Radboud University Nijmegen

Information Science

Bachelor's thesis

The IT anamnesis?

Diagnosing software requirements

Author:
H.C.B. (Christiaan) Hillen
Student number: 3028410

Supervisor:
Dr. S.J.B.A. (Stijn) Hoppenbrouwers

June 25, 2012

Abstract

Central to this thesis is the question of what concepts, considerations and general patterns in anamnesis protocols may be applicable to problem analysis in information technology. The medical community has a long-standing tradition in problem analysis in the form of the medical anamnesis whereas the IT community still has a lot to learn about interviewing skills and using a methodological approach to problem analysis as basis for requirements engineering.

As it stands, digital architects do not get enough training in these essential interviewing skills. Also, because of the relative infancy of this field, there is no single universal method of problem analysis as there is in the medical community. The medical method consists of four elements, each serving a particular purpose in the process of problem analysis and finally the formulation of a diagnosis and treatment.

It is my conclusion that the information technology community can benefit from the development of such a universal method, instead of the plethora of tools that are currently in existence and are all trying to compete for the favour of the consultant. This will allow all digital architects to work together, as their method is universal, and focus the development of tools to suit this particular practice, improving quality and reducing quantity. The end result will be a decrease in costs and an increase in success rates for information technology projects.

Contents

1 Introduction

This bachelor's thesis is written by a student Information Science who previously obtained a bachelor's degree in health science. It is from this combined knowledge of the medical and the technical domains that the initial idea for the research question for this thesis was formed. In the medical community there is a long tradition of problem analysis in the form of doctor-patient interviewing, better known as medical history taking or anamnesis. It is through this anamnesis that the medical professional aspires to determine the reasons behind the patient's consultation of the doctor and the analysis of the problem the patient has, begins. In decades of research the anamnesis has been standardized with respect to the elements it contains and how such interviews should be conducted. These general patterns are built upon theories and research and have been formalized into protocols which, with a degree of flexibility, dictate the process of the anamnesis.

In the information technology (IT) community, such protocols do not exist. There is no clear-cut method of problem analysis, nor are there any guidelines in the literature as to how this "IT anamnesis" should or even could be conducted. When it comes to diagnosing the initial needs of a customer and the reasons he decides to contact an IT company for assistance with a problem, there are no standards to which this interaction is held, no protocol to assist the IT consultant in determining what is going on, and what the motives of the customer are. This results in IT projects failing, the wrong problems being solved, or the right problems being solved in a way the client does not agree with. Could IT benefit from the vast knowledge-base on the anamnesis and adapt it to its own field in order to prevent the squandering of resources and provide the customer with a better consultancy experience? This is the thought behind this thesis.

Throughout this thesis, the terms doctor and patient are used to describe these distinct roles in the interaction during the anamnesis, although any medical and paramedical professional is meant here and not all of them refer to their patients as such but prefer calling them clients.

2 Method

Central to this thesis lies the following question: What concepts, considerations and general patterns in anamnesis protocols are applicable to problem analysis in IT? This question implies a few things from which I derive the subquestions to be answered. First off, there is the assumption that there are concepts, considerations and general patterns to be found in anamnesis protocols and so we formulate the first subquestion: What are the concepts, considerations and general patterns in anamnesis protocols? I would like to add the remark that "protocols" are looked at in a wide perspective. Not only protocols will be examined, but also methods of teaching the skills necessary in medical history taking and research that has been done under the banner of "medical history" as part of the clinical process.

The second subquestion that needs to be answered is about the domain of IT problem solving. For this, I assume that there is a theory behind the way IT problems are solved, but it is not a clear one. Because of this, I will build my own theory from literature and experiences. This gives us the second subquestion: What are the concepts, considerations and general patterns in IT problem analysis? The research question implies that there is something in the domain of the anamnesis that is applicable to the domain of IT problem analysis and by looking at the subquestions this apparent overlap is made clearer. The reason for this is that I see the anamnesis as a form of medical problem analysis. This by itself is an assumption, but one that I feel confident enough about to make given my medical background.

The answer for the questions will be sought in literature, both in peer reviewed scientific papers and in textbooks used by medical faculties to teach the method of medical history taking. This choice is made because there are no papers that describe the method of the anamnesis in general, only specifics concerning certain types of patients and circumstances according to the general form of medical research: What (type of) patient is researched; what is the intervention to be researched; what other intervention is this compared to; what is the outcome of this research? In such papers, the anamnesis is only handled on the sidelines in that remarks might be made on what one might notice during the anamnesis of such a specific patient or specific questions one should ask when a specific pathology is suspected.

Because of this, I have to resort to the medical textbooks to get the broader view; the basic method for the anamnesis as being taught without the situational specifics.

3 The medical anamnesis

In the medical community, the exchange of information between medical professional and patient is an integral part of diagnosis and treatment (Maguire & Rutter, 1976; Schouten, 1985). As such, taking the medical history by interviewing the patient is one of the first interactions between doctor and patient in the medical problem identification and solving process. The way in which the anamnesis is conducted is in part determinate to the quality of the information collected, and this in turn influences the quality of diagnostics. Because of the influence the method of anamnesis has on the quality of diagnostics, quality control methods have been set in place in the form of protocols describing the to-be-discussed topics during the anamnesis.

The anamnesis in its current form is organized according to a disease-oriented paradigm (Haidet & Paterniti, 2003). This means that the whole anamnesis revolves around the disease that the patient presents; in other words, the doctor faces a disease first, and the patient comes in second. This adversary needs to be identified properly by questioning the patient and performing tests. On the highest level, this questioning of the patient consists of four distinct elements, that can and do quite often intermingle during the actual interview (Formijne & Mandema, 1979; Bickley et al., 1999).

3.1 The opening, including the chief complaint

Before a doctor starts with questions about the actual complaints, it is customary to introduce him- or herself, find out who the patient is, and explain the purpose of the interview. In case such things have not already been registered beforehand, the doctor will also have to record the patient's address, insurance information and other relevant personal data. This is also the moment first impressions are made, and the doctor will already get a first feel for the patient's mood and state. Although this first contact is a hard topic to research and as such there is little research to be found on the topic, it is a vital part of the doctor-patient relationship-building. The placement of furniture in the room, the way the doctor is dressed and groomed and how he introduces himself can assure the patient or cause distress. Patients expect a crisp white coat and a nonchalantly draped stethoscope over the shoulders of an older male, perhaps with some gray hair, the stereotypical doctor in advertisement and media, and at the very least, the white coat is omnipresent in the real medical world.

Once the data gathering phase is over and the patient is at ease with the setting, or at least as comfortable as possible, the first topic of interest is the Chief Complaint (CC) or the reason why the patient contacted the doctor in the first place. This can be as simple as asking the question "So what can I do for you?" or "What is the reason you decided to consult me?" and a talkative patient will start with their story. With less talkative patients the doctor will resort to a more closed form of question to coerce the patient into telling their story.

The patient presents their view on the CC they have, the reason they decided the consultation of a doctor was necessary. This story is steered by the doctor using open questions in order to obtain information about the patient's symptoms. The questions asked by the doctor are referred to as the differential diagnosis and are aimed towards completing the set of symptoms presented by

the patient in order to form a better diagnosis (Bolden, 2000). One can imagine that there are dozens of ailments have the same common symptoms and it is only through the identification of specific symptoms that distinctions can be made. A simple example is the common cold versus the flu, both have the same symptoms save one: a fever. If a patient presents the common set of symptoms, it is up to the doctor to ask if the patient suffers from a fever as well to differentiate between the common cold and the flu. This is an important step in diagnosing the patient and is known as the differential diagnosis. Sometimes this part is explicitly mentioned in the patient's medical report, making it clear that the doctor did think of certain pathologies, but ruled them out on account of certain symptoms. This is especially true with students who are still learning to diagnose, as it shows their teachers that they did not just guess the right diagnosis but applied proper deduction.

