RDA Data Type Registries Working Group Output

April 2015

Executive Summary

The RDA Data Type Registries (DTR) Working Group (WG) was approved at the first RDA Plenary (March 2013, Gothenburg, Sweden). The basic goal of this group was to aid data sharing efforts through improved data typing, specifically to make clear the details and assumptions buried in other peoples' data. This was seen primarily as a problem in defining a data model appropriate to a wide potential collection of data types, prototyping that model in a registry, and developing a federation strategy across multiple registry instances, all following an analysis of use cases and related efforts. Larry Lannom of CNRI and Daan Broeder of MPI took on the co-chair tasks.

The WG attracted a large degree of interest, both at the conceptual level and in the details of the prototype, which the co-chairs took as confirmation of the relevance of the issue. The prototype was successfully deployed and a number of use cases implemented, allowing us to gain experience with DTR issues and discuss the community's reactions and comments. The scope of the issues involved, however, proved to be too broad especially with respect to community specific typing needs for a single WG and therefore a follow-on WG, provisionally named Data Typing, will be proposed. The follow-on WG will primarily try to identify the data model that will allow people to specify and represent data types from select communities. In the end, the outcomes of the DTR WG can be summarized as:

- Confirmation that detailed and precise data typing is a key consideration in data sharing and reuse and that a federated registry system for such types is highly desirable and needs to accommodate each community's own requirements
- Deployment of a prototype registry implementing one potential data model, against which various use cases can be tested
- Involvement of multiple ongoing scientific data management efforts, across a variety of domains, in actively planning for and testing the use of data types and associated registries in their data management efforts
- Integration with one additional RDA WG (Persistent Identifier Types) and at least one Interest Group (RDA/CODATA Materials Data, Infrastructure & Interoperability IG)
- Development of a set of questions that require further consideration before a detailed recommendation on data typing can be issued.

Finally, we believe that the DTR WG served as an excellent example of the benefits that RDA can and will bring to solving the problems of data sharing, by bringing together what would otherwise be disparate domain-specific groups to focus on common problems at the data level as opposed to the domain level. The remainder of this report will provide details on the outcomes summarized above.

What is the Problem Addressed by the Working Group?

Automated processing of large amounts of scientific data, especially across domains, requires that the data can be parsed without human intervention[1]. Within a given domain that functionality can simply be built into the software, e.g., the piece of information that appears in this location is always a temperature reading in centigrade or, at a different level of granularity, this data set is structured according to Domain Standard A including base types X, Y, and Z where the base types are things like temperature readings in centigrade. This knowledge, easily available within a given domain or a set of closely related research groups, can be built into domain specific processing workflows. But outside of that domain or environment the 'local knowledge' approach can begin to fail and more precision in associating data with the information needed to process it is required. This also applies across time as well as domains. What is well known today may be less well-known twenty years hence but age will not necessarily reduce the value of a data set and indeed may increase it.

We are using the term 'type' here as the characterization of data structure, contexts, and assumptions. Also, we expect that such 'typing' to be applicable at multiple levels of granularity, from individual observations up to and including large data sets. Optimizing the interactions among all of the producers and consumers of digital data requires that those types be defined and permanently associated with the data they describe. Further, the utility of those types requires that they be standardized, unique, and discoverable. The goal of this working group was to address these issues through evaluation of use cases, existing efforts, and potential infrastructural solutions, including the development of one or more type registries.

Simply listing and describing types in human readable form, say in one or more open access wikis, would certainly be a good start. But full realization of the potential of types in automated data processing will benefit from a common form of machine-readable description of types, i.e., a data model and common expression of that data model. This would not only aid in discoverability but also in the analysis of relations among types and evaluation of overlap and duplication as well as possible bootstrapping of automated data processing in some cases.

Types will be at different levels of granularity, e.g., individual observations, a set of observations composed into a time series, a set of time series describing a complex phenomenon, and so forth. The ease of composing lower level, or base, types into more complex composite types would be an advantage of a well-managed type system.

---

[1] Note that 'without human intervention' does not mean full automation for reuse but that each instance of reuse does not require complete human re-interpretation of the data.

An immediate and compelling use case for a managed system of types comes directly out of persistent identifiers (PIDs) for data sets. Accessing a piece of data via a PID, either as a direct reference or as the result of a search, requires resolving the identifier to get the information needed to access the data. This information must be understandable by the client, whether that client is a human or a machine, in order for the client to act on it. For a machine, it must be explicitly typed. A type registry for PID information types would appear to be an early requirement for coherent management of scientific data.

