

Proposed Guideline Clarifications for American Recovery and Reinvestment Act of 2009

Erik Wilde, Eric Kansa, and Raymond Yee
[School of Information, UC Berkeley](#)

UCB ISchool Report 2009-029
March 2009

Available at <http://repositories.cdlib.org/ischool/2009-029/>

Abstract

The *Initial Implementing Guidance for the American Recovery and Reinvestment Act of 2009* provides guidance for a feed-based information dissemination architecture. In this report, we suggest some improvements and refinements of the initial guidelines, in the hope of paving the path for a more transparent and useful feed-based architecture. This report is meant as a preliminary guide to how the current guidelines could be made more specific and provide better guidance for providers and consumers of Recovery Act spending information. It is by no means intended as a complete or final set of recommendations.

Contents

1	Introduction	2
2	Technical Considerations	2
3	Site Discovery	3
4	Feed Discovery	3
5	Feed Types	4
6	Data Models	4
7	Feed Format	8
8	Validation	10
9	Feed Usage	11
10	Conclusions	12

This work is licensed under a [Creative Commons Attribution 3.0 United States License](#).

1 Introduction

The *Initial Implementing Guidance for the American Recovery and Reinvestment Act of 2009* [22] published on February 18, 2009, provides guidance for a feed-based information dissemination architecture. The goal of this architecture is to make sure that spending can be openly and transparently tracked at the agencies receiving funds from the Recovery Act, and to make sure that data can be aggregated and re-published on the portal being set up at <http://recovery.gov>.

The feed-based publishing of data proposed by the *Office of Management and Budget (OMB)* represents an exciting first step in making government transparency on the Web a reality, and it is in line with the current world-wide efforts to improve access to government data and services using the Web [2]. However, as yet, the OMB guidelines for feed implementation (in terms of how to publish feeds and what to publish in them) remain underspecified and require some modification and expansion to be more effective. This document serves as a guide towards making feeds work better for their stated transparency and accountability goals.

This report is meant as a preliminary guide to how the current guidelines could be made more specific and provide better guidance. It is by no means intended as a complete or final set of recommendations. Furthermore, this document only looks at the data dissemination aspect, and only at the specific case of the information regarding the Recovery Act. The strategy we follow with the guidelines laid out in this report follow the idea of the *Plain Web* [29], always attempting to use the simplest and most widely known and supported technology for any given task.

2 Technical Considerations

Along with this report, a sample set of simulated datasets and sample files is made available at <http://isd.ischool.berkeley.edu/stimulus/2009-029/>, and all concrete file names mentioned in examples throughout this document within this technical report are considered to be relative to that URI. The data that has been used for the sample data is mostly fictional and partly based on the information already available through the first recovery Web sites. All sample data can be found in `feeds.xml`, which conforms to `feeds.xsd`. Using `feeds.xslt`, the sample dataset has been converted into a set of “Web sites” and feeds, and the main purpose of this was to get some data to build on as a prototyping and experimentation platform.

All feed data generated from the sample dataset is supposed to be in the form we recommend for later use in feeds for Recovery Act information, with the exception of geocoding information that is included in the feeds as explicit GeoRSS markup (see Section 7 for details). While we do not recommend to require GeoRSS, it might be recommended, and it would be easy to generate GeoRSS information from the address information that is required in the data model of the published information (Section 6). For demonstration purposes, the published feeds in the sample dataset have already been geocoded. Since we recommend to only use *Atom*[19] as the feed format (Section 7), all feeds produced for the sample dataset are Atom feeds.

These guidelines are intended as guidelines for the feeds published by the individual agencies, so that interested parties can access the relevant data directly from these sites. It should be noted, though, that large parts of these guidelines could equally be applied to feeds being re-published by the OMB on `recovery.gov`, specifically Sections 5-8.

Given that the current guidelines purely focus on information dissemination,¹ we have chosen the same focus for this report. Specifically, while we are proposing Atom as the foundation for information dissemination, we are not proposing any services beyond read-only services. However, for a more concrete set of guidelines including submission services, we strongly recommend to consider the *Atom Publishing Protocol*

¹The only service which requires “write access” mentioned in the current guidelines is the requirement to send the individual reports by email to the *General Services Administration (GSA)*.

(AtomPub) [12] as the foundation for services beyond read-only. In particular, services such as the validation services discussed in Section 8, could be implemented as AtomPub services, or could be used to support AtomPub services, so that data submission can be implemented in a user-friendly way.

3 Site Discovery

Making sources of authoritative information about Recovery-relating spending and outcomes easy to find is a key requirement for the effort's transparency measures. Meeting this requirement will involve both appropriate administrative and technical processes. The maintainers of [recovery.gov](#) will need to determine that the information provided by participating agencies is vetted, properly updated, and properly maintained.

Once [recovery.gov](#) has vetted information sources, it needs to point to those sources in ways that facilitate their reliable discovery and use by members of the public. While search engines are powerful tools, they are not substitutes for an agency's authoritative vetting of information sources and links. This information should be expressed in both human-readable web pages, and in formats easier for machine processing such as the *Extensible Markup Language (XML)* [7]. The [recovery.xml](#) file demonstrates one simple way of facilitating discovery of vetted Web sites.

The recommendations made on [page 58](#) of the current guidelines require all agencies to set up recovery Web sites at <http://agency.gov/recovery>, but not all agencies have their own domains, and it is also not entirely clear how to find a list of agencies that could have recovery sites. <http://www.recovery.gov/?q=content/agencies> currently publishes a list of existing recovery sites, but it is only published as a Web page and not available in machine-readable form.

We recommend to make a list of all recovery sites available in machine-readable form. [recovery.xml](#) is a very simple format for this which allows users to find both all known recovery Web sites, and the feeds made available through them (see Section 4 for feed discovery). While [recovery.xml](#) is a static file in our example dataset, we strongly recommend to implement this information as a feed as well, so that consumers of that feed can get easy access to the currently known set of recovery sites, and also have a simple way of being informed about extensions and/or updates to that set.

