jet INTL JOURNAL OF ELECTRONICS AND TELECOMMUNICATIONS, 2015, VOL. 61, NO. 2, PP. 151–157
Manuscript received January 30, 2015; revised June, 2015. DOI: 10.1515/eletel-2015-0019

Automated On-Demand Generation Of Patient Summary Documents

Oliver Krauss, Barbara Franz, and Andreas Schuler

***Abstract*—Patient summary documents provide crucial information about a patient, like allergies and adverse reactions, which are necessary for an efficient and safe treatment and offer a quick overview of the patients health status. Automatically generating patient summaries from Electronic Health Records (EHR) reduces the workload of medical personnel. Nevertheless, existing approaches do not take several challenges that occur in live operation into account . Based on a health standard-compliant approach, a system for on-demand generation of patient summaries was implemented and evaluated using real data. This work shows several problems which could be identified. Those problems are not covered sufficiently by current research. Possible approaches to a solution are suggested, which have to be further investigated in future work.**

***Keywords*—patient summary, IHE, e-Health, ELGA, epSOS**

I. Introduction

CURRENT trends towards a highly interconnected medical community present new opportunities and challenges. The amount of health records medical personnel, such as physicians and nurses, has to handle, increases, while the time available for the treatment of a patient decreases. Nevertheless, the availability of crucial information like allergies and adverse reactions of a patient is necessary to provide an effective and safe treatment. In particular in the case of an emergency or transfer of a patient to another medical institution or physician, medical staff need to get a quick but thorough overview over a patients health status. Thus, a summarization of a patients health records, which presents vital information at first glance, could reduce workload and improve the quality of care. [1] [2]

The information that should be provided by a patient summary has been identified in a European guideline for a patient summary document [2]. It has also been part of the epSOS project, which aims to provide seamless cross-border healthcare to European citizens [3], and of the ELGA project, which, at the time of writing, is in the process of defining a patient summary [4]. Required information includes for example blood type, previous surgeries, medical prescription, allergies and adverse reactions. Organizational information, like the patients contact info and family physician is necessary as well. Figure 1 shows a comparison between the content entailed in the patient summary definitions of epSOS and ELGA.

Current work like [5] and [6] use a manual or semi-automatic approach to generate a patient summary document from EHRs. Several challenges can be identified throughout the lifecycle of a patient summary, when providing patient summary documents on-demand and automatically, e.g.: How to decide, which parts of an EHR should be included in the patient summary and how to cope with information that was not prepared for mechanical processing. Additionally, it has to be considered how to trace the data sources in a patient summary, for example when other medical institutions might need further information from the author. Since repeatedly processing large amounts of data raises concerns about system performance, persistent patient summaries might be necessary. Nevertheless, document persistence requires consideration of changes in the information source. Security aspects and laws concerning medical data and patient privacy rights, which might vary between countries, also have to be taken into account.

This paper shows a fully automated approach to generate and maintain a patient summary document as well as making the source-data and who accessed the generated documents traceable.

II. Methods

The presented approach is based on an Integrating the Healthcare Enterprise (IHE) [7] compliant system, as suggested in [8], and relies on the Health Level 7 Version 3 Clinical Document Architecture Release 2 (HL7 V3 CDA R2,

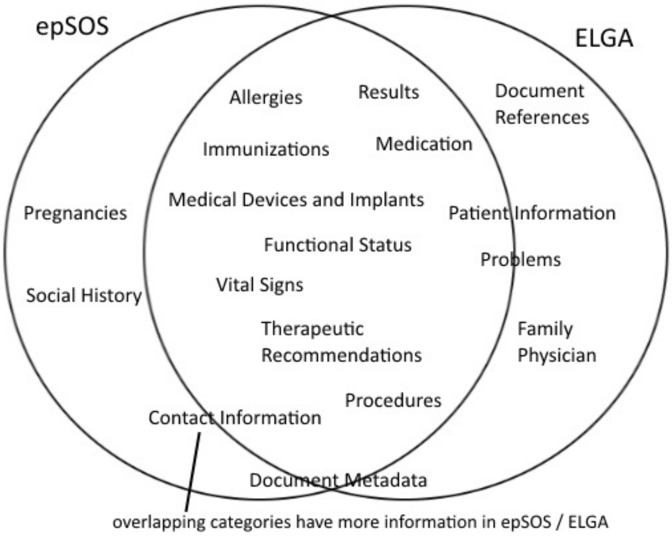


Fig. 1. Data contained in epSOS and ELGA patient Summary [3] [4]

The work presented was conducted in the project CDAMobile. A joint project between Systema Human Information Systems GmbH and the Upper Austria University of Applied Sciences.

