ACR Assist™

ACR Assist
Proposed Format for Specifying Point-of-Care Computer-Assisted Reporting/Decision Support Modules for Radiologists

Tarik K. Alkasab, MD, PhD; Harlan B. Harvey, MD, JD; Sepehr Sadeghi;

Version 1.0
October 12, 2016

ACR® American College of Radiology
Quality is Our Image

1891 Preston White Drive, Reston, VA 20191

CONTENTS

1 INTRODUCTION

1.1 Purpose

Specify a format for defining interactive guidelines and algorithms for radiologists' use as they create reports for imaging exams.

1.2 Materials and Methods

The proposed format is based on an Extensible Markup Language (XML) schema specified using the RelaxNG Compact syntax. A reference implementation that allows testing of modules was implemented in the JavaScript language using Node.js and Angular framework

1.3 Results

The definition language consists of sections that specify metadata about the algorithm, the data elements that feed in to the algorithm, the flowchart logic of the algorithm, and the actions to be taken for each endpoint. The format represents both highly structured data for generating consistent descriptions as well as algorithm-based guidance for further workup of radiology findings. The reference implementation interprets such files, allows users to specify values for data elements, and interactively presents the generated text at each endpoint

1.4 Discussion

Radiology practice contains a large number of guidelines and algorithms for workup of specific clinical scenarios and findings. In order for these to be delivered within the context of a radiologist's point-of-care (such as a PACS workstation or a voice recognition/report generation system), this content needs to be specified in a structured format. The proposed format defined can serve as the interface between the creators of radiology reporting guidelines and the vendors of radiology information systems.

1.5 Conclusion

The proposed format for representing radiology guidelines will enable content creators to create tools that vendors can use to extend the commercial tools currently in use.

2 BACKGROUND AND SIGNIFICANCE

Radiologists practice in an extremely broad field, where even a single imaging exam can present significant findings that cross broad system and specialty guidelines. For example, an abdominal CT might present congenital pathology of the hepatobiliary system, traumatic injury to the musculoskeletal system, and infectious disease of the genitourinary tract. The diagnostic radiologist's role is to combine the patient's clinical context with imaging findings and his or her own knowledge base to generate usable diagnostic impressions that help the other providers advance the patient's care. This need to describe a diverse domain of imaging findings and integrate with a broad imaging and clinical

knowledgebase has resulting in a very open-ended radiology practice with the expected high variability in the description of findings, diagnostic impressions, and recommendations for further care found in radiology reports (citations on variability).

In spite of the openness of radiological practice in general, there are specific clinical scenarios where the field has come to consensus on appropriate reporting and management recommendations. The most notable area is breast imaging, where the American College of Radiology (ACR) developed and promulgated the Breast Imaging Reporting and Data System (BI-RADS) in response to the need for more standardized reporting of mammography, partly in response to the United States federal Mammography Quality Standards Act{Joe:2014ge}. The BI-RADS system includes a lexicon for standardized description of imaging findings as well as standardized recommendations for follow-up imaging and other clinical management. Backed by the requirements of federal law, the system has resulted in a much lower degree of variability in the reporting of breast imaging findings and recommendations for management.

Guidelines for reporting specific clinical scenarios have been developed in other areas of radiology. These guidelines are created by many different actors, and include the ACR white papers on incidental findings on body CT exams{Berland:2010hn}, the Fleischner Society for Thoracic Radiology guidelines for management of pulmonary nodules{MacMahon:2005do}, and the Society of Radiologists in Ultrasound guidelines on the management of thyroid nodules detected at ultrasound{Frates:2005is}. These guidelines typically take the form of white papers which lay out best practices for what features radiologists should include in their descriptions of particular findings when they occur. Also like BI-RADS, the guidelines typically describe recommendations for next clinical steps, especially for recommended follow-up imaging exams. However, unlike in breast imaging, in spite of these consensus guidelines, radiologists' reports continue to display a high degree of variability in these areas (citations on variability in the face of guidelines).

