Structure the World

by Mike Bergman - Monday, August 03, 2009

http://www.mkbergman.com/533/structure-the-world/

Multiple Techniques and Data Structs can Make the Vision a Reality

Linked data and subject and domain ontologies provide the organizing framework. Techniques for converting, tagging and authoring structure provide the content. In combination, we now have in hand the necessary pieces to enable all of us to "structure the World."

In this vision, the nature of the links or connections between data need not be complicated to gain tremendous benefit. Similar to Metcalfe's Law for the increasing value of networks as more nodes (users) get added, adding connections to existing data is a powerful force multiplier.

We can call this the Linked Data Law: the value of a linked data network is proportional to the square of the number of links between data objects [1]. Further, if we are purposeful to include connective links where appropriate as we add more data (that is, nodes), this multiplier effect becomes even stronger.

Structured Dynamics is dedicated to help make this prospect real. Meaningful progress in doing so requires only a relatively few moving parts or techniques. Yet, because we sometimes bounce from talking or focusing on one part versus the others, we can lose context or sight of the overarching vision. The purpose of this article is to re-set and calibrate that overall vision.

The Vision: Data Federation of Any Desired Content

The vision is to get all data and information to interoperate, regardless of legacy or form. Much of this data is already structured, either from databases or simpler forms of data structs. Some of this information is unstructured or semi-structured, requiring extraction and tagging techniques. And new information is
being constantly generated, which warrants better means to author and stage for interchange and interoperability.

No matter the provenance, all information has context and scope. As a chunk from here, and a piece from there, gets added to our linked data mix, having means to characterize what that data is about and how it can be meaningfully inter-related becomes crucial. Sometimes these contexts are informed by existing schema; sometimes they are not. But, in any case, it is the role of ontologies to both position these datasets into an "aboutness" framework and to help guide how the data can be described and related to other data. This part of the vision invokes semantics and coherent structures (schema or ontologies) for positioning and mapping datasets to one another.

As both the means for representing any extant data format and as the means for describing these conceptual relationships or schema, RDF provides the canonical data model. A single target representation and common data model also means we can develop and design a smaller universe of tools to operate and provide functionality over all of this data. Indeed, because our RDF data model and its ontologies are so richly structured, we can design our tools with generic functionality, the specific operation and expression of which is based on the inherent structure within the data and its relationships. This vision of **data-driven apps** leads to extreme leverage, incredible flexibility, and inherent "meshup" capabilities for tools.

Further, because we use Web identifiers (URIs) for our data and concepts and because we expose and access this linked data via the Web, we use the proven and scalable architectures of the Web itself for how we design our systems. This [Web-oriented architecture](http://www.mkbergman.com) (WOA) provides a completely decentralized and loosely coupled deployment model that can work ranging from public and open to private and proprietary, applicable to data and participants alike.

From the outset, it is essential to recognize that thousands of contributors are enabling this vision. So, while Structured Dynamics naturally uses its own tools and techniques to flesh out the various parts of this vision below, realize there are many players and many tools from which to choose [2]. For that is another aspect of this vision that is quite powerful: providing choice and avoiding lock-in.

**RDF: The Canonical Data Model**

The core construct -- or fulcrum, if you will -- of the vision is the RDF (Resource Description Framework) data model [3]. I have written elsewhere on the [Advantages and Myths of RDF](http://www.mkbergman.com), which explains more precisely the advantages of that model. RDF provides a common data model to which any external format or schema can be converted and represented. It also provides a logic model and basis for building vocabularies that can inform and drive generic tools.

In the context of data interoperability, a critical premise is that a single, canonical data model is highly desirable. Why?

Simply because of $2N \leq N^2$. That is, a single reference ("canon") structure means that fewer tool variants and converters need be developed to talk to the myriad of data formats in the wild. With a canonical data model, talking to external sources and formats (N) only requires converters to and from the canonical form (2N). Without a canonical model, the combinatorial explosion of required format converters
becomes $N^2$ [4].

