SRA Application Notes

Last Updated: 2014 Jan 29

National Center for Biotechnology Information (US)
Bethesda (MD)

National Center for Biotechnology Information (US), Bethesda (MD)

NLM Citation: SRA Application Notes [Internet]. Bethesda (MD): National Center for Biotechnology Information (US); 2010-.

This documentation provides application notes for the Sequence Read Archive (SRA) at the National Center for Biotechnology Information.

Table of Contents

Active Notes

Short Read Archive (SRA) Requirements Meeting

Created: July 29, 2007; Updated: November 20, 2010.

Status	Historical
Active Date	2007
Inactive Date	
Scope	INSDC SRA

Date: July 27, 2007

Place NISC Conf Room Rockville MD USA

1. Attendees

1 Mike Attili	Helicos	mattili at helicosbio.com
2. Vladimir Alekseyev	NCBI	aleksey at ncbi.nlm.nih.gov
3. Inna Belaia	NCBI	belaia at mail.nih.gov
4. Toby Bloom	BI	tbloom at broad.mit.edu
5. Vivian Bonazzi	NHGRI	bonazziv at mail.nih.gov
6. James Bonfield	Sanger	jkb at sanger.ac.uk
7. Kevin Bradtke	Genecodes	kbradtke at genecodes.com
8. Deanna Church	NCBI	church at ncbi.nlm.nih.gov
9. Guy Cochrane	EBI	cochrane at ebi.ac.uk
10. Anthony Cox	Illumina	anthony.cox at solexa.co.uk
11. David Dooling	Wash U	ddooling at watson.wustl.edu
12. Adam Felsenfeld	NHGRI	felsenfa at exchange.nih.gov
13. Paul Flicek	EBI	flicek at ebi.ac.uk
14. Tim Hunkapiler	ABI	tim at discoverybio.com
15. Steve Leonard	Sanger	srl at sanger.ac.uk
16. Elaine Mardis	Wash U	emardis at wustl.edu
17. Garbor Marth	BC	marth at bc.edu
18. Jason Miller	JCVI	jmiller at jcvi.org
19. Donna Muzny	BCM	donnam at bcm.tmc.edu
20. Jeffrey Reid	BCM	jgreid at bcm.tmc.edu
21. Harris Shapiro	JGI	hjshapiro at lbl.gov
22. Martin Shumway	NCBI	shumwaym at ncbi.nlm.nih.gov

Table continues on next page...

Table continued from previous page.

23. Asim Siddiqui	BCGSC	asims at bcgsc.ca
24. Bill Spencer	Roche	bill.spencer at roche.com
25. Kristen Stoops	Helicos	kstoops at helicosbio.com
26. Kris Wetterstrand	NHGRI	wettersk at mail.nih.gov
27. Eugene Yaschenko	NCBI	yaschenk at mail.nih.gov
28. Mike Zody	BI	mczody at broad.mit.edu

2. Data Production and Archive Requirements (led by David Dooling)

One approach is to show only the base coverage and coverage depth at each base of the reference.

A table of data requirements for each of the vendors was presented.

2.1. What do the centers currently do about data retention?

WUGC – We store SFF and metrics from 454 runs. We store R and D directories for two months, then purge them. For Illumina, we store everthing, but hope to store only the resulting .prb files.

WIBR – We keep all raw data for 1-2 months. Then we keep only a sub-sampling of images for subsequent quality analysis. We do our own alignments so the secondary analysis is not retained. For SOLiD, we do not pull images off the machine as this would be impractical. Just pull off processed intensity files (about 600 GB for 2 slides).

BCM – We are keeping everything for now but would like to get to a 3 month retention period. We would like to keep post-image analysis files.

SC – We keep images for a few weeks. We keep SFF from 454. We would like to keep SRF from Illumina once that is ready, and alignments are kept in a local format.

2.2. What should be publicly archived?

- For 454, SFF plus meta data should be archived. The new SRF format should subsume SFF.
- Everybody agreed archiving image data is infeasible.
- Metrics from the instrument runs should not be archived. Even thought these are minimal in size, they are interesting only to production QC at the Centers and not to users of the archive.
- Everybody agreed it is important to accept experimental meta data in a form which can be readily archived (as opposed to now, where it is embedded in each trace record).

- Everybody agreed that Sanger data should remain in the current Trace Archive. Existing deposits of 454 data could be duplicated in the SRA, but this would entail additional migration work.
- Everybody expressed the desire that SRF be the transaction medium for run data. It was felt that once adopted, the vendors would augment or replace existing delivery formats with SRF.
- There was a question about logging four intensities (one per channel per cycle) rather than one (for the principal base call). Many Sanger-type sequencing analysis tools used only the intensity value of the principal base call at any consensus position. However, with new technologies having multiple high intensities per cycle would be an error.
- There was consensus that one should not archive images even for limited rescoring R+D purposes. This is because such research will be taking place at the Centers, and for those who want to engage in it, they should go buy an instrument, rather than relying on the SRA.
- Only a subset of quality values in the old phred range are actually used by these technologies and it's not clear what they represent. Can fewer bits be used to encode them (other than the current 7 needed to encode 1-100)?
- here was a question about whether it would be necessary to store negative values for processed intensities, or simply truncate them to zero. This could increase the SFF file size by as much as 10%.