Numerous explanations have been offered for this ongoing widespread variation in radiologist practice from published guidelines. However, one important contributing factor is that thus far, efforts to integrate the guidelines with the radiologist clinical workflow have been limited. In other areas of clinical medicine, "best practice alerts" and other methods of integrating clinical decision support into the physician workflow in the electronic medical record (EMR) have been show to improve compliance with guidelines (citations on CDS in EMRs). However, these systems would be unlikely to be effective in altering radiologist practice because the EMR is not typically central to the radiologist workflow. Instead, we have proposed to develop a computer-assisted reporting/decision support (CAR/DS) framework to integrate such guidelines into voice recognition software (VRS), the typical radiologist's tool for report generation (do we need a citation on the penetration of voice recognition among radiologists in the US?).

The proposed CAR/DS framework would allow guideline-creating groups like the ACR to define clinical guidelines in a standard, open definition language. This definition language would permit the specification of the data elements involved in the guideline, the branching logic by which decisions are made, and a description of the expected output and actions. Commercial voice recognition software products could then include an engine that would load these guideline definitions at run-time and use them to mediate an interaction with the radiologist in the given clinical situation. For example, the ACR might create a guideline definition for the workup of an incidentally discovered adrenal nodule seen on a CT examination (cite Berland 2010). Once the guideline has been properly encoded in the CAR/DS format, a VRS product could load that definition. When a radiologist encounters an adrenal nodule and

begins to describe it, the VRS would recognize that the loaded module might be of assistance to the radiologist, and prompt him or her that guidance is available. If the radiologist agrees to accept help, the VRS could then prompt the radiologists to provide the necessary descriptions of the adrenal nodule (e.g., size, presence of macroscopic fat, stability from prior imaging exams). Using the defined branching logic, the VRS could then generate appropriate standard language and recommended imaging follow-up for the adrenal nodule, and insert the generated text into appropriate places in the radiologist's report draft. This integration of the clinical guideline directly into the radiologist report generation process would lead to much improved standardization of radiologist descriptions for this clinical scenario, and much higher compliance with recommended approaches to follow-up imaging.

In this paper, we propose an initial version for the CAR/DS guideline definition language. We also offer a brief description of a reference implementation of a tool that can enact the described user interaction, logic, and report text generation. We believe that forward-looking VRS vendors will deliver value to their customers by integrating into their products an implementation of CAR/DS based on this proposed guideline definition language. We hope that this initial effort can serve as the basis for a community-owned open standard for guideline definition which vendors will embrace in their products.

3 MATERIALS AND METHODS

Because CAR/DS guideline definitions are expected to be structured, machine-readable documents, Extensible Markup Language (XML) with a defined schema was chosen as the default base file format (cite http://www.w3.org/TR/xml/). An XML schema allows guidelines to be expressed as XML which can be validated using appropriate software tools. Though validity does not guarantee correct functioning of a guideline definition, an invalid guideline is very unlikely to perform correctly. In addition, a schema allows for creation of guideline definition in an XML editor which can prompt a user with which elements and attributes are allowed in context, which greatly simplifies the guideline authoring process. The RelaxNG Compact Syntax (RNC) for defining XML schemas was chosen as the syntax for expressing the guideline definition language schema (citation of van der Vlist's book *RELAX NG*, O'Reilly, 2003). RNC has the advantages of compactness and ease of use while freely available tools allow transforming the schema into the more widely-supported XML Schema Definition syntax.

To enable testing of CAR/DS definitions, reference software was created. This program loads guideline definition files, enacts the specified user interaction, processes the user-entered data according to the given logic, and generates and presents the defined report text. The software consists of a web application, where both server-side and client side components are written using the JavaScript programming language. The server-side component can be run on most computing platforms using the Node.js runtime environment (cite http://nodejs.org); the client-side application can be run in most modern web browsers, and uses a variety of common client-side frameworks, most notably the Knockout data-binding framework and the Bootstrap front-end framework (cite http://knockoutjs.com and http://getbootstrap.com).

The schema and the software implementation are freely available for public usage via the Github code-sharing service at https://github.com/acrscm/decision-support-schema and https://github.com/acrscm/acr-assist-simulator, respectively.

4 RESULTS

A CAR/DS guideline must define the data elements that will serve as its inputs and outputs. It must also define the rules by which inputs are turned into outputs via branching logic, and for possible outputs, specify templates for report text to be inserted. Therefore, at the highest level, a CAR/DS guideline definition contains descriptive metadata, data element definitions, a flowchart-like logic tree, and a set of templates associated with the possible endpoints.