Note, in general, such a canonical data model merely represents the agreed-upon internal representation. It need not affect data transfer formats. Indeed, in many cases, data systems employ quite different internal data models from what is used for data exchange. Many, in fact, have two or three favored flavors of data exchange such as XML, JSON or the like. More on this is discussed in a section below.

As this diagram shows, then, we have a single internal representation that is the target for all data and format converters and upon which all tools operate. These tools are themselves expressed as Web services so that they may be distributed and conform to general WOA guidelines. In addition, there may be multiple external "hubs" that represent alternative data models or formats or schema conversions (say, for relational databases). So long as we have converters between these alternate "hubs" and our canonical RDF form we can allow a thousand flowers to bloom:

Other canonical forms could be advocated. Yet RDF has the logical basis to represent any data form and any schema or conceptual structure. It is based on a robust set of open standards and languages and tools. It may be serialized in many formats. It can be grounded in description logics and, in appropriate forms, reasoned over and expressed in vocabularies and schema suitable for the most complex of conceptual structures and semantics. RDF is the data model explicitly designed for the Web, the clear global information basis for the foreseeable future.

For more than 30 years -- since the widespread adoption of electronic information systems by enterprises -- the Holy Grail has been complete, integrated access to all data. With the canonical RDF data model,
that promise is now at hand.

**Conversion: So Many Structs, So Little Time**

Diversity is a truism of human communications as captured by the biblical Tower of Babel and the many thousands of current human languages. Diversity in data formats, serializations, notations and languages is a similar truism. We term the expression of each of these varied forms of data a *struct*.

While an internal canonical representation of data makes sense for the reasons noted above, pragmatic information systems must recognize the inherent diversity and chaos of data in the real world. The history of trying to find single representations or to impose standards via fiat have singularly failed. That will continue to be so due in part to inertia and legacy, sunk investments, existing infrastructure, and the purposes for the data.

In pursuing a vision of data interoperability, then, conversion is an essential glue for cementing understanding with what exists and will exist.

**RDB-to-RDF**

Arguably the largest source of structured data are enterprise and government information systems, with the predominant data representation being the relational data model managed by relational schema. Much of this data is also cleaner and mission critical compared to other sources in the wild. Fortunately, there are many logical and conceptual affinities between the relational model and the one for RDF [5].

Just as there are many RDFizers for simpler forms of data structs (see next), there are also nice ways to convert relational schema to RDF automatically. Given these overall conceptual and logical affinities the W3C is also in the process of graduating an incubator group to an official work group, RDB2RDF, focused on methods and specifications for mapping relational schema to RDF.

Amongst all techniques covered in this paper, Structured Dynamics views the layering of RDF ontologies over existing relational data stores as one of the most promising and important. Given the advantages of RDF for interoperability, this area should be a major emphasis of current and new vendors and service providers.

**RDFizers**

Much data, however, resides in much smaller datasets and often for less formal purposes than what is found in enterprise databases. Some of this data is geared for exchange or standardization; much is emerging from Web and Internet applications and uses; and much might be local or personal in nature, such as simple lists or spreadsheets.

RDF is well suited to convert ("RDFize") these simpler and more naïve data formats. In my original census about 18 months ago, as reported in *Structs*: Naïve Data Formats and the ABox, I listed about 90 converters. My most recent update now lists nearly double that number, with about 150 converters [6]:

- URN handlers (in addition to Embedded Microformats, REST-style Web service, Metadata extractors)
Many of the sources above come from new and emerging Web-based APIs, which are also huge sources of content growth. Also note that alternative formats to RDF (e.g., microformats) or leading serializations and encodings (e.g., XML, JSON) also have many converter options.

For many typical naïve data structs, the data is represented as attribute-value pairs, which easily lend
themselves to conversion to RDF as instance records [8]. See further the Authoring section below.