O. Krauss and A. Schuler are with the Research Department, Upper Austria University of Applied Sciences, Austria, 4232, Hagenberg (e-mail: andreas.schuler@fh-hagenberg.at; oliver.krauss@fh-hagenberg.at).

B. Franz is with the Department of Software Engineering, Upper Austria University of Applied Sciences, Austria, 4232, Hagenberg (e-mail: barbara.franz@fh-hagenberg.at).

short: CDA) [9] standard as data format. Both the technology and the format of the source data were selected because they are widely in use in healthcare facilities, and allow for interoperability between healthcare organizations [1] [10]. IHE provides syntactic interoperability in form of interfaces, while CDA provides semantic interoperability in form of data structuring and identification.

Instead of implementing a new IHE based solution, existing systems which are IHE compliant, like Tiani Spirit or Open Health Tools (OHT), were considered for use. However they are not applicable for an automatic on-demand approach, since OHT does not implement the on-demand document definition of IHE, and Tiani Spirit only considers machine-readable information in their approach.

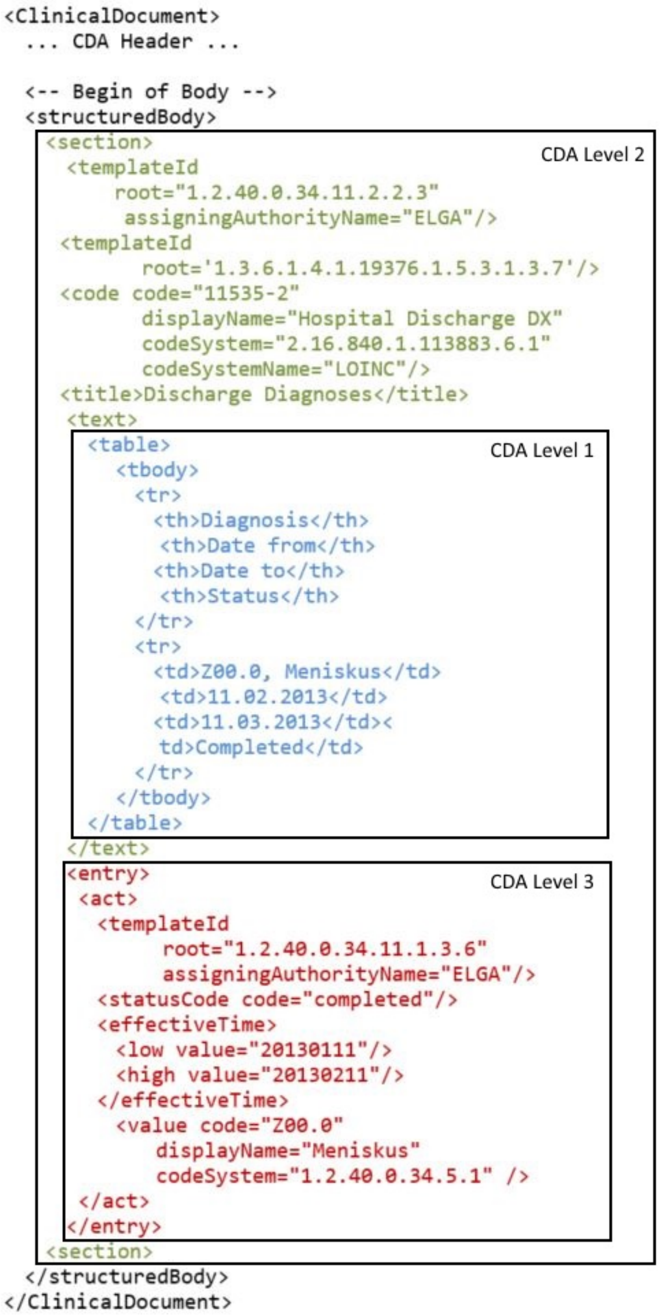
Another solution, EHR Archetypes, which is a model-based approach to maintaining EHR documents, can be used for searching and processing clinical documents [11] [12], and is compatible for use in IHE systems [13] [14]. Since the available source documents are not based on archetypes, this approach was also not suitable.

A. *Document Information and Aggregation*

In a medical environment most information about a patient is stored in documents that record medical and administrative information about that patient such as present and past treatments. The CDA standard was defined by HL7; a non-profit organization dedicated to creating standards for exchanging health information between medical facilities, as part of the HL7 V3 standard [15]. The data model of CDA is based on a Refined Message Information Model (R-MIM), which is derived from the HL7 Reference Information Model (RIM) [15]. A CDA document is an XML document which is split into header and body. The header contains document metadata, for example information about the document type, the patient, the author and the organization that administrates the document [10]. The document type gives information about the contents of the document. From the document class it can be determined which types of medical information may be included, or must be included [9]. In some cases the document class also gives information what level the CDA document implements [9]. The body contains medical information and is classified in three different levels [9], [10]:

1) Level 1 content consists of unstructured information. It includes embedded non-xml content, such as pdf files or images, and uncoded, unstructured sections.
2) Level 2 requires a basic structure in XML, which divides medical information into sections identified by codes, thus defining the class of information available, i.e allergies or vital signs. These codes are based on existing coding systems like LOINC [16] and SNOMED CT [17].
3) Level 3 extends level 2 documents and requires fully structured medical information according to defined CDA templates, which can be identified by codes and the HL7-RIM.

