Gathering and Exploring Scientific Knowledge in Pharmacovigilance

Pedro Lopes[1], Tiago Nunes[1], David Campos[1], Laura Ines Furlong[2], Anna Bauer-Mehren[2], Ferran Sanz[2], Maria Carmen Carrascosa[2], Jordi Mestres[2], Jan Kors[3], Bharat Singh[3], Erik van Mulligen[3], Johan Van der Lei[3], Gayo Diallo[4], Paul Avillach[4,5], Ernst Ahlberg[6], Scott Boyer[6], Carlos Diaz[7], José Luís Oliveira[1*]

1 DETI/IEETA, University of Aveiro, Aveiro, Portugal, **2** Research Programme on Biomedical Informatics (GRIB), IMIM Hospital del Mar Research Institute and Universitat Pompeu Fabra, Barcelona, Spain, **3** Erasmus University Medical Center, Rotterdam, The Netherlands, **4** LESIM-ISPED, Université de Bordeaux, Bordeaux, France, **5** LERTIM, EA 3283, Faculté de Médecine, Université de Aix-Marseille, Marseille, France, **6** AstraZeneca, Molndal, Sweden, **7** Synapse Research Management Partners, Barcelona, Spain

Abstract

Pharmacovigilance plays a key role in the healthcare domain through the assessment, monitoring and discovery of interactions amongst drugs and their effects in the human organism. However, technological advances in this field have been slowing down over the last decade due to miscellaneous legal, ethical and methodological constraints. Pharmaceutical companies started to realize that collaborative and integrative approaches boost current drug research and development processes. Hence, new strategies are required to connect researchers, datasets, biomedical knowledge and analysis algorithms, allowing them to fully exploit the true value behind state-of-the-art pharmacovigilance efforts. This manuscript introduces a new platform directed towards pharmacovigilance knowledge providers. This system, based on a service-oriented architecture, adopts a plugin-based approach to solve fundamental pharmacovigilance software challenges. With the wealth of collected clinical and pharmaceutical data, it is now possible to connect knowledge providers' analysis and exploration algorithms with real data. As a result, new strategies allow a faster identification of high-risk interactions between marketed drugs and adverse events, and enable the automated uncovering of scientific evidence behind them. With this architecture, the pharmacovigilance field has a new platform to coordinate large-scale drug evaluation efforts in a unique ecosystem, publicly available at http://bioinformatics.ua.pt/euadr/.

Citation: Lopes P, Nunes T, Campos D, Furlong LI, Bauer-Mehren A, et al. (2013) Gathering and Exploring Scientific Knowledge in Pharmacovigilance. PLoS ONE 8(12): e83016. doi:10.1371/journal.pone.0083016

Editor: Dermot Cox, Royal College of Surgeons, Ireland

Received May 21, 2013; **Accepted** November 8, 2013; **Published** December 11, 2013

Funding: This work was supported by the European Commission (EU--ADR, ICT--215847), FCT (PTDC/EIA-- CCO/100541/2008), and Instituto de Salud Carlos III FEDER (CP10/00524). The Research Programme on Biomedical Informatics (GRIB) is a node of the Spanish National Institute of Bioinformatics (INB). The funders had no role in study design, data collection and analysis, decision to publish, or preparation of the manuscript.

Competing interests: The authors E Ahlberg and S Boyer work for Astrazeneca, Molndal, Sweden. C Diaz works for Synapse Research Management Partners, Barcelona, Spain. This does not alter our adherence to all the PLOS ONE policies on sharing data and materials.

* E-mail: jlo@ua.pt

Introduction

Pharmacovigilance plays an essential role in the post-market analysis of newly developed drugs [1, 2]. Pharmaceutical companies' competition along with rigorous regulatory evaluation procedures empowers a complex research and development process before launching a new drug into the market. Notwithstanding, drug safety continues to be a relevant concern for healthcare involving worldwide policy stakeholders, from regulatory health authorities to specialised law firms. Post-market pharmacovigilance complements the traditional pre-market austere drug approval process, where the European Medicines Agency (EMA) [3] and the US Food and Drug Administration (FDA) [4] establish guidelines for new medicine approval, requiring intense testing and trials [5]. Along with these recommendations, pharmaceutical companies must also define thorough risk management plans for post-market drug stages [6, 7].

Pharmacovigilance research is based on the analysis of "signals". The World Health Organization (WHO) defines signals as undisclosed assertions on direct relationships between adverse events – effects on the human organism - and a drug [8]. To generate comprehensive signal datasets, clinicians and researchers use spontaneous reporting systems (SRS). Electronic SRSs are already in place throughout some European countries and the USA. Likewise, other solutions, such as general practitioners' databases analysis, post market studies or prescription monitoring, among others, are being

thoroughly explored. Nevertheless, the majority of data is not publicly available for researchers, which, jointly with other barriers, severely limits signal detection [9, 10].

Although drug companies are required to track and manage adverse events reported by clinicians, lawyers or patients, the detection process relies mostly on the physician's ability to recognise a given trait as a drug adverse event. In addition to this underreporting, results are also biased due to selective reporting (reporting only certain drugs or conditions), placing the threshold of reported ADRs between 1-10% [11-13].

Whereas the problem for collecting and filtering ADR data from multiple distributed nodes has already been studied in the past [14], researchers continue to pursuit the best strategies to delve into the wealth of collected data in conjunction with other post drug administration inputs. With data and text-mining techniques scavenging millions of electronic medical records, pharmacovigilance researchers are now faced with the problem of delivering knowledge-oriented tools and services that exploit the scope of collected data. Ultimately, the adequate exploration of these data will pave the way for improved drug evaluations, critical for pharmaceutical companies, regulatory entities and researchers [15].

And herein lies the grand problem for contemporary pharmacovigilance: how to enable any researcher to assess and explore the wealth of collected data across a variety of algorithms and tools? In summary, researchers need new automated strategies to mechanistically understand the scientific evidence behind specific drug and adverse event interactions, through the processing of data mined from millions of electronic medical records and analysed independently by multiple algorithms. To overcome this problem, six key challenges arise for researchers and developers.

- **Scalability**. Controlling a flexible amount of algorithms, each providing independent access to knowledge, with its independent set of features and offering access to closed functionalities.
- **Interoperability**. The integration of multiple knowledge providers requires that a solution akin to a "common language" must be setup so that the various tools and algorithms can interact with each other and with a central software choreographer.
- **Management**. This brings two challengers: (1) how to store and make the collected data available to all researchers, and (2) how to organise and coordinate the set of available knowledge providers.
- **Reproducibility**. The replication of all research steps, including data and used knowledge providers must be available for other researchers and for further auditing.
- **Accessibility**. All the data and features must be presented in a unified workspace, publicly available to all interested stakeholders.
- **Security**. At last, interactions between knowledge providers, implemented software and researchers must be established through secure channels.

The strategy introduced in this manuscript, and its underlying architecture, implementation and prototype, successfully covers the aforementioned challenges, introducing a pioneering solution to deliver pharmacovigilance studies to researchers worldwide.