2.3. Should the results of secondary analysis be archived?

A major part of the value provided by the new technologies is resequencing. Alignments to reference sequences, and analysis derived from these alignments, provide the primary starting point for investigators. There was a discussion about whether these should therefore be captured. In some cases Centers are performing their own secondary analysis. In order to control the scope of the SRA, only primary analysis results (the results of processing a sequencing run without respect to a reference) would be archived. The expectation is that other analyses would be archived by downstream resources even if these don't exist at the moment.

3. Database Structure (led by Vladimir Alekseyev)

3.1. Should project registration be required for all SRA submissions?

The proposal is to ask submitters to describe their project in terms of an experimental design. Such a descriptor could be linked in as a first class object in the Entrez system.

- There is an effort underway to develop an international project id (ISNCD) that would represent project tracking information mirrored at NCBI, Ensembl, DDJB, and possibly other archives. So NCBI_PROJECT_ID will be subsumed by this new id.

- If we separate project metadata from data submissions, then there will be an asynchrony problem. Others suggested one should allow, for small projects at least, unified submission of meta data and data ("in-line" submission feature).
- The group wanted careful definition of fields in a RFC type document. There was a discussion about how deeply to represent the project meta data. The consensus seemed to be that some metadata will be useful to be able to query against when extracting data from the SRA, but these should not be required fields, nor should we attempt to design an ontology for the various experiments as these are being addressed by other efforts (for example CAMERA and Gemina). Also, the meta data acquisition should be flexible, allowing for center defined tag-value pairs.
- One observation is that meta data submitted and stored on the project level will be small so there is no need to make it efficient. Also, the SRA submission process may accept meta data as a proxy for other resources, some of which have yet to be designed. The SRA itself is not intended to track project metadata.

Another popular feature will be "hold until publish". Now that submissions are tracked by experiment or project, this will be easy to implement.

3.2. Introduction to the "Spot" Abstraction

A common property of the new technologies is that they gather, through image processing, one intensity function for each reaction container (well/spot/bead), which we will call a "spot". Adapters, paired end reads, linkers, bar codes, and other subsequences can be represented as annotations on the native read that partition it. In order to access the usable sequence itself, or the other component "technical reads", the SRA would supply through its meta data directives for how to parse the native sequence to extract these objects.

All reads would receive an accession in the form run.spot.read. This accessioning scheme is indexed, rather than random access. Thus reads do not have a "name" as such. The scheme presented encodes enough information to locate a read individually while eliminating the need to store a tag for each read that would use almost as much space as the read itself. The accession is stable in that it is decided at the time of submission. Therefore, any downstream process will be able to refer to it so long as the order within the submission remains unchanged.

- There was quite a lot of discussion about whether reads need to be named. The consensus developed that doing so would be too expensive in terms of space requirements. The SRF format will be supporting read names, but this was done in order to address an application space beyond SRA. Therefore, SRA should not archive read names, although these can be used in submissions.
- There will be a need to call out in both SRF and SRA whether encoding of intensity values is in terms of base space, flow space, or color space.
- There was a discussion about whether padding the accession string is a good idea. EMBL is trying to migrate away from that. On the other hand, it is convenient for searching and sorting to have fixed length strings.

- There is a need to make the notion of an experiment flexible. It should be as big as a genome project and as small as a lane or region.
- There is a need to allow for incremental submissions, particularly when data sets are huge.
- There was a discussion about whether to allow for many-to-many experiment to project mapping. Right now the abstraction says that a project or study is composed of one or more experiments, each of which may generate a run of data.
- Do we need to know the total number of spots in the submission or expected total for the experiment?

3.3. Can we reasonably expect to submit, archive, and download all this data?

There was a lively and perhaps inconclusive discussion about whether the level of detail being proposed in the SRA will result in an unmanageable torrent of data. It was proposed that even under full compression, 1 GB of sequencing data will result in 10 GB of storage data, and that with 100-200 new technology instruments producing each week this could amount to deposits of 1 TB per day. Has there been any planning or modeling of what might happen if this situation were to materialize?

A related question is whether centralization of the archive makes any sense. Would it not be better to provide a central indexing service that leads back to the Centers, who will actually provision the data requested by users? Centers responded by saying that they do not want to be in the business of satisfying user requests for data, and that they were looking to NCBI to handle this.

4. Database Submission Format (led by Asim Siddiqui)

The SRF format was reviewed. Issues about read ids vs. read names were debated.

SRF is a separate effort that will hopefully conclude with a 1.0 specification sometime in August.

5. Data Retrieval (led by Gabor Marth)

This discussion tried to anticipate uses for the SRA. Some key points:

- here is a need to track the provenance of the source material (DNA). How was it isolated, was it methyl filtrated etc. These would have bearing on library construction and assembly. At the same time, one should not try to invent an ontology to describe this, just useful fields.
- Another need for data tracking is library stats: expected insert size etc.
- A discussion took place about whether to provision all the data from a run, or actively quality filter the data down to the "usable" subset. While this might be convenient for some applications, it is also fraught with issues. Historically, the Trace Archive accepted all data from a run regardless of quality level. Also, the issue

of whether something is usable because of low quality of contamination is often not knowable until downstream processes are applied.
- A similar discussion ensued about accepting reads that did not align to the reference sequence used in the experiment. The observation was made that not aligning to a reference sequence is not a reason to not submit.

