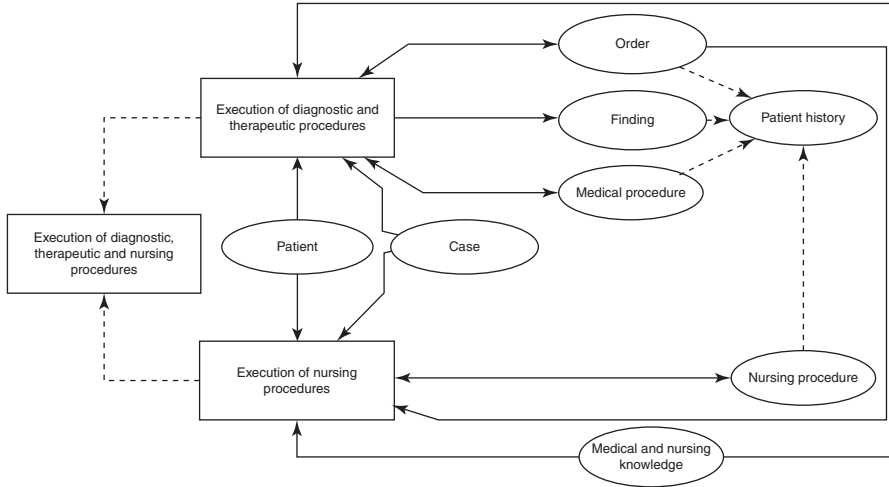


Fig. 3.7 Function *execution of diagnostic, therapeutic, and nursing procedures*, its subfunctions and interpreted and updated entity types (for the legend refer to Sect. 2.14.1)

Execution of Diagnostic and Therapeutic Procedures

The planned diagnostic and therapeutic procedures must be executed. All procedures must be documented. Findings and reports must be transmitted (as quickly as necessary) back to the ordering unit and presented to the responsible health care professional. They must be unambiguously assigned to the correct patient. The responsible physician should be informed about new results, and critical findings should be highlighted.

The function *execution of diagnostic and therapeutic procedures* can be specialized to:

- *execution of operations,*
- *execution of intensive care,*
- *execution of irradiation,*
- *execution of chemotherapy,*
- *execution of radiological examinations,*
- *execution of lab examinations,*
- *execution of prophylaxis and medication.*

Execution of Nursing Procedures

The planned nursing procedures (concerning medication, excretion, decubitus, hair and nail care, skin care, wound treatment, body washing, oral and dental care, nutrition and liquid balance, thrombosis) are executed. All *patient care*

procedures, their impact on the patient’s health status, and changes to the care plan must be documented. The responsible physician must be informed of any facts relevant to the therapy.

Coding of Diagnoses and Procedures

The health care facility must be able to document and code all stated diagnoses and all executed medical procedures in a correct, complete, quick, and patient-oriented way. These data form the basis for billing, for reports, and for research. Diagnoses and medical procedures are also used for *controlling*. In addition, legal requirements stipulate that some of the data must be documented and communicated. This data is also important for medical research.

Diagnoses and medical procedures are recorded and coded in a standardized way (e.g., using the ICD11 for diagnoses codes [4]) and then processed. Diagnoses and medical procedures are at least partly derivable from clinical documentation. To support their documentation, adequate coding catalogs must be offered and maintained. Tools are needed that help health care professionals to quickly find the correct codes for the diagnoses and medical procedures.

The function *coding of diagnoses and procedures*, its decomposition in subfunctions, and the entity types to be interpreted and updated are summarized in Fig. 3.8.

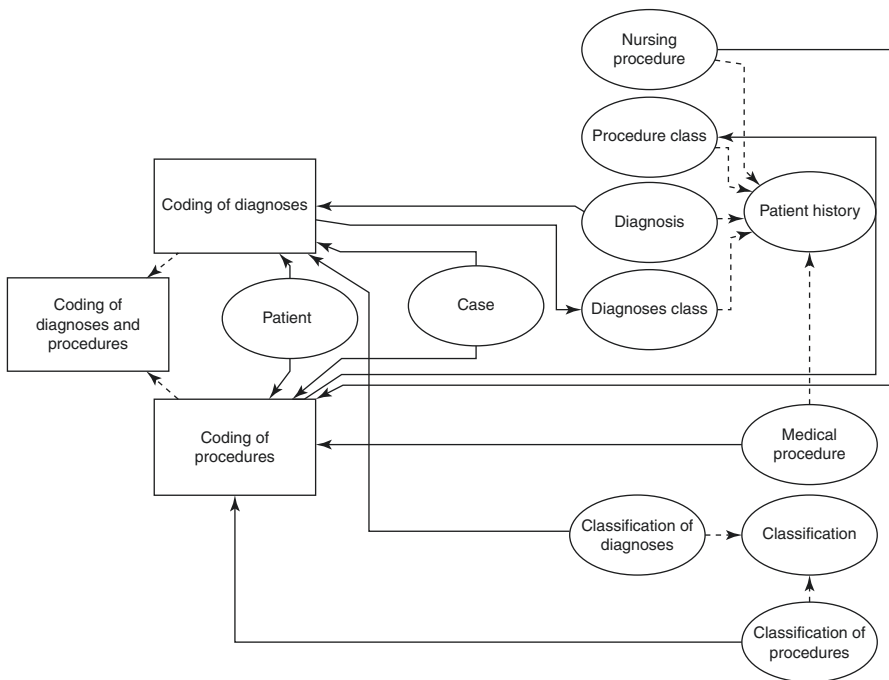


Fig. 3.8 Function *coding of diagnoses and procedures*, its subfunctions and interpreted and updated entity types (for the legend refer to Sect. 2.14.1)

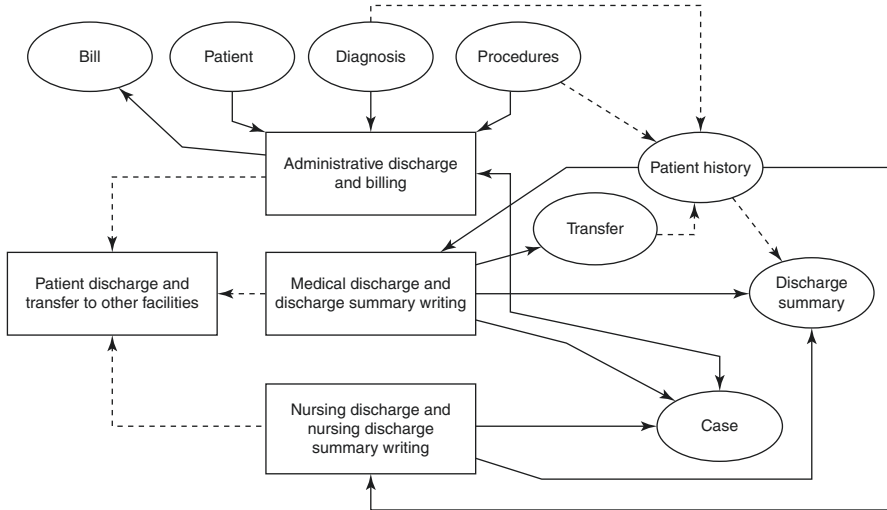


Fig. 3.9 Function *patient discharge and transfer to other facilities*, its subfunctions and interpreted and updated entity types (for the legend refer to Sect. 2.14.1)

Patient Discharge and Transfer to Other Facilities

In case of inpatient care in a health care facility after the patient's care has been completed, the patient is discharged and sometimes referred to other facilities (e.g., GPs or rehabilitation centers). *Patient discharge and transfer to other facilities* (short: discharge) covers *administrative, medical, and nursing discharge*.

This function can be decomposed as follows (Fig. 3.9):

Administrative Discharge and Billing

The process of administrative patient discharge initiates final billing and the fulfillment of legal reporting requirements (e.g., statistics on diagnoses and procedures). In recent years, diagnosis-related group (DRG) systems have been introduced for patient billing in most countries. That means that bills for patient treatment are no longer calculated based on daily rates but on the DRG in which a patient case was classified. Diagnoses, procedures, patient's age, and other criteria serve as an input for the calculation of a DRG.

Medical Discharge and Medical Discharge Summary Writing

Medical discharge entails the completion of the documentation and the writing of a discharge summary by the attending physician. The discharge summary includes the relevant diagnoses, important findings, therapeutic procedures, current patient state, and recommendations for further treatment. The hospital must be able to transmit this and other information (e.g., radiological images) to other facilities as

quickly as possible. To speed up this process, a short report (i.e., physician's discharge letter) is often immediately communicated to the next facility containing, for example, the diagnoses and therapeutic treatments. It is then later followed by a more detailed report.

Nursing Discharge and Nursing Discharge Summary Writing

Nursing discharge entails the completion of the documentation and the writing of a nursing discharge summary by the attending nurse. The nursing discharge summary comprises, for example, information about activity level, diet, and wound care.

3.3.2.2 Supply and Disposal Management, Scheduling, and Resource Allocation

The health care facility must offer sufficient and well-organized resources for *patient care*. This is true for wards (ward management), outpatient units (outpatient management), and service units (department management). Efficient process organization is extremely important, for example, in outpatient units or service units, and can be supported, for example, by providing working lists for individual staff members by issuing reminders about appointments or by visualizing actual process flow. The facility's information system must be able to support communication between all persons involved in *patient care*. This comprises synchronous (e.g., telephone) and asynchronous (e.g., blackboards, brochures, email) communication. Staff members must be able to be contacted within a prescribed period of time.

Supply and Disposal Management

Supply and disposal of materials, food, drugs, and so on must be guaranteed. All of a facility's departments should be able to order from up-to-date catalogs. The corresponding service units (stock, pharmacy, and kitchen) must be able to deliver correctly and on time.

Supply and disposal management can be decomposed as follows (Fig. 3.10):

- *Catering*

According to their health status, patients have different nutritional needs. It must be ensured that the patients are provided with the right dietary food at the right time.
- *Material and medication management*

Nurses and physicians must be able to anticipate shortages of material such as medical strips, bandages, or needles to order new material from a central supplier in time.

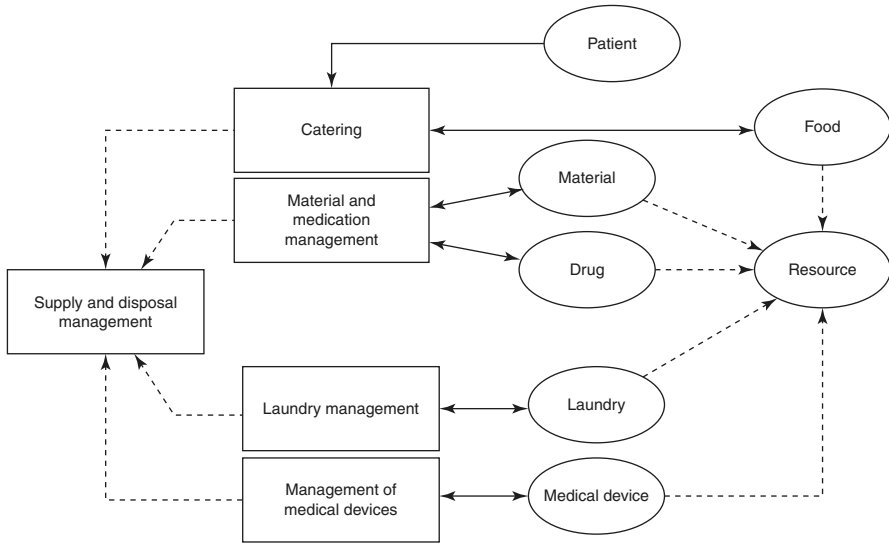


Fig. 3.10 Function *supply and disposal management*, its subfunctions and interpreted and updated entity types (for the legend refer to Sect. 2.14.1)

- *Laundry management*

The hospital must permanently be supplied with linen, towels, sterile scrubs, surgical masks, etc.

- *Management of medical devices*

In addition to other resources, medical devices must be registered and maintained according to legislation. Due maintenance must be organized, documented, and completed.

Scheduling and Resource Allocation

Various resources are needed for *patient care* in health care facilities. The function *scheduling and resource allocation* comprises staff planning, bed planning, room planning, and device planning. All resource planning activities must be coordinated. When procedures are scheduled, the demands of both the service unit and the ordering unit with regard to scheduling the appointment must be considered. Request, reservation, confirmation, notification, postponement, and cancellation must be supported. All involved staff members and patients should be informed about the appointments. Postponements and cancellations should be communicated to all involved persons in time.

This function can be decomposed into the functions *appointment scheduling*, *scheduling and resource planning with the medical service unit*, and *scheduling and resource planning with the patient transport service* (Fig. 3.11). *Appointment scheduling* was also listed as a subfunction of *patient admission*.

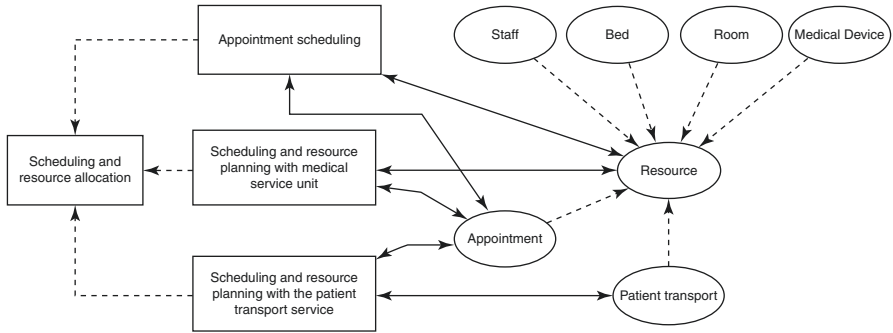


Fig. 3.11 Function *scheduling and resource allocation*, its subfunctions and interpreted and updated entity types (for the legend refer to Sect. 2.14.1)

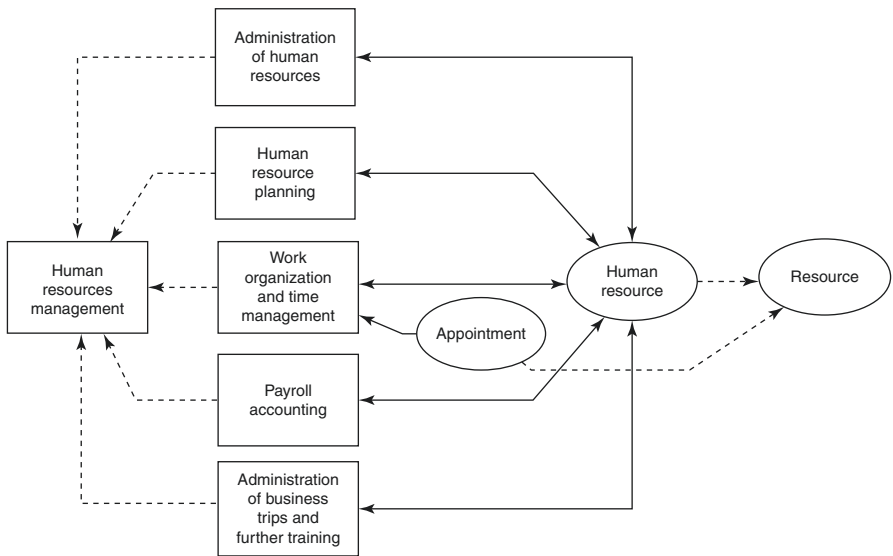


Fig. 3.12 Function *human resources management*, its subfunctions and interpreted and updated entity types (for the legend refer to Sect. 2.14.1)

Human Resources Management

This contains all tasks for the development and improvement of the productivity of staff. It comprises, for example, staff and position planning, the staff register, staff scheduling, and staff payroll.

This function can be decomposed as follows (Fig. 3.12):

- *administration of human resource master data,*
- *human resource planning,*

- *work organization and time management,*
- *payroll accounting,*
- *administration of business trips and further training.*

3.3.2.3 Administration of Health care Facilities

Administration of health care facilities supports the organization of *patient care* and guarantees the financial survival and the economic success of the hospital. Subfunctions are:

Patient Administration

Patient administration comprises the administrative tasks in a health care facility dealing more or less immediately with patients. Thus, it is an aggregation of the subfunctions

- *appointment scheduling,*
- *administrative admission,*
- *patient identification,*
- *visitor and information service,*
- *coding of diagnoses and procedures,*
- *administrative discharge and billing* (Fig. 3.13).

Archiving of Patient Information

Relevant data and documents containing patient information must be created, gathered, presented, and stored in such a way that they are efficiently retrievable during the whole process of patient treatment. These data and documents are primarily stored in patient records. A mixture of paper-based and computer-based patient records is still used today. Certain legal requirements usually must be considered.

This function can be decomposed as follows (Fig. 3.14):

- *Opening of a patient record*
Administrative admission triggers the *opening of a patient record*. The patient record may be electronic or paper-based or a mixture of both. Standards for document filing formats must be established and used.
- *Administration and allocation of patient records*
 A paper-based archive must be able to manage patient records and make them available upon request within a defined time frame. The exact location of each record should be known (e.g., in which archive, on which shelf). Robot systems may store and gather paper-based records automatically. Lending and return of records (e.g., for patients coming for multiple visits) must be organized, while respecting different access rights that depend on the role of the health care professionals in the process of *patient care*.

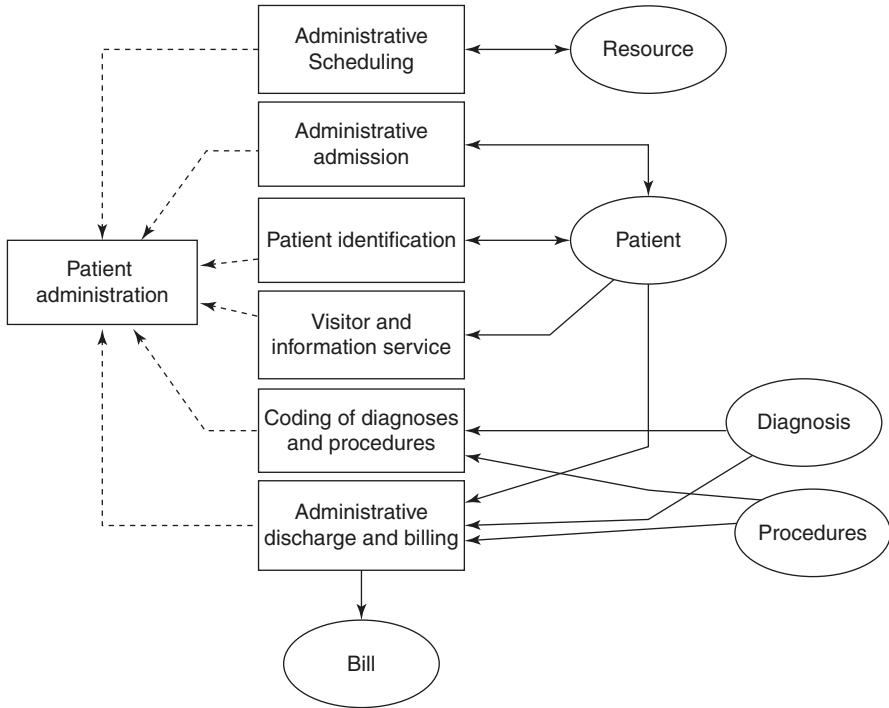


Fig. 3.13 Function *patient administration*, its subfunctions and interpreted and updated entity types (for the legend refer to Sect. 2.14.1)

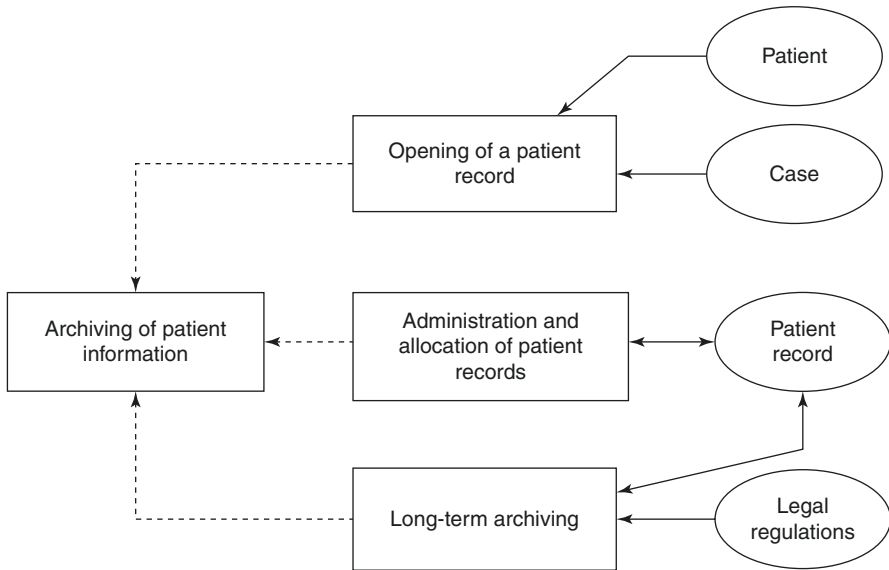


Fig. 3.14 Function *archiving of patient information*, its subfunctions and interpreted and updated entity types (for the legend refer to Sect. 2.14.1)

- *Long-term archiving*

After discharge of the patient, patient records must be archived for a long time (e.g., for 10 or 30 years, depending on the legal regulations). The archive must offer enough space to allow the long-term storage of patient records. Their authenticity and correctness can be proven more easily, for example in case of legal action, if they are archived in accordance with legal regulations.

Quality Management

Quality management comprises all activities of a health care facility’s administration to assure and continuously improve the quality of *patient care*. This includes setting goals, defining responsibilities, and establishing and monitoring the processes to realize these goals. This covers, for example, internal reporting containing quality indices. *Quality management* requires information about patients and treatments as well as knowledge about diagnostic and therapeutic standards.

This function can be decomposed as follows (Fig. 3.15):

- *Internal quality management*

Internal quality management assures a defined quality of all processes and outcomes of the hospital. An internal reporting system, which presents quality-related indicators, is also covered. Medical, nursing, and administrative guide-

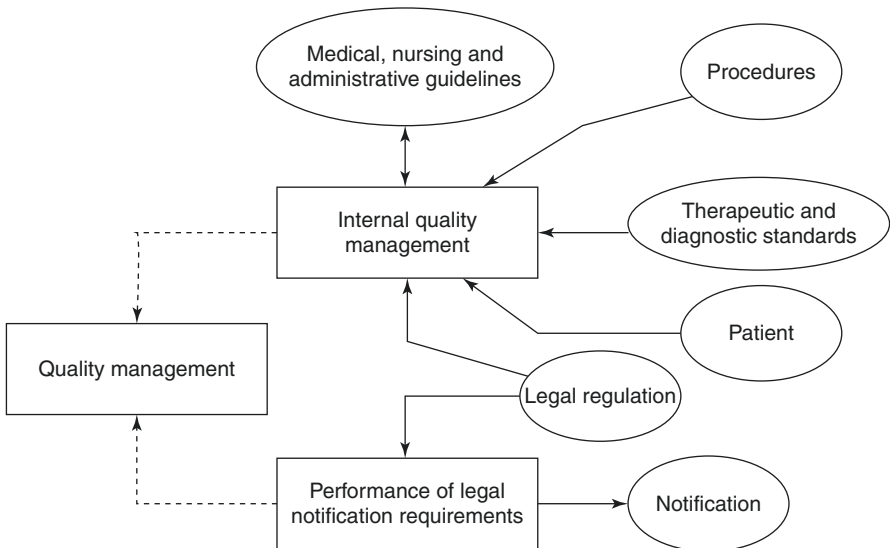


Fig. 3.15 Function *quality management*, its subfunctions and interpreted and updated entity types (for the legend refer to Sect. 2.14.1)

lines may be defined, stored, and presented. There exists a structured complaint management.

