Semantics and use of `responseDate`

Intended use

- responseDate is included with every OAI-PMH response:

  *"a UTC datetime indicating the time that the response was sent. This **must** be expressed in UTC"*

- Re-exported in provenance container as the harvestDate:

  "the responseDate of the OAI-PMH response that resulted in the record being harvested from the original repository"

- Also used for sequential harvesting. Get next from date from the responseDate

Problem 1

- responseDate values such as

  2002-11-26T19:18:50+40:00

  schema validate because the OAI-PMH response date doesn't sufficiently restrict the acceptable values on date fields.

- must not have TZ +40:00 the text of the spec permits only Z (Zulu format)

Solution for problem1

- Tighten the schema.
- Introduce dateTimeZuluType, e.g.:

```
<!--
Restrict dateTime to allow only "Zulu" format specification
of dateTime in UTC. This requires simply that we demand that
the last character is a 'Z' since that is the only valid use
of Z in dateTime.-->
 <simpleType name="dateTimeZuluType">
    <restriction base="dateTime">
      <pattern value=".+Z"/>
    </restriction>
 </simpleType>
```
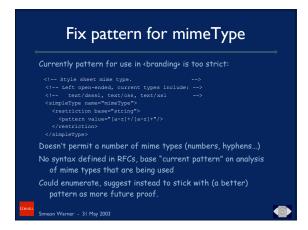
- Okay, that was the easy bit...

Problem 2

- What about bad (but properly expressed) responseDate values?
- How do we test for bad?
- How would downstream sites use harvestDate elements in the provenance containers?
  - If consistent, even bad values would be useful for dedupping and in choosing the most up-to-date record
  - Bad values might not make sense when compared with other datestamps

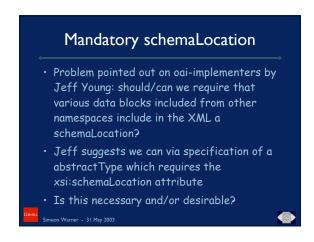## Solution of problem 2

- Test: compare with local clock (known to be good?). How much skew to allow?
- If bad:
  - Abandon harvest?
  - Use bad datestamp in provenance? Use bad datestamp in next from date?
  - Substitute local value for time of harvest in provenance records? Use for from data in next harvest?
  - Use earlier of responseDate and local datestamp as from address?

Tightening schema and specification

A laundry list of fixes…

Simeon Warner - 31 May 2003

Empty sets

- Specification does not explicitly say whether sets may be empty
- I assume that empty sets are allowed (doesn't break anything)
- Should this be made explicit?

(pointed out by Naomi)

Simeon Warner - 31 May 2003

Fix pattern for mimeType

Currently pattern for use in <branding> is too strict:

```
<!-- Style sheet mime type.               -->
<!-- Left open-ended, current types include: -->
<!--   text/dsssl, text/css, text/xsl        -->
<simpleType name="mimeType">
  <restriction base="string">
    <pattern value="[a-z]+/[a-z]+"/>
  </restriction>
</simpleType>
```

Doesn't permit a number of mime types (numbers, hyphens…)

No syntax defined in RFCs, base "current pattern" on analysis of mime types that are being used

Could enumerate, suggest instead to stick with (a better) pattern as more future proof.

Simeon Warner - 31 May 2003

Mandatory schemaLocation

- Problem pointed out on oai-implementers by Jeff Young: should/can we require that various data blocks included from other namespaces include in the XML a schemaLocation?
- Jeff suggests we can via specification of a abstractType which requires the xsi:schemaLocation attribute
- Is this necessary and/or desirable?

Simeon Warner - 31 May 2003

Attribute in <request>

- Error in specification, section 4.5 has badArgument response which includes attributes in <request>. Must zap.
- BUT, specification currently avoids specifying condition (section 3.2):
  - Must have attributes if no error
  - Must not have attributes if badVerb or badArgument
  - What about other error/exception conditions?
- Clarify or specify?

Simeon Warner - 31 May 2003