Materials and Methods

Background

Large-scale projects such as Research on Adverse Drug Events and Reports (RADAR) [16], Observational Medical Outcomes Partnership (OMOP) , Mini-Sentinel [17] or Exploring and Understanding Adverse Drug Reactions by Integrative Mining of Clinical Records and Biomedical Knowledge (EU-ADR) [18], among others, are pushing forward innovative strategies to improve active pharmacovigilance scenarios.

The RADAR project adopts a strategy where a highly specialized team reviews incoming drug adverse reports. Despite generating invaluable curated results, this approach implies a large waste of intensive manual labour. In opposition, OMOP, Mini-Sentinel and EU-ADR adopt more automated strategies.

OMOP and EU-ADR share a common setup, where data from partners are automatically collected, mined and analysed. Partners' data are translated to a common data model (CDM), anonymized, summarized and imported into a central integrative data repository for further statistical processing.

OMOP is applied to two distinct surveillance scenarios, tackling the identification of well-known drug associations and the identification of previously unknown signals [19]. This identification is validated through the application of multiple analytical procedures over a broad number of summarized patient records. In fact, OMOP's initial stage finished with an assessment of the best methods to identify pharmaceutical risk in healthcare data [20]. Selected algorithms are now being applied in the project's second stage.

EU-ADR distributed pipeline, discussed in detail in the following section, is very similar to OMOP's. The major difference resides on the partners' data. Whereas in OMOP most data stems from private contractors in the United States of America, in EU-ADR, data are obtained from European nationwide registries. Regarding the statistical analysis, EU-ADR's core longitudinal observation algorithms are LGPS and LEOPARD [21].

Despite featuring a distributed architecture, Mini-Sentinel is very different from the setup used in OMOP and EU-ADR. Whereas in the latter projects data are summarized and submitted for statistical analysis, in Mini-Sentinel data queries go through a complex network [22]. Like similar projects, partners in the Mini-Sentinel program translate their dataset to the project's CDM. However, in Mini-Sentinel, data never leaves the original institution [23]. This addresses FDA's concerns regarding security and privacy.

With Mini-Sentinel's strategy, data queries are "executable programs" that are sent to partners for in-premises execution [24]. Once queries are received, partners can analyse requested data, execute them and validate the results before reporting to the query authors. Mini-Sentinel's project coordination then assembles generated results and transfers summary data to the query authors. In spite of being more

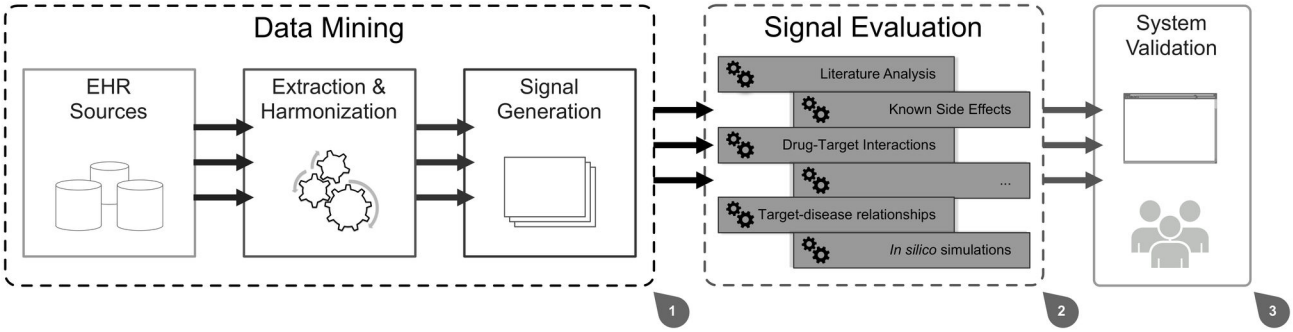

Figure 1. EU-ADR initiative data flow. 1) Data extracted from electronic health record (EHR) resources are semantically harmonized for data mining, generating a raw drug-event pair list. 2) The signal substantiation process analyses the submitted data, re-ranking the signal list, based on multiple algorithms. 3) Users trigger data analysis and exploration to validate the system operability.

doi: 10.1371/journal.pone.0083016.g001

secure, this strategy can lead to delays in query answering, specially considering the project's 7-day average response time.

With most of these projects in their infancy, a correct assessment of their results is a long-term task. Nevertheless, comparison frameworks are being put in place to better evaluate and compare the qualitative results of each project [25].

The EU-ADR Project

The foundation for EU-ADR's strategy relies on in-depth semantic data mining of electronic health records from several European countries. This process generates filtered data that can be easily substantiated through distributed computational tools [26] – Figure 1.

Project partners provide demographics, drug use and clinical data for over 20 million patients from several European countries. These data include clinical history, drug prescriptions, vaccinations, or lab test results [27, 28]. From a pharmacovigilance perspective and in a European or worldwide scale, mining the amount and type of data collected in these databases is of tremendous importance for an improved post-marketing drug evaluation [29].

Data mining techniques are used to extract the most relevant information from these data sources [30, 31], taking in account the privacy and ethical concerns regarding the collected data [32]. Next, data are harmonized into a unified dataset containing the list of drug-event pairs identified in the mined records [33]. This initial detection process generates a raw signal list, as the signal detection techniques highlight all possible relationships discovered in the wealth of collected data [34] – Figure 1-1.

As mentioned, the creation of these rich signal datasets is a well-established task. However, the actual data analysis and exploration tasks are still missing. Each signal in the raw list provided by the data mining tools must be substantiated for adequate validation. That is, the signal must be analysed by multiple algorithms to identify its real risk, and, if it exists, to provide a scientific explanation behind the cause, the drug, and the effect, a specific adverse reaction. This step, highlighted in Figure 1-2, differentiates the EU-ADR project pipeline from other related projects.

The signal substantiation algorithms can range from simple literature analysis, to drug target interaction matching. This is the key pharmacovigilance challenge to the EU-ADR project: how to design an architecture that can leverage on the data acquired from millions of electronic health records by enhancing its automated evaluation through any number and kind of distributed algorithms?

At last, results for these algorithms must be easily available for researchers [35] – Figure 1-3. The data flow ends at the researchers' workspace, where they can validate the system, explore the resulting scientific evidence and, if required, proceed to take the necessary steps to prevent new occurrences for the high-risk drug event interactions.

A Distributed Pharmacovigilance Platform

The architecture of a distributed platform in the context of pharmacovigilance must tackle the six mentioned challenges - scalability, interoperability, management, reproducibility, accessibility and security. The proposed service-oriented architecture is shown in Figure 2 and its components described in Table 1.

This architecture is built on top of multiple interactions, exploiting the components' dynamics. Results from the semantic harmonization of mined records, the raw signal list, are securely stored on the knowledge base, being easily accessible to all the other components. Once the users select the knowledge providers from the provider registry, the data are transmitted to the service execution engine, which then contacts each of the knowledge providers for service execution. The outputs of the analysis algorithms are next stored in the platform's knowledge base, and delivered to the researchers through the web application engine. All these interactions are controlled and securely mediated by the application engine.

Scalability & Interoperability. The provider registry and the service execution engine ensure scalability and interoperability.

