German Medical Data Sciences 2023 — Science. Close to People.
R. Röhrig et al. (Eds.)

doi:10.3233/SHTI230705

A Tool for Specifying Data Quality Checks for Clinical Data Management Systems – A Technical Case Report

Florian ULBRICH[a], Frank A. MEINEKE[a], Florian RISSNER[b], Alfred WINTER[a], and Matthias LÖBE[a,1]
[a] *Institute for Medical Informatics Statistics, Epidemiology (IMISE) University of Leipzig*
[b] *Center for Clinical Studies, Jena University Hospital*

Abstract. Introduction: Prospective data collection in clinical trials is considered the gold standard of clinical research. Validating data entered in input fields in case report forms is unavoidable to maintain good data quality. Data quality checks include both the conformance of individual inputs to the specification of the data element, the detection of missing values, and the plausibility of the values entered. **State-of-the-Art**: Besides Libre-/OpenClinica there are many applications for capturing clinical data. While most of them have a commercial approach, free and open-source solutions lack intuitive operation. **Concept**: Our ocRuleTool is made for the specific use case to write validation rules for Open-/LibreClinica, a clinical study management software for designing case report forms and managing medical data in clinical trials. It addresses parts of all three categories of data quality checks mentioned above. **Implementation**: The required rules and error messages are entered in the normative Excel specification and then converted to an XML document which can be uploaded to Open-/LibreClinica. The advantage of this intermediate step is a better readability as the complex XML elements are broken down into easy to fill out columns in Excel. The tool then generates the ready to use XML file by itself. **Lessons Learned**: This approach saves time, is less error-prone and allows collaboration with clinicians on improving data quality. **Conclusion**: Our ocRuleTool has proven useful in over a dozen studies. We hope to increase the user base by releasing it to open source on GitHub.

Keywords. Data Quality, EDC, LibreClinica, OpenClinica, CRF, CRF Rules, Validation

1. Introduction

1.1. Background

Clinical trials are a critical component of medical knowledge, allowing researchers to evaluate the safety and effectiveness of new treatments, drugs and interventions. One of the key elements of clinical trials is the use of case report forms (CRFs) to collect and manage data on study subjects. CRFs are typically designed by the study sponsor or the

[1] *Corresponding Author*: Dipl.-Inf. Matthias Löbe, Institut für Medizinische Informatik, Statistik und Epidemiologie, Universität Leipzig, Härtelstraße 16-18, 04107 Leipzig, matthias.loebe@imise.uni-leipzig.de

principal investigator and they are used to capture specific information about each subject/patient enrolled in the study.

Case Report Forms can vary in complexity and format depending on the study design and the data being collected. Some CRFs are relatively simple, gathering only basic information such as patient demographics and medical history, while others can be more complex, capturing detailed information on study endpoints, adverse events and other outcomes of interest.

They are usually filled out by study personnel, such as clinical research coordinators or nurses who are trained to ensure that the data is collected accurately and consistently. The data is then used to analyze the safety and efficiency of the study intervention and to draw conclusions about its effectiveness in treating the condition of interest.

Overall, the design and implementation of CRFs is a complex and time-consuming process that requires careful consideration of the trial design, the data to be collected and the regulatory requirements governing the study. In addition to that, the collection and management of CRF data must adhere to strict data privacy and security regulations to protect the privacy of study subjects and to ensure the integrity of the data gathered.

1.2. Objective and Requirements

To address the challenge of data quality, case report forms need validation rules to ensure the data collected is accurate, complete and consistent. Validation rules are typically designed to detect errors, inconsistencies or omissions in the data entered into the CRF and to provide feedback to the data entry personnel or study site staff.

