MEDINFO 2013
C.U. Lehmann et al. (Eds.)

doi:10.3233/978-1-61499-289-9-903

Building a Time-Saving and Adaptable Tool to Report Adverse Drug Events

Yves Parès, Gunnar Declerck, Sajjad Hussain, Romain Ng, Marie-Christine Jaulent

INSERM, UMRS 872, Eq. 20, Paris, France

Abstract

The difficult task of detecting adverse drug events (ADEs) and the tedious process of building manual reports of ADE occurrences out of patient profiles result in a majority of adverse reactions not being reported to health regulatory authorities. The SALUS individual case safety report (ICSR) reporting tool, a component currently developed within the SALUS project, aims to support semi-automatic reporting of ADEs to regulatory authorities. In this paper, we present an initial design and current state of of our ICSR reporting tool that features: (i) automatic pre-population of reporting forms through extraction of the patient data contained in an Electronic Health Record (EHR); (ii) generation and electronic submission of the completed ICSRs by the physician to regulatory authorities; and (iii) integration of the reporting process into the physician's work-flow to limit the disturbance. The objective is to increase the rates of ADE reporting and the quality of the reported data. The SALUS interoperability platform supports patient data extraction independently of the EHR data model in use and allows generation of reports using the format expected by regulatory authorities.

Keywords:

Adverse Drug Events Reporting, Semantic Interoperability, Secondary Use of EHR.

Introduction

Post-market drug surveillance studies currently face an under-reporting problem [1,2]. Suspected adverse drug reactions must be spontaneously reported by medical professionals to the regulatory authorities. However, only a small minority of adverse drug events (ADEs), around 5%, is reported [3,4]. For instance, in the United States less than 1% of ADEs are reported to the Food and Drug Administration (FDA), despite being frequently described in the electronic health record (EHR) systems [5]. This led to the discovery that this alarming situation is due to two facts:

- detecting ADEs is a complex cognitive process (a causal relation between a given drug and a given observable condition must be hypothesized), and can be overlooked by busy physicians,
- filling out a report on a specific occurrence of an ADE (called an ICSR, for individual case safety report) is a tedious, time-consuming and error-prone operation. Medical professionals are not always aware enough of the importance of reporting for patient safety to spend time on this activity.

To increase the rates of ADE reporting, we argue that encountered ADEs from EHR in a hospital could be semi-automatically sent to the regulatory authorities. At present, the reporting process is almost fully manual and paper-based in most countries. Therefore, a possibility of semi-automatic submission of encountered ADEs as ICSR forms, would drastically improve the current practices [6].

The SALUS project [7] aims to build a system (which we call here the SALUS platform) of several components that would be integrated in hospital work flows in order to facilitate post-marketing drug surveillance and pharmacovigilance. To achieve an acceptable level of integration for ADE reporting, our ICSR reporting system addresses several issues regarding adaptability and interoperability:

- Reusing information stored in the EHR independently of the data model in use, so that the health practitioner (HP) may never have to manually input elements of data that are already known in the EHR (the input form should be pre-populated), which also carries the beneficial side-effect of decreasing the risk of typos in the reports,
- Adapting to the ICSR report format as well as the sending protocol and medium required by the local authority, i.e., the regulatory authority to which the hospital must send the reports,
- Adapting to the HPs' habits and preferences so that the reporting process may disturb them the least possible.

In this paper, we will first provide some insights about how the SALUS platform aims to solve the first point and then focus on how our ongoing work on the *ICSR reporting system* will eventually address the last two points. The ICSR reporting system is among the set of components of the SALUS platform, which handles the interaction between the HP and the local authority. The general approach is to raise the level of abstraction on which the platform works, and to reduce the dependencies on the specificities of the hospital, EHR and authority concerned.

Related Work

There have been some earlier attempts regarding EHR data extraction for ICSR pre-population [5] as well as automatic ADE detection [3]. The proof of concept of the ASTER project [5,8] shows how data can be extracted from an EHR and sent to the FDA. The pre-populated report contains data relative to demographics and product name when the physician sees it. The physicians who were involved in ASTER testing process agreed on its interest for their daily work. ASTER had some limitations that SALUS aims at overcoming, such as (i) the extraction being tied to a specific EHR data model, and (ii) the generated ICSR report form not following the FDA specifications for reporting and necessitating an intermediate manual processing. Facilitating the pre-population step from the EHR data model has been addressed by the IHE Drug Safety Content profile (DSC) [9], an integration profile built on top of the Retrieve Form for Data Capture profile (RFD) [10]. RFD specifies a generic protocol for handling information collected from EHRs, and DSC describe the conversion of data from HL7 Continuity of Care Documents (CCD) to the standard E2B data model [11]

used for ADE reporting. The limitations are that mappings to E2B are only partially defined, and reversible pseudonymization of personal information, a step that might be necessary depending on the local laws, is not covered.

