Data modelling, data access, and code

Norman Gray

Starlink/University of Glasgow

UNIVERSITY
of
GLASGOW

Overview

- HDX as a data model
- HDX as a data access layer
- Problems HDX does not attempt to solve
- Examples
- Summary

`http://www.starlink.ac.uk/HDX`

UNIVERSITY
of
GLASGOW

Summary

HDX is two things:

- a flexible, extensible, *data model* for astronomical images, tables and other metadata; and
- a *data access layer*, consisting of Java packages which handle the XML and URIs involved, and cast a variety of local or network resources into the HDX data model.
- . . . but it does not attempt to solve *all* problems.

UNIVERSITY
of
GLASGOW

HDX as a data model

Contents

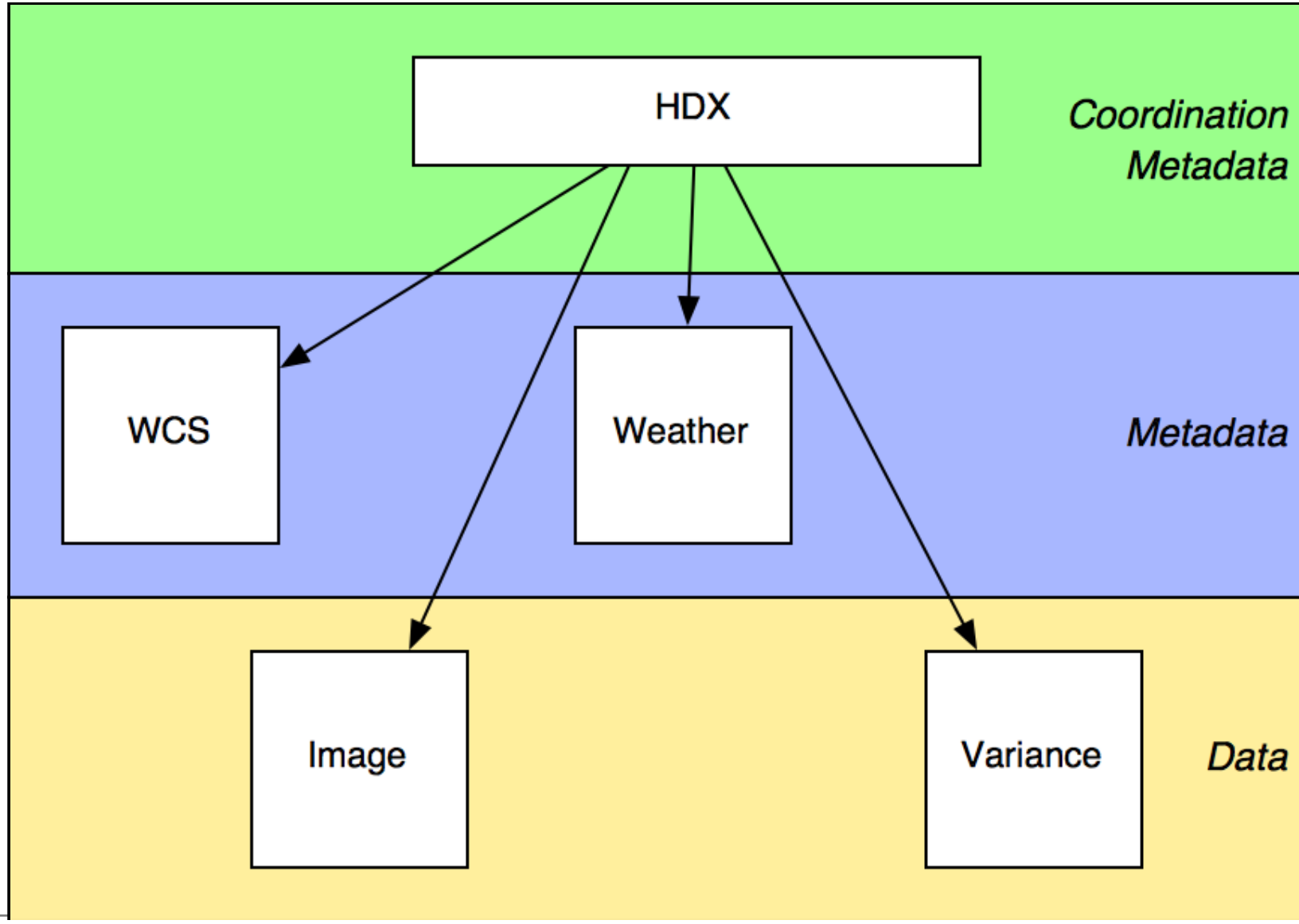
UNIVERSITY
of
GLASGOW

HDX is. . .