An example of the different levels in CDA can be seen in Figure 2. The figure also shows how the different CDA levels build on each other. CDA Level 1 can stay as standalone. CDA Level 2 wraps around the level 1 data and structures it in sections which have codes to identify them. CDA Level 3 then enhances the level 2 data with additional machine readable entries.



```
<ClinicalDocument>
  ... CDA Header ...

  <-- Begin of Body -->
  <structuredBody>
   <section>
    <templateId
        root="1.2.40.0.34.11.2.2.3"
         assigningAuthorityName="ELGA"/>
   <templateId
           root='1.3.6.1.4.1.19376.1.5.3.1.3.7'/>
   <code code="11535-2"
           displayName="Hospital Discharge DX"
           codeSystem="2.16.840.1.113883.6.1"
           codeSystemName="LOINC"/>
   <title>Discharge Diagnoses</title>
    <text>
     <table>
       <tbody>
         <tr>
           <th>Diagnosis</th>
            <th>Date from</th>
           <th>Date to</th>
            <th>Status</th>
         </tr>
         <tr>
           <td>Z00.0, Meniskus</td>
            <td>11.02.2013</td>
           <td>11.03.2013</td><
            td>Completed</td>
         </tr>
       </tbody>
     </table>
    </text>
    <entry>
     <act>
       <templateId
             root="1.2.40.0.34.11.1.3.6"
             assigningAuthorityName="ELGA"/>
       <statusCode code="completed"/>
       <effectiveTime>
         <low value="20130111"/>
         <high value="20130211"/>
       </effectiveTime>
         <value code="Z00.0"
             displayName="Meniskus"
             codeSystem="1.2.40.0.34.5.1" />
     </act>
    </entry>
   <section>
  </structuredBody>
</ClinicalDocument>
```

Fig. 2. Example of the different CDA Levels

The CDA standard was used for both the source data and the generated document. Each CDA document has a unique identifier. When extracting data from a CDA document this identifier can be used to make the information traceable. This is possible by defining a CDA template which combines the source document-id with the extracted data. CDA templates allow the extension of the CDA standard by defining additional data-items that can be used in the document, or restricting

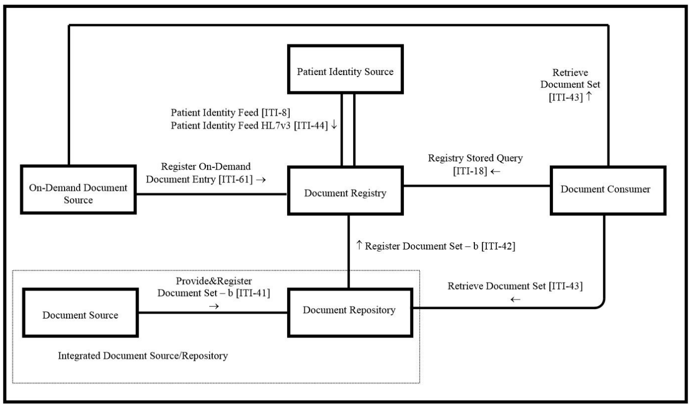

Patient Identity Source
Retrieve Document Set [ITI-43] ↑
Patient Identity Feed [ITI-8]
Patient Identity Feed HL7v3 [ITI-44] ↓
Register On-Demand Document Entry [ITI-61] →
Registry Stored Query [ITI-18] ←
On-Demand Document Source
Document Registry
Document Consumer
↑ Register Document Set – b [ITI-42]
Provide&Register Document Set – b [ITI-41] →
Retrieve Document Set [ITI-43] ←
Document Source
Document Repository
Integrated Document Source/Repository

Fig. 3. IHE Cross-Enterprise Document Sharing - b [7]

the use of data items. Adding a new template to a CDA does not affect existing software that processes the CDA, since the software only processes known templates and ignores unknown ones. [9] [10]

Using the CDA standard as format for the source-data allows the recognition of what information should be present in the generated patient summary. The assigned codes in Level 3 allow to identify the type of information presented, and as such can be used to determine whether an item shall be included in the patient summary. This holds partially true for the CDA Level 2 in which at least a section of data can be identified. This allows to determine if the section might hold information needed for the patient summary. Even though this does not allow an exact recognition, including Level 2 data in the target-document is still preferable than losing valuable information.