5.1. How will assemblies use this data?

There is a localization issue when referring to reads in the assembly or alignment context. If reads are accessed in storage order, then the time needed to perform random access retrieval will dominate any assembly download or display function. Therefore, reads will have to be reordered. The question arises whether the SRA will do this on retrieval. Various proposals include using a prefix on read ids in order to embed tracking information needed for localization. Then there would need to be a directive that would tell the output streamer to formulate the ids in a certain way.

5.2. What are the units of retrieval?

The use case for the short reads may determine the retrieval chunks. They could be run/region or plate/slide/lane order, or some other locality. There may be context-driven retrieval. This area will require further requirements development.

Everybody agreed that the user should be informed as to the expected size and time of the data download, and for the user to have the opportunity to cancel it. A web tool that would report download status similar to submission status might be warranted.

6. Data Submission Software (led by Guy Cochrane)

There will be three submission activities for a project:

1. Registration activity (email, web?)
2. Meta data submission to SRA (xml)
3. Experimental results submission to SRA (srf)

These activities might happen at different times, or the same time for small projects.

6.1. How does one mask off data that is not part of the experiment?

One of the issues is how to deal with contaminants. These are often not found until relatively late in the project life cycle, which may be well after the sequencing data have been submitted. There was a debate about whether tracking of contaminants should be the responsibility of the SRA, or downstream archives. Clearly it is convenient to be able to download a contaminant-free dataset.

Therefore, we may need facilities to:

- Mask data within a run
- Suppress data within a submission

- Withdraw a submission

6.2. Should data that cohabit a run but otherwise are not related share a submission ?

There was a consensus that Centers should endeavor to split up unrelated portions of a run so that each portion maps to an experiment. But this may require development of SRF slicing and dicing utilities. There was a suggestion to publish the specs and interfaces for such utilities, and let the vendors develop these.

7. Policy Issues: Recommendations to Funding Agencies (led by Elaine Mardis)

7.1. Will there be support for medical resequencing?

No. The policy development is underway, but we should assume this is not in scope at this time.

7.2. Is it ok to just submit bases and qualities?

There was a discussion about whether one should allow archival bases and quality data only. This would certainly be simpler and faster. But experience with Trace Archive was that in the long run requiring archival of the intensity files was very rewarding, and that migrating early submissions proved impossible. Another issue supporting full disclosure of intensity data is that vendors will want maximal representation of their data. Leaving out intensities will raise more questions than answers. Also, what is the value added by using four channels of quality scoring if intensity data is tracked? Finally, it is in general it is difficult to later change the rules to make them more stringent.

8. Summary (led by Deanna Church)

- It appears that the data model proposed is adequate.
- SRF is the submission and probably retrieval medium.
- SRF will be supported by all the vendors.
- Read names will be accepted but not tracked by the SRA.
- Read names might be auto generated by the SRA.
- Distinct project registration is important, but there should be an inline solution.
- SRA should be tracking meta data at the levels of experiment and run.
- NCBI will issue a straw man xml schema and gather further comments.
- NCBI aims for an October release of the SRA.
- SRF will be finalized in August.

9. Supplementary items

Vladimir Alekseyev's Presentation for the meeting

XML Specification Version 1.2

Created: September 24, 2010; Updated: October 29, 2010.

Status	Active
Active Date	2010-11-01
Inactive Date	2011-08-01
Scope	INSDC SRA

1. Overview

This document summarizes the proposed changes for Release 1.2 of the Sequence Read Archive (SRA) schemas governing XML metadata. Release 1.2 is an expansion of Release 1.1, which was in introduced in March 2010. The goal of this release is to update choices, introduce new features, and specify a usable Analysis object usable for BAM file submissions. These changes are being introduced with the objective of not invalidating any current valid XML documents.

Major new features in this release are:

- New Analysis schema supports BAM file submissions
- Respecification of processing pipeline and directives
- Reinstantiation of spot descriptor, platform, and processing blocks at the level of SRA Run.
- Addition of choices to many controlled vocabularies

1.1. Notice

The features and modalities described in the XML schema DO NOT constitute a statement of features and mechanisms available in the SRA. The schema changes frequently must precede actual implementation. New feature rollouts and functionality changes are made asynchronously with XML schema changes.

1.2. Related Documents

The SRA schema for this release can be obtained from this site: http://www.ncbi.nlm.nih.gov/viewvc/v1/trunk/sra/doc/SRA_1-2

1.3. Revision History

Drafts A-K created 17 May 2010 and updated through 20 Oct 2010. Approved for release by INSDC partners 20 Oct 2010. Scheduled for release 01 Nov 2010.

