Michael Mischke Reeds
Chiron Diagnostics
1401 Harbor Bay Parkway
Alameda, CA 94502

September 1, 1997

CORBAmed Technology Desk
Object Management Group Inc.
Framingham Corporate Center
492 Old Connecticut Lpath
Framingham, MA 01710-4568
USA

Re: CORBAmed RFI 3 Response: Clinical Decision Support

Dear Decision support SIG, CORBAmed and OMG members:

Enclosed with this letter is the response of Chiron Diagnostics to the CORBAmed Request for Information. Our response is relevant to the following categories of the RFI:

- Care plan management
- Diagnostic assistance
- Drug management

Chiron is a diversified biotechnology company that has developed breakthrough medical knowledge in many areas, including the discovery of the Hepatitis C virus. The informatics group of Chiron Diagnostics is working to ensure that medical knowledge can be represented in ways that enable advanced decision support systems to be developed. We send this brief response to the RFI to share with you some of our ideas about the component architecture for clinical decision support.

Chiron Diagnostics would possibly respond to a future decision support RFP in the focus areas of care plan management, drug management, data mining, and clinical research. We would prefer to respond to a more focused RFP rather than a general decision support RFP and suggest that, at the least, a distinction between clinical decision support (often but not always real-time) on the one hand and data mining/data warehousing (prospective or retrospective) on the other.

I will be in Dublin for the OMG meeting in September and would appreciate the opportunity to present and further amplify the ideas contained in the following response.

Sincerely,

Michael Mischke Reeds
Director, Knowledge & Software Engineering
Chiron Diagnostics

CORBAmed RFI 3 Response:

Clinical Decision Support

Submitting organization:
Chiron Diagnostics

Contact person:
Michael Mischke Reeds
Director, Knowledge- and Software-Engineering
Chiron Diagnostics
1401 Harbor Bay Parkway
Alameda, CA 94502

phone: (510) 337-8352
fax: (510) 748-8440
email: michael.reeds@cc.chiron.com

Types of clinical decision support

The term *clinical decision support* has a range of meanings: from simple warnings that a potassium lab result is abnormally low, to complex guidelines that model chronic disease management. The CORBAmed decision support RFP needs to encompass at least the following:

- Simple *alerts and reminders*, such as the low potassium dialog already mentioned, or a warning that a proposed drug therapy poses a risk due to a known adverse interaction with another drug already prescribed.
- *Clinical pathways*, a sequence of tasks such as the procedures that are conducted each day following an appendectomy. This allows a care delivery system to standardize medical practice around prescribed clinical pathways. While this is undoubtedly useful in cases in which the medical situation is relatively simple, clinical pathways are limited in their ability to support exceptions or decisions that go beyond the routine. Clinical pathways are similar to traditional workflow systems and have similar limitations.
- *Protocols*, a precisely specified procedure, which incorporates strict criteria for executing tasks and making decisions. A typical protocol consists of eligibility criteria, a sequence of tasks, which may be specified in the form of a flowchart, and conditions that would require a patient to be removed from the protocol.
- *Guidelines*, a description of clinical best practice which is intended to be interpreted by the healthcare professional according to patient needs and clinical circumstances. Guidelines provide a general structure for defining tasks that may involve decisions between alternative treatments, treatments that may run concurrently, and interactions with other guidelines. This is the most complex of all clinical decision support cases.
- *Problem-solving components* that perform well-defined tasks, typically to inform a specific clinical decision or to support higher-level support functions such as reminders, alerts, protocols or guidelines. For example, medical researchers investigating breast cancer may have discovered statistical relationships between some of the determinants of survival following surgical removal of a tumor. So they create an application in which the user inputs values for age, menopausal status, estrogen receptor status, S-phase fraction, etc. and the application returns the likelihood of disease-free survival following removal of a local tumor. The core of this application is a predictive model, which if packaged as a problem-solving component, could be reused in another application. Another example of a problem-solving component takes as input a standard clinical protocol description and relevant patient data and generates as output the qualitative likelihood that a patient is eligible for the given protocol.

Clearly there is a broad range encompassed in clinical decision support, and an RFP that focused only on the above, to the exclusion of datamining and related retrospective activities, would be sufficiently rich to generate useful discussion and, more importantly, result in interfaces that can and will be implemented.

Adding in the additional complexity of data mining to the RFP might well lead to responses of great generality.

