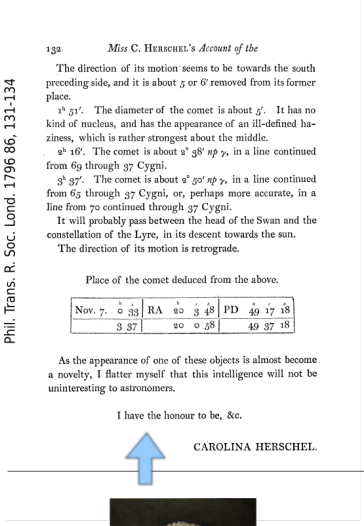
Creating and Manipulating Universal Metadata Definitions

James Hester

Australian Nuclear Science and Technology Organisation

Setting the scene



132 *Miss* C. HERSCHEL's *Account of the*

The direction of its motion seems to be towards the south preceding side, and it is about 5 or 6′ removed from its former place.

1h 51′. The diameter of the comet is about 5′. It has no kind of nucleus, and has the appearance of an ill-defined haziness, which is rather strongest about the middle.

2h 16′. The comet is about 2° 38′ *np* γ, in a line continued from 69 through 37 Cygni.

3h 37′. The comet is about 2° 50′ *np* γ, in a line continued from 65 through 37 Cygni, or, perhaps more accurate, in a line from 70 continued through 37 Cygni.

It will probably pass between the head of the Swan and the constellation of the Lyre, in its descent towards the sun.

The direction of its motion is retrograde.

Place of the comet deduced from the above.

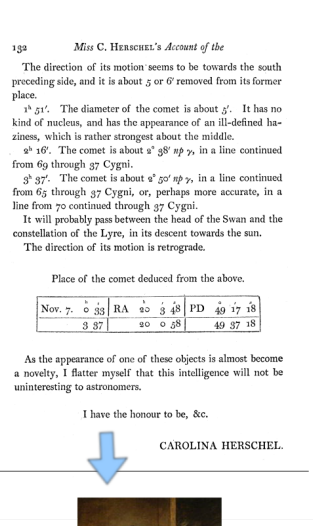
Nov. 7.	0	33	RA	20	3	48	PD	49	17	18
	3	37		20	0	58		49	37	18

As the appearance of one of these objects is almost become a novelty, I flatter myself that this intelligence will not be uninteresting to astronomers.

I have the honour to be, &c.

CAROLINA HERSCHEL.

Phil. Trans. R. Soc. Lond. 1796 86, 131-134



132 *Miss* C. HERSCHEL's *Account of the*

The direction of its motion seems to be towards the south preceding side, and it is about 5 or 6′ removed from its former place.

1h 51′. The diameter of the comet is about 5′. It has no kind of nucleus, and has the appearance of an ill-defined haziness, which is rather strongest about the middle.

2h 16′. The comet is about 2° 38′ *np* γ, in a line continued from 69 through 37 Cygni.

3h 37′. The comet is about 2° 50′ *np* γ, in a line continued from 65 through 37 Cygni, or, perhaps more accurate, in a line from 70 continued through 37 Cygni.

It will probably pass between the head of the Swan and the constellation of the Lyre, in its descent towards the sun.

The direction of its motion is retrograde.

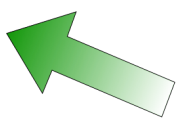
Place of the comet deduced from the above.

Nov. 7.	0	33	RA	20	3	48	PD	49	17	18
	3	37		20	0	58		49	37	18

As the appearance of one of these objects is almost become a novelty, I flatter myself that this intelligence will not be uninteresting to astronomers.