We introduce a new tool that generates rules for CRFs, designed specifically for use with OpenClinica and the fork LibreClinica. Since LibreClinica is actively being developed and OpenClinica development moved to a cloud-based solution not being open source, we will refer only to LibreClinica in the following. Our requirements were threefold: First, non-IT personnel such as clinical researchers and data managers should be able to understand and formulate quality rules. Second, the solution should be less prone to errors than writing the rules in XML code manually. Third, generating rules of very common appearance should be simplified.

The ocRuleTool streamlines the process of creating and implementing validation rules by allowing users to generate rules based on the structure and content of their CRFs. The goal is to extend the normative Excel data dictionary template so that it can be populated with field identifiers and variables to restrict input to a desired convention. Such restrictions can be upper and lower bounds for numeric values or more complex data types like dates in a chronological order. Another option are radio buttons to differentiate behavior in fields for when you are in a situation to select from a list of medications of which each has different limits. Rules can also be as easy as requiring an input before saving the entry. It is highly important that not only IT personnel but also study assistants will be able to write rules like inclusion or exclusion criteria for study arms.

Writing XML code manually can cause a range of different errors. Forgotten parenthesis are often the source of troublesome bugs. It is also quite demanding to keep all the different tag names and their respective hierarchy in mind to address the right sections and field attributes of a case report form. By choosing a tabular form it is easy to navigate to the right field and apply the desired barriers.

In this paper, we will describe the design and implementation of our tool as well as its use case and limitations with LibreClinica. We will also discuss the benefits it provides for healthcare organizations and clinical researchers including improved rules designing process leading to reduced data entry errors and increased efficiency in the data collection process.

We will demonstrate how our tool can generate validation rules for LibreClinica CRFs, using a simple command-line interface. These rules can be customized to fit specific needs of a given clinical trial and can be easily integrated into LibreClinica to ensure the quality and consistency of the data collected. Overall, the ocRuleTool represents a significant advancement in designing and implementing case report forms for clinical trials. By simplifying the process of generating validation rules, it allows researchers to focus on the important work of collecting and analyzing data, leading to better outcomes for patients and improved medical knowledge.

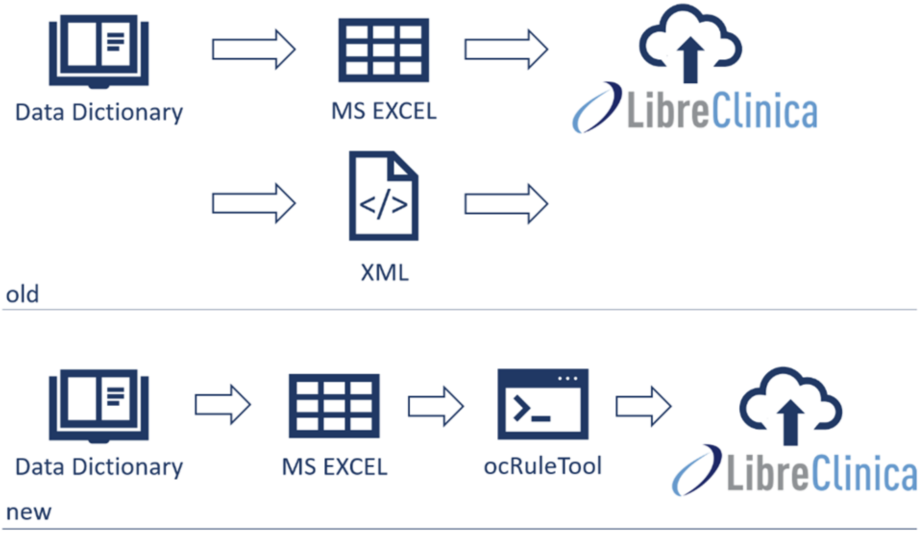


Figure 1. Old vs. new CRF specification workflow using the ocRuleTool.

2. State of the art

2.1. Related Work

One of the challenges in designing and implementing validation rules for case report forms is the need to manage and analyze the large amounts of medical data collected during clinical trials. A prominent application for this use case is LibreClinica [1], an open-source fork of OpenClinica which is used extensively in clinical research [2].