Finally, assigning PIDs to types, not to be confused with assigning types to PID resolution results, would aid in their management and use. All of the arguments for using persistent identifiers for important digital information that must remain accessible over long periods of time will apply equally well to whatever form of records are kept for data types.

What Was the Goal of the Working Group and What Did it Accomplish?

Overall Goal

The overall goal of the group was to define a data model for data types, to prototype its use in a functioning registry, to experiment with some domain-specific data type sets, and to define a federation strategy across multiple registries. The assumed benefit of this, on which there was fairly general agreement in Plenary Breakout sessions, which is where most of the discussions took place, was that precise typing of data sets and collections, combined with one or more registries that define those types in a standard fashion, would benefit every sector of data management, especially interoperability and reuse. The WG further agreed that it would not attempt to define the methods of association of data and type but would work towards a standard approach for registering and discovering types as well adding value to their use by linking types to services.

Generic Use Cases

The WG considered a variety of use cases in building a general approach to dealing with data types. Figure 1, below, illustrates these at a high level (this image was used as the WG Poster at RDA Plenary 2). We assume that types are embedded or otherwise associated with the typed data. For human understanding and/or machine processing any registered type can be sent as a query to the set of federated type registries. Each type is identified with a resolvable persistent identifier (PID) that leads to the correct registry. The type description that comes back from the registry can be used by humans to understand and use the data or by machines to process the data. In addition to structural and descriptive details, each type can further be associated with one or more services, such that data of type X could be sent to service Y to get output of type Z.

In addition to the generic case of using a type registry to explicate the details of a given piece or set of data in order to further process or reuse that data, types could be used in discovery, e.g., find all data sets across a given set of repositories

containing data of a given type which becomes of course most useful when the semantics of a type are defined. A third generic use case was also identified – use type descriptions to validate data acquisition, i.e., does a give piece of data such as the report of an experiment contain all of the elements in all of the correct units to meet the requirements of a given data collection exercise.

Data Type Registries Working Group

Data Types are:

- Characterizations of data at any level of granularity
- Identified, defined, and registered

Registered Data Types are used for:

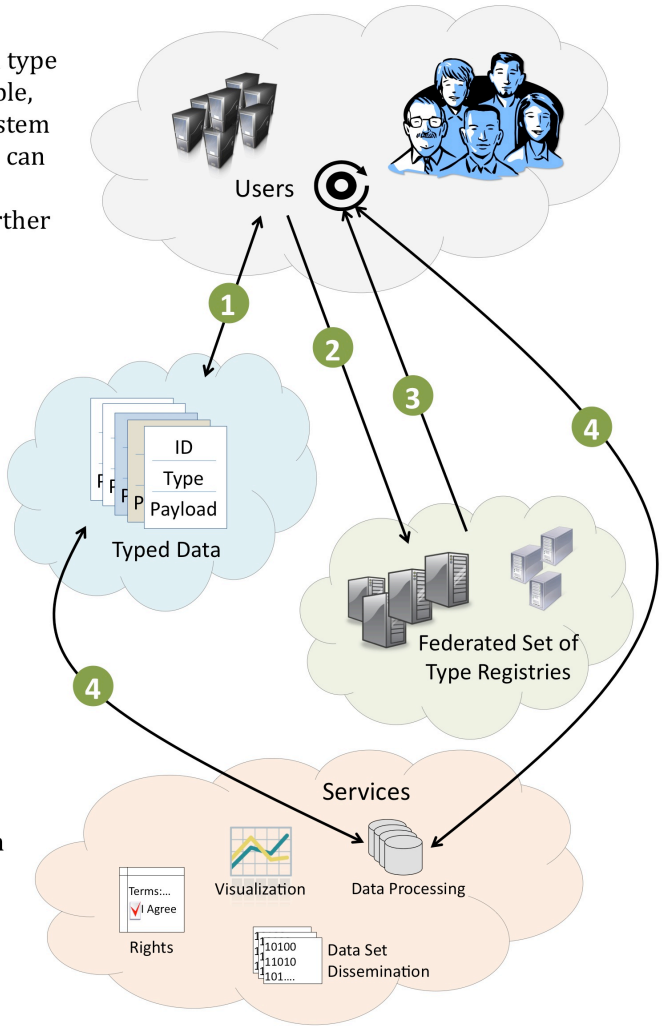
- Interpreting data (by humans)
- Processing data (by machines)

1. Users (processors or people) encounter data of an unknown type as part of an id/type/value triple, e.g., an identifier resolution system or a data repository. Registries can explicate those types and/or provide structure to enable further processing of the data.