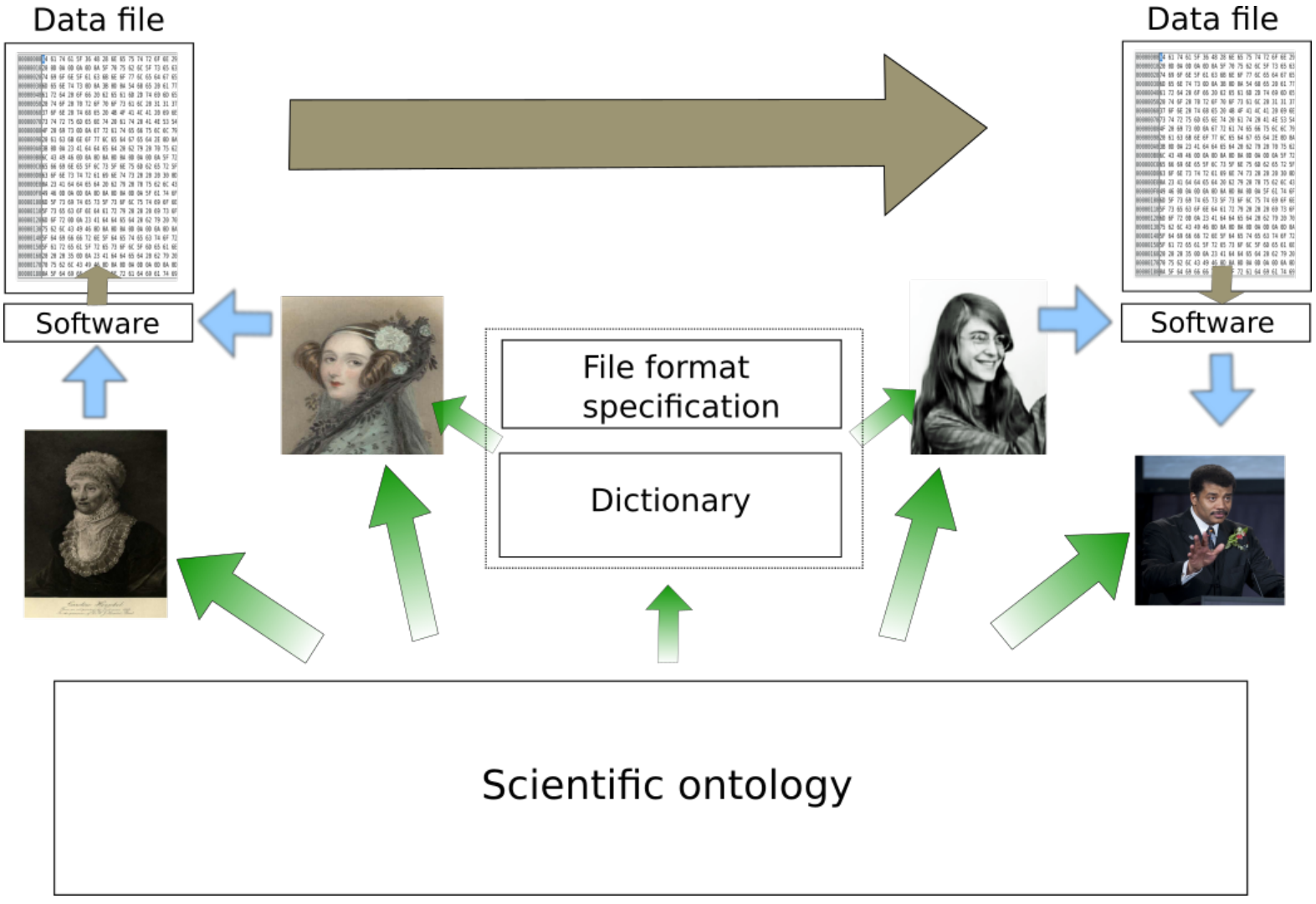
I have the honour to be, &c.

CAROLINA HERSCHEL.

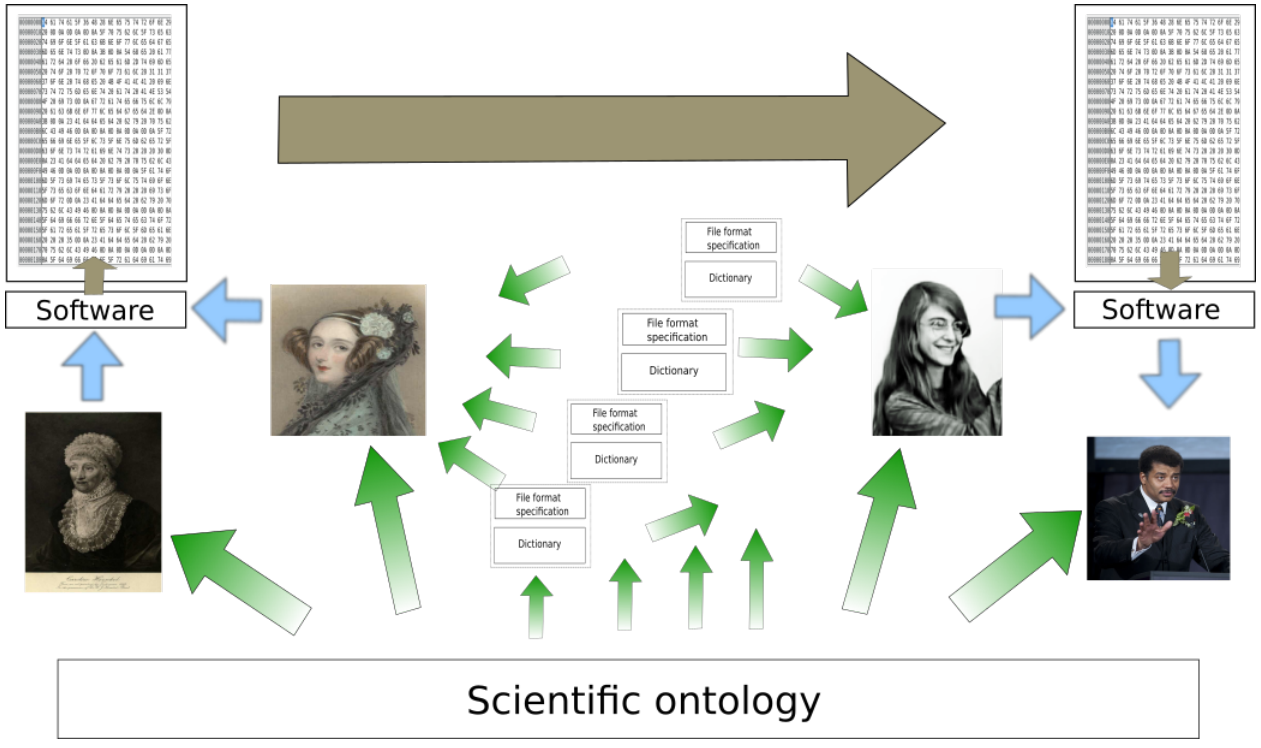


Shared understanding: grammar, terms, symbols

Computer-aided knowledge transfer



Multiple formats



Format explosion!