**Tagging: The 80% Solution**

An apocryphal statistic is that 80% to 85% of all information resides in unstructured text [9]. Besides lacking recent validation, this claim from a decade ago often attributed to Merrill Lynch also precedes much of the Internet and the emergence of metadata and tagging. Nevertheless, what is true is that written text content is ubiquitous and the majority of it remains untagged or uncharacterized by any form of metadata.

While such information can be searched, it only matches when exact terms match. This means that related information, particularly in the form of conceptual relationships and inferencing, can not be applied to untagged text content.

While information extraction -- the basis by which tags for entities and concepts can be obtained -- has been an active topic of research for two decades, it is only recently that we have begun to see Web-scale extractors appear. Examples include Yahoo's term extractor, Thomson Reuter's Calais, or Google's Squared, to name but a few.

**scones**

In Structured Dynamics' case we have been working on the scones (Subject Concepts Or Named EntitieS) extractor for quite a while. scones uses rather simple natural language processing (NLP) methods as informed by concept ontologies and named entity (instance record) dictionaries to help guide the extraction process. The co-occurrence of matches between concepts and entities also aids the disambiguation task (though additional modules may be invoked with alternative disambiguation methods). In prototype forms, the resulting tags can be managed separately or fed to user interfaces or re-injected back into the original content as RDFa.

There are literally dozens of such extractors and services presently available on the Web and many that are available as open source or commercial products. Some are mostly algorithm based using machine-learning techniques or statistics, while others are gazetteer- or dictionary-driven.

These systems will lead to rapid tagging of existing content and the removal of some of the early "chicken-and-egg" challenges associated with the semantic Web. These systems will also be combined with the many existing bookmarking and tagging services.

So, just as we will see federation and interoperability of conventional data, we will also see linkages to relevant and supporting text content accompanying it. This combination, in turn, will also lead to richer browsing and discovery experiences.

**Authoring: The Neglected Third Leg of the Stool**

In addition to conversion and tagging, authoring is the third leg of the stool to expose structured data. It is a neglected leg to the structured content stool, and one important to make it easier for datasets to be easily exposed as RDF linked data.
One of the reasons for the proliferation of data structs has been the interest in finding notations and conventions for easier reading and authoring of small datasets. There have literally been hundreds of various formats proposed over decades for conveying lightweight data structures. Most have been proprietary or limited to specific domains or users. Some, such as fielded text, structured text, simple declarative language (SDL), or more recently YAML or its simpler cousin JSON, have become more widely adopted and supported by formal specifications, tools or APIs. JSON, especially, is a preferred form for Web 2.0 applications.

What has been less clear or intuitive in these forms, again mostly based on an attribute-value pair orientation, is how to adequately relate them to a more capable data model, such as RDF. In JSON or YAML, for example, the notations include the concepts of objects, arrays and datatypes (among other conventions). Other structures lack even these constructs.

To take the case of JSON as might be related to RDF, there are a couple of efforts to define representation conventions from Talis and GBV for serializing RDF. There was a floated idea for an RDF version of JSON called RDFON that has now evolved into the TURF approach. JDIL (JSON data integration layer) instructs how to add namespaces to JSON to enable encoding RDF. Jim Ley, Kanzaki Masahide and Dave Beckett (likely among others) have written simple and straightforward RDF and Turtle parsers and converters for JSON. And, still further examples are Beckett's Triplr and Sören Auer's ASKW Triplify lightweight conversion services involving many different formats.

Because JSON is easily readable, can drive many Web 2.0 applications and widgets, and lends itself to fast conversions and tools in various scripting languages, Structured Dynamics was commissioned by the Bibliographic Knowledge Network (BKN) to formalize a BibJSON specification suitable for BibTeX-like data records and citations with an extensible schema to be converted to RDF.