Materials and Methods

To efficiently report detected ADEs, we propose a time-saving and adaptable tool to report ADEs [6] which targets three aspects of the interoperability problem:

- *EHR data extraction*: Finding and formalizing the set of the common data elements (CDEs) between the different content models, making then the reporting tool at the step of pre-populating the ICSR form, oblivious of which content model is actually queried in the EHR.
- *User interface*: Adapting the contents of the reporting system input form so that it complies with local authority's format, which the reporters are used to and then expect to encounter.
- *ICSR sending*: Following the local authority protocols.

The main technical steps in building our ICSR reporting systems for solving the above-mentioned problem are discussed in the following subsections.

SALUS Interoperability Approach through an Ontology of Common Data Elements

It would be necessary to make N*(N-1) mappings when we define structural mappings between information models through syntactic mapping mechanisms. The SALUS alternative is to use a common ontology, the SALUS Core Ontology (SCO), currently under development [12]. It has the role of representing the semantics of reference information models used by both the EHR systems and the regulatory authorities. The SALUS Core ontology aims to act as a common denominator for the set of information based on the already existing standards used in clinical care (HL7 CCD, EN 13606 EHR Extracts or proprietary models). It is built through a systematic approach by examining the content models, extracting and harmonizing CDEs from these and representing the related terminology systems as ontologies and linking them with the CDEs in an ontological framework [12].

ICSR Reporting System

Our ICSR reporting system is based on the SALUS platform and designed to ensure the possibility of extracting patient data needed to prepopulate ICSR forms, by converting the data elements expressed in the information models used in the EHRs to ICSR data models used by the local authorities. The ICSR data model on which we focus is ICH E2B(R2) [11], the standard used by the World Health Organization (WHO) Collaborating Centre for International Drug Monitoring. It specifies both an information model for ICSRs and a protocol for their electronic transmission. The EHR systems use a diversity of content models; and ICH E2B(R2), albeit tending to be an international standard, is still underutilized by regulatory authorities.

Then, while using the E2B model and protocol to prototype our ICSR reporting system, we have to keep in mind that eventually it will not be the only specification that the reporting system should be able to use. This means that we have to work along the way to minimize the efforts needed to adapt the tool to other models.

Initial Design of the ICSR Reporting System

The ICSR reporting system is designed to ensure

- All transactions with other SALUS platform components necessary to extract patient data from the EHR to prepopulate the ICSR;
- All operations necessary to complete the filling of the ICSR in compliance with the specifications and its correct transmission to the pharmacovigilance regulatory authorities.

The ICSR reporting system is composed of 3 main components: ICSR reporting manager, ICSR reporting tool, and ICSR report generator. Interfaces and high-level connections inside the system and with the other components of the SALUS platform are shown in Figure 1.

The ICSR reporting tool supports several additional functionalities: recording an ICSR to be completed and reported later; accessing previously sent and waiting to be completed ICSRs; updating and sending an ICSR reported in a previous session; and finalizing and sending an ICSR. The ICSR prepopulation process can be triggered in two different

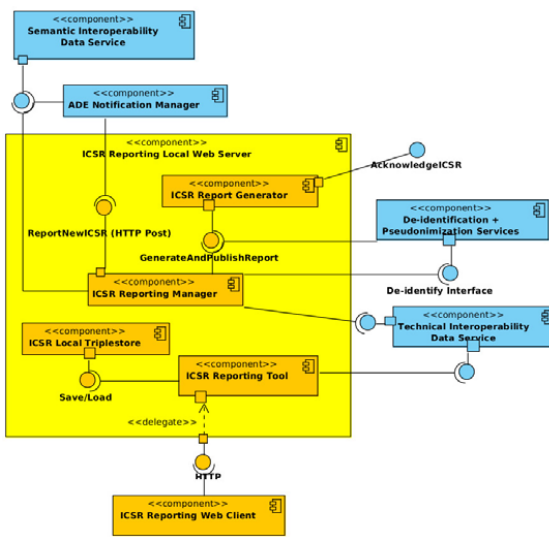

Figure 1 - Components of the ICSR reporting system

circumstances: (i) the HP decides to report an ADE he himself detected or (ii) the ADE notification tool, another key component of the SALUS platform performing continuous screening of EHR data, detects a potential ADE and suggests through an alert for the HP to complete a report.