- NeXus - CIF; 192 image formats; ESRF beamline-specific arrangements
- canSAS, BioSAS,

Ontology development is hard work, don't want to duplicate effort. What to do?

1. Decouple from formats
2. What is necessary and sufficient for a metadata definition?

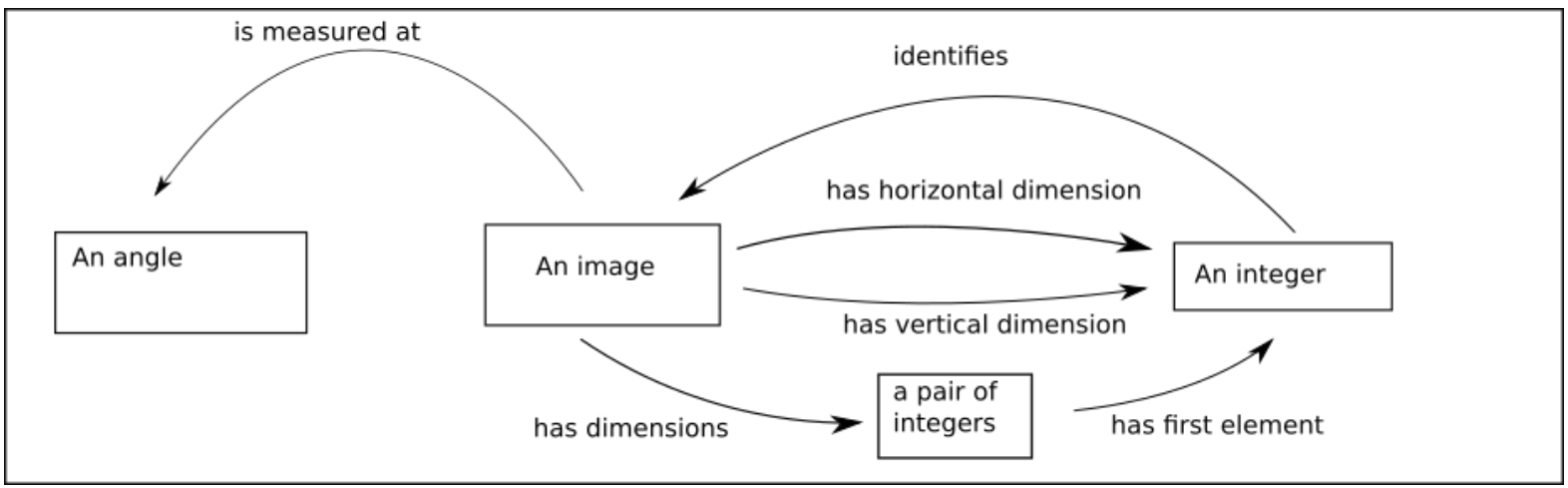
Decoupling from formats

Facts about the world cannot depend on the medium used to communicate them.

Universal characteristics of data: ontologies

"Olog" (ontology log) of Spivak and Kent (PLoS ONE, 2012, **7**(1), e24274)