Use cases

We will not provide detailed use cases here for the types of clinical decision support just listed. However, even a high-level description of two illustrative examples illustrates the significant differences between these types, differences which need to be captured in the RFP requirements.

1) Alerts and reminders might require interactions between the following components:

- A decision support component that can interpret rules (or other forms of knowledge representation) and apply them to a clinical case at hand, issuing appropriate reminders or alerts.
- A patient data repository storing the clinical data that is fed into the decision support component. It is critical that this data is coded using a controlled vocabulary (e.g. SNOMED, ICD-9, UMLS). If the data is not so coded, then there is little chance that it will enable a decision support system to function.
- A lexicon query service that allows the decision support component to map from the coding schemes of the patient data repository to whatever coding is reflected in the knowledge representation being applied (e.g. warnings about drug interactions require being able to interpret drug inputs).
- A GUI application, either the DSS component itself or an EMR, that displays the alert window. The message from a DSS to an EMR to display a warning or reminder would be very simple in structure, since all that is needed is to pass a text string, perhaps specify a window type, and return a user response (which button was clicked in the dialog).

2) Protocols and guidelines require much more complex interactions such as the following:

- A decision support component that can interpret the standard description of a protocol or guideline and apply it to a clinical case at hand. (Note that the knowledge representation required here is much more complex than that required for alerts and reminders.) As will be discussed below, simple rule-based approaches are very difficult, if not impossible, to scale to support real-world protocols and guidelines.
- A patient data repository as above, though again here the nature of the queries may be of a different type. For example, a protocol may have among its eligibility criteria that a patient not have had more than two episodes of anemia in the past six months, requiring the ability to query the patient data repository in the form of temporal abstractions (since anemia is an abstraction, not a simple atomic data element). These abstractions may themselves require additional problem-solving components.
- A lexicon query service as above, in order to mediate between any of the components communicating coded datea with one another.
- A medication knowledge base may be needed in order to calculate drug dosages or to flag potentially harmful drug interactions. Since this is likely to be a standard component, it would be useful to have an RFP for medication query services.
- Problem-solving components may be needed to predict the likelihood of response to a treatment, to determine whether a patient is eligible for a clinical trial, or to lay out a treatment plan for a specific patient based on a guideline specification. There needs to be a standard interface by which problem-solving components advertise their capabilities, and we recommend that this be added to the RFP requirements. Chiron has specific ideas here about such an interface that we are eager to submit as an RFP, perhaps as a specialized Trader service.
- An EMR system may be needed to act on the decision support output. For example a decision support system may recommend changing medication. If the doctor and patient agree with the recommendation, a message needs to be sent to an EMR's medication order system, with subsequent updates to a patient data repository. In the opposite direction, the EMR may query the DSS for a recommendation. We can imagine a rich set of messages between a DSS and an EMR, and believe it to be important that it not be assumed that the DSS functionality actually be embedded within an EMR system.

Issues raised

Even this cursory description of use cases suggests a number of issues which should be considered in the drafting of the RFP:

1) Degree of coupling between object interfaces and knowledge representation formats

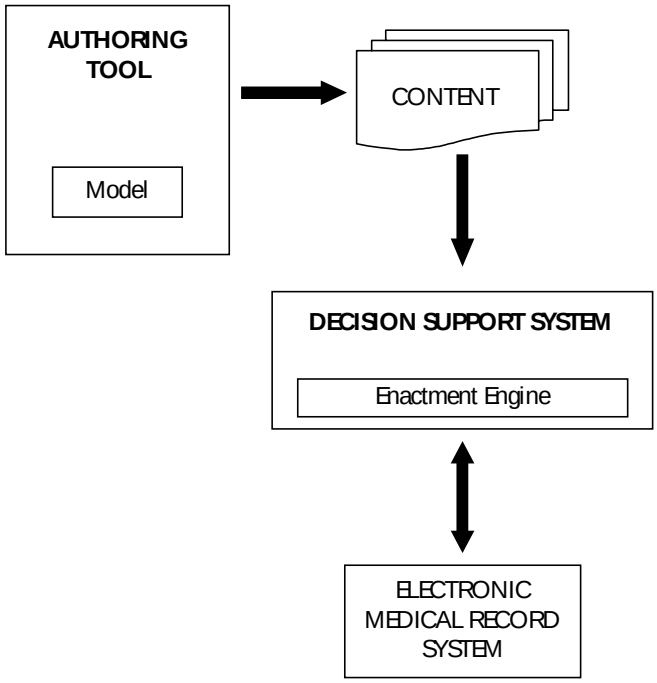
While the CORBAmed RFI focuses on the interfaces between the decision support component(s) and other components, it is difficult to specify interface requirements without having some understanding of the knowledge representation formats being interpreted and applied in the decision support system. We agree with Harm Scherpbier's statement in the Arden syntax submission to this RFI: "Although technically one can separate the inside (knowledge representation) from the outside interfaces of a DSS, they are tightly coupled. The knowledge representation drives the requirements for the interfaces, and the interfaces determine the capabilities allowed inside the DSS." Clearly it is difficult to send a message to a DSS to apply a rule to a patient if the DSS doesn't know what a rule is (or worse if there are many ways to interpret rules). Similarly it is difficult to apply a protocol or guideline if there is not agreement on standard representations for these knowledge representation formats.