B. Data Provision and Transfer

IHE uses existing standards such as HL7 V3 and CDA to enable healthcare interoperability. The IT Infrastructure framework describes the profile Cross-Enterprise Document Sharing (XDS-b). It can be used to store and transmit medical documents. IHE defines a unique identifier to each managed document, which is not the same as the unique identifier for a CDA. [7]

Figure 3 shows the IHE XDS Profile. The profile consists of actors which communicate with each other over transactions. This includes the On-Demand Document (ODD) Source Actor which is used to generate documents at the time of request. The actor communicates with a Document Consumer that requests documents which are then generated on-demand, and a the Document Registry to register new on-demand documents. An option to persist the generated documents exists as well. This lets the ODD Source Actor communicate over the Provide&Register Document Set [ITI-41] with a Document Registry where the document is stored (not shown in Figure 3). Figure 4 shows how the involved actors interact with each other during the retrieval of a patient summary:

- The ODD Source Actor registers a patient summary document for a specific patient in the Document Registry.
- A Document Consumer requests existing documents for a specific patient from the Document Registry, and receives a response confirming that an on-demand entry for a patient summary exists.
- The Document Consumer requests the document from the ODD Source Actor, who starts generating the document at the time of the request. The ODD Source Actor has the option to persist the generated document and register it in the Document Registry.
- When the Document Consumer repeats his request for the patient summary, the ODD Source Actor generates a new patient summary.

If the option to persist the document is used, the On-Demand Document Actor checks if there were any changes in the source documents. Without any recorded changes the Actor returns the already existing document. If there were changes in the source date, the document is updated and returned to the requesting Document Consumer. The updated document is registered in the Document Registry as a new revision of the existing document. [7]

According to IHE the ODD Source Actor does not describe a use-case where an IHE XDS system itself is the data source for the ODD Source Actor [7]. This however is necessary for the presented method to work. An important part of the IHE logging definition is that it can be determined who accessed what information at what time [18]. Since the ODD Source Actor is accessing source documents with the access rights of the Document Consumer requesting the patient summary, the IHE access logs will show access to documents that the Consumer itself never or only partially retrieved, since the data was processed and aggregated before being sent to the Consumer. The access to the patient summary returned by the ODD Source Actor itself is logged as well. The IHE Transaction Retrieve Document Set [ITI-43] requires to log any access to a document, including the information who accessed it and when. Implementing the transaction thus makes access to the patient summary itself traceable [18]. The written access log does not reference the access on the source documents however [18] which does not make it possible to determine what source-documents were provided for the ODD Source Actor in order to generate the document.

Granular access rights were not discussed either in IHE-XDS when persisting the on-demand documents. Considering that not every Document Consumer has access to the same patient data, different consumers will retrieve Patient Summaries with different information in them. This is not a problem as long as the resulting documents are not persisted. If however, they are persisted, this would mean that not one patient summary exists per patient, but one document per patient / requester combination. Furthermore the persisted documents need to have access rights defined on them. Allowing anyone but the original requester to retrieve the document means that he may have indirect access through the patient summary to information he should not be able to see. On the other hand allowing only the original requester access forces the ODD Source Actor to generate a document for each different requestor. This would hinder the performance that can be gained by persisting the document in the first place. One advantage of persisting the generated documents does allow for a complete trace of what information was delivered to the

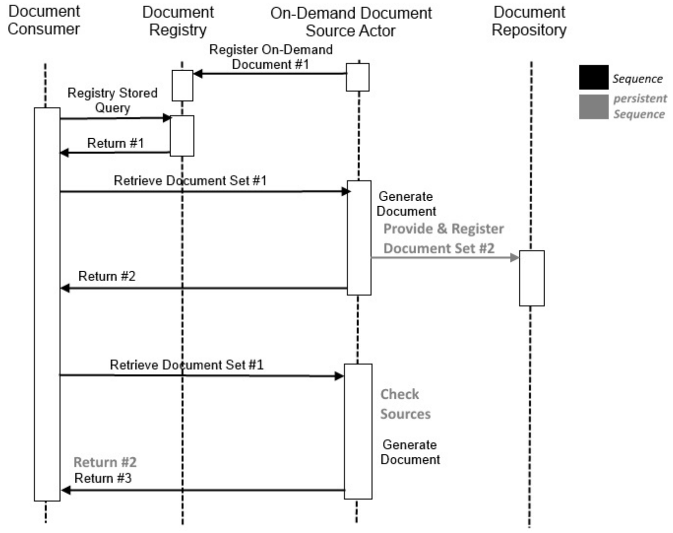


Fig. 4. Simplified Sequence of retrieving an on demand document.