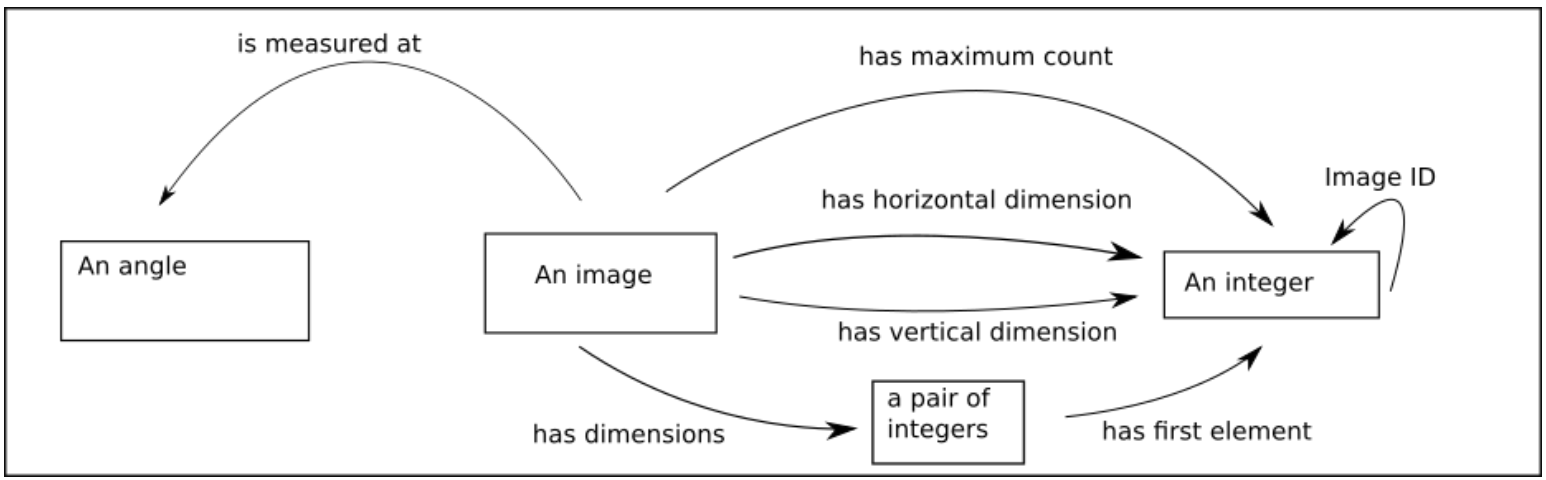
- Precise formulation of a conceptual worldview
- Isomorphic to relational database schema
- Is "really" a diagram of a mathematical category
  - Category: objects of a "class" (e.g. Sets, Groups) related by composable "morphisms" (often functions) with path equivalences
  - Widely applicable to many mathematical areas (as we would hope). "Mathematics of mathematics"
- Advantages:allow information fusion; modular



Ologs

Our entire ontology can be expressed as a set of concepts mapping (in the mathematical sense) domains onto ranges.

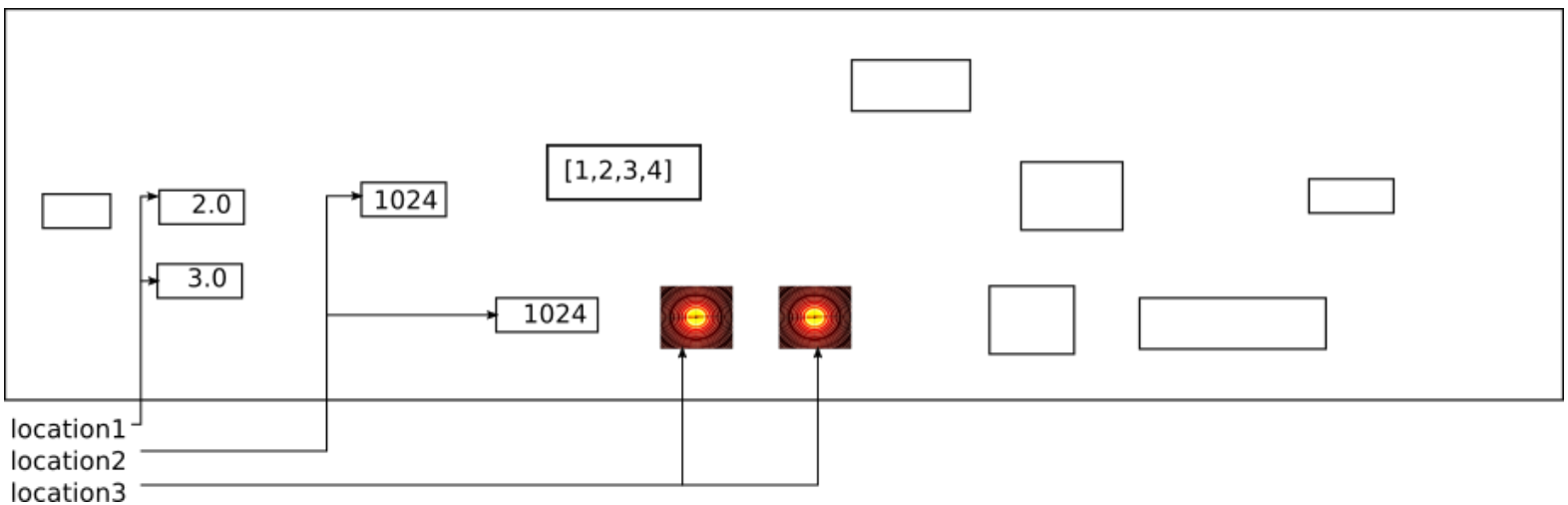
- Use the identity function for arbitrary values (e.g. image number)
- Use tuples if the value depends on multiple other values (e.g. (rotation angle, sample number))



Universal characteristics II: data files

Given an olog-type description of our domain, each data file is simply documenting instances of this ontology.

