

# Implementing CDS Hooks Communication in an Arden-Syntax-Based Clinical Decision Support Platform

Martin SPINETH<sup>a</sup>, Andrea RAPPELSBERGER<sup>b</sup> and Klaus-Peter ADLASSNIG<sup>a,b,1</sup>

<sup>a</sup> Medexter Healthcare GmbH, Borschkegasse 7/5, 1090 Vienna, Austria

<sup>b</sup> Section for Artificial Intelligence and Decision Support, Medical University of Vienna, Spitalgasse 23, 1090 Vienna, Austria

**Abstract.** Introduction: Arden Syntax is a Health Level Seven International standard for the development of clinical decision support (CDS) solutions. CDS Hooks is a specification for an interface to call CDS services from an electronic health record. Implementing the CDS Hooks interface into ArdenSuite, an Arden-Syntax-based CDS platform, will further enhance its interoperability with external systems. Methods: The development of this interface consisted of three parts: (1) The RESTful interface defined by CDS Hooks had to be implemented; this included the support of specifically formatted data for the input and output of the CDS functions. (2) The existing FHIR connector had to be adjusted. In CDS Hooks, a FHIR server can be defined for each CDS service call individually. In the current version of the connector, a FHIR server has to be configured in advance to call the CDS service. (3) Additional data points that did not exist previously, such as a description or hook for each CDS service, had to be collected and saved. Results: Basic support of the CDS Hooks API was established in the ArdenSuite and tested. The FHIR connector was adapted. Discussion: The missing parts for full CDS Hooks support were implemented. As a use case, we employed Hepaxpert, an interpretive system for hepatitis serology tests. CDS Hooks defines FHIR as the only data source. ArdenSuite, in contrast, offers the option of various data sources with connectors and extensions.

**Keywords.** Decision Support Systems, Clinical; Arden Syntax; CDS Hooks; Interoperability; Hepaxpert

## 1. Introduction

### 1.1. Arden Syntax medical knowledge representation and processing

Arden Syntax [1] is a standard developed by Health Level Seven (HL7) International [2]. It is used to represent and process medical knowledge by representing it in a computer-executable format, which is then used in clinical decision support (CDS) systems [3]. The most recent version is 2.10 and was published in November 2014. Backward compatibility with older versions of Arden Syntax was preserved.

---

<sup>1</sup> Corresponding Author, Univ.-Prof. DI Dr. Klaus-Peter Adlassnig, Medexter Healthcare GmbH, Borschkegasse 7/5, 1090 Vienna, Austria; E-Mail: kpa@medexter.com.

Medical logic modules (MLMs) form the basic knowledge elements of Arden Syntax. Each MLM contains sufficient logic for at least one clinical decision, one complex calculation, or even an organizational decision. Arden Syntax receives data from external sources through various modes. The most basic method is with the outside call of the MLM via direct input parameters. An MLM can also receive data from another MLM that was executed previously. A further versatile method of receiving data is via so-called curly brace expressions. By definition, these statements can connect to, and receive data from other external sources. Here the data-source-specific query for retrieving data is separated from the remaining Arden Syntax code by curly braces.

Arden Syntax is able to resemble natural language, which makes it easier to write and read, even with less extensive programming skills [4].

1.2. CDS Hooks clinical decision support service

CDS Hooks [5] is a hook-based pattern for automatically invoking CDS functions from within a physician’s electronic health record (EHR) workflow. The specification had an HL7 International ballot cycle opened until 7 May 2018; its 1.0 version is still under development.

All communication based on this application programming interface (API) is defined by using fixed representational state transfer (RESTful) interfaces and the JavaScript object notation (JSON) format for data representation with fixed data fields.