2. Users query Type Registries with the unknown types.

3. Type Registries response (model under discussion) includes type definitions, relationships, properties and, potentially, pointers to relevant services or software that can be used to interpret or process the typed data or used to request an external service.

4. Optionally, the typed data or a reference to the typed data can be sent to a type-appropriate service or application to be rendered or processed as needed.



Alternate use of a type ecology:

- Search for data by type after using a registry to discover types of use to you.
- Use types as short-cut for dependent services to figure out if a given data object has what is needed for processing.

Prototype Registry

The first instance of Data Type Registry was publically released in February of 2014. As of this writing of this document (April of 2015) the most recent version of that prototype remains publically available at

http://typeregistry.org/registrar/

As stated on that web page this registry is NOT

• A canonical registry of scientific definitions. The Type records in the system are just examples recorded to evaluate the usefulness of a Data Type Registry.

• A technology alternative to existing Data Type and Format Registries or the broader Semantic Web effort.

• A showcase of all possible assumptions applicable to various domains and disciplines.

We emphasize this because of the number of comments by those who expected to find a full-fledged data type registry. This site is primarily intended to demonstrate how such a registry might be used and is open to experimentation, which results in some quite uneven content.

The prototype implements an API, summarized by the following table. Detailed examples are on the registry site.

Resource	**Description**
GET /objects/<id>	Retrieves an object by id.
POST /objects/?type=<type>	Create an object by type.
PUT /objects/<id>	Update an object by id.
DELETE /objects/<id>	Delete an object by id.
GET /objects/?query=<query>&pageNum=<num>&pageSize=<size>	Search for objects.

The 'About' section of the above referenced site explains the connection of this prototype to other projects, primarily those associated with CNRI, the home organization for one of the co-chairs and the current source of the registry software. This information is provided for contextual and historical reasons, not because the co-chairs regard this as a solely CNRI project going forward. We regard this as another example of the benefits that any organization can realize by bringing existing work into the RDA to expose it to a larger community, enhancing both the RDA community and the organization that brings a project into RDA.

Data Model

The current version of the Data Type Registry data model, which specifies the elements constituting a data type description, is shown as a JSON schema in Appendix A of this document. An example of a type record in hydrology is shown in Appendix B.

The basic elements of this model are:

- Identifier
- Type Name
- Human Readable Description
- Provenance (including contributors/source, creation date, modification date)
- Related Standards and Recommendations
- Expected Uses
- Representations and Semantics
- Properties Specific to this Type
- Relationships to Other Types

Fundamental to this data model and any coherent federation system across type registries is the notion of primitive types and extended types. Primitive types, including those known to information technologists, such as Boolean and string, those known to physical scientists, such as mass and temperature, and those standard observation type of record structures known to other domains, such as time series and survey skip patterns, are presumed to be useful in the creation of extended types, which are composed using primitives combined with properties specific to the extended types. Uniformity and reuse of these primitive types, assuming it can be done, would seem to be key to lowering the level of effort needed to establish a useful data type ecology.

This data model is the least settled part of the WG activities and is seen as a major focus of any follow-on group.

Related Efforts

The term 'registry' has many different connotations and meanings across various information management activities and domains. The IANA Mime type registry is a clear example of an existing effort that the WG needed to recognize and leverage in pursuit of its goals. Many other examples of format and service registries, controlled vocabularies, and ontologies were brought to the group's attention over the course of the effort, both during the Plenary sessions and in email and forum discussions. A very few examples brought to our attention include:

- XBRL International Inc. for Business
- SCIDIP-ES
- DataNet Federation Consortium Format Registry
- DataONE Object Format Registry
- Universal Data Format Registry
- PRONOM format registry

- Various activities in organizations such as the Alliance for Permanent Access and the Digital Curation Centre
- Schema.org
- Uniform Content Descriptors (UCD) used in astronomy

The DTR WG and any follow-on groups cannot and should not attempt to replicate the work of these and other projects but can leverage that work and attempt to provide a common approach to discovering and using the work that they have and will continue to produce. In some ways a group of federated type registries can be seen as multiplexing across existing efforts, providing a common API and system of unique identifiers for referential integrity of existing efforts and providing a low barrier of entry to adding to and enhancing existing work. The 'Related Standards and Recommendations' component of the proposed data model would provide the connectivity.