- The structure of the data file must therefore:
  - Allow association of values with concepts: the "location" of the concept
  - Allow multiple values to be associated with a single concept
  - Allow corresponding values to be identified
- The data file does not have to provide anything else, in particular, restating the relationship between concepts



Finding universal characteristics: CIF

- Semantically necessary features:
  - Concept ⇒ Dataname
  - Value ⇒ Value or entire loop column for that dataname
  - Correspondence ⇒ Same column index
- Semantically unnecessary features:
  - Collection of datanames together into loops

```
_computing_cell_refinement      'X-AREA (Stoe & Cie, 2005)'
_computing_data_reduction      'X-RED (Stoe & Cie, 2005)'
_computing_structure_solution      'SHELXS97 (Sheldrick, 2008)'
_computing_structure_refinement      'SHELXL97 (Sheldrick, 2008)'
_computing_molecular_graphics      'X-STEP32 (Stoe & Cie, 2000)'
_computing_publication_material      'SHELXL97 (Sheldrick, 2008)'
loop_
    _atom_site_type_symbol
    _atom_site_label
    _atom_site_fract_x
    _atom_site_fract_y
    _atom_site_fract_z
    _atom_site_U_iso_or_equiv
    _atom_site_adp_type
    _atom_site_calc_flag
    _atom_site_refinement_flags
    _atom_site_occupancy
    _atom_site_disorder_assembly
    _atom_site_disorder_group
   O O1 1.10305(16) 0.14200(13) 0.09457(7) 0.0364(3) Uani d . 1 . .
   O O2 1.03152(16) -0.04894(11) 0.07258(7) 0.0313(3) Uani d . 1 . .
   O O3 -0.36429(14) 0.34551(10) 0.47252(5) 0.0216(2) Uani d . 1 . .
   O O4 -0.21122(16) 0.32212(12) 0.56227(6) 0.0294(3) Uani d . 1 . .
   N N1 1.00462(17) 0.05502(13) 0.09634(7) 0.0222(3) Uani d . 1 . .
   N N2 0.40683(16) 0.13800(12) 0.22729(6) 0.0205(3) Uani d . 1 . .
   N N3 0.30983(16) 0.04473(12) 0.22891(6) 0.0192(3) Uani d . 1 . .
   N N4 -0.29651(17) 0.09198(12) 0.35372(6) 0.0190(3) Uani d . 1 . .
   C C1 0.84626(18) 0.07543(14) 0.12848(7) 0.0189(3) Uani d . 1 . .
   C C2 0.8210(2) 0.18816(14) 0.16069(8) 0.0207(3) Uani d . 1 . .
   H H2 0.9010 0.2503 0.1607 0.025 Uiso calc R 1 . .
   C C3 0.6730(2) 0.20566(14) 0.19291(8) 0.0207(3) Uani d . 1 . .
   H H3 0.6533 0.2803 0.2148 0.025 Uiso calc R 1 . .
   C C4 0.55405(18) 0.11203(14) 0.19253(7) 0.0188(3) Uani d . 1 . .
   C C5 0.5827(2) -0.00013(15) 0.15898(8) 0.0236(3) Uani d . 1 . .
   H H5 0.5026 -0.0622 0.1585 0.028 Uiso calc R 1 . .
```

Finding universal characteristics: NeXus

- Semantically necessary features:
  - Concept ⇒ NeXus class; NeXus class + class attribute; NeXus class + leaf dataset
  - Value ⇒ All values taken by the object referred to by the Concept, or if a class, the HDF5 locations of the classes (or any arbitrary unique identifier)
  - Correspondence ⇒ Same array index; from same node in the hierarchy?
- Semantically unnecessary features:
  - Parent-child structure (insofar as it does not contribute to the above)
- Date (1991) *"An Introduction to Database Systems"* p 374: a hierarchy is a relational database with additional complexity

```
/entry1:NXentry
 /instrument:NXinstrument
    name SuperHyperX
    /monochromator:NXmonochromator
      wavelength 1.5
      wavelength#units = angstrom
      /transx:  NXtransformations 0.1
        #transformation_type 'translation'
        #vector (1,0,0)
        #offset (0,0,0)
        #offset_units mm
      /transy:  NXtransformations -0.2
        #transformation_type 'translation'
        #vector (0,1,0)
        #offset (0,0,0)
        #offset_units mm
      /crystal:  NXcrystal
        usage 'Bragg'
        type 'PG'
        /transx:  NXtransformations 0.0
    ...
    /analyser:NXcrystal
      usage 'Bragg'
      type 'Ge'
      /transx:  NXtransformations 0.0
        #transformation_type 'translation'
        #vector (1,0,0)
        #offset (0,0,0)
        #offset_units mm
        #depends_on
'/entry1/instrument/detector/rotz'
```

What does B "depends on" A mean?

Formally, we have a mathematical mapping f from domain A to range B:$\{\forall x \in A, \exists y \in B : f(x) = y\}$

But e.g. F_{calc} depends on (h, k, l), space group, atomic positions, displacement parameters, wavelength...do we need to list them all?

- Ontologically: They exist regardless of whether we choose to list them. And don't forget the unknown unknowns.
- For a particular format: No, only those that vary within the "data unit".

What is a data unit?

Both NeXus (NXEntry) and CIF (data blocks) can encapsulate multiple units of data in a single file.