LibreClinica is a web-based electronic data capture (EDC) system designed for use in clinical research. Its primary functionality is to allow researchers to collect and manage data from clinical trials, observational studies, registries and other research projects. One of the key features is enabling users to create data entry forms and allowing them to collect the specific data they need for their study. LibreClinica provides primitive built-in data validation rules to ensure that entries into the system meet certain standards and requirements. Researchers can also create their own validation rules to ensure data accuracy. Other functionalities are user and permission management, data visualization and data in- and export. LibreClinica also provides APIs and web services for integration with other software and local workflows without the need for extensive programming skills.

While there are hundreds of similar solutions available, especially commercially, few are free and open source. One widely used application for creating data collection forms, especially for medical and translational research projects, is REDCap [3]. Developed at Vanderbilt University (USA) it was made international through the REDCap Consortium, which now includes thousands of institutions. Over 1.4 million projects have been generated with REDCap.

Data quality checks can also be applied asynchronously. An alternative approach for defining rules and executing them on a collected dataset retrospectively is the dataquieR package described in [4].

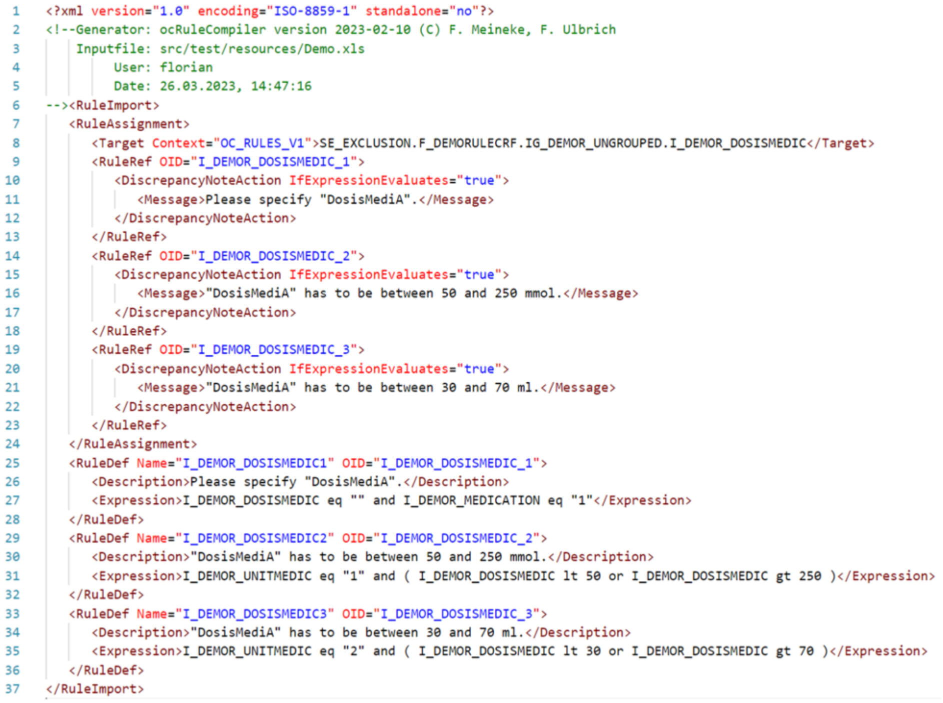
The field of data quality is very broad, even if restricted to health data. Numerous frameworks exist, one popular being the Harmonized Data Quality Assessment Terminology and Framework by Kahn et.al. [5] used in the Medical Informatics Initiative [6].

2.2. Shortcomings

LibreClinica requires users to create their own validation rules in an eXtensible Markup Language (XML) format. Designing rules in XML has several disadvantages, particularly in the context of clinical research. The major issue is that editing XML files is not a task that can be performed by the investigators or CRF designers.