Ongoing Efforts

As of the writing of this document (April 2015) there are a number of ongoing efforts to integrate a Data Type Registry of the form currently available as a prototype. These include the following organizations and projects:

- The Materials Genome Initiative
- The Deep Carbon Observatory
- The International Digital Object Identifier (DOI) Foundation (IDF)
- EUDAT
- The U.S. National Institute of Standards and Technology (NIST)
- The U.S. Census Bureau
- The U.S. Department of Agriculture (USDA)

In addition there are a number of ongoing discussions that may yet yield additional projects. The current co-chairs hope that these efforts can be carried over into the follow-on activities and new WG.

Data Type Registry Requirements

A final set of requirements for a federated set of data type registries will need to wait for a core data model, an outcome which we now look for from a follow-on group. That said, a number of recommendations can be made based on the work of this WG, with the hope that this list will be discussed and further validated in future RDA efforts:

- Every type in a data type registry must be identified with a resolvable persistent identifier
- Types should reference related standards and recommendations in order to leverage existing efforts
- Primitive types should be established and used, when possible, in the construction of more complex types
- A common API should be available across all type registries

- Type registries should be federated such that a single service can search across all known registries or some defined subset[2]
- Type registries should include or enable referencing related services based on types
- The establishment of a data type registry for any community should be subject only to the needs and requirements of that community, i.e., there should be no higher level governance beyond the maintenance of whatever standards and processes are needed for effective federation across type registries

[2] Here we may encounter the issue of great unevenness of searchable items across nodes of a federation. One DTR prototype instance, for example, already has 40K defined types, which could easily overwhelm equally or more significant types in smaller DTRs.