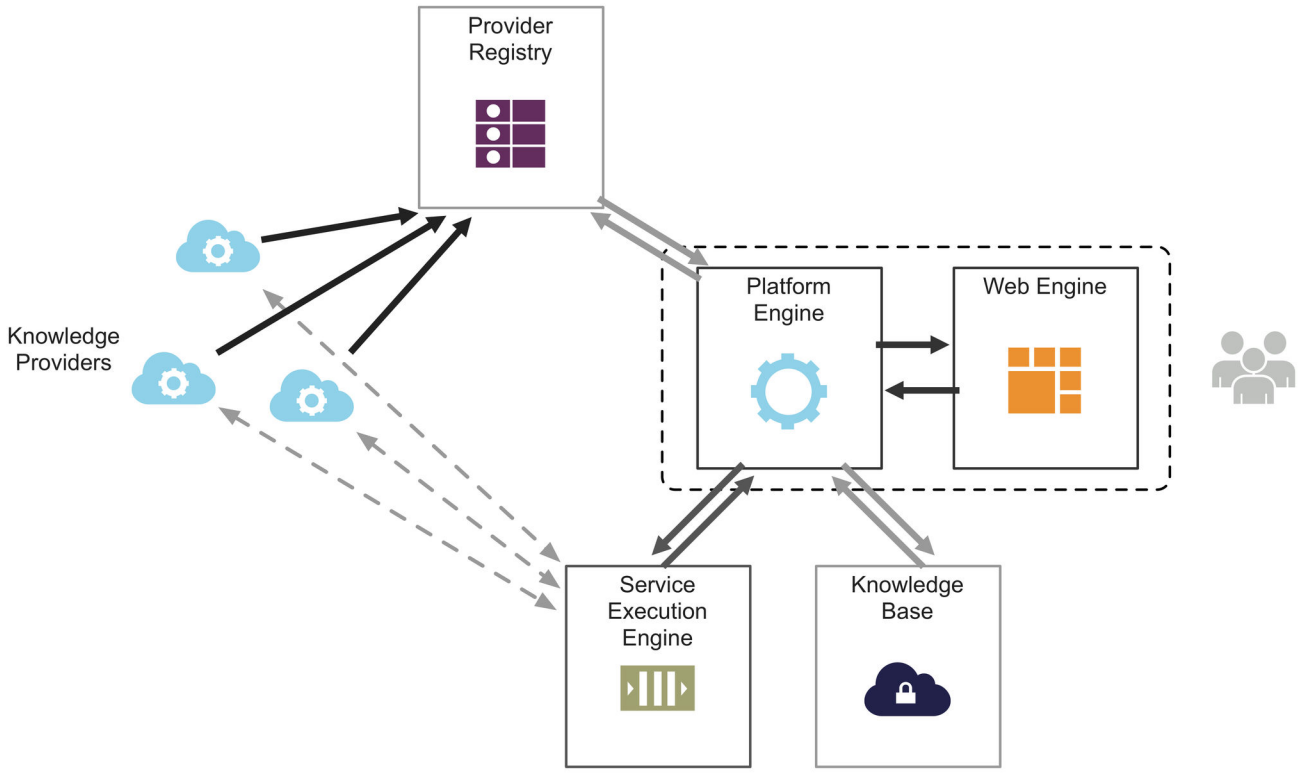

Figure 2. General architecture for the distributed pharmacovigilance platform.
doi: 10.1371/journal.pone.0083016.g002

Table 1. Architecture component descriptions, implementation and operability purpose.

Component	Operability	Implementation	Description
Service execution engine	External	Java Taverna	With each knowledge provider delivering service-based access to its algorithm, the service execution engine is responsible for performing the service calls with the input data read from the knowledge base and retrieving the output data towards the platform.
Knowledge base	Internal	Cloud-based	The knowledge base stores all relevant data from the integrated and imported pharmacovigilance datasets. Data are stored in a cloud-based environment, moving the inherent complexities associated with secure data storage to en efficient cloud provider.
Provider registry	Internal	Java	The provider registry acts as the main knowledge provider controller. This is where new knowledge providers must register their interfaces and endpoints so that they can be made available for future use.
Knowledge Providers	External	Independent XML-based standard	The knowledge providers deliver independent access to various pharmacovigilance data analysis and exploration algorithms. Access to knowledge providers is service-based.
Platform engine	Internal	Java	The platform engine is the architecture core component, where all the tasks are executed and the interactions controlled.
Web engine	Internal	Google Web Toolkit	The web engine powers the distributed platform user interactions through an innovative web-based workspace.

doi: 10.1371/journal.pone.0083016.t001

To completely remove any interdependence amongst knowledge providers, a common data description and exchange language was created. Additionally, to foster the seamless integration of knowledge providers, data input and output formats must be compatible.

A new XML schema (XSD) was designed to accomplish this. The flexibility of XSD enables the validation of both content and structure, preventing erroneous data transfers and reinforcing the overall platform robustness. The schema structure is divided in three main sections, each focusing on one operability perspective.

• **Monitoring**. Real-world use of knowledge providers can result in an assorted amount of errors: general communication errors, such as failure to connect to a database, or domain-specific errors, such as invalid data. Therefore, the created

schema covers the domain-specific errors with a set of "status codes" for each of the possible conditions. For example, status code with the value "41" identifies an internal service problem regarding the database connection.

• **Scoring**. Each of the signals in the ranked list has a score that determines its relevant risk within the dataset. When the data are being assessed by the knowledge providers, the scoring attributes will provide each evaluated signal with a numeric value, between 0 (zero) and 1 (one), measuring the relative relevance and impact according to the scientific evidence found to explain the interactions of a given drug-event pair.

• **Annotation**. When a scientific explanation is found for a given set of drug-event pairs, the output is annotated with reliable evidence for the interaction, providing researchers with valuable knowledge and allowing them to evaluate the signal, share the results and reproduce their research in the future. These annotations appear in the form of connections to relevant resources, such as literature (PubMed links), proteins (UniProt links), chemical compounds (SMILE codes) or pathways (Reactome links), among others.

The schema is available online (http://bioinformatics.ua.pt/euadr/euadr_types.xsd), enabling anyone to create and add new algorithms to this plugin-based distributed platform, thus becoming one of the project's knowledge providers.

Knowledge Providers. Interoperability amongst various knowledge providers required the design of a strategy to explore the true value of the created data exchange standard. Whilst the schema is an essential component of the distributed platform interoperability features, it is useless by itself: the execution of knowledge providers' algorithms must be intermediated by a distinct component, the service execution engine.

Another drawback regarding the implementation of knowledge providers relates to their internal algorithms. Whereas in some cases the algorithms are relatively straightforward, in the majority of scenarios the algorithms require multiple service-service interactions and data processing tasks.

This added another complexity layer to our architecture: the knowledge providers required heterogeneous interactions within their algorithms, a challenge that could not be tackled at the distributed platform level. Hence, the use of scientific workflows arises as a solution [36]. A crucial workflow requirement is that the inputs of each activity must match the precedent activity outputs to maintain consistency, a feature already accomplished with the platform interoperability standard. Dealing with workflow execution operations requires the implementation of workflow management applications, whose goal is to abstract the programming side of the application, assisting in the creation of workflows without writing a single line of code [37].