The emerging result of that BibJSON effort will be published shortly. The specification includes conventions and vocabularies for creating bibliographic and citation instance records, for specifying structural schema, and for creating linkage files between the attributes in the record files with existing and new schema. BibJSON is itself grounded in IRON, which is an instance record and object notation developed by Structured Dynamics that can be serialized as JSON (called irJSON), XML (called irXML) or comma-separated values (or CSV comma-delimited files, called commON).

The purpose of these notations and serializations is to provide easier authoring environments and scripting support to RDF-ready datasets. This approach has the advantage of shielding most users from the nuances or lengthiness of RDF (though the N3 serialization also works well).

The design and development of commON was especially geared to using spreadsheets as authoring environments that would enable easy creation of instance record tables or simple hierarchical or outline structures. For example, here is a sample portion of Sweet Tools specified in a spreadsheet using the commON notation:
<table>
<thead>
<tr>
<th>Title</th>
<th>URL</th>
<th>Description</th>
<th>Author</th>
<th>Date</th>
<th>Source</th>
<th>Keywords</th>
<th>Actions</th>
</tr>
</thead>
<tbody>
<tr>
<td>Artificial Reality</td>
<td><a href="http://www.mkbergman.com/">http://www.mkbergman.com/</a></td>
<td>A discussion on the concept of artificial reality, its potential impact on society, and the ethical considerations surrounding its development.</td>
<td>Mike Bergman</td>
<td>08-03-2009</td>
<td>AI3: Adaptive Information</td>
<td>Artificial reality, ethics, society</td>
<td></td>
</tr>
<tr>
<td>The Future of Work</td>
<td><a href="http://www.mkbergman.com/">http://www.mkbergman.com/</a></td>
<td>A look at the potential changes in the workplace, driven by advances in technology and shifts in the economy.</td>
<td>Mike Bergman</td>
<td>08-03-2009</td>
<td>AI3: Adaptive Information</td>
<td>Future of work, technology, economy</td>
<td></td>
</tr>
<tr>
<td>Big Data: Opportunities and Challenges</td>
<td><a href="http://www.mkbergman.com/">http://www.mkbergman.com/</a></td>
<td>A discussion on the potential opportunities and challenges presented by the rapid growth of big data, and the need for new approaches to data analysis and decision-making.</td>
<td>Mike Bergman</td>
<td>08-03-2009</td>
<td>AI3: Adaptive Information</td>
<td>Big data, opportunities, challenges</td>
<td></td>
</tr>
<tr>
<td>Future of Energy</td>
<td><a href="http://www.mkbergman.com/">http://www.mkbergman.com/</a></td>
<td>An examination of the potential energy sources of the future, and the role of technology in shaping the energy landscape.</td>
<td>Mike Bergman</td>
<td>08-03-2009</td>
<td>AI3: Adaptive Information</td>
<td>Future of energy, technology, energy sources</td>
<td></td>
</tr>
<tr>
<td>The Impact of Technology on Society</td>
<td><a href="http://www.mkbergman.com/">http://www.mkbergman.com/</a></td>
<td>An analysis of how technology shapes society, and the need for responsible innovation to ensure its positive impact.</td>
<td>Mike Bergman</td>
<td>08-03-2009</td>
<td>AI3: Adaptive Information</td>