The story the patient presents and the complaint described herein is atypical, it is rare to have the complete set of symptoms being described by the patient as complaints in his initial story. Many symptoms are simply ignored or not noticed by patients making diagnosis harder (Geld, 1980). Also, the patient is by no means obliged to tell the truth, or to even tell the doctor everything that is on his mind. Therefore, whatever the patient says should not be taken to much at face value, but rather from the point of view behind his manifest statements (Deutsch & Murphy, 1961). It is only after more dedicated research by the doctor that a complaint can be identified as a certain symptom. It is the doctor's task to decide what complaints are relevant for the diagnosis and this obviously takes skill and experience.

The goal of the CC phase is to identify the reason for the consultation and to establish a professional relationship. When this is done, the phase is followed by a more detailed interview on the specifics of the CC in order to distinguish symptoms from complaints and to get a better image of the CC.

3.2 History of the present illness

Once the patient has given his initial story, the doctor needs to start make sense of the complaint. This is done by asking the appropriate questions about the History of the Present Illness (HPI) and link those answers to the CC. The CC can be just one aspect of the present illness and it might be that the patient is a chronic patient who has been suffering from the same disease for decades and the CC is a complication linked to the illness or something completely unrelated.

Whichever is the case, the CC and HPI are combined and the six attributes of the currently experienced symptoms are examined (Bickley et al., 1999). These are:

- What is the location of the symptom.
- What is the quality of the symptom. What is it like.
- What is the quantity of the symptom. How bad is it.
- What is the timing of the symptom. When did it start, how long does it last, is there a pattern during the day.

- What is the setting in which the symptom occurs, including environmental, personal activities, and other circumstances surrounding the illness and how do these occurrences influence the activities of daily life
- Are there any associated manifestations.

Another version of these attributes are found in the mnemonic OPQRST (onset, provocation, quality, radiation, severity, and time). Although these are typically used by novice students (Lacasse & Maker, 2008). In examining these attributes, the doctor needs to keep in mind that asking leading questions to force patients to choose from among a limited number of responses will distort the patient's story (Barsky, 2002). Because of this, the questions asked are usually open, unless a specific detail must be confirmed of denied in which case the doctor will use closed questions.

To illustrate this further, a fishing analogy is used: There are two ways to catch fish, with a net and with a line. In case of net-fishing, there is no good way to predict what will be caught, and there will be a lot of debris coming along in the net. It is also very time consuming and the desired type of fish might not be found. With line-fishing however, the angler can adapt his tackle to the circumstances and species he is trying to catch. Going back to the medical history, casting a net will lead to a huge but heteroclite catch whereas using a lure (specific questions tailored to rule the diseases in the differential diagnosis in or out) will catch that specific fish if it is there (Lacasse & Maker, 2008).

A problem with interviewing people and asking questions about things they did not think of before, or fishing with a line, is that they will alter their reports of the past to make them consistent with their current state (Aneshensel et al., 1987). As such, patients will link prior complaints to the CC, and will continue to do so on later interview dates. At these later dates, patients will link previously unlinked complaints to their illness as they attempt to keep their story consistent and to make sense of their situation. Recalling events becomes less reliable as the events are further back in time, resulting in even more blurring of their story (Barsky, 2002).

3.3 Review of systems

After getting the patient's story straight and having asked all the questions directly related to the CC and HPI, and the doctor feels confident that he got all the information the patient can provide, the doctor moves on the Review Of Systems (ROS). The ROS consists of a long list of simple questions about the patient related to the state of systems of the human body such as skin, head, eyes, ears, respiratory, gastrointestinal and urinary systems. It also contains general questions such as questions about recent (unexplained) weight gain or loss and sleeping habits as these can indicate that something might be wrong. It is not uncommon for patients to not have noticed gradual changes until the doctor asks these questions, resulting in a "Now that you mention it" response.

This is done in part to question the clearly related systems, and in part to identify possibly related symptoms that the patient has no knowledge of given his (assumed) lack of medical training and which were missed by the doctor in the CC and HPI phases. The ROS is not always done completely, it is up to the doctor's discretion to decide what should and what should not be asked, given their knowledge of pathology and the already presented symptoms during the

CC and HPI. Sometimes the CC and HPI are so clear that the ROS is kept to the basics as there seems no complex underlying problem.

Although the CC and HPI phases are universal among all medical professions, the ROS phase is quite distinct. For instance, a physiotherapist has left-/right-handedness on his ROS list, whilst an oncologist typically has no use for such information.

3.4 Past medical history

Once the complete picture of the present state of the patient and his current CC and its HPI are visible, the doctor needs to build up an image of the past of the patient as a CC is almost always related to past ailments or past ailments need to be taken into account when treating the current one. For instance, patients with diabetes (diabetes mellitus) require special precautions when they need an operation, and someone with a hernia (prolapsus disci intervertebralis) should not receive certain types of manual therapy. Unless a detailed record is present on the previous medical treatment and consultations the patient had, this process can be time-intensive. If a record is present, the doctor will briefly review the history and see if new details need to be added. Keeping in mind that, as mentioned in the HPI section, as events go further into the past, they become less clear and memories of them might become altered and incorrect to comply with current experiences of the patient. Written medical records therefore are to be trusted above the patient's memory when it comes to old events. Although this is a personal observation, it has proven correct on various occasions, especially when it comes to patients with ailments that have affected memory, such as Korsakoff's syndrome. In this light, it is essential that the records are kept up to date, and are not meddled with.

This final phase ends with a review of all the information that was exchanged in order to see if anything is missing, incorrect or needs further clarification. When all is said and the verbal anamnesis is at its end, the physical examination can take place to further examine the patient and his ailment. After the physical examination, the doctor will form his initial or definitive diagnosis depending on the need for lab results or other diagnostic tools such as magnetic resonance imaging or computerized tomography. The diagnosis is shared with the patient and the process of designing a treatment can begin. This concludes the medical problem analysis.

4 Teaching of and training in the anamnesis

Over the last century there have been significant changes in the way medical history has been taken and recorded although the four elements have remained a constant. In teaching and training the anamnesis, before the 1940's it was difficult to learn from the exchange between the patient and doctor because there were no records other than memory which is unreliable when it comes to exact formulations of questions and responses. It was not until the 1940's in psychiatry that doctors started to commit the anamnesis to audio recording for later reference. Case reports including the discourse never became the custom; only the doctor's account of the patient's narrative was recorded in the written media. For the most part, the patient's medical record was little more than a card containing the fee charged, and was a sparsely penned notation of complaints, results from the physical exam and the drugs prescribed (Stoeckle & Billings, 1987). Over time this has changed, into detailed (electronic) medical records containing notes on every action by medical personnel and every contact the patient has had with doctors.

4.1 Methods of teaching

The earliest teachings of the anamnesis were little more than sitting in with a trained doctor and noting how things were done, only to repeat them later on in one's own practice. With the use of audio records it was possible to analyse the anamnesis in more detail and ask questions after the session about certain details and ways of questioning, allowing for great specificity on conduct.

At first, the only topics handled in the anamnesis were the strictly biological aspects of ailments that were observable by the doctor himself. Unlike today, where at the very least, observations and their logical consistency according to knowledge of pathology are checked against lab results. But not only the biological aspect plays a role these days, the psychological and sociological aspects of the patient are also taken into account, resulting in the biopsychosocial model; a more holistic view of pathologies that has been in use since the 1970's, although the approach is still disease oriented.

Teaching the methods of the anamnesis thus first were simply oral exercises, which evolved into demonstrations as the student was allowed to sit in, or at least view or hear a recording, and ended up in its current form where students actually interview patients and are supervised while doing so. This is however, the culmination of training, which starts with case reviews, basic training in interviewing techniques and practising on other students and actors. All medical students thus learn the basic phases of the anamnesis, reading medical charts, and communication with patients before they ever actually encounter one. It is in this preparation that students learn what to ask, why to ask, how to ask, how to act, and what not to do. Only after acquiring at least a basic proficiency in these skills are they allowed to interact with patients. The initial patients to be interviewed have been selected beforehand and it is already known what ails them. Starting with easy and uncomplicated cases and working their way up to the complex cases, students acquire more experience and knowledge on the actual process of the anamnesis and get a better feeling for the clinical method as they go along.

This training provides the student with a great deal of background infor-

mation that will shape the way the student thinks and the way he conducts the anamnesis. But not only the explicit training is a source of information for the student. Also, what questions to ask during the anamnesis can be found between the lines in the medical textbooks, because of their description of diagnostic symptoms and signs of medical disorder. This information can be used implicitly to guide questioning in the search of the diagnosis and the differential diagnosis. It is this vast background knowledge that requires years of training and studying and even more years to have all this knowledge mentally available at a moment's notice.