2. Explanation of Changes

2.1. Changes to All Documents

2.1.1. LinkType extended

LinkType redefined to include the a choice of the following link types

- SRA_LINK
- URL_LINK
- XREF_LINK
- ENTREZ_LINK
- DDBJ_LINK
- ENA_LINK

2.1.2. New SRA.common.xsd

- SRA.common.xsd factored out of the "COMMON BLOCK" that was included with each SRA schema. New namespace called com: created for commonly used types. [EBI]

To import this file, do

```
<xs:import schemaLocation="SRA.common.xsd" namespace="SRA.common"/>
```

To use a feature, do

```
<xs:element name="STUDY_ATTRIBUTE" type="com:AttributeType"/>
```

- SRA.*.xsd now imports SRA.common.xsd
- Common Block has been refactored to the SRA.common.xsd under namespace com:
- SpotDescriptorType refactored to SRA.common.xsd
- PlatformType refactored to SRA.common.xsd
- ProcessingType refactored to SRA.common.xsd

2.2. Changes to SRA Experiment

2.2.1. Add new instrument models

New instrument values have been added to Experiment :

- "Illumina HiSeq 2000" [Illumina],
- "AB SOLiD 4 System" [LifeTech],
- "AB SOLiD 4hq System" [LifeTech],
- "AB SOLiD PI System" [LifeTech],
- "454 GS Junior" [Roche/454],
- "454 GS FLX Titanium" [Roche/454], to succeed "454 Titanium"
- "Illumina Genome Analyzer IIx" [Illumina]

Note that the use of instrument model in Run was deprecated in version 1.1.

2.2.2. Changed EXPERIMENT/PLATFORM/ILLUMINA/CYCLE_COUNT

Changed this to optional field to eliminate need to always specify a deprecated field. [BI]

2.2.3. Add new library strategy/library selection combinations

New values for LIBRARY_STRATEGY and LIBRARY_SELECTION have been added to Experiment [EDACC]

- Methylation-Sensitive Restriction Enzyme Sequencing strategy.

<LIBRARY_STRATEGY>MRE-Seq</LIBRARY_STRATEGY>

<LIBRARY_SELECTION>Restriction Digest</LIBRARY_SELECTION>

- Methylated DNA Immunoprecipitation Sequencing strategy.

<LIBRARY_STRATEGY>MeDIP-Seq</LIBRARY_STRATEGY>

<LIBRARY_SELECTION>5-methylcytidine antibody</LIBRARY_SELECTION>

- RNA-Seq strategy

A new choice RNA-Seq was added to LIBRARY_STRATEGY to support the general choice for sequencing that targets total RNA, with the following new choices for LIBRARY_SELECTION (others are possible):

- CAGE
- RACE
- Size fractionation
    - Direct sequencing of methylated fractions sequencing strategy.

<LIBRARY_STRATEGY>MBD-Seq</LIBRARY_STRATEGY>

<LIBRARY_SELECTION>MBD2 protein methyl-CpG binding domain</LIBRARY_SELECTION>

This combination entails direct sequencing of methylated fractions following enrichment by methyl-CpG binding domain

- Whole exome sequencing strategy

"WXS" (whole exome sequencing) as a library strategy. [ESP-GO]

2.2.4. Improved documentation for spot descriptor choices.

2.2.5. New Library Source terms

Added TRANSCRIPTOMIC and METAGENOMIC to EXPERIMENT/LIBRARY_DESCRIPTOR/LIBRARY_SOURCE as a way to give further detail to

submitters formerly using NON_GENOMIC (which was a holdover choice from the Trace Archive).

2.2.6. PLATFORM nodes

Make all the nodes in PLATFORM consistent to allow for universal query of instrument model.

- EXPERIMENT.PLATFORM.COMPLETE_GENOMICS.INSTRUMENT_MODEL=none
- EXPERIMENT.PLATFORM.PACBIO_SMRT.INSTRUMENT_MODEL=none

2.2.7. Change to sample pool descriptor

SAMPLE_DESCRIPTOR.POOL.MEMBER.READ_LABEL made optional, to support pools that

are not barcoded (and therefore don't need a read label). [BI]

2.2.8. Restored expected_number_runs

The attribute expected_number_runs restored (un-deprecated). [EDACC]

This field is actually being used on one roadmap project.

2.2.9. Added TARGETED_LOCI block

Added "TARGETED_LOCI" as a library element [HMP, TCGA]. This block allows the submitter to specify one or more gene target(s) or probe set(s) used by the targeted sequencing or hybridization array.

A controlled list will be offered, to consist initially of

- 16S rRNA
- exome
- other

where the submitter can add in free text to identify alternate locus or refine the description..

2.2.10. Added POOLING_STRATEGY

Added POOLING_STRATEGY as a library element, to help indicate the

sample multiplexing intent of the submitter. Choices include:

- None
- Simple pool
- Multiplexed samples
- Multiplexed libraries
- Spiked library

- Other

This block is added at the level of the library design because in the future sample pools may referenced as an element in BioSamples, rather than a pool spec in SRA experiment.