<td>Technology, society, responsible innovation</td>
<td></td>
</tr>
<tr>
<td>The Impact of Social Media on Political Discourse</td>
<td><a href="http://www.mkbergman.com/">http://www.mkbergman.com/</a></td>
<td>A discussion on the role of social media in shaping political discourse, and the challenges it presents for democracy.</td>
<td>Mike Bergman</td>
<td>08-03-2009</td>
<td>AI3: Adaptive Information</td>
<td>Social media, political discourse, democracy</td>
<td></td>
</tr>
<tr>
<td>The Impact of Technology on Privacy</td>
<td><a href="http://www.mkbergman.com/">http://www.mkbergman.com/</a></td>
<td>A discussion on the potential impact of technology on privacy, and the need for new approaches to protect sensitive information.</td>
<td>Mike Bergman</td>
<td>08-03-2009</td>
<td>AI3: Adaptive Information</td>
<td>Technology, privacy, information protection</td>
<td></td>
</tr>
<tr>
<td>The Future of Work</td>
<td><a href="http://www.mkbergman.com/">http://www.mkbergman.com/</a></td>
<td>A look at the potential changes in the workplace, driven by advances in technology and shifts in the economy.</td>
<td>Mike Bergman</td>
<td>08-03-2009</td>
<td>AI3: Adaptive Information</td>
<td>Future of work, technology, economy</td>
<td></td>
</tr>
<tr>
<td>Big Data: Opportunities and Challenges</td>
<td><a href="http://www.mkbergman.com/">http://www.mkbergman.com/</a></td>
<td>A discussion on the potential opportunities and challenges presented by the rapid growth of big data, and the need for new approaches to data analysis and decision-making.</td>
<td>Mike Bergman</td>
<td>08-03-2009</td>
<td>AI3: Adaptive Information</td>
<td>Big data, opportunities, challenges</td>
<td></td>
</tr>
<tr>
<td>Future of Energy</td>
<td><a href="http://www.mkbergman.com/">http://www.mkbergman.com/</a></td>
<td>An examination of the potential energy sources of the future, and the role of technology in shaping the energy landscape.</td>
<td>Mike Bergman</td>
<td>08-03-2009</td>
<td>AI3: Adaptive Information</td>
<td>Future of energy, technology, energy sources</td>
<td></td>
</tr>
<tr>
<td>The Impact of Technology on Society</td>
<td><a href="http://www.mkbergman.com/">http://www.mkbergman.com/</a></td>
<td>An analysis of how technology shapes society, and the need for responsible innovation to ensure its positive impact.</td>
<td>Mike Bergman</td>
<td>08-03-2009</td>
<td>AI3: Adaptive Information</td>
<td>Technology, society, responsible innovation</td>
<td></td>
</tr>
<tr>
<td>The Impact of Social Media on Political Discourse</td>
<td><a href="http://www.mkbergman.com/">http://www.mkbergman.com/</a></td>
<td>A discussion on the role of social media in shaping political discourse, and the challenges it presents for democracy.</td>
<td>Mike Bergman</td>
<td>08-03-2009</td>
<td>AI3: Adaptive Information</td>
<td>Social media, political discourse, democracy</td>
<td></td>
</tr>
<tr>
<td>The Impact of Technology on Privacy</td>
<td><a href="http://www.mkbergman.com/">http://www.mkbergman.com/</a></td>
<td>A discussion on the potential impact of technology on privacy, and the need for new approaches to protect sensitive information.</td>
<td>Mike Bergman</td>
<td>08-03-2009</td>
<td>AI3: Adaptive Information</td>
<td>Technology, privacy, information protection</td>
<td></td>
</tr>
</tbody>
</table>