Taverna emerged as the *de facto* standard for desktop-based workflow management in the life sciences [38]. Taverna's success is due to its flexibility, which allows researchers to create complex workflow-based algorithms just by dragging and dropping boxes in its workbench. Alternatives to Taverna, such as Galaxy [39] or BioFlow [40], are focused on providing workflow management functionality in a web-based interface. However, this was not a requirement for our scenario and, at the time of development, these tools do not offer an API as advanced as Taverna's.

With Taverna in place, the provider registry collects metadata for Taverna workflows, and contains algorithms that can be downloaded for local use or executed online in the distributed platform. In addition to maintaining a list of available workflows, Taverna's integration also required the implementation of a service execution engine. This solution allows the combination of comprehensive data analysis and exploration algorithms within the distributed platform. In summary, we need to feed the workflows with XML input data, execute them and extract the resulting data from the XML output. Figure 3 illustrates the steps required to execute knowledge providers' workflows.

The service execution engine is a Java tool built to execute Taverna's command line interface with custom input arguments. These parameterized system calls run in their own independent OS process, increasing the overall platform performance and scalability. Workflow executions are also a background non-blocking asynchronous process. For researchers, this means that they can use all the application features whilst the workflows are being executed in the background.

Knowledge Management. The adequate management of scientific data is critical to the success of the proposed distributed pharmacovigilance platform. Not only we need to consider how to make all relevant knowledge accessible at all times, we also need to implement adequate data sharing features: data must be exchanged between knowledge providers and collaboration is one of the underlying premises for research reproducibility.

The knowledge base is stored on a cloud environment [41]. This means that while the underlying data storage layer is distributed through multiple independent data nodes, the access is unified and centralized through a single access point. Common data storage issues such as persistence, security and access are controlled by the cloud-based layer, leaving the relevant data handling tasks to the platform engine [42].

In the EU-ADR project context, five key datasets are stored, detailed next.

• **Drugs**. Dataset containing the complete list of ATC codes and respective drug names.

• **Adverse events**. Dataset listing the adverse events mined from the project's pharmacovigilance data.

• **Imported data**. Researcher-submitted datasets containing statistical data regarding specific drug-event mapping conditions.

• **Results**. Datasets with the results from the knowledge providers' algorithms.

• **Users**. Dataset containing the user details and sharing/collaboration preferences.

Collaboration features are implemented according to two distinct methods: project-based and *ad hoc* sharing. With the project-based collaboration option, new *projects* with any number of users can be configured. This allows a broad

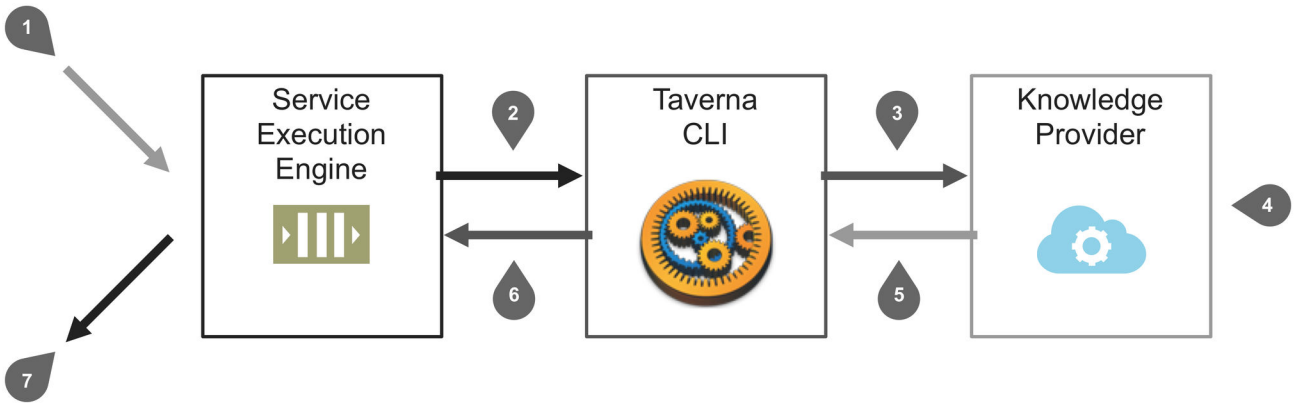

Figure 3. From user input to system output the platform engine controls the execution of workflows as follows:. 1) One or more knowledge provider algorithms are selected to evaluate researcher-submitted datasets. The platform engine sends the request to the service execution engine. 2) An XML file with the input data (obeying the platform's interoperability standard) is generated and its path provided to the service execution engine, along with the path for the workflows associated with each knowledge provider algorithm. The workflow execution is then triggered by a system call. 3) The Taverna command line tool loads the knowledge provider's workflow, starting the processing tasks. 4) The knowledge provider execution proceeds internally, executing the miscellaneous workflow tasks. 5) The workflow delivers an XML data file (obeying the platform's interoperability standard) with the algorithm output. 6) The service execution engine loads the XML output file and transfers the results to the platform engine. 7) The engine stores the data in the knowledge base and makes it promptly available for delivery in the web workspace.

doi: 10.1371/journal.pone.0083016.g003

number of users to manage a topic-specific dataset, fostering a deeper collaboration amongst researchers through the sharing of submitted data and obtained results.

Ad hoc collaboration is a user-specific approach. Users can share their datasets and results to any other user in the platform through their registration email. This is a more granular approach, where the users can define what data to share and what their collaborators can view or change.

Accessibility. Accessing knowledge and executing available features are key challenges behind pharmacovigilance software [43]: managed data and knowledge providers' algorithms must be accessible at all times. To accomplish this, the architecture relies on two advanced components: the platform engine and the web engine. The former is the main application controller, coordinating all the others components. The latter manages the presentation layer, providing access to a web-based workspace. The implementation of both is detailed in Figure 4.

Taking into account the accessibility and interoperability requirements, the platform engine is implemented as a Java web application. For improved data handling, Hibernate (http://www.hibernate.org/) was used as a data abstraction layer and object/relational mapper, thus reducing database coupling with the application. This shields the development from future changes in the domain model storage system and eases the use within the Java object-oriented environment.

Additional components were also used, such as Spring Security (http://static.springsource.org/spring-security/site/) for improved security features, Apache POI (http://poi.apache.org/) for enhanced data import and export, Log4j for logging purposes and Apache Maven (http://maven.apache.org/) for project dependency management, building and deployment.

The platform engine mediates the interactions within the distributed ecosystem. It controls the entire architecture and its data flows, moving the data from the knowledge base towards the service execution engine, establishing secure connections in all transactions, and regulating the provider registry system, among others. In a sense, the platform engine is an intelligent proxy, coordinating everything that happens with the distributed platform internal components.