It is by no means clear that the same knowledge representation approach will work well in all the clinical decision support types listed above. For example, rule-based approaches such as Arden syntax are probably sufficiently expressive to support simple applications of reminders and alerts. However rule-based approaches are significantly limited in ways that currently make them unable to support protocols and guidelines of real-world complexity. It is not our intention here to criticize first-generation rule-based expert systems. Rather, it is to raise the more basic issues: 1) that defining interfaces between DSS and other objects requires facing the knowledge representation standard issues; and 2) that multiple knowledge representation approaches may be necessary to support the various types of clinical decision support that users desire. CORBAmed may not want to get into the area of knowledge representation standards, but if it attempts to define interfaces in ignorance of these issues, the likelihood of successful implementation is surely diminished.

2) Supporting reuse of decision support components and domain knowledge content

Researchers in both knowledge-based systems and in software engineering have looked to *reuse* as a methodology for reducing the high cost of software development and maintenance. With a reuse approach to software construction, developers adapt existing software components at a fraction of the cost of developing a system from scratch. If the CORBAmed RFP can encourage and enable significant reuse, both of components and of knowledge representation content, this will be of great benefit to the entire healthcare industry, not to mention the broader benefit to software development in other domains. Development and maintenance of knowledge-based systems without reuse is known to be expensive. Originally, first-generation knowledge-based systems consisted of an inference engine and a knowledge base that included both facts about the domain and rules that controlled the processing of those facts. Unfortunately, these systems did not scale well to large knowledge bases, as the set of rules and facts quickly became unwieldy and difficult to maintain. In response to this problem, researchers isolated knowledge about the *process* used to solve some problem from knowledge of specific facts about a particular domain. Thus, second-generation knowledge-based systems are composed of two classes of large-grained components: (1) problem-solving components and (2) knowledge bases used by the problem-solving components. This enables reuse of both elements: problem-solving components can be used in multiple domains, and domain knowledge encoded in knowledge bases can be applied in multiple problem contexts.

The following figure illustrates some of the reuse and sharing potential we see being enabled by the decision support RFP:



The process begins with a domain expert using an authoring tool which employs an implicit or explicit model, i.e. some way of representing domain knowledge. A domain expert uses the authoring tool to generate content, e.g. a protocol expressed in a particular protocol description format. That content is published or shared, stored in a knowledge base, and then can be "played" in an enactment engine embedded within a decision support system. The enactment engine probably makes use of problem-solving components (not pictured). The DSS then interacts with an EMR to access patient data, to order procedures, and so on.

It would be of great value to the health care industry if vendors and customers could develop, share, and reuse both the knowledge representation content (e.g. rules that drive alerts or descriptions of clinical trial protocols) and the problem-solving components that make up the decision support system. This requires standard formats for medical knowledge content and standard interfaces for problem-solving components.

3) Separating content knowledge from control knowledge

The point about first-generation systems not drawing a sufficient distinction between content knowledge and control knowledge deserves more amplification, since some important lessons were learned in the 1980s at considerable cost. The problems of unpredictable behavior in rule-based representations,[1] and the difficulties of long-term maintenance of large knowledge bases encoded as rules,[2] are widely recognized. Contemporary methodologies for the development of knowledge-based systems thus de-emphasize the use of traditional rule-based frameworks, and instead build upon software modules that generate well-defined problem-solving behaviors.[3]

In second-generation systems, cleaner separation between *content knowledge* (e.g., knowledge about antiretroviral medications) and *control knowledge* (e.g., the solution strategies encoded in problem-solving methods) leads to systems that are easier to build and maintain. To illustrate this distinction, if one wanted

---

[1] Heckerman DE, Horvitz EJ. *The myth of modularity in rule-based systems for reasoning with uncertainty*. In: Lemmer JF, Kanal LN, eds. *Uncertainty in Artificial Intelligence*. Amsterdam: North Holland, 1988; **2**:23–34.