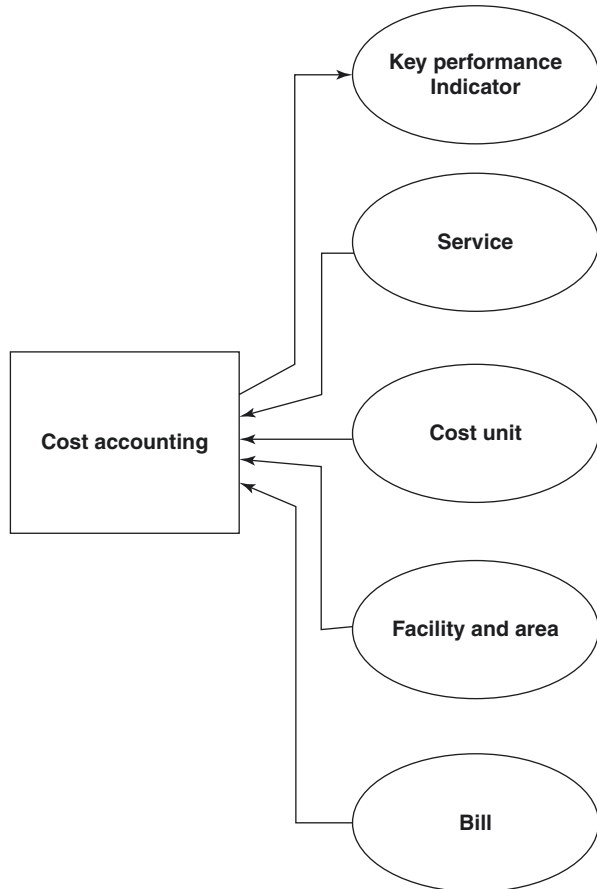
- *Performance of legal notification requirements*

Legal notification requirements for quality assurance must be completed.

Cost Accounting

For controlling purposes, it is necessary to keep track of services, their costs, and who has received the services. *Cost accounting* usually investigates which costs incur (cost-type accounting), where costs incur (cost center accounting), and for what activities or services costs incur (cost unit accounting). According to the accounting purpose, the time period to be observed and the scope of the costs to be accounted have to be defined. The results of *cost accounting*, i.e., key performance indicators (KPI), serve as input for *controlling* (Figs. 3.16 and 3.17).

Fig. 3.16 Function *cost accounting*, its subfunctions and interpreted and updated entity types (for the legend refer to Sect. 2.14.1)



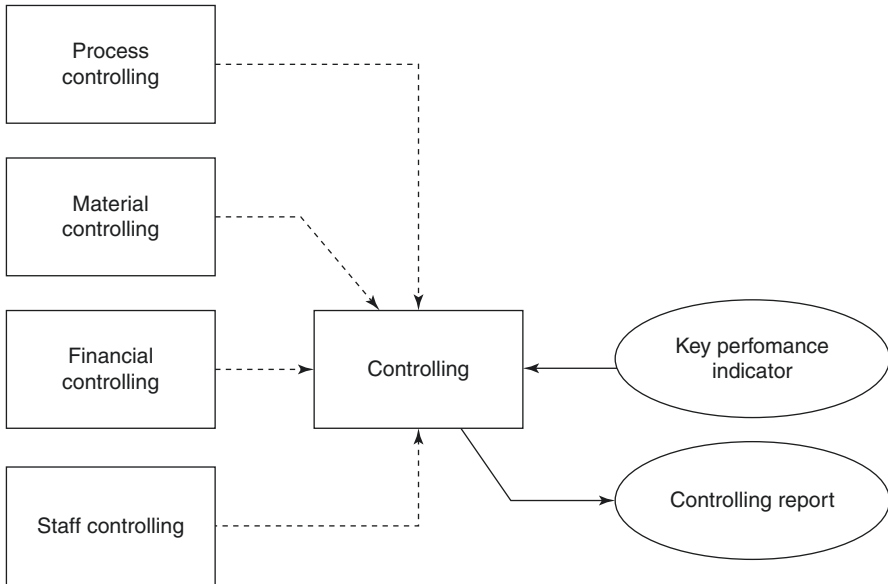


Fig. 3.17 Function *controlling*, its subfunctions and interpreted and updated entity types (for the legend refer to Sect. 2.14.1)

Controlling

The health care facility must be able to gather and aggregate data on its operation in order to control and optimize it. This covers, for example, *staff controlling*, *process controlling*, *material controlling*, and *financial controlling*. In hospitals, for example, the number of patient cases, the length of patients' stays in the hospital, and the case mix index, which is calculated from the patients' DRGs, are KPIs serving as input for controlling reports (Fig. 3.17).

Financial Accounting

All the facility's financial operations must be systematically recorded to meet legal requirements. *Financial accounting* comprises, for example, *debtor accounting*, *credit accounting*, and *asset accounting*. This type of accounting requires information from bills and creates new values for KPIs (Fig. 3.18). The health care facility must support general statistical analysis, for example, calculation and analysis of economic data.

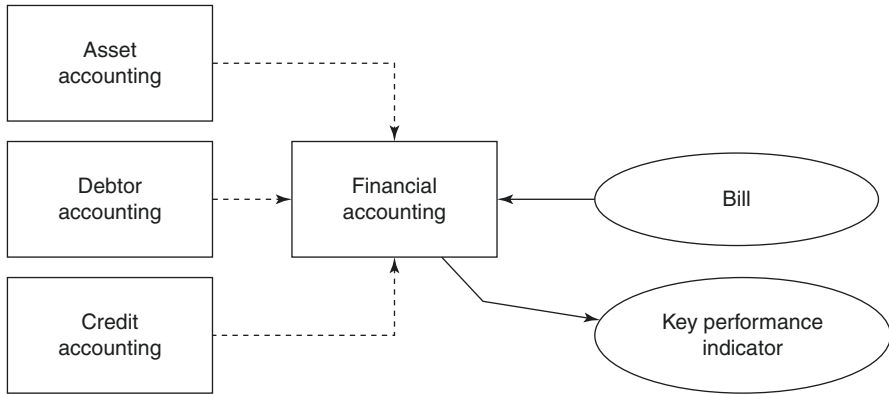
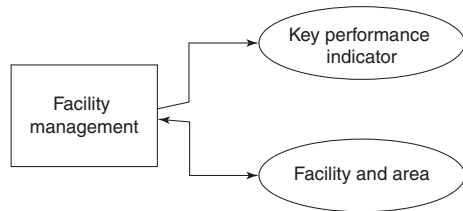


Fig. 3.18 Function *financial accounting*, its subfunctions and interpreted and updated entity types (for the legend refer to Sect. 2.14.1)

Fig. 3.19 Function *facility management* and interpreted and updated entity types (for the legend refer to Sect. 2.14.1)



Facility Management

The management of buildings, areas, and utilities of the health care facility is usually subsumed by the term *facility management*. As this often also involves high costs, this management area also influences KPIs (Fig. 3.19).

Management of Information Systems¹

Management of information systems plans the information system of an enterprise and its architecture, directs its establishment and its operation, and monitors its development and operation with respect to the planned objectives. Different management levels have different perceptions and interests.

¹Now we have come full circle. Our book deals with information management and especially strategic information management. Since information systems are the subject of information management, and information systems are designed to support all necessary functions of an enterprise, they should support information management as well. For a more thorough explanation of strategic information management, refer to Sect. 4.3.

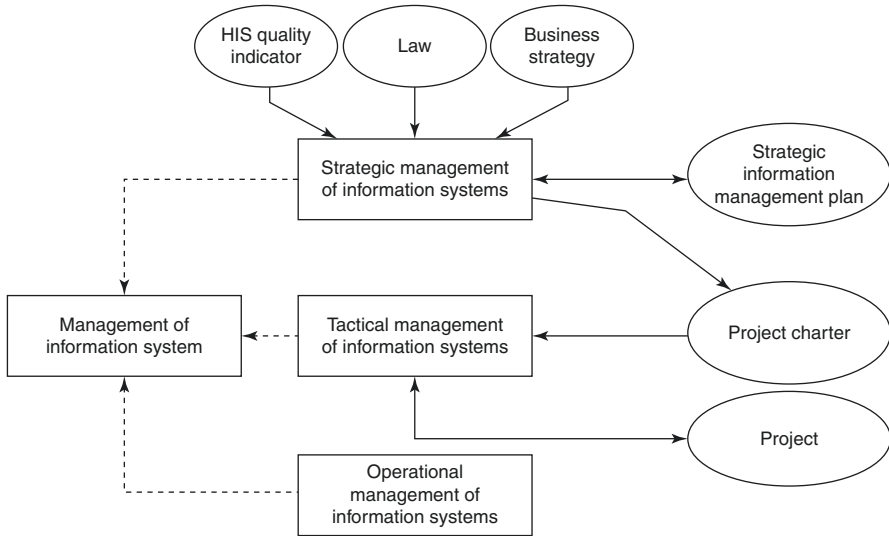


Fig. 3.20 Function *management of information systems*, its subfunctions and interpreted and updated entity types (for the legend refer to Sect. 2.14.1)

This function can be decomposed as follows (Fig. 3.20):

- *Strategic management of information systems*
Strategic management of information systems deals with the enterprise's information processing as a whole and establishes strategies and principles for the evolution of the information system. An important result of strategic management activities is a *strategic information management plan* that is aligned with the hospital's business strategy. It includes the direction and strategy of information management and the architecture of the enterprise information system.
- *Tactical management of information systems*
Tactical management of information systems deals with particular functions or application components that are introduced, removed, or changed. Usually, these activities are done in the form of *projects*. Such tactical information management projects are initiated by *strategic management of information systems*. Thus, *strategic management of information systems* is a vital necessity for *tactical management of information systems*. The result of tactical information management projects is the information system.
- *Operational management of information systems*
Operational management of information systems is responsible for operating the components of the information system. It ensures its smooth operation in accordance with the strategic information management plan. Additionally, operational information management plans, directs, and monitors permanent *IT services* for the users of the information system.

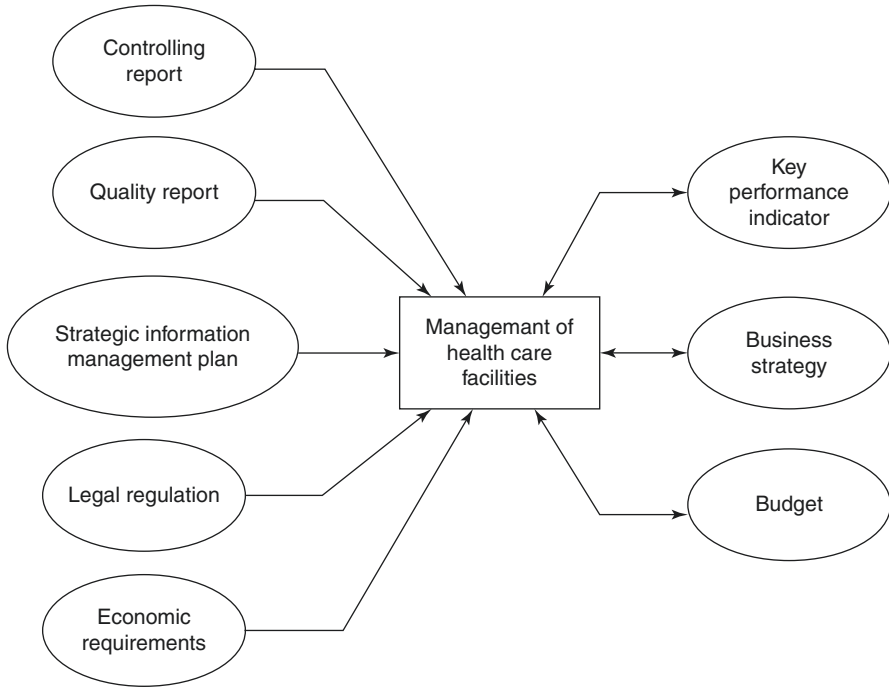


Fig. 3.21 Function *management of health care facilities*, its subfunctions and interpreted and updated entity types (for the legend refer to Sect. 2.14.1)

3.3.2.4 Management of Health care Facilities

Management of health care facilities decides on questions of fundamental importance for the health care facility’s development (goals, strategic decisions, personnel decisions and decisions about budget, investments, or key treatments). *Management of health care facilities* must focus on high quality of *patient care*, taking into account economic as well as legal and other requirements. Information needed and produced is shown in Fig. 3.21.

3.3.2.5 Clinical Documentation: A Function?

It may be surprising that “documentation” is not listed as a function in the previous sections. In fact, clinical documentation, which comprises medical documentation and nursing documentation, is a time-consuming and often unpopular duty of health care professionals. Moreover, every function described so far requires a lot of documentation. The results of diagnostic procedures have to be written down, medical histories have to be documented, and so on. Documentation takes place every time a function is performed, new information is generated, and respective data are stored

somehow somewhere. Thus, every introduced function which updates an entity type is a documenting function. In fact, documentation is a part of all functions updating an entity type, even of those performed by patients.

3.4 Logical Tool Layer: Types of Application Systems in Health care Facilities

After having looked at data to be processed and at functions to be performed we now describe tools for data, information, and knowledge processing at the logical tool layer of information systems, i.e., the application components supporting these functions and storing and processing the data.

Health care facilities still have non-computer-based application components and use paper as a physical tool. For example, parts of clinical documentation while performing functions are occasionally still done with paper-based patient records. Thus, despite the growing portion of electronic documents, the “paperless hospital” still seems to be a remote ideal today. There might be a continuing need for some paper-based documents. Typical application components that are still paper-based comprise, for example, the patient chart, the patient record, and clinical text books and knowledge sources.

In spite of this still existing significance of non-computer-based information processing in health care, we will focus here on application systems, i.e., the computer-based application components.

In Sect. 3.3.2, we could see that many functions are commonly found in all health care facilities. Although they use different application systems to support the functions, there are typical application systems to support health care professionals in health care facilities. We take a closer look at the respective types in the following subsections. In Sect. 6.8, however, we will look at types of application components to support patients and informal caregivers.

We will sum up the characteristics of the application systems in tables enumerating the supported functions; please refer to Sect. 3.3 for their detailed descriptions. For every function, we also list typical features that the application system should offer. A feature is a functionality offered by the application system’s software which directly contributes to the fulfillment of one or more functions (Sect. 2.9).

Please note that we refer to some of the application system types as “XYZ information system” (e.g., *laboratory information system*). Although this sounds contradictory to our definition of the term “information system,” we did so in order to integrate the popular names and hope you will not be confused.

In this section, you will get to know the types of application systems that are used in health care. In later Sects. 3.5–3.9, we will explain step by step how and under which conditions the application systems described here can be assembled, i.e., integrated, in the information system. Finally, in Sect. 3.10, we will discuss the

physical data processing systems on which the application systems need to be installed. Finally, in Chap. 6, we will take a closer look at specialized application systems in specific health care settings.

3.4.1 Patient Administration Systems

Whether hospital, medical office, or rehabilitation center, a health care facility needs to administer patients. *Patient administration* comprises the functions from Sect. 3.3.2.3.1 as listed in Table 3.1.

Application systems supporting *patient administration* are referred to as *patient administration systems* or sometimes “patient management systems.” They must provide correct, complete, and up-to-date administrative patient data for all other application components. In addition, all other application components must be able to transmit relevant administrative patient data (e.g., diagnoses) to the *patient administration system*. Therefore, the *patient administration system* can be regarded as the center of the administrative memory of the facility’s information system.

Table 3.1 Set of functions and related features typically to be supported by *patient administration systems*

Function (from Sect. 3.3.2)	Typical application component features
<i>Appointment scheduling</i>	Provide means for scheduling patients’ appointments Provide means for ordering transport services
<i>Administrative admission</i>	Provide forms for entering or updating patient administrative information (name, address, birthdate, relatives, admission diagnosis, etc.) Merge patient information from two records Provide means for ordering patient transfer within the facility Admit patients to the ward or outpatient unit For inpatient care: assign patients to rooms and beds Provide means for preparing facility-wide statistics
<i>Patient identification</i>	Retrieve patient information from database Generate a unique PIN and a CIN Administrate the transinstitutional PIN
<i>Visitor and information service</i>	Provide relatives with information on the location of a patient
<i>Coding of diagnoses and procedures</i>	Provide catalogs and other means for coding patients’ diagnoses Provide catalogs and other means for coding patient-related procedures Provide means for verifying codings done in departments Provide forms for preparing a bill for the patient insurance
<i>Administrative discharge and billing</i>	Provide means for initiation of final billing for inpatients and outpatients Provide reminder for fulfilling of legal reporting requirements

During *patient admission*, *patient administration systems* must support the retrieval of patient data (e.g., by name or birthdate) to avoid duplicate and erroneous registration of patients. The resulting identification numbers PIN and CIN are of utmost importance for the whole information system. They are the basis for correct assignment of patient-related data to patients and thus are the very precondition for a valid patient record—regardless if it is electronic or not. Without correct PIN and CIN, all the high-tech of a modern information system would be useless.

Application systems providing a correct PIN are often called *MPI*. Hence, a *patient administration system* is an *MPI*. It is essential to have exactly one *MPI* in a health information system—regardless whether it is an institutional health information system where the *MPI* provides the PIN or a transinstitutional health information system (THIS) where the *MPI* provides the transinstitutional PIN and, if necessary, cross-references the institutional PINs.

Both PIN and CIN as well as the related patient information must be made available to other application components of the facility. If the patient already has a paper-based patient record, the application component should automatically trigger the transfer of this record from the patient record archive to the unit the patient has been admitted to.

In addition, *patient administration systems* may update patient information in case of changes or merge patient information from two cases if a new PIN was wrongly assigned to a patient.

Patient administration systems are usually not only used by administrative staff but also by nurses and doctors at the ward. In the latter case, they also support daily management activities by health care professionals that occur on a ward. In particular, they support the assignment of patients to beds and rooms with the features shown in Table 3.1.

Some countries have different regulations for inpatient and outpatient billing. Particular features may therefore be needed for billing in outpatient departments and medical offices.

3.4.2 Medical Documentation and Management Systems (MDMS)

In addition to a *patient administration system* to support the management of patients, health care facilities need another application system to support functions related more closely to organizing and documenting patients' diagnostics and treatment. These functions are summarized in Table 3.2.

Application systems designed to especially support the functions in Table 3.2 are called *medical documentation and management systems (MDMS)*. They typically contain specialized modules for different medical fields (e.g., ophthalmology, psychiatry, dermatology) and usually offer generic forms for free text, semi-structured, or structured (e.g., drop-down lists) data entry for medical

Table 3.2 Set of functions and related features typically to be supported by *medical documentation and management systems*

Function (from Sect. 3.3.2)	Typical features
<i>Medical admission</i>	Provide forms for documenting medical history Provide forms for documenting diagnosis Scan documents from referring physician and other sources
<i>Decision-making, planning, and organization of patient treatment</i>	Provide forms for documenting patient's informed consent Provide forms for documenting planned tasks Provide guidelines for care planning Provide context-related medical knowledge
<i>Execution of diagnostic and therapeutic procedures</i>	Provide forms for entering clinical data (free text or structured) Provide forms for preparing findings
<i>Medical discharge and medical discharge summary writing</i>	Provide means for writing the discharge summary, for example, by collecting and presenting available data which should be compiled for the summary Provide means for finalizing documentation
<i>Human resources management</i>	Provide means for managing staff Create a roster Assign doctors to patients or rooms
<i>Coding of diagnoses and procedures</i>	Algorithm for finding correct codes for diagnose and procedures given in natural language

documentation as well as support for speech recognition, reporting, and analysis features. The more data are structured, the easier patient-related computer-based decision support and statistical data analysis are. It is important that users are able to adapt the features to their needs (e.g., by defining which items have to be documented and which constraints the entered data must meet). When reports are generated, the reuse of already-documented data (e.g., diagnoses, findings from radiology, or lab) should be supported.

Besides clinical documentation, the *coding of diagnoses and procedures* is very important. Coding components must support the easy search for suitable diagnosis and procedure classes and their respective codes in classifications for a given medical field. Alternatively, free text can be analyzed using natural language recognition methods. If these coding components are separate from the *MDMS*, it must be guaranteed that the codes can be transferred to *MDMS*. The *MDMS* should also allow an adequate layout of diverse reports. When several persons are involved in the creation of a report (e.g., discharge summaries may be dictated by a junior physician, written by a secretary, and approved by a senior physician), the application components should support the management and distribution of different versions of a document having different status (e.g., preliminary or approved).

Medical documentation is the basis for decision-making, planning, and organization of patient treatment. The *MDMS* must therefore support the medical staff by providing medical knowledge, which should be preselected using documented data about the patient's conditions. Ideally, a respective "infobutton" [5] should be implemented.

3.4.3 *Nursing Management and Documentation Systems (NMDS)*

Although patient treatment and *patient care* or nursing are inherently intertwined, these two areas are often considered separately, even in information systems. Corresponding to the areas of responsibility of physicians and nurses, a distinction is also made, for example, between medical informatics and nursing informatics. Unfortunately, this often also results in other application systems being used to support the nursing process, even though doctors and nurses need to work together particularly closely.

The nursing process comprises *nursing admission*, nursing care planning with definition of problems, formulation of nursing aims, and planning of nursing tasks, then *execution of nursing procedures* and *nursing discharge and nursing discharge summary writing*. Like medical diagnoses and procedures, nursing diagnoses and procedures must be coded. The working hours of nursing staff, who usually work in shifts, must be carefully managed. These functions, together with related features, are listed in Table 3.3.

Application systems designed to especially support the functions in Table 3.3 are referred to as *nursing management and documentation systems (NMDS)* or sometimes “nursing information system.”

Table 3.3 Set of functions and related features typically to be supported by *nursing management and documentation systems*

Function (from Sect. 3.3.2)	Typical features
<i>Nursing admission</i>	Provide forms for documenting the nursing history
<i>Medical and nursing care planning</i>	Support creation of a nursing care plan Provide forms for documenting diagnosis and problems Provide forms for documenting nursing aims Provide forms for documenting nursing tasks
<i>Execution of nursing procedures</i>	Provide forms for documenting performed tasks Provide forms for documenting the outcome of nursing tasks
<i>Coding of diagnoses and procedures</i>	Provide catalogs and other means for coding of nursing diagnosis Provide catalogs and other means for coding of nursing procedures
<i>Nursing discharge and nursing discharge summary writing</i>	Provide forms for writing the nursing discharge summary Provide means for finalizing nursing documentation Communicate discharge information
<i>Human resources management</i>	Provide means for managing ward staff Provide means for creating a roster Assign nurses to patients or rooms

The *NMDS* must support the documentation of all steps of the nursing process. To support nursing care planning, the definition and use of predefined nursing care plans (comprising recent problems of the patient, nursing goals, and planned nursing tasks) is helpful.