Document Consumer.

III. Results

Figure 6 shows the implemented sequence of generating the patient summary shown in Figure 4 as Generate Document:

- The metadata of all available documents belonging to the patient, the patient summary is generated for, are retrieved by querying the Document Repository. The Document Repository returns four documents with id #5, #7, #9 and #12.
- The retrieved document metadata items are then filtered by their document type. Documents with a document type that do not contain relevant information for a patient summary are removed.
- The remaining documents are retrieved from the Document repository. The ODD Source Actor requests documents #5 and #7 .
- The retrieved documents are filtered for important content by using the codes existing in the CDA levels.

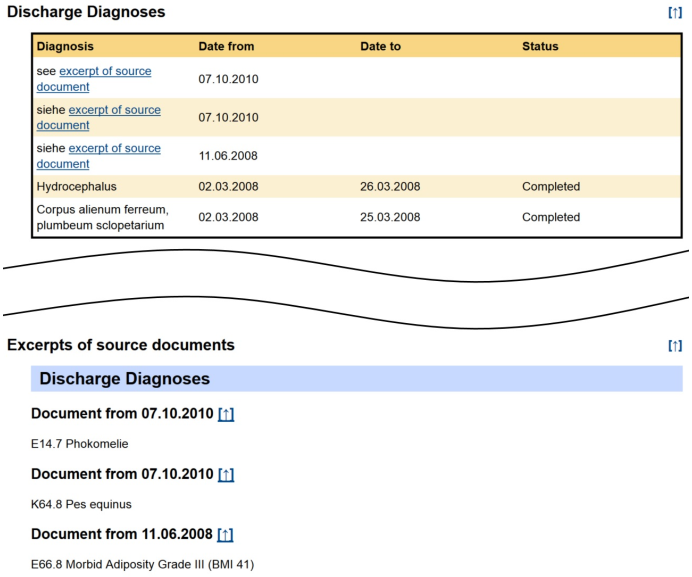
The described method does not use the persistent option of the On Demand Document source actor. The previously described problems when persisting the document concerning handling access on the document were too great compared to the performance gains that using persistence would have brought. Not using the persistent mode comes with the disadvantage of not being able to determine what information was exactly contained in the generated patient summary after the document is sent to the Document Consumer. To lessen this problem an additional log to the one required in Retrieve Document Set [ITI-43] was defined. This log records all documents used to generate a patient summary and identifies all of the source documents as well as the generated document by their IHE document id, repository id and homeCommunity id.

The On-Demand Document Source Actor clearly defines how the documents can be retrieved from it, but a Document Consumer can only check if a document is already handled by the ODD Source Actor, not actually request a new one to be generated for a specific patient. In addition to the transactions defined in XDS-b the solution implements an additional transaction outside of IHE that the Document Consumer can use to trigger Register On-Demand Document Entry [ITI-61] and create a patient summary for a specific patient. This transaction can be initiated by the Document Consumer before the sequence shown in Figure 4, so the ODD Source Actor will register a new document for the patient needed by the Document Consumer. The transaction requires a patient identifier and the document type that should be generated by the ODD Source Actor. If the actor can't generate a document of the requested type or can't identify the patient an error message will be returned to the Document Consumer. If the patient can be found by the ODD Source Actor, and it supports the document type, the actor will check, if it already manages a document for the patient. If it doesn't it will register a new document in the Document Registry and return the new document id of the now managed document, otherwise it will return the id of the already managed document. Since the transaction returns the id of the document this allows the Document Consumer to omit a Registry Stored Query [ITI-18] to the Document Registry since it already knows the id of the document, which then can be requested by the ODD Source Actor directly.

The information for the patient summary was extracted from the CDA source documents by implementing an XML-StAX parser. The parser first analyzes the document header. If the document is too old or it can be determined by the document-class or implemented templates that the document does not contain viable information it will not be processed further. This step was taken primarily for performance reasons.

The parser then processes the document body. CDA Level 1 data does not have codes to identify the data contained, and may not even be structured at all. No satisfactory method to analyze and process the data in a Level 1 document was found, and thus it was decided that source data existing in CDA-Level 1 format will be ignored for the scope of this solution. In Level 2 the entire section is transferred to the patient summary. A section relevant for a patient summary is identified by the code of the section. While this does not allow recognition if the entire section is relevant for medical personell, no important information is omitted. Level 3 is identified by the machine-readable entries. Each data-item with a code that identifies it as relevant for a patient summary is transferred to the generated document.