- . . . metadata
- . . . not primarily a data transport format
- . . . stand-off metadata
- . . . not exclusive
- . . . simple
- . . . extensible/evolvable
- . . . not just XML

UNIVERSITY
of
GLASGOW

1000 words



STARLINK

UNIVERSITY
of
GLASGOW

. . . plus a few more

- ‘metadata’ refers to single bitbuckets (in principle)
- . . . but avoids talking about interrelationships
- Makes things simpler if we can keep different types of metadata separate/independent.

UNIVERSITY
of
GLASGOW

XML is useful

The close relationship with XML allows us

- to specify very general and standardised APIs for objects which fit into this data model (the W3C DOM, Java Source, SAX, . . .);
- to take advantage of the wide variety of secondary XML standards such as XPath (and XPointer) for specifying components within large aggregations of data, XQuery for specifying searches, and XSLT for specifying transformation on data aggregates;
- to place URIs at the centre of our data structuring strategy.
- HDX/XML is format-, language- and wavelength-neutral.

UNIVERSITY *of* GLASGOW

NDX

- The most developed substructure presently defined within HDX
- describes images
- precursor is NDF, which Starlink has used heavily for over a decade
- simpler than one might expect
- . . . simpler than NDF's prototype versions
- . . . and simpler than some current proposals for metadata structures
- vindicates the drive for simplicity

UNIVERSITY
of
GLASGOW

NDX contains. . .

```
<hdx>
  <ndx>
    <image uri="file:/tmp/mydata.fits"/>
    <variance uri="file:/tmp/mydata-var.fits"/>
    <quality uri="http://telescope.edu/instrument/bad-pixels.fits"/>
  </ndx>
  <wcs tbd="true"/>
  <etc>[...]</etc>
</hdx>
```

- an `image`, `variance`, `quality`,
- plus other structures not yet fully specified in XML terms.

There is Java library support for sophisticated operations on the data thus expressed.

UNIVERSITY *of* GLASGOW

Tables

- The table module within HDX is under active development
- . . . but it is likely to remain rather simple.
- This work will integrate with VOTables as much as possible.
- Java support presents VOTables in this simpler model.

UNIVERSITY
of
GLASGOW

XML validity

- There is *no* DTD or Schema definition of HDX
- (if there were, it would be

```
<!ELEMENT hdx ANY>
```

  . . . so there doesn't seem a great deal of point).
- When code registers new types with the HDX system (such as NDX or tables), it also registers validators.
- A data set could be transported, processed, serialised, without ever seeing an angle-bracket once.

UNIVERSITY
of
GLASGOW

(RDF)

- The ‘Semantic Web’ is *not* just a fancy gizmo for creating Google++
- RDF is for specifying data models (termed ‘Ontology’ in this particular part of the forest).
- All about saying “X `IsA` Y, and the precise meaning of Y is. . . ”.
- The point of that, in turn, is to allow various types of high-level processing of resources, of which document metadata, and thence intelligent search engines using it, is the most obvious but probably not the most interesting example.

UNIVERSITY
of
GLASGOW

HDX as a data access layer

Contents

- HDX as a data model
- HDX as a data access layer
  - HDX is where the infrastructure is
  - uk.ac.starlink.hdx
  - uk.ac.starlink.ndx and .array
- Problems HDX does not attempt to solve
- Examples
- Summary

UNIVERSITY
of
GLASGOW

HDX is where the infrastructure is

Although the notional HDX DTD, above, is trivial, the Java packages which support it can naturally provide a good deal of infrastructure.

- Data is structured using XML concepts,
- can be manipulated as XML when this is appropriate,
- but for processing, this would be horrendously inefficient,
- and so the data access classes present alternative Java interfaces to the objects they manage, designed for computational efficiency.

