ICT for Health Science Research
A. Shabo (Shvo) et al. (Eds.)

doi:10.3233/978-1-61499-959-1-26

A REST Service for the Visualization of Clinical Time Series Data in the Context of Clinical Decision Support

Philipp UNBERATH [a,1], Hans-Ulrich PROKOSCH [a] and Stefan KRAUS [a]
[a] *Medical Informatics, University Erlangen-Nürnberg, Erlangen, Germany*

Abstract. *Background:* University Hospital Erlangen provides clinical decision support (CDS) functions in the intensive care setting, that are based on the Arden Syntax standard. These CDS functions generate extensive output, including patient data charts. In the course of the migration of our CDS platform we revised the charting tool because although the tool was generally perceived as useful, the clinical users reported several shortcomings. *Objective:* During the migration of our CDS platform, we aimed at resolving the reported shortcomings and at developing a reusable and parameterizable charting tool, driven by best practices and requirements of local clinicians. *Methods:* We conducted a requirements analysis with local clinicians and searched the literature for well-established guidelines for clinical charts. Using a charting library, we then implemented the tool based on the found criteria and provided it with a REST interface. *Results:* The criteria catalog included 18 requirements, all of which were successfully implemented. The new charting tool fully replaced the previous implementation in clinical routine. It also provides a web interface that enables clinicians to configure charts without programming skills. *Conclusion:* The new charting tool combines local preferences with best practices for visualization of clinical time series data. With its REST interface and reusable design it can be easily integrated in existing CDS platforms.

Keywords. clinical decision support, patient data charts, visualization

1. Introduction

In 2006, University Hospital Erlangen (UHER) introduced a commercial patient data management system (PDMS) in the intensive care unit (ICU) setting [1]. To extend the PDMS with various features demanded by local clinicians, we integrated a commercial clinical decision support (CDS) platform [2]. The CDS system is based on the Health Level 7 (HL7) Arden Syntax standard, where CDS functions are implemented in the form of Medical Logic Modules (MLMs) [3]. MLMs are typically used for data-driven clinical event monitoring. At UHER, however, they are mostly used for user-driven information retrieval at the point of care. While data-driven MLMs usually produce less extensive output, user-driven MLMs often yield substantial output that may include formatted text, tables, and charts.

[1]Corresponding Author: Philipp Unberath, Chair of Medical Informatics, Friedrich-Alexander-Universität Erlangen-Nürnberg, Wetterkreuz 13, 91058 Erlangen, Germany. E-Mail: philipp.unberath@fau.de

Prior to this study, the generation of charts at UHER has been implemented by means of a specific sub-MLM, which itself makes use of a JavaScript chart library. Although this chart MLM has been in routine use for about seven years and proved to be suitable for practical use, clinicians regularly demanded additional features, such as the highlighting of an optimal range for specific laboratory results. This led to an evaluation of our custom setup, which included the comparison with other systems.

In 2015 Sittig et al. [4] evaluated the graphical display of clinical parameters of 8 electronic health records (EHRs) based on 11 criteria that had been selected using literature and expert opinion. None of the examined EHRs met all criteria. Our previously used charting tool met six of them.

When UHER migrated from the standard compliant Arden Syntax platform to an experimental version termed PLAIN [5] in 2017, we decided to redevelop the chart MLM, extending it with various features demanded by local physicians. Moreover, we set out to fulfill the criteria for clinical charts published by Sittig et al.

The purpose of this paper is to describe our current setup in clinical use and present our tool as a way of standardized visualization of time series data in the context of CDS.

2. Methods

Based on the criteria for graphical displays proposed by Sittig et al. we developed a catalog of requirements for our charting tool. Additionally, we conducted interviews with local physicians to determine further requirements. During the interviews, physicians where asked to prioritize the requirements to decide which of them would be implemented in the new charting tool.

Subsequently, the architecture of the application was defined and potentially suitable JavaScript libraries were analyzed. Our search concentrated on open-source libraries, which were not work in progress (version number ≥ 1.0) and still actively developed. The library should also natively support as many of the requirements as possible without the need to implement features within the library.

The new charting tool was designed as a REST service using the PHP Slim framework[2] as the back end for the REST interface. The configuration of a chart is specified with a JSON document. These documents are described and validated via a JSON schema, which is used together with Docson[3] to automatically generate documentation for all parameters of the interface. The document for a chart consists of an array of data points, as well as 30 optionally configurable parameters and the option to draw additional auxiliary lines in the graph.