After parsing the required source data the patient summary is generated by using a template engine. The described method produces a document which contains both CDA-Level 2 and Level 3 data. Mixing those different items in a single section produces two problems. First, automatically processing the generated document is not possible anymore, since the sections may mix data that exists in different levels. This means that if the document is parsed by the existing level 3 entries, the level 2 data which is not represented in those entries will be lost. Doing the opposite and only processing the level 2 invalidates the viability of the level 3 data. Secondly another problem that arises is data cluttering when displaying the document. Since



Discharge Diagnoses [↑]

Diagnosis	Date from	Date to	Status
see excerpt of source document	07.10.2010		
siehe excerpt of source document	07.10.2010		
siehe excerpt of source document	11.06.2008		
Hydrocephalus	02.03.2008	26.03.2008	Completed
Corpus alienum ferreum, plumbeum sclopetarium	02.03.2008	25.03.2008	Completed

Excerpts of source documents [↑]

Discharge Diagnoses

Document from 07.10.2010 [↑]

E14.7 Phokomelie

Document from 07.10.2010 [↑]

K64.8 Pes equinus

Document from 11.06.2008 [↑]

E66.8 Morbid Adiposity Grade III (BMI 41)

Fig. 5. Human readable representation of a section with level 3 data and *level 2 reference* mixed, and the additional section at the end of the document

the CDA standard demands that level 3 data also needs to exist in a human readable format, the data needs to be added where the level 2 data already is. Since the data cannot be mixed there will be duplicate data representations. For example vitaldata may have two tables with bloodpressure measurements (one from level 2 and one from level 3). This problem also occurs when the same sections are extracted from several CDA level 2 source documents.

There is a solution for the two problems described above which was implemented by defining an additional CDA template. This template describes two new CDA level 3 entries on CDA level 2 data. The first entry *level 2 entry* allows to create a CDA level 3 entry which contains the the id of the cda source document, its creation date as an effectiveTime tag, the section code, section identifier, and the text content of the section as CDATA. The template has the effect of making the content of an extracted CDA level 2 section identifiable. It allows a parser to identify in CDA level 3 that the data present in the human readable part came from several different source documents, what the id of those documents is, how old they are, and what information came from what document. The second entry *level 2 reference* allows referencing *level 2 entry* from a different section. It contains only the identifier of *level 2 entry*.

The CDA template described above is used in different ways when generating the sections of the patient summary. If the source data for a section was only extracted from level 2 documents the generated section will contain all of those data-items combined. The items are sorted by creation date of the source-document from newest to oldest. Above of each extracted text a header is generated that contains the source-document type and creation date. At the beginning of the section an overview table is generated that contains a link to each of the headers in the section. This was done to present a short overview of the items contained in the section for the reader. Each of the added items also contains a level 3 entry *level 2 entry*. This allows separating the data when the patient summary itself is processed, and also improves performance of said processing compared to splitting the data from level 2. If the source data was only extracted from level 3 documents the level 3 entries are added to the section, and the human readable level 2 is auto-generated by different templates depending on the type of data, all sorted by age, newest first. If data that belongs in one section was extracted from both level 2 and level 3 documents then the *level 2 reference* is applied. instead of mixing the items only the level 3 entries are added to the section. For each level 2 item a *level 2 reference* is added in level 3. In the human readable part of the section the level 3 entries are generated exactly as if only level 3 data was present. In adddition for each *level 2 reference* a link is added that points to a human readable enty contained in an additional section at the end of the document. This additional section contains all level 2 data that is referenced by a *level 2 reference*. The section is generated exactly the same way as a pure level 2 section would be. This method allows mixing level 2 and level 3 data items while maintaining one view, for example a table, wich is still sorted by date and contains either the data from the level 3 element or a link instead of the data.

Figure 5 shows the concept in action on a discharge diagnosis. The section on the upper part of the figure was generated from cda level 3 data and contains *level 2 references* which are represented in the human readable format as the link ”see excerpt of source document”. This link refers to the document excerpt of the same date in the section in the lower part of the image. This section only contains *level 2 entries*. The patient summary in the figure was generated with test-data and does not contain information of a real patient.

To make it traceable where the data in the patient summary was extracted from a different CDA template was defined. This template contains an item which can be added to any CDA level 3 entry. It contains the IHE document id, repository id and the homeCommunity id the data was extracted from. Using this information each level 3 data entry, as well as the previously defined *level 2 entry*, in a patient summary can be traced to its original source document. If different datasources like a medical databases are used a similar template can be defined to trace the generated entry to the source.