AI3: Adaptive Information | http://www.mkbergman.com/ | A non-profit organization that focuses on the role of technology in shaping society, and the need for responsible innovation to ensure its positive impact. | Mike Bergman | 08-03-2009 | AI3: Adaptive Information | Technology, society, responsible innovation |

8 / 12
Once the philosophy and role of naïve data structs is embraced -- with an appreciation of the many converters now available or easily written for translating to RDF -- it becomes easier to determine data forms appropriate to the tools and natural work flow of the users and tasks at hand. Under this mindset, the role of RDF is to be the eventual conversion target, but not necessarily what is used for intermediate work tasks, and in particular not for authoring.

Getting it All Organized

OK, so now all of this stuff is converted, tagged or authored. How does it relate? What is the relation of one dataset to another dataset? Is there a context or framework for laying out these conceptual roadmaps?

Two years ago as we looked at the state of RDF and the incipient semantic Web as promised via linked data, we saw that such a specific framework was lacking. (Though there were existing higher-level ontologies, either their complexity or design were not well-suited to these purposes.) It was at that time that Frédérick Giasson and I began to formulate the UMBEL (Upper Mapping and Binding Exchange Layer) ontology, which eventually led to our more formal business partnership and Structured Dynamics.

What we sought to achieve with UMBEL was a coherent reference framework of about 20,000 subject concepts, connected and acting like constellations in the information sky for orienting content and new datasets. At the same time, we wanted to create a general vocabulary and approach that would lend themselves to creation of domain-specific ontologies, which would also naturally tie in and inter-relate to the more general UMBEL structure.

This objective was achieved, though UMBEL deserves an upgrade to OWL 2 and some other pending improvements. A number of domain ontologies have been created and now relate to UMBEL. So, rather than being an end to itself, UMBEL was one of the necessary infrastructure pieces to help make the vision herein a reality.

Similar approaches may be taken by others with new domain ontologies based on the UMBEL vocabulary with tie-in as appropriate to existing subject concepts, or by mapping to the existing UMBEL structure.

Of course, UMBEL is not an absolute condition to the vision herein. However, insofar as users desire to see multiple datasets inter-related, including the use of existing public Web data, something akin to UMBEL and related domain ontologies will be necessary to provide a similar roadmap.

Making it All Available

The parts and techniques discussed so far pertain almost exclusively to data and content. But, these structures so created now can inform data-driven applications which also now must be deployed. To do
so, Structured Dynamics is committed to what is known as a **Web-oriented architecture** (WOA):

\[
\text{WOA} = \text{SOA} + \text{WWW} + \text{REST}
\]

WOA is a subset of the **service-oriented architectural** style, wherein discrete functions are packaged into modular and shareable elements ("services") that are made available in a distributed and loosely coupled manner. WOA generally uses the representational state transfer (REST) architectural style defined by Roy Fielding in his 2000 doctoral thesis; Fielding is also one of the principal authors of the Hypertext Transfer Protocol (HTTP) specification.

REST provides principles for how resources are defined and used and addressed with simple interfaces without additional messaging layers such as SOAP or RPC. The principles are couched within the framework of a generalized architectural style and are not limited to the Web, though they are a foundation to it.

Within this design we need a suite of generic functions and tools that are driven by the structure of the available datasets. The deployment vehicle and design we have implemented to provide this WOA design is structWSF [10].

StructWSF is a platform-independent Web services framework for accessing and exposing structured RDF data. Its central organizing perspective is that of the dataset. These datasets contain instance records, with the structural relationships amongst the data and their attributes and concepts defined via ontologies (schema with accompanying vocabularies). The master or controlling Web service in the framework is the module for granting access and use rights to datasets based on permissions.

The structWSF middleware framework is generally RESTful in design and is based on HTTP and Web protocols and open standards. The initial structWSF framework comes packaged with a baseline set of about a dozen Web services in CRUD, browse, search and export and import. More services can readily be added to the system.

All Web services are exposed via APIs and SPARQL endpoints. Each request to an individual Web service returns an HTTP status and a document of resultsets (if the query result is not null). Each results document can be serialized in many ways, and may be expressed as either RDF or pure XML.

In initial release, structWSF has direct interfaces to the Virtuoso RDF triple store (via ODBC, and later HTTP) and the Solr faceted, full-text search engine (via HTTP). However, structWSF has been designed to be fully platform-independent. The framework is open source (Apache 2 license) and designed for extensibility.

**No End in Sight**

Like all visions, there are many aspects and many improvements possible. This vision is definitely a work-in-progress with no end in sight.
But, meaningful movement embracing the full scope of this vision is doable today. Structured Dynamics welcomes inquiries regarding any of these aspects, improvements to them, or application to your specific needs and problems.