4 Feed Discovery

While recovery Web site (as discussed in Section 3) are important as portals for human users, more important for transparency purposes is information access to the machine-readable reports on recovery spending. One of the most important requirements for feeds is that they should be easy to find. There are two options (not mutually exclusive) which can facilitate feed discovery:

1. *Setting up a central registry.* Essentially this would be a comprehensive directory of sites and feeds (this can be as simple as the [recovery.xml](#) file presented in the previous section, or a feed-based representation of that information). More sophisticated versions of this could be run by a database that supporting queries about sites and feeds. For instance, a database driven registry could return a list of all feeds related to "transportation" or "energy" related agencies.
2. *Predictable ways to autodiscover feeds.* OMB can stipulate feed auto-discovery requirements as part of the requirements for the set up of "recovery" Web pages at <http://agency.gov/recovery>. The addition of a feed autodiscovery requirement for pages will give users (and software, such as web crawlers) a reliable method for discovering an agency's feeds. However, to be fully machine-readable, autodiscovery would have to use a way how the different feed types (Section 5) can be differentiated.

Unfortunately, feed autodiscovery currently is not standardized anywhere. There are developments worth noting however, including [best practices](#), a draft for setting up an IANA registry for link relations [18], and

a proposed “`feed`” link relation in the draft for *HTML 5* [13]. While not formally standardized, feed autodiscovery is well supported by current browsers, and could be implemented reliably with a well-defined set of implementation guidelines for the recovery Web sites; essentially pre-defined HTML code fragments that should be used to link to an agency’s feeds.

Since one major goal of these guidelines is to promote a distributed scenario, and not just a centralized repository, we recommend to require feed autodiscovery for all recovery Web sites, and to support this by providing HTML code fragments. In addition, [recovery.gov](#) should provide a repository of all known feeds (in a way similar to [recovery.xml](#)), and make this available in machine-readable form.

5 Feed Types

The current guidelines seem to require three feeds ([page 54](#)) for the three types of items that agencies are expected to make available. These feed types are

- Major Communications feeds,
- Formula Block Grant Allocation feeds, and
- Weekly Report feeds.

It is not quite clear, though, whether agencies are free to have just one feed carrying three types of entries, or whether three separate feeds are required. To simplify information dissemination, it would be better to require three separate feeds. A fourth feed that aggregates content from the these other required feeds would also be useful, and could be allowed as an optional part of the guidelines. In any event, feed discovery requirements would have also have to specify how each of the required three feeds (and the optional combined fourth feed) can be found from an agency’s recovery Web page.

If the guidelines decide to allow combined feeds, they should specify a way how entries in that fourth feed should be marked as being republished entries from one of the other three feeds. We recommend to require three separate feeds for the three types of communications items published by agencies. Furthermore, we recommend to allow and encourage the creation of a combined fourth feed, and to clearly specify of how entries in that feed have to be related to their “original publication” in one of the three separate feeds.

Atom has various mechanisms to facilitate this kind of republication, `atom:source` can retain an entry’s original context from its `atom:feed` in a new feed, and `atom:link` could be used (with some convention to be specified in the guidelines) to point to the original location of the feed and/or entry.

6 Data Models

While this report focuses on the way how data should be made available, a central question of course is how the data itself should be represented. Starting on [page 55](#), the current guidelines contain a list of data elements. These data elements are specified using SQL datatypes, but certain details are not mentioned at all, for example whether there are repeating structures.² A government Web site provides templates for the information contained in these tables, but we could not access the Web site due to access restrictions. It seems, though, as if templates are provided in various formats, and the currently published data uses Excel, Word, and the *Hypertext Markup Language (HTML)* [25] as data formats.

We recommend that instead of using proprietary data formats such as Excel or Word, the guidelines should only allow machine-readable open data formats, so that data from the reports can be easily extracted

²Some of the already published reports do repeat some structures, which is why our proposed schemas support repeating structures.

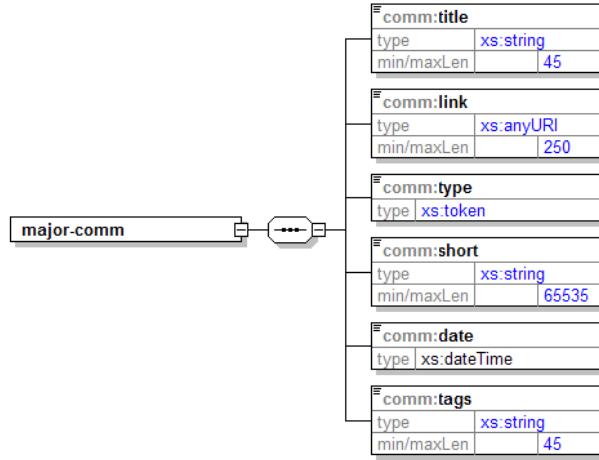


Figure 1: Schema for Major Communications Data

and processed with widely available tools. The two most important candidates for structured data on the Web are these two formats:

- *Extensible Markup Language (XML)*: XML [7] has become the most widely used data format on the Web, and is based on a tree model. XML’s main features is a similarity to HTML (which makes it reasonably easy to get used to for people with HTML knowledge), support for semi-structured data, and widespread support built into almost all modern platforms and programming languages. XML has a built-in schema language called *Document Type Definition (DTD)*, but there are other schema languages available as well, such as *XML Schema (XSD)* [28] and *RELAX NG* [14]
- *Resource Description Framework (RDF)*: [16] is a popular data format in the *Semantic Web* community and uses a model of triple-based graphs. It has strong backing from the *World Wide Web Consortium (W3C)*, but is not as widely supported or used as XML technologies. RDF is a format for data and RDF schemas (often referred to as *ontologies*) are using schema languages such as *RDF Schema (RDFS)* [8] and the *Web Ontology Language (OWL)* [23].

While XML’s tree model may look like a limitation of the data format, it is important to note that XML used together with the architectural style of *Representational State Transfer (REST)* [11] results in a graph, because one of the key constraints of REST is the fact that representations should be “connected” through hyperlinks, allowing applications to traverse those links while they are interacting with resources. Thus, while RDF in itself uses a graph-based model, XML employed in a RESTful architecture creates the same kind of structure.