4.2 Social interaction and empathy

It was long believed that medical students - being human beings - had enough experience from daily life to handle the social interaction between doctor and patient (Stoeckle & Billings, 1987). This however, is not the case as patients and students come from all walks of life and differ greatly in their social and empathic capabilities. Also, in life we are considered equals when talking to others, or are so on most occasions. In the anamnesis however, there is a clear distinction between doctor and patient, wherein the doctor has authority and is seen as an expert by the patient, which is made explicit by the uniform of the doctor, his pristine white coat.

Because students typically do not encounter such a relationship from the perspective of being the doctor until medical school, it is wrong to assume that they will simply know how to act in the correct manner without some form of training. This training entails how to act around patients, and explains in detail the doctor-patient relationship and the responsibilities and ethics that come with it.

The doctor must take care to keep his distance, in such that the relationship between doctor and patient is always a professional one. On the other hand, a doctor soliciting the patient's perspective has a positive effect on a patient's adherence, trust and satisfaction, so this is a delicate balance between keeping a distance and showing empathy (Haidet & Paterniti, 2003).

4.3 Asking questions is hard

In the anamnesis, with one patient you can simply ask "What can I do for you?" whilst with another, you need to be more specific "What is the reason for your consultation" and sometimes even "I hear you have a problem with... can you tell me more about it?". The method of questioning currently in use in the medical settings comes from psychiatry just as the habit of recording sessions which leads to a better understanding of the anamnesis.

The easiest distinction to be made in questioning is that of the open and closed question. In a closed question, the response is limited to a set of answers already presented or can simply be answered with a yes or no. In an open question, it is not possible to respond with a simple yes or no answer or a selection from a set of given answers. Although this might seem trivial, closed questions lead a patient into answering in terms that have been preselected by the doctor. This robs the patient of degrees of freedom in choosing their own terms to describe what is going on.

However, it is not always needed to restrict the patient in such a manner, and a more "narrative" method of questioning can be utilized. In the narrative approach, the patient is coerced into giving their own perspective of what is going on as this allows their perspective to be expressed in the medical chart. This however, is a difficult task for the medical professional as he needs to keep track of two narratives at the same time. On the one hand there is the patient's perspective, and the other is the biomedical perspective, which has an influence on the way each party understands what is going on. With the narrative approach, a medical history is not so much taken by one party as it is being built by both parties (Haidet & Paterniti, 2003).

4.4 The value of experience

Is experience valuable in taking a medical history? Yes and no. Yes, because it will make a doctor notice subtleties that might otherwise have been overlooked, or combinations of symptoms that are rare and a doctor only encounters once every few years. No, because experience can turn into mindless repetition and "going through the motions" rather than keeping an open mind which will make a doctor overlook subtleties and rare combinations of symptoms that share characteristics with often diagnosed pathologies.

Experience is valuable, but it is by no means an excuse to rush through routines. As exercise therapist you will encounter hundreds of patients with lower back pain and although the symptoms are very much alike, the devil is in the details. Even the treatments can look alike, but need to be tailored to the specific patient. It is from these similarities that guidelines are developed by panels of experienced practitioners in order to describe a best practice. These guidelines are obviously valuable, as long as they remain guidelines not to be strictly adhered to.

4.5 Encountered problems and shortcomings

A study from 1976 showed that over eighty percent of medical students lack consistency in covering the more personal topics of the anamnesis, are to imprecise or use jargon which confuses the patient (Maguire & Rutter, 1976). Two decades later, in 1995, this was still the case and there is no reason to believe that there will be significant changes made in the oncoming years as well (Stewart, 1995). Communication difficulties can be described with reference to problems of diagnosis, a lack of patient involvement in the discussion or the inadequate provision of information to the patient. With over fifty percent of psychosocial and psychiatric problems being missed and in fifty percent of the cases, physician and patient do not agree on the main presenting problem, there is still a lot that goes wrong in doctor-patient communication (Stewart, 1995).

There is a clear difference in the doctor's (biomedical) perspective and that of the patient (personal). As the anamnesis is a part of the therapy, and as such an integral part of the treatment because of its social effects on patients, it satisfies a basic human need for expression (Haidet & Paterniti, 2003). But because of this difference, doctor and patient are speaking different languages. Not only verbally, but also somatically, and only to the experienced interpreters of this somatic language is it as clear as a programming language to the computer programmer (Deutsch & Murphy, 1961). Learning to speak both languages is

part of the training of doctors, but some are more proficient than others and the amount of training differs widely amongst the various disciplines within medicine.

5 Theories behind the anamnesis

Physician education has actually demonstrated that it affects the patient's emotional status, whereas patient education has been demonstrated to affect physical health, level of function, blood pressure and blood glucose levels. In other words, a better trained doctor is a positive influence on the health of the patient. Asking questions about the patient's understanding of the problem, concerns and expectations, and about his perception of the impact of the problem on function leads to a reduction in patient anxiety and symptom resolution. Finally, asking the patient about his feelings reduces psychological distress (Stewart, 1995).

5.1 Content of the anamnesis

As we have seen, the anamnesis consists of a verbal and a physical phase. In the verbal phase, we have the clear distinction between the CC, the HPI, ROS and PMH elements of the anamnesis. Each of these elements has a clear goal to it:

- CC: Setting up the professional relationship and identify the reason for consulting the doctor.
- HPI: Sensemaking of the CC and filling gaps in knowledge.
- ROS: Review all systems to identify other possible problems, though they need not be related to the CC.
- PMH: Uncovering the complete medical history of the patient to put the CC into context.

As we can see from this overview, the doctor works from the present and the obvious outwards in expanding views. First there is only the CC, then the immediate surroundings of the CC, to make sure nothing is hiding in the surroundings, a complete review is done and finally this new episode in the patient's medical lifeline needs to put into place amongst the previous medical events because it just might be related to these previous events. This provides for the complete image of what is going on, in the proper medical context.

This last statement is important as it again expresses the disease-orientated paradigm. Although with the biopsychosocial view, the patient and his social surroundings have become more important, the emphasis remains with the disease.

5.2 Human relationships

It would seem to be stating the obvious to say that doctors and patients alike are human beings, but all too often this is forgotten. A doctor is not a computer that takes an amount of input, lets loose a set of algorithms and produces a diagnosis and treatment plan. Although attempts have been made to create such systems, as of yet, they are not in widespread use because patients benefit from human interaction and feel more comfortable with a human than a machine. The method of the anamnesis is set up in such a way that the patient is made comfortable and the doctor is as professional as possible. The white coat, the

uniform of the doctor, is part of this. It serves as a cover-all for any expressions of personality that might influence the patient, as well as protecting the doctor's clothing and show the patient that the doctor is clean. It is strange however that even in China doctors wear white coats, as in their culture, white is the colour of death and mourning. This would seem an association one would not want to have with doctors.

The initial introduction in the anamnesis serves as a buffer between first contact and the serious matters to be discussed. This allows the patient to calm down if anxious and to get a feel for the surroundings and the person in front of them. The same holds true for the doctor, who needs a few moments to get an impression of the patient so he can align his methods of interaction accordingly. Imagine a doctor treating every patient the same and what effect that would have on different personality types to understand how crucial this initial introduction is.

5.3 Questions and language

As already discussed, both open and closed questions are used in the anamnesis. Open questions are used by the doctor to give a certain direction without leading the patient. Closed questions are used when a specific answer is required to confirm or reject a hypothesis.

Not only how questions are asked, but also what is asked is of importance. The use of jargon can pose problems for patients without medical training. It is up to the doctor to guard against using terms that the patient does not understand, and to frequently ask if the patient understands what he is saying. Explaining complex medical issues to patients without any notion of basic human anatomy and physiology is very hard. Also, in presenting bad news in whatever form, anything said after such a message will not be heard by the patient. Take the sentence "If we don't act now, this benign growth will turn into something possibly malignant." As soon as the word "malignant" is used, even with the adverb "possibly", this will startle most patients in such a manner that further communication becomes a blur to them. The rest of the sentence is also not understood anymore, and the patient might well go home feeling that he has a terminal illness rather than something that can be solved with a simple local anaesthetic and the skilful wielding of a scalpel.