3. Results

Our requirements catalog consisted of the 11 criteria of Sittig et al. as well as seven additional local requirements. Four of them refer to customization of details within the chart and three relate to more general properties of the chart. Local physicians requested the options to customize:

[2]https://www.slimframework.com/
[3]https://github.com/lbovet/docson

- the color and width of lines,
- the color and size of points,
- the background color of the chart,
- and the color of reference ranges.

The more general requirements are:

- a smooth and interactive display,
- a legend and highlighting of specific points on mouseover,
- and the options to zoom and pan within charts.

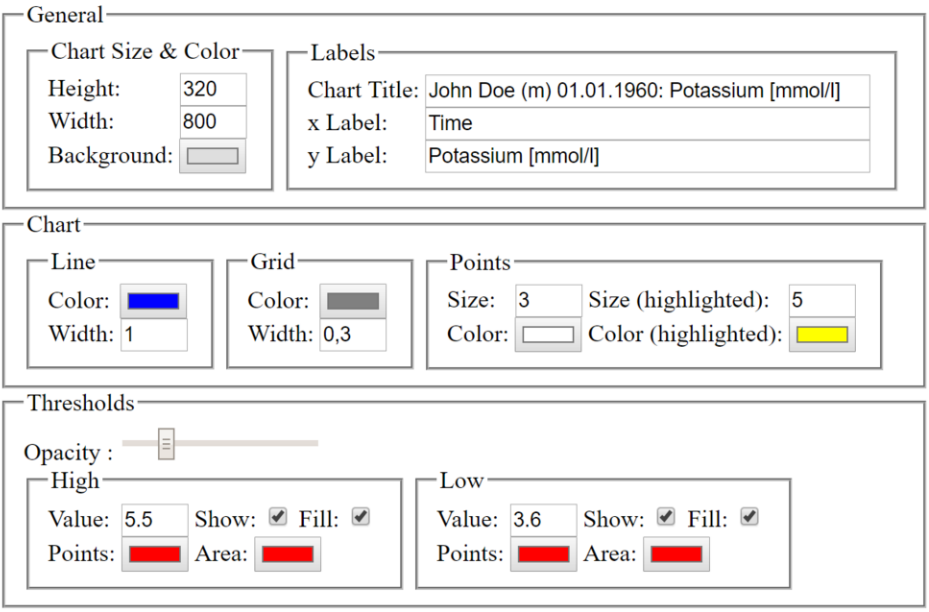

Figure 1. The parametrization section of the template generator.

The search for a suitable JavaScript library yielded over 20 different candidates. Seven of them were further examined after excluding all libraries which were either still work in progress or their development appeared to be discontinued. Ultimately, dygraphs[4] was chosen out of the seven remaining libraries under consideration, as it supported all of the 18 requirements out of the box. This library also supports interactive charts as standard and is highly customizable in display and behavior.

A straightforward graphical user interface (Figure 1) was implemented to allow physicians to customize diagrams without programming skills or knowledge about the technical background of the application. The interface enables clinicians to explore the configuration options using sample data or generate style templates for charts.

The integration of the charting tool in the local CDS environment was achieved in the form of a sub-MLM, that can be called as the user-defined function (UDF) @*lineplot* (Figure 2) in PLAIN. This UDF passes the data points and the configuration parameters to the REST interface of the charting tool and returns the Javascript code for the chart.

[4]http://dygraphs.com/

```
emr := @patientrecord casenumber;
WRITE @lineplot {
    patname: emr.adt.fullname,
    birthdate: emr.adt.birthdate,
    gender: emr.adt.gender,
    values: emr.bga.potassium,
    caption: "Potassium [mmol/l]",
    yLabel: "Potassium [mmol/l]",
    upperlimit: 5.5,
    lowerlimit: 3.6
};
```

Figure 2. Basic usage of the charting tool in PLAIN. The code snippet produces an output chart like the one in Figure 3.