The ICSR reporting tool is a web application and the ICSR reporting web client is the set of web pages which the HP accesses. The ICSR reporting manager (IRM) is part of the SALUS platform Semantic Services. It consists of the interoperability components that have to deal with every content model and convert between them through the Core Ontology, in contrast to the rest of the components that might ask and operate on a specific content model. The IRM gets invoked by the ADE notification manager if new ADE notifications need to be reported, it acts then mostly as a routing component, as the real work is done by the component to which it sends the data: the ICSR reporting tool (IRT). It is warned by the IRM (through the Technical Interoperability Data Service, a layer that separates the Semantic Services from the rest of the SALUS platform) in order to generate, pre-fill the ADE report and have the physician extend it with additional information using the Reporting web client.

The IRT invokes the ICSR local triplestore service to save and load both sent and pending ICSR reports. The ICSR local triplestore component addresses the storage of the administrative data concerning the physicians and the hospital, as this is not retrievable from the EHR but still not likely to change, so we want it to be pre-populated as well. It also handles the storage of the acknowledgments when the protocol requires the authority to send them, offering to the reporter the ability to track them.

The IRM interacts with the components of other systems. It invokes the Semantic Interoperability Data Service to retrieve relevant patient data from the local EHR in the form of RDF triples represented in SALUS Core ontology. It invokes the De-identification Service, which removes or replaces the patient identifiable data and then invokes Pseudonymization Service that generates a replacement identifier for the patient ID. It is then the task of the Pseudonymization Service to send the data to the ICSR Report Generator so that it can build the final report in the mandated format and send it to the regulatory authorities and/or pharmacovigilance centre(s), or simply print it out if the authority requires manual signatures and physical forms.

ICSR Reporting Tool: Design and Interface

To illustrate the need of having an easily adaptable user interface, we consider the example of the E2B(R2) international standard versus the Italian AIFA model. The authorities working with the AIFA model require much less content than the Uppsala Monitoring Centre, the antenna of the World Health Organization for international drug monitoring which makes use of E2B. AIFA results then in a far lighter content model. As we intend to integrate the system within the regular hospital workflow, we shouldn't require the reporting physician to fill in data not required by authorities, nor should he encounter unnecessary fields that should remain blank.

However, the idea of not showing unnecessary fields goes even farther. For this purpose, E2B(R2) specifies that some fields should be completed only in certain circumstances. For instance, all the fields concerning the seriousness of the adverse drug event are only of use if the reporter has actually selected that the reaction was found serious.

Bearing these facts in mind, we drew the required functionalities that our ICSR reporting tool needs to address:

- Quickly adapt to the local standard and language,
- Dynamically fold and unfold parts of the form as needed,
- Dynamically verify that the completed fields follow the specification in use, for instance the terminology required.

The two last points are what we call the *controls* of the form.

In contrast to available solutions for making a template of the web interface and populating it afterwards[1], we adopted an expressive domain specific language (DSL) [13]. We call this DSL the *ICSR form description*. It allows us to describe the ICSR form, change this description whenever needed, and let the system fully derive the interface appearance and controls. The ICSR form description would be obtained ideally by a direct transcription of the authority's specification. For instance, E2B(R2) is quite thorough and provides a lot of sections that are very much alike. The sections about the reporter and the physician, for instance, feature the same fields, only for different persons. We do not want to have to repeat the exact same sequence of fields several times and alter them all when a change needs to be done. The ICSR form description DSL is hosted and implemented by the general-purpose language Clojure[2]. It allows us to efficiently perform raw structural data manipulation and design expressive DSLs.

Results

The SALUS ICSR reporting tool aims to provide an interactive interface to the practitioners for reporting the detected ADEs in a standardized format to regulatory authorities. The tool enables automatic pre-population of ICSR by (*i*) extracting data available in the patient EHR and converting this data to the data model requested for reporting, (*ii*) providing assistance to the manual completion of information that couldn't be automatically prefilled, and (*iii*) providing assistance for transmitting the reporting form to regulatory authorities. An overview of the user interface in the current state of the project is provided in Figure 2 which shows the GUI in perspective with a fraction of the E2B(R2) file that is produced (shown in top-left part).