6 The IT anamnesis

This chapter is titled the IT anamnesis to emphasize the parallels between the medical and the IT problem identification interview. As there is very little literature on this phase in the development of IT systems, most will be written from personal experience as a student in Information Science haven received training in requirements engineering and software development. There are however some well-known frameworks for enterprise architecture, a field that is closely related to digital architecture and requirements engineering.

Central to the IT anamnesis is the term architecture, and those concerning themselves with the design of this architecture are called architects, digital architects to be more precise, and they come from a varied background in the fields of computer and information science. A digital architecture plan or description is the result of the IT anamnesis and can be seen as the outline of the treatment in the medical analogy, the goal of the anamnesis. A plan is made in case a new system is needed, a description is made in case an existing system needs to be altered. The definition for architecture is taken from the ISO/IEC/IEEE 42010 standard and reads "fundamental concepts or properties of a system in its environment embodied in its elements, relationships, and in the principles of its design and evolution."

6.1 Enterprise and IT architecture

Enterprise architecture is defined as a coherent whole of principles, methods, and models that are used in the design and realization of an enterprise's organizational structure, business processes, information systems, and infrastructure. Better alignment between business and IT leads to lower costs, higher quality, better time to market, and greater customer satisfaction (Lankhorst, 2009). Thus, IT architecture can be seen as the solution to the problem of structuring IT applications to best suit the business (Britton & Bye, 2004). The difference between enterprise and IT architecture is that enterprise architecture concerns itself with the entire organization and the information flows within it, and IT architecture concerns itself with only a subset, the IT infrastructure that this organization uses. As such, the one cannot be complete without the other, unless an organization does not utilize any IT system or has no information flows. An IT system cannot function isolated from the organization in which it is embedded, so IT architecture also has to encompass a view on the organization (Nuseibeh & Easterbrook, 2000).

In enterprise architecture, the business vision and strategy are translated into effective changes in the organization by creating, expressing, and developing the core requirements, models and principles that are pursued by the organization and how to reach them. In other words, enterprise architecture draws up a concrete blueprint of how the company works, not on paper, but in reality. After creating such a blueprint, changes can be made accordingly to align the company's operations with its core principles if the alignment is suboptimal. As organizations are highly complex, individuals suffer from bounded rationality. Bounded rationality implies that no individual can oversee the entire process and so make decisions based on complete knowledge (Simon, 1965). It is because of this that organizations strive to become closed systems, so that external uncertainties can be reduced or eliminated altogether and predicting

	What?	How?	Where?	Who?	When?	Why?	
Planner							Scope (contextual)
Owner							Enterprise model (conceptual)
Designer							System model (logical)
Builder							Technical model (physical)
Subcontractor							Detailed representation (out of context)
	Data	Function	Network	People	Time	Motivation	

Table 1: A minimal representation of the Zachman Framework

future events becomes easier. Although achieving a complete closed status is impossible, organizations attempt to reduce this complexity by using communication and management tools such as models and guidelines. These models can be created by enterprise architects in order to visualize information flows and departmental interaction and by IT architects to visualize the IT facilities that are in place or can be installed to assist in decision making. These models serve as means of visualizing tacit knowledge both within and about the organization in order to get all participants to talk about the same things.

These models are set up in interaction with the client, and the parallels with patients are clear. In both cases, a model (diagnosis) is made of the current situation and where things need to go from here.

One of the best-known models is the Zachman Framework, which was first introduced in 1987 and is but a logical structure for organizing and classifying the descriptive representations of an enterprise that are significant to the management of the enterprise as well as to the development of the enterprise's systems (Zachman, 1987; Lankhorst, 2009).

The Zachman Framework originally only consisted of the first three columns (What, How and Where) and this was later enhanced with three more columns and is still used today in its current form as seen in table 1.

Each cell within this framework is filled with some form of diagram, list or model that details the specifics of that cell. For instance, in the Designer/Why cell, a rules diagram can be found, and in the Owner/How cell, a process model is placed. The benefit of the Zachman Framework is that it represents the organization as a whole and it is easy to understand. Naturally, the filling of each cell can become quite confusing, but given that it is placed in a certain cell does imply the meaning of this filling and will help to understand the contents. A drawback of this framework is the large number of cells that need to be filled to get the complete picture. The relationship between the different cells is not well-specified as well. That being said, Zachman receives credit for providing the first comprehensive framework for enterprise architecture and his work is still extensively used (Lankhorst, 2009).

The ultimate goal of architecture is twofold: "A formal description of a system, or a detailed plan of the system at component level, to guide its implementation" and "The structure of components, their inter-relationships, and the principles and guidelines governing their design and evolution over time."(Lankhorst, 2012). Within software engineering, there is a single overriding goal: To deliver on-time, high-quality, operational software that contains

functions and features that meet the needs of all stakeholders (Pressman, 2010). From these goals we can see that to reach the goal of the software engineers, the digital architects must get right the requirements or the functions and features the stakeholders want. If they fail, the project is doomed.

6.2 Prior to requirements engineering, first contact with the client

When a client contacts an IT professional, expresses a need for an information system and the IT professional responds to this request, the communication process had begun, not unlike the patient entering the office of a doctor. For this to go smoothly, Pressman (2010) formulates eight principles that guide the practice of the software engineer:

1. Divide and conquer
2. Understand the use of abstractions
3. Strive for consistency
4. Focus on the transfer of information
5. Build software that exhibits effective modularity
6. Look for patterns
7. When possible, represent the problem and its solution from a number of different perspectives
8. Remember that someone will maintain the software

Not every principle here is applicable to the architecture process but let us have a closer look. Principle one can clearly be seen in the Zachman Framework, the organization as a whole is divided into smaller and better manageable pieces to be modelled. This is also the case with the medical anamnesis where the anamnesis as a whole is divided into smaller sections, each with a distinct purpose and product.

Principle two again is quite clear, as models are being used to describe specific sections of the organization, a model to fill each cell of the Zachman Framework. In the medical anamnesis, the doctor might refer to an anatomical model to explain his thoughts or findings to the patient.

Principle three is not that explicit, but rather common sense. Use general patterns that have been field-tested, if it's talking to a client and wanting to know what he wants, or talking to a patient and diagnosing his ailment. There is a process that when followed will guide towards the goal, but stay agile and be ready to improvise when needed.

Principle four concerns itself with user interfaces and the flow of information there and the flow of information within an organization but parallels can be drawn towards the medical anamnesis in the information that makes up the ailment of the patient and the knowledge of the doctor. Not only the information that needs to flow, but also how it flows and perhaps is altered during the transfer need to be kept in mind.

Principle five has no parallel with the medical world but is something that helps improve maintainability, and extensibility. It is quite rare to find an existing organization that is not a tangle of processes that are modular, herein lies a challenge for architects to find ways to represent the organization as a modular whole, in order to represent the individual processes, flows, actors and other elements that make up this whole and interact with each other.

Principle six is very straightforward in all fields, patterns make your life easier once you identify them because you can use them to predict the future, of sorts. Humans are natural pattern seekers, it is what we do in life. The drawback of this is that we tend to find patterns where there are none and it is all simply a matter of coincidence. Patterns and modularity combined also allow you to design standardised products that suit similar settings although each application of such a product will still require tweaking.

Principle seven allows more people to understand the problem and solution, because we are all different and come from different backgrounds we each need a different explanation to make things clear. Giving different representations that all explain the same problem and solution also make sure that the problem and solution are well described.

Principle eight goes for all things if we go up a level. Other humans will be using what you have produced, so make it understandable for them as well. Personal scribblings on a patient's chart might not be legible for everyone, yet they may contain essential information that others will need to know about if for whatever reason the regular doctor cannot continue with the treatment. Likewise for software, you might know how the code fits together, but someone who has not designed it from the ground up will have a hard time understanding it because they did not grow into it.

These eight principles will be a background framework for the entire process, and will not always be thought of deliberately.

