Using the SRA Identifier Block

10 May 2012 Final

Overview

The purpose of the SRA Identifier block is to capture in one place all keys that are used as IDs. An ID can identify exactly one record within a context. A record may have multiple IDs. A record's ID must be unique within a context, and all objects in a context must have an ID. These properties do not hold for "names" or other monikers.

The SRA Identifier block contains the following identifiers:
- PRIMARY_ID – Primary key to an INSDC database.
- SECONDARY_ID – Secondary key or defunct primary key to an INSDC database
- EXTERNAL_ID – Identifier from another database qualified by a namespace.
- SUBMITTER_ID – Local identifier qualified by a namespace.
- UUID – Universally unique identifier which requires no namespace.

The identifiers block contains a set of identifiers. The identifiers may occur in any order or combination so long as exactly one PRIMARY_ID is present.

Goals

- Consolidate use of identifiers for each SRA document
- Distinguish between accessions and named IDs
- Add support for UUIDs
- Improve flexibility for submitter identification of records

Features

- Tracks archived assigned ids
- Tracks submitter assigned ids
- Supports tracking of object mergers and replacements via secondary ids and aliases
- Tracks alternate or external ids assigned by different databases or archives
- Support for 3rd party assigned ids including catalog ids
- Explicit support for UUIDs

Design

The IdentifiersType is defined in the SRA.common.xsd schema, please look in the following location(s):
- http://www.ncbi.nlm.nih.gov/viewvc/v1/trunk/sra/doc/SRA_1-4/SRA.common.xsd?view=co
- Here is the relevant type code from SRA.common.xsd:

```xml
<xsd:complexType name="IdentifierNodeType">
    <xsd:simpleContent>
        <xsd:extension base="xs:string">
            <xsd:attribute name="label" use="optional" type="xs:string"/>
        </xsd:extension>
    </xsd:simpleContent>
</xsd:complexType>

<xsd:complexType name="AccessionType">
    <xsd:simpleContent>
        <xsd:extension base="com:IdentifierNodeType"/>
    </xsd:simpleContent>
</xsd:complexType>

<xsd:complexType name="NameType">
    <xsd:simpleContent>
        <xsd:extension base="com:IdentifierNodeType">
            <xsd:attribute name="namespace" use="required" type="xs:string"/>
        </xsd:extension>
    </xsd:simpleContent>
</xsd:complexType>

<xsd:complexType name="UUIDType">
    <xsd:simpleContent>
        <xsd:extension base="com:IdentifierNodeType"/>
    </xsd:simpleContent>
</xsd:complexType>

<xsd:complexType name="IdentifierType">
    <xsd:sequence>
        <xsd:element name="PRIMARY_ID" type="com:AccessionType" minOccurs="0" maxOccurs="1"/>
        <xsd:element name="SECONDARY_ID" type="com:AccessionType" minOccurs="0" maxOccurs="unbounded"/>
        <xsd:element name="EXTERNAL_ID" type="com:NameType" minOccurs="0" maxOccurs="unbounded"/>
        <xsd:element name="SUBMITTER_ID" type="com:NameType" minOccurs="0" maxOccurs="unbounded"/>
        <xsd:element name="UUID" type="com:UUIDType" minOccurs="0" maxOccurs="1"/>
    </xsd:sequence>
</xsd:complexType>
```

Data Types

The IdentifierNodeType abstract type extends xs:string with the following attributes:

- **label** – whether and how to display a tag string.

Four concrete types subclass IdentiferType in order to suggest the business use of the identifier:

AccessionType – A key in an INSDC primary database.

NameType –A key in an external database.

The following attributes are required:

- **namespace** – Namespace (database) of the external name

UUIDType – A key that is universally unique and requires no namespace.

Data Structure

PRIMARY_ID – A primary identifier, or key, in the INSDC primary database (accession). Example: SRR330090. Exactly one primary identifier is required in every IDENTIFIER block. This value is equivalent to the document/@accession attribute.

SECONDARY_ID – A foreign key in the INSDC primary database (accession), or a defunct primary key in the INSDC primary database. Example: SRR330091. Any number of secondary identifiers may be present.

EXTERNAL_ID – A key in an external database qualified by the name of the database. Example: Coriell NA12878. Any number of external names may be present.

SUBMITTER_ID – A key that resolves within the submitter's namespace.   Exactly one local name must be present on submission. Local names are not needed for data download or exchange between archives. This value is equivalent to the (document/@alias, document/@center_name) attribute tuple.