The web application engine adopts a Model-View-Presenter pattern and is implemented with the Google Web Toolkit (GWT) framework (https://developers.google.com/web-toolkit/). GWT compiles Java code to a browser-targeted JavaScript representation, resulting in an extremely effective web application. In addition, various user interaction components were added to provide a cleaner perspective over the huge datasets and easy access to data analysis and exploration features. To improve on GWT's user interactions library, the Ext GWT package (GXT) (http://www.sencha.com/products/gxt) was used. This extends the widgets bundled with GWT core distribution to provide a more complete set of user interaction features required by the presentation layer. The combination of GWT's basic widgets with GXT ones was further improved with Google Gin (http://code.google.com/p/google-gin/) for dependency injection, achieving a decoupled architecture.

Security. Security is a primary concern for any new software, especially considering the rigorous constraints of this field, for both researchers and private pharmaceutical companies [44, 45]. In the proposed architecture, security

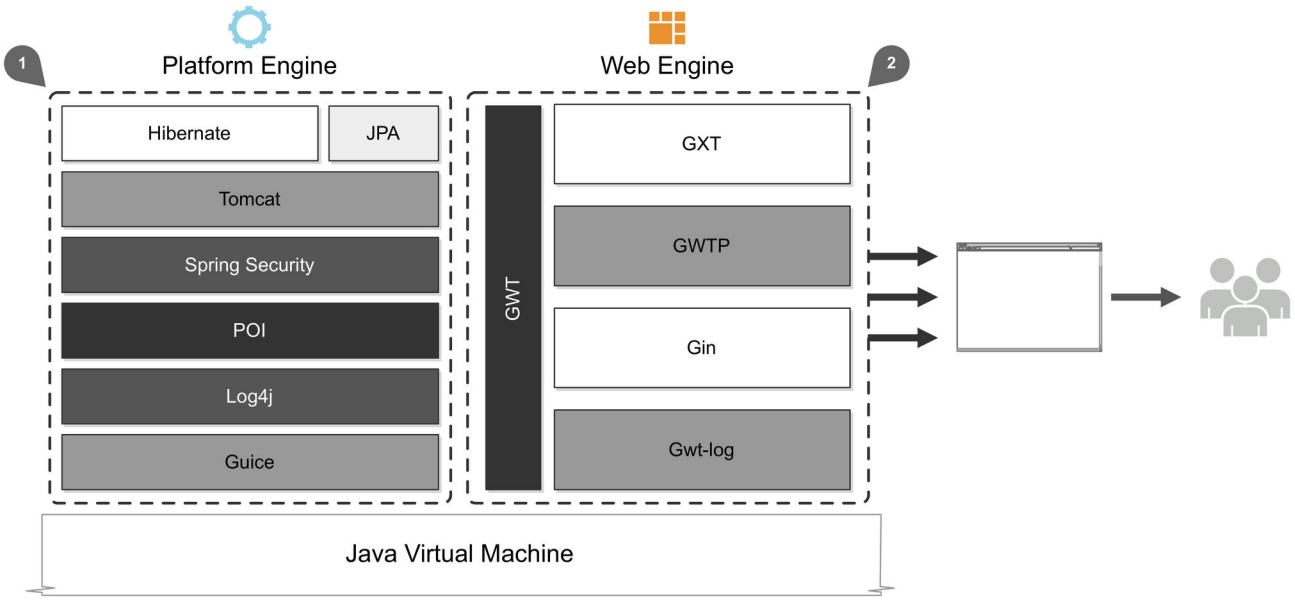

Figure 4. Internal platform implementation software overview. Used components are implemented in Java, leveraging the open-source nature of this solution. 1) Platform engine components include Hibernate, JPA, Spring Security, POI, Log4j, Guice and custom code to control the application and serve it as a Tomcat web application. 2) The Web engine relies on Google Web Toolkit to generate a highly responsive web workspace. Add-ons such as GXT and Gin were used to improve the user interactions' performance and reliability.

doi: 10.1371/journal.pone.0083016.g004

measures are applied at three levels: interactions, data and users.

All interactions within the distributed platform go through a secure HTTP channel. Knowledge providers' services must be deployed in a HTTPS endpoint and the use of valid certificates is enforced. However, this measure only secures the execution of workflows within the service execution engine. Hence, in addition to securing the knowledge providers, the implemented architecture also delivers secure access to the web workspace, which is served through and HTTPS channel.

At the data level, managed data are securely stored in the knowledge base using encryption measures to obfuscate the actual content. For instance, data owner details are used to "salt" the data, further bolstering its illegibility. By using a cloud-based approach, the logical storage is decoupled from the physical storage, further improving the overall security. Moreover, security and privacy features are delegated to the cloud-based controller. Likewise, partner data introduced in the system has already been carefully anonymized.

At last, on a user perspective, collaboration preferences can be tightly controlled. The distributed platform collaboration features facilitate granular data sharing, so that researchers know whom they are sharing data with and how those data are being used. Considering the sensible nature of the majority of stored data, access to the web workspace requires registration and all actions are tracked, assisting the monitoring of everything that happens within the system.

Results

The pharmacovigilance context opens various opportunities to build new data analysis and exploration ecosystems. With collaboration from partners within the EU-ADR project it was possible to implement a prototype of the proposed distributed platform. The involvement in a large-scale European project allowed for the implementation of real-world algorithms by multiple knowledge providers. This partnership brought access to a comprehensive dataset of drugs, events and statistical data. The resulting EU-ADR Web Platform, is available online at http://bioinformatics.ua.pt/euadr/. This portal is being used within and beyond the EU-ADR project scope, generating successful research results in various areas, such as the identification of drug agents causing acute myocardial infarction [46].

Pharmacovigilance Algorithms

The initial service-oriented architecture implementation includes four knowledge providers, each with its own pharmacovigilance algorithm and made available as a Taverna workflow using secure services. The algorithms are deployed independently in distinct physical and logical settings. The first three algorithms provide a score, between 0 and 1, for each input signal, marking whether or not there is scientific evidence behind the drug-event pair.

The first algorithm, literature analysis, adopts a semantics-based approach [47] that processes Medline annotations looking for particular MeSH terms and metadata related to the

submitted drug-event pair. Using the MeSH thesaurus, matches for the subheadings “chemically induced” and “adverse effects” are searched in associated publications. The “Pharmacological Action” knowledge from MeSH thesaurus is also used to refine the search.

When no matches are found, the partial scoring for the given drug-event pair is 0 (zero). In the opposite, with 3 or more publications found, the signal is scored with 1 (one). Between 0 and 3 (exclusive) publications, the partial score will be of 0.5. Positive scores imply that scientific literature has been published on the association between the drug and the event. In these cases, the knowledge provider annotates the output with PubMed ID links of the discovered publications.

The second strategy, involves a signal filtering co-occurrence process, evaluating the relationships between drugs and side effects that might have been reported previously in Medline literature, DailyMed [48] or DrugBank [49]. Data from these resources are previously indexed, including titles and abstracts from Medline, summary product characteristics from DailyMed, and ATC codes with potential adverse events from DrugBank. The algorithm then performs a chi-square test to determine if the co-occurrence of the given drug-event pair is different than what would be expected by chance.