Due to the fact that the XML format is built like a tree with elements having attributes and containing child elements, the structure is confusing and hard to understand. Figure 2 shows an example of medication dosage that should be present (completeness) and, depending on the unit, conforms to two different intervals. In this case the root element is <RuleImport> containing multiple <RuleAssignment> and <RuleDef> (rule definition) elements. The <RuleAssignment> wraps up the <Target> and at least one rule definition.

This makes creating validation rules a time-consuming task. Simple rules like specifying that one event can only happen after another takes a lot of lines and is hard to read. Furthermore, problems are difficult to debug since errors are only shown after the upload to LibreClinica. Additionally, long-term management of changes in XML rules required to reflect changes in the CRF specification are hard to track.



```
<?xml version="1.0" encoding="ISO-8859-1" standalone="no"?>
<!--Generator: ocRuleCompiler version 2023-02-10 (C) F. Meineke, F. Ulbrich
    Inputfile: src/test/resources/Demo.xls
         User: florian
         Date: 26.03.2023, 14:47:16
--><RuleImport>
   <RuleAssignment>
      <Target Context="OC_RULES_V1">SE_EXCLUSION.F_DEMORULECRF.IG_DEMOR_UNGROUPED.I_DEMOR_DOSISMEDIC</Target>
      <RuleRef OID="I_DEMOR_DOSISMEDIC_1">
         <DiscrepancyNoteAction IfExpressionEvaluates="true">
            <Message>Please specify "DosisMediA".</Message>
         </DiscrepancyNoteAction>
      </RuleRef>
      <RuleRef OID="I_DEMOR_DOSISMEDIC_2">
         <DiscrepancyNoteAction IfExpressionEvaluates="true">
            <Message>"DosisMediA" has to be between 50 and 250 mmol.</Message>
         </DiscrepancyNoteAction>
      </RuleRef>
      <RuleRef OID="I_DEMOR_DOSISMEDIC_3">
         <DiscrepancyNoteAction IfExpressionEvaluates="true">
            <Message>"DosisMediA" has to be between 30 and 70 ml.</Message>
         </DiscrepancyNoteAction>
      </RuleRef>
   </RuleAssignment>
   <RuleDef Name="I_DEMOR_DOSISMEDIC1" OID="I_DEMOR_DOSISMEDIC_1">
      <Description>Please specify "DosisMediA".</Description>
      <Expression>I_DEMOR_DOSISMEDIC eq "" and I_DEMOR_MEDICATION eq "1"</Expression>
   </RuleDef>
   <RuleDef Name="I_DEMOR_DOSISMEDIC2" OID="I_DEMOR_DOSISMEDIC_2">
      <Description>"DosisMediA" has to be between 50 and 250 mmol.</Description>
      <Expression>I_DEMOR_UNITMEDIC eq "1" and ( I_DEMOR_DOSISMEDIC lt 50 or I_DEMOR_DOSISMEDIC gt 250 )</Expression>
   </RuleDef>
   <RuleDef Name="I_DEMOR_DOSISMEDIC3" OID="I_DEMOR_DOSISMEDIC_3">
      <Description>"DosisMediA" has to be between 30 and 70 ml.</Description>
      <Expression>I_DEMOR_UNITMEDIC eq "2" and ( I_DEMOR_DOSISMEDIC lt 30 or I_DEMOR_DOSISMEDIC gt 70 )</Expression>
   </RuleDef>
</RuleImport>
```

Figure 2. Example of the XML dialect for quality checks in LibreClinica [7], limited to three rules from Figure 3 for readability.

3. Concept

Microsoft Excel is a spreadsheet software being widely used for organizing, analyzing and manipulating data in various fields. We extended the normative LibreClinica data dictionary Excel template to add validation rules for CRFs in a clear and easily readable format (see Figure 3).