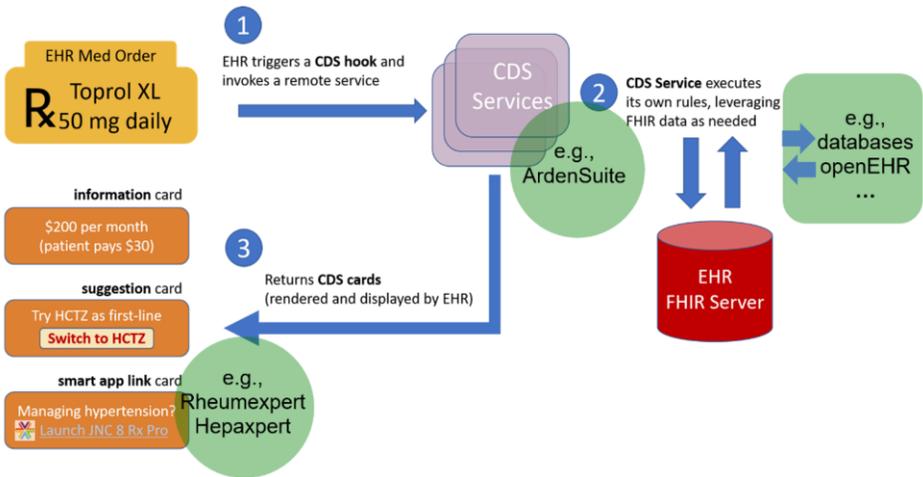


Figure 1. Scheme of a CDS Hooks communication. (Adapted from: [5]; light green boxes by Medexter)

A basic scheme of the CDS Hooks communication is shown in Figure 1. A physician prescribes a medication to a patient. This action triggers a hook in the EHR [(1) in Figure 1] which leads to the EHR calling all of the CDS services registered under this hook. The CDS services execute their rules and retrieve fast healthcare interoperable resources (FHIR) data as needed [(2) in Figure 1]. Finally, the CDS services can return one or more cards [(3) in Figure 1], which are displayed by the EHR. Cards may contain plain text (information cards), suggestions for changes to the intended action of the physician (suggestions cards), or links to an external app [6] (smart app link cards). According to the CDS Hooks definition, the user interface presentation of the cards to the physician is the task of the EHR that invoked the CDS service.

### 1.3. Hepatitis serology test interpretation by Hepaxpert

Hepaxpert [7] is an app developed by Medexter Healthcare for the textual interpretation of hepatitis A, B, and C serology antigen and antibody test results. After the test results are entered in the app, an interpretive text for this specific combination of serology test results is displayed. An early version of Hepaxpert has been described in [8]. Currently a total of 61,440 possible combinations of test results can be interpreted.

### 1.4. Objectives of this technical study

The aim of the study was to extend the interoperability of the ArdenSuite CDS platform to EHRs, such that CDS Hooks [5] is used to invoke decision support with Hepaxpert. To achieve this goal, the CDS Hooks API had to be integrated into the ArdenSuite CDS platform [9, 10]. This connection was then tested by invoking the execution of the Hepaxpert MLMs via a RESTful interface that fits the CDS Hooks definition, and by returning an information and a smart app link card. Here the information card includes Hepaxpert's interpretive text for the presented hepatitis serology test results, and the link card the respective link to a cloud-based web version of Hepaxpert.

## 2. Methods

For this project, Medexter's ArdenSuite [9] serves as a representation of an Arden-Syntax-based CDS solution. As shown in the middle part ("*processing*") of Figure 2, ArdenSuite is split into the ArdenSuite integrated development environment (IDE) and the ArdenSuite server. The CDS Hooks API is integrated into the server, because this is a part of the ArdenSuite that is meant to communicate with external systems [9]. The IDE is a tool for the development of Arden Syntax MLMs.

For the integration of the CDS Hooks API into the ArdenSuite server, we used the latest available version of the CDS Hooks specification. The layout for this implementation is indicated by green boxes in Figure 1. The process of development consisted of three main parts: (1) The implementation of the RESTful interface that is defined in the CDS Hooks specification. This includes support of the defined format of the RESTful body, the return of the CDS results as cards, and a converter of JSON-formatted data into Arden Syntax objects such that the data can be used in Arden Syntax MLMs. (2) The pre-existing FHIR connector had to be adapted in order to support an individual FHIR connection for each MLM, provided via the MLM call. (3) The management and use of data points required by the CDS Hooks API needed to be implemented. This included the management of hooks for each MLM, the data points that need to be pre-fetched by the calling EHR, a CDS-Hooks-required description of each MLM, as well as other data points demanded by the CDS Hooks specification.