4.1 Metadata

The metadata section contains general information about a CAR/DS guideline which may or may not be used by any given implementation. Refer the technical white paper document (https://github.com/acrscm/acr-assist-decision-support-schema/blob/master/ACR%20Assist%20Schema%20Technical%20White%20paper.pdf) for more information about metadata section. The first portion of the metadata section of the definition file is an information section that contains a short text label for the guideline; a text-based description of the guideline; and contact information for relevant authors of the document. Figures and files such as the primary flowchart or the PDF of the actual reference can also be specified in this subsection. In addition, there are citations to relevant articles from the literature, ideally including the primary published source for the guidelines. This section also contains links to other ontologies; for example, relevant SNOMED or RadLex data entries for possible diagnoses arrived at using the guideline might be listed.

The metadata section also contains information about which exams and patients the guideline would be relevant to. For example, an algorithm might specify that it is intended for CT exams of the abdomen body region (where these could be specified according to a standard coding scheme, such as RadLex Playbook or ACR Common). In addition, appropriate sexes (male, female, or both) and an appropriate minimum and maximum age can be specified in this portion of the metadata. This can allow an implementing system to not offer a guideline which is intended for adult or female patients when a user is creating a report on a pediatric or male patient.

Finally, the metadata section provides clues as to how a reporting system might recognize when a user is describing a finding for which the guideline might be applicable. This takes the form of text cues which a report author might insert into the report which an implementing VRS application might use as the cue to offer the guideline to the user. These text cues can be explicit phrases (“adrenal nodule”) or words used in the context of another word (“cyst” used in the context of “kidneys”). Negative phrases can also be specified (“simple hepatic cyst”).

4.2 Data Element Definitions

The data element definitions specify the input values used to drive a decision tree, the constant values, and possibly intermediate or output values associated with an algorithm. Three main types of data elements can be described using the data format: external and fixed values, user-provided data, and results of computation.

The current schema allows the definition of global values, similar to constants in the C programming language that can be referred to elsewhere in the guideline. These are intended to be used to define threshold values or parameters in a linear regression. In keeping with the "don't repeat yourself" principle, these can be specified once in a guideline definition; when thresholds change, the value needs to be updated in a simple location in the definition.
In addition, the schema allows for the specification of basic patient and exam data, such as age, sex, modality, and body part. In this way, a guideline's logic can branch based on a patient's age, offering different recommendations for patients younger and older than 35 years of age. We envision future versions of the schema specifying the inclusion of more complex patient data extracted from an electronic medical record, such as smoking status or history of malignancy.
choica finding), or multiple-choice values (e.g., presence or absence of findings in multiple locations). For each data element, associated data can be specified; whether an element is required for the guideline to return a result, where it should be displayed, how the user should be prompted to reply. In addition, more detailed text and/or images can be included which might be offered by a given implementation as a tooltip. Different data elements can also include more type-specific information, such as the values and textual representations of the different possible choices for a choice-based data element. Simple data element definitions are illustrated in Figure 2 with the reference implementation rendering of that data element as a question. Refer the technical white paper document for more information about data elements section.

Computed data elements, whose values depend on other data, can also be defined. These can be simple Boolean values or more sophisticated textual constructs or arithmetical calculations. These take advantage of the same logic syntax described below for defining the overall logic tree of the definition. There are several uses for these computed elements. One common case is to allow for intermediate values in calculations or logic, to prevent re-specifying the same logic or calculation over and over again. For example, a cancer staging guideline might include a computed element that determines the T-stage for the exam, which can then be used and re-used in the logic tree without having to re-specify the logic for determining T-stage at every decision point that depends on it. Alternatively, a regression model (e.g., the Brock University cancer prediction equation for pulmonary nodules; cite that paper) might be encoded as a computed data element to offer a prediction of an outcome, which could be included. A second common use is to more carefully craft text to be included in a report where the text has complex dependencies on the data inputs.

4.3 Logic Tree

Clinical guidelines are frequently defined as flowchart-like decision trees. For representation in the CAR/DS format, this logic must be formally encoded as a branching structure of binary decision points based on Boolean logic and associated outputs or further decision points; this is illustrated in Figure 3. Refer the technical white paper document for more information about Logic tree section.