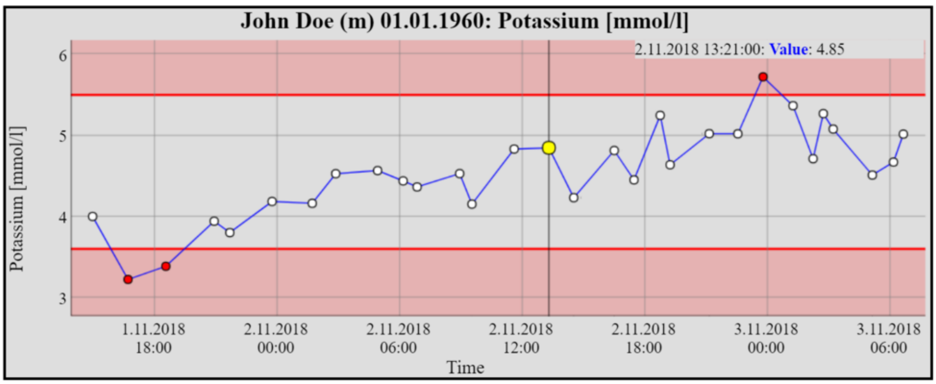

Figure 3. An example chart generated by the developed tool.

4. Discussion

The new charting tool was introduced to UHER about one year ago and has been in productive use ever since. It was observed that in particular the diagnosis-related group (DRG) coders benefit from the tool: The primary use is to quickly identify clinical parameters leaving their defined reference range. The overall feedback from clinical users indicates that the new charting tool is perceived as an improvement in comparison to the old tool and substantially facilitates the interpretation of diagnostic time series data.

It is interesting to note that, although the x-label was initially required by the criteria catalog of Sittig et al., it was perceived unnecessary by clinicians in our setting, because the x-axis always displays temporal data.

While the charting tool supports creation of charts that fulfill all of the aforementioned requirements, it is still necessary to explicitly set the configuration items by the caller. In favor of an easy to use interface every configuration element has a default value assigned, which can not always represent the requirements. As an example, the title of a chart would need to be specified to meet the criteria catalog but is left empty by default in our implementation.

One capable and health-centered visualization tool is the Web Visualization library of the Open mHealth project [6]. It was designed for mobile health data and builds upon their common data schema. While its standardized data model and the provided mapping tool of HL7 messages are strong arguments for its use, the project is mainly centered around mobile health rather then clinical application. That means we would need to implement all parameters to be visualized in their data model, as our PDMS does not support an HL7 data export. Even if an export interface is available, the mapper has to be extended to support the clinical parameters to be displayed.

During our search for suitable technologies we also examined D3.js[5] as it is one of the most frequently used graphical JavaScript libraries. Although compliant with all our requirements, we assumed that the additional programming efforts would outweigh the added value. Moreover, the code would have been more complicated, which hinders our aim of a reusable and adaptable solution.

As there is no standard for the visualization of clinical time series data, every system uses self-designed charting tools for the display of patient data. Sittig et al. showed that a variety of widely used commercial EHRs do not optimally visualize data, which does not only hinder an easy and quick interpretation of the data, but may promote misinterpretation.

The approach of this study combines best practices coming from literature and experts to generate a tool for unambiguous and easy to interpret time series charts. The extensible and parameterizable design of this tool allows its adaption to local preferences.

5. Acknowledgments

The present work was performed in (partial) fulfillment of the requirements for obtaining the degree "Dr. rer. biol. hum." from the Friedrich-Alexander-Universität Erlangen-Nürnberg (PU).

References

[1] Bürkle, Thomas, et al. "Implementation of a patient data management system-an evaluation study of workflow alterations." Studies in health technology and informatics 160.Pt 2 (2010): 1256-1260.

[2] Kraus, Stefan, et al. "Integrating Arden-Syntax-based clinical decision support with extended presentation formats into a commercial patient data management system." Journal of clinical monitoring and computing 28.5 (2014): 465-473.

[3] Hripcsak, George, et al. "Rationale for the Arden syntax." Computers and Biomedical Research 27.4 (1994): 291-324.

[4] Sittig, Dean F., et al. "Graphical display of diagnostic test results in electronic health records: a comparison of 8 systems." Journal of the American Medical Informatics Association 22.4 (2015): 900-904.

[5] Kraus, Stefan. "Generalizing the Arden Syntax to a Common Clinical Application Language." Studies in health technology and informatics 247 (2018): 675-679.

[6] Estrin, Deborah, and Ida Sim. "Open mHealth architecture: an engine for health care innovation." Science 330.6005 (2010): 759-760.

[5]https://d3js.org/