To test the current version of the CDS Hooks API for ArdenSuite, we developed an MLM that returns its conclusion in the form of a card. The MLM obtains a patient's id directly from the EHR via the RESTful call. The MLM establishes a connection to a FHIR server and requests hepatitis serology test results. If they exist in the patient's record, the MLM returns a textual interpretation of the test results (information card) and a smart app link card to the calling EHR; the latter contains a link to the Hepaxpert app.

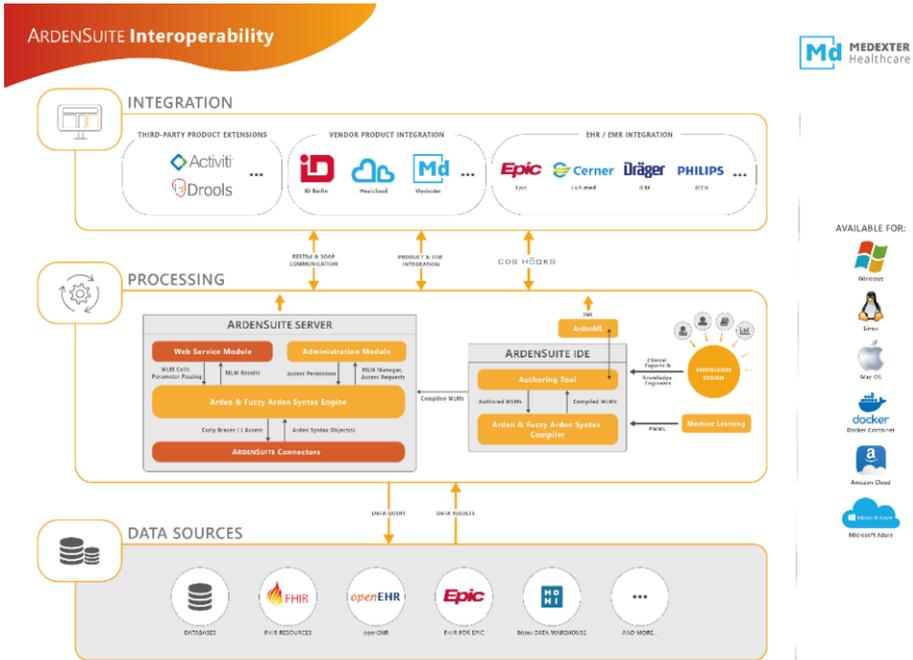


Figure 2. ArdenSuite interoperability with external systems and external data sources [9].

### 3. Results

The parts of the RESTful interface defined by CDS Hooks were implemented and tested. As MLMs are uploaded, a request to insert data needed for the CDS Hooks API is shown. These data are saved in a dedicated database table which includes the hook, a title, a description, and an id. These data are returned when the discovery endpoint is called. When calling an MLM using the CDS Hooks API, an error is returned if the RESTful body does not follow the format defined in the specification. At the current point of development, FHIR data can be received from an FHIR server and used within an MLM, but the FHIR server’s specifications need to be configured in advance in the ArdenSuite server.

The format of the returned data from the ArdenSuite server was extended to support the format specified by CDS Hooks. If the CDS Hooks API is used to call an MLM, the format is automatically set to JSON and the text the MLM returns is surrounded by “{ “cards”: [ ... ] }”, with “...” representing the output of the MLM. In this solution, the MLM author is responsible for formatting the output so that it correctly represents cards as defined by CDS Hooks.

To test this API, an MLM was developed and a FHIR server was prepared with test data. The MLM was executed using two test patients, one with and one without hepatitis serology test results. The correct cards were returned for the patient with hepatology serology data, whereas no card was returned for the patient without hepatitis serology data.