The *NMDS* offers support for using predefined nursing terminologies and nursing classification such as NANDA, NIC, and NOC.²

3.4.4 Computerized Provider Order Entry Systems (CPOE)

The care of patients is a highly collaborative process. Not only are different facilities involved, but many service units as well as different health care professionals with different tasks are also involved within the facilities. The interaction of these units and people is handled by orders that one unit directs to another unit. This is summarized in the function *order entry*. *Order entry* can comprise both ordering of diagnostic or therapeutic procedures and ordering of drugs.

Application systems designed to especially support the function *order entry* and to provide the features in Table 3.4 are referred to as *computerized provider order entry systems (CPOE)* or sometimes *computerized physician order entry systems*. *CPOE systems* support formulation of the order, *appointment scheduling*, printing of labels, and the communication of the order to the service unit.

When ordering drugs, physicians may choose the most appropriate drug or generic drug from drug catalogs. The *CPOE system* may then also offer decision support functionalities such as dosage calculation, drug–drug interaction checks, drug allergy checks, or drug lab checks to prevent prescription errors.

When ordering diagnostic or therapeutic procedures, the results (e.g., lab values or X-ray report) must then be communicated back to the ordering facility. *CPOE systems* offer service catalogs that present the available service types of the different service units (e.g., laboratory, radiology, surgery). Order sets that describe a typical set of combined orders (e.g., a combination of diagnostic procedures that have to be performed in a given situation) can support the ordering process; they are activated and modified by the ordering clinician. Some *CPOE systems* support receiving and presenting findings. However, this is usually done by other application components such as the *radiology information system (RIS)* or *laboratory information system (LIS)*.

²NANDA = North American Nursing Diagnosis Association (the abbreviation is often used synonymously for the international classification of nursing diagnoses), NIC=Nursing Interventions Classification, NOC=Nursing Outcomes Classification.

Table 3.4 Set of functions and related features typically to be supported by *computerized provider order entry systems*

Function (from Sect. 3.3.2)	Typical features
<i>Order entry</i>	Provide orders for patient-related drugs Provide drug catalogs Provide means for minimizing adverse drug events Calculate dosage of drugs Provide orders for patient-related examinations Select orders from order sets Provide means for scheduling a patient’s appointment View patient-related appointments

3.4.5 Radiology Information Systems (RIS)

Facilities that handle the *execution of radiological examinations* exist as departments of hospitals, for example. There, they execute radiological examinations primarily on the inpatients of the respective hospital. Such facilities may also be legally and economically independent facilities, however, which then typically serve the care of outpatients, for example, from medical offices.

When radiological examinations are needed, wards or medical offices order them and schedule an appointment. In the case of outpatient care, however, the patients often have to perform the function *arrange appointments* themselves. The examinations may then be done using imaging devices, for example for computed tomography, magnetic resonance imaging, ultrasonography, or digital radiography. The imaging devices are called modalities. Based on the generated images, a specialist in radiology creates a report, which is then sent and presented (often together with selected pictures) to the ordering physician. The related functions are summarized in Table 3.5. Application systems designed to especially support the functions and to provide the features shown in Table 3.5 are usually called *radiology information systems (RIS)*. Please note that according to our definition, these application systems are not information systems, even if the name suggests that they are.

The *RIS* offers some features which are also provided by the *patient administration system* (3.4.1), i.e., features for registering patients, *appointment scheduling*, organization of examinations and staff (workflow management), provision of patient data and examination parameters, creation of radiology reports, documentation and coding of activities, and statistics. A special feature is the close connection to the modalities: The *RIS* typically provides a working list (i.e., patient name and requested examination) for the modalities and receives confirmation on the completion of a radiologic examination from the modalities. Due to these special features, software for *RIS* typically comes from specialized vendors.

Table 3.5 Set of functions and related features typically to be supported by *radiology information systems*

Function (from Sect. 3.3.2)	Typical features
<i>Execution of radiological examinations</i>	Receive orders Assign orders to modalities Provide forms for writing a report
<i>Medical admission</i>	Provide means for check-in of patient
<i>Appointment scheduling</i>	Assign patients to modalities Offer free appointments to health care professionals and patients
<i>Coding of diagnoses and procedures</i>	Provide catalogs and other means for coding radiological diagnoses Provide catalogs and other means for coding radiological procedures
<i>Management of medical devices</i>	Manage modalities
<i>Scheduling and resource allocation</i>	Provide means for preparing work schedules

3.4.6 *Picture Archiving and Communication Systems (PACS)*

Even if radiologists occasionally still produce images on film, the modalities generally produce digital images. Digital images are stored in *picture archiving and communication systems (PACS)*. These application components allow the storage, retrieval, management, manipulation, and presentation of large amounts of image data and their quick communication from the storage medium to the attached radiologists' workstations. Image data can also be sent from the *PACS* to the ordering departments or facilities in a teleradiology network, for example. Quick communication may require a prefetching strategy to retrieve image data from slower storage devices and to provide it to faster devices well in advance to the situation in which they will be needed.

Application software products for *PACS* also comprise means for image processing and 3D reconstruction. They are often offered by vendors, which also offer physical data processing systems such as storage, networks, and modalities.

Alternatively, *vendor-neutral archives (VNAs)* (Sect. 3.9.3) can be used to store image data and other patient-related documents. VNAs are connected to the *RIS*, the image processing components of the *PACS*, and other application systems using communication standards (especially DICOM). The use of the communication standards here follows the rules provided in the Integrating the Health care Enterprise (IHE) profiles for sharing documents (Sects. 3.7.2.4 and 3.7.2.6).

Obviously, *RIS* and *PACS* in one health care facility should be closely connected. They should also have a tight connection to the facility's *patient administration*

Table 3.6 Set of functions and related features typically to be supported by *picture archiving and communication systems*

Function (from Sect. 3.3.2)	Typical features
<i>Execution of radiological examinations</i>	Retrieve pictures Display pictures Modify presentation of pictures “Hang” pictures like in the analog world Communicate pictures Archive pictures Search for old pictures

system, MDMS, CPOE system, and the patient data management system (PDMS) in order to allow quick access to reports and imaging pictures from every unit.

Table 3.6 sums up the functions being supported and the features usually offered for these functions.

3.4.7 Laboratory Information Systems (LIS)

Similar to radiology departments, laboratories exist both as functional areas within a facility, for example, a hospital, and as independent enterprises that offer their services to other health care facilities. Laboratories perform *execution of lab examinations* (Table 3.7). During *execution of lab examinations*, specimens of patients (e.g., blood sample, tissue sample) are used. In contrast to radiology departments, no *appointment scheduling* is required in laboratories. Depending on the type of laboratory, different examination technologies are used (e.g., chemical analysis of blood samples, microscopical analysis of tissue samples). Chemical analysis is usually done using automated equipment. Depending on the order, the sample is usually distributed automatically to various analytical devices, which are regularly checked for their precision in order to conform to quality management requirements. In addition, the laboratory physician checks all results of a sample for plausibility (so-called validation).

Application systems supporting *execution of lab examinations* and offering features as in Table 3.7 are called *laboratory information systems (LIS)*.

Laboratory information systems support the management of the whole analysis procedure: receipt of the order and sample, distribution of the sample and order to the different analytical devices, collection of the results, technical and clinical validation of results, communication of the findings back to the ordering department or facility, and general quality management procedures. The validation of laboratory results is more effective when patient-related clinical data (e.g., recent diagnoses, drug medication) are accessible to the laboratory physician. The LIS in a particular health care facility should therefore be closely connected to the facility’s MDMS, patient administration system, CPOE system, and PDMS. Application software products for LIS are usually also sold by specialized vendors.

Table 3.7 Set of functions and related features typically to be supported by *laboratory information systems*

Function (from Sect. 3.3.2)	Typical features
<i>Execution of lab examinations</i>	Receive orders Receive blood samples Assign order and blood samples to devices Collect results from devices Display earlier lab results of a patient Validate results Prepare report Communicate report to ordering unit Provide means for preparing statistics

Table 3.8 Set of functions and related features typically to be supported by *operation management systems*

Function (from Sect. 3.3.2)	Typical features
<i>Medical admission</i>	Provide means for check-in of patient Provide forms for documenting medical history Provide forms for documenting diagnosis
<i>Decision-making and patient information</i>	Provide forms for documenting patients' informed consent
<i>Execution of operations</i>	Display vital parameters from monitoring devices Provide forms for documenting procedures and outcomes
<i>Coding of diagnoses and procedures</i>	Provide catalogs and other means for coding diagnoses Provide catalogs and other means for coding procedures Provide means for creating statistics
<i>Patient discharge and transfer to other facilities</i>	Provide forms for preparing operation reports Provide means for ordering patient transfer to ward
<i>Supply and disposal management, scheduling, and resource allocation</i>	Provide means for managing rooms Provide means for managing appointments Provide means for managing medical devices Provide means for ordering drugs, materials, and laundry Provide means for creating operation plans (daily, weekly) Provide means for preparing work schedules

3.4.8 Operation Management Systems (OMS)

Invasive procedures for patients at a particular health care facility are performed in the operating rooms (ORs) at this facility. Usually, patients stay in the OR for only a few hours. During this time, they are prepared for the operation, the operation is performed, and finally, for a period of time after the operation, the patients' state is monitored. This results in performing the functions listed in Table 3.8.

Application systems supporting functions and offering features as in Table 3.8 are called *operation management systems (OMS)*.

Operation management systems support *execution of operations* as a specialization of *execution of diagnostic and therapeutic procedures*. They allow operation

date and time to be assigned and should therefore be available on the wards as well as in the offices and management units of the OR. Depending on the planned operations, an operation plan can be created for a day or a week. The data necessary for efficient planning are the diagnoses of the patient, the planned operation (medical procedure), the surgeons and other staff involved (human resources), the planned time for operation (appointment), and the available OR (space). Therefore, the *OMS* should be closely connected to the *MDMS*.

A vast amount of data must be documented during each operation, including the members of the operating team, the operative procedure, the date and time, duration of the operation, materials (e.g., implants) used, and other necessary data to describe the operation and its results. Surgeons cooperate tightly with anesthesiologists during the operation. For their documentation, anesthesiologists need a lot of data, which must usually be documented by surgeons and vice versa. To avoid transcriptions, an *OMS* should therefore also provide supportive features for anesthesiologists in an integrated way.

Usually, the planning data are taken from the operation planning component to be updated and completed during and after the operation. These data can be used to create an operation report, which may be completed with further comments by the surgeons. Therefore, word processing capability is needed. Operation data needed for billing must be coded and then communicated to the administrative application components. The *OMS* should also allow extensive data analysis (e.g., operation lists for junior surgeons).

3.4.9 Patient Data Management Systems (PDMS)

Patients in critical situations are treated in intensive care units (ICU). These patients are generally in an unstable state and within seconds may enter into a life-endangering situation. Thus, the detailed and complete presentation of all vital parameters (e.g., blood pressure, pulse, breathing frequency) is required for a successful therapy. This is only possible when automated monitoring devices continuously measure and record various parameters. In addition, parameters that could point to the initial deterioration of the patient's status should be automatically detected and should lead to an immediate alert of the treating health care professionals. Functions being performed in ICU are outlined in Table 3.9.

Application systems supporting functions and offering features as in Table 3.9 are called *patient data management systems (PDMS)*.

Patient data management systems are specialized to automatically monitor, store, and clearly present a vast amount of patient-related clinical data in ICUs. They also support scoring (e.g., TISS,³ SAPS⁴) and may offer features for decision support and various statistical analyses.

³Therapeutic Intervention Scoring System.

⁴Simplified Acute Physiology Score.

Table 3.9 Set of functions and related features typically to be supported by *patient data management systems*

Function (from Sect. 3.3.2)	Typical features
<i>Medical admission</i>	Provide means for check-in of patient Provide forms for documenting medical history Provide forms for documenting diagnosis
<i>Medical and nursing care planning</i>	Provide means for preparing a care plan Offer decision support for care planning
<i>Execution of diagnostic, therapeutic, and nursing procedures</i>	Display vital parameters from monitoring devices Display warning messages Provide forms for documenting clinical procedures Provide forms for documenting medications Create work list for a group of patients Print forms Provide means for creating statistics
<i>Coding of diagnoses and procedures</i>	Scoring of the patient (e.g., TISS, SAPSII) Provide catalogs and other means for <i>coding of procedures</i> Provide catalogs and other means for <i>coding of diagnoses</i>
<i>Patient discharge and transfer to other facilities</i>	Provide forms for writing a transfer letter or discharge summary Provide means for finalizing documentation Communicate discharge information Provide means for ordering patient transfer to other units
<i>Supply and disposal management, scheduling, and resource allocation</i>	Assign staff to patients or rooms Provide means for ordering consumables Provide means for ordering drugs Provide means for ordering laundry Provide means for managing medical devices Organize patient transport Work scheduling

After transfer to a regular ward, a short summary of therapies on the ICU should be created and communicated to the application components used at the ward. In addition, a connection to the application components for *order entry* and result reporting is necessary. Software for a *PDMS* is sold both by specialized vendors and by vendors that also offer automated monitoring tools.

Intensive monitoring is also required during anesthesia. Therefore, *PDMS* are also used in the preparation, execution, and follow-up of operations. *PDMS* for anesthesia have additional features for anesthesia planning and execution.

3.4.10 Enterprise Resource Planning Systems (ERPS)

As part of the *administration of health care facilities*, there are functions to be performed that differ little from those in facilities in other industries. These include, in particular, the functions *controlling*, *financial accounting*, *facility management*, *human resources management*, *quality management*, and *supply and disposal management* (Table 3.10).

Table 3.10 Set of functions and related features typically to be supported by *enterprise resource planning systems*

Function (from Sect. 3.3.2)	Typical features
<i>Controlling</i>	Provide means for overhead cost management Provide means for product costing Provide means for cost center accounting Provide means for cost element accounting
<i>Financial accounting</i>	Provide means for accounts payable Provide means for accounts receivable Provide means for general ledger accounting Provide means for <i>asset accounting</i>
<i>Facility management</i>	Provide means for preventive maintenance Provide means for control of security issues Provide means for incident tracking
<i>Human resources management</i>	Provide means for organizing recruitment Provide means for organizing training Provide means for organizing career development Provide means for performance evaluation
<i>Quality management</i>	Provide a collection of internal processes and regulations Provide means for editing and graphically illustrating collections of internal processes and regulations
<i>Supply and disposal management</i>	Provide means for managing logistics Provide means for stock-keeping Provide means for order management Provide means for managing and monitoring disposal

Application systems supporting functions and offering features as in Table 3.10 are called *enterprise resource planning systems (ERPS)*. *ERPS* enable health care facilities to manage their financial, human, and material resources.

A close connection is needed to the facility’s *patient administration system*, in particular, but also to the other application components mentioned before, in order to obtain, for example, billing data and legally required diagnoses and procedure codes. Most of the application software products used for *ERPS* in health care facilities are not specific to health care but are also used in other industries outside health care where similar administrative functions have to be supported.

One major goal of the *ERPS* is the documentation and billing of all accountable services. The types of data needed and the details of billing depend on the country’s health care system.

3.4.11 Data Warehouse Systems (DWS)

For decision-making, the management of a health care facility (Table 3.11) requires up-to-date information about the facility’s operation as a whole. The management is, for example, interested in answers to questions such as: What are the top ten diagnoses of the patients treated in our facility? Which department of the facility

Table 3.11 Set of functions and related features typically to be supported by *data warehouse systems*

Function (from Sect. 3.3.2)	Typical features
<i>Management of health care facilities</i>	Integrate data from different application components Structure data Analyze data Provide means for data mining
<i>Execution of clinical trials</i>	Provide means for recruiting patients

causes the highest material costs? In which department do patients stay longest on average?

In order not to jeopardize the performance of these application systems with resource-consuming data analyses, dedicated application systems are used that extract the required data from the source application systems at certain intervals and then make them available for analysis. Application systems that support functions and offer features as in Table 3.11 are called *data warehouse systems (DWS)*.

Data warehouse systems are also used to support medical research, especially for clinical trials. Loading data, for example from the *MDMS*, *LIS*, *RIS*, or *CPOE system* into a *DWS*, the recruitment of patients for clinical trials, and the statistical *evaluation* of patient data can be effectively supported.

Data warehouse systems can help to answer the aforementioned questions by pooling management-relevant data from other systems such as the *ERPS* and the *CPOE system* and providing means to analyze these data through data mining techniques. A *DWS* supporting *management of health care facilities* is often called a business intelligence system.

Extracted data in the *DWS* have been transferred and aggregated into a suitable format and then actively loaded into the data warehouse (push principle). A specific request on the data will only access data already loaded into the data warehouse.

An ISO standard exists which provides implementation guidance for data warehouses in health information systems [6].

3.4.12 Document Archiving Systems (DAS)

As part of the function *archiving of patient information*, long-term archiving is a particular challenge for both paper-based and digital documents. Application systems supporting this function and offering the features listed in Table 3.12 are called *document archiving systems (DAS)*.

Dependent upon the type of data and national laws, patient-related data must be stored for up to 30 years. For *long-term archiving*, confidentiality, availability, and integrity of the data according to the Open Archival Information System model (ISO 14721) must be guaranteed. Availability means that data must be retrievable and readable at any time throughout the archiving period. To ensure integrity, data must be complete and unchanged. In health care, authenticity of the author and the time of the creation of a document are important aspects of integrity. For example, unauthorized alteration of the date of creation results in the loss of integrity of

Table 3.12 Set of functions and related features typically to be supported by *document archiving system*

Function (from Sect. 3.3.2)	Typical features
<i>Long-term archiving</i>	Import documents Scan documents Index document content Manage storage formats Manage storage media Provide access to archived information Attach digital signatures Communicate documents to other applications

archived data. The conditions of confidentiality, availability, and integrity can hardly be met by the individual application systems introduced before. This is especially true as these application systems cannot be expected to be in use for up to 30 years. It therefore makes sense to assign this task to a dedicated application system and copy the documents to be archived there.

Document archiving systems offer long-term archiving of patient-related and perhaps of other data and documents. Archiving is based on sustainable standardized data formats, document formats, and interfaces. The *DAS* also provides standardized indexing of document content and regular updates of storage media. Qualified electronic signatures, for example in compliance with European directive 1999/93/EC [7], can be used to guarantee long-term integrity of stored data in case the *DAS* is enabled to renew outdated signatures and hash algorithms. *DAS* are typically closely linked to all application systems that generate data and documents which it must store for a long time. It obtains copies of data and documents from those systems, indexes them, and stores them, while enabling fast retrieval. Paper-based documents can be integrated through scanning, which makes it possible to eliminate the physical space needed for paper-based archiving. *DAS* can typically archive not only text-related documents but also images, videos, and other multimedia data. All these documents may be stored using established non-proprietary industry standards such as:

- ASCII (American Standard Code for Information Interchange),
- PDF/A, an ISO standard for long-term archiving of documents based on the Portable Document Format (PDF),
- XML (Extensible Markup Language),
- TIFF (Tagged Image File Format),
- JPEG and MPEG, acronyms for the names of the committees that created the standards (Joint Photographic Experts Group and Moving Pictures Expert Group),
- Digital Imaging and Communications in Medicine (DICOM, Sect. 3.7.2.4),
- CDA (Clinical Document Architecture), an ANSI standard for the structuring of clinical documents (Sect. 3.7.2.2).

It makes sense to use vendor-neutral archives (Sect. 3.9.3) to realize *DAS*.

Patient-related medical data and documents stored in the *DAS* in a facility have to be made available to the medical management documentation system of this facility in order to enable their users to directly access information from earlier patient contacts.

3.4.13 Application Systems for Patients and Informal Caregivers

In contrast to the previously described application systems in health care facilities, application systems for patients and informal caregivers cannot be structured that clearly by the tasks they perform and their typical functionalities. This is due to the fact that there are a multitude of mobile health (mHealth apps), desktop, and browser applications available that support patients in better understanding, dealing with, and (independently) managing their health condition. In addition to application systems dealing solely with self-diagnosis, *self-treatment*, or knowledge management, there are also a number of combined application systems, i.e., mHealth apps, that provide knowledge, support therapy, and assist patients in keeping their appointments at the same time. Some typical types of application system for patients and informal caregivers are described below. These are examples and by no means represent an exhaustive list.

3.4.13.1 Patient Portals

Patient portals are offered by health care facilities. They are primarily available for patients of this facility and informal caregivers to provide them with an overview of their health data, organize documents, and actively manage themselves. *Patient portals* can also be used to improve communication between health care professionals and patients. For example, relevant documents can be uploaded to the portal by patients before admission to a hospital or rehabilitation center and are made available there for early access. This improves not only the preparation for admission to a facility but also creates transparency of information, which is crucial for patient empowerment.

Patient portals support *medical knowledge management* including document management and, in some cases, also *appointment scheduling*. Thereby, they offer features such as those described in Table 3.13.

In addition to *patient portals* that focus exclusively on the needs of patients, there are also special portals for relatives. Such portals mostly support relatives in their role as informal caregivers. Simple portals for relatives usually provide information about a disease and how to deal with it. They are therefore also referred to as information platforms. Information can be provided in different ways, ranging from simple structured information sets to comprehensive e-learning services. More comprehensive portals for relatives also offer opportunities for *medical knowledge management* as previously described for *patient portals*.

Table 3.13 Set of functions and related features typically to be supported by *patient portals*

Function (from Sect. 3.3.1)	Typical features
<i>Medical knowledge management</i> including document management	Administer user account Provide/upload documents Retrieve documents Fill in forms View treatment Information retrieval
<i>Appointment scheduling</i>	Manage appointments (view, edit, and schedule)

3.4.13.2 Telemonitoring Systems

Telemonitoring systems are provided by health care facilities or by health insurance companies. They are primarily used to monitor patients' state of health remotely, reinforcing a patient-centered health care approach. The aim is to detect critical or conspicuous changes in the state of health in order to be able to address them as quickly as possible. Therefore, they are used particularly frequently in post-acute monitoring of patients, for example, after discharge from a hospital or a rehabilitation center, as well as in chronic diseases, such as with Mr. Russo's heart failure.

In principle, *telemonitoring systems* acquire a patient's health data, transmit it to a health care professional, and represent it to that professional (and in some cases also to the patient) in an appropriate form. The monitoring of the health status can be done either in real time or time-delayed. In addition to a clear visualization of the recorded data and its provision to a treating health care professional, some *telemonitoring systems* also handle the partial or complete analysis of the data. Health care professionals either receive aggregated information on a patient's health status or receive alerts as soon as a critical value or abnormal changes in health status are detected.

Telemonitoring systems thus support the patient's self-management abilities and *self-diagnostics* through continuous monitoring. Thereby, they offer features such as those described in Table 3.14.