[2] Clancey WJ. The epistemology of a rule-based expert system: A framework for explanation. *Artificial Intelligence*. 1983; **20**:215–251.

[3] David J-M, Krivine J-P, Simmons R., eds. *Second Generation Expert Systems*. Berlin: Springer-Verlag; 1993.

to develop care guidelines in the area of HIV/AIDS, one would first specify a domain model that defines the terms and relationships that will be referenced in the guideline—for example: clinical trials, drug regimens, medication prescriptions, laboratory tests, and relationships among them. We refer to the domain model as an *ontology* of the application area, following its usage in the literature.[4] [5] [6]

The ontology does not indicate *which* drug regimens might be used in a particular guideline; the ontology specifies only the classes of concepts that are relevant in an application area, rather than the instances of those classes. Then when a guideline author writes a guideline, using a knowledge acquisition tool, he references the concepts defined in the ontology. Ideally, all components of the DSS interoperate with a single ontology, so changes to the domain ontology automatically propagate through all the system components. This is clearly much easier than visually inspecting every module of a large system searching for possible references to the modified domain knowledge, which is what would be necessary with a classic rule-based system. Types of changes that are readily captured by changes to the domain ontology include the addition of new tests, new clinical interventions, and new patient description data.

In many ways, an ontology can be viewed as the equivalent of a *schema* in database terms. Whereas controlled terminologies such as ICD provide an enumeration of instances of clinical descriptors—with only implicit representation of an organizing framework for the classes of those descriptors—ontologies emphasize the organizing framework, at the expense of being exhaustive about listing possible instances of the classes of concepts. We believe that clinical decision support systems, in order to be safe, scaleable and maintainable, need to be based on knowledge representation schemes that reference standard clinical vocabularies (the lexical layer) as well as standard domain ontologies (the semantic layer). As a result, we encourage the DSS RFP to address issues of syntax, lexicon, and ontology as a cohesive whole. To be effective, this will require networking with the Agent Task Force, OOAD, BOD-TF, and other OMG committees, since these issues are clearly of relevance beyond the domain of medicine.

4) The role of explanation

One of the areas where decision support systems may be most useful is when doctor and patient are facing a difficult decision, such as changing antiretrovial therapy for an AIDS patient, where the stakes are very high and the relevant data can be complex. A decision support system built upon an HIV domain ontology would "know" about combination therapies for HIV and about cross-resistance between different antiretroviral drugs. The system could therefore explain that drug A is recommended over drug B because there is known cross-resistance between drug B and the (failed) combination therapy the patient is already taking, resulting in a low probability of successful response for drug B. We believe that a clinical decision support system based on an explicit separation between domain knowledge and control knowledge is much better suited to providing meaningful and useful explanations. This is in contrast to a first-generation rule-based system that can offer only the most meager of explanations, such as listing the rules that have been fired. We recommend that the RFP include a requirement for explanation functionality.

Related work in the OMG

The medical domain is especially complex, and nowhere is this more apparent than in clinical decision support. We believe that the issues raised above have relevance much beyond medicine. In particular when we talk about the separation of content knowledge from control knowledge and the suggestion that both lexical and semantic layers are necessary, clearly this has implications for many other groups within the OMG. The following is a short list of some of the related work that is relevant to the DSS RFP:

[4] Regoczei S, Plantinga EPO. Creating the domain of discourse: Ontology and inventory. *International Journal of Man-Machine Studies.* 1987; **27**:235–250.

[5] Guarino N. Concepts, attributes, and arbitrary relations: Some linguistic and ontological criteria for structuring knowledge bases. *Data and Knowledge Engineering*. 1992; **8**:249–261.

[6] van Heijst G, Falasconi A, Abu-Hanna G, et al. A case study in ontology library construction. *Artificial Intelligence in Medicine*. 1995; **7**:227–255.

Data Interchange	CF – RFP 3
Semantics Work Group	
Analysis & Design	OA&D RFP 1
Meta Object Facility	CFTF RFP 5
Business Object Facility	BODTF RFP 1
Agent Facilities	CF RFP 3
Rule Management	CF RFP 8
Work Flow Management	BODTF RFP 2
Lexicon Query	CORBAmed RFP
Patient Identification Service	CORBAmed RFP