XML or RDF are metamodels, which allow models to be constrained by schemas, and then exchanged by using well-defined serializations (also referred to as syntaxes). For XML as it is used on the Web today, there are two popular XML serializations:

- *Extensible Markup Language (XML)*: XML itself defines a syntax [7], which is based on markup very similar to HTML. While this syntax is rather verbose, it is widely used because it is easy to understand and use for anybody accustomed to HTML.

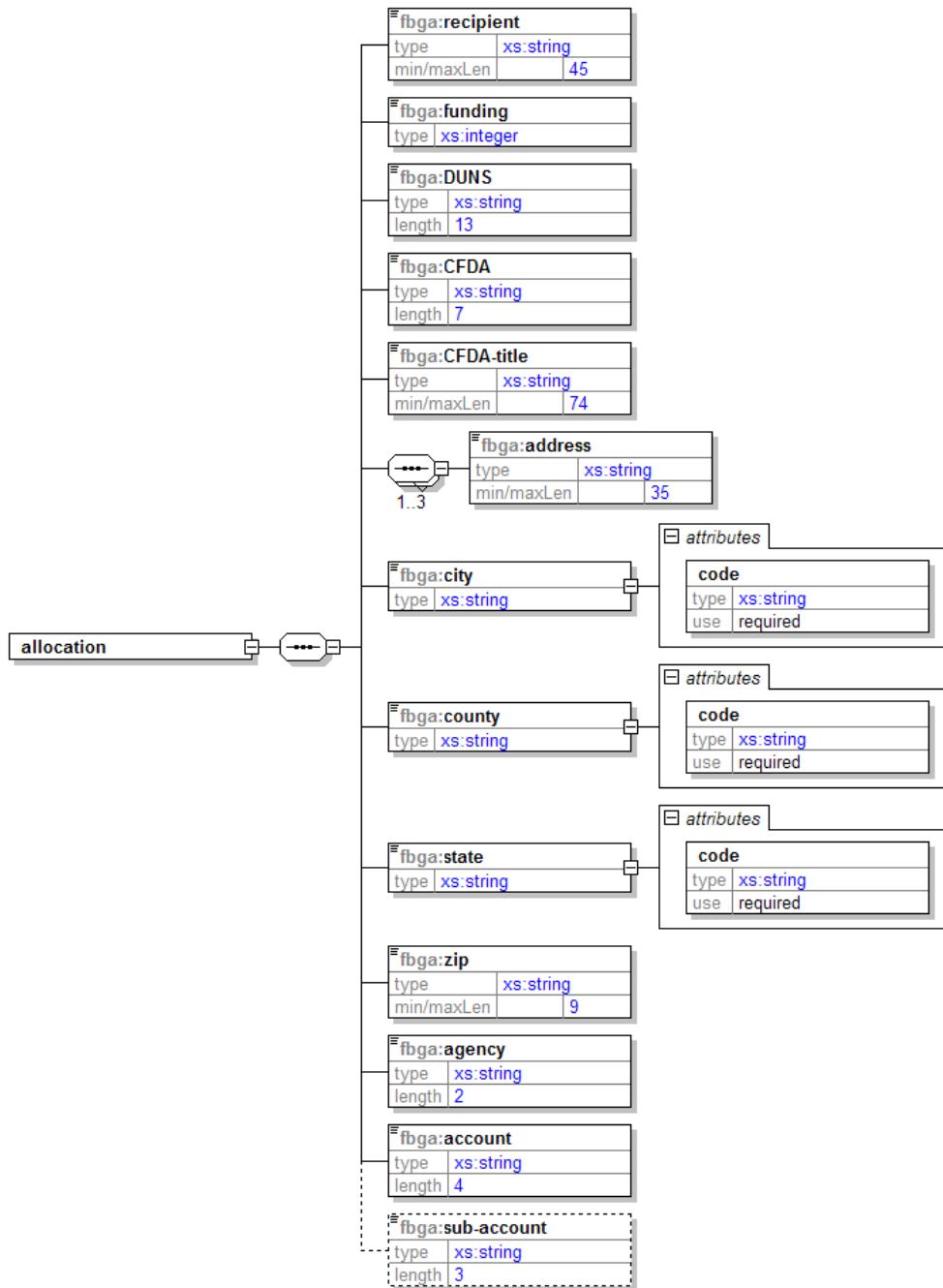


Figure 2: Schema for Formula Block Grant Allocation Data

- *JavaScript Object Notation (JSON)*: With the growing popularity of *Asynchronous JavaScript and XML (Ajax)* applications, an increasing amount of XML data processing has to be done in JavaScript.

Instead of parsing XML on the client-side, JSON [9] defines a way how XML data can be encoded as a JavaScript object, so that client-side processing is very simple. While JSON is limited in the XML it can represent, it has proven its utility in many Ajax applications.