Appendix A: Data Type Registry Data Model Expressed as a JSON schema


```
{
  "type": "object",
  "required": [
    "name",
    "description"
  ],
  "properties": {
    "identifier": {
      "type": "string",
      "net.cnri.registrar": {
        "autoGeneratedFieldName": "handle"
      }
    },
    "name": {
      "type": "string",
      "maxLength": 128,
      "title": "Type Name",
      "net.cnri.registrar": {
        "showInPreview": true,
        "isPrimary": true
      }
    },
    "description": {
      "type": "string",
      "format": "textarea",
      "maxLength": 2048,
      "title": "Description",
      "net.cnri.registrar": {
        "showInPreview": true,
        "excludeTitle": true
      }
    },
    "standards": {
      "type": "array",
      "format": "table",
      "title": "Applicable Standards or Recommendations",
      "uniqueItems": true,
      "items": {
        "type": "object",
        "title": "Standard",
        "required": [
          "issuer",
          "name"
        ],
        "properties": {
          "issuer": {
            "title": "Issued By",
            "type": "string",
            "net.cnri.registrar": {
              "suggestedVocabulary": [
                "DTR",
```

```
            "ISO",
            "W3C",
            "ITU",
            "RFC"
          ]
        }
      },
      "name": {
        "title": "Standard Name",
        "type": "string",
        "maxLength": 1024,
        "description": "TID or standard number/name"
      },
      "details": {
        "title": "Details",
        "type": "string",
        "format": "textarea",
        "maxLength": 2048
      },
      "natureOfApplicability": {
        "title": "Nature of Applicability",
        "type": "string",
        "enum": [
          "extends",
          "constrains",
          "specifies",
          "depends"
        ]
      }
    }
  },
  "provenance": {
    "type": "object",
    "title": "Provenance",
    "properties": {
      "contributors": {
        "type": "array",
        "format": "table",
        "title": "Contributors of this Record",
        "items": {
          "title": "Contributor",
          "type": "object",
          "required": [
            "identifiedUsing",
            "name"
          ],
          "properties": {
            "identifiedUsing": {
              "title": "Identified Using",
              "type": "string",
              "net.cnri.registrar": {
                "suggestedVocabulary": [
                  "Handle",
                  "ORCID",
```

```
                "URL",
                "Text"
              ]
            }
          },
          "name": {
            "title": "Name",
            "type": "string",
            "maxLength": 2048
          },
          "details": {
            "title": "Details",
            "type": "string",
            "format": "textarea",
            "maxLength": 1024
          }
        }
      }
    },
    "creationDate": {
      "title": "Creation Date",
      "type": "string",
      "format": "datetime",
      "net.cnri.registrar": {
        "autoGeneratedFieldName": "creationDate"
      }
    },
    "lastModificationDate": {
      "title": "Last Modification Date",
      "type": "string",
      "format": "datetime",
      "net.cnri.registrar": {
        "autoGeneratedFieldName": "modificationDate"
      }
    }
  }
},
"expectedUses": {
  "type": "array",
  "maxItems": 3,
  "format": "table",
  "title": "Expected Uses",
  "items": {
    "type": "string",
    "title": "Use",
    "format": "textarea",
    "maxLength": 4096
  }
},
"representationsAndSemantics": {
  "type": "array",
  "format": "table",
  "title": "Representations and Semantics",
  "uniqueItems": true,
  "items": {
```

```
        "type": "object",
        "title": "Representation and Semantic Expression",
        "required": [
          "expression",
          "value"
        ],
        "properties": {
          "expression": {
            "title": "Expression",
            "type": "string",
            "net.cnri.registrar": {
              "suggestedVocabulary": [
                "Format",
                "Character Set",
                "Encoding",
                "Measurement Unit"
              ]
            }
          },
          "value": {
            "title": "Value",
            "description": "Unicode, UTF-8, Meter, etc.",
            "type": "string",
            "maxLength": 1024
          },
          "details": {
            "title": "Details",
            "type": "string",
            "format": "textarea",
            "maxLength": 2048
          }
        }
      }
    },
    "properties": {
      "type": "array",
      "title": "Properties",
      "description": "Type dependencies used for expressing how this
type is built from other types",
      "items": {
        "type": "object",
        "headerTemplate": "{{self.name}}",
        "title": "Property",
        "required": [
          "identifier",
          "name"
        ],
        "properties": {
          "name": {
            "type": "string",
            "description": "Name assigned to dependent type in this
context",
            "title": "Name",
            "maxLength": 256
          },
```

```
          "identifier": {
            "type": "string",
            "title": "TID of Existing Data Type",
            "net.cnri.registrar": {
              "handleReferenceType": "dataType",
              "handleReferenceName": "{{../name}}"
            }
          },
          "representationsAndSemantics": {
            "type": "array",
            "format": "table",
            "title": "Representations and Semantics",
            "description": "Restrictions on representations and
semantics of the dependent type in this context",
            "uniqueItems": true,
            "items": {
              "type": "object",
              "title": "Representation and Semantic Expression",
              "required": [
                "expression",
                "value"
              ],
              "properties": {
                "expression": {
                  "title": "Expression",
                  "type": "string",
                  "net.cnri.registrar": {
                    "suggestedVocabulary": [
                      "Format",
                      "Character Set",
                      "Encoding",
                      "Measurement Unit"
                    ]
                  }
                },
                "value": {
                  "title": "Value",
                  "description": "Unicode, UTF-8, Meter, etc.",
                  "type": "string",
                  "maxLength": 1024
                },
                "details": {
                  "title": "Details",
                  "type": "string",
                  "format": "textarea",
                  "maxLength": 2048
                }
              }
            }
          }
        }
      }
    },
    "relationships": {
      "type": "array",
```

```
      "format": "table",
      "title": "Experimental. Likely to change soon",
      "description": "Intent: How the properties are related to each
other, e.g., grouping of properties, cardinality, etc., should be
captured here.",
      "items": {
        "type": "object",
        "title": "Relationship",
        "required": [
          "name",
          "relativeNames"
        ],
        "properties": {
          "name": {
            "title": "Name of Relationship",
            "type": "string",
            "maxLength": 256
          },
          "relativeNames": {
            "type": "array",
            "minItems": 1,
            "format": "table",
            "title": "Relative Names",
            "items": {
              "type": "string",
              "title": "Name of Property"
            }
          },
          "details": {
            "title": "Details",
            "type": "string",
            "format": "textarea",
            "maxLength": 2048
          }
        }
      }
    }
  }
}
```

Appendix B: Data Type Registry Example

General:

identifier: "11314.3/6debc53338e99ff15731"

name: "Stream Gauge"

description: "Information that defines stream discharge at a specific location and time interval. Useful for the geosciences community."

Standards:

issuer: "ISO"; *name*: "4375:2000"; *nature of applicability*: "depends"

Provenance:

contributors

identified using: "Text"; name: "Mostafa Elag"; details: "A Researcher in the geosciences community from UIUC."

Identified using : "Text"; name: "Giridhar Manepalli"; details: "A data infrastructure expert from CNRI."

Creation date: "2014-08-07T04:28:21.479Z"

Last modification date: "2014-09-08T15:28:00.733Z"

Expected Uses:

"Used for comparing outputs of surface runoff discharge models as applied to data pertaining to a specific watershed."

Representation And Semantics:

expression: "Measurement Unit", *value*: "Cubic Meter per Second"

Properties:

name: "value"; identifier: "11314.3/f0f2c4382dcf8d257462";

name: "coordinate"; identifier : "11314.3/4102c3ebe68bed21d644"

name: "timestamp"; identifier: "11314.3/6386f4ebd23e9baace50"

Relationships (experimental section):

name: "Primary Key"; relative names: ["value"]