Tech Focus: Representing Idiosyncratic Data for NIEM IEPDs

By: Bill Blondeau, WIJIS Data Architect, Wisconsin Office of Justice Assistance

Editor's Note: NIEM News readers represent a broad audience of decision makers, stakeholders, practitioners, and users. Tech Focus features will cover topics that are more technically oriented for experienced National Information Exchange Model (NIEM) users and programmers. This month's Tech Focus is about the Wisconsin Justice Information Sharing (WIJIS) Program. This Tech Focus article offers an overview of the issues faced by WIJIS as well as a link to a much more comprehensive article. Special thanks to Bill Blondeau for providing both versions of this article for NIEM News readers.

Real-world situations sometimes yield data that are too irregular to model efficiently via the **Unified Modeling Language™** (UML) class diagrams stipulated by the Information Exchange Package Documentation (IEPD) lifecycle process. WIJIS recently encountered such a situation and found an effective and generally applicable solution that is well supported by NIEM—external unconstrained Resource Description Framework (RDF) attachments defined at the Instance level, rather than the Class level.

When You Cannot Completely Model Your Problem Space

The recommended definitional artifact of information exchange in NIEM is the IEPD. The IEPD specification defines both a structure for IEPD representation and a process (the IEPD Lifecycle) by which IEPDs are conceived, designed, built, and, over time, updated.

The "Information Exchange Mapping and Data Modeling" stage in the IEPD Lifecycle process prescribes a very powerful and mature information representation—the UML Class Diagram. Class diagrams are efficient, expressive, easily understood, and provide good guidance when creating software artifacts to represent static structures, such as information exchange messages.

Class diagrams, however, were never designed to represent information structures that are not completely understood in advance. This can be a nontrivial issue for certain kinds of problem domains. Sometimes the data really are highly idiosyncratic or maybe it is just very difficult, for practical reasons, to assemble a high-confidence sample set of those data. Either way, the practical result is the same—you do not expect your model to hold up in the real world. You expect that new, unprecedented data will come along, in the course of real operations, and break it.

In Wisconsin, the WIJIS program ran into that problem in the course of developing a NIEM-conformant exchange of drug investigation information. This work was funded in part by the Bureau of Justice Assistance, through the National Governors Association's NIEM Policy

Academy project. As a spinoff to that project, WIJIS has developed a suggestion for what seems to be a satisfactory approach to this problem—one that requires absolutely no changes to the IEPD definition or process and no changes in the structure of the implementation of the NIEM model. In fact, this approach is only feasible in the first place because the NIEM design team made some very good fundamental decisions.

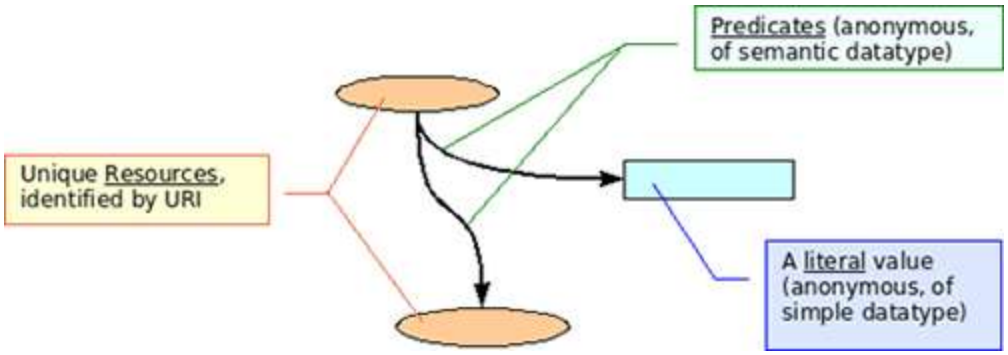
NIEM Information Exchange Packages (IEPs) are the foundation of NIEM's business value as an enabler for nationwide information interoperability and sharing. Broad adoption with collaborative sharing of adoption and use experience, best practices, lessons learned, IEP Documentation (IEPD) reuse, and cooperative development IEPDs for nationwide information exchange are critical for achieving the full potential of NIEM. Since the release of NIEM 2.0, there has been a groundswell of IEP development. There are now more than 70 NIEM IEPs registered in the GJXDM/NIEM IEP Documentation (IEPD) Clearinghouse (www.it.ojp.gov/iepd/). In addition, the DHS has more than 50 IEPDs, most of which are in development or operational.