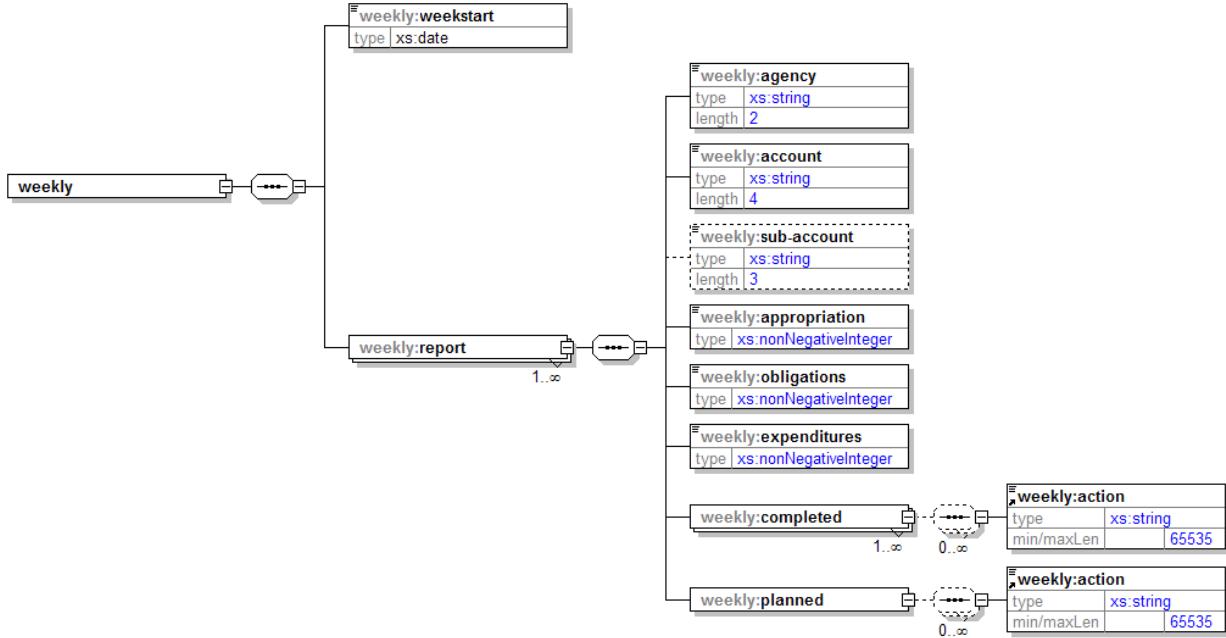


Figure 3: Schema for Weekly Update Report Data

A third XML serialization is the *Efficient XML Interchange (EXI)* [26] format (defining a binary serialization of XML), but this format is still under development. For RDF serializations, there are also two popular formats in widespread use on the Web:

- *RDF/XML*: The original syntax for RDF used XML, and this still is a widely used syntax for RDF [3]. However, processing RDF/XML can be cumbersome with plain XML tools, because RDF/XML has many syntax variations for serializing the same RDF graph.
- *RDF in Attributes (RDFa)*: RDFa [1] has been developed as an approach to embed RDF data in XHTML [24] (the XML-compliant version of HTML). The idea of RDFa is to combine human-readable HTML content with machine-readable RDF, and it can be regarded as a format for standardizing RDF-based microformats.

In addition to the bigger question of the metamodel (such as XML and RDF), there also is the question of how datatypes in that metamodel should be defined. For Web-based applications, the datatypes from *XML Schema (XSD)* [4] have become the defacto standards, they have been included in the *XQuery 1.0 and XPath 2.0 Data Model (XDM)* [10] and are therefore an integral part of *XQuery* [5] and *XSL Transformations (XSLT) 2.0* [15].

For the purpose of our prototype, we have implemented the data model of the three types of feeds in XSD, mostly translating the SQL datatypes into XSD datatypes, and making some assumptions about details of the data models (such as optional and repeatable parts) that are not specified in the current guidelines.

To facilitate data use, aggregation, and alternative visualizations, information representation in a simple XML format would be ideal. We are not aware of any published schema that match the current recovery requirements. Because of this, we created some simple draft schema that meet current requirements for the major communications information ([comm.xsd](#) shown in Figure 1), the formula block grant allocation information ([fbga.xsd](#) shown in Figure 2), and the weekly report information ([weekly.xsd](#) shown in Figure 3). It should be emphasized that these are simply drafts for demonstration purposes only. We anticipate that the final shape and structure of these schema will change significantly when specific requirements for these transparency measures are better understood.

The current schemas ([comm.xsd](#), [fbga.xsd](#), and [weekly.xsd](#)) are too simple to be useful for the eventual guidelines. However, we do believe they should be as simple as possible, so that the data can be easily processed with standard tools. The current implementation publishes all reports in the feeds in XHTML, and uses Atom links to make the XML data available (more about this in Section 7). We recommend that XML should be required as the data format, and that the schemas should be specified in an established XML schema language (Section 8 discusses how these schemas should be used for data quality assurance).

A consideration which is outside of the scope of this report is the question whether there also should be well-defined two-way mappings from the feed schemas to the [*National Information Exchange Model \(NIEM\)*](#) or any other model that could be regarded as a useful context for that data.

7 Feed Format

The current guidelines specify the feed format as “preferred: Atom 1.0, acceptable: RSS”, and do not mention RSS versions. This means that feeds can use 10 different formats, *Atom*[19] plus 9 different RSS variants.³ While it would be the best approach to only allow Atom, it would be good to at least limit RSS to specific versions, such as RSS 2.0.

Another important issue not specifically addressed by the guidelines is how exactly information has to be published through the feed. Atom allows various ways in which this could be done, when seen in conjunction with various data formats (as discussed in Section 6):

1. The `atom:entry` can carry the information’s metadata, and the `atom:content` points to the information in an external document. In this case, the `atom:content` element is empty, and the machine-readable version is the only available version. The disadvantage of this approach is that no human-readable version is made available.
2. Similar to the previous version, but instead of pointing to the information from the `atom:content` element, the information is embedded. Since we are recommending XML, the embedded data does not have to be encoded and can be encoded as literal XML. The disadvantage of this approach is the same as for the previous approach; the fact that no human-readable version is made available.
3. If a human-readable version should be made available, it should be included in the `atom:content` element. This can be done by using `html` or `xhtml` types, where XHTML is preferable because it does not have to be escaped and can be processed as well-formed XML. The machine-readable content then can be made available by using an `atom:link` and pointing to it. Preferably, the URIs should follow a well-known template, and the URI should be persistent identifiers for this information.
4. Similar to the previous approach, XHTML can be embedded into the `atom:content` element, but the information is embedded into the XHTML fragment by using RDFa. This makes the feed self-contained, but has the disadvantage of not establishing a persistent URI for the information. If that is required, this technique can be combined with the previous one, supporting both RDFa and XML.