First contact for a customer and for a consultant can be very intimidating. On the one hand there is the customer who has a vague notion of what he wants, but does not have the technical skill and language to explain himself, and on the other hand there is the consultant who has to balance technical knowledge and language with the limitations of the customer in order to speak the same language. If you have ever heard computer scientists talk amongst each other you can get an idea of how intimidating this is for the uninitiated. Unfortunately, as a stereotype, computer scientists are not very social beings. Or at least, they are social amongst their kin but find it hard to express themselves in such a way that others will understand them completely. It is the task of the digital architect to speak the language of the computer scientists and at the same time speak the language of the customer, he is effectively a translator between these two parties.

Clients can have the notion that anyone in the IT sector has trouble with using normal language and it is up to the digital architect to convince the client otherwise and reassure him that everything will become quite clear and understandable. The client might even have some form of technophobia and is reluctant to get a complex computer system, but is pressed by his surrounding to do so, just as not all patients want to see a doctor but their relatives insist. The digital architect will have to familiarize himself with the technical proficiency of his client and conform to that level within the first meeting.

6.3 Requirements engineering and the creation of architecture

Software has become deeply embedded in virtually every aspect of our lives, and as a consequence, the number of people who have an interest in the features and functions provided by a specific application can be quite large. As a result of this, getting to grips with the problem is a challenge, but one needed to be undertaken before actually writing any code (Pressman, 2010).

But what exactly is requirements engineering? Rather than giving my own definition I rely upon Zave (1997) for a definition: "Requirements engineering is the branch of software engineering concerned with the real-world goals for, functions of, and constraints on software systems. It is also concerned with the relationship of these factors to precise specifications of software behaviour, and to their evolution over time and across software families."

Everything that comes after the requirements phase builds upon what has been done in this phase. If there is something missing in the requirements, or something is unclear and left to the interpretation of the code writers one can imagine that things go wrong quickly. The pressure to get things right is thus quite high, but the correct definition of all the software requirements is paramount to the success of the project. Many software projects fail because what was built was not in line with the customer's requirements, simply because they were never properly assessed and coding commenced right away. Not unlike a surgeon starting an operation without checking the patient's blood pressure and possible allergies to materials used during the surgery.

To prevent, or at least reduce, the number of failed projects in IT, a clear and correct architecture is paramount. One that consists of technical and requirements documents that provide a complete description of the system-to-be. This can only be achieved by a professional in this field, the digital architect, who is most often trained in several academic fields concerning computer science, organizational theory, business administration and psychology. A digital architect uses knowledge from all these fields because on the one hand he needs to have a knowledge of the "building materials" he has at his disposal (i.e. software, databases, programming languages, hardware) and on the other hand he needs to understand where the project will be realized (organization theory) and who will be interacting with it (human computer interaction).

The architecture to be created must identify all components of the problem space, show their relationships and define the terminology, rules and constraints in these relationships (Britton & Bye, 2004). In order to create this architecture, information from and about the client is needed. Getting this information is a three step process:

Divergence → convergence → specification

In divergence, the client has given a preliminary explanation on what he thinks he wants. It is clear to everyone that this image is incomplete, and so further questioning is needed. The architect will have to ask a lot of questions to get the details of every idea the client has, something not unlike questioning a patient who presents only the CC, there is always a lot more to it than just that. This step, the architect tries to create an outline of what the client wants and

this will become the basis for further talks, he tries to elicit requirements from the client. Through models, analysis and communication a first overview of the requirements will be presented to the client, bringing the process to the second phase.

Once the client and architect agree upon the basic structure of the system, the architect starts the convergence phase. Clients typically name requirements such as "infinite uptime", nobody likes a system to break down or needed to be taken offline for maintenance. However, this is not realistic and the architect will have to negotiate with the client as to what is acceptable, in this particular case the argument can be that infinite uptime costs almost infinite money because there will have to be a lot of backup systems, backup backup systems, ad infinitum. Explaining these difficulties and limitations with computer systems can be a very hard nut to crack. Try explaining hash functions to a lay person and you will discover the difficulties that architects and other professionals have to face.* Upon negotiation it might turn out that the client will accept a 98 percent uptime, and speeds that are actually possible according to physics. Clients have a lot of requirements, that can be identified to be of four types according to the MoSCoW method: Must, Should, Could and Won't (Brennan, 2009).

- **Must** requirements are those requirements of which it is absolutely essential that the final system conforms to them. Without these, the project is a failure.
- **Should** requirements are those requirements that have a high priority. These are however not absolutely essential in this form as they might be satisfied in other ways.
- **Could** requirements are those requirements that are icing on the cake. Resources do not have to be initially allocated to accomplish these, only if time and resources allow would it be nice to have these as well.
- **Won't** requirements are those requirements that will remain in the pipeline, to be kept in mind while designing the current system without making plans to actually implement them in this release. It is uncertain if a future release will incorporate them or if these requirements will be dropped eventually.

A complete understanding of the situation in which an architecture is to be realised takes into consideration the principles behind the situation as inherent laws as well as in the form of imposed laws and the guidelines governing the situation. Inherent laws are observable and validateable and are not subject to design or alteration, such as the natural laws. Imposed laws can also be validated but are created by man and so can be altered to suit the stakeholders. Finally, the guidelines attempt to coerce users into conforming to these imposed laws. These laws and guidelines are reflected in the MoSCoW method, the inherent

*One of my teachers actually managed to explain the basics of hashing, using small Lego cars. Put together it is the unhashed file, taken apart it is hashed. It is very hard to recreate the original car, but it is easy to confirm that a certain pile of bricks is the hash of a specific car. Not all computer and cryptography related difficulties and limitations lend themselves to Lego analogies however, and the search continues for simple ways of explaining such things to non-technical people.

laws are however already a part of the system as it is impossible not to conform to them unless that law proves to be false.

The third phase is that of specification, in which the final requirements are stated, modelled and presented to the builders who will take it from there. It is not uncommon for requirements to evolve during this phase and so the IT architect stays in close contact with both the builders and the client to communicate any issues as they present themselves.

The three steps are however not completely one-way. During the process it is possible that new requirements are identified and need to be explored further, going back to a new divergence phase. This is unlike the well-known waterfall method of software engineering, which was first mentioned in academic literature as an example of a flawed method of software engineering(Royce, 1970). It is better to see each phase as containing numerous smaller phases with the same cycle.

7 Teaching of and training in the IT anamnesis

There are a few different fields to train in to become an architect. There are digital, enterprise, and IT architects, and all do different things although there are similarities between them. What they have in common is a body of knowledge that mainly exists of unrelated best practices in methods and frameworks, and even best practice can be taken with a grain of salt (Op't Land et al., 2009).

7.1 Courses at University

During the bachelor's phase of Information Science at the Radboud University, the only training received in the IT anamnesis is during two courses: Requirements Engineering, and Software Engineering. In the first course, the fundamentals of the requirements process are explained and a first attempt is made in drafting the requirements for an actual project. This is not without its problems as the resulting requirements are not acted upon by the same group after establishing them so there is no confrontation with mistakes. Also, the requirements interviews are conducted in a very informal manner with an entire group of students present during these talks, something that does not represent the real situation during a professional career. This course does have its merits though, as for the first time during the bachelor phase, the students get a feel for what they will be doing in their career and many students get very enthusiastic because of this. The limited time allocated to this course does however provide significant problems as a selection is to be made of what to teach and what not to teach. Interviewing skills are something that is not taught at this moment, nor are the exact reasons behind the creation of the models during this course. As a result, students can go two ways: Either they see the intentions behind the course and take to heart what is being conveyed and try to grasp the theory, or they see the documents to be created as the goal and the course becomes an exercise in pleasing the one who will grade these documents. This is however, inherent to all courses at university.

In the second course, there is a mix of students from various background working on the realization of a piece of software for an actual external client. The role of the information science students in this course is leading the requirements engineering effort. The main source of information is the client, and so interviews take place in order to uncover the requirements for this project. This is the first time student will have to interview a real client, the first IT anamnesis. Those students who saw the Requirements Engineering course as just another course to pass then struggle with this situation as there is no corrector who looks at the documents. The information science student is now expected to be the requirements engineering expert within this group and he can make or break the entire project single-handedly by either performing a good interview or a bad one.