UUID – A key that is universally unique and needs no namespace. UUIDs are not used by the Archive but rather are provided as part of the SRA xml schema to serve downstream applications, including non-INSDC SRA mirrors.

Compatibility

The existing NameGroup and RefNameGroup attribute groups currently used to identify records will continue in use.

Semantics

The IdentifierType is implemented by each SRA archive with additional business rules governing use of namespaces and scope of identifiers.

Replacement tracking

The IdentifierType can be used to name record(s) replaced (taken over) by the current record.   The transitive closure of these replacing relations is a set of currently active records with replaced descendants. The converse relation (replaced by) can be computed from this forest so it is not tracked explicitly.

Persistence

One goal of the IDENTIFIERS block is to document data migration, replacement, and equivalency relationships independently of the life cycle of the record, so that Archive users who form dependencies on a certain SRA record can always recover the relationship to other records even if the record has been suppressed.

Use Cases

Data Migration

The IDENTIFIERS block can be used to manage the transition of metadata from one record to another and provide a trackback mechanism to recover previous incarnations.   This would include:

- Tracking a record in the archive (or prior to archiving) with a submitter supplied identifier.
- Tracking a record's identifier before and after a data migration.
- Tracking a record's identifier before and after a data consolidation.
- Tracking a changes in an identifier used for a dependency

Data Replacement

The IDENTIFIERS block can be used to indicate that the content has been replaced, and identify the previous record that represented the content.   A run may have been mis-loaded due to errors in the original load process or a misrepresentation of the metadata that caused the data to be interpreted differently. If the result of the mis-load is an SRA archive image that is substantially different then the run's accession will be replaced. Another example is where duplicate runs have been discovered, and each run can be mapped to its duplicates although only one of them is retained in the archive.

Data Equivalency

The IDENTIFIERS block can be used to point to records that are equivalent and can be used interchangeably. An example is the BioProject and SRA study identifiers, which for a time will both be active identifiers of a study record (until migration from SRA study to BioProject is completed). Another example is where equivalent records have been discovered in multiple SRA instances. This would happen when a submitter has sent the same submission to both NCBI and EBI, for example. Over time, the INSDC may elect to retain one instance and suppress the other one, but the ID block can be used to maintain the equivalence relation.

Examples

SRA document identifiers

The document can contain IDENTIFIERS block in co-existence with existing NameGroup attribute group :

```
<RUN xmlnamespace="" run_center="BI" run_date="2011-08-04T04:00:00Z" instrument_name="SL-HAC">
    <IDENTIFIERS>
      <PRIMARY_ID>SRR354028</PRIMARY_ID>
      <SUBMITTER_ID  namespace="BI" >BI.PE.110804_SL-HAC_0370_BFCB02H8ACXX.6.UNMATCHED.srf</SUBMITTER_ID>
    </IDENTIFIERS>
```
**IDENTIFIER values must agree with NameGroup values

The document can contain IDENTIFIERS block in lieu of existing NameGroup attribute group:

```
<RUN>
    <IDENTIFIERS>
      <SUBMITTER_ID  namespace="BI" >BI.PE.110804_SL-HAC_0370_BFCB02H8ACXX.6.UNMATCHED.srf</SUBMITTER_ID>
      <PRIMARY_ID>SRR354028</PRIMARY_ID>
    </IDENTIFIERS>
```

This gives a migration path for adoption of Identifier block in place of the name group attributes group, or a method for reverse construction of the NameGroup attributes from the ID block.

SRA document references

Document dependency references to other documents can be encoded with or without the RefNameGroup attributes.