The building blocks of the logic structure are decision points and branches, corresponding to the diamond shape and emanating lines on a flowchart. Each decision point contains a series of branches, where a branch is composed of a condition and either a nested decision point or a pointer to a flowchart endpoint. Depending on the current state of the system (i.e., the values of the data elements), one of

the several branches may be followed. The condition is built up using fundamental conditional operators (equals, less than, greater than or equal to, contains) and composite operators (AND, OR, NOT). These conditions allow for comparison of the values of data elements against constants or other data elements. For each decision point, a default branch can be specified without a condition that will be followed if no other branch's condition evaluates to true.

Starting at the outermost decision point, implementing software finds the contained branches, and evaluates the condition of each sequentially. The first branch whose condition evaluates to true is then followed; if the branch leads to an endpoint, that endpoint is the output of the algorithm specified by the logic tree. If the branch contains a nested decision point, that decision point is then evaluated recursively according to the same logic (i.e., the condition of its branches are evaluated sequentially, first branch selected whose condition is true). Implementing software can therefore walk recursively through the defined logic tree for a given set of inputs and eventually arrive at an end point for a given set of inputs. When the state changes (because the system's user inputs new or changed data), the logic tree is re-evaluated to determine the new end point.

In addition to the decision points, branches, conditions, and endpoint references that make up the structure of the logic tree, other elements for user interaction can be specified within the logic tree. Specifically, at each branch, data elements can be specified as "not relevant", which would cue implementing software to either hide or disable the inputs associated with those data elements. Different elements can be specified as required or not within a branch, and individual choices could be specified as being not available along particular branches of the logic tree.

4.4 Endpoint Definitions

Each endpoint of the defined logic tree specifies actions to be taken when user inputs lead to that output. Primarily, these actions consist of pieces of text to be inserted into the report, possibly with input or computed values inserted at appropriate places within the text. However, other kinds of clinical actions, including a structured recommendation for further imaging, can also be defined as part of each endpoint. A sample endpoint definition is illustrated in Figure 4.

An endpoint definition lays out the text to be inserted in a report by an implementing reporting system. Pieces of text can be defined to insert into the findings section of the report, into the impression section, and into a recommendation section (if available). The definition of the text to be inserted is based on a simple template language that permits the insertion of data element values and the conditional insertion of text. In addition, "partial" templates can be defined that act as template sub-routines, and can be re-used in multiple endpoint definitions.

In addition to the text to be inserted into the report, more structured information can be included in the endpoint for the potential use of implementing software. One of the most important types of structure information is a structured recommendation, especially a structured recommendation for additional imaging. This data structure includes encoded information on the reason for the recommendation, a specification of the imaging exam being recommended, acceptable alternatives to the preferred exam, and the timeframe during which the recommended exam should be obtained (an example is shown in Figure 4).

4.5 User Interaction

The reference implementation of the software allows users to interact with a CAR/DS guideline so they can test how different inputs lead to different outputs. Loading and running the reference implementation program starts a web application which users can access using a web browser (see Figure 5). This application searches a local directory for XML files containing CAR/DS guidelines and creates a menu of them for the user to select. When the user selects a module, they are presented with a form of inputs corresponding to the defined data elements. When they provide values into the form, the software executes the logic tree for the inputted values and determines an endpoint. For that endpoint, the software fills in the relevant data into the templates and generates output text, which is shown to the user. As the user changes the inputs, the output endpoint and text are updated interactively.

Newly developed module definition files can be added to a standard directory. Existing XML files can be edited in place. After editing or adding a definition file, reloading the web browser session will cause the application to reflect the new data.

5 DISCUSSION

We have presented and made available a new definition schema along with an example file and a reference implementation of software that can execute the defined guidelines for computer assisted reporting/decision support at the point of care for radiologists. Expert groups such as the ACR can create guideline definitions based on their existing content, such as the incidental findings white papers using this new format. Vendors of VRS systems can then implement systems which allow radiologists to use this content at the point of care when they encounter relevant clinical scenarios. We hope that the ACR and many other groups will include encoding guidelines into this new format as part of their content development process.

The definition serves to separate the content of the guideline from the implementation. It is up to individual VRS vendors to decide how to actually implement the user interaction in the most useful possible way.