Similarly to the first algorithm, when interactions are found in the indexed knowledge base, the signal gets a scoring of 1 (one). The annotation section of the output will include identifiers and connections to the relevant resources (Medline, DailyMed or DrugBank).

The third algorithm, signal substantiation, generates a network based on the drug-event pair containing the interactions with proteins targeted by the drug and associated events, and with biological pathways [50]. This results in drug-target and event-target profiles that are searched for common sets of proteins, the intersecting portion of the graph.

The output of this algorithm, a comprehensive list of proteins and pathways related to the drug-event pair, is annotated to the knowledge provider output along with the partial signal classification score.

Once the data are processed through these algorithms, the results must be combined to better assess the plausibility of a given drug-event relationship. The fourth algorithm, evidence combination, uses the scores from the other knowledge providers to arrive at a degree of belief that takes available evidence into account. The algorithm uses the Dempster–Shafer theory [51] to evaluate the initial data combined with algorithm results to reach a measurable belief level that a particular drug-event pair has a low, medium or high risk. Algorithms weight and relevance in the final measurement can be customized to better fit the research context. This final risk measurement is the most important outcome of the performed pharmacovigilance research as it summarizes the relative risk for each drug-event pair in context of available knowledge.

These algorithms have been deployed independently by EU-ADR project partners, which reinforce the proposed platform suitability to environments requiring software interoperability.

Web Workspace

EU-ADR Web Platform’s key feature is the execution of advanced post-marketing adverse drug reaction studies. Researchers upload and investigate drug-event datasets, create targeted drug studies and work with their peers through the available collaboration features. Each researcher has its own personal workspace, where they can browse existing datasets (personal or shared); upload custom drug-event pair datasets; or create drug-specific datasets, based on the overall platform data.

A researcher interested in studying potential adverse reactions of patients treated with a given drug, *XYZ* for the purpose of this discussion, begins its study by automatically generating a dataset focused on the targeted drug. The system then combines this drug with the 11 potential adverse events considered in EU-ADR’s context, evaluates the resulting dataset using the available knowledge providers and combines all individual pieces of evidence into an aggregate score representing the predicted risk of each drug-event relationship – Figure 5. Signals classified as moderately or highly risky should be further investigated by analysing presented evidence and following hyperlinks to biomedical literature, as well as to external drug and biological data resources.

Conclusions

Despite the thorough research and development standards, post-market pharmacovigilance plays a key role in the assessment of existing medicines and creation of new drugs. Nevertheless, research over the last decades has focused on identifying and measuring specific adverse drug reactions in a post-marketing stage [52-54]. The holistic assessment of widespread electronic medical records empowers valuable insights over adverse drug events. Notwithstanding the value of these data *per se*, the development of new strategies to fully exploit the scientific background regarding reported events is vital.

This manuscript details the creation of such strategy, proposing a pharmacovigilance-focused distributed platform and introducing an open framework for the better exploration of the wealth of available pharmacovigilance data by all pharmacogenomics stakeholders. The EU-ADR Web Platform is a unique tool that allows researchers to exploit the wealth of data from a European cohort, combined with independent drug-event datasets. In addition to being a step forward relative to existing solutions [55], the designed strategy accurately tackles multiple challenges behind the development of state-of-the-art software within the pharmacovigilance domain: scalability, interoperability, management, reproducibility, accessibility and security.

- The plugin-based provider registry ensures that the platform is scalable. Where the standard defines the knowledge providers’ interfaces, the provider registry stores metadata regarding the available algorithms, making them available as workflows for local or remote execution.
- A new interoperability language was developed to ensure that all knowledge providers understand the data being

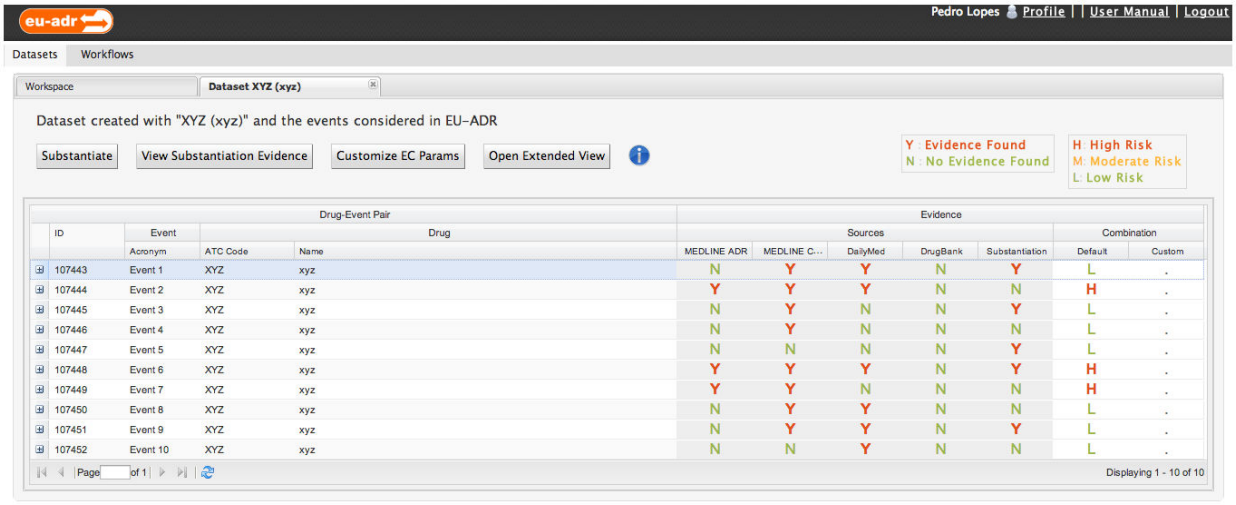

Figure 5. EU-ADR Web Platform workspace interface for an undisclosed drug (*XYZ*) exploration scenario containing the signal list that results from distributed knowledge provider algorithm outputs and evidence combination statistical analysis. Workflow results are labelled with Y in case sufficient evidence is found to support a potential drug-event relationship, or N otherwise. Evidence combination yields a score of H, M or L, indicating High, Moderate or Low risk respectively, of a drug-event relationship being in fact an ADR signal.

doi: 10.1371/journal.pone.0083016.g005

exchanged, enabling accurate interactions within the distributed platform ecosystem.

• With knowledge providers managed through provider registry, collected data are stored in a cloud environment, streamlining the associated knowledge management tasks [56].

• This proposal enables research reproducibility through the collection of multiple datasets, which include easily reproducible analysis results. This step is further improved through the use of a cloud-based knowledge base, storing all gathered and submitted data, and ensuring availability, reliability and an eased access for all the architecture components.

• The platform's data analysis and exploration features are accessible through a web interface, constantly available to every researcher in any kind of system or device.

• This new architecture enforces the establishment of secure communication channels amongst the platform and the knowledge providers, the security of datasets and the restricted web-based workspace.

A prototype implementation of this strategy is in place in the context of the European EU-ADR project, extending the interoperability amongst project partners. The EU-ADR Web Platform connects distributed knowledge analysis algorithms, and is available online for public use at http://bioinformatics.ua.pt/euadr/.