We also welcome you to come back and visit our blogs (Fred's is found here). We try to speak on various aspects of this vision in all of our posts and are pleased to share our experience and insights as gained.

[1] Metcalfe's law states that the value of a telecommunications network is proportional to the square of the number of users of the system (n^2), where the linkages between users (nodes) exist by definition. For information bases, the data objects are the nodes. Linked data works to add the connections between the nodes. We can thus modify the original sense to become the Linked Data Law: the value of a linked data network is proportional to the square of the number of links between the data objects. I first presented this formulation about a year ago in What is Linked Data?

[2] This piece introduces for the first time a couple of efforts-in-progress by Structured Dynamics. For a general tools listing, see my own Sweet Tools listing of about 800 semantic Web and-related tools.

[3] As quoted in The Lever. "'Archimedes, however, in writing to King Hiero, whose friend and near relation he was, had stated that given the force, any given weight might be moved, and even boasted, we are told, relying on the strength of demonstration, that if there were another earth, by going into it he could remove this." from Plutarch (c. 45-120 AD) in the Life of Marcellus, as translated by John Dryden (1631-1700).


[5] An excellent piece on those relations was written by Andrew Newman a bit over a year ago; see Andrew Newman, 2007. "A Relational View of the Semantic Web," published on XML.com, March 14, 2007; http://www.xml.com/pub/a/2007/03/14/a-relational-view-of-the-semantic-web.html. RDF can be modeled relationally as a single table with three columns corresponding to the subject-predicate-object triple. Conversely, a relational table can be modeled in RDF with the subject IRI derived from the primary key or a blank node; the predicate from the column identifier; and the object from the cell value. Because of these affinities, it is also possible to store RDF data models in existing relational databases. (In fact, most RDF "triple stores" are RDBM systems with a tweak, sometimes as "quad stores" where the fourth tuple is the graph.) Moreover, these affinities also mean that RDF stored in this manner can also take advantage of the historical learnings around RDBMS and SQL query optimizations.

[6] The largest source for RDFizers, which it calls Sponger cartridges, is from OpenLink Software in relation to its Virtuoso universal server. Most of its converters use XSLT stylesheets to translate to RDF, but the system has other conversion capabilities as well. Two additional OpenLink resources are a clickable diagram of converters and relationships with links and an online storehouse of available XSLT converters. In addition, two other sources -- the W3C's Semantic Web wiki with converter listings and MIT's Simile program and listing of RDFizers -- have a rich set of listings. Note that many of the categories shown on the table also have multiple sources of converters, so that the absolute number of converters has also grown faster than the unique formats supported.

[7] GRDDL (Gleaning Resource Descriptions from Dialects of Languages) is a W3C markup format for getting RDF data out of XML and XHTML documents using explicitly associated transformation algorithms, typically represented in XSLT GRDDL accommodates a wide variety of dialects (see one listing) and can be combined with arbitrary transformation mechanisms (though currently mostly based on XSLTs).
We characterize instance records as representing the "ABox", in accordance with our working definition for description logics:

"Description logics and their semantics traditionally split concepts and their relationships from the different treatment of instances and their attributes and roles, expressed as fact assertions. The concept split is known as the TBox (for terminological knowledge, the basis for T in TBox) and represents the schema or taxonomy of the domain at hand. The TBox is the structural and intensional component of conceptual relationships. The second split of instances is known as the ABox (for assertions, the basis for A in ABox) and describes the attributes of instances (and individuals), the roles between instances, and other assertions about instances regarding their class membership with the TBox concepts."

One of the more recent discussions of this percentage is by Seth Grimes, Unstructured Data and the 80 Percent Rule, 2009.

structWSF is also designed to integrate with third-party apps and content management systems (CMSs) to provide the user interfaces to these functions. The first implementation of this design is conStruct SCS, a structured content system that extends the basic Drupal content management framework. conStruct enables structured data and its controlling vocabularies (ontologies) to drive applications and user interfaces.