2.2.11. Added default_ length, base_coord attributes to SPOT_DESCRIPTOR

Added **default_ length**, **base_coord** attributes to EXPECTED_BASECALL and EXPECTED_BASECALL_TABLE. The default_length parameter can specify whether the spot should have a default length for the tag. If provided, the specified number of bases is assigned to this tag regardless of matching criteria. If 0, or not provided, then the tag is "missed" if the match criteria fail. Moreover, submitters should switch to using the EXPECTED_BASECALL_TABLE in preference to EXPECTED_BASECALL. [NCBI, BI]

2.2.12. Removed requirement for fields in PROCESSING.QUALITY_SCORES

Removed requirement for deprecated fields in EXPERIMENT.PROCESSING.QUALITY_SCORES [BI]

- <xs:element name="NUMBER_OF_LEVELS" maxOccurs="1" minOccurs="0" type="xs:int"/>
- <xs:element name="MULTIPLIER" maxOccurs="1" minOccurs="0" type="xs:double"/>

2.2.13. New PIPELINE spec in PROCESSING

New element PIPELINE in ProcessingType added to describe the pipeline used in processing the data. This includes a way to specify the sequence of steps in the processing pipeline, programs and their versions, and processing directives. This was simplified from an earlier proposal, now there is simply a sequence of steps. Here is an example of an acceptable processing block under SRA 1.2:

```
<PROCESSING>
  <PIPELINE>
    <PIPE_SECTION section_name="base caller">
      <STEP_INDEX>1.0</STEP_INDEX>
      <PREV_STEP_INDEX>NIL</PREV_STEP_INDEX>
      <PROGRAM>454BaseCaller</PROGRAM>
      <VERSION>1.1.01.20</VERSION>
    </PIPE_SECTION>
    <PIPE_SECTION section_name="SRA conversion">
      <STEP_INDEX>1.1</STEP_INDEX>
      <PREV_STEP_INDEX>1.0</PREV_STEP_INDEX>
      <PROGRAM>toSRA</PROGRAM>
      <VERSION>1.34</VERSION>
    </PIPE_SECTION>
  </PIPELINE>
</PROCESSING>
```

2.2.14. New PROCESSING_DIRECTIVES spec in PROCESSING

A new feature is the explicit enumeration of treatments to the data applied by the submitter, or requested treatments of the data requested by the submitter to be applied by the Archive. Initially this will cover the sample multiplexing directives (no demultiplexing, submitter demultiplexed), but will be expanded in the future to track all treatment requests.

2.3. Changes to Study

2.3.1. New STUDY_TYPE choices

- Exome Sequencing
- Pooled Clone Sequencing.

These were requested by Sanger/EBI.

2.3.2. CENTER_NAME deprecated

The submission and ownership is adequately tracked in the related SUBMISSION object, and the STUDY@center_name attribute.

2.3.3. RELATED_STUDIES

RELATED_STUDIES is intended to be used as a mechanism to bind the record to the emerging BioProject record (successor to genomeprj record), as well as binding to other resources that track studies (GEO and dbGaP at NCBI, and EGA and ArrayExpress at EBI). This feature should replace the use of PROJECT_ID, a holdover from the Trace Archive, and which has been deprecated.

In order to negotiate a design error in SRA 1.1, a new branch choice has been created called RELATED_STUDY that should be used in preference to the currently effective STUDY, which is now deprecated. The link to a named database was implemented (XRefLinkType) in order to constrain the choice of related project to a named database.

```
<xs:element name="RELATED_STUDY" maxOccurs="unbounded" minOccurs="1">
  <xs:complexType>
    <xs:sequence>
      <xs:element name="RELATED_LINK" type="com:XRefType"
         minOccurs="1" maxOccurs="1">
        <xs:annotation>
          <xs:documentation>
            Related study or project record from a list of supported databases.
            The study's information is derived from this project record rather
            than stored as first class information.
          </xs:documentation>
        </xs:annotation>
      </xs:element>
      <xs:element name="IS_PRIMARY" type="xs:boolean"
       minOccurs="1" maxOccurs="1">
        <xs:annotation>
```

```
          <xs:documentation>
            Whether this study object is designated as the primary source
            of the study or project information.
          </xs:documentation>
        </xs:annotation>
      </xs:element>
    </xs:sequence>
  </xs:complexType>
</xs:element>
```

2.4. Changes to Sample

There are no changes to SRA Sample in this revision.

2.5. Changes to Submission

In SUBMISSION, added required "schema" attribute to MODIFY action in order to force submitter to specify the namespace of the intended target. "target" is made optional, and will be ignored. Henceforth, the MODIFY source file will contain all the needed references. [BI]

2.6. Changes to Run

2.6.1. Replicated descriptors at Run level

- Replicated SPOT_DESCRIPTOR at the level of Run. If specified at Run, it will override the setting at the level of Experiment.
- Replicated PLATFORM at the level of Run. If specified at Run, it will override the setting at the level of Experiment.
- Replicated PROCESSING at the level of Run. If specified at Run, it will override the setting at the level of Experiment.

2.6.2. New Filetype support

- Added bam as a filetype for RUN.
- Added kar as a supported filetype for RUN, as native SRA format in serialized form.

2.7. Respecified ANALYSIS object

2.7.1. Removed deprecated branches:

- ANALYSIS_TYPE/REPORT
- ANALYSIS_FILES/FILE/filetype/.pdf
- ANALYSIS_FILES/FILE/filetype/.sam (will be delivered in .bam only)

2.7.2. Specified REFERENCE_ALIGNMENT branch

The ANALYSIS/ANALYSIS_TYPE/REFERENCE_ALIGNMENT has been completely specified in order to serve as the metadata container for alignment files delivered in BAM format. Several mechanisms have been furnished to allow submitters to specify the

reference sequence. Some additional business rules may also be applied by each Archive to constrain reference choices.