Acknowledgements

We wish to thank all the members of the EU-ADR project.

Author Contributions

Conceived and designed the experiments: JLO JvL CD. Performed the experiments: PL TN DC LF ABM FS MC JM JK BS EvM GD PA EA SB. Wrote the manuscript: PL JLO.

References

1. McClure DL (2009) Improving Drug Safety: Active Surveillance Systems Should be Paramount. Pharmaceutical Medicine 23: 127-130. doi:10.1007/BF03256760.
2. Shibata A, Hauben M (2011) Pharmacovigilance, signal detection and signal intelligence overview. Proceedings of the 14th International Conference on Information Fusion (FUSION). Chicago, IL. pp. 1-7.
3. Ema Ema (2010) Annual Report. European Medicines Agency.
4. Robb MA, Racoosin JA, Sherman RE, Gross TP, Ball R et al. (2012) The US Food and Drug Administration's Sentinel Initiative: Expanding the horizons of medical product safety. Pharmacoepidemiol Drug Saf 21: 9-11. doi:10.1002/pds.2311. PubMed: 22262587.
5. Xu L, Anchordoquy T (2011) Drug delivery trends in clinical trials and translational medicine: Challenges and opportunities in the delivery of nucleic acid-based therapeutics. J Pharm Sci 100: 38-52. doi:10.1002/jps.22243. PubMed: 20575003.
6. Nelson JC, Cook AJ, Yu O, Dominguez C, Zhao S et al. (2012) Challenges in the design and analysis of sequentially monitored postmarket safety surveillance evaluations using electronic observational health care data. Pharmacoepidemiol Drug Saf 21: 62-71. doi:10.1002/pds.2324. PubMed: 22262594.
7. Staffa JA, Dal Pan GJ (2012) Regulatory Innovation in Postmarketing Risk Assessment and Management. Clin Pharmacol Ther 91: 555-557. doi:10.1038/clpt.2011.289. PubMed: 22297386.
8. Stahl M, Edwards IR, Bowring G, Kiuru A, Lindquist M (2003) Assessing the Impact of Drug Safety Signals from the WHO Database Presented inSIGNAL': Results from a Questionnaire of National Pharmacovigilance Centres. Drug safety 26: 721-727
9. Lazarou J, Pomeranz BH, Corey PN (1998) Incidence of adverse drug reactions in hospitalized patients: a meta-analysis of prospective