³<http://diveintomark.org/archives/2004/02/04/incompatible-rss> explains this variety of RSS interpretations that has been created by how RSS evolved.

```

<entry>
  <link rel="alternate" type="application/xml+stimulus"
    href="http://isd.ischool.berkeley.edu/stimulus/2009-029/dot/2009/3/10/fbga.xml"/>
  <id>http://isd.ischool.berkeley.edu/stimulus/2009-029/dot/2009/3/10/fbga.xml</id>
  <updated>2009-03-14T19:51:08.481-07:00</updated>
  <title>UC Berkeley, School of Information, ISD Program</title>
  <georss:point>37.871652 -122.258536</georss:point>
  <published>2009-03-10T00:00:00Z</published>
  <content type="xhtml">
    <div xmlns="http://www.w3.org/1999/xhtml">
      <table><tbody>
        <tr>
          <th align="right" valign="top">Recipient:</th>
          <td>UC Berkeley, School of Information, ISD Program (DUNS 1642990396473)</td>
        </tr>
        <tr>
          <th align="right" valign="top">Funding:</th>
          <td>$166,000.00</td>
        </tr>
        <tr>
          <th align="right" valign="top">CFDA:</th>
          <td>Design Recovery Transparency Web Services (2004031)</td>
        </tr>
        <tr>
          <th align="right" valign="top">Address:</th>
          <td>UC Berkeley, I School<br/>102 South Hall Road<br/>Berkeley, CA 94720</td>
        </tr>
        <tr>
          <th align="right" valign="top">Program Source:</th>
          <td>Agency B3; Account b3a9; Sub-Account 4ae</td>
        </tr>
      </tbody></table>
      <p>Link to <a href="http://isd.ischool.berkeley.edu/stimulus/2009-029/dot/2009/3/10/fbga.xml">
        XML version</a>.</p>
    </div>
  </content>
</entry>

```

Figure 4: First Entry of [dot/fbga.xml](#) Feed

Based on our recommendation that XML should be used as the data format for the information made available, we propose to recommend options 3 or 4 from the above list, with option 3 (linking to an XML document that has to be permanently available at a URI following a URI template) being required, and option 4 (adding RDFa to the content) being optional. We furthermore recommend that the `atom:content` should be required to be of type `xhtml`, and that it should be encouraged to add a link to the XML from with the XHTML.

One example for such a feed is [dot/fbga.xml](#) shown in Figure 8, the markup for the entry shown in Figure 6 (which is simply a visualization of this and other feeds) is shown in Figure 4. The XHTML in the entry's `atom:content` has been generated from the XML document at [dot/2009/3/10/fbga.xml](#) (which conforms to the XSD schema shown in Figure 2), and simply is a human-readable rendering of that information. In addition, the XML is linked to via `atom:link` as well as in the XHTML code, so that the

```

<fbga:allocation date="2009-03-10">
  <fbga:recipient>UC Berkeley, School of Information, ISD Program</fbga:recipient>
  <fbga:funding>166000</fbga:funding>
  <fbga:DUNS>1642990396473</fbga:DUNS>
  <fbga:CFDA>2004031</fbga:CFDA>
  <fbga:CFDA-title>Design Recovery Transparency Web Services</fbga:CFDA-title>
  <fbga:address>UC Berkeley, I School</fbga:address>
  <fbga:address>102 South Hall Road</fbga:address>
  <fbga:city code="4bac8">Berkeley</fbga:city>
  <fbga:county code="dbd">Alameda</fbga:county>
  <fbga:state code="CA">California</fbga:state>
  <fbga:zip>94720</fbga:zip>
  <fbga:agency>B3</fbga:agency>
  <fbga:account>b3a9</fbga:account>
  <fbga:sub-account>4ae</fbga:sub-account>
</fbga:allocation>

```

Figure 5: XML Document [dot/2009/3/10/fbga.xml](#) (Source for Figure 4)

XML can be easily found.

One important feature that is not implemented in the feeds we have set up is *Feed Paging and Archiving* [17]. We strongly recommend to make feed paging and archiving mandatory, so that the feeds are not just a temporary way of communicating that information has become available. Instead, the feeds should be available as persistent and permanent access points, so that accessing information via feeds can be done robustly and reliably. To make feed paging and archiving more predictable and easier to implement, we also recommend that the guidelines should at least recommend, or maybe even require, a naming scheme for paged and archived feeds. In such a publishing scenario of feeds as well as information items in XML being available via persistent URIs, it becomes significantly easier to build RESTful services based on these resources.

Atom does support a `atom:rights` element on feeds and entries for capturing licensing information, but this field is intended for human-readable text only. The *Atom License Extension* [27] adds a new link relation to Atom for linking to a license, either for a complete feed or for individual entries. Why we are not making any specific recommendation about the license being attached to the data being disseminated by the feeds discussed here, we strongly recommend to make licensing information available in a machine-readable way, using the standardized mechanism of the Atom licensing extension.

While *GeoRSS* [20] should not be required, it would be easy for agencies to geocode their feeds. The entry shown in Figure 4 uses GeoRSS markup to geocode the location of the allocation, and this has been done by using the address information found in Figure 5 (which is in a structured form that works well for geocoding) and geocoding it through a geocoding service.

8 Validation

The feed-based architecture of the Recovery Act guidelines represent an important development in making government processes more transparent and accountable. However, the approach and technologies may be somewhat unfamiliar to the IT personnel of many agencies. Rich and detailed guidance on implementing the XML formats and the feeds would be invaluable, as well as providing assistance in the form of templates, sample implementations, and easy ways to check for conformance. References to outstanding examples of successful implementations would also be useful.

A feed validation service can have a great impact in seeing this effort to success. The validation service would

1. verify that a tested stimulus feed adheres to requirements, and
2. provide clear error and warning messages to guide feed publishers in debugging.

The feed validation service can serve a secondary function of populating a registry of valid feeds. Thus, it can both facilitate publication of feeds that meet OMB requirements and it can help in their discovery. The schemas presented in Section 6 would be a starting point for implementing validation of the XML documents. Feed validators are available [online](#) or as [download](#) and can be used as a starting point to validating feeds. Based on guideline recommendations, feed validation should be augmented with more specific validation and error and warning messages, so that agencies can get clear feedback about their conformance.