- A "data unit" is defined by the set of non-ID datanames that take multiple values within that unit (or by the set of datanames that are constant within that unit, but we don't necessarily know all members of this set).
  - A single sample rotated about one axis: 'rotation angle'.
  - Multiple samples rotated about several axes each: '(sample number,axis number,rotation angle)'

The overall framework

Any data transfer framework can be logically separated into a catalogue, format adaptor(s) and format(s):

A catalogue: We attach canonical names to concepts in the scientific ontology, describe the range ("type") mathematically (e.g. $\mathbb{R}^3$) and identify all other catalogue name(s) it depends upon.

- Units: dimensions only are included as part of the range (e.g. 'length').
- Values from arbitrary finite sets (e.g. refinement flags) are listed together with their meanings
- Easily merged with similar catalogues

A format adaptor: A subset of the catalogue is mapped to a particular format by specifying:

- How to locate values corresponding to a given canonical name (e.g. dataname correspondence table, assumed values) and handling of units
- Which dependencies of each canonical name are in scope for this format
- How values are represented for each canonical "type" (e.g. IEEE-754, text, binary integer, enumerated value correspondence).

Simple example: single-scan XAFS

Catalogue

...

Absorption.transmission: The absorption measured in transmission mode. Depends on: *Monitor.I0*, *Detector.Counts*, *Sample.pos*, *Sample.composition*. Positive real unitless number.

...

Format adaptor document for XDI:

Canonical	XDI	Notes
Absorption.transmission	mutrans	Column label
Scan.step	-	Use row number
etc.		

1. "Energy is the only multiply-valued dataname"
2. "All numeric values provided as text, interpretable as a floating point number in C, all other values are ASCII strings".

```
# XDI/1.0 GSE/1.0
# Column.1: energy eV
# Column.2: mutrans
# Column.3: i0
# Element.edge: K
# Element.symbol: Co
# Scan.edge_energy: 7709.0
# Mono.name: Si 111
# Mono.d_spacing: 3.13555
# Beamline.name: 13-ID-C
# Beamline.collimation: none
# Beamline.harmonic_rejection: detuned
# Facility.name: APS
# Facility.energy: 7.00 GeV
# Facility.xray_source: APS undulator A
# Scan.start_time: 2001-06-26T21:21:20
# Detector.I0: 10cm  N2
# Detector.I1: 10cm  N2
# Sample.name: Co metal foil
# Sample.prep: standard foil (Joe Wong boxed set)
# ///
# room temperature
# measured at beamline 13-ID-C
# vert slits = 0.3 x 0.3mm (at ~50m)
#----
#   energy      mutrans          i0
  7509.0000      -0.51329170        165872.70
  7519.0000      -0.78493490        161255.70
  7529.0000      -0.79281251        132048.70
  7539.0000      -0.79845258        121655.70
  7549.0000      -0.80458247        117735.70
  7559.0000      -0.81016444        116106.70
  7569.0000      -0.81581403        116202.70
  7579.0000      -0.82160876        116268.70
  7589.0000      -0.82726997        115529.70
  7599.0000      -0.83255491        115659.70
  7609.0000      -0.83804077        116147.70
```

XDI Specification: https://github.com/XraySpectroscopy/XAS-Data-Interchange

Single-scan XAFS: CIF alternative

Format adaptor document for CIF:

1. A table matching canonical names to (currently non-existent) CIF datanames, with units indicated as necessary
2. "Energy is the only independent multiply-valued dataname"
3. "All numeric values are as provided in the CIF syntax specification, all other values are UTF-8 encoded strings"

Benefits

- Ontology development is decoupled from file format:
  - New concepts and dependencies can be added by specialist non-programmers without compromising existing applications
  - A single ontology is available for programmers
  - Data framework designers can choose the file format most suitable for their application
  - Format-specific items do not pollute the ontology (e.g. lossless image compression method)
- Different approaches can use the same ontology
  - E.g. Peak intensity as a function of (h,k,l) or peak intensity of a given (h,k,l) as a function of wavelength
- By expressing multiple pre-existing ontologies in this framework:
  - Existing ontologies can be merged or used simultaneously
  - Data files can be algorithmically translated between ontologies
  - Ontologies can be harmonised

Format adaptor simplifications

Formats are completely free to find the efficient (e.g. retrieval time) representation of a value. For example, in mapping terms an image is $\{(x, y) \in \mathbb{Z}^2 \to \mathbb{R}\}$. Under suitable assumptions we can store the image as a contiguous or compressed block in our file.

Why not use a relational database (e.g. SQLite) as the file format?