SRA Study / BioProject / dbGaP study Reference

```
SRA Study:
Refname group example (same as previous schema versions):
  Reference by local names:
    <STUDY_REF refcenter="BI" refname="Ceratotherium_simum_simum_WGS"/>

    Is equivalent to:
    </STUDY_REF>
          <SUBMITTER_ID namespace="BI" >Ceratotherium_simum_simum_WGS</SUBMITTER_ID>
    <STUDY_REF>
  Reference by accession
<STUDY_REF accession ="SRPxxxxxx"/>

   Is equivalent to:
    <STUDY_REF>
         <IDENTIFIERS>
           <PRIMARY_ID>SRPxxxxxx</PRIMARY_ID>
         </IDENTIFIERS>
    </STUDY_REF>
BioProject:
    <STUDY_REF refcenter="BioProject" refname="PRJNA74583"/>
  Is equivalent to:
    <STUDY_REF>
         <IDENTIFIERS>
           <EXTERNAL_ID namespace="BioProject">PRJNA74583</SUBMITTER_ID>
         </IDENTIFIERS>
    </STUDY_REF>

dbGaP:
    <STUDY_REF refcenter="dbgap" refname="phsxxxxxx"/>
  Is equivalent to:
    <STUDY_REF>
         <IDENTIFIERS>
           <EXTERNAL_ID namespace="dbgap">phsxxxxxx</SUBMITTER_ID>
         </IDENTIFIERS>
    </STUDY_REF>
```

Sample Reference

Sample reference using the identifiers block should contain only 1 identifier: either a primary_id, an external_ id or a submitter_id

```
experiment-to-sample and experiment-to-BioSample reference

Referencing SRA samples by accession
        <SAMPLE_DESCRIPTOR>
          <IDENTIFIERS>
            <PRIMARY_ID>SRSxxxxxx</PRIMARY_ID>
          </IDENTIFIERS>
        </SAMPLE_DESCRIPTOR>
Referencing SRA samples by submitter_id/alias
        <SAMPLE_DESCRIPTOR>
          <IDENTIFIERS>
            <SUBMITTER_ID namespace="JGI">10908</SUBMITTER_ID>
          </IDENTIFIERS>
        </SAMPLE_DESCRIPTOR>
Referencing BioSamples by accession
        <SAMPLE_DESCRIPTOR>
          <IDENTIFIERS>
            <EXTERNAL_ID namespace="BioSample">SAMNxxxxxx</EXTERNAL_ID>
          </IDENTIFIERS>
```

```
        </SAMPLE_DESCRIPTOR>
Referencing dbgap samples
        <SAMPLE_DESCRIPTOR>
          <IDENTIFIERS>
            <EXTERNAL_ID namespace="phsxxxxxx">submitted_sample_id</EXTERNAL_ID>
          </IDENTIFIERS>
        </SAMPLE_DESCRIPTOR>
```

Replaced Record

The information that a certain record has been replaced is not indicated in the IDENTIFIERS block, but is tracked in the SRA database and livelist.

```
<RUN run_date="2008-11-24T23:08:44Z" instrument_name="GA-5">
    <IDENTIFIERS>
        <PRIMARY_ID>SRR292241</PRIMARY_ID>
    </IDENTIFIERS>
```

Replacer Record

This example shows how a record, SRR390728, replaces a predecessor SRR292241:

```
<RUN run_date="2008-11-24T23:08:44Z" instrument_name="GA-5">
    <IDENTIFIERS>
        <PRIMARY_ID>SRR390728</PRIMARY_ID>
        <SECONDARY_ID>SRR292241</SECONDARY_ID>
    </IDENTIFIERS>
```

Elected Record

This example shows how one record, SRR351940, has replaced 9 others (elected as successor), as in the case where several 'readgroup' runs provisionally reference the same bam file, one is selected for cSRA loading and the remaining runs are suppressed.

```
<RUN>
    <IDENTIFIERS>
        <PRIMARY_ID>SRR351940</PRIMARY_ID>
        <SECONDARY_ID>SRR351941</SECONDARY_ID>
        <SECONDARY_ID>SRR351942</SECONDARY_ID>
        <SECONDARY_ID>SRR351943</SECONDARY_ID>
        <SECONDARY_ID>SRR351944</SECONDARY_ID>
        <SECONDARY_ID>SRR351945</SECONDARY_ID>
        <SECONDARY_ID>SRR351946</SECONDARY_ID>
        <SECONDARY_ID>SRR351947</SECONDARY_ID>
        <SECONDARY_ID>SRR351948</SECONDARY_ID>
        <SECONDARY_ID>SRR351949</SECONDARY_ID>
    </IDENTIFIERS>
```

Successor Record

This example shows how one record, SRR351940, has replaced another kind of record, analysis object SRZ019522.

```
<RUN>
    <IDENTIFIERS>
        <PRIMARY_ID>SRR351940</PRIMARY_ID>
        <SECONDARY_ID>SRZ019522</SECONDARY_ID>
    </IDENTIFIERS>
```