Agencies could also get support in the form of centrally provided tools for helping them to produce XML. This could be done by providing simple Web forms and producing XML as a result. In a more sophisticated scenario, it could also be done by providing *XForms* [6], which could either be used in server-side scenarios, if required as central services, or in client-side scenarios for supporting local (and possibly offline) data entry.

The exact toolset that would best be used for such a validation suite would have to be determined later, but it should be an integral part of the guidelines that agencies are required to check against the validation service. Apart from agencies actively checking their implementation conformance, guideline compliance could constantly be monitored (since the feed architecture is designed to support fully automated access to all information), and non-conforming agencies could be alerted by a (fully or partly automated) watchdog service.

9 Feed Usage

One of the main goals of the guidelines presented here is to make the information about recovery spending as easily accessible as possible. The combination of syndication based on feeds, and XML for machine-readable data, achieves the goals of open and accessible data formats. Using a simple XSLT transformation ([feeds.xslt](#)), we have experimented with simple ways to use the information carried in the feeds.

Figure 6 uses [timemap](#), a combination of Google Maps and [SIMILE Timeline](#). In this case, instead of visualizing the feeds directly, [feeds.kml](#) has been used, which also has been generated by [feeds.xslt](#). The *Keyhole Markup Language (KML)* [21] is the data format supported by Google Maps and Google Earth, and recently has been adopted by the *Open Geospatial Consortium (OGC)*. Figure 7 shows the same KML document visualized in Google Earth.

Figure 8 is a screen shot of how a standard browser displays one of the feeds ([dot/fbga.xml](#)). Since the feed's entries contain XHTML (as shown in Figure 4), the display is human-readable and conveys the most important information. In addition, the entry as well as the content's XHTML link to the XML document underlying this entry's content (the XML is shown in Figure 5), and therefore the data is accessible both in human-readable as well as in machine-readable form.

We also want to emphasize that the uses and visualizations discussed and demonstrated here are by no means exhaustive. Civil-society organizations, researchers, journalists, and others will no doubt develop their own innovative analyses of these data and combine these data with other publicly available information (such as U.S. census data, information derived from Wikipedia, etc.). The possibilities for novel applications, visualizations, and analyses are virtually endless. The main point of our discussion is not to promote any one visualization. Instead, we hope to suggest approaches that make recovery data easily accessible for the widest number of applications.

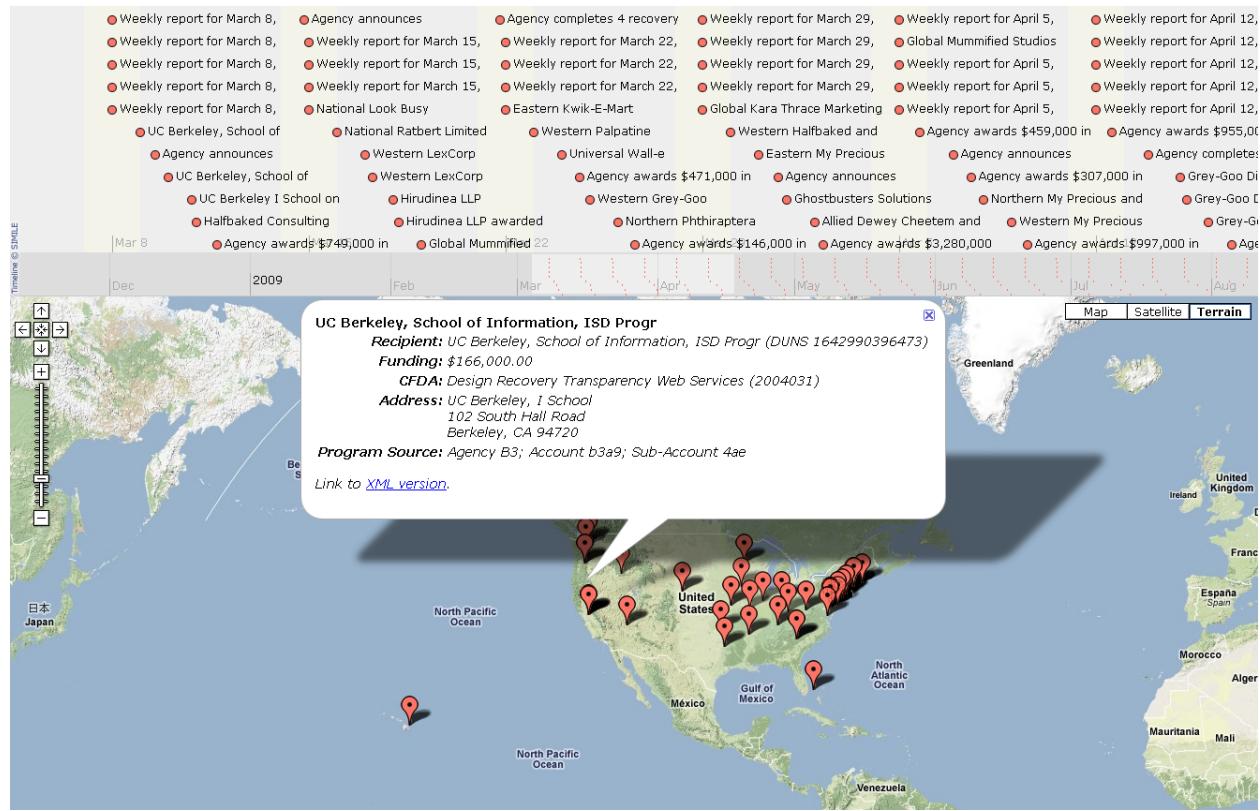


Figure 6: Timemap Display of the Aggregated Example Feeds ([feeds.kml](#))

10 Conclusions

As discussed above, the feeds represent a major positive development in making government processes more open to citizen oversight. However, the scale of the US government is vast, and includes many federal, state, and local agencies with varying levels of technical capacity and expertise. Re-engineering government information disclosure processes to meet these transparency goals is a tremendous undertaking.