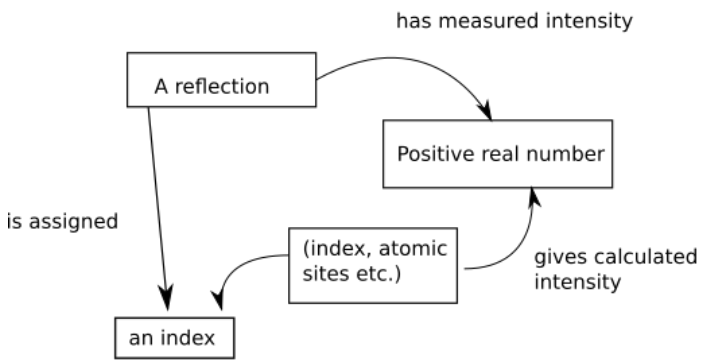
- Cons
  - many relational databases don't have good support for storing images.
  - (appear?) inefficient if columns have many identical entries
  - we are only transferring data, not manipulating it
- Pros
  - All other layouts have builtin bias: "good for some applications, bad for others" (Date op. cit. p 783). "A relational database is the second-best solution for any problem" [citation needed]
  - As we move to big data, we will tend to store data non-locally, and retrieve the bits we need. The non-local storage will probably be a relational database.
  - Matches underlying semantic structure i.e. less work for programmer to manipulate data

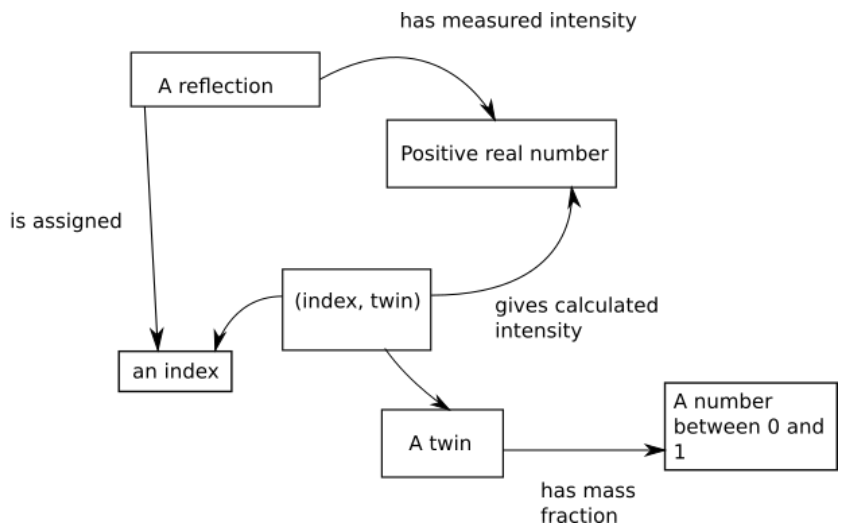
Dealing with format limitations

- Some file formats will be unable to easily or efficiently express the whole range of desirable value types
  - Matrices
  - Arbitrary length strings
  - Images
  - Inhomogeneous structures
- Not the ontology's concern (choose a better file format):
  - Inhomogenous types unlikely to be ontologically primitive, so the combined concept can be derived from the individual definitions and inhomogeneous types left out of the format
    - Example: (Atom type, atomic weight, f', f") compared to (h,k,l)
  - The format designer is free to represent these types in whatever way makes sense for them

Handling expansion of the ontology

- New items in the ontology do not compromise existing formats as the ontology is modular.
- A new dependency does not compromise existing formats as the dependencies are restricted by the format adaptor.
- A datafile generated against a new expanded ontology can be used in older software by splitting into data units matched to the simpler ontology.





Applications I: Universal file input API

A uniform API to all possible data formats should provide an analogue of the following function:

get_values(data_unit_handle,location,type): return all values associated with `location` in consistent order

For convenience:

get_value(data_unit_handle,key_name,key_value,location,type): return the `value` of `location` that corresponds to the value `key_value` of `key_name`