This is not unlike the future career of these students, as they will be seen as the experts in systems design. The amount of responsibility resting on their shoulders is immense. IT projects often cost millions, and although a requirements engineer cannot guarantee the success of the project, he has significant influence in what will be created and so can steer the project in the completely wrong direction (Kulak & Guiney, 2004). Getting it wrong can cost millions, getting it right might save million. This is the impact that requirements engi-

neering can have and at its basis lies a good anamnesis, a skill that needs to be learned, a skill that needs to be taught.

7.2 Filling out forms

In the previous section it was already stated that there are two types of students in the Requirements Engineering classes: Those who see this course as something to pass, and those who see it as an opportunity to learn about requirements engineering. This is of course a completely black and white image and there are many shades of gray in between but nevertheless, this is an issue with courses.

Because students have no prior experience, the teacher has no choice but to offer ample guidelines and deadlines on the various documents and the preferred form. For better or for worse this allows students to sit back and relax a bit. To make matters slightly worse, because of the large amount of documentation to be created and the limited time allocated to this course, the students work in groups. Where there are groups, there will always be students who go along for the ride and contribute little more than their name on the final document despite the best efforts of teachers to prevent this from happening. Vice versa there are students within each group that take things personally and want to get the best possible grade, thus inadvertently helping the others obtaining a better grade as well. Students are provided document templates and a book such as Kulak & Guiney (2004) to guide them and learn requirements engineering according to these two tools. It might be impressed upon the students that this book is the Best Way, just as the tools from other courses are thought of as the tools of the trade. The problem with this is that if all you have is hammer, than every problem you encounter is a nail. Students hammer away at the problem until it fits nicely within the frameworks they have learned, and there is much rejoicing.

The goal of the students seems to be making the information given fit with the model to be used rather than using the model as a tool to discern relevant information. It might be that students perceive the limitations of models as useful to create boundaries as to what to put in a document and what to leave out, thus not having to do unnecessary work for their grades. This difference in focus is however not a problem of students alone. In modelling these two foci can also be found. On the one hand there is the focus on information that is seen as relevant to create the model itself (pragmatic focus) and on the other hand there is the focus on information that is needed to make the information fit within the model (semantic-syntactic focus) (Hoppenbrouwers & Wilmont, 2010). Both foci have their purpose, as long as they are both pursued, which is not always the case. It is my personal experience that students lean towards a semantic-syntactic focus because a model that is semantically and syntactically complete, lets you pass the course, whereas the pragmatic focus serves as the reason the model was created, which was to pass the course as well, only with a higher grade. Although the created documentation might later actually be used in a different course, this was of no concern to the modellers as they themselves would not be involved anymore. And as the only reason the model was created during the Requirements Engineering course was to pass the course, and not to create the actual system, this was perceived as less important.

8 Theories behind the IT anamnesis

As digital architects come from a technical background and so they display a common characteristic all engineers subscribe to: There is a "the Best Way" to be found out there and they will not rest until they have found it. This has resulted in countless attempts to create The method, model, language, and other tools to reach perfection in requirements engineering. This seems in stark contrast with the medical community where progress seems slower in the continued development of the anamnesis and documentation from decades past is still relevant (Stoeckle & Billings, 1987; Aloia & Jonas, 1976; Maguire & Rutter, 1976).

8.1 Paralysis by abundance

ArchiMate 2 Certification for People, OCUP advanced, TOGAF 9, IAF essentials, ORM consultant, Zachman Certified Enterprise Architect, just a few of the hundreds of possible certifications out there that a digital architect can pursue. The IT community as a whole is filled with certifications, workshops and courses that employers require employees to get, follow and pass. A BSc or MSc is just the very first step in this life-long obsession with certification. No matter the language, framework or model, there is a certificate for it that shows the world that the holder has received training in using it. Which certificates are actually worth something and which ones are just to give off the impression of professionalism? This is a question I did not find an answer to, as every sector in the IT business has their own set of certificates and there is no clear overview to be found.

Is there a best path through this? What certificates to pursue is in part up to the employee, and in part up to the employer who can demand that certain certificates be attained before being allowed contact with high-profile clients. These required certificates make sure that each architect in a company has the same basic toolkit, which will guarantee that each architect will use the same hammer, which will result in a limited view on problems. The benefit of this is that if an architect leaves for whatever reason, another can take his place and still understand the documentation.

8.2 Organizations are like computers, there is a Best Way

Digital architects have a technical background, most often in the field of computer science or information science. Their training thus involves working with computers, and learning how to program them to accomplish certain tasks in the most efficient way, preferably with the least lines of code. This way of thinking becomes a part of everyday life and digital architects will see ways to improve upon existing systems everywhere they look, in search for the Best Way. The Holy Grail of the Best Way is all around us, it implies that such a way exists for everything, and should be sought, religiously.

Organizations are systems, and as such can be improved upon. Digital architects have this gut feeling that there is an optimal way for these organizations to function and they can find it hard to keep from commenting on everything that they perceive as being suboptimal, it is not the Best Way. This way of thinking can be found in the open source community where thousands of programmers

work on the same project, in search of optimalization, and finding the Best Way is greeted with great reverence and honour. Organizations and people in management are not concerned with the Best Way, at least not in the way the digital architects are. Their Best Way is the way in which the profit of the organization is maximised, even if the system itself functions suboptimally. It is the result that counts, and how it is achieved is of lesser importance. Digital architects however, do not only want to achieve results, but want to do so elegantly. There is a beauty in perfect systems, "it just works" is not good enough.

Digital architects have to keep this balance between their inner urges to make it perfect and what their clients want. This might also explain the vast number of tools out there, perfection is strived for, and so long as it is not found, the quest continues, even though the customer could not care less and wants a system that just works.

9 Comparison

The goal of this thesis is to find similarities and differences between the medical and IT anamnesis and see what IT can learn from the vast body of knowledge that the medical community has built up in this and the previous century. Now that both anamnesis have been described, there are clear similarities and differences to be found.

9.1 Similarities

Both fields continuously attempt to find the best method of taking the anamnesis, and continue to improve their methods, whilst being backed up by academia. Both fields also have an ultimate goal in mind, the realization of a diagnosis and a treatment. Although when looking closer, IT sometimes seems to forget this goal altogether and go for a completely different goal that is the perfect application of a certain model or language to the problem at hand, without subsequently offering a treatment. The tool becomes a goal in itself rather than a means to an end, and care should be taken to prevent this from happening.

In both fields, talking with the client is very important as they are the source of information. These talks can become very confusing as there are countless possibilities to use jargon and models that the client does not understand because of a lack of training. Observation is also a part of the methods, but this will only reveal so much. It is nevertheless an important aspect of the methods, as it will indicate if the client is comfortable with the settings and the interview. Nonverbal communication is an integral part of this.

Overall, both methods use the same basics of divergence, convergence and specification which seems to be the case in all exploring types of research. First get a feel for the lay of the land, then find what is interesting and lastly, investigate it.

9.2 Differences

The biggest difference between the medical and IT anamnesis is that in the IT anamnesis, stakeholders can be far more numerous and distributed. These stakeholders can have conflicting interests which may not always be easy to articulate and so are hard to satisfy (Nuseibeh & Easterbrook, 2000). In the medical anamnesis, in general there is just one patient, and his quality of life is central to the whole process. There are no stakeholders with vastly different interests and views. Because of these big differences in clients and stakeholders, there are dozens, if not hundreds, of models, flowcharts and methods described that will help the architect in solving the puzzle of requirements whilst a doctor has a clear-cut method of solving the medical puzzle. Herein lies one of the biggest advances the architect can make, which I will discuss in the results. Also, because of the rapid evolution of IT over the last decades and its relative infancy when compared to the medical community it seems that the speed of development of new methods (i.e. models, languages and processes) to facilitate the anamnesis is much greater than that of the medical community. All the tools used are continuously evolving, where in the medical anamnesis, the main method is still a one-on-one interview with the patient, without aid of IT systems other than perhaps a piece of software used that replaces the piece of

paper the doctor uses to take notes. Although in the future the anamnesis will become paperless, it is hard to imagine the anamnesis not being this one-on-one interview, that will still focus on the CC, HPI, ROS and PMH. This however is a strength of the medical anamnesis as medical records span a lifetime, needing a form of consistency en methodology.