The range of functions and complexity of *telemonitoring systems* varies greatly in practice. For example, simple *telemonitoring systems* only record a patient's state of health via manual data entries by patients, for example, by querying the symptoms associated with a disease. There are also a number of *telemonitoring systems* that, in conjunction with other systems such as a blood pressure monitor, can also record vital parameters and evaluate them independently in combination with other patient-related data. The range of functions thus extends from simple observations to comprehensive application systems which, in addition to recording, also perform analysis, management, and initial diagnosis.

Telemonitoring systems are increasingly supplemented by functionalities for patient education and patient coaching.

3.4.13.3 Self-Diagnosis Systems

Simple *self-diagnosis systems* in the form of web applications or mHealth apps are used to provide people suffering from ailments with information about what illness they might have. The so-called symptom checker systems provide possible diagnoses or disease information matching the symptoms entered by the patient in just a few minutes. This works particularly well for simple, common diseases. For rarer, more complex conditions, however, the results provided are significantly less precise.

Self-diagnosis systems are also used in various disciplines for remote diagnosis, for example, when a patient is unable to visit a physician. In this case, *self-diagnosis systems* are considered telemedicine applications that are used for communication between physicians and patients.

Self-diagnostic systems support *self-diagnostics* and self-management of patients. Thereby, they offer features such as those described in Table 3.15.

Self-diagnosis systems usually operate on the basis of artificial intelligence and neural networks. They automatically analyze a patient’s entries, from simple texts to image data. As mentioned before, some of these application systems transmit their results automatically or at the request of a patient to a (specialist) physician in the sense of telemedicine. Such *self-diagnosis systems* are used, among other things, in dermatology for the prevention and early detection of skin diseases such as skin cancer.

Table 3.14 Set of functions and related features typically to be supported by *telemonitoring systems*

Function (from Sect. 3.3.1)	Typical features
<i>Self-diagnostics</i>	Recording/measurement (e.g., vital parameters and symptoms) Monitoring of vital parameters Analysis Reporting Diagnosis/detection of risks Visualization Sending alarms

Table 3.15 Set of functions and related features typically to be supported by *self-diagnosis systems*

Function (from Sect. 3.3.1)	Typical features
<i>Self-diagnostics</i>	Analysis (e.g., image analysis, analysis of medical/therapeutic parameters, and text analysis) Provision and evaluation of assessments/tests Diagnosis Reporting Information provision/provision of information Patient education

It is important to note that no form of *self-diagnosis systems* can replace professional assessments by physicians, not to mention professional treatment. Instead, the aim is to involve patients to a greater extent in the care process (patient empowerment) and to improve communication between patients and health care professionals.

3.4.13.4 Self-Treatment Systems

Self-treatment systems range from simple treatment-assisting systems to strengthen patients' self-management skills to individual therapy systems that can be used either as a complement or as an alternative to standard therapy. In addition to software-based mHealth apps, desktop and web applications, more and more hardware-based systems are also being used. Through the implementation of different sensor technologies, these enable not only the guidance of therapies, but also a monitoring of the state of health or even an evaluation of therapeutic measures.

Overall, the feature set of *self-treatment systems* varies greatly and depends, among other things, on the intended use and the specific area of application. Table 3.16 shows some typical features to support patient *self-treatment*.

Information-only application systems represent the simplest form of *self-treatment systems*. These simply contain descriptions to educate patients about one or more alternative therapies. For example, Mr. Russo can access information about a healthy diet for lifestyle changes or learn about ways to reduce stress, such as yoga exercises. More sophisticated *self-treatment systems* include not only information and guidance but also concrete instructions on how to perform a therapeutic or complementary measure. For example, Mr. Russo is not only recommended certain yoga exercises, but they are also explained in detail, for example, in the form of texts or instructional videos. Thereby, Mr. Russo can exercise along with the application system and mark off individual workouts to save his progress. Embedded assessments, whether automated based on sensor technology or as queries via manual data entry, also help to record the current state of health and changes caused during the therapy period. These can either be feedback to the patient or be forwarded in the form of telemedicine applications to a health care professional for further interpretation and subsequent therapy adjustment.

Reminders to perform an activity, such as exercise sessions or taking medication, are also integral parts of *self-treatment systems*.

Table 3.16 Set of functions and related features typically to be supported by *self-treatment systems*

Function (from Sect. 3.3.1)	Typical features
<i>Self-treatment</i>	Therapy (instruction, evaluation, adaptation) Monitoring Provision and evaluation of assessments/tests Reporting Information provision/provision of information Reminder (e.g., for taking medication) Serious games

3.4.14 Other Application Systems

In addition to the application systems introduced so far, we can usually find many other types of application systems, often serving specific needs of the respective health care facility and its departments. For example, depending on the size of the hospital, a hospital may have its own pharmacy department which needs a *pharmacy information system* to supply wards and patients with the right drugs in the right dose. Depending on the specializations of a hospital, there may be, for example, a *cardiovascular information system (CVIS)* or a *dialysis information system*, which are specialized *MDMS*.

Table 3.17 lists some of these specific application components. This list is not intended to be exhaustive, however.

Until now, we have focused on “classical” application systems, i.e., software installations in health information systems primarily supporting the functions as listed in Sect. 3.3. But there is an increasing number of installations of software in health care settings that primarily control medical devices. Hence, medical devices can increasingly be considered to be application systems and in many cases to be specialized *MDMS*. Consequently, they not only provide information (e.g., findings and images) via respective interfaces, but also need information from other application components (e.g., patient, case, order). This close interconnection is often referred to by the term “converging technologies.” Due to the considerable risks for patient safety, reasonable diligence must be exercised when integrating these converging technologies into the computer-based part of health information systems.

Table 3.17 Further specific application components in health information systems (examples)

Application component	Description
Blood bank management systems	Support blood donor services, blood analyses, administration of blood bottles
Cardiovascular information systems (CVIS)	Provide many features of <i>clinical information systems (CIS)</i> and <i>radiology information systems (RIS)</i> while tailoring the special needs of cardiology departments
dialysis information systems	Provide many features of <i>CIS</i> while tailoring the special needs of dialysis departments; have interfaces to hemodialysis machines
Oncology information systems	Provide many features of <i>CIS</i> while tailoring the special needs of oncology departments
Orthopedics information systems	Provide many features of <i>CIS</i> while tailoring the special needs of orthopedics departments; can include a computer-aided design (CAD) system for transplant planning
Pathology information systems	Have similar features as <i>LIS</i> , for example, receiving orders, writing reports
Pharmacy information systems	Support the workflow in pharmacy departments: receiving drug orders, managing the drug stock, distributing drugs throughout the hospital
Teleradiology systems	Enable evaluations of radiological images from (external) radiologists’ remote workplaces and may be closely connected to <i>RIS</i> and <i>PACS</i>

3.4.15 *Clinical Information Systems (CIS) and Electronic Health Record Systems (EHRS) as Composite Application Systems*

Not every health information system contains an *MDMS*, an *NMDS*, or a *CPOE system* as separate, identifiable application systems. Instead, these components are often closely integrated modules of the so-called *clinical information systems (CIS)*.

Clinical information systems are also often called *electronic health record systems (EHRS or EHR systems)*. As introduced in Sect. 2.10, the electronic health record (EHR) is a complete or partial health record stored on an electronic storage medium. Given this definition, every computer-based application component supporting the *execution of diagnostic and therapeutic procedures* or other subfunctions of *patient care* (e.g., *MDMS*, *outpatient management system*, *NMDS*, *PDMS*) contains at least a partial EHR. In the *CIS* of a certain health care facility, these partial EHRs are often integrated and made available to the professionals from all areas of the facility to provide one harmonized view on the data of a patient. Because of this harmonized view, the term *electronic health record system (EHRS)* as a particular application component has become quite common.

3.5 Logical Tool Layer: Data Integrity

In the previous section, we saw that there can be very many different types of application systems in health information systems. Therefore, we are dealing with very complex information systems in health care. In information systems, it is generally important to make sure that the data stored is correct. In complex health care information systems, this is a particular challenge.

For the correctness of data in health information systems, we use the term *data integrity*. There are many aspects of data integrity. Here we want to focus on the following aspects:

- the unique and correct identification of objects (object identity),
- the correct representation of relationships between these entities (referential integrity),
- the consistency of duplicated data among all application components in a health information system (*data consistency*).

In this section, you will learn more about these aspects of data integrity.

Object identity originates from object-oriented programming and means that an object has an existence that is independent of its value. Thus, two objects may look the same, i.e., they have the same value, but can still be different. Applying this to the representation of entity types in a database leads to the requirement that the representation of every entity must be uniquely identifiable. In a health information system, this is especially important for entity types such as “patient” and “case,” but also for “finding,” “order,” and all other entity types (Sect. 3.2.3). This

identity is needed, since all data need to be assigned to a specific patient and their cases. For example, if application components exchange information on the patient entity “Mr. Russo,” they must be sure they are communicating about the same entity (the “real” Mr. Jakub Russo, not another patient who has by chance the same name).

Experience has shown that object identity of the entity type “patient” can be guaranteed only when every patient receives a unique number, the PIN. The PIN should have no internal meaning. That is, it is created continuously and is usually numerical. Past attempts to generate a PIN from data collected from the patient, for example, from the date of birth and the name, have led to considerable problems, for example when a date of birth is corrected and thus the PIN also changes. In this case, object identity could be compromised. For example, the “real” Mr. Jakub Russo has the PIN 3050.1515 that is used throughout all application components and in the communication between them when referring to this real person.

Similar actions should be taken for the entity type “case.” A CIN, which should also have no apparent meaning, should be assigned for every case. If all application components of a health information system ensure that a case identified with a CIN is always assigned to the correct patient, the CIN can be used as an identifier for the patient. During order entry, for example, the CIN can then be used to uniquely identify a patient when ordering a laboratory test.

In every health care facility, assigning a PIN and a CIN to a patient is part of the function *patient identification* of this patient, a subfunction of *patient admission*. The PIN must be used in all parts of the hospital, and thus in all application components, for the identification of the patient and will also be used during future visits. Since PIN and CIN are assigned during *patient admission*, only one admission has to be performed and only one PIN has to be assigned to patients during their visit, regardless of which ward they are treated on. For example, Mr. Russo was assigned a PIN (and CIN) when he was first admitted to Ploetzberg Hospital. This PIN will now be used whenever Mr. Russo is admitted to this hospital again.

This works well if there is exactly one dedicated application system supporting *patient admission*, namely the *patient administration system*. The *patient administration system* must have direct access to a database that holds data allowing the reidentification of all past patients and cases in the hospital. To have this *MPI* as part of the *patient administration system* is an obvious choice.

MPI are also required for *patient identification* in HIS which cover more than one health care facility. In tHIS, data about a patient are to be used across different facilities, for example, when exchanging data between two hospitals or between a hospital and a nursing home. *MPI* as a dedicated application system provides a unique transinstitutional identification of a patient across separate *patient administration systems*. This transinstitutional PIN is cross-mapped by the *MPI* to the PINs of the respective health care facilities.

Object identity provides a unique identification for entities. Referential integrity builds on top of this and means the correct assignment of entities to each other. For example, a number of cases must be correctly assigned to a certain patient, or laboratory findings must be assigned to a given case. Object identity is the precondition for referential integrity.

Object identity for patients and cases needs to be achieved among all application components, regardless of whether they are computer-based or not. Without object identity, there is no referential integrity, and without referential integrity, results cannot be related back to the correct patient. Therefore, without the correct usage of the PIN and CIN, the pure installation of technical means such as communication networks and computer systems is practically useless, as the precondition for data integrity is not met.

Object integrity and referential integrity form two elements of data integrity. The third element we want to look at is data consistency. *Data consistency* means that copies of data on the same entity are identical.

In health information systems architectures with many different application systems, the application systems often use the same data on patients and thus store them redundantly in their own database systems. This means that there are copies of data (e.g., of patient name, patient diagnosis, laboratory findings) that represent the same information about one particular entity. We call these copies of the same data “duplicates.”

Obviously, these duplicates are supposed to be identical in all application systems where they are stored. In this case, we denote these duplicates as consistent. If the data are not identical and thus represent contradictory information, we call the duplicates inconsistent. *Transaction management* tries to make sure that data are and stay consistent—we then talk of replicates, meaning that copies of data are automatically held consistent. In health information systems having only one application system with only one database system, no redundant data exist, as all data are the so-called unicums—they only exist once.

3.6 Logical Tool Layer: Architectural Styles

As defined in Sect. 2.11, the architecture of an information system describes its fundamental organization, represented by its components, their relationships to each other and to the environment, and the principles guiding its design and evolution. The components of health information systems comprise functions, business processes, and tools for data and information processing, especially application components.

With regard to the functions and processes, there are considerable differences between information systems of different kinds of health care settings. For example, there are very clear differences in functions and processes between hospitals on the one hand and patients’ home environments on the other. However, there are very great similarities at the domain layer of hospital information systems, as the hospitals’ goals and thus the hospitals’ functions are in general the same. All functions presented in Sect. 3.3 should thus be supported in every hospital information system. Remember that, from our point of view, these functions can be supported by non-computer-based or computer-based application components.

However, even for the same kind of health care setting, there are usually significant differences in information systems architectures with respect to the types and relationships of tools used and the way they are integrated. We will introduce a multidimensional taxonomy, which can be used to characterize different styles of architectures. This taxonomy will help to describe real health information system architectures, to compare them, and to assess them.

We will first take a look at the logical tool layer and later (in Sect. 3.10.3) look at health information systems' architectural styles at the physical tool layer. In doing so, we concentrate on the computer-based part of health information systems.

Architectural styles at the logical tool layer of the computer-based part of health information systems are characterized by the

- number of databases being used to store data (especially patient-related data),
- number of application systems used to support the functions,
- number of different application software products used to install the application components and the number of vendors of these products, and
- patterns of communication links between the application components used.

These facets will be introduced as semantic dimensions which can be used to categorize hospital information systems' architectures.

3.6.1 *Number of Databases: Central vs. Distributed*

Application systems of health information systems may store data on certain entity types persistently. Usually, a database is used for this purpose. We will use the number of databases storing data in health information systems as the basis to distinguish possible data distribution styles at the logical tool layer of health information systems: the DB^1 and the DB^n style.

3.6.1.1 **DB¹ Architecture**

If a health information system (or its sub-information system) comprises only one database to store all patient-related data, we call this a DB^1 architecture (Fig. 3.24). This single database system is often called the central database system for this (sub-)information system.

The precondition for the DB^1 architecture (also: *central architecture*) is that all application systems store their data only in the central database and that this database is open for accessing and storing data there. There are two ways for realizing this style:

- The application software products of the application systems originate from the same vendor who designed the database, they are all self-developed by a health care facility, or they have been developed particularly for this health care facility.

- There exists an appropriate API along with standards to query and manipulate the database. This can be achieved by using an *open platform* (Sect. 3.9.3) where the database schema on the persistence layer does not need to be known.

If application components from different vendors are used and the previous conditions are not fulfilled, the so-called DB^n style can usually be found.

3.6.1.2 DB^n Architecture

In health information systems based on commercial software components from several different vendors, we can usually find DB^n architectures (also: *distributed architecture*). This means that several application components store data on certain entity types persistently and contain their own database systems. Figure 3.22 presents this style.

As a consequence of this style, patient-related data must be stored redundantly in different application components. For example, data on the entity types “patient” and “case” may be stored in different application components, such as the *patient administration system*, *LIS*, and *RIS*.

In this architecture, great emphasis must therefore be placed on the consistency of redundant data (Sect. 3.8.1). For example, the architecture must define which system is the responsible source for which data elements. It may be useful to state, for example, that data on patient and case may be created and changed only by the *patient administration system* (however, the other components may locally store and use a copy of these data). We will discuss these topics in more detail in Sect. 3.9.1.

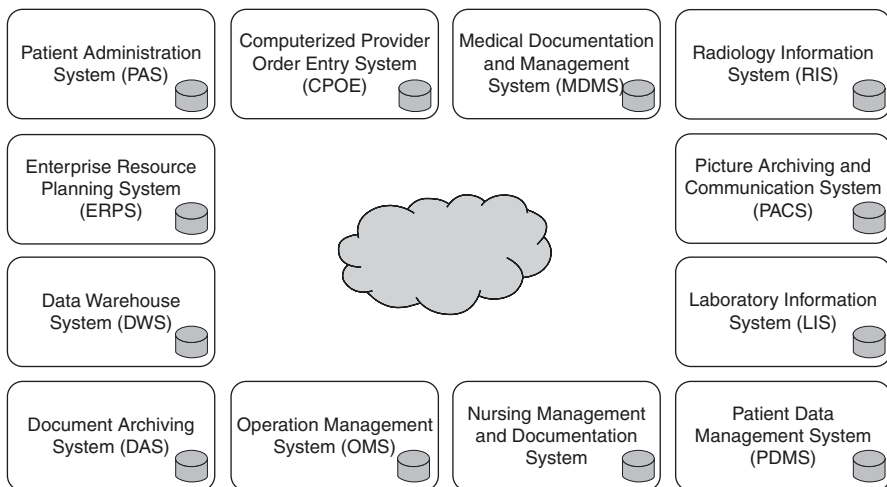


Fig. 3.22 DB^n style with multiple application systems, each with its own database system, using 3LGM² symbols. The cloud in the center indicates that some as yet unknown means is needed to link the components

It is quite obvious that in DB^n architectures consistency of redundant data can only be achieved if the application systems are able to communicate with one another, i.e., they must be interoperable. We will discuss interoperability of application systems in more detail in Sect. 3.7.1.

3.6.1.3 Mixed DB^1/DB^n Architectures

In practice, you will hardly find health information systems with a pure DB^1 style. Even if one central application component with the central database has been installed in order to support the functions, it is hardly possible to stop implementation of further application components; and those application systems will usually come with their own databases. Hence, even if a considerable part of these health information systems is DB^1 styled, they are in sum actually DB^n styled. Similarly, even DB^n styled health information systems contain sub-information systems which are DB^1 styled.

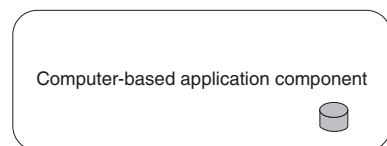
We will refer to the mixed style by the string “ DB^1/DB^n ”.

3.6.2 Number of Application Systems: *Monolithic vs. Modular*

In the simplest case, the overall health information system consists of only one computer-based application system which supports most of the functions. This application system would then look like the one rock on which the whole hospital rests. Respective health information systems are commonly called *monolithic*. We will refer to this style by the string “ AC^1 ”. Of course, this application component contains the central database and AC^1 correlates with DB^1 . A graphical representation of such a (DB^1, AC^1) architecture is presented in Fig. 3.23.

Especially in large health care facilities such as university medical centers or even in home settings, however, one application component is usually not sufficient to support the different functions. This leads to architectures with a multiplicity of application components, which can be denoted by AC^n and are called *modular architectures*.

Fig. 3.23 (DB^1, AC^1) architecture using 3LGM² symbols. The rectangle denotes the application system that contains a database system (denoted by the cylinder)



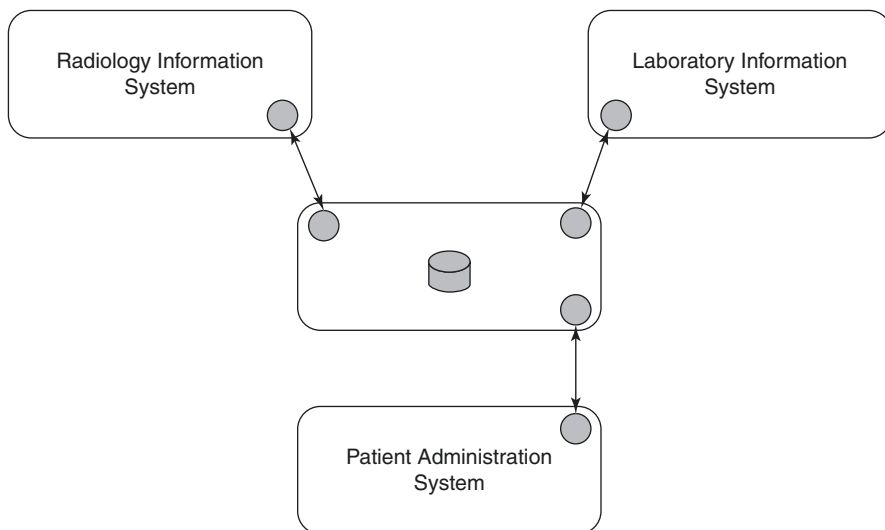


Fig. 3.24 (DB^1, AC^n) architecture with multiple application systems, using 3LGM² symbols. Only one application system (in the center) contains a database system

However, modular architectures can still be DB^1 architectures: For example, the *RIS*, the *LIS*, and the patient administration system are connected to a certain application system which shares one single database management (Fig. 3.24). The application system in the middle may be an *open platform* (Sect. 3.9.3). Such (sub-)information systems can be described by (DB^1, AC^n).

Anyhow, there is still widespread use of a combination of many application components and many databases, i.e., of (DB^n, AC^n) architectures, as in Fig. 3.22.

3.6.3 Number of Application Software Products and Vendors: All-in-One vs. Best-of-Breed

The terms “homogeneity” and “heterogeneity” are commonly used to describe whether a health information system consists of somehow similar components or very different ones. A practical measure is the number of application software products installed or the number of vendors delivering these products to a health information system. We denote a *homogeneous architecture*, i.e., a (sub-)health information system with software from only one vendor, as V^1 . Consequently, independent of the number of application systems, V^1 -HIS use only application software products that all come from the same vendor. On the contrary, *heterogeneous architectures* comprise software from several vendors and are denoted as V^n .

Obviously, (DB^1, V^1) architectures are more common than (DB^1, V^n) architectures.

An (AC^n, V^n) architecture where the different application systems are based on software from different vendors is commonly denoted as *best-of-breed architecture*, pointing to the fact that the health care setting combines the “best” application software products from different vendors. This best-of-breed architecture is typically DB^n although (DB^1, V^n) architectures could become more common in the future.

On the contrary, a monolithic (AC^1, V^1) architecture and even an (AC^n, V^1) architecture emphasizes that the health care facility selected only application software products from exactly one vendor to support as many functions as necessary. This homogeneous architecture is also called *all-in-one architecture*.

3.6.4 Communication Pattern: Spaghetti vs. Star

Application systems that are part of an AC^n architecture have to be connected as we discussed before. One way would be to directly connect those application systems that need to exchange certain patient-related data.

For example, if data on the entity types “patient” and “case” are needed in the *patient administration system* in the *RIS* and in the *LIS*, direct communication links between these components seem to be a possible solution. Hence, a communication link allowing for the transfer of patient data between the *patient administration system* and the *RIS* may be introduced. Consequently, when connecting several application systems, this will lead to an increasing number of bidirectional communication links. This architecture is called *spaghetti style*. All these different links must be supported and managed. As the number of application systems rises, the number of links grows nearly exponentially. The maximum number of communication links between n application components ($n \geq 2$) is $\sum_{x=1}^{n-1} x$. We denote architectures with this spaghetti-styled communication pattern as CP^n . Figure 3.25 presents a (DB^n, AC^n, V^n, CP^n) architecture.

To reduce the large number of links, one can use smarter methods and tools to organize and implement the interoperability of application systems, for example, by installing an application system in the “middle”, ensuring communication between the application systems.