Submitter alternate identifiers

Submitted records can retain their alternate identifiers and these can be treated as identifiers rather than attributes of the record. The label attribute calls out the display field.

```
<RUN center_name="BI" alias="70291ABXX110301.7.tagged_393.bam" run_center="BI" run_date="2011-03-
01T05:00:00Z" instrument_name="SL-HBZ" accession="SRR404010">
  <IDENTIFIERS>
    <PRIMARY_ID>SRR404010</PRIMARY_ID>
    <SUBMITTER_ID namespace="BI">70291ABXX110301.7.tagged_393.bam</SUBMITTER_ID>
    <SUBMITTER_ID namespace="BI" label="read group platform unit"
>70291ABXX110301.7.CCAGTTAG</SUBMITTER_ID>
  </IDENTIFIERS>
…
```

Submitter replaced identifiers

Submitters can replace an identifier with a new one without disturbing the linkage to existing SRA accessions. However, the primary identifier must be supplied and the defunct identifier must be removed by an update submission.

existing…
```
<RUN alias="454_O.mykiss_GD3412001" accession="SRR090454" center_name="INRA">
<IDENTIFIERS>
    <PRIMARY_ID>SRR090454</PRIMARY_ID>
    <SUBMITTER_ID namespace="INRA">454_O.mykiss_GD3412001</SUBMITTER_ID>
  </IDENTIFIERS>
…
```

updated…
```
<RUN alias="454_O.mykiss_GD3412001" accession="SRR090454" center_name="INRA">
<IDENTIFIERS>
    <PRIMARY_ID>SRR090454</PRIMARY_ID>
    <SUBMITTER_ID namespace="INRA">454_O.mykiss_GD3412001</SUBMITTER_ID>
    <SUBMITTER_ID namespace="INRA">454_O.mykiss_GB5RBPX02</SUBMITTER_ID>
  </IDENTIFIERS>
…
```

Commonly used external identifiers

In lieu of a local identifier, a submitter can use a supported external identifier. A good example is a cell line DNA isolate sample from one of the Coriell NA12878:

```
<SAMPLE>
<IDENTIFIERS>
    <PRIMARY_ID > SRS000090</PRIMARY_ID>
    <EXTERNAL_ID namespace="Coriell" label="Catalog ID">NA12878</EXTERNAL_ID>
    <EXTERNAL_ID namespace="Coriell" label="Catalog ID">GM12878</EXTERNAL_ID>
  </IDENTIFIERS>
…
```

Universally unique identifiers

A downstream user of SRA xml data could annotate it with a universally unique identifier. This requires no namespace because it is universally unique (according to the generation method). The INSDC SRAs do not use UUIDs and these are ignored on submission.

```
<RUN alias="68b329da9893e34099c7d8ad5cb9c940" accession="SRR090454" center_name="">
<IDENTIFIERS>
    <PRIMARY_ID>SRR090454</PRIMARY_ID>
    <UUID> 68b329da9893e34099c7d8ad5cb9c940 </UUID>
  </IDENTIFIERS>
…
```

Submission Considerations

The PRIMARY_ID will appear in each record obtained from the SRA, but clearly it is unknown at the time of submission. Consequently, for submitted records the PRIMARY_ID should be omitted:

```
<EXPERIMENT>
  <IDENTIFIERS>
    <SUBMITTER_ID namespace="BI">658005.WR28289.HS_PF_Sen_42.C0DJUACXX120207.P</SUBMITTER_ID>
  </IDENTIFIERS>
```

For modify submissions, the submitter can either use the assigned PRIMARY_ID or reuse the original SUBMITTER_ID if it successfully identifies the record to modify.

NCBI will continue to support submissions without IDENTIFIER blocks. The existing NameGroup and RefNameGroup attribute groups will be processed and identifiers will be derived from these attributes.

Exchange Considerations

For mirror archives, NCBI will supply SRA records with the IDENTIFIER block and the NameGroup and RefNameGroup attributes. The attributes will be derived from the appropriate content in the IDENTIFIER block.

Mirror archives may choose to continue relying on the NameGroup and RefNameGroup attributes and not support the IDENTIFIER block. In this case the IDENTIFIER block will be written as follows:

PRIMARY_ID := @accession
SECONDARY_ID @namespace=@center_name := @alias

or for individual submitters that do not have a center_name,

SECONDARY_ID @namespace=null := @alias

The IDENTIFIER block can be dropped from the mirror.