9.3 Projecting the four elements

The four elements of the medical anamnesis are there for good reasons, amongst them is providing a framework which can be used to conduct interviews. Can we project these elements onto the IT anamnesis, or are there perhaps already similar frameworks in IT?

Although it is true that if you want to know what someone wants, you can just go and ask them, this is very hard to do when it comes to computer systems (Kulak & Guiney, 2004). Systems are often too complex to get a good answer, but any answer is better than none at all. There is always a reason for the client to contact a software firm. This can be seen as their CC, the reason they feel they cannot continue with the present situation. So instead of asking what they want, a good question would be to ask why they feel they need an IT solution. Something must have changed because previously, the client experienced a fit with the environment, which has now turned into a misfit which asks for a realignment within the organization to get a proper fit again. Getting this small but vital piece of information in the open will give insight in what moves the client. If the client has no clear answer to this "why" question, would it still be wise to provide a solution? Perhaps the impression of a need is nothing more than a want and the client can do without a new system. Within this element, asking the client what kind of solution they want can cause the client to go into great detail about the problem, which is not what we want to begin with. Before going deep into the problem, more background knowledge is needed to see if the solution the client has in mind is indeed the solution needed or that perhaps another road needs to be taken.

The next element, the HPI, is the story behind the fit and the ensuing misfit. Where in the CC, only the need that resulted from the misfit is looked at, now the path leading up to this moment is also examined.

- Is there a specific department within the organization that is suffering most from the problems or is the organization as a whole suffering.
- What is the problem like. How does it manifest itself.
- How big of a problem is this within the organization. How long can it continue in its present form. Is the survival of the organization under threat.
- When were the first problems identified. Did the organization react to it back then.
- What is the setting of the problem. Who is closest to it and can explain things on this level.
- Are other elements within the organization feeling the effects of this problem.

This element will put the CC in proper context, so now we know what the problem is about, we get the client's point of view which really is the first step towards requirements engineering. Before we even start designing a system, we want to know what is going on and how the client sees his problem.

The ROS is next on the list, and might well be the biggest contribution the medical community can make towards the IT anamnesis. I do not have a clear image in what specific questions would need to be asked in this phase, but I believe that such a list can prevent a lot of wrong solutions being offered. The ROS is akin to domain analysis. Domain analysis does not look at the specific system, but at the domain in which this system will reside. The intent is to identify common problem solving elements that are applicable to all applications with the domain (Pressman, 2010). Domain analysis will likely reveal common patterns within the problem that were not identified as such by the client, so also do not turn up in the client's point of view. Perhaps because these patterns are a part of the culture of the organization, have slipped into the realm of "that's how we have always done things around here" and are now simply taken for granted (Schein, 1985). It might also be that the CC, HPI and ROS combined turn out to be a pattern that the requirements engineer has encountered before in which case previous solutions could provide part of the solution to the current problem.

Finally, the PMH needs to be taken into account. It will be quite rare to come across a client who has no IT systems in use. Perhaps the client is currently using an off-the-shelf system and wants something that is specifically designed to meet his needs. The choice of system which was used previously might influence the client's wishes. Are there other systems already in use within the organization and does the new solution have to communicate with these systems? Now we are moving into the territory of requirements engineering with regards to the specific things we need to keep in mind when designing the new system. Although organizations have the choice to remove all the current systems and start with a completely new system, this is hardly ever done because of the investments already made. The older an organization becomes, the harder it becomes for them to adjust to changes in the environment and the less likely it is they will make this choice to radically change their current systems. It has all worked in the past, so why stop using it now, perhaps with a few minor changes it can still be used for another decade. This is the so called liability of ageing (Baum & Shipilov, 2006) and it can result in an organization dying. This can be a hard bargain for the requirements engineer, on the one hand there is an ageing system which might be coded in a language few programmers are proficient in these days, and on the other there is the client who does not want to make radical changes. To create a bridge with the medical world, sometimes you have bad news, and only a high risk operation can save the patient's life. It is up to the patient to take this risk, or continue with the stopgap measures and see what the future will bring. Some organizations do not want to hear what has to be done and will gladly play handsomely to continue on the present course. The choice is up to the requirements engineer and his sense of ethics as to what solutions he offers such a client.

But things are not always this grim, old systems might be painlessly replaced and old infrastructures integrated easily within the new system. Different computer languages can work together, or the code might be in a language that is still in extensive use and understood.

Once this last phase is completed, the requirements engineer should have a complete image of what is going on. Not only the perspective of the client, but also an overview of the domain in which the system is to be realized and what systems are already in place that need to be kept in mind when designing the new system. This being done, we can now really start the requirements engineering process, building upon a solid and understood foundation.

10 Results

The IT anamnesis is still very much in its infancy and trying to find "the Best Way" whereas the medical anamnesis has had roughly the same shape throughout the last decades. The four basic elements (CC, HPI, ROS and PHM) of the medical anamnesis are its definite strengths. The medical anamnesis can keep this form because there is very little variation in stakeholders and situations in which the anamnesis takes place. The IT anamnesis has a much harder time finding such a universal template, but would clearly benefit from it as this would standardize the process of requirements engineering in such a way as to allow the fine-tuning of each of these elements. As it stands, requirements engineering has no set method and so each time a requirements engineering process is started, the method is made up on the fly. To accomplish this, tools and languages are used that have proven to work in previous processes or in which the consultant has received training. As the consultant becomes more familiar with a certain tool, he becomes more proficient with it and thus gets better results, leading to a cycle wherein the consultant uses the same tool each time and starts to believe that it is indeed the Holy Grail, and not a false idol that he is worshipping while wearing blindfolds to all other tools around him. I cannot shake the feeling, that the creation of a model or language is more important and prestigious than actually helping the client. This has resulted in the development of hundreds of models, methods and languages that are being used by different firms who all claim to have the Best Way which is simply not true.

Although there are individual differences in the execution of the medical anamnesis, we can find the four basic elements in there in some shape or form, which is not the case with requirements engineering as it stands today unless we go up enough levels of abstraction that it is no longer of any use as a guideline and reach the divergence, convergence and specification path. IT can benefit from identifying basic elements in the anamnesis and developing those.

Another lesson to be learned is within the ROS phase of the medical anamnesis. Within this phase it is customary to ask questions about bodily systems, even if they do not seem linked to the CC at first. This list allows for the identification of signals from the body that could have been missed if not asked about directly yet turn out to be of vital importance. At this moment, IT does not have any such list which can be used to get a complete picture of the system in which an IT system is to be realised. Although the Zachman Framework can be used for this to some extent, it is far to complex to be used as a checklist by itself. Elements of it however can serve this purpose, as long as we keep in mind that filling out the entire framework will take considerable time and resources. Because the Zachman framework is not designed for this purpose and is far to elaborate, its essentials might be extracted and used in the development of a dedicated checklist. The same goes for other current tools. Rather then using existing tools for purposes for which they were not conceived it would be more useful to develop a new tool which is intended to be used as this checklist, and thus would be far more efficient.

Lastly, a major shortcoming can be identified in the training of the IT anamnesis, or rather, the lack of it. In the medical anamnesis, extensive training takes places over the course of several years and residencies during the six-year path of becoming a basic doctor. Further refinement is done during specialization to

becoming a medical specialist, which can take another four years. The journey to becoming an MSc Information Scientist takes just four years and assumes that the student is then ready to start working as a digital architect. With little to no experience in one-on-one contact with clients however, problems could arise. Talking with stakeholders is not something we learn in everyday life and is actually an essential skill to have when being involved with requirements engineering. We are not even talking about all the subtleties that are involved in talking with people and holding an interview, even basic training in talking with people on their own level and finding the right answers is not taught at this moment. It is my belief that any digital architect can benefit greatly from extensive interview training with real customers to develop patterns of interaction that will allow the architect to focus on the subtleties of interaction without having to keep thinking about the basics and being anxious.

11 Discussion

As with any literature review a selection had to be made on the literature to be used. The choice was made to first use the resources at hand, the Radboud University library and my own bookshelf, and after that any scientific papers to be found, provided that the library had a subscription to the journal so that a copy of the paper was available. Because of this, the literature used was bounded by several factors and certain papers were not available that I would have liked to read.