According to our personal experiences, data entry checks in clinical trials can be grouped into a manageable number of classes of rule types

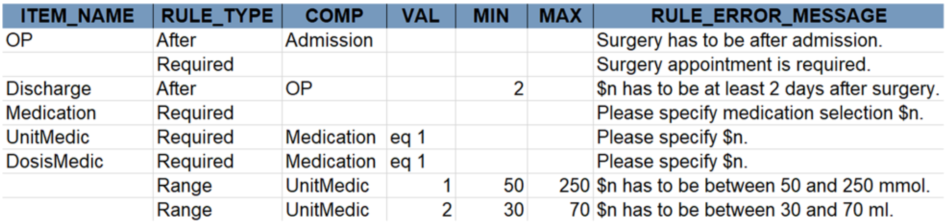
1) conformance validations such as less, greater than or ranges
2) completeness validation such as field being (conditionally) required or empty
3) temporal validations such as less order of events (dates
4) other more complex validations (made possible using custom expressions).

It is also a good practice to provide an error message that is displayed in case of rule violations. The supported rule patterns are summarized in Table 1.

The Excel-file is then processed by the ocRuleTool via a command line interface (CLI) to generate the XML dialect ready to be uploaded to LibreClinica where the rules can be viewed and managed. Figure 3 shows eight data quality rules defined each in a single line in the LibreClinica CRF specification sheet instead of dozens of lines of XML code (see Figure 2). Figure 4 shows the triggering of a rule upon violation of the condition in the eCRF of a concrete subject.

Table 1. Supported Rule Pattern. The five patterns mentioned first account for more than 95% of all data quality rules in practice.

Rule Type	Explanation
After	Used to compare dates. Similar to "less than". Tests if COMP is less than ITEM_NAME. Can be enhanced with MIN and MAX as a range of days after the event.
Before	Used to compare dates. Similar to "greater than". Tests if COMP is greater than ITEM_NAME. Can be enhanced with MIN and MAX as a range of days before the event.
Required	There must be data within the item.
Empty	The item must be empty.
Range	Defines a range of values for the item. Use in combination with MIN and MAX.
Expression	Use for a custom statement in COMP: [item] [operator] [value]



ITEM_NAME	RULE_TYPE	COMP	VAL	MIN	MAX	RULE_ERROR_MESSAGE
OP	After	Admission				Surgery has to be after admission.
	Required					Surgery appointment is required.
Discharge	After	OP		2		$n has to be at least 2 days after surgery.
Medication	Required					Please specify medication selection $n.
UnitMedic	Required	Medication	eq 1			Please specify $n.
DosisMedic	Required	Medication	eq 1			Please specify $n.
	Range	UnitMedic	1	50	250	$n has to be between 50 and 250 mmol.
	Range	UnitMedic	2	30	70	$n has to be between 30 and 70 ml.

Figure 3. Example spreadsheet. The comp column can be used to formulate conditional dependencies.

4. Implementation

4.1. Solution / Results

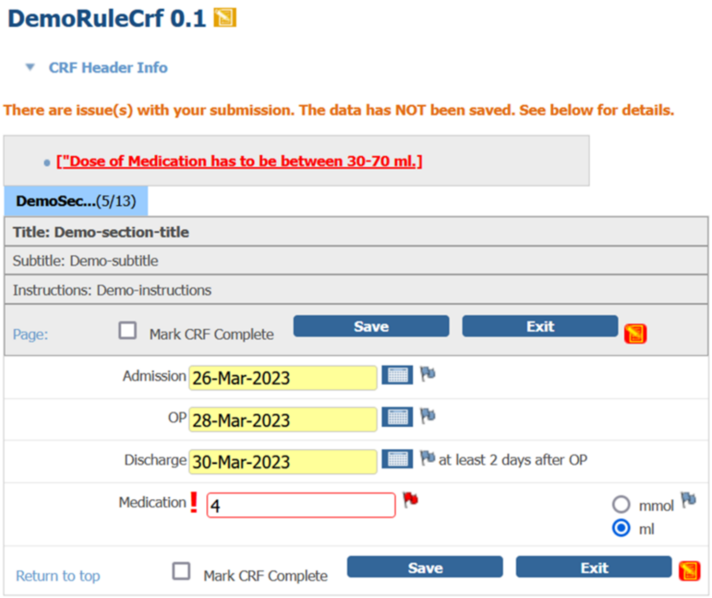
The ocRuleTool is a pure Java application with an easy to use command line interface. Development is managed with Maven. Maven is a popular build automation tool used primarily for Java projects. It provides a comprehensive and flexible way to manage dependencies, build and package Java applications and manage the entire build process from start to finish. It simplifies the management of dependencies, making it easy to add and update third-party libraries and frameworks that your project depends on. It also provides a standardized build process that can be easily replicated across different projects, making it easier to maintain and manage large codebases.