The freely available definition software and the reference implementation combine to create an authoring environment for developing new CAR/DS modules. Potential authors can download the definition file and use an XML editor of their choice to author a new module, possibly using an existing module as a template. They can then download and run the reference implementation software to test whether the code is correct and the desired output is generated for each possible set of inputs.

We hope this will serve as the starting point for an ongoing process. The imaging informatics community can take ownership of this definition format and use it to express the kinds of content they believe that should be created for radiologists at the point of care. Vendors can then choose to implement subsets of the functionality and compete to provide the richest implementation of the community-specified functionality.

One important area for improvement would rely on the existence of a registry of common data elements. If and when such a registry exists, moving structured data into and out of radiology reports will become much more straightforward, and the value of the radiology report is increased.

Another important extension will be to specify how the structure data to be generated from the use of such algorithms is to be made available.

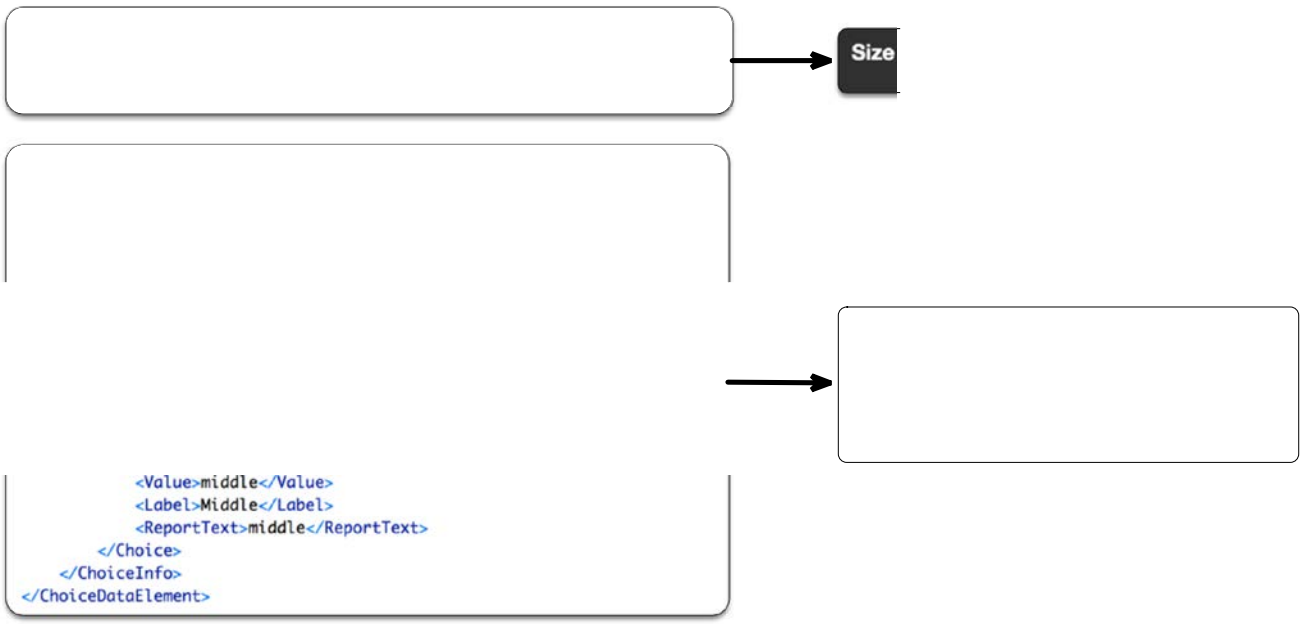
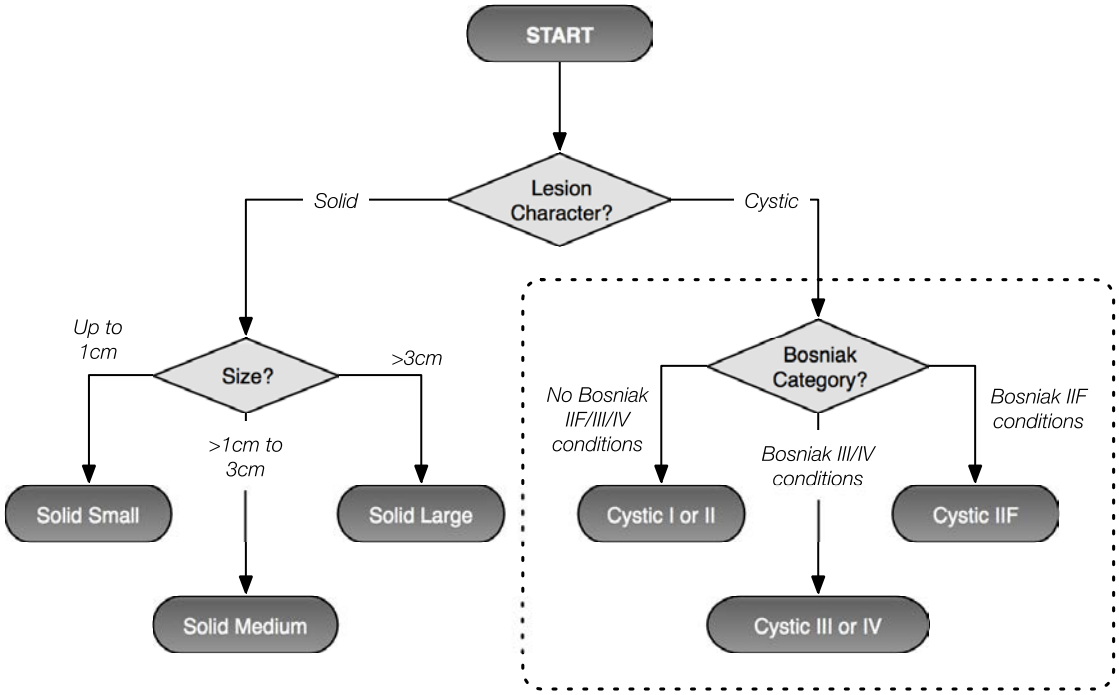