In the literature referenced there is mention of an overwhelming choice in models and languages used in digital architecture. The Zachman Framework and MoSCoW are just two of these that turn up frequently. Other languages and frameworks such as the Unified Modelling Language, Object Role Modelling, ArchiMate, and Business Process Modelling Notation could have also been discussed, but this would not result in a different outcome. Each tool claims to be the Best Way within certain domains of application, yet no scientific proof was found to substantiate these claims.

Both the medical and IT fields have been explored from personal experience. The IT field in particular leans on my personal experience with requirements engineering and training as a digital architect. As such, a student from a different university, following a different curriculum (at the University of Utrecht for example) but also becoming an information scientist will probably have a different view on requirements engineering. Here the analogy of the hammer fits perfectly again, although I cannot imagine the differences being so great that such a student would not recognize my results.

A student from another university might be holding a different hammer but he will agree with me that there is no common guideline, nothing that can be called the IT anamnesis and contains the same common elements. This is supported by literature, although only through absence. Throughout medical literature there are references to the four basic elements of the anamnesis, such references are not found in IT literature however.

12 Conclusion

The medical anamnesis has a method to it that every medical professional uses. This allows for the standardization of this process, and produces consistent results. The method has been under development for decades and is now very mature and effective.

The IT anamnesis is still in its infancy, and professionals are still searching for the best way of conducting this anamnesis. This has resulted in a myriad of methods, languages, models and other tools that are all seeking to become the standard method within this field. As of yet however, there still is no single tool that covers every occasion.

Although the medical anamnesis does not cover every situation, its basics do. The four distinct phases in questioning that can be found in every single anamnesis all over the world and the training of this method through the bachelor and master phases of education. Also, the ROS list that each medical specialization uses, provide a great catchall for those details that somehow were missed in the other phases and turn out to be relevant nonetheless.

These are the lessons IT can learn from the medical community:

- Find a method, a fixed set of elements in the requirements process, that can be used as a standard in every requirements situation.
- Within this method, make sure there is a basic list of questions that need to be asked in every requirements process, to catch common elements that might otherwise have been neglected.
- Add a significant amount of training in interview skills and the application of this method to the BSc and MSc sections of the education in becoming a digital architect.

I realize that the development of such a method would be very difficult as there is a huge diversity in stakeholders. However, IT would benefit from even the slightest common guidelines in the requirements process as every single source I consulted on requirements engineering described a different method altogether. The only similarities between these methods, which are also shared by the medical method, are those of divergence, convergence and specification and even this is not practised deliberately by IT students.

The method does not have to use a single tool, but needs to provide the same result whichever tool is used and it is up to the individual to select his tool of choice, with the skill and knowledge to switch to a different tool as needed rather than sticking to a single tool because his firm always uses it.

The basic questions are something that warrants further research as well, as there must be similarities between every IT project. Such a list can prove to be very valuable if it indeed turns out that the failure of IT projects turn out to have similar causes. It would be very interesting to research the common causes of IT project failures and see what can be done to remedy them, the financial benefits alone warrant such research.

Lastly, the basic interviewing skills are something to have a closer look at. It seems that of all the majors at University, only the medical, and to a lesser extent the psychology major, actually have extensive training in dealing with people and interviewing them. Although there are courses about Intervention

Methodology and Communication Management, during all these lectures, the only people trained on are other students. There are no internships, no exams in interviewing, no basic training whatsoever. Before we decide what exactly to train, any digital architect would benefit from even the most basic training in people skills and interviewing. This does not even have to encompass a method yet, just getting comfortable in talking with strangers and asking them all kinds of questions in search for answers. Interviewing, talking, public speaking and even facilitating a group session on discussing a topic should be second nature to the digital architect, yet they do not receive any specific training while still attending University, leaving this to the organization the student will start his career with.

Before we can even consider finding "the Best Way", let us first focus on the basics, there is still a lot of catching up to do with those masters of the anamnesis, the doctors.

References

Aloia, J., & Jonas, E. (1976). Skills in history-taking and physical examination. *Academic Medicine*, *51*(5), 410–415.

Aneshensel, C., Estrada, A., Hansell, M., & Clark, V. (1987). Social psychological aspects of reporting behavior: lifetime depressive episode reports. *Journal of Health and Social Behavior*, *28*(3), 232–246.

Barsky, A. (2002). Forgetting, fabricating, and telescoping: the instability of the medical history. *Archives of internal medicine*, *162*(9), 981–984.

Baum, J., & Shipilov, A. (2006). Ecological approaches to organizations. *The Sage handbook of organization studies*, (pp. 55–110).

Bickley, L., Hoekelman, R., & Bates, B. (1999). *Bates' guide to physical examination and history taking*. Philadelphia: Lippincott.

Bolden, G. (2000). Toward understanding practices of medical interpreting: interpreters' involvement in history taking. *Discourse Studies*, *2*(4), 387–419.

Brennan, K. (2009). *A guide to the business analysis body of knowledge (Babok Guide)*. Toronto: International institute for business analysis.

Britton, C., & Bye, P. (2004). *IT architectures and middleware: strategies for building large, integrated systems*. Boston: Pearson Addison Wesley.

Deutsch, F., & Murphy, W. (1961). *The clinical interview: A method of teaching associative exploration*. Boston: International Universities Press.

Formijne, P., & Mandema, E. (1979). *Leerboek der anamnese en der fysische diagnostiek*. Utrecht: Bohn, Scheltema & Holkema.

Geld, A. v. d. (1980). *De persoonsgeschiedenis: inleiding tot de psychologische en pedagogische anamnese*. Lisse: Swets & Zeitlinger.

Haidet, P., & Paterniti, D. (2003). "building" a history rather than "taking" one: a perspective on information sharing during the medical interview. *Archives of internal medicine*, *163*(10), 1134–1140.

Hoppenbrouwers, S., & Wilmont, I. (2010). Focused conceptualisation: framing questioning and answering in model-oriented dialogue games. *The Practice of Enterprise Modeling*, *68*, 190–204.

Kulak, D., & Guiney, E. (2004). *Use cases: requirements in context*. Boston: Addison-Wesley.

Lacasse, M., & Maker, D. (2008). Fishing and history taking. *Canadian Family Physician*, *54*(6), 891–892.

Lankhorst, M. (2009). *Enterprise architecture at work: modelling, communication and analysis*. New York: Springer-Verlag.

Lankhorst, M. (2012). *Agile service development, combining adaptive methods and flexible solutions*. New York: Springer-Verlag.

Maguire, G., & Rutter, D. (1976). History-taking for medical students: 1 - deficiencies in performance. *The Lancet*, *308*(7985), 556–558.

Nuseibeh, B., & Easterbrook, S. (2000). Requirements engineering: a roadmap. In *Proceedings of the Conference on the Future of Software Engineering*, (pp. 35–46). ACM.

Op't Land, M., Proper, E., Waage, M., Cloo, J., & Steghuis, C. (2009). *Enterprise architecture: creating value by informed governance*. New York: Springer-Verlag.

Pressman, R. (2010). *Software engineering: A practitioner's approach, seventh edition*. New York: McGraw-Hill.

Royce, W. (1970). Managing the development of large software systems. In *proceedings of IEEE WESCON*, vol. 26. Los Angeles: IEEE Computer Society.

Schein, E. (1985). *Organizational culture and leadership, A dynamic view*. San Francisco: Jossey-Bass.

Schouten, J. (1985). *Anamnese en advies*. Houten: Stafleu.

Simon, H. A. (1965). *Administrative behavior. A study of decision-making processes in administrative organization*. London: Macmillan.

Stewart, M. (1995). Effective physician-patient communication and health outcomes: a review. *CMAJ: Canadian Medical Association Journal*, *152*(9), 1423–1433.

Stoeckle, J., & Billings, J. (1987). A history of history-taking. *Journal of general internal medicine*, *2*(2), 119–127.

Zachman, J. (1987). A framework for information systems architecture. *IBM systems journal*, *26*(3), 276–292.

Zave, P. (1997). Classification of research efforts in requirements engineering. *ACM Computing Surveys (CSUR)*, *29*(4), 315–321.