For example, most hospital information systems following the DB^n style use a *communication server*. By using a communication server, no direct communication links between application systems are needed. Links are needed only between the application systems and the communication server. The number of links that must be managed can consequently be low—ideally, only n links exist for n application systems.

We call this style *star architecture* and denote it as CP^1 . Figure 3.26 presents a (DB^n, AC^n, V^n, CP^1) architecture. Note that star architectures do not necessarily contain communication servers, as the center of the star may also be an application system with a central database. Consequently, (DB^1, AC^n) architectures will also be CP^1 architectures. Furthermore, we may call a service-oriented architecture (SOA)

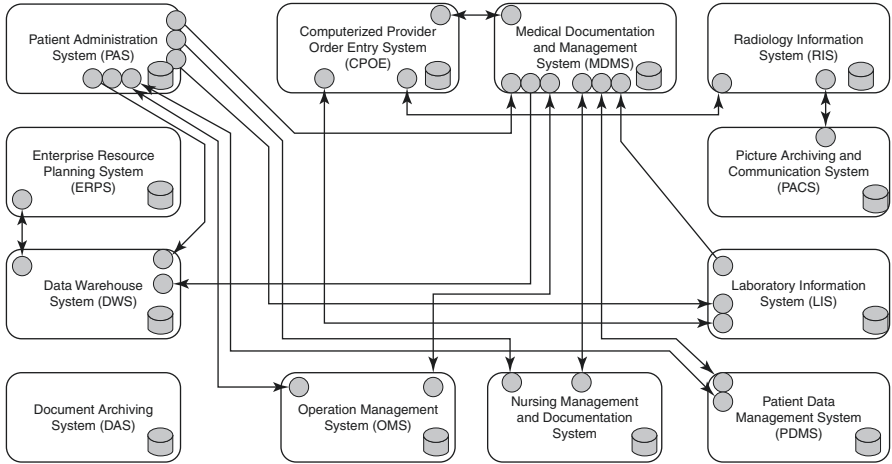


Fig. 3.25 (DBⁿ, ACⁿ, Vⁿ, CPⁿ) architecture with multiple application systems, using 3LGM² symbols, with several bidirectional communication interfaces. This representation is also called a “spaghetti” architectural style

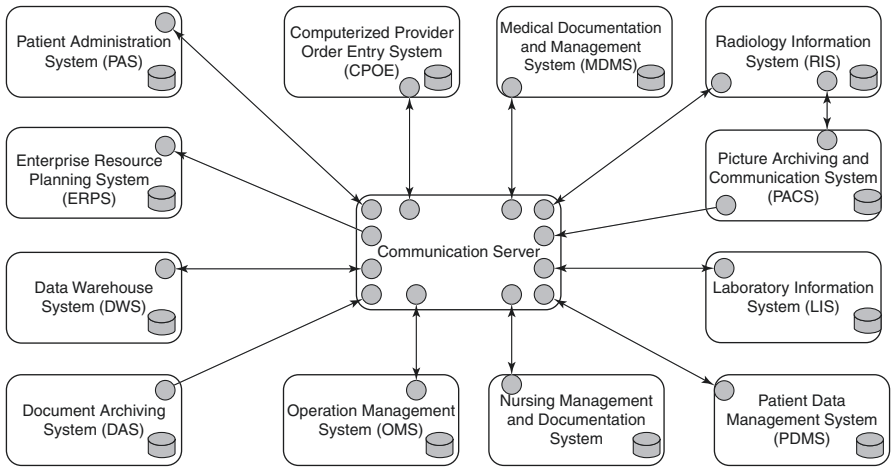


Fig. 3.26 (DBⁿ, ACⁿ, Vⁿ, CP¹) architecture with multiple application systems connected by a specific application component, using 3LGM² symbols. This representation is also called a “star” architectural style

with one application component serving as a broker or as a service bus a CP¹ architecture as well. But for AC¹, this concept obviously does not make sense and neither CP¹ nor CPⁿ should be applied.

3.7 Logical Tool Layer: Interoperability and Standards

As we saw in the previous section, health information systems are characterized by containing many application systems, i.e., they are usually of the ACⁿ architecture type. And we saw that they can only guarantee data integrity in their health information systems if the application systems communicate with each other. Even more, this communication is the absolute prerequisite for the health information system to fulfill its task of ensuring the necessary data and knowledge logistics.

Thus, application systems need to be able to communicate—they need to be interoperable. Without interoperability, no meaningful integration of application systems within health information systems is possible. Integration means that the application systems are put together in such a way that the resulting information system—as opposed to its parts—displays a new quality. There are various kinds and aspects of integration, as we will discuss later in Sect. 3.8.

Interoperability in general is the ability of two application systems to exchange information with each other and to use the information that has been exchanged. Saying that two application components are interoperable thus indicates that (1) they have interfaces and (2) they are—on the most minimal level of interoperability—basically able to send and receive data via these interfaces.

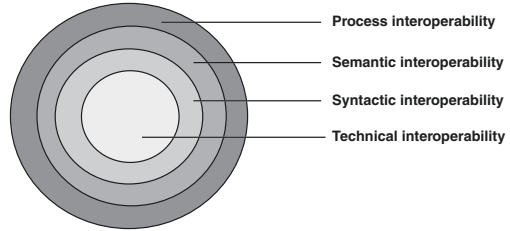
In this chapter, you will first be given an introduction into various aspects of interoperability within health information systems. Costs of the implementation and maintenance of health information systems can be significantly reduced when the application system's interfaces are based on standards. In Sect. 3.7.2, we will discuss recent interoperability standards and how they support the various aspects of interoperability.

3.7.1 Aspects of Interoperability

How does communication really work? If two application components want to meaningfully communicate, a consensus must exist about four aspects:

- the technical means of data exchange,
- the syntax of the exchanged data and messages,

Fig. 3.27 Four aspects of interoperability describing meaningful communication between application systems. (Adapted from [8])



- their semantics (i.e., meaning),
- the organization of communication with a given process.

This leads to four aspects of interoperability: technical interoperability, syntactic interoperability, *semantic interoperability*, and process interoperability (Fig. 3.27).

We will describe interoperability as a property or capability of application systems or of application software products used to install application systems.

Please note that the four aspects of interoperability do not allow dichotomous statements about application systems or application software products. Rather, they are concepts that can exist to varying degrees and relate to different aspects. For example, semantic interoperability may refer to more aspects in one application system than in another application system without completely denying semantic interoperability in the other application system.

3.7.1.1 Technical Interoperability

Technical interoperability, as the most minimal level of interoperability, is the ability of an application system to send or receive “bits and bytes” in a reliable and standardized way and thus possesses respective interfaces and implements protocols. The data may be sent or received as a file, string, or continuous stream. Files and strings of data can be considered messages (Sect. 2.2). Web technologies such as REST application programming interfaces (APIs) provide such technical interoperability.

If the application system wants to send messages to or receive messages from an application system installed on a different computer, these computers need to be physically interoperable (Sect. 3.10.2) and the interfaces have to be able to interact with lower layer communication protocols at the physical tool layer as defined in the ISO/OSI reference model (Sect. 3.10.2).

For example, the *patient administration system* is technically interoperable if it possesses an interface for reliably sending or receiving a message to or from another interoperable application system such as the *RIS*. The message exchange must work regardless of whether both application systems are installed on the same or on different servers.

To be able to consider the structure, content, or meaning of the exchanged messages, higher levels of interoperability are needed.

3.7.1.2 Syntactic Interoperability

An application system is syntactically interoperable if it is technically interoperable and additionally uses a predefined structure for the exchanged messages. Thus, when technically exchanging a data file, string, or stream, specific data elements can be identified, such as beginning and end of a message or of message segments containing data on certain entity types such as “patient,” “case,” or “finding.” Syntactically interoperable application systems are thus able to obtain information from received messages—exchanged data becomes machine-readable.

A quite simple approach to structure exchanged data are data formats such as CSV (comma-separated values) or XML. However, these formats cannot support syntactic interoperability by themselves, as they need additional agreements or rules, for example, on where to find the patient’s name or birthdate in the given dataset.

Communication standards such as Health Level 7 Version 2 (HL7 V2), DICOM, and Health Level 7 Clinical Document Architecture (HL7 CDA) provide a data or document format together with clear rules on where to find which data element. Thus, they support syntactic interoperability. All these communication standards are presented in more detail in the next sections.

For example, the *patient administration system* is syntactically interoperable if it is technically interoperable and can use HL7 V2, specifically the message type “administrative admission.” Thereby, the *patient administration system* can send Mr. Russo’s administrative data to the *RIS*. If the *RIS* is syntactically interoperable as well, it will be able to incorporate these data in its database.

3.7.1.3 Semantic Interoperability

Two application systems are semantically interoperable if they are syntactically interoperable and can exchange information (in the form of messages) that can be meaningfully interpreted by both and processed further.

We can also say that the information transferred is not only “machine-readable” but also “machine-processable.” This means that the application system can “understand” the content of messages and act accordingly, for example, by using data for automatic decision support. Beyond mere syntactic agreement between two actors and agreement on data types or structures defined in a reference model for data representation, a prerequisite for this mutual understanding is a homogeneous, jointly agreed definition of the underlying content concepts, for example, a clinical concept such as “systolic blood pressure.” Such concepts are called detailed clinical models. To achieve this, two application systems must adhere to such content models which define possible content (e.g., as openEHR archetypes) or actual content (e.g., as HL7 FHIR resources or openEHR templates, Sect. 3.7.2.8).

Modelers who create detailed concept representations which are used for communication between systems or for querying data repositories frequently draw on already existing specialized terminology standards or terminologies such as

SNOMED and LOINC (Logical Observation Identifiers Names and Codes) to enhance their models. Thus, they support semantic interoperability, as they provide a common and unambiguous vocabulary within shared information models (Sect. 3.8.2) between application systems. LOINC and SNOMED are presented in more detail in Sects. 3.7.2.9 and 3.7.2.10.

An *information model* describes the entity types and their relationships at a conceptual level, independent of any specific implementation or protocols. Information models are intended for designers. In contrast, a data model is defined at a lower level of abstraction. Data models are intended for implementers of databases.

In addition to these two terminologies, also known as nomenclatures, there are numerous classifications in medicine. Classifications are mainly used for statistical purposes, for example, counting similar objects, or for grouping similar treatment cases together for billing purposes. Occasionally, however, they are also used to support semantic interoperability although they were not actually designed for this purpose.

Some examples of classification systems used in health care are presented in Table 3.18.

Please note that semantic interoperability cannot be achieved without technical and syntactic interoperability. The ability to exchange structured messages is the precondition for exchanging meaningful data. If application systems have agreed to exchange, for example, LOINC codes within a message, the receiver can automatically extract these LOINC codes. To be semantically interoperable, the application system must be able to interpret the extracted LOINC code and understand the context in which it is used.

3.7.1.4 Process Interoperability

Finally, process interoperability addresses whether application systems are able to cooperate in certain organizational contexts, especially in certain processes, by meaningfully exchanging information. Such processes may involve, for example, ordering a radiological examination and then sending back the findings or providing a document for use outside one's own facility and then using the document from

Table 3.18 Examples of classification systems used in health care

Classification systems in health care	Concepts represented
NANDA—North American Nursing Diagnosis Association	Nursing diagnosis
NIC—Nursing Intervention Classification	Nursing interventions
NOC—Nursing Outcome Classification	Nursing outcomes
ICF—International Classification of Functioning, Disability, and Health	Health and social status, disabilities
ICD—International Classification of Diseases	Medical diagnosis
ICNP—International Classification of Nursing Practice	Nursing problems, interventions, and outcomes

outside. Process interoperability depends on successful technical, syntactic, and semantic interoperability.

An application system is interoperable with respect to a particular process if it is syntactically interoperable and capable of both sending all messages required for the collaborative processing of that process and correctly interpreting the messages received. Through this communication, the application system must be able to correctly fulfill the role it has been assigned in this process.

The more business-related the processes to be supported are, i.e., the more they correspond to the function in Sect. 3.3, the more semantic interoperability an application system must have.

Process interoperability requires the ability to handle many send and receive operations in a specific way and sequence required by the supported process. Since the communication standards mentioned so far essentially refer to individual sending and receiving operations, regulations are required that relate to the complex organizational context and the processes to be supported. IHE provides such regulations in the form of the so-called profiles. The IHE organization offers to review and certify application software products to determine whether they are interoperable with respect to a specific process and profile.

Although these profiles merely describe the use of existing interoperability standards, they have now taken on a normative character. For this reason, we also describe IHE in Sect. 3.7.2.5.

3.7.2 Interoperability Standards

As said before, the costs of the implementation and maintenance of interfaces within heterogeneous health information systems can be significantly reduced when standards for interoperability are used. Furthermore, mapping processes are usually accompanied by a loss of information or semantics.

Within the logical tool layer of health information systems, there exist standards for syntactic interoperability (also called communication standard or message standards), for semantic interoperability (common information models, terminology standards), and for process interoperability. For technical interoperability there is, for example, the already mentioned REST standard at the logical tool layer. This is complemented by so-called protocols at the physical tool layer, which will be discussed in Sect. 3.10.2.

An abundance of standards exists, some covering more than one aspect of interoperability. Thus, it is not trivial to assign them to the above-mentioned levels.

As the most widely implemented and well-established standards, HL7 V2 messages and DICOM will be described. However, there are more. HL7 CDA is a standard for structuring medical documents and contributes to syntactic interoperability. HL7 FHIR simplifies the exchange of clinical data between heterogeneous application systems using web technologies for data transfer. DICOM allows digital images and related metadata to be shared. IHE in general focuses on workflow processes

and enables process interoperability by providing guidelines on how to apply other standards, for example, DICOM, HL7 V2, and HL7 FHIR. In particular, Integrating the Health care Enterprise Cross-Enterprise Document Sharing (IHE XDS) focuses on content-agnostic, i.e., encapsulated, sharing of documents within and across different health care organizations. For the exchange of bio-signals and vital signs between point-of-care devices, the ISO/IEEE 11073 standard is available. openEHR focuses on the representation of clinical information in detailed clinical models and allows for the creation of a semantically fully defined EHR. SNOMED CT is a universal, multilingual clinical terminology for supporting semantic interoperability that includes more than one million relationships of different types between concepts. LOINC also supports semantic interoperability and is the most widely used standard for measurements and tests in health care, for example, laboratory tests. The Clinical Data Interchange Standards Consortium (CDISC) provides a number of standards, of which ODM-XML (Operational Data Model) is well-known for the representation of clinical research (trial) data and metadata, being widely used in electronic data capture (EDC) systems.

We will now discuss some of these standards in more detail. The list is by no means complete, with new standards emerging and others disappearing frequently, as well as current ones changing.

3.7.2.1 Health Level 7 Version 2 (HL7 V2)

Health Level 7 Version 2 (HL7 V2) [9] is the most-implemented communication standard in hospital information systems for the transfer of messages with data on the entity types “patient” and “case” and the other entity types as described in Sect. 3.2.3, but excluding image data.

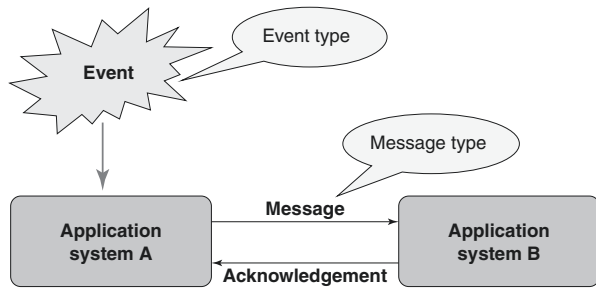
HL7 V2 has been maintained by the international standards organization Health Level Seven International (HL7) since the 1990s. It can be best considered as a standard to support mainly syntactic interoperability between application systems.

HL7 describes event types and dedicated message types that are exchanged between application systems when triggered by a specific event.

HL7 assumes that a message is sent from application system A to another application system B through the occurrence of an event of certain event type (Fig. 3.28). The message type that is used for the message depends on the occurring type of event. The message type describes the structure of the sent message and determines the meaning of the individual parts of the message.

Following the arrival of the message, application system B confirms the receipt of the message through a receipt message (ACK) that is sent back to application system A. If a communication server such as in Sect. 3.9.2 is used to send a message, for example, from the *patient administration system* to the LIS, then the communication server first takes over the role of the receiving application system B. As

Fig. 3.28 Event-driven communication with HL7 V2



a second step, the communication server, as the sending application component A, sends the message to *LIS*, which takes over the role of B.

HL7 possesses an extensive catalog of event types. For example, A01 describes the event “admission of a patient,” A02 “transfer to another organizational unit,” A03 “discharge of a patient,” and R01 “completion of an examination result.”

In addition, HL7 provides a list of standardized message types, such as *admission, discharge, and transfer (ADT)*, for messages related to admission, discharge, and transfer of patients. Other message types are ORM for a general order message (e.g., ordering a radiological examination), ORU for an unsolicited observational result (e.g., radiology report), and BAR for patient accounting message.

Message types are assigned to event types. For example, if the *LIS* in the laboratory of a hospital registers the occurrence of an event of type R01, then it can send a message of message type ORU.

All HL7 V2 messages are structured into segments, with each segment containing fields. For example, the ADT message contains at least the following segments: message header (MSH), event description (EVN), patient identification (PID), and patient visit information (PV1). With each segment, the relevant information is provided in different fields, each separated by “|”. The following example shows a very simplified pattern of an ADT message where a patient is admitted to a specified ward:

```

MSH|SENDING COMPONENT|RECEIVING COMPONENT|DATE OF MESSAGE|
EVN|A01|DATE OF EVENT|
PID|PATIENT IDENTIFIER|NAME^FIRST NAME|BIRTH OF DATE|SEX|
PV1|PATIENT CLASS|WARD ID|DATE OF ADMISSION|
  
```

Despite this standardization, the use of “plug and play” equipment is often not possible due to various reasons. On the one hand, HL7 leaves users with a certain degree of freedom with regard to semantic interoperability. Consensus must be reached between the communicating application systems, whether, for example, the

choices “male,” “female,” “other,” and “unknown” for gender can be documented as “m”, “f”, “o”, and “u”, with “0”, “1”, “2”, and “3”, or in some other fashion. Furthermore, there also exists the problem of using catalogs of terms. For example, if a *CIS* wants to order a radiological examination, it must have an up-to-date copy of the service catalog of the radiological unit.

On the other hand, manufacturers of application software products sometimes offer HL7 interfaces to their products that cannot send or receive all required event types and/or all necessary message types. In this case, a thorough analysis is necessary before deciding on a purchase.

Finally, there exist different sub-versions of HL7 V2 (2.1, 2.2, 2.3, 2.4, 2.5, 2.6) which partly differ in their definition of message content.

When a message was constructed in accordance with the previous explanations, it is left up to the particular implementation how communication between the physical data processing systems will occur. A message can then, for example, be written in a text file or transported by disk or using an FTP file transfer. The exchange format and protocol for technical interoperability at the physical tool layer is to be decided on in every single communication link.

3.7.2.2 Clinical Document Architecture (CDA)

The Clinical Document Architecture (CDA) is an XML-based document standard for storing and exchanging clinical content in the form of documents (e.g., a discharge letter, a lab finding). CDA was developed and is maintained by the HL7 organization and has been accredited by ANSI, the American National Standards Institute.

CDA is one element of the Health Level 7 Version 3 (HL7 V3) standard. HL7 V3 is a message standard developed by the HL7 organization in the 2000s. Unlike HL7 V2, where messages were developed in a pragmatic way, messages in HL7 V3 are derived from an information model, the so-called HL7 Reference Information Model (RIM).

The message encoding within CDA is based on XML, thereby supporting syntactic interoperability. CDA documents are persistent records of medical information (such as diagnostic findings, discharge summaries, or lab reports). CDA supports free text as well as fully structured, machine-processable information, thus also supporting semantic interoperability. Document templates (so-called implementation guidelines) define CDA-based document structures for specific use cases such as a discharge summary or a radiology report. CDA is used in several countries for sharing clinical documents on a national level.

CDA documents comprise two parts: the CDA header that presents metadata on the document such as sending institution, patient name, and type and date of document; and the CDA body that contains the specific information (e.g., the lab result).

The following example (adapted from [10]) shows a very simplified extract of an XML-based lab result message from the CDA body. The elements specify the type of observation (glucose 12 h fasting), the time of the observation

(February 15, 2021, 7.30 a.m.), and the status (the observation is completed). The value for the actual result is shown in the value element (182 mg/dL), and the data type is PQ = physical quantity. The glucose lab value is also given in coded form (1554–5), which is a code from the LOINC system (“LN,” Sect. 3.7.2.10). Using LOINC codes supports semantic interoperability. LOINC is uniquely identified by the LOINC Object Identifier (OID) (“2.16.840.1.113883.6.1”). All relevant terminology systems in health care possess a unique OID which allows application systems to exchange the information which terminology systems is being used.

```
<observationEvent>
<id root="2.16.840.1.113883.19.1122.4">
<code code="1554-5" codeSystemName="LN" codeSystem=
"2.16.840.1.113883.6.1" displayName="GLUCOSE^POST 12H"/>
<effectiveTime value="202102150730"/>
<statusCode code="completed"/>
<value xsi:type="PQ" value="182" unit="mg/dL"/>
</observationEvent>
```

Since HL7 V3 is not compatible with HL7 V2 and a “translation” between both formats is not trivial, HL7 V3 is primarily used for transinstitutional message exchange for which version 2 is not well suited. Both HL7 versions can be expected to coexist for a longer time.

3.7.2.3 Health Level 7 Fast Health care Interoperability Resources (HL7 FHIR)

HL7 FHIR (Fast Health care Interoperability Resources) is a recent HL7 standard based on the experience from other HL7 standards that was developed in and has been continuously updated since 2014. The standard is built upon web technologies, most notably REST APIs and simple HTTP operations. The focus is on sharing medical data in terms of transferring them from one application system to another in a standardized manner and thus make medical data portable. FHIR supports technical interoperability (by using web technology standards), syntactic interoperability (via clearly defined resources), and semantic interoperability (via an underlying information model and the use of reference terminologies).

The two basic building blocks are information models called “resources” which represent clinical, identifying, or financial information as well as APIs. The basic resources follow the so-called 80/20 approach, i.e., to meet 80% of the needs by implementing 20% of the requirements. FHIR neither aims to cover all clinical information nor to implement an EHR but pragmatically focuses on frequent, generic use cases. HL7 develops the FHIR specification and manages the resources, for which it also defines a number of rules on how to create them. FHIR provides terminology linking and validation.

To implement specific clinical use cases, generic FHIR resources are adapted into “profiles,” or more specific information models. Thus, institutional or local adaptations are implemented, albeit with a trade-off towards loss of interoperability.

Apart from supporting data sharing by providing mechanisms for interfacing different application systems with likely diverging underlying information models and persistence structures, a native “FHIR server” implements a set of APIs and resources and allows for operations, for example, to create, update, or search resources.

3.7.2.4 Digital Imaging and Communications in Medicine (DICOM)

Digital Imaging and Communications in Medicine (DICOM) [11] is an open standard maintained since the 1980s by the DICOM Standards Committee of the National Electrical Manufacturers Association. DICOM addresses the integration requirements of the medical imaging sector.