Figure 1. Data elements defined in the guideline definition language and rendered using the reference application. The first pair demonstrates a data element designed to request and holds a number value entered by the end user. The second demonstrates a choice data element, where the end user selects one of several defined values. Note that the data element definitions include user interface elements, such as the text of the question label and choices and processing data such as whether a data element is required or not.




```
<DecisionPoint>
    <Label>Bosniak criteria?</Label>
    <Branch>
        <Label>Bosniak III/IV</Label>
        <OrCondition>
            <EqualCondition DataElementId="enhancement" ComparisonValue="ofSeptaeSolidPart"/>
            <EqualCondition DataElementId="cystSolidPart" ComparisonValue="true"/>
            <EqualCondition DataElementId="cystSeptae" ComparisonValue="thickOrIrregular"/>
            <EqualCondition DataElementId="cystWalls" ComparisonValue="irregularOrThick"/>
        </OrCondition>
        <EndPointRef EndPointId="cysticBosniak3or4"/>
    </Branch>
    <Branch>
        <Label>Bosniak IIF</Label>
        <OrCondition>
            <EqualCondition DataElementId="cystCalcifications" ComparisonValue="thickOrNodular"/>
            <EqualCondition DataElementId="cystSeptae" ComparisonValue="manyThin"/>
            <AndCondition>
                <GreaterThanCondition DataElementId="size" ComparisonValue="30"/>
                <EqualCondition DataElementId="cystDensity" ComparisonValue="hyperdense"/>
            </AndCondition>
        </OrCondition>
        <EndPointRef EndPointId="cysticBosniak2f"/>
    </Branch>
    <DefaultBranch>
        <Label>Bosniak I/II</Label>
        <EndPointRef EndPointId="cysticBosniak1or2"/>
    </DefaultBranch>
</DecisionPoint>
```

Figure 2. Logic is represented in the guideline definition language. Above, the flowchart represents the logic for a guideline outlining management of an incidental adrenal mass as a series of decision points,

branches, and end points. Below, the XML represents the decision points, branches and end points within the dashed line on the flowchart.