## 4. Discussion

As described in the Master's thesis entitled "Interoperability with a clinical decision support rule engine" [11], some similarities exist between the ArdenSuite and the approach of the CDS Hooks API. Both approaches to CDS put the CDS services in the background without its own graphical user interface. CDS applications such as Hepaxpert, which have their own representation user interface, are termed apps in the CDS Hooks context. CDS services in the sense of CDS Hooks only communicate with the medical user by returning so-called cards to the EHR. This is similar to Arden Syntax MLMs processed by an Arden Syntax engine on an ArdenSuite server, with the difference that the CDS Hooks specification requires a specific RESTful interface and format.

The current implementation of the CDS Hooks API in the ArdenSuite environment is an extension of its functionality and does not replace any of the previously existing functions or interfaces. In other words, the CDS Hooks RESTful interface as well as the previously existing RESTful and SOAP interfaces can be used to call MLMs. In addition, all previously developed connectors and extensions are compatible. MLMs developed for ArdenSuite cannot only access FHIR servers for retrieving external data, but also SQL databases or openEHR data sources as described in [12]. This extends the possible data sources for ArdenSuite. In contrast, the CDS Hooks specification is able to only access FHIR data.

## References

- [1] Health Level Seven International. Health Level Seven Arden Syntax for Medical Logic Systems, Version 2.10, November 2014 [Internet]. Ann Arbor: Health Level Seven International; 2014 [cited 2018 Apr 04]. Available from: [http://www.hl7.org/implement/standards/product\\_brief.cfm?product\\_id=372](http://www.hl7.org/implement/standards/product_brief.cfm?product_id=372).
- [2] Health Level Seven International [Internet]. Ann Arbor: Health Level Seven International; c2018 [cited 2018 Apr 04]. Available from: <http://www.hl7.org/index.cfm>.
- [3] M. Samwald, K. Fehre, J. de Bruin, K.P. Adlassnig, The Arden Syntax standard for clinical decision support: experiences and directions, *J Biomed Inform* **45**(4) (2012), 711–718.
- [4] G. Hripcsak, Writing Arden Syntax Medical Logic Modules, *Comput Biol Med* **24** (1994), 331–363.
- [5] CDS Hooks [Internet], Ann Arbor & Boston: HL7 & Boston Children's Hospital; c2018 [cited 2018 Apr 04]. Available from: <https://cds-hooks.org>.
- [6] SMART Health IT [Internet], Boston: Boston Children's Hospital Computational Health Informatics Program and the Harvard Medical School Department of Biomedical Informatics; c2018 [cited 2018 Apr 04]. Available from: <https://smarthealthit.org/>.
- [7] Medexer Healthcare GmbH [Internet], Vienna: Medexer Healthcare; c2018 [cited 2018 Apr 04]. Hepaxpert. Available from: <https://www.medexer.com/products-and-services/clinical-solutions/laboratory-medicine>.
- [8] K.P. Adlassnig, W. Horak, Development and retrospective evaluation of HEPAXPERT-I: a routinely-used expert system for interpretive analysis of hepatitis A and B serologic findings, *Artif Intell Med* **7** (1995), 1–24.
- [9] Medexer Healthcare GmbH [Internet], Vienna: Medexer Healthcare; c2018 [cited 2018 Apr 04]. ArdenSuite. Available from: <https://www.medexer.com/products-and-services/ardensuite>.
- [10] K.P. Adlassnig, K. Fehre, Service-oriented Fuzzy-Arden-Syntax-based clinical decision support, *Indian J Med Inform* **8** (2) (2014), 75–79.
- [11] M. Spineth, Interoperability with a clinical decision support rule engine [master thesis], University of Applied Sciences Technikum Wien, Forthcoming, Vienna, 2018.
- [12] M. Spineth, A. Rappelsberger, K.P. Adlassnig, Achieving interoperability between Arden-Syntax-based clinical decision support and openEHR-based data systems, *Stud Health Technol Inform* **248** (2018), 338–344.