To ensure syntactic interoperability between E2B and data models used in EHR systems, those models have been mapped and represented in SALUS core ontology. The general objective of those mappings is to enable the ICSR prepopulation process by ensuring that patient data needed by (*i*) the E2B(R2) standard form or (*ii*) national adverse event reporting forms (AIFA Italian form for the moment) can be extracted automatically from the patient EHR DWH. Data models that have been mapped to date include: E2B(R2), HL7 CCD templates, ISO/CEN EN 13606 based templates (to represent medical summaries), OMOP Common Data Model. The SALUS core ontology comprises a set of data elements (SALUS CDE) used as a bridging model: a unique correspondence table is used to link the different data models together. To design the prepopulation mechanisms we have consequently built SPARQL queries targeting directly the SALUS CDEs, not the data models used in the EHRs (e.g., CDA xpaths). Medical summaries are represented as N3 instances of SALUS CDEs, so that the ICSR Reporting Tool only has to query the later to extract the data required for prepopulation.

Discussion

The main principles that have been used to guide the conception of SALUS ICSR reporting tool have several advantages compared to current ADE reporting solutions. Existing ADE reporting systems (e.g., *ASTER reporting tool, FDA MedWatch Online Voluntary Submission System, UMC Vigiflow*) are step-by-step systems comprised of elaborated forms, and do not focus on requesting relevant data, which requires extensive amount of time from HPs to complete the forms. Whereas, SALUS ICSR reporting tool provides user-customizable forms, displayed on one single page, with already pre-populated fields and leave the decision to the user to revise existing data and provide additional ones.

The choice of a web application also enhances interoperability, ease of installation in a hospital and coherence with other SALUS tools. Using web standards allows far easier extension to hand-held devices: running the ICSR reporting web client on a tablet that could connect to the information system of the hospital, thus allowing its use in bed-side care, is but the logical extension if we pursue the idea

[1] http://www.oracle.com/technetwork/java/javaee/jsp/index.html, http://php.net or http://github.com/cgrand/enlive are examples.

[2] http://clojure.org

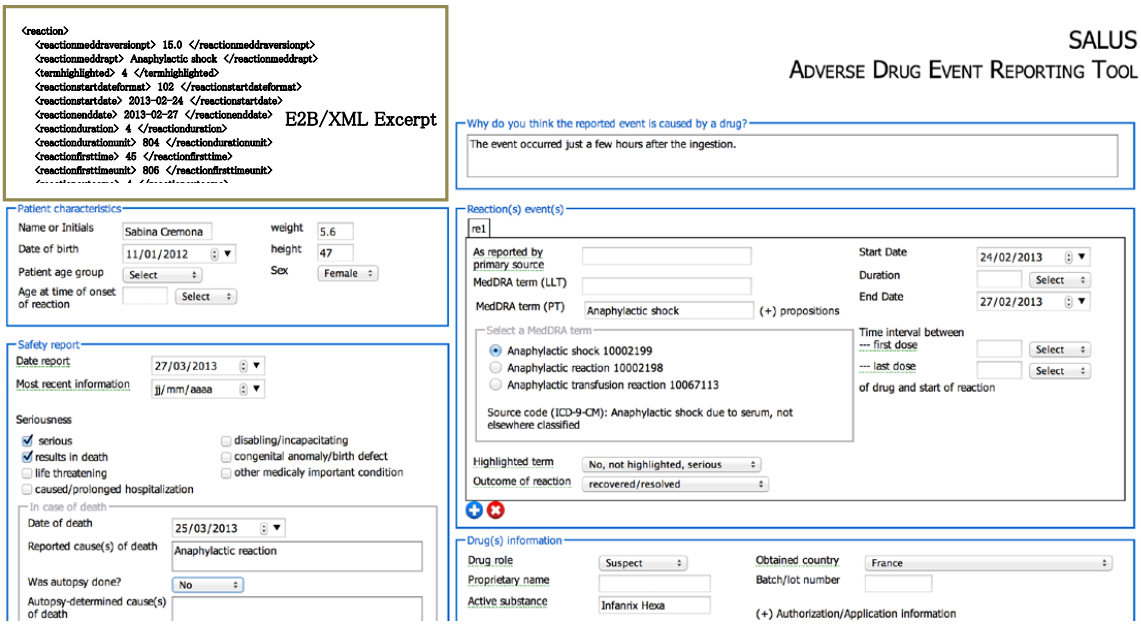

Figure 2 - GUI of the reporting tool, where fields are prepopulated.
The E2B/XML excerpt shows (top-left) what is generated out of the Reaction event(s) section.

of reducing HP's overhead and augmenting his capacity to report quickly. Thanks to those design principles, we hope that the main objective of the tool will be reached: ease the ICSR reporting process and make it more attractive for HPs; reduce time necessary to fill ICSR forms; reduce errors due to double data entry; and increase the quality of the reported data (and thus its usability to detect new ADRs).