DICOM supports technical interoperability at the physical tool layer (by defining network protocols such as Transmission Control Protocol/Internet Protocol (TCP/IP)), syntactic interoperability (by its message formats), and—to some extent—semantic interoperability (by an underlying information model).

DICOM defines not only file and message formats for all types of medical imaging modalities (e.g., computed tomography, digital X-ray, magnetic resonance imaging, ultrasound, nuclear medicine imaging), but also a network protocol at the physical tool layer with a set of well-defined network services. These services permit, for example, an imaging modality to retrieve a “worklist” describing the patients to be examined from the *RIS*, to transmit the images and X-ray dose information created during an examination to the *PACS*, to confirm that the images have been archived successfully (and can thus be deleted locally), and to notify the *RIS* that the imaging procedure has been completed. Other services permit a diagnostic workstation to retrieve current and prior imaging studies, to print a hardcopy on a medical printer, or to store a report and results of measurements performed on the images. Unlike HL7 V2, the DICOM standard defines a complete network protocol stack (based on TCP/IP, Sect. 3.10.2) using efficient binary encoding and, optionally, image compression techniques. The capabilities of two communicating systems (such as the services and encodings supported by both systems) are dynamically negotiated whenever a new network connection is initiated, which permits a tight integration because systems can to some degree adapt to the capabilities of their communication peers. DICOM has gained widespread acceptance in the medical imaging sector as well as in all medical disciplines that rely heavily on digital images (in particular radiology and cardiology).

It should be noted that the *RIS* is also dependent on messages from the *patient administration system* and must also send, for example, billing data there. As discussed above, this communication should be carried out on the basis of HL7 V2. Orders from wards and outpatient units will also most likely reach radiology as HL7 messages, whereas the results and images will come back as DICOM messages. The common initiative IHE (Sect. 3.7.2.5) has taken on the task of settling this complex interplay.

3.7.2.5 Integrating the Health care Enterprise (IHE)—Integration Profiles

Integrating the Health care Enterprise (IHE) is an organization founded by medical professional societies in 1998 together with the health care IT industry [12]. Its aim is to improve the integration of application systems in health care and to check and certify if application software products to be used for the application systems have the process interoperability needed for this integration.

The approach taken by IHE is to analyze typical work processes occurring in health care. IHE identifies the application systems involved in these processes and the information that should be exchanged between these application systems to support the diagnostic and therapeutic processes as good as possible. For example, IHE has defined working processes in the form of “integration profiles” for areas such as cardiology, pathology, radiology, and pharmacy. It has also defined integration profiles for the exchange of clinical documents between different health care facilities (IHE XDS, Sect. 3.7.2.6).

IHE then selects existing standards, i.e., especially HL7 V2 and DICOM, for each “transaction” (information exchange between application systems) and restricts the options offered by these standards for each transaction so that plug-and-play interoperability becomes possible.

By defining work processes, IHE supports process interoperability. Together with standards such as HL7, HL7 CDA, HL7 FHIR, openEHR, and DICOM, it indirectly supports syntactic and semantic interoperability.

IHE furthermore offers comprehensive test software enabling vendors to test their products’ interfaces for IHE-conformant process interoperability and organizes large cross-vendor testing events called “IHE Connectathons” (from “connection” and “marathon”).

IHE is not a standards body as such but fills a very important gap by selecting the most appropriate set of standards for a typical clinical workflow. It reduces the indeterminacy of the way of interaction, for example, between HL7 V2 and DICOM, by proposing clear rules for their joint use. The resulting technical specifications are published annually as the IHE Technical Frameworks.

Although the profiles are not communication standards, they gain normative character through their increasing dissemination and can be regarded as quasi-standards for the use of communication standards.

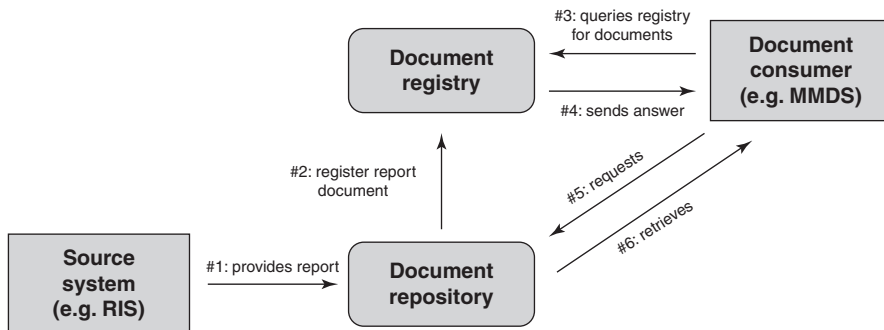


Fig. 3.29 IHE XDS.b actors and interactions (simplified), using the example of a radiology report. #1–#6 indicate order of messages

3.7.2.6 IHE Cross-Enterprise Document Sharing (XDS)

One of the above-mentioned technical frameworks is the IT Infrastructure Framework (ITI), which specifies standards-based implementations for exchanging clinical information. Among other things, it describes how to share documents between different health care organizations. IHE Cross-Enterprise Document Sharing (IHE XDS) is used, for example, for the Austrian ELGA (national EHR) and partly in the German Medical Informatics Initiative.

IHE XDS defines neither the content of these documents nor their internal structure, making it “content-agnostic.” Instead, it defines actors along with their interactions and integration profiles of both these components. IHE XDS.b is a prominent integration profile defining affinity domains and actors.

Figure 3.29 shows the actors in this profile and their interactions. A source system (e.g., an *RIS*) provides a new document, for example, a radiology report on Mr. Russo, along with its metadata to the document repository. This document is then registered in a document registry. A document consumer, for example, another clinical application system, such as the *MDMS*, at a different site in the health care network, can query the document registry to locate a relevant document and then query and subsequently retrieve this document from the decentral document repository. Document repositories are usually decentral, while the registry is central.

3.7.2.7 ISO/IEEE 11073

The ISO/IEEE family of standards for the exchange of data between medical devices comprises, for example, a standard for exchanging bio-signals and vital parameters between point-of-care devices [13]. Furthermore, the standard enables a dynamic exchange and reconfiguration of devices and remote control, for example, of infusion pumps.

Medical devices can be considered as combining an application system at the logical tool layer with a dedicated physical data processing system at the physical tool layer, which is often also called an appliance.

ISO/IEEE 11073 can be used, for example, to receive and display data from infusion pumps, respirators, ECGs, etc. on patient monitors in the ICU and for integrating medical devices in operating rooms. It is also used for receiving data from mobile wellness devices (e.g., fitness tracker) and personal health applications (e.g., diabetes app).

ISO/IEEE 11073 supports technical interoperability by defining a complete communication protocol stack and, for example, includes binary encoding of data that is optimized for near-real-time transmission. It also supports syntactic interoperability by defining the formatting and syntax for the exchanged messages.

While ISO/IEEE 11073 enables the exchange of especially vital parameters between the application systems of the medical devices, HL7 V2 and DICOM facilitate the data exchange between the application systems mentioned in Sect. 3.4. However, it is also necessary to be able to exchange data between these worlds. For example, patient demographic data is also required for the devices, and vital parameters should be able to be transferred to the EHR. Gateways are used to realize this communication.

3.7.2.8 open Electronic Health Record (openEHR)

openEHR (open Electronic Health Record) is an open standard specification which focuses on the standardized representation of clinical information in EHRs. It has been maintained by the openEHR Foundation since the early 2000s and supports semantic and syntactic interoperability.

The specification follows a detailed (maximum) clinical information modeling approach and provides a formalism—the archetype definition language (ADL)—to define *archetypes* as the basic building blocks representing clinical concepts. Furthermore, it includes the archetype query language (AQL), which—while syntactically similar to the ubiquitous SQL—operates on clinical concepts and—unlike SQL—requires no knowledge of the persistence data structures (e.g., database schemata).

openEHR follows a two-layer modeling paradigm, fundamentally separating clinical content from technical details. The first layer is the reference model, which describes the smallest logical blocks along with their implementation, for example, data types and structures or basic EHR components. As part of the archetype model, the second layer defines the domain model representation in a maximum approach, meaning that an *archetype* (e.g., blood pressure or height), as the most basic and self-contained clinical concept, should serve all potential uses of this concept within the health care domain.

In other words, archetypes are reusable as semantic building blocks in different contexts without further modification. Within an archetype, each single item or data

instance can be linked or bound to a terminology (e.g., LOINC or SNOMED CT). Archetype classes include “composition,” “section,” and “entry,” whereby the latter again includes “observation,” “evaluation,” “instruction,” “action,” and “cluster.” To create more complex information units on the basis of these interchangeable building blocks in the context of specific use cases, openEHR uses templates. An operation report, the results of an assessment test, or a form/document could, for example, be represented as templates, each assembled using archetypes just like construction bricks. A number of tools support the collaborative creation of openEHR archetypes and templates by clinicians and modelers, their governance, and the automated creation of software artifacts from them, which can be used in application systems of all kinds.

openEHR enables the definition of a complete EHR along with methods to structure the EHR and manipulate its contents, for example, by writing (adding) compositions to the record. It also includes versioning. The open specification includes openly available REST APIs for programmers and vendors and supports several implementation/representation formats. Finally, it makes it possible to build a full, vendor-independent EHR within and across health care facilities.

3.7.2.9 Systematized Nomenclature of Medicine Clinical Terms (SNOMED CT)

SNOMED was developed in the 1970s and is maintained by SNOMED International, a non-profit international standards organization. SNOMED CT (Systematized Nomenclature of Medicine Clinical Terms), first published in 2003, is currently the most comprehensive terminology in the health domain and is used in many countries worldwide. Users must obtain a license for use. The nomenclature aims to support the development of high quality, precisely defined content by standardizing medical terms and by providing ontological components such as associations and relationships between encoded concepts and entities.

SNOMED CT supports the semantic interoperability of application systems. It can be used for encoding and exchanging machine-processable data. SNOMED CT codes (like other codes from terminology standards) can be embedded in information models of other standards such as HL7 FHIR or openEHR.

SNOMED CT consists of concepts, descriptions, and relationships. A concept represents a unique clinical concept such as diabetes mellitus (disorder), which has several children concepts, such as diabetes mellitus type 1 or diabetes mellitus type 2, and so on (Fig. 3.30). Each further level in the hierarchy represents a more specific concept (partitive or taxonomic relationship). Associations between concepts may be of different types, for example, is-a (diabetes mellitus is a disorder of the endocrine system) or finding-site (structure of the endocrine system). SNOMED CT also contains synonyms and is available in multiple languages.

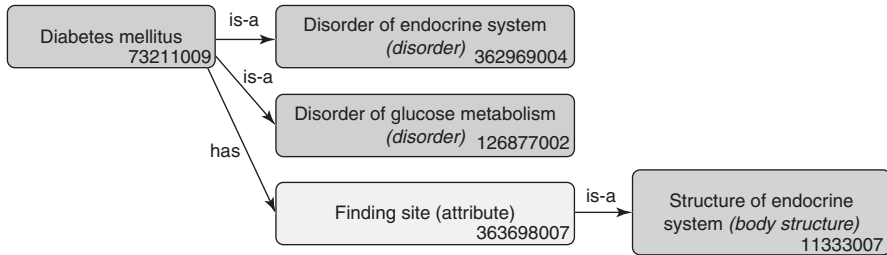


Fig. 3.30 SNOMED CT example of associations between concepts. (Taken from the SNOMED CT browser [14]; use of SNOMED CT is with the permission of SNOMED International)

3.7.2.10 Logical Observation Identifiers Names and Codes (LOINC)

Logical Observation Identifiers Names and Codes (LOINC) is a widely used terminology system for health measurements, tests such as laboratory and clinical tests, and observations. LOINC was developed in the 1990s and is maintained by the Regenstrief Institute in the United States [15].

The system defines unique names, codes, and identifiers that can be used in application systems or within communication messages such as HL7 or DICOM messages as well as in CDA documents (Sect. 3.7.2.2 shows an example of this), openEHR archetypes, or FHIR profiles. LOINC thus supports the semantic interoperability of application systems.

LOINC codes are constituted of six components, for example, LOINC code 1554-5:

1. component (analyte): what is measured, for example, Glucose^{post 12H} CFst,
2. property: the attribute or characteristic measured, for example, mass concentration (MCnc),
3. time: the point or interval of time,
4. system (specimen), for example, serum/plasma,
5. scale: type of value, for example, numerical quantitative (Qn), such as “125”,
6. method (optional): classification of the measurement/observation method.

The LOINC “long common name” for this code example is “Glucose [Mass/volume] in Serum or Plasma --12 hours fasting.”

3.7.2.11 Clinical Data Interchange Standards Consortium (CDISC)

The Clinical Data Interchange Standards Consortium (CDISC) spans a wide variety of standards, which are mainly—but not exclusively—designed for and used in clinical studies and trials in the pharmaceutical industry. CDISC standards were developed in the late 1990s and are maintained by the non-profit Clinical Data Interchange Standards Consortium.

CDISC standards are free to use and cover different domains and parts of the complete process to develop clinical products such as new drugs. The use of some of these standards is mandatory for reporting and submitting results from clinical trials to regulatory authorities, for example, to the FDA (U.S. Food and Drug Administration).

Common CDISC standards include the “foundational standards,” which cover all necessary processes within research studies. Along the evolution of a research study, these standards cover planning (model for planning: Protocol Representation Model), data acquisition (model for data collection: Clinical Data Acquisition Standards Harmonization), organizing and managing data (e.g., Study Data Tabulation Model (SDTM), model for Questionnaires, Ratings, and Scales), and for data analysis (Analysis Data Model).

SDTM is the standard in which study data are submitted to regulatory authorities. Apart from these foundational standards, CDISC also supports a controlled terminology and includes widely used standards for data exchange, such as ODM-XML.

By employing these CDISC standards, researchers can achieve syntactic and semantic interoperability, share their data and metadata, and use a variety of software solutions (e.g., different EDC systems and analytical tools) along the study process.

3.8 Logical Tool Layer: Types of Integration

Interoperable application systems can be joined together, i.e., integrated, in such a way that their combination has new properties that the application systems did not have on their own. Integration is a union of parts making a whole, which—as opposed to its parts—displays a new quality.

Integrating application systems lead to integrated health information systems. An integrated computer-based health information system typically offers better support for functions and business processes than a non-integrated information system consisting of isolated application systems. In non-integrated information systems, users may, for example, need to manually transfer data from one application system to another, or inconsistencies between data representations and semantics may exist that hinder the multiple use of data.

There are various ways of integrating application systems, resulting in different types of integration. Each type of integration leads to specific new properties of the information system. In this section, you will learn about the following types of integration in more detail:

- data integration,
- semantic integration,
- user interface integration,
- context integration,

- feature integration,
- process integration.

Note that the basic prerequisite for all types of integration is that the application systems to be integrated must be technically and syntactically interoperable so that communication is possible at all.

While interoperability makes a statement about a single application system or about a single application software product and its specific properties and capability, integration always refers to a set of application systems and the way they are connected in a specific information system.

3.8.1 Data Integration

Data integration is achieved in integrated health information systems when data that have been recorded and stored once in one application component are made available in a coherent and uniform way wherever they are needed, i.e., in other application components, without having to be reentered. Data integration has two effects:

- Data on different entity types being recorded and stored in different application systems can be combined for joint analysis and use in one particular application system.
- New data on entity types need to be recorded, changed, deleted, or otherwise edited just once—even if the data are to be used in several application components.

In DB¹ architectures, i.e., architectures with a central database, data integration is comparatively easy to achieve. *Open platforms* are a technology that can be used to implement this even in (ACⁿ, Vⁿ) architectures (Sect. 3.9.3).

In case of a decentralized DBⁿ architecture, data integration and a uniform view on the data is more challenging. Transaction management and communication servers are integration technologies suitable to meet this challenge (Sects. 3.9.1 and 3.9.2).

Data integration helps to increase data consistency (Sect. 3.5) and to reduce unnecessary double data entry:

Assume that Mr. Russo was treated in the cardiology department, and a detailed medical history was collected and entered into the *MDMS*. Now Mr. Russo's health is deteriorating dramatically, and he must be moved to the ICU of the same hospital. If the medical history needs to be documented again, in case the information from the cardiology department cannot be accessed, there would be no data integration. If the ICU can assess and add data to the existing medical history, regardless of where and in which application system the already existing data was entered, data integration is achieved. In this example, the medical history may be sent from the *MDMS* to the *PDMS*, either directly or via respective integration technology and tools (Sect. 3.9).

To support data integration with non-computer-based application components, computer-based application components need to be able to print out data or to scan data such as paper-based documents (e.g., signed patient consent documents or a paper-based order entry form).

3.8.2 *Semantic Integration*

Like data integration, *semantic integration* is concerned with data. Semantic integration is achieved when semantically interoperable application systems actually use the same system of concepts, i.e., they interpret data the same way. For example, when the sending application system uses the abbreviation “GOT” for a lab value, but the receiving application system does not know this code and uses “ASAT” for the same concept instead, then semantic integration is not achieved.

To achieve semantic integration of application systems, the application systems must agree on the way that information is represented, for example, within a hospital or a network of care facilities. This means that there needs to be an agreement on a common information model, for example, on Mr. Russo’s blood pressure. This agreement comprises two aspects: We need to agree on how to actually represent the data values (e.g., systolic and diastolic values) by data. And we also need to agree on the underlying clinical concept and represent metadata such as the context of the measurements (resting blood pressure or exercise test), the measurement method (cuff or intra-arterial on an ICU), and so on. Contextual information can make a huge difference. For example, for a *clinical decision support system (CDSS)*, which tries to interpret the blood pressure values to determine whether the value is normal or abnormal in a specific situation. Development of such information models is time-consuming multi-professional work, and agreeing on and maintaining them requires effort and strict governance of models. In the long term, however, these information models reduce tedious and costly mapping work and maintenance of mapping tables between application systems. Furthermore, it enables the reuse of data in contexts that differ from the original scenario in which the data was recorded, for example, for research or clinical decision support (CDS).

Thus, for semantic integration, the application systems must agree on some common terminologies. However, using terminology systems such as LOINC, SNOMED, or ICD (Sect. 3.7.1.3) does not guarantee semantic integration. Instead, for real semantic integration, application systems must also agree on common information models. Such common information models contain a set of abstract concepts (e.g., patient, case, observation, entry, series of pictures) which are close to our concept of entity types and their relationships among each other. They thus try to formally represent the reality of health care through a set of concepts and their relationships. If two application components agree on the same information model, exchange of information and thus semantic integration are facilitated. Both openEHR (Sect. 3.7.2.7) and HL7 FHIR (Sect. 3.7.2.3) enable the creation of common information models, derived from a reference model.

Semantic integration can be elegantly and thoroughly supported by health data dictionaries. Such medical data dictionaries are central catalogs of health-related concepts and terms that offer the possibility of representing the semantic relationships among all data stored in a health information system and of linking that local vocabulary to internationally standardized nomenclatures and knowledge sources (“terminology linking” in openEHR). Such health data dictionaries (sometimes also called Medical Data Dictionaries, MDD) can be independent application components or part of existing application components.

3.8.3 *User Interface Integration*

User interface integration is guaranteed when different application systems represent data and organize their user interfaces in a unified way. For example, different application systems should display the patient’s name always at the same place on their user interface. Icons for patients should code gender with the same colors. Alerts and warnings should be presented using the same colors and layout. Or the birthdate of a patient should also be displayed in the same format (e.g., yyyy-mm-dd).

The responsibility for integrating application systems in such a way that user interface integration is achieved lies with *tactical management of information systems* (Sect. 4.4). Especially at the specification stage of a new application software product, the relevant user interface requirements must be clearly described. This can be supported by user interface guidelines that are commissioned by standardization, governmental agencies, or leading health IT vendors.

3.8.4 *Context Integration*

Even if data, semantic, and user interface integration is ensured, problems can arise when users use multiple application systems on one workstation, as the user and patient context may become lost through the change from one application system to another. *Context integration* is achieved if the context is preserved when the application system is changed. Or, more generally, the aim is that a task that has already been executed once for a certain purpose, such as login to a component or selecting a patient, does not need to be repeated again in the information system in order to achieve the same purpose.

Assume that the nurse Peter Smith first wants to order a lab examination for the patient Jakob Russo at his workstation before documenting certain nursing procedures. The nurse would first log in to the *CPOE system* and create the user context “Peter Smith.” Then Peter searches for the patient Jakob Russo and orders the requested examination for him. This creates the patient context “Jakub Russo.” For the next task, the nurse would have to start the *NMDS*. Without context integration, the user context “Peter Smith” would not be created in this application system

automatically and Peter would have to log in again. And without context integration, Peter would have to again select the patient and restore the patient context “Jakub Russo” before being able to document the nursing procedures. Context integration would be achieved when both login (then also called “single sign-on”) and patient selection (maybe called “single patient look-up”) do not have to be repeated again, even when changing the application system.

Health Level 7 provides a standard for single sign-on and single patient lockup, called CCOW (Clinical Context Object Workgroup).

3.8.5 *Feature Integration*

Feature integration is achieved when features needed in more than one application system are implemented only once and can be invoked by other application systems.

In a hospital, for example, *coding of diagnoses and procedures* is a feature that should be supported by the *patient administration system*, the *OMS*, the *RIS*, and the *LIS*. The health information system would display feature integration with respect to *coding of diagnoses and procedures* if only one application component (e.g., the *patient administration system*) provides the features needed for coding diagnoses and procedures and all other application components can invoke and use these features.

Feature integration reduces the need to exchange data between application systems, reduces the danger of inconsistent data, and decreases the need to train users on multiple application systems.

On the technical level, feature integration can be supported by providing the use of features as services within an SOA (Sect. 3.9.3). Those services can be invoked by other application systems.

Overall, feature integration is an important quality characteristic within heterogeneous health information systems, as it allows features to be shared among application systems.

3.8.6 *Process Integration*

An integrated health information system should support the business processes effectively. From this perspective, *process integration* is indeed the overall vision of integration within heterogeneous information systems, i.e., especially in (ACⁿ, Vⁿ) architectures.

Process integration is guaranteed when business processes are effectively supported by a set of cooperating application systems showing process interoperability.

Typically, as we have seen, functions of a health care setting are supported by many different, yet interrelated application systems. A user must therefore use different application systems for one task.

For example, while writing the radiological report on Mr. Russo, the radiologist must work with the *CIS*, the *RIS*, and the *PACS*. To enable effective work in this process, these application systems should interoperate as transparently and smoothly as possible. If the radiologist receives the best possible support in report writing, without any interruption due to the heterogeneity of application systems, then we can say the information system is process integrated.

Obviously, process integration builds to some extent on the other integration qualities such as data integration (*CIS*, *RIS*, and *PACS* are able to exchange and jointly use patient diagnosis data without the need to reenter it), user interface integration (e.g., *CIS* and *RIS* have comparable user interfaces), context integration (e.g., when shifting between *CIS* and *RIS*, user patient context is maintained), semantic integration (e.g., procedures documented in the *RIS* are automatically understood by the *patient administration system*), and feature integration (the *RIS* invokes the billing function of the *patient administration system*). Good process integration builds on this, supports comprehensive information logistics, and prevents users from transcription and media breaks.