To access cells in our Excel document we chose the Apache POI framework. Apache POI provides support for reading and writing data to Excel spreadsheets, including creating and modifying worksheets, manipulating cells and ranges, and formatting data. It also supports working with formulas, charts, and other advanced features of Excel.

Apache POI is widely used in various Java-based applications, such as data analysis tools, financial reporting applications, and document management systems. It is free and open-source, released under the Apache License 2.0, making it a great choice for working with Microsoft Office documents.

The information retrieved by POI is processed by various classes and the DocumentBuilder from W3C to match the XML tree required by LibreClinica. With a simple command executing the jar and giving the Excel-file as a parameter the ocRuleTool generates the rules.xml which is ready to upload to LibreClinica. Several options are provided to add important functionality like character encoding.

To verify the transformation and for enabling the community to add features in the future, the project is well documented and tested with JUnit, a widely used Java testing framework. It provides a set of APIs for writing and running unit tests, which are automated tests designed to verify the functionality of individual units or components.



DemoRuleCrf 0.1

▾ CRF Header Info

There are issue(s) with your submission. The data has NOT been saved. See below for details.

- ["Dose of Medication has to be between 30-70 ml.]

DemoSec...(5/13)

Title: Demo-section-title

Subtitle: Demo-subtitle

Instructions: Demo-instructions

Page: □ Mark CRF Complete

Admission 26-Mar-2023

OP 28-Mar-2023

Discharge 30-Mar-2023 at least 2 days after OP

Medication ! 4

○ mmol

● ml

Return to top □ Mark CRF Complete

Figure 4. Short caption.

4.2. System in Use

There are multiple clinical trials that benefited from the rule validation modeling with the ocRuleTool. One of them is the ACTION [8] study which investigated whether the detection of bacterial DNA in ascites is associated with shortened survival in patients with liver cirrhosis who have signs of infection. Another use case was the multicentric MEDUSA [9] study, which aimed to provide better healthcare for patients suffering from sepsis. The ocRuleTool has been used in more than 14 studies in total up to now.

5. Lessons Learned

The ocRuleTool is a software solution that fulfills the requirements of an easy way to create validation rules for LibreClinica. It straightforward to implement new rules to filter out inconsistent data entries in CRFs. The tool also indicates missing data fields and explains errors with short messages, so that the user can identify violations. This has been confirmed by CRF designers who found that not only is the rules designing process easier but most important the result is significantly less error prone.

It was a beneficial decision to add more documentation later on to improve readability and make the tool more accessible for additional features. Valuable was also the decision to build on an extensive framework with Apache POI to extract information from MS Excel sheets and work with them.

A shortcoming of the ocRuleTool is the lack of an automatic variable name synchronization. When you update a CRF in LibreClinica and change the name, this is obviously not reflected in the Excel sheet for the rules and needs to be done manually. A solution for this problem would be working with a database which queries the labels from LibreClinica and integrates the changes made in the ocRuleTool. Steps have already been made to address this issue.