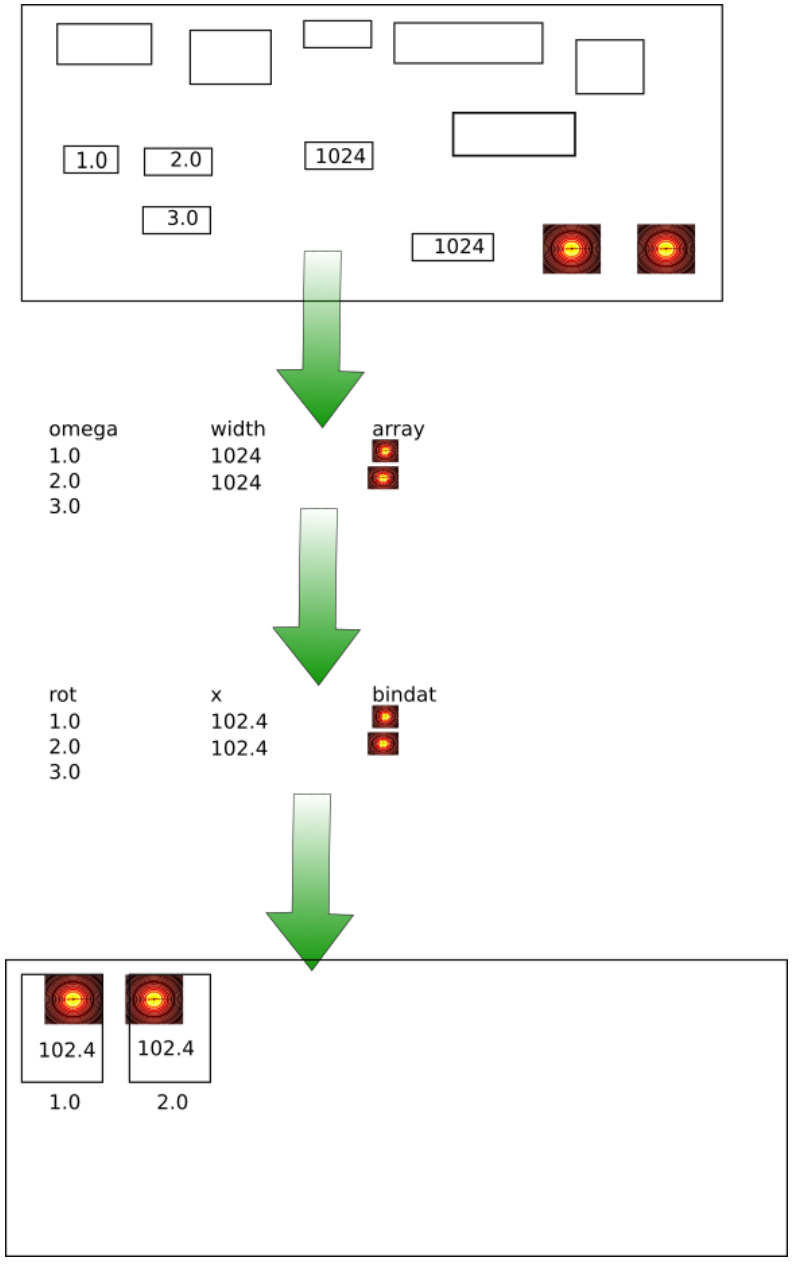
iterate_location(data_unit_handle,location,type): return an iterator over the values at `location`.

The API must define the concrete representation of each type, and the units. The programmer using the API knows the expected mathematical type so untyped functions provide no additional value.

Suggestion

Format adaptors to provide pseudocode for the `get_values` function

Applications II: File format translation



After defining the appropriate internal representation(s) of numeric values and common units:

1. Extraction of the (dataname,value) pairs (see previous slide)
2. Determination of target file datanames, algorithmic transformation of values
3. Application of semantically-irrelevant, format-specific structuring rules:
   - Clear separation of "scientifically meaningful" from "useful for other reasons"
   - For example CIF: provide correct dataname groupings
   - For example NeXus: lay out hierarchy for efficient use

Application III: Universal concordance

A list of canonical datanames from various ontologies, with equivalents.

Checking dataname equivalence:

- the mapping is identical
- invariant paths ("facts") are preserved

Minimum: list of names and equivalents in various namespaces:

```
...
nexus-nxmx:NXSample/NXBeam/incident_wavelength cif:_diffrn_radiation_wavelength.wavelength
...
```

see CBF-NXmx work
(https://sites.google.com/site/nexuscbf/mapping-draft/functional-mapping)

Full: add algorithmic derivations, text definitions and dependencies

Application IV: Metadata catalogue creation recipe

1. Brainstorm or collect concepts that you need
2. Determine what other concepts each multiple-valued concept depends upon
3. Assign canonical names
4. Describe the range of values for each concept
5. (Optional) What are possible "data units"?
    - Which concepts will take multiple values?
    - Or, what is held constant?
    - Exclude "universal" concepts: the same values for all data files
        - For example, atomic number
        - These concepts belong in the ontology, but are pointless in a data file

Proposals

- That the IUCr promote a machine-readable metadata "concordance"
  - Can start extracting metadata from other projects now
  - Create a semantic web
- That a metadata item is defined to consist of:
  - a canonical name
  - value type, range and dimension, as appropriate
  - common dependencies
  - a description sufficient to determine how values are unambiguously derived from the dependencies
- That format-specific elements are kept logically separate from metadata catalogues (e.g. number representation, units)
- That format-specific work include pseudocode for the 'get_values' function in terms of available file format APIs.

Acknowledgements

- The work of Graeme Winter, Tobias Richter, Jonathan Sloan, Herbert J. Bernstein on the CBF - NeXus concordance
- The publications of Spivak [Information and Computation 217 (2012) 31-51], Spivak and Kent [see above]
- Literature on the relational model for databases
- The work of the PDB on linking CIF to the relational model
- The extensive work of Hall, Spadaccini et. al. on developing dREL and DDLm
- Discussions on the XAFS format development list
- ANSTO

Comparison to CIF DDLm

DDLm (a language for describing an ontology in machine-readable form) has about 64 attributes, of which:

- 19 describe the range (e.g `enumeration`, `range`, `type`)
- 9 describe the concept (`description`, `example`, `method`)
- 3 or 4 describe the dependencies (`category`, `method`)
- 10 describe the canonical name in different regimes (`definition.id`, `alias`, `xref`)
- 23 are dictionary management (`scope`, `class`, `import`)