UNIVERSITY
of
GLASGOW

uk.ac.starlink.hdx

The HDX package at present provides:

- handling of FITS, XML and NDF for input and output, as part of a pluggable structure for input handlers;
- registration and optional validation of new types;
- namespace processing for input XML, allowing HDX structure to be overlaid on ‘foreign’ XML if key attributes are included;
- URI and `xml:base` resolution.

It has hooks, but not yet code, for:

- general URI or URN to URL resolution (registry operations);
- resource discovery (replica management).

UNIVERSITY *of* GLASGOW

uk.ac.starlink.ndx and .array

- Classes for manipulating N-dimensional arrays in an efficient and general fashion.
- Consistent interface copes with large arrays, pixel errors, simple or sophisticated pixel quality marking, world coordinate systems, varying pixel ordering schemes, processing history and extensible metadata storage.
- At the applications level, code does not need to be aware of the format (FITS, HDS, . . .) in which the data is stored.

UNIVERSITY
of
GLASGOW

Problems HDX does not attempt to solve

Contents

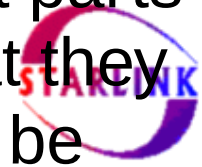
- HDX as a data model
- HDX as a data access layer
- Problems HDX does not attempt to solve
  - Problems HDX does not attempt to solve
  - Problems HDX does not attempt to solve
- Examples
- Summary

UNIVERSITY
of
GLASGOW

Problems HDX does not attempt to solve

HDX does not attempt to address problems where an adequate solution already exists

- HDX does not attempt to be a data transport or data description system, since formats like FITS manage that perfectly well. Where FITS struggles is in trying to describe interrelationships between data, or in attempting to describe highly structured metadata
- One-size-fits-all will never be finalised. If different parts of a spec address different problems, the fact that they are artificially yoked together means that it could be dificult to evolve those parts at different rates.



UNIVERSITY
of
GLASGOW

Instead…

HDX does not attempt to address problems where an adequate solution already exists

- Instead, a componentised approach, with various metadata standards developed separately, and tied together in any particular instance by the larger-scale coordination of HDX.
- HDX is able to be simple, because it deals, cleanly and straightforwardly, with a particular layer of the architectural problem.

UNIVERSITY
of
GLASGOW

Examples

Contents

- HDX as a data model
- HDX as a data access layer
- Problems HDX does not attempt to solve
- Examples
  - Basic HDX object
  - Namespacing
- Summary

UNIVERSITY
of
GLASGOW

Basic HDX object

```
<hdx>
  <ndx>
    <image uri="file:/tmp/mydata.fits"/>
    <variance uri="file:/tmp/mydata-var.fits"/>
    <quality uri="http://telescope.edu/instrument/bad-pixels.fits"/>
  </ndx>
  <wcs tbd="true"/>
  <etc>[...]</etc>
</hdx>
```

It is straightforward (and obviously preferable), but not yet standardised, to refer to individual FITS extensions in a URI, and thus to keep the image and variance together in a single FITS file.

Can embed HDX within FITS.

UNIVERSITY *of* GLASGOW

Namespacing

XML Namespacing can be used to include HDX information within ‘foreign’ XML:

```
<foreign xmlns:x="http://www.starlink.ac.uk/HDX">
  <x:hdx>
    <x:ndx>
      <x:image uri="file:/tmp/mydata.fits"/>
    </x:ndx>
  </x:hdx>
</foreign>
```

or even:

```
<mystructure x:hdxname="ndx" xmlns:x="http://www.starlink.ac.uk/HDX">
  <mypointer x:hdxname="image">
    mydata.fits
  </mypointer>
  <comment x:hdxname="title">My data</comment>
</mystructure>
```

Any changes made to this virtual XML structure may optionally be automatically shadowed in the original XML!

UNIVERSITY of GLASGOW

Summary

HDX is two things:

- a flexible, extensible, *data model* for astronomical images, tables and other metadata; and
- a *data access layer*, consisting of Java packages which handle the XML and URIs involved, and cast a variety of local or network resources into the HDX data model.
- . . . but it does not attempt to solve *all* problems.

UNIVERSITY
of
GLASGOW