Next phase of SALUS project is to include a test phase with a real implementation of the tool in hospitals. SALUS project is currently still in the prototype phase and no end user evaluation has been performed yet. A first prototype working with fake EHR data and reduced functionalities has been produced for Month 14 (March 2013). The evaluation phase with the two pilot sites end users will only begin once a second prototype, fully integrated with other SALUS platform components, will be produced (planed for Month 24, January 2014). Three test/evaluation phases will be performed: (1) a testing phase checking the robustness of the prepopulation and conversion mechanisms with real EHR data; (2) a functional and non-functional software quality evaluation following the evaluation reference model ISO/IEC CD 25040, evaluating e.g., the time it takes to prepopulate the ICSR form or its ability to manage fault cases (the user cancels the reporting procedure, the system crashes, etc.); and (3) end user evaluation in pilot site with real implementation of the tool. For the end user test phase, the following evaluation criteria will possibly be taken into account: if the IRT (*i*) facilitates the work of the GP filling ICSR forms and minimize the time needed to complete them; and (*ii*) maximize the relevant information that can be used by pharmacovigilance regulatory bodies. The methodology relevant for such an evaluation remains to be decided in its details. End users satisfaction questionnaires can be used. Measure of the mean time needed to fill and send ICSR with and without the tool can also be compared. The quality of the data made available to the ICSR reporting system shall be evaluated by pharmacovigilance professionals.

Concerning the ICSR form description DSL, some additional concerns can be raised:

- We put one restriction onto how the ICSR form is displayed in the user interface: that it should fit on one page. This seemed first like a legitimate concern and simplification, but our goal is to adapt to a reporter's preferences. We have to consider that maybe some reporter would prefer a paginated form. As this would not normally alter the ICSR form definition and as the technology we use could be leveraged to do that, it would be a good idea to add this flexibility, and allow the generation of sections scattered on several pages;
- We specify how the content (fields, sections and controls) is laid out orthogonal to what the content itself is. However, currently it is not as easy to modify simple layout generation rules as it is for the content. Just like we built a DSL to describe the content, we could build one to describe the rules that govern its final display;
- DSL makes readable and easily modifiable descriptions but might require some learning from the medical professionals. A good extension in a second phase of the SALUS project (i.e., after its installation and testing in the pilot-sites of the project) would be – if the adaptability proves valuable – an editor targeting the reporters themselves.

Furthermore, future challenges that were not predicted upfront will have to be addressed by the SALUS project:

- ICSR pre-population can only work if structured data is available in the EHR. However, tests conducted in the pilot-sites (in Germany, the Technical University of Dresden and in the Lombardy region in Italy) showed that a substantial amount of data in the EHR is still only free text instead of exploitable structured data [14]. Natural language processing tools may be needed to exploit EHR data;
- Sometimes, mappings between E2B data elements and those data models could only be partially achieved: some E2B sections are simply not present in other data models (e.g., “Seriousness of the ADE” or “Recurrence of ADE on readministration” have no corresponding

section in HL7 CCD templates) or value sets only partially overlap. Conversion mechanisms need to be used.

- We need to ensure the mappings between terminologies, as E2B requires the use MedDRA for medical data whereas EHR systems often resort on LOINC, ICD10 or SNOMED-CT. This is not a straightforward operation, as the terminologies often have various levels of granularity, then mappings can only be approximate. A checking of pre-populated data will consequently be necessary from the physician before submitting the ICSR to the regulatory authorities. Secondly, terminologies are evolving; therefore those mappings need to be regularly updated. One solution that will be used in SALUS to address this problem is reusing current available mappings banks, as Bioportal or UMLS metathesaurus;
- Medical judgment must be exercised to select the relevant patient history items from the EHR. A fully automated extraction of data is consequently not possible: a manual selection must be made. However the tool could support the physician in the selection of relevant data, e.g. in making available a browser helping to select in the EHR the entries that must be transferred to the ICSR form (e.g., a particular event in the past drug history section which is pertinent for the understanding of the case);
- We also need to take the country-specific ethical and legal dimension into account. Patient data must generally be de-identified before being accessed and can't leave the clinical care zone, i.e., the zone where identified data is maintained and accessed locally. For patient privacy reasons, the ICSR has also to be de-identified and pseudonymized before being sent to regulatory authorities. In some circumstances, this phase can be skipped, but this remains exceptional and depends on national regulatory policies. For instance, the German law states that the name of the patient must be kept in the report in cases of ADE representing a potential danger to the entire population.