2.8. New SRA Package Object

A new schema SRA.package.xsd has been introduced in order to provide a container for any combination of SRA XML documents, and to allow for applications using SRA objects to aggregate them in any form. SRA packages are not now supported for submission, but eventually will be used in preference to tar archive files.

3. Deprecated Fields

SRA 1.2 contains the following fields, branches, and options that should no longer be used in current submissions.

SRA.common.xsd	SPOT_DECODE_METHOD	
SRA.common.xsd	NUMBER_OF_READS_PER_SPOT	
SRA.common.xsd	'454 Titanium'	use '454 GS FLX Titanium'
SRA.common.xsd	'GS 20'	use '454 GS 20'
SRA.common.xsd	'GS FLX'	use 'GS FLX'
SRA.common.xsd	'Solexa 1G Genome Analyzer'	use 'Illumina Genome Analyzer'
SRA.common.xsd	CYCLE_SEQUENCE	use SEQUENCE_LENGTH
SRA.common.xsd	CYCLE_COUNT	use SEQUENCE_LENGTH
SRA.study.xsd	CENTER_NAME	use STUDY@center_name
SRA.study.xsd	PROJECT_ID	use RELATED_STUDIES instead
SRA.study.xsd	RELATED_STUDIES/STUDY	use RELATED_STUDIES/ RELATED_STUDY instead
SRA.experiment.xsd	LIBRARY_STRATEGY/BARCODE	use another library strategy
SRA.experiment.xsd	LIBRARY_SOURCE/NON GENOMIC	use METAGENOMIC or TRANSCRIPTOMIC instead
SRA.experiment.xsd	PROCESSING/BASE_CALLS	use PIPELINE instead
SRA.experiment.xsd	PROCESSING/QUALITY_SCORES	use PIPELINE instead
SRA.experimentxsd	@expected_number_spots	
SRA.experimentxsd	@expected_number_reads	
SRA.run.xsd	'_seq.txt, _prb.txt, _sig2.txt, _qhg.txt'	use 'Illumina_native' instead
SRA.run.xsd	@total_spots	
SRA.run.xsd	@total_reads	

Table continues on next page...

Table continued from previous page.

SRA.run.xsd	@number_channels	
SRA.run.xsd	@format_code	
SRA.run.xsd	@instrument_model	use PLATFORM/INSTRUMENT_MODEL instead
SRA.run.xsd	@run_file	
SRA.run.xsd	@total_data_blocks	
SRA.submission.xsd	HoldForPeriod	
SRA.submission.xsd	@submission_id	use alias instead
SRA.submission.xsd	@handle	

4. Future Planned Revisions

The next revision is anticipated to be contracting revision (one that potentially invalidates current documents). The main changes will be to remove deprecated fields. This will involve migration of data in anticipation of future schema changes.

SRA XML Schema 1.3 Release Notes

Draft C – 14 Jul 2011

Created: January 14, 2011; Updated: August 11, 2011.

Status	Active
Active Date	2011-08-11
Inactive Date	
Scope	INSDC SRA

1. Overview

This document summarizes the proposed changes for Release 1.3 of the Sequence Read Archive (SRA) schemas governing XML metadata. This schema will be used by the SRA archive instances and has been developed under the auspices of the International Nucleotide Sequence Database Collaboration (INSDC, insdc.org).

Release 1.3 is a change over Release 1.2, which was in introduced in October 2010. While the schemas are incompatible, all data have been migrated so that documents submitted or modified before release remain valid. The goal of this release is to update choices, introduce new features, and specify an Analysis object usable for BAM file submissions. These changes are being introduced with the objective of not invalidating any current valid XML documents.

Major new features in this release are:

- Addition of new instrument models
- Require certain fields that have already been migrated
- Allow for modification of already-loaded analysis objects

1.1. Notice

The features and modalities described in the XML schema DO NOT constitute a statement of features and mechanisms available in the SRA. The schema changes frequently must precede actual implementation. New feature rollouts and functionality changes are made asynchronously with XML schema changes. Each SRA implementation by INSDC partners may impose additional business rules not reflected in the schema.

1.2. Related Documents

The SRA schema for this release can be obtained from this site: http://www.ncbi.nlm.nih.gov/viewvc/v1/trunk/sra/doc/SRA_1-3

1.3. Revision History

Drafts C- 2011-07-14 for approval by INSDC partners

2. Explanation of Changes

2.1. Changes to All Documents

2.1.1. Adjustment to import statements

All document importing SRA.common.xsd now point to a resolvable URL:

http://www.ncbi.nlm.nih.gov/viewvc/v1/trunk/sra/doc/SRA/SRA.common.xsd?view=co

Note that this will be implemented on the final distribution copy of the schema files.