studies. JAMA 279: 1200-1205. doi:10.1001/jama.279.15.1200. PubMed: 9555760.
10. Meyboom RHB, Lindquist M, Egberts ACG, Edwards IR (2002) Signal Selection and Follow-Up in Pharmacovigilance. Drug Safety 25: 459-465. doi:10.2165/00002018-200225060-00011. PubMed: 12071784.
11. Alvarez-Requejo A, Carvajal A, Bégaud B, Moride Y, Vega T et al. (1998) Under-reporting of adverse drug reactions Estimate based on a spontaneous reporting scheme and a sentinel system. Eur J Clin Pharmacol 54: 483-488. doi:10.1007/s002280050498. PubMed: 9776440.
12. Grootheest V (1999) Attitudinal survey of voluntary reporting of adverse drug reactions. Br J Clin Pharmacol 48: 623-627. PubMed: 10583035.
13. De Bruin ML, Van Puijenbroek EP, Egberts AC, Hoes AW, Leufkens HG (2002) Non-sedating antihistamine drugs and cardiac arrhythmias–biased risk estimates from spontaneous reporting systems? Br J Clin Pharmacol 53: 370-374. doi:10.1046/j.1365-2125.2002.01569.x. PubMed: 11966667.
14. Coloma PM, Schuemie MJ, Trifirò G, Gini R, Herings R et al. (2011) Combining electronic healthcare databases in Europe to allow for large-scale drug safety monitoring: the EU-ADR Project. Pharmacoepidemiol Drug Saf 20: 1-11. doi:10.1002/pds.2206. PubMed: 21182150.
15. Härmark L, Van Grootheest AC (2008) Pharmacovigilance: methods, recent developments and future perspectives. Eur J Clin Pharmacol 64: 743-752. doi:10.1007/s00228-008-0475-9. PubMed: 18523760.
16. Trontell AE (2005) The RADAR Project and the FDA. JAMA 294: 1206. doi:10.1001/jama.294.10.1206-a. PubMed: 16160128.
17. Platt R, Carnahan R (2012) The US Food and Drug Administration's Mini-Sentinel Program. Pharmacoepidemiology and Drug Safety 21: 1-303. doi:10.1002/pds.3324.
18. Trifiro G, Fourrier-Reglat A, Sturkenboom MCJM, Díaz Acedo C, Van Der Lei J et al. (2009) The EU-ADR project: preliminary results and perspective. Stud Health Technol Inform 148: 43-49. PubMed: 19745234.
19. Stang PE, Ryan PB, Racoosin JA, Overhage JM, Hartzema AG et al. (2010) Advancing the Science for Active Surveillance: Rationale and Design for the Observational Medical Outcomes Partnership. Ann Intern Med 153: 600-606. doi: 10.7326/0003-4819-153-9-201011020-00010. PubMed: 21041580.
20. Ryan PB, Madigan D, Stang PE, Marc Overhage J, Racoosin JA et al. (2012) Empirical assessment of methods for risk identification in healthcare data: results from the experiments of the Observational Medical Outcomes Partnership. Stat Med 31: 4401-4415. doi:10.1002/sim.5620. PubMed: 23015364.
21. Schuemie MJ (2011) Methods for drug safety signal detection in longitudinal observational databases: LGPS and LEOPARD. Pharmacoepidemiol Drug Saf 20: 292-299. doi:10.1002/pds.2051. PubMed: 20945505.
22. Platt R, Wilson M, Chan KA, Benner JS, Marchibroda J et al. (2009) The New Sentinel Network — Improving the Evidence of Medical-Product Safety. N Engl J Med 361: 645-647. doi:10.1056/NEJMp0905338. PubMed: 19635947.
23. Behrman RE, Benner JS, Brown JS, McClellan M, Woodcock J et al. (2011) Developing the Sentinel System — A National Resource for Evidence. Development - New England Journal of Medicine 364: 498-499. doi:10.1056/NEJMp1014427.
24. Curtis LH, Weiner MG, Boudreau DM, Cooper WO, Daniel GW et al. (2012) Design considerations, architecture, and use of the Mini-Sentinel distributed data system. Pharmacoepidemiol Drug Saf 21: 23-31. doi:10.1002/pds.2336. PubMed: 22262590.
25. Gini R, Coppola M, Ryan PB, Righetti G, Peri I et al. (2013) Frameworks for Data Extraction and Management from Electronic Healthcare Databases for Multi-Center Epidemiologic Studies: a Comparison among EU-ADR, MATRICE, and OMOP Strategies. 29th International Conference on Pharmacoepidemiology and Therapeutic Risk Management. Montréal. Canada.
26. Trifirò G, Patadia V, Schuemie MJ, Coloma PM, Gini R et al. (2011) EU-ADR healthcare database network vs. spontaneous reporting system database: preliminary comparison of signal detection. Stud Health Technol Inform 166: 25-30. PubMed: 21685607.
27. Wilke RA, Xu H, Denny JC, Roden DM, Krauss RM et al. (2011) The Emerging Role of Electronic Medical Records in Pharmacogenomics. Clin Pharmacol Ther 89: 379-386. doi:10.1038/clpt.2010.260. PubMed: 21248726.
28. Coloma PM, Trifirò G, Schuemie MJ, Gini R, Herings R et al. (2012) Electronic healthcare databases for active drug safety surveillance: is there enough leverage? Pharmacoepidemiol Drug Saf 21: 611-621. doi: 10.1002/pds.3197. PubMed: 22315152.
29. Chan KA, Hauben M (2005) Signal detection in pharmacovigilance: empirical evaluation of data mining tools. Pharmacoepidemiol Drug Saf 14: 597-599. doi:10.1002/pds.1128. PubMed: 16134080.
30. Wilson AM, Thabane L, Holbrook A (2004) Application of data mining techniques in pharmacovigilance. Br J Clin Pharmacol 57: 127-134. PubMed: 14748811.
31. Koh HC, Tan G (2005) Data mining applications in healthcare. J Healthc Inf Manag 19: 64-72. PubMed: 15869215.
32. Fu Y, Chen Z, Koru G, Gangopadhyay A (2010) A privacy protection technique for publishing data mining models and research data. ACM Trans Manage Inf Syst 1: 1-20.
33. Avillach P, Coloma PM, Gini R, Schuemie M, Mougin F, et al. (2012) Harmonization process for the identification of medical events in eight European healthcare databases: the experience from the EU-ADR project. Journal of the American Medical Informatics Association.
34. Trifirò G, Pariente A, Coloma PM, Kors JA, Polimeni G et al. (2009) Data mining on electronic health record databases for signal detection in pharmacovigilance: which events to monitor? Pharmacoepidemiol Drug Saf 18: 1176-1184. doi:10.1002/pds.1836. PubMed: 19757412.
35. Oliveira JL, Lopes P, Nunes T, Campos D, Boyer S, et al. (2012) The EU-ADR Web Platform: delivering advanced pharmacovigilance tools. Pharmacoepidemiology and drug safety.
36. Hollingsworth D (1995) The Workflow Reference Model.
37. Petkov S, Oren E, Haller A (2005) Aspects in Workflow Management. Galway, Ireland: DERI, Digital Enterprise Research Institute. p. 20.
38. Ludascher B, Altintas I, Berkley C, Higgings D, Jaeger E et al. (2006) Taverna: Scientific Workflow Management and the Kepler System. Research Articles. Concurrency-Computation: Practice and Experience 18: 1039 - 1065. doi:10.1002/cpe.994.
39. Goecks J, Nekrutenko A, Taylor J (2010) Galaxy: a comprehensive approach for supporting accessible, reproducible, and transparent computational research in the life sciences. Genome Biol 11: R86. doi: 10.1186/gb-2010-11-8-r86. PubMed: 20738864.
40. Jamil H, El-Hajj-Diab B (2008) BioFlow: A Web-Based Declarative Workflow Language for Life Sciences. IEEE Congress on Services. Honolulu, HI, USA IEEE Computer Society. pp. 453 - 460.
41. Lopes P, Oliveira JL (2012) COEUS:"semantic web in a box" for biomedical applications. Journal of biomedical semantics 3: 1-19.
42. Shucheng Y, Cong W, Kui R, Wenjing L (2010) Achieving Secure, Scalable, and Fine-grained Data Access Control in Cloud Computing. Proc. IEEE INFOCOM. San Diego, CA, USA. pp. 1-9.
43. Cheung K-H, Yip KY, Townsend JP, Scotch M (2008) HCLS 2.0/3.0: Health care and life sciences data mashup using Web 2.0/3.0. Journal of Biomedical Informatics 41: 694-705.
44. Rui Z, Ling L (2010) Security Models and Requirements for Healthcare Application Clouds. Cloud Computing (CLOUD), 2010 IEEE 3rd International Conference on. pp. 268-275.
45. Baker A, Vega L, DeHart T, Harrison S (2011) Healthcare and Security: Understanding and Evaluating the Risks. In: Robertson, M., editor editor. Ergonomics and Health Aspects of Work with Computers. Berlin: Springer Berlin / Heidelberg. pp. 99-108
46. Coloma PM, Schuemie MJ, Trifirò G, Furlong L, van Mulligen E et al. (2013) Drug-Induced Acute Myocardial Infarction: Identifying 'Prime Suspects' from Electronic Healthcare Records-Based Surveillance. System - PLOS ONE 8: e72148. doi:10.1371/journal.pone.0072148.
47. Avillach P, Mougin F, Joubert M, Thiessard F, Pariente A et al. (2009) A semantic approach for the homogeneous identification of events in eight patient databases: a contribution to the European eu-ADR project. Stud Health Technol Inform 150: 190-194. PubMed: 19745295.
48. de Leon J (2011) Highlights of Drug Package Inserts and the Website DailyMed: The Need for Further Improvement in Package Inserts to Help Busy Prescribers. J Clin Psychopharmacol.
49. Wishart DS, Knox C, Guo AC, Cheng D, Shrivastava S et al. (2008) DrugBank: a knowledgebase for drugs, drug actions and drug targets. Nucleic Acids Res 36: D901-D906. PubMed: 18048412.
50. Bauher-Mehren A, Mulligen Ev Avillach P, Carrascosa M, Singh B et al. (2012) Automatic filtering and substantiation of drug safety signals. PLOS Computational Biololgy 8.
51. Zadeh LA (1986) A simple view of the Dempster-Shafer theory of evidence and its implication for the rule of combination. AI Magazine 7: 85.
52. Vilar S, Harpaz R, Chase HS, Costanzi S, Rabadan R, et al. (2011) Facilitating adverse drug event detection in pharmacovigilance databases using molecular structure similarity: application to rhabdomyolysis. Journal of the American Medical Informatics Association.
53. Papay J, Yuen N, Powell G, Mockenhaupt M, Bogenrieder T (2012) Spontaneous adverse event reports of Stevens–Johnson syndrome/toxic epidermal necrolysis: detecting associations with medications.

Pharmacoepidemiol Drug Saf 21: 289-296. doi:10.1002/pds.2276. PubMed: 22139991.
54. Sommet A, Durrieu G, Lapeyre-Mestre M, Montastruc J-L, Association of French PharmacoVigilance Centres (2012) A comparative study of adverse drug reactions during two heat waves that occurred in France in 2003 and 2006. Pharmacoepidemiology and Drug Safety 21: 285-288.
55. Wang K, Bai X, Li J, Ding C (2010) A service-based framework for pharmacogenomics data integration. Enterprise - Information Systems 4: 225-245. doi:10.1080/17517575.2010.498525.
56. Dudley JT, Butte AJ (2010) In silico research in the era of cloud computing. Nat Biotechnol 28: 1181-1185. doi:10.1038/nbt1110-1181. PubMed: 21057489.