The data contained in the patient summary was extracted from several different types of documents which were at the time of writing defined in ELGA. Since these are not internationally applicable only the general types of documents and data extracted from the documents will be discussed. Table I shows which document types were used and what data was extracted from them. If these items are compared to Figure 1 several differences between the implementation and the data suggested by epSOS [3] and ELGA [4] can be observed. The extracted data item diagnosis cannot be found in Figure 1. From the diagnosis medically trained professionals can determine *Problems*, which are contained in the patient summary definitions of both epSOS [3] and ELGA [4]. Furthermore the section laboratory result is not contained in the patient summary definitions of epSOS and ELGA either. From the laboratory results the bloodgroup of a patient which

TABLE I
DOCUMENT TYPES AND EXTRACTED DATA FOR A PATIENT SUMMARY

Document	Data
Discharge letter	Diagnosis Vital Signs Allergies Procedures
e-Medication	Medication
Laboratory result	Laboratory data
Any document	Contact Information Patient Information Document References Family Physician

is contained in *Results*, as well as *Immunizations* and *Problems* can be identified. Further the sections *Pregnancies*, *Social History*, *Medical Devices* and *Implants*, *Functional Status* and *Therapeutic Recommendations* are completely missing from the presented solution. These sections are missing because the available source documents for testing did not contain this information.

The generated documents were validated in two steps. First a basic xml validation was applied to check if the generated patient summary is well formed. Secondly a schematron validation was applied to check if the document is a valid CDA document. Further checks were omitted since the generated patient summary contains self-defined templates which are not part of any defined document type as of yet.

Detailed information on the findings described in this section, such as logs, cda templates and the extracted data items, can be found in [19].

IV. CONCLUSION

The fully automated generation of patient summary documents from the data existing in a healthcare environment on demand is possible. It remains to be seen if the generated patient summaries hold enough value for a physician to quickly assess a patients health-situation. This has to be evaluated in a next step for this research, in cooperation with medical professionals, possibly in the form of a clinical study .

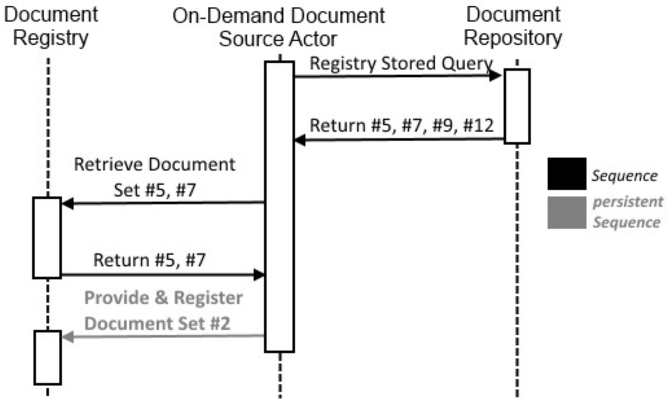


Fig. 6. Sequence of generating a patient summary.

Concerning the processing of the existing data in a healthcare environment, the presented solution still leaves open the question how to handle CDA Level 1 data. Testing the presented method showed that the CDA source documents did not hold enough structured information [4]. The metadata of many source-documents also did not hold the information of the document-type. This forced the solution to ignore the types of the available documents, resulting in a performance loss compared to the intended solution of only processing document-types that hold information relevant for a patient summary. It was determined that most of the available data existed in Level 2. This reinforces the correctness of the decision to include Level 2 data, which only might have valuable information, since the information available in Level 3 is at the time of writing in no way enough to generate a useful patient summary.

The repercussions of using the IHE system itself as a data source for the On-Demand Document Source Actor instead of an external data source was not considered in the definition of the actor. This, however, would be an interesting application, as doing so would allow for data-aggregation over several healthcare facilities by using an already interoperable system. The results show that the persistent mode of the ODD Source Actor should be implemented if it is important to exactly determine who accessed what information. If access rights are more relevant though, for example because of data-privacy laws [20] [21] as was the case when implementing the solution, the non-persistent mode of the ODD Source Actor should be implemented.

REFERENCES

[1] Auer Clemens, Milisits Carina, Reimer Sebastian, *ELGA-Handbuch: Die Elektronische Gesundheitsakte*. Vienna, Austria: Manz, 2014.

[2] EU Commission Directorate General Health & Consumers, *Guidelines on minimum/nonexhaustive patient summary dataset for electronic exchange in accordance with the cross-border directive 2011/24/EU*. 2013. http://ec.europa.eu/health/ehealth/docs/guidelines_patient_summary_en.pdf. Last access 2015-01-31.

[3] Spanish Ministry of Health and Social Politics (ESNA), *Final definition of functional service requirements - Patient Summary*. 2012. http://www.epsos.eu/uploads/tx_epsosfileshare/D3.2.2_Final_Definition_Functional_Service_Req_Patient_Summary.pdf. Last access 2015-01-31

[4] Stefan Sabutsch), *Das Patient Summary aus der ELGA Perspektive*. 2013. http://www.ehealthsummit.at/sites/default/files/ehsat_Notfalldaten%20und%20Patient%20Summary_Sabutsch.pdf. Last access 2015-01-31

[5] Paizoni, *A Rule-Based Summarization Approach for CDA Documents in an IHE compliant System*. Master Thesis, University of Applied Sciences Upper Austria, Hagenberg, 2011.