2.2. Changes to SRA.Common.xsd

2.2.1. Add new instrument models

New instrument values have been added to Platform block :

- "Illumina HiSeq 1000" [Illumina],
- "Illumina MiSeq" [Illumina],
- AB SOLiD 5500xl SOLiD System
- AB SOLiD 5500 SOLiD System
- "PacBio RS" [Pacific Biosciences]
- "Complete Genomics" [Complete Genomics] (platform already exists)
- ION_TORRENT (new platform) and instrument models:
    - Ion Torrent PGM

2.2.1. Remove deprecated instrument models

- Solexa 1G Genome Analyzer (use Illumina choices)
- GS 20 (use 454 GS 20)
- GS FLX (use 454 GS FLX)
- 454 Titanium (use 454 GS FLX Titanium)

2.2.2. Add GapDescriptor

A new structure called the GapDescriptor is introduced that will encode the placement of spot subsequences (tags) against a reference or assembly substrate. This structure encodes mate pair gaps and tandem read gaps. It is possible to express gaps distances in three ways: as mean/standard deviation, as min-max range, and as histogram. Orientation of the tag pairs can be described as “innie”, “outie”, “normal”, and strand-opposite “anti-normal”, following the nomenclature of the Celera Assembler.

Introduction of the GapDescriptor element was motivated by the need to describe CompleteGenomics platform sequencing. It is also intended that the GapDescriptor replace the LIBRARY_LAYOUT element in the LibraryType. The GapDescriptor can be specified at the level of Run in order to override any general settings at the level of experiment.

2.2. Changes to SRA Experiment

2.2.2. New Library Source choice METATRANSCRIPTOMIC

This was requested by EBI.

2.2.3. Removed deprecated library strategy choice BARCODE

This change was requested by EBI. No records have this designation.

2.3. Changes to Study

2.3.1. The RELATED_STUDIES/STUDY block removed

In preparation for migration to BioProjects, this deprecated block has been removed.

2.4. Changes to Sample

2.4.1. TAXON_ID now required

The TAXON_ID field in the SAMPLE_NAME block is now required. All records already have this.

2.5. Changes to Submission

2.5.1. Submission handle removed

The deprecated SUBMISSION/@handle attribute has been removed.

2.5.2. Submission submission_id removed

The deprecated SUBMISSION/@submission_id attribute has been removed.

2.5.3. PROTECT action is now a complex type

This is a technical improvement requested by a major submitter.

2.6. Changes to Run

2.6.1. Replicated descriptors at Run level

- Replicated GAP_DESCRIPTOR at the level of Run. If specified at Run, it will override the setting at the level of Experiment.
- Replicated SPOT_DESCRIPTOR at the level of Run. If specified at Run, it will override the setting at the level of Experiment.
- Replicated PLATFORM descriptor at the level of Run. If specified at Run, it will override the setting at the level of Experiment.
- Replicated PROCESSING descriptor at the level of Run. If specified at Run, it will override the setting at the level of Experiment.

2.6.2. Require checksum and checksum method

The DATABLOCK/FILES/FILE/@checksum and @checksum_method are now required attributes.

2.6.3. New Filetype Choice

The filetype option "PacBio_HDF5" has been created to support the native loader for PacBio.

2.6.3. Old Filetype removed

The filetype option "_seq.txt, _prb.txt, _sig2.txt, _qhg.txt" has been eliminated in favor of "Illumina_native".

2.7. Changes to Analysis

2.7.1. DATA_BLOCK not required for modification

The DATA_BLOCK is now required for add submissions, but no longer for modify submissions.

3. Deprecated Fields

SRA 1.3 contains the following fields, branches, and options that should no longer be used in current submissions.

Document	Field	Use instead
SRA.common.xsd	SPOT_DECODE_METHOD	
SRA.common.xsd	NUMBER_OF_READS_PER_SPOT	
SRA.common.xsd	'454 Titanium'	use '454 GS FLX Titanium'
SRA.common.xsd	'GS 20'	use '454 GS 20'
SRA.common.xsd	'GS FLX'	use '454 GS FLX'
SRA.common.xsd	'Solexa 1G Genome Analyzer'	use 'Illumina Genome Analyzer'
SRA.common.xsd	FLOW_SEQUENCE	
SRA.common.xsd	KEY_SEQUENCE	
SRA.common.xsd	FLOW_COUNT	Use SPOT_LENGTH
SRA.common.xsd	CYCLE_SEQUENCE	
SRA.common.xsd	CYCLE_COUNT	use SPOT_LENGTH
SRA.common.xsd	SEQUENCE_LENGTH	Use SPOT_LENGTH
SRA.study.xsd	CENTER_NAME	use STUDY@center_name

Table continues on next page...

Table continued from previous page.

SRA.study.xsd	PROJECT_ID	use RELATED_STUDIES instead
SRA.study.xsd	RELATED_STUDIES/STUDY	use RELATED_STUDIES/ RELATED_STUDY instead
SRA.experiment.xsd	LIBRARY_STRATGEY/BARCODE	use another library strategy
SRA.experiment.xsd	LIBRARY_SOURCE/NON GENOMIC	use METAGENOMIC or TRANSCRIPTOMIC instead
SRA.experiment.xsd	PROCESSING/BASE_CALLS	use PIPELINE instead
SRA.experiment.xsd	PROCESSING/QUALITY_SCORES	use PIPELINE instead
SRA.experiment.xsd	@expected_number_spots	
SRA.experiment.xsd	@expected_number_reads	
SRA.run.xsd	'_seq.txt, _prb.txt, _sig2.txt, _qhg.txt'	use 'Illumina_native' instead
SRA.run.xsd	@total_spots	
SRA.run.xsd	@total_reads	
SRA.run.xsd	@number_channels	
SRA.run.xsd	@format_code	
SRA.run.xsd	@instrument_model	use PLATFORM/INSTRUMENT_MODEL instead
SRA.run.xsd	@run_file	
SRA.run.xsd	@total_data_blocks	
SRA.submission.xsd	HoldForPeriod	
SRA.submission.xsd	@submission_id	use alias instead
SRA.submission.xsd	@handle	

4. Future Planned Revisions

The next revision is anticipated to be contracting revision (one that potentially invalidates current documents). The main changes will be to remove deprecated fields. This will involve migration of data in anticipation of future schema changes.