Conclusion

The SALUS ICSR reporting tool aims to simply a process that is considered crucial for post-market clinical studies. Our hypothesis is that alleviating this process would both induce physicians to report more and increase the quality of the reports. Two axes are to be privileged to reach this goal: pre-population through EHR data re-use and integration of the tool in the regular physicians' environment and work flow. However, while fully automatic reporting is not within reach, we can make the HP's experience the easiest possible as his expertise and intervention are still needed.

Acknowledgments

The research leading to these results has received funding from the European Community's Seventh Framework Programme (FP7/2007-2013) under grant agreement no ICT-287800, SALUS Project [7].

References

[1] Hazell L, Shakir SA. Under-reporting of adverse drug reactions: a systematic review: Drug Safety, 29(5), 2006; pp. 385-396.

[2] van der Heijden PG, Puijenbroek EP, van Buuren S, van de Hofstede JW. On the assessment of adverse drug reactions from spontaneous reporting systems: The influence of under-reporting on odds ratios: Stat Med 21, 2002; pp. 2027-2044.

[3] Bates DW, Evans RS, Murff H, Stetson PD, Pizziferri L, Hripcsak G. Detecting adverse events using information technology: Journal of the American Medical Informatics Association 10(2), 2003; pp. 115-128.

[4] Cullen DJ, Bates DW, Small SD, Cooper JB, Nemeskal AR, Leape LL. The incident reporting system does not detect adverse drug events: a problem for quality improvement: Jt. Comm. J. Qual. Improv. 21; pp. 541-548, 1995.

[5] Linder JA, Haas JS, Iyer A, Labuzetta MA, Ibara M, Celeste M, Getty G, Bates DW. Secondary use of electronic health record data: spontaneous triggered adverse drug event reporting: Pharmacoepidemiol Drug Saf. 19(12), 2010; pp. 1211-1215.

[6] Declerck G, Hussain S, Parès Y, Daniel C, Jaulent MC, Laleci GB. Semantic-sensitive extraction of EHR data to support adverse drug events reporting. In: Proceedings of the Semantic Web Application and Tools for Life Science Workshop, Paris, France, 2012.

[7] SALUS Project, Scalable, Standard based Interoperability Framework for Sustainable Proactive Post Market Safety Studies [Internet]. Available from: http://www.salusproject.eu

[8] Brajovic S, Piazza-Hepp T, Swartz L, Dal Pan G. Quality assessment of spontaneous triggered adverse event reports received by the Food and Drug Administration: Pharmacoepidemiol Drug Saf. 21(6), 2012; pp. 565-570.

[9] IHE Quality, Research and Public Health (QRPH) Technical Framework Supplement Drug Safety Content Profile (DSC), Integrating the Healthcare Enterprise (IHE), Trial Implementation Supplement, Aug. 2010.

[10] IHE IT Infrastructure Technical Framework Supplement, Retrieve Form for Data Capture (RFD), Trial Implementation, Aug. 19, 2011.

[11] ICH guideline E2B (R2), Electronic transmission of individual case safety reports - Message specification (ICH ICSR DTD Version 2.1), Final Version 2.3, Document Revision Feb. 1, 2001.

[12] Laleci GB, Yuksel M, Dogac A. Providing Semantic Interoperability between Clinical Care and Clinical Research Domains: Accepted for publication in IEEE TITB. Available from http://www.srdc.com.tr/publications/2012/SALUSSemanticInteroperability.pdf.

[13] Zdun U. Concepts for Model-Driven Design and Evolution of Domain-Specific Languages. In: Proceedings of the International Workshop on Software Factories at OOPSLA 2005, San Diego, CA, USA, 2005; pp. 1-6.

[14] SALUS Deliverable D8.1.1: Pilot Application Scenario and Requirement Specifications of the Pilot Application. May 28, 2012. Available in the Public Document section of http://www.salusproject.eu.

Address for correspondence

Yves Parès, PhD student
email : yves.pares@gmail.com
Centre de Recherche des Cordeliers
15 rue de l'école de médecine, Escalier D
75006 Paris, France