Process integration can be achieved, among other things, through the adoption of IHE profiles (Sects. 3.7.2.5 and 3.7.2.6) on a systematic basis.

Overall, process integration is an important quality characteristic within heterogeneous health information systems, as it describes a situation where different application systems interoperate in an optimal way so that business processes are best supported.

3.9 Logical Tool Layer: Integration Technologies and Tools

To achieve data integrity (Sect. 3.5) and to support various types of integration (Sect. 3.8) in ACⁿ architectures, specific integration technologies are available. These integration technologies support interoperation of application systems and efficient and secure health information exchange between health care providers, and in some cases also with patients.

In this section, you will learn about transaction management, communication servers, vendor-neutral archives, and SOAs as examples of this kind of integration technologies.

3.9.1 Transaction Management

Transaction management ensures that every update of correct data in one or more databases will lead to another state in which the data in these database(s) are still correct. This turns out to be not trivial, especially for complex update operations, the so-called transactions. It is even more complicated in settings of more than one database system, i.e., in DBⁿ architectures.

Transaction management guarantees data integrity and especially data Consistency through **A**tomicity, **I**solation, and **D**urability of any transaction (ACID conditions). Atomicity guarantees that either all of the tasks of a transaction are performed or none of them are; isolation makes intermediary updates of data inside the transaction invisible; and durability stands for persistence of the transaction's results.

The "2-phase commit protocol" was developed for transaction management in DBⁿ architectures. Among other things, this protocol is intended to ensure data consistency in case of data redundancy. In the initial phase, it checks if the transaction can be carried out by all affected application systems. Only if the changes are possible everywhere, they are actually carried out in a second phase in all application systems. To carry out the protocol, the application systems must be tightly coupled by *synchronous communication*, and the database schemata of all involved database systems must be known. For an application system in a health information system, this means that an interface must be provided where both updates and the cancellation of these updates are possible. This is not the case for commercial application components available today. Generally, the individual database schemata of each component are also not known. Due to these reasons, the 2-phase commit protocol to ensure data consistency has usually not yet been implemented within DBⁿ architectures of health information systems.

Nevertheless, to guarantee data consistency, the following more asynchronous approach is typically chosen within a health information system:

For every redundantly stored entity type, one application component is determined as the *primary application system* for this entity type. Thus, data on entities of this type can only be inserted, deleted, or changed in this primary application system. For example, the *patient administration system* is typically the primary application system for the entity types "patient" and "case." Consequently, data on patients and cases can be created, deleted, or changed only in the *patient administration system*. Therefore, this application system is sometimes called the "leading system."

Transactions in a database of the primary application system are carried out without regard to whether the corresponding operations can also be (immediately) carried out in the databases of the other affected application components. *Patient admission* is thereby carried out through the *patient administration system* independent of, for example, what is going on in the *RIS*. It may happen that the *RIS* is not capable of inserting corresponding data on the patient or the case that were sent by the *patient administration system* into its database at the same point in time. The *RIS* is then obliged to catch up on database operations at a later point in time.

3.9.2 *Communication Server*

A quite simple way of communication between application systems is the asynchronous exchange of messages. *Asynchronous communication* means that the application system sending a message will continue its tasks without interruption even when awaiting a response message from the communication partner. Message

exchange generates queues of messages that have to be managed. Besides direct, point-to-point communication between two application systems, tools for managing message queuing can also be used. These so-called queue managers support the sending of messages from the sender to the receiver and the distribution of a single message to numerous receivers (multicasting).

In health information systems, queue managers are typically referred to as communication servers. A communication server is an application system standing at the center of the logical tool layer of a health information system (Fig. 3.31). This architectural principle is also called star architecture (Sect. 3.6.4) and can be found in many health information systems.

Generally speaking, a *communication server* is an application system used for the asynchronous receiving, buffering, and sending of messages. It can also be used to transform messages and monitor the traffic between application systems and may provide intermediate storage of messages in case the intended recipient of the messages is not yet available (e.g., system down). An application system can send a message to the communication server over its communication interface. The communication server will then relay the message to one address or many addresses (multicasting) when these application systems are ready to receive. In the meantime, the communication server buffers the sent messages in a queue (message queuing). In the case that the receiving application system is awaiting messages in a format based on a different interoperability standard than the sending application system sends, the communication server can transform the sent message. The application systems have to provide communication interfaces for sending messages to the communication server and to receive messages from it.

In this way, the communication server is a tool to support data integration in a health information system of (DBⁿ, ACⁿ) architecture. Furthermore, communication servers prevent (DBⁿ, ACⁿ) styled health information systems from *spaghetti architectures* and support CP¹ architectures. Application components can be more easily exchanged, thus supporting the expandability of the health information system's architecture.

Some communication servers may also provide *synchronous communication*. For this, the sending application system will pause from the time that it sends a message to the time that it receives the respective answer. In this way, communication servers can be used to invoke services as described in Sect. 3.9.4 and to provide feature integration as well.

In health information systems with DBⁿ architecture and redundant data storage, the application components store the data needed for supporting their functions by themselves. This holds especially for data on the entity types "patient" and "case." Consequently, the communication server is used to ensure that every application system will find these data in its database system whenever the data is needed without having to request this data from the *patient administration system*. The following example describes how the *patient administration system* and the communication server can be integrated based on asynchronous communication and use of the interoperability standard HL7 (Sect. 3.7.2.1) in order to ensure data integration as well as data consistency regarding the entity types "patient" and "case":

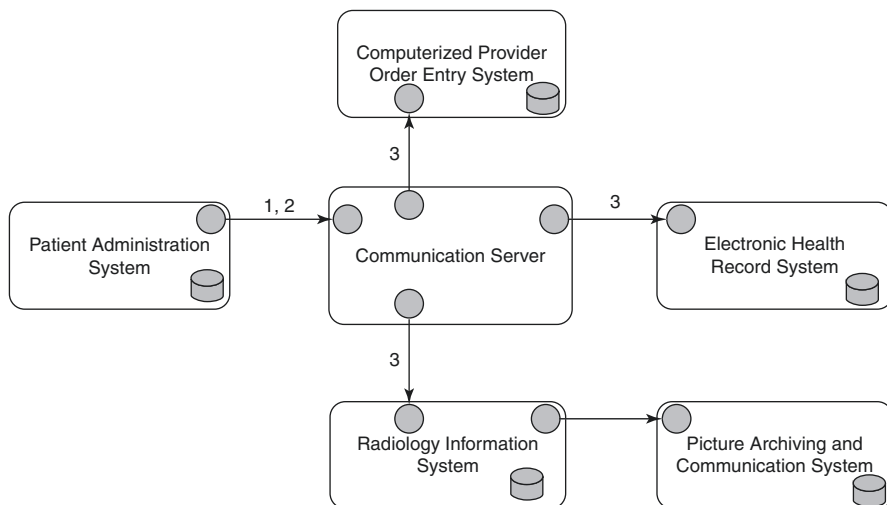


Fig. 3.31 Architectural style with multiple application systems, each with its own database system, using 3LGM² symbols. A communication server links the application components

Mr. Russo has just been admitted to Ploetzberg Hospital in the administration unit. Since the *patient administration system* is the primary application system for the entity types “patient” and “case,” *patient admission* is exclusively supported by the *patient administration system* (Sect. 3.9.1). Even before Mr. Russo leaves the administration unit to go to the cardiology ward, the following messages are exchanged between the application systems (Fig. 3.31 highlights steps 2 and 3):

1. The *patient administration system* creates a message that indicates that Mr. Russo was just admitted to Ploetzberg Hospital (event called “A01” by HL7). This message includes data on the entity types “patient” and “case” using a pre-defined standard (such as HL7).
2. The *patient administration system* sends this message to the communication server.
3. The communication server multicasts this message to all application systems that may need data on the entity types “patient” and “case” during the Mr. Russo’s stay (e.g., *CPOE system*, *EHR*, *RIS*). Usually, nearly all application systems of health information systems need this administrative data.
4. All application components receiving the message will store the data on the entity types “patient” and “case” for Mr. Russo in their own database system in order to have them available in case they are needed.

A similar process takes place when administrative data on Mr. Russo is updated. If his address changes, for example, this is first documented in the *patient administration system*. This application system then sends a message on this update to the

communication server (2) that multicasts this message again (3), as described above. Since the *patient administration system* is the primary application system of patient and case, all updates of related data must be performed using the *patient administration system*.

By following these processes, data consistency is provided and application systems will have actual data available in case they are needed. Above all, users in the radiology department, for example, can always find the administrative data on their patients in their *RIS* without the *RIS* having to make a request to the *patient administration system* beforehand.

3.9.3 Open Platforms and Vendor-Neutral Archives

An application system is called *open platform* if it stores patient data based on open specifications and open information models and provides open Application Programming Interfaces (API) for storing and querying these data. “Open” in this context means that the specifications, information models, and interfaces are published and can be used by any vendor.

An *open platform* follows the idea that both data and metadata are stored in a centralized database, with strict separation between application logic and data storage.

The advantage of the *open platform* approach is that other application systems with different functionalities from different vendors can directly work on the same database (DB¹ architecture). This facilitates both *data integrity* (Sect. 3.5) and *data integration* (Sect. 3.8.1) even in Vⁿ architectures. The interoperability standard openEHR (Sect. 3.7.2.8) provides a universal, agreed-upon information model.

Open platforms were first used in medical image archives and *PACS*, then typically called vendor-neutral archives (VNAs).

For example, after admitting Mr. Russo to the cardiology ward, blood samples have to be analyzed at the laboratory and radiological images have to be taken. Using an *open platform* as in Fig. 3.24, the LIS and the RIS would be able to access the same database as the patient administration system even if they are from different vendors.

To support communication between *open platforms* and other application systems, standards such as HL7 FHIR, DICOM, or IHE profiles are used (Sect. 3.7.2).

While an *open platform* may seem difficult to introduce in a long-grown environment of different application systems with their own proprietary information models, it reduces the ever-increasing cost of mappings on the communication server side whenever a new application system is introduced. It also eliminates costly mapping processes when migrating patient data from a previous to a new application system. Open APIs also enable new vendors to integrate their application software products in given information systems by using the agreed information models and thus enable a growing “application ecosystem.”

3.9.4 *Service-Oriented Architectures*

Service-oriented architectures (SOAs) represent architectural styles of health information systems that provide the means and tools to use services for the collaboration of application systems. A service is an encapsulated feature provided by an application system in order to be invoked by other application systems (Sect. 2.9).

This results in synchronous communication between application systems, meaning that the process in the invoking application system, after starting the communication with the providing application system, is paused until a response from the partner is obtained.

The *patient administration system* may, for example, store data on the entity type “patient” and may provide a service for retrieving name, birthdate, and address of certain patients. In this case, the patient data is called a resource. Other application systems can access this resource by invoking the retrieve service and handing over a certain PIN. They will then obtain the desired data—if access is granted. Another service may be provided by application systems allowing a message to be sent to these application systems by invoking this service and handing over the message.

Therefore, this technology is very good at achieving data integration and ensuring data integrity. In particular, it facilitates the construction of DB¹ architectures using VNAs.

Moreover, the *patient administration system* could provide a service for *patient admission*. The *RIS* could thus invoke this admission service, handing over data on a certain person who should be admitted to the health care facility. As a result, the *patient administration system* executes all necessary actions needed to perform *patient admission*. With such more complex services, feature integration can be achieved in addition to data integration.

When using World Wide Web technology, the resources are identified by certain Uniform Resource Identifiers (URIs). Application systems can provide basic services for GET, PUT, POST, and DELETE operations on their resources by offering the so-called RESTful APIs. For health information systems, resources as well as services for operating on the resources are standardized by FHIR (Sect. 3.7.2.3).

3.10 Physical Tool Layer

Application components are logical tools, and they cannot exist without physical tools as a basis. In an information system, we can describe this basis with the physical data processing systems at the physical tool layer. As stated earlier, physical data processing systems can be human actors, non-computer-based physical tools, or computer systems.

If application components that exchange data and interact with each other are installed on different physical data processing systems at the physical tool layer,

these physical data processing systems have to communicate as well, i.e., integration is also needed there.

We will now have a closer look at typical computer-based physical data processing systems and their integration. You will learn about *physical interoperability* and integration and how challenges in terms of availability and security of application systems and data are addressed in the design of communication networks and the construction and operation of data centers.

3.10.1 Physical Data Processing Systems

3.10.1.1 Servers

Servers are used to provide sophisticated features to clients (Sect. 3.10.1.2). Servers can run databases (database server), they can run the back-end part of application software products (application server), or they can support printing (printer server). Terminal servers run the front-end part of application software products, which traditionally have been implemented on and run by clients. If terminal servers are used, mere terminals (Sect. 3.10.1.2) for displaying output and receiving input are sufficient. Further server types are name servers (for domain name system (DNS) management), DHCP servers (for dynamic IP assignment), mail servers (for email services), and web servers (for website management).

There is a strong trend to virtualize servers. A “real” (i.e., physical) server can simulate lots of the so-called virtual servers. Every virtual server runs a particular instance of an operating system and can be used to implement application software products. Thus, these virtual servers behave nearly identical to physical servers. This approach makes computing power in a data center much more flexible and scalable.

Virtual servers also result from coupling physical servers in order to maximize availability through redundancy. These kinds of coupled servers are called a server cluster. It makes sense to localize the members of the cluster at different sites of the data center (Sect. 3.10.3).

3.10.1.2 Clients and Personal Devices

Clients comprise all data processing tools that are immediately available to the various user groups within a hospital, for example:

- stationary personal computers (PCs) located at defined places in a health care facility (e.g., the PC in the ward office),
- terminals (also called thin clients) which have only functionalities for displaying and for data entry but have no storing capability,
- mobile computers such as laptops, tablets, or smartphones used for mobile information access.

Mobile devices are especially important in health care settings because devices must provide both data and support of functions wherever patients are. This applies not only to health care facilities, but especially to the patient's home environment.

In medicine, there is a plethora of other personal devices in use to capture data from and about individuals, making it available for further processing in application systems. Sensors for recording vital signs such as heart rate, blood oxygen saturation levels or an ECG are not only used in the ICU of a hospital. Smartphones and smart watches also have such sensors and can provide corresponding data. These personal devices also provide sensors for personal identification, such as fingerprinting or retinal eye scanning. Smart cards, which are connected via special readers, are also used for identification.

Overall, large health care facilities such as a 1500-bed university medical center will have about 4000 clients (PCs) and 3000 mobile devices. This causes not only investment costs, but also enormous maintenance costs. The energy consumption of these devices is also considerable.

If we assume a power consumption of approx. 300 watts for one PC, this results in a total consumption of approx. 1.2 MW (megawatts) for all PCs together. The mobile devices' energy consumption must be added.

3.10.1.3 Storage

Health information systems must store not only large amounts of data but also very important data that may be crucial for the patients' health and life. This requires storage devices of high capacity and high reliability. The storage media that are used range from magnetic discs for random access to magnetic tapes and optical media for backup and archiving.

Storage devices, regardless of the media used, are not specifically attached to and exclusively used by certain servers. Moreover, storage area networks (SANs) provide storage services of different kinds to all servers integrated in this network. This technology allows the storage capacities to be scaled up or down quite easily depending on actual storage needs. Additionally, SANs help to maximize availability through redundancy. It makes sense to localize the members of an SAN at different sites of the data center (Sect. 3.10.3).

3.10.2 *Physical Interoperability and Integration by Communication Networks*

The types of integration introduced in Sect. 3.8 relate to the logical tool layer and, more precisely, to sets of application systems. Let us supplement this catalog of integration types with *physical integration*.

Physical integration refers to a set of physical data processing systems and is guaranteed if the necessary physical communication is possible for each required data exchange. This requires having only one computer, for example, a mainframe,

hosting all application systems or having physical interoperability of the physical data processing systems to be connected. For this, the devices must have hardware interfaces for data exchange. In other words, physical interoperability of physical data processing systems is a prerequisite for technical interoperability of application systems that are installed on different physical data processing systems (Sect. 3.7.1.1). In other words, physical integration between physical data processing systems is a prerequisite for any kind of integration between application components.

Communication networks can be implemented by optical fibers, copper cables, and wireless LAN technologies. Different communication protocols such as Ethernet or Token Ring can be used.

The ISO/OSI reference model provides a framework for describing communication between computers at seven levels. Standards for the respective communication protocols are available for each level. With respect to the three-layer graph-based metamodel (3LGM²), these seven levels interconnect the physical with the logical tool layer of a health information system. Level 7 may be considered the logical tool layer which corresponds to the name of the communication standard HL7 (Sect. 3.7.2.1). A set of protocols for each of the seven levels is called a protocol stack.

In order to be able to use the mobile devices required in medicine, appropriate wireless networks are necessary. Wireless local area networks (WLAN) are commonly used inside buildings, both in health care facilities and in private environments. Outside buildings, for example to connect ambulance vehicles but also to feed a limited campus of a health care facility, the same mobile networks that are also used for mobile telephony are used. Currently, fifth-generation mobile networks (5G) are being introduced. These enable high data transmission rates but require a close-meshed network of antennas. The health effects of the associated electromagnetic fields are subject to controversial debate.

The network that exclusively connects the physical data processing systems of a facility is also called intranet. Extensive security measures are required at the interface between the intranet and the internet to prevent unauthorized access to the physical data processing systems within the intranet from outside the facility. These include, in particular, the so-called firewalls. Firewalls are physical data processing systems on which incoming and outgoing communication is monitored by software.

To optimize security, the intranet itself is also partitioned into further security zones. Depending on the need for protection and the permitted user group of application systems, the physical data processing systems on which these application systems are installed are assigned to such security zones. Communication into and out of a security zone is also monitored using dedicated hardware.

3.10.3 Data Centers

The servers must fulfill special requirements related to availability, stability, performance, maintainability, and redundancy. For health care facilities, this can only be economically achieved if the servers are centralized in a data center.

The reliability of the data center, with all its equipment and physical data processing systems, is the very basis for the reliability of the health information system as a whole. Although system stability of application components deals with the quality of the software used, system stability must be enhanced by redundancy. Redundancy is not needed at the logical tool layer, however, but at the physical tool layer.

This requires data centers that provide at least two redundant sites as well as mirrored servers and storage devices at each site, a stable energy supply, high-capacity air conditioning, server rooms that are burglar-proof and disaster-proof, and effective means for fire protection. Even if an information management department ensures an effective backup, the continuous operation of the data center through appropriate redundancy is of utmost importance.

Data center staff needs software tools for supervising proper operation of servers, communication network components, PCs and terminals, and the application components running on top of these physical tools.

For two redundant sites of the data center housing the servers of a large academic center, one must expect an energy consumption of 0.5 MW. This energy consumption results not only from the power consumption of the servers but also from the energy requirements of the air conditioning systems, which must dissipate the electrical energy converted into heat. Consideration should therefore be made as to whether the waste heat from the computers can be used for other useful purposes such as heating the facility's domestic water.

These and the previously mentioned figures on client energy consumption alone make it clear that medical informatics specialists, and information managers in particular, not only have responsibility for good information systems but also for the impact of information systems on the global climate.

Not only the high energy consumption, but also the major technical challenges involved in operating a data center, suggest that smaller health care facilities in particular should consider whether servers in the cloud, and thus commercial data centers, can be used instead of own data centers. However, this requires stable and sufficiently powerful communication links. In addition, the data protection regulations of the respective country must be observed. Conversely, large health care facilities can consider whether their own data center can be offered to smaller surrounding facilities for use. The large facility could then become a cloud provider for the smaller facilities.

3.10.4 Data Security

Data security means protecting the data of an information system both from destruction and loss and from illegal use by unauthorized persons. This is particularly important in health information systems for two reasons. Firstly, health data is highly sensitive and represents intimate information about individuals that must never fall into the hands of third parties without the consent of the individuals concerned. Secondly, a health care facility is at risk of becoming completely incapacitated if the

data in the patients' health records can no longer be accessed. This can cost lives. Recent cases of the so-called ransomware attacks on hospitals, for example, illustrate the associated dangers. In ransomware attacks, data is encrypted by criminals and a key for decryption is only made available after payment of a ransom.

It is therefore imperative to invest in protective measures at all levels. For example, as mentioned above, communication networks must be protected through their structure and suitable hardware and software, data backups must be performed and the data copy must be protected against attacks, and highly qualified personnel must be available. We can only hint at the required know-how in this textbook and therefore refer to the relevant, current technical literature and the training and continuing education programs on data security.

In health care facilities, it is the responsibility of the management of information systems to ensure data security, while corporate management must provide the necessary financial resources for this purpose.

In the private environment, this responsibility lies with the individuals themselves. It is the task of politics, but also of the professional societies for medical informatics, to ensure that people are informed, educated, and advised.

3.11 Examples

3.11.1 *A Reference Model for the Domain Layer of Hospital Functions*

The reference model for the domain layer of hospital information systems consists of the subset of functions and entity types described in Sects. 3.2.3 and 3.3 that are relevant for hospitals. The reference model focuses on the activities in *patient care*. Thus, the main enterprise function is *patient care*, and there are maintenance functions supporting *patient care* such as *supply management*, *scheduling and resource allocation*, *hospital administration*, *hospital management*, and *research and teaching* (Fig. 3.32).

Furthermore, the enterprise functions in that reference model bear a relation to each other by entity types which they can update or interpret. To define entity types within the reference model of the domain layer, the HL7-RIM⁵ was used.

The Reference Model for the Domain Layer of Hospital Information Systems is available as a 3LGM² model⁶ and for this reason can be immediately used for modeling hospital information systems. Following the definition of reference models in Sect. 2.13, the Reference Model of the Domain Layer can be used as a model pattern for the domain layer of hospital information systems and, additionally, can help

⁵http://www.hl7.org/Library/data-model/RIM/modelpage_mem.htm.

⁶http://www.3lgm2.de//Modelle/Referenzmodelle_und_Beiispielmodelle/RM_DomainLayer_version2_English.z3lgm.