While specific guidelines and validation services, such as those proposed in this document, can do much to promote better information disclosures, many agencies will still lack the ability to implement these measures. Therefore, we also propose that the [recovery.gov](#) site organizes a professional support network so that agencies can find the expertise needed to implement these important transparency measures. This support network can include information schools, computer science departments, and other academic units, as well as civil society groups such as the *Sunlight Foundation* and the *Electronic Frontier Foundation (EFF)*. Engagement with the academic and civil society communities can help not only implement transparency measures such as the feeds, but will also demonstrate the value of enhanced citizen participation and oversight in our nation's economic recovery efforts.

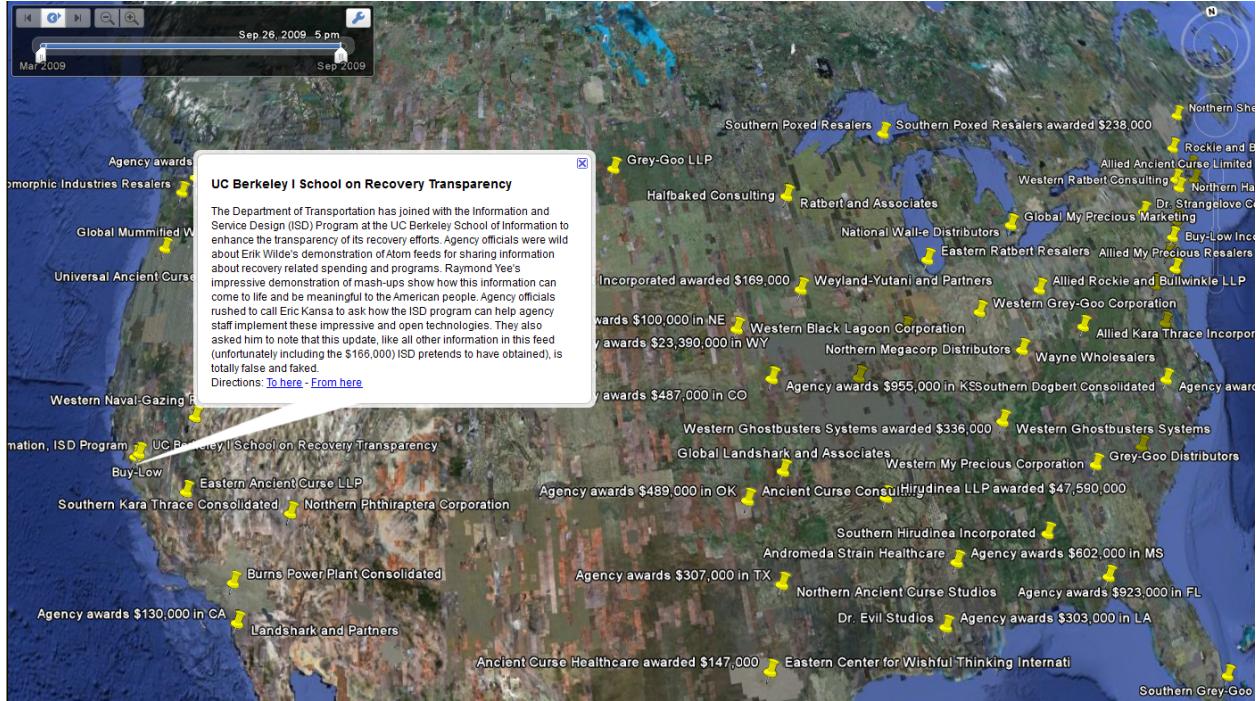


Figure 7: Google Earth Display of the Aggregated Example Feeds ([feeds.kml](#))

References

- [1] BEN ADIDA, MARK BIRBECK, SHANE McCARRON, and STEVEN PEMBERTON. RDFa in XHTML: Syntax and Processing — A Collection of Attributes and Processing Rules for Extending XHTML to Support RDF. World Wide Web Consortium, Recommendation REC-rdfa-syntax-20081014, October 2008.
- [2] JOSÉ M. ALONSO and KEVIN NOVAK. Improving Access to Government through Better Use of the Web. World Wide Web Consortium, Working Draft WD-egov-improving-20090310, March 2009.
- [3] DAVE BECKETT. RDF/XML Syntax Specification (Revised). World Wide Web Consortium, Recommendation REC-rdf-syntax-grammar-20040210, February 2004.
- [4] PAUL V. BIROL and ASHOK MALHOTRA. XML Schema Part 2: Datatypes Second Edition. World Wide Web Consortium, Recommendation REC-xmleschema-2-20041028, October 2004.
- [5] SCOTT BOAG, DONALD D. CHAMBERLIN, MARY F. FERNÁNDEZ, DANIELA FLORESCU, JONATHAN ROBIE, and JÉRÔME SIMÉON. XQuery 1.0: An XML Query Language. World Wide Web Consortium, Recommendation REC-xquery-20070123, January 2007.
- [6] JOHN M. BOYER. XForms 1.0 (Third Edition). World Wide Web Consortium, Recommendation REC-xforms-20071029, October 2007.