A much more detailed and technical description of some of the technical and procedural characteristics of this proposed solution is available at http://wijiscommons.org/articles/drug_case_rdf/. Three things to note about that article:

- It is currently in peer review and early "Request for Comment" status, which means we can and will repudiate or deny anything later if we want. (It also means, of course, that substantive comments are most welcome.)
- WIJIS has not implemented this scheme at all, and the devil is always in the details. Logically, this should work beautifully. Practical implementation in the real world could be another matter entirely.
- The article describes how we were led to the question of representing irregular data via narcotics intelligence practices. This should not be taken to imply that we found intelligence data inherently difficult to characterize. The particular situation we encountered was simply that the potential problem domain included data maintained in numerous ad-hoc systems, with no standardization of content or structure, and that the requirements for operational secrecy were quite high. Putting together an adequate set of sample data would have been tough and well beyond our resources or those of the investigators with whom we worked.

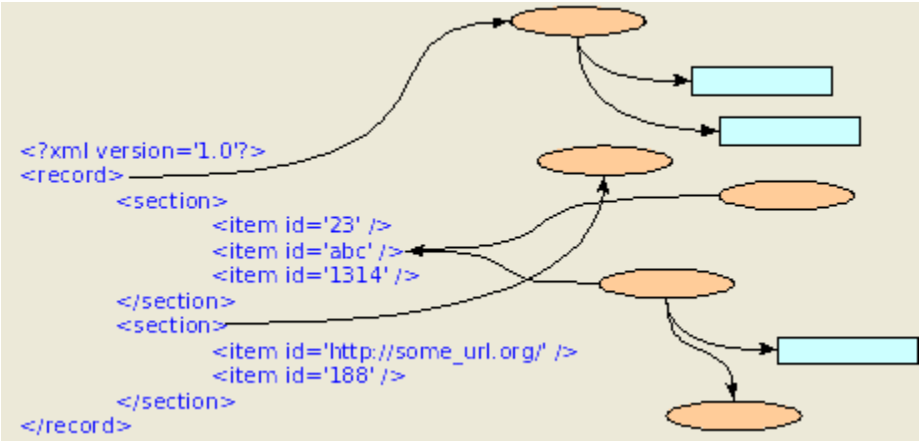
External RDF Attached to IEPD-Conformant Instances

WIJIS's proposed approach uses the W3C's RDF language, which is highly flexible and easily adapted at any time, to represent unruly instance data. RDF is all about associations between items. In fact, RDF is a graphical, rather than text-based, language; it depicts the data as a diagram of links and items, in which links are associations (called "predicates") of known types, and items are either unique Resources or literal values:



Because RDF's link types are identified by URI, it is very flexible. If some instance data does not seem to be expressible by any known RDF association definition, it is easy enough to invent a new association type on the fly and use it. In this proposed technique, the RDF is therefore unconstrained by any preconceived restrictions.

The RDF is attached externally to the IEPD data by a convention for assigning URIs to the individual data items in the instance. (If you can assign something a URI, you can write RDF statements about it.) Conceptually, that works something like this:



In this diagram, the RDF items are linked to individual parts of the XML document.

Well, that is the concept. What would an actual serialized RDF document connected to the XML look like? There are a couple of ways of doing it, and discussing them is perhaps overly technical for this article. But they are not hard to do—RDF serialization is a well-understood practice.

One important note—this has nothing whatsoever to do with implementing the NIEM model in RDF. That is a very large (and praiseworthy) effort that would result in the same familiar NIEM 2.0 model but written in RDF instead of the XML-centric DOM. That is all NIEM—just a different way of expressing the data. This current proposal, on the other hand, is much more modest—nothing but a suggested technique for attaching non-NIEM (and non-any-other-data-

standard) information to an IEPD-compliant instance message—attaching it to the "outside," you might say.

Why NIEM Works

The reasons for this are fairly technical—sort of inside baseball for abstract data designers—and are discussed in the referenced article. For present purposes, it is sufficient to say that NIEM:

- was defined as an *abstract*, rather than an XML, model
- was built to be RDF-compatible from the beginning

Neither of these two characteristics is fundamentally necessary in order to establish the logical integrity of this proposal. However, if NIEM had been defined as, for instance, an XML data model, with no stipulation of RDF compatibility, WIJIS would not have been able to afford to establish that integrity.

And that is really perhaps the most important thing to take away from this article. It may be that our proposal will turn out to be unworkable in practice. It may be that it contains some as-yet-unnoticed lapse in logic that will embarrassingly doom the whole idea, but that does not change the fact that NIEM's beneficial characteristics made the venture cost-effective in the first place and will probably have the same effect for other efforts.

This is not a small thing. Good data model design pays off, and it keeps paying off as you encounter new situations. The NIEM designers really do deserve recognition for getting these particular things right.

For more information and/or questions, please contact Bill Blondeau at Bill.Blondeau@wisconsin.gov.