[6] D. Bosca, L. Marco, D. Moner, J. A. Maldonado, L. Insa, M. Robles, *Detailed Clinical Models Governance System in a Regional EHR Project*. XIII Mediterranean Conference on Medical and Biological Engineering and Computing 2013, IFMBE Proceedings Volume 41, pp. 1266-1269, 2014.

[7] ITI Technical Committee, *IHE IT Infrastructure (ITI) Technical Framework Volume 1*. 2014. http://www.ihe.net/uploadedFiles/Documents/ITI/IHE_ITI_TF_Vol1.pdf. Last access 2015-01-31.

[8] C. Chronaki, A. Estelrich, G. Cangioli, M. Melgara, D. Kalra, Z. Gonzaga, L. Garber, E. Blechman, J. Ferguson, S. Kay, *Interoperability standards enabling cross-border patient summary exchange*. Studies in Health Technology and Informatics 205:256-260, 2014.

[9] Health Level Seven International, *HL7 Clinical Document Architecture*. 2005. http://www.hl7.org/implement/standards/product_brief.cfm?product_id=7. Last access 2015-01-31.

[10] Keith W. Boone, *The CDA TM Cookbook*. London, Springer, 2011.

[11] Christoph Rinner, *Inhaltsbasierte Suche in standardisierten elektronischen Gesundheitsakten mittels Archetypen*. Technical University Vienna, Faculty for Informatics, 2013.

[12] Christoph Rinner, M. Kohler, G. Hbner-Bloder, S. Saboor, E. Ammenwerth, G. Duftschmied, *Vermeidung EHR-bedingter Iinformationsberflutung mittels des Zweimodell Ansatzes*. Proceedings of eHealth 2012, pp. 95-101, 2012.

[13] Christoph Rinner, M. Kohler, G. Hbner-Bloder, S. Saboor, E. Ammenwerth, G. Duftschmied, *Searching for document contents in an IHE-XDS EHR architecture via Archetype-based indexing of document types*. Proceedings of Medinfo 2013, Stud. Health Technol. Inform. 192:1092, 2013.

[14] G. Duftschmied, Christoph Rinner, M. Kohler, G. Hbner-Bloder, S.Saboor, E.Ammenwerth, *The EHR-ARCHE project: Satisfying clinical information needs in a Shared Electronic Health Record System based on IHE XDS and Archetypes*. International Journal of Medical Informatics, 82(12):1195-207, 2013.

[15] Health Level Seven International, *HL7 V3 Guide* Release 1. 2012. http://www.hl7.org/implement/standards/product_brief.cfm?product_id=186. Last access 2015-01-31.

[16] LOINC, *Logical Observation Identifiers Names and Codes (LOINC)*. 2015. http://loinc.org. Last access 2015-01-31.

[17] Tim Benson, *Principles of Health Interoperability HL7 and SNOMED, Health Information Technology Standards* 2nd Edition. London, Springer, 2012.

[18] ITI Technical Committee, *IHE IT Infrastructure (ITI) Technical Framework Volume 2b*. 2014. http://www.ihe.net/uploadedFiles/Documents/ITI/IHE_ITI_TF_Vol2b.pdf. Last access 2015-01-31.

[19] Krauss Oliver, *Automatisierte Erstellung eines Patient Summary durch Verwendung des IHE On-Demand Document Supplement*. Master Thesis, University of Applied Sciences Upper Austria, Campus Hagenberg, Faculty Software Engineering, 2014.

[20] Nationalrat, *Bundesgesetzblatt fr die Republik sterreich 111. Bundesgesetz: Elektronische Gesundheitsakte-Gesetz ELGA-G. GP XXIV RV 1936 AB 1979 S. 179. BR: AB 8818 S. 815*. 2012. http://www.elga.gv.at/fileadmin/user_upload/uploads/download_Papers/Gesetze_u.a._Rechtsgrundlagen/BGBLA_2012_I_111.pdf. Last access 2015-01-31.

[21] Nationalrat, *Bundesgesetzblatt fr die Republik sterreich 165. Bundesgesetz Datenschutzgesetz 2000 DSG 2000 GP XX RV 1613 AB 2028 S. 179. BR: 5992 AB 6034 S. 657*. 1999. http://www.elga.gv.at/fileadmin/user_upload/uploads/download_Papers/Gesetze_u.a._Rechtsgrundlagen/DSG_2000.pdf. Last access 2015-01-31.