Should GLIF define a standard file format?

Why should GLIF define a standard file format and not just a standard interface for a guideline server?

If they can do it, why can't we?

The DICOM standard for medical images supports a messaging standard for communication between DICOM compliant devices/applications and a physical file format for persistence of images. Also consider the web (HTML and HTTPD) as an analogous system: HTML is the file format, HTTPD is the server. One does not need the server to open up the file.

Arguably, a guideline is more complex than a DICOM image or a web page. Nonetheless, it is not impossible to create a file format for a guideline. In fact, there are formats that are used today to store the guidelines in Protege. The issues are, perhaps, of what the specifications of the format should be such as how many guidelines can a file contain, what is a unique id for a guideline and the objects in a file, etc.

Freedom from architectural constraints

If we say that guidelines will only be accessible through interfaces, we are forcing some kinds of architectural constraints (client/server) on applications. Even the simplest of applications like a viewer or a validator would have to be written as a client-server app.

The less the merrier

Further, not every user will have the capability to be on a network to download guidelines as they are published, say, on ACP's guideline server. Perhaps, ACP might ship CD-ROMs every quarter to its subscribers who are Internet-challenged. In what file format would ACP save the guidelines? One may argue that it doesn't matter, since some server software installed on the subscriber's PC will be able to read whatever format ACP sends the guideline in. For one, potential users such as ACP would like us to specify a file format. They wouldn't want to invent one. Secondly, our network-deprived subscriber may also get guidelines from the AMA. Now, if AMA's CD-ROMs contained guidelines in some other format, someone would have to write another filter/reader for the server.

If we do not define a standard format, everyone else will and may lead to chaos

2. What is the scope of identifiers of various objects in a file? Are they unique to a single file?
3. Should the format of the file be RDF or XML?

Scenario

- I have a knowledge base/server containing all the protocols that are active in my oncology clinic.
- each protocol is stored in an individual file.
- each visit in the protocol is encoded as a (sub)guideline and contained in the same file as the protocol.
- common toxicity criteria, which are shared by many protocols, are stored in a separate file on my server.
- I have a protocol execution application that is independent of any single protocol.

When a patient is put on breast cancer protocol X-123, how would the application know which guideline from the many guideline objects in the X123.glif file should it look for as a starting point? Would that information be stored in a separate application-specific knowledge base?

Aziz Boxwala