Formula Block Grant Allocation Feed 28 Total

<p>Andromeda Strain Healthcare Department of Transportation Apr 20, 12:00 AM</p> <p>Recipient: Andromeda Strain Healthcare (DUNS 127248644714) Funding: \$602,000.00 CFDA: Belly-button Lint Consolidation Study Initiative (2132623) Address: P.O. Box 139 Suite: 3888 Jackson, MS 39205 Program Source: Agency B3; Account b3a9; Sub-Account 025</p> <p>Link to XML version.... Read more...</p> <hr/> <p>Western Naval-Gazing Research and Associates Department of Transportation Apr 19, 12:00 AM</p> <p>Recipient: Western Naval-Gazing Research and Associates (DUNS 2327508099793) Funding: \$24,320,000.00 CFDA: Process Irradiation Prevention Initiative (1613444) Address: 101 N. Carson Street Suite: 6577 Carson City, NV 89701 Program Source: Agency B3; Account b3a9; Sub-Account 360</p> <p>Link to XML version.... Read more...</p> <hr/> <p>Grey-Goo Systems Department of Transportation Apr 16, 12:00 AM</p> <p>Recipient: Grey-Goo Systems (DUNS 1272438213705) Funding: \$191,000.00 CFDA: UC Berk. ISchool (Wilde, Kansa, Yee) Recovery Atom Feeds (1026087) Address: 1900 Kanawha Boulevard, E. Suite: 6122 Charleston, WV 25305 Program Source: Agency B3; Account b3a9; Sub-Account 1b7</p> <p>Link to XML version.... Read more...</p> <hr/> <p>Dunder Mifflin Paper Company Systems Department of Transportation Apr 12, 12:00 AM</p> <p>Recipient: Dunder Mifflin Paper Company Systems (DUNS 1986857519745) Funding: \$955,000.00 CFDA: Process Termination Evaluation Project (1421525) Address: 300 SW 10th Ave., Ste. 212S Suite: 8460 Topeka, KS 666121590 Program Source: Agency B3; Account b3a9; Sub-Account 3da</p>	<p>Search Articles: <input type="text"/> Search</p> <p>Article Length: <input type="range"/></p> <p>Sort By: Date Title Source New</p> <hr/> <p>Recent Articles: All Today Yesterday Last Seven Days This Month Last Month</p> <hr/> <p>Source: Formula Block Gra...</p> <hr/> <p>Actions: Update Now Mail Link to This Page Add Bookmark...</p>
---	--

Figure 8: Formula Block Grant Allocation Feed in Browser ([dot/fbga.xml](#) in Safari)

[7] TIM BRAY, JEAN PAOLI, C. MICHAEL SPERBERG-MCQUEEN, EVE MALER, and FRANÇOIS YERGEAU. Extensible Markup Language (XML) 1.0 (Fifth Edition). World Wide Web Consortium, Recommendation REC-xml-20081126, November 2008.

[8] DAN BRICKLEY and RAMANATHAN V. GUHA. RDF Vocabulary Description Language 1.0: RDF

- Schema. World Wide Web Consortium, Recommendation REC-rdf-schema-20040210, February 2004.
- [9] DOUGLAS CROCKFORD. The application/json Media Type for JavaScript Object Notation (JSON). Internet RFC 4627, July 2006.
 - [10] MARY F. FERNÁNDEZ, ASHOK MALHOTRA, JONATHAN MARSH, MARTON NAGY, and NORMAN WALSH. XQuery 1.0 and XPath 2.0 Data Model (XDM). World Wide Web Consortium, Recommendation REC-xpath-datamodel-20070123, January 2007.
 - [11] ROY THOMAS FIELDING and RICHARD N. TAYLOR. Principled Design of the Modern Web Architecture. *ACM Transactions on Internet Technology*, 2(2):115–150, May 2002.
 - [12] JOE GREGORIO and BILL DE HÓRA. The Atom Publishing Protocol. Internet RFC 5023, October 2007.
 - [13] IAN HICKSON and DAVID HYATT. HTML 5 — A Vocabulary and Associated APIs for HTML and XHTML. World Wide Web Consortium, Working Draft WD-html5-20090212, February 2009.
 - [14] INTERNATIONAL ORGANIZATION FOR STANDARDIZATION. Information Technology — Document Schema Definition Languages (DSDL) — Part 2: Grammar-based Validation — RELAX NG. ISO/IEC 19757-2, November 2003.
 - [15] MICHAEL KAY. XSL Transformations (XSLT) Version 2.0. World Wide Web Consortium, Recommendation REC-xslt20-20070123, January 2007.
 - [16] GRAHAM KLYNE and JEREMY J. CARROLL. Resource Description Framework (RDF): Concepts and Abstract Syntax. World Wide Web Consortium, Recommendation REC-rdf-concepts-20040210, February 2004.
 - [17] MARK NOTTINGHAM. Feed Paging and Archiving. Internet Draft draft-nottingham-atompub-feed-history-11, June 2007.
 - [18] MARK NOTTINGHAM. Link Relations and HTTP Header Linking. Internet Draft draft-nottingham-http-link-header-04, February 2009.
 - [19] MARK NOTTINGHAM and ROBERT SAYRE. The Atom Syndication Format. Internet RFC 4287, December 2005.
 - [20] OPEN GEOSPATIAL CONSORTIUM. An Introduction to GeoRSS: A Standards Based Approach for Geo-enabling RSS feeds. OGC 06-050r3, July 2006.
 - [21] OPEN GEOSPATIAL CONSORTIUM. OGC KML. OGC 07-147r2, Version 2.2.0, April 2008.
 - [22] PETER R. ORSZAG. Initial Implementing Guidance for the American Recovery and Reinvestment Act of 2009. Office of Management and Budget Memorandum M-09-10, February 2009.
 - [23] PETER F. PATEL-SCHNEIDER, PATRICK HAYES, and IAN HORROCKS. OWL Web Ontology Language Semantics and Abstract Syntax. World Wide Web Consortium, Recommendation REC-owl-semantics-20040210, February 2004.
 - [24] STEVEN PEMBERTON. XHTML 1.0: The Extensible HyperText Markup Language (Second Edition). World Wide Web Consortium, Recommendation REC-xhtml1-20020801, August 2002.
 - [25] DAVE RAGGETT, ARNAUD LE HORS, and IAN JACOBS. HTML 4.01 Specification. World Wide Web Consortium, Recommendation REC-html401-19991224, December 1999.

- [26] JOHN SCHNEIDER and TAKUKI KAMIYA. Efficient XML Interchange (EXI) Format 1.0. World Wide Web Consortium, Working Draft WD-exi-20080919, September 2008.
- [27] JAMES M. SNELL. Atom License Extension. Internet RFC 4946, July 2007.
- [28] HENRY S. THOMPSON, DAVID BEECH, MURRAY MALONEY, and NOAH MENDELSOHN. XML Schema Part 1: Structures Second Edition. World Wide Web Consortium, Recommendation REC-xmlschema-1-20041028, October 2004.
- [29] ERIK WILDE. The Plain Web. In *Proceedings of the First International Workshop on Understanding Web Evolution (WebEvolve2008)*, pages 79–83, Beijing, China, April 2008.