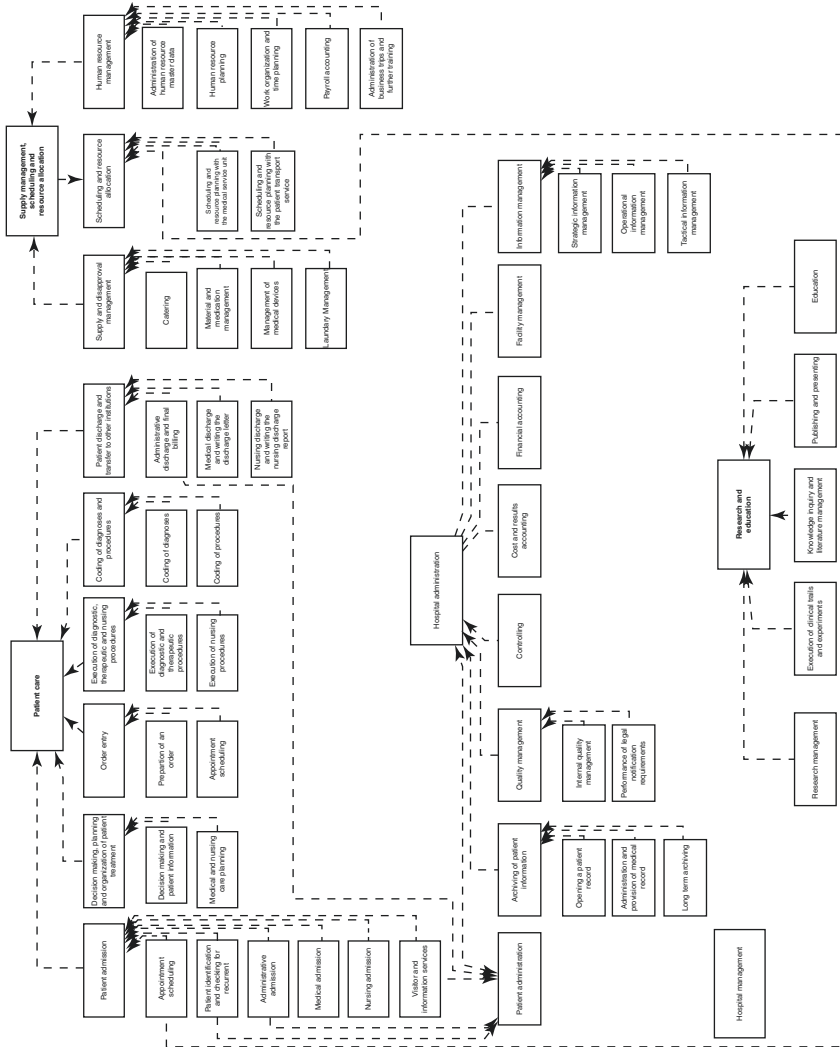


Fig. 3.32 Reference model of hospital functions at the domain layer

to compare hospital information systems by means of a uniform terminology used for the domain layer. This means that, for each enterprise function of the Reference Model of the Domain Layer, it is possible to determine the support by application components in different information systems.

3.11.2 The Domain Layer of CityCare

CityCare is a health care network in the city where Mr. Russo and his family live. In CityCare, Ploetzberg Hospital, Ernst Jokl Hospital, and a specialist medical office for sports medicine cooperate in order to treat patients suffering from joint injuries.

Mr. Russo’s son, an amateur football player, injured his knee joint during his last match. Patients suffering from joint injuries first consult CityCare’s primary care sports physician. If the physician suspects that surgery is needed, she orders MRI diagnostics at Ploetzberg Hospital. After receiving the findings from Ploetzberg Hospital, the sports physician admits patients needing arthroscopic surgery to Ernst Jokl Hospital for further treatment.

Figure 3.33 describes the domain layer of the CityCare scenario. Functions introduced in Sect. 3.3 were used and specialized or decomposed to model the domain layer. The entity types used are described in Sect. 3.2.

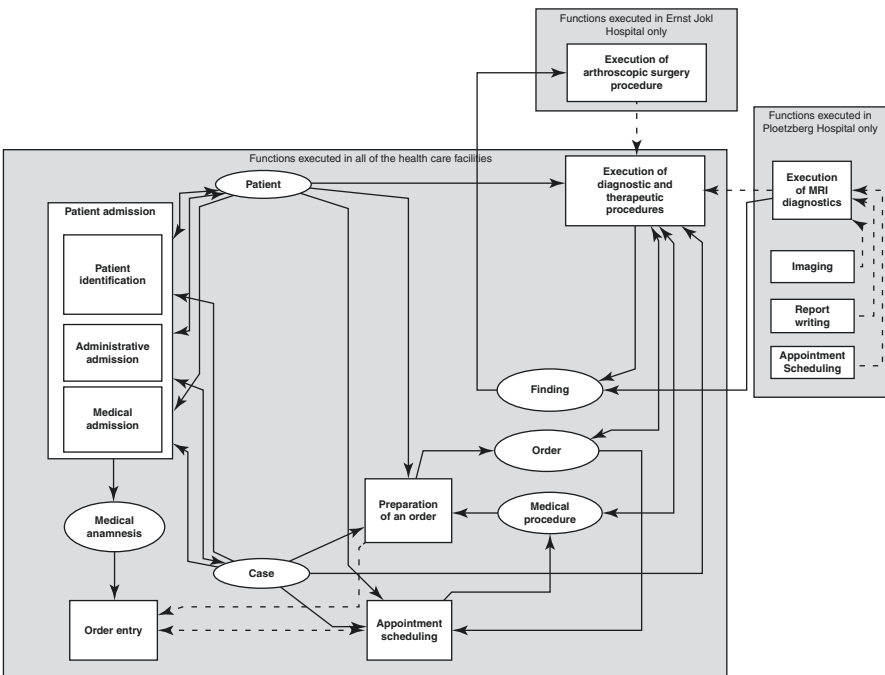


Fig. 3.33 Domain layer of CityCare

All three health care facilities cooperating in CityCare perform activities regarding the functions of *patient admission* and *execution of diagnostic and therapeutic procedures*. These functions performed by each health care facility are modeled only once at the domain layer. Ploetzberg Hospital is the only hospital offering “execution of MRI diagnostics,” whereas Ernst Jokl Hospital is the only hospital offering “execution of arthroscopic surgery procedure.”

Entity types such as “patient,” “case,” and “order” that are used or updated in all three facilities are also modeled once at the domain layer. To visualize different health care facilities, labels (see grey rectangles) were used in the 3LGM² model.

3.11.3 The Logical Tool Layer of CityCare

At the logical tool layer shown in Fig. 3.34, the application systems that are relevant for the described scenario in CityCare and their communication are modeled. For a better overview, the institutional information systems and the jointly used application systems of the network are labeled by grey rectangles.

The specialist medical office has one application system that is used for *patient administration* and medical documentation. Both Ploetzberg Hospital (PH) and Ernst Jokl Hospital (EJH) have their own *patient administration system* and *MDMS*. These are different installations of different application software products

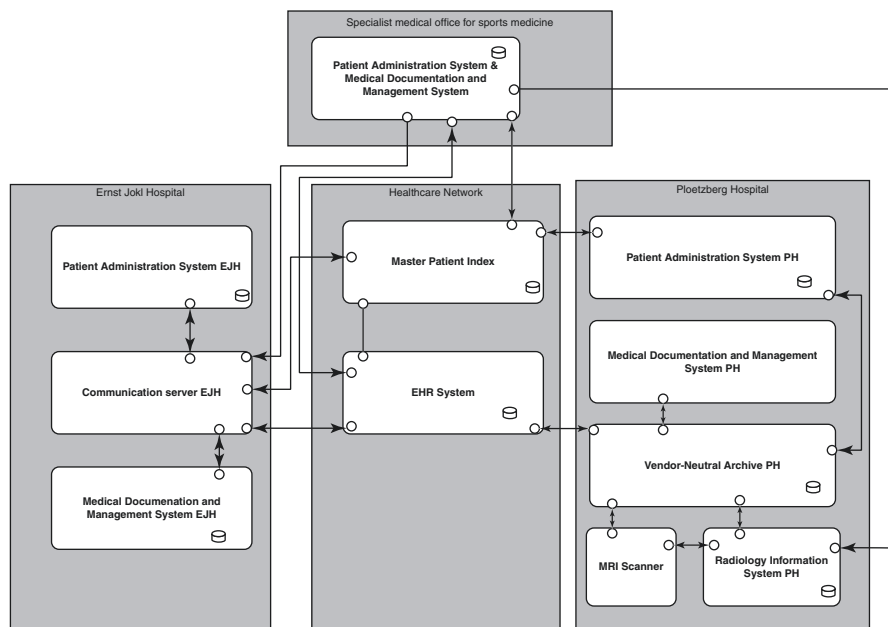


Fig. 3.34 Logical tool layer of CityCare

which are therefore modeled as single application systems. For MRI diagnostics, Ploetzberg Hospital additionally uses an *RIS* and an MRI scanner which is also modeled as an application system. Images that are generated by the MRI scanner and all other medical and administrative data from the *MDMS*, the *RIS*, and the *patient administration system* are stored in the Vendor-Neutral Archive PH, an *open platform*. Thus, these other application systems do not need their own database systems. All three facilities access the central EHR system, a *clinical information system*, of the health care network. This EHR system comprises patient information such as diagnoses, findings, and discharge letters received from all facilities. To manage the PINs from the different facilities, the health care network uses an *MPI* that assigns a transinstitutional PIN for the patients and maps the different PINs of the facilities to this transinstitutional PIN. This is a precondition for being able to build the *EHRs* in the health care network.

The matrix view in Fig. 3.35 shows which functions are supported by which application systems in CityCare. The application system of the specialist medical office, for example, supports six different functions. We can also identify functional redundancies. For *administrative admission*, for example, all three facilities use different application systems. Within a health care network like CityCare, it would also be possible to support this function with one application system that is used by each facility. Application systems serving solely as an integration technology, such as the *EJH* communication server, do not need to be linked with functions at the domain layer in 3LGM² models. For the *VNA* and the *EHR* system, no supported functions are modeled either—finding appropriate functions is part of the exercise in Sect. 3.12.6. For the correctness of the model, application systems are only assigned to “leaf functions” but not to superordinate functions such as “execution of MRI diagnostics.”

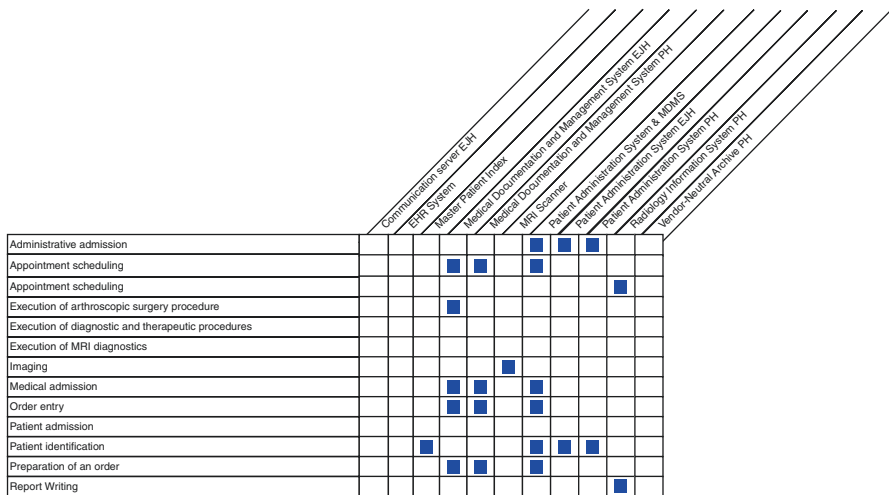


Fig. 3.35 Matrix view visualizing the relations between functions and application systems in CityCare

The architecture of Ernst Jokl Hospital (in the left of Fig. 3.34) can be described as a star-based (DB^n, AC^n, V^n) architecture. There are at least two major application systems (AC^n), probably from different vendors (V^n), with their own respective database system (DB^n). The application systems—the patient administration system and the *MDMS*—exchange data via a communication server (star-based architecture) using HL7 V2 messages. Ploetzberg Hospital also features a star-based architecture which is, however, different from Ernst Jokl Hospital’s architecture. At the center of Ploetzberg Hospital’s (DB^1, AC^n, V^n) architecture, there is a *VNA* implemented as an *open platform* based on uniform information models in the openEHR standard. The hospital’s interoperability experts control these information models and stipulate and enforce their use whenever a new vendor system is added. The *RIS* at Ploetzberg Hospital’s radiology department communicates with the MRI scanner using the DICOM standard. The DICOM images generated by the MRI scanner as well as the radiological reports that are documented with the help of the *RIS* are stored in the *VNA*.

The institutional information system of the specialist medical office has a (DB^1, AC^1, V^1, C^n) architecture because it only consists of a single application system with one database system. The application system combines the functionalities of a *patient administration system* and an *MDMS*. It could therefore be regarded as the *CIS* (or *EHR*) of a medical practice (Sect. 3.4.15). This comprehensive application system sends orders for radiological examinations or orders for surgical procedures to Ploetzberg Hospital or Ernst Jokl Hospital, respectively. For this, it uses the HL7 V2 standard.

All three care facilities work together in the CityCare health care network, which provides a centralized *EHR* that can receive as well as provide patient EHR data on demand. For data persistence, the central *EHR* implements the openEHR standard for highly structured EHR data and an additional database system for documents. To receive patient data, the *EHR* provides different interfaces for the three facilities in the network: an HL7 FHIR interface for Ernst Jokl Hospital’s communication server, an HL7 V2 interface for the specialist medical office, and a native openEHR interface for Ploetzberg Hospital. For cross-network identification of patients, the CityCare network has implemented an *MPI* based on an IHE integration profile. In the future, it is planned to implement a portal both for external physicians and for the patients themselves so they can access their data in the central *EHR*.

If Mr. Russo presents at Ernst Jokl Hospital, the local physician will look for available information on Mr. Russo in the hospital’s *MDMS*. The system can then send a query to the CityCare network, asking for available information on Mr. Russo. The query is sent via Ernst Jokl Hospital’s communication server to the network’s central *EHR system* using the unique PIN for Mr. Russo. The central *EHR system* sends back Mr. Russo’s data using the HL7 FHIR standard to the Ernst Jokl Hospital communication server and the data is forwarded to the EJK *MDMS*.

3.11.4 The Physical Tool Layer of CityCare

At the physical tool layer of CityCare, parts of the data centers of Ploetzberg Hospital and Ernst Jokl Hospital as well as a joint data center of the health care network are modeled (Fig. 3.36). The specialist medical practice only houses three PCs on which the patient management and MDMS can be used. The application system is installed on an application server which is hosted by Ernst Jokl Hospital's data center. Both hospitals also host application servers on which the application systems of the respective hospitals are running. For storing data, SANs are used. In the data center of the health care network, there is an application server for the EHRS and another server for network management.

The relations between the logical tool layer and the physical tool layer are visualized by a matrix view (Fig. 3.37). The specialist medical practice only houses three PCs on which the patient management and MDMS can be used. The application system is installed on an application server which is hosted by Ernst Jokl Hospital's data center. Both hospitals also host application servers on which the application systems of the respective hospitals are running. For storing data, SANs are used. In the data center of the health care network, there is an application server for the EHRS and another server for network management.

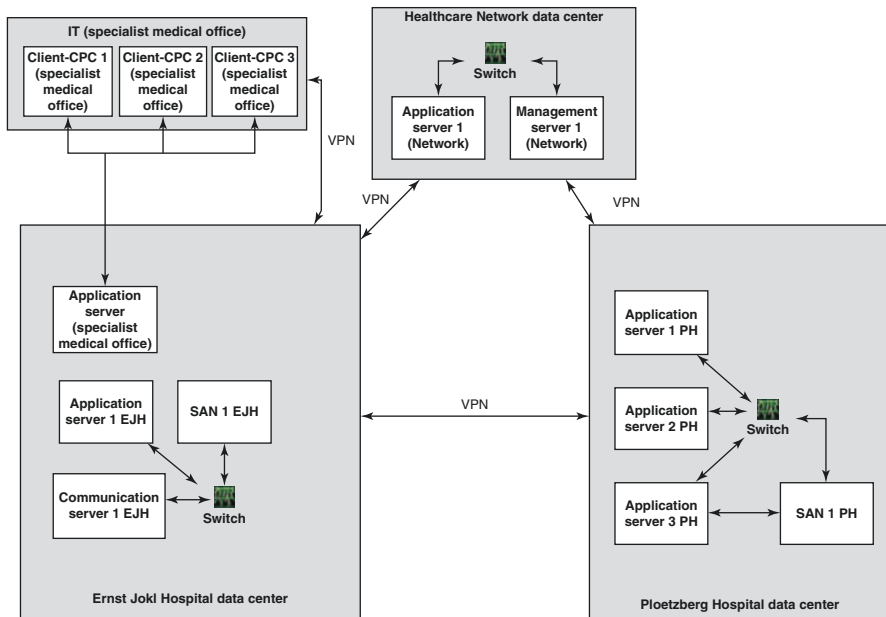


Fig. 3.36 Physical tool layer of CityCare

	Application server (specialist medical office)	Application server 1 (Network)	Application server 1 PH	Application server 2 PH	Application server 3 PH	Client-PC 1 (specialist medical office)	Client-PC 2 (specialist medical office)	Client-PC 3 (specialist medical office)	Communication server 1 E/JH	Healthcare Network data center	Management server 1 (Network)	Plaza/berg Hospital data center	SAN 1 E/JH	SAN 1 PH	Switch	Switch
Communication server E/JH																
EHR System		■														
Master Patient Index		■														
Medical Documentation and Management System E/JH			■													
Medical Documentation and Management System PH				■												
MRI Scanner																
Patient Administration System & MDMS		■				■	■	■								
Patient Administration System E/JH			■													
Patient Administration System PH				■												
Radiology Information System PH					■											
Vendor-Neutral Archive PH				■												

Fig. 3.37 Matrix view visualizing “installation” relations between application systems and physical data processing systems

3.12 Exercises

3.12.1 Domain Layer: Differences in Hospital Functions

Look at the functions presented in Sect. 3.3.2. Now imagine a small hospital (e.g., 350 beds) and a large university medical center (e.g., 1500 beds). What are the differences between these hospitals with regard to their functions? Explain your answer.

3.12.2 Domain Layer: Different Health care Professional Groups and Health care Facilities

Look at the functions listed in Sect. 3.3.2. Look at the relationships between the functions and the different health care professional groups (physicians, nurses, administrative staff, others) working in hospitals and medical offices. Select one health care professional group and describe which functions are most important for this group.

3.12.3 Domain Layer: The Patient Entity Type

Look at the entity type “patient” that is interpreted and updated by various functions. Which functions update the patient information, which functions interpret it?

3.12.4 *Logical Tool Layer: Communication Server*

Imagine a hospital information system that comprises four application systems: a *PAS*, an *MDMS*, a *RIS*, and a *PDMS*. The hospital is now considering the introduction of a communication server to improve data integration. Discuss the short-term and long-term pros and cons of this decision. Which syntactic and semantic standards could be used?

3.12.5 *Logical Tool Layer: Integration from the User's Point of View*

During a night shift, a nurse uses the *patient administration system* to conduct the administrative *patient admission*. The nurse then uses the *NMDS* to plan nursing care. Now consider the types of integration presented in Sect. 3.8 and discuss how this nurse would recognize a high (or low) level of data integration, semantic integration, user interface integration, context integration, feature integration, and process integration.

3.12.6 *CityCare*

The following questions can be answered by reading the text and analyzing the 3LGM² figures of the CityCare Example 3.11.

- (a) The *EHRS* and the *VNA* in CityCare are not linked with any function they support. Which function of the domain layer may (partly) be supported by these application systems? Which functions (as introduced in Sect. 3.3) that are supported by these application systems could be added at the domain layer?
- (b) In which database systems shown in the logical tool layer (Fig. 3.34) should the entity type “patient” be stored?
- (c) The *MPI* should receive messages containing PINs (entity type “patient”) from all *patient administration systems*. Why is there no communication link between the *MPI* and the *patient administration system* of Ernst Jokl Hospital?
- (d) According to the matrix view, which functions are supported redundantly in CityCare? Discuss pros and cons of the functional redundancies in this scenario. What redundancies would you resolve and how?
- (e) Which functions in which health care facility cannot be performed anymore if “Application Server 1 Ernst Jokl Hospital” fails? Suggest a change to the physical tool layer that would minimize the risk of missing function support in case a single application server fails.
- (f) For the CityCare network, would it make sense to implement further profiles from IHE? Explain your decision.

References

1. European Parliament. General Data Protection Regulation (GDPR). 2016. <https://eur-lex.europa.eu/legal-content/EN/TXT/PDF/?uri=CELEX:32016R0679&from=EN>. Accessed 15 Jan 2023.
2. Roach KE, Budiman-Mak E, Songsiridej N, Lertratanakul Y. Development of a shoulder pain and disability index. *Arthritis Care Res.* 1991;4(4):143–9.
3. SNOMED International. CT Editorial Guide: Procedure. 2021. <https://confluence.ihtsdotools.org/display/DOCEG/Procedure>. Accessed 15 Jan 2023.
4. World Health Organization (WHO). ICD-11 International Classification of Diseases 11th Revision. Geneva: World Health Organization; 2021. <https://icd.who.int/en>. Accessed 15 Jan 2023
5. Cimino JJ. Infobuttons. 2011. <https://people.dbmi.columbia.edu/~ciminoj/infobuttons.html>. Accessed 15 Jan 2023.
6. International Standards Organization. ISO/TR 22221:2006 Health Informatics—Good principles and practices for a clinical data warehouse. 2021. <https://www.iso.org/standard/40783.html>. Accessed 15 Jan 2023.
7. European Parliament. Directive 1999/93/EC of the European Parliament and of the council of 13 December 1999 on a Community framework for electronic signatures. 1999. <https://eur-lex.europa.eu/legal-content/EN/TXT/PDF/?uri=CELEX:31999L0093>. Accessed 15 Jan 2023.
8. European Telecommunications Standards Institute. ETSI White Paper No. 3—Achieving Technical Interoperability. 2008. <https://portal.etsi.org/CTI/Downloads/ETSIApproach/IOP%20whitepaper%20Edition%203%20final.pdf>. Accessed 15 Jan 2023.
9. HL7 International. Health Level Seven Version 2 (V2) Product Suite. 2021. http://www.hl7.org/implement/standards/product_brief.cfm?product_id=185. Accessed 15 Jan 2023.
10. HL7 International. Examples section of the v3Guide. 2008. https://wiki.hl7.org/index.php?title=Examples_section_of_the_v3Guide. Accessed 15 Jan 2023.
11. Digital Imaging and Communication in Medicine. The DICOM Standard. 2021. <https://www.dicomstandard.org>. Accessed 15 Jan 2023.
12. Integrating the Health Care Enterprise (IHE). Making Healthcare Interoperable. 2021. <https://www.ihe.net>. Accessed 15 Jan 2023.
13. International Standards Organization. ISO/IEEE 11073–20701:2020 Health Informatics—Device interoperability—Point-of-care medical device communication. 2020. <https://www.iso.org/standard/78227.html>. Accessed 15 Jan 2023.
14. SNOMED International. SNOMED CT Browser. 2021. <https://browser.ihtsdotools.org/>. Accessed 15 Jan 2023.
15. LOINC from Regenstrief. LOINC The international standard for identifying health measurements, observations, and documents. 2021. <https://loinc.org>. Accessed 15 Jan 2023.

Open Access This chapter is licensed under the terms of the Creative Commons Attribution 4.0 International License (<http://creativecommons.org/licenses/by/4.0/>), which permits use, sharing, adaptation, distribution and reproduction in any medium or format, as long as you give appropriate credit to the original author(s) and the source, provide a link to the Creative Commons license and indicate if changes were made.

The images or other third party material in this chapter are included in the chapter's Creative Commons license, unless indicated otherwise in a credit line to the material. If material is not included in the chapter's Creative Commons license and your intended use is not permitted by statutory regulation or exceeds the permitted use, you will need to obtain permission directly from the copyright holder.