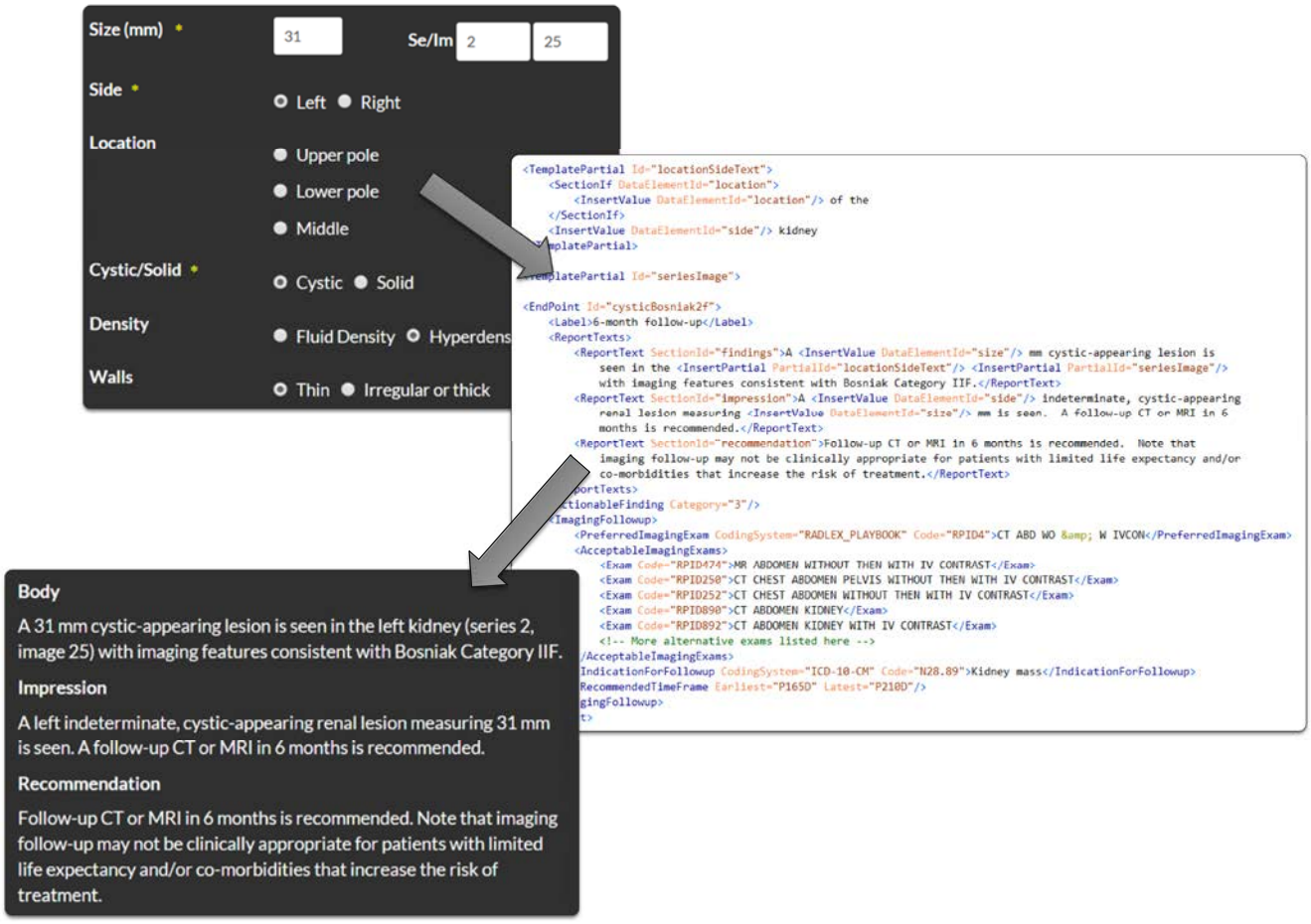


Figure 3. End point definitions specify text to be inserted into the report. At upper left, a radiologist has entered a series of inputs that result in an endpoint being selected. For the given end point, text templates are provided which specify the text to be included in the report, which can include findings, impression, and recommendation text. The templates can insert values from the inputs to improve the applicability of the text. An end point can also define other actions to be taken, such as activation of a critical results management system or a recommendation for follow-up imaging.

Figure 4. Reference implementation software loads CAR/DS definition XML files and allows the user to interact with them. Once the user has chosen a module to interact with, the inputs for relevant data elements are shown at the left. As the user changes the values, the output for the report text is updated on the right.