The main motivation for this project is to fill a gap in offering a complete open source solution for handling CRFs and their management. Naturally further iterations will aim to refrain from using Excel and substitute it with LibreOffice to align with the goal of being completely free.

With the open-source release there may be other requirements for new users. It is possible to submit improvements or completely new features as a pull request on GitHub, to further increase the range of services.

Despite being an extra step the process of designing CRFs was made more agile and user friendly in comparison to writing complex XML files. There was a little controversy about using the command-line-interface. Implementing a graphical user interface (GUI), potentially fully integrated to LibreOffice could be discussed for future releases or collaborations.

6. Conclusion

Originally this application was designed as a quick solution with a very specialized use case. Over time, the ocRuleTool found more usage and a more comprehensive documentation was written. To give back to the community, it was decided to release the tool as open source so that more researchers could benefit from it. As the best approach for an easy way to share this project and allow distributed development we decided to put it on GitHub [10]. We hope that the open-source nature of this project and the opportunity to add features inspire many medical experts to add this application to their tech stack. Data quality rules are a significant source of knowledge for interpreting data sets. Especially in data sharing, there is a risk that the lack of explicit specification of these algorithms limits reuse and reproducibility. Future work will address the requirement to harmonize data quality rules for common data elements.

Declarations

Conflict of interest: The authors declare, that there is no conflict of interest.

Acknowledgements

Research was supported by DFG grant WI 1605/10-2.

References

[1] Reliatec GmbH (2019) LibreClinica [Source Code] https://github.com/reliatec-gmbh/LibreClinica

[2] Löbe M, Meineke F, Winter A. Scenarios for Using OpenClinica in Academic Clinical Trials. Stud Health Technol Inform. 2019; PMID: 30942748.

[3] Harris PA, Taylor R, Thielke R, Payne J, Gonzalez N, Conde JG. Research Electronic Data Capture (REDCap) - a Metadata Driven Methodology and Workflow Process for Providing Translational Research Informatics Support. 2009; DOI: 10.1016/j.jbi.2008.08.010.

[4] Richter A, Schmidt CO, Krüger M, Struckmann S. R Packages for Data Quality Assessments and Data Monitoring: A Software Scoping Review with Recommendations for Future Development. 2021; DOI: 10.21105/joss.03093.

[5] Kahn MG, Callahan TJ, Barnard J, Bauck AE, Brown J, Davidson BN, Estiri H, Goerg C, Holve E, Johnson SG, Liaw ST, Hamilton-Lopez M, Meeker D, Ong TC, Ryan P, Shang N, Weiskopf NG, Weng C, Zozus MN, Schilling L. A Harmonized Data Quality Assessment Terminology and Framework for the Secondary Use of Electronic Health Record Data. EGEMS (Wash DC). 2016; DOI: 10.13063/2327-9214.1244.

[6] Löbe M, Kamdje-Wabo G, Sinza AC, Spengler H, Strobel M, Tute E. Towards Harmonized Data Quality in the Medical Informatics Initiative - Current State and Future Directions. Stud Health Technol Inform. 2022; DOI: 10.3233/SHTI210904.

[7] OpenClinica, LLC (2023) https://docs.openclinica.com/3-1/rules/rules-creating-rules/

[8] https://drks.de/search/de/trial/DRKS00000631

[9] Schwarzkopf, Daniel; Rüddel, Hendrik; Thomas-Rüddel, Daniel O.; Felfe, Jörg; Poidinger, Bernhard; Matthäus-Krämer, Claudia T.; Hartog, Christiane S.; Bloos, Frank. Perceived Nonbeneficial Treatment of Patients, Burnout, and Intention to Leave the Job Among ICU Nurses and Junior and Senior Physicians. Critical Care Medicine. 2017; DOI: 10.1097/CCM.0000000000002081

[10] IMISE (2023) ocRuleTool [Source Code] https://github.com/IMISE/ocRuleTool