SRA XML Schema 1.4 Release Notes

Final – 10 May 2012

Created: February 9, 2012; Updated: May 10, 2011.

Status	Active
Active Date	2012-05-15
Inactive Date	
Scope	INSDC SRA

Overview

This document summarizes the proposed changes for Release 1.4 of the Sequence Read Archive (SRA) schemas governing XML metadata. This schema will be used by the SRA archive instances and has been developed under the auspices of the International Nucleotide Sequence Database Collaboration (INSDC, insdc.org).

Release 1.4 is an expansionary change over Release 1.3, which was in introduced in August 2011. These changes are being introduced with the objective of not invalidating any currently valid XML documents.

Major new features in this release are:

- Addition of new instrument platform, CAPILLARY, and new instrument models
- Enhancement of choices for library and experiment
- Introduction of an IDENTIFIERs block to track multiple active and inactive accessions and IDs.
- Support for BAM file submission through SRA Run

Notice

The features described in the SRA XML schema DO NOT constitute a statement of features and mechanisms available in the SRA. The schema changes frequently must precede actual implementation. New feature rollouts and functionality changes are made asynchronously with XML schema changes. Each SRA implementation by INSDC partners may impose additional business rules not reflected in the schema.

Related Documents

The SRA schema for this release can be obtained from this site: http://www.ncbi.nlm.nih.gov/viewvc/v1/trunk/sra/doc/SRA_1-4

Using_the_SRA_Identifier_Block.pdf

Revision History

Draft D- 2012-04-20 submitted for approval by INSDC partners

Draft E- 2012-04-30 submitted for approval by INSDC partners

Draft F- 2012-05-07 submitted for approval by INSDC partners

Draft G- 2012-05-09 submitted for approval by INSDC partners

Final – 2012-05-10 approved for release by INSDC partners

Explanation of Changes

Introduction of IDENTIFIERS block to all documents

An IDENTIFIERS block of IdentifierType has been added to all documents. This block is intended to give more flexibility in how IDs are tracked. IDs include primary and secondary accessions, equivalent records in other databases, submitter primary and secondary names for records. The number of IDs is unbounded. Whether they are active or not (replaced or deprecated) can be indicated. A uuid (universally unique ID) ID type is supported, although this will not be used by INSDC SRA archives.

Changes to PlatformType

Add new platform CAPILLARY

This platform choice is intended to support handling of Traces in the SRA. The instrument model choices are:

- AB 3730xL Genetic Analyzer
- AB 3730 Genetic Analyzer
- AB 3500xL Genetic Analyzer
- AB 3500 Genetic Analyzer
- AB 3130xL Genetic Analyzer
- AB 3130 Genetic Analyzer
- AB 310 Genetic Analyzer

Add new instrument models

New instrument values have been added to Platform block :

- Remove “none” as an instrument model for Complete Genomics, PacBio
- Correct name for AB 5500, 5500xl instruments
- Add 454 FLX+
- Add AB SOLiD 3.0 plus
- Add Illumina HiSeq 2500
- Add Illumina HiScanSQ
- Add Ion Proton

Changes to LibraryDescriptorType

Added Library Strategies

- WGA (whole genome amplification) to replace some instances of RANDOM
- Added miRNA-Seq for micro RNA and other small non-coding RNA sequencing
- Added Tn-Seq for gene fitness determination through transposon seeding.

Added Library Selections

- Added MDA (multiple displacement amplification)
- Added Padlock Probes capture strategy to be used in conjuction with Bisulfite-Seq

Make LibraryName optional

The LibraryName field is not needed except from bulk submitters who may submit multiple experiments per library.

Added options to TARGETED_LOCI

The PROBE_SET block was made optional (a technical change). In addition, the following items were added to the locus attribute:

- 18S ribosomal RNA
- RBCL
- matK
- COX1
- ITS1-5.8S-ITS2

Changes to RunType

New Filetypes Added

In order to support submission reference alignments in BAM format, the filetypes table has been augmented with these new filetypes:

- BAM header
- Reference fasta
- Complete Genomics native

Title block added

The TITLE block will allow for expansion of run to include all sequencing for the experiment or to include a certain logical fraction. The TITLE block can be used to distinguish which fraction.

Changes to ExperimentType

SPOT_DESCRIPTOR made optional in Experiment

The SPOT_DESCRIPTOR block is used by the loader to cognate the input data during load into the SRA. If the data are never transformed, then it can serve as the permanent map of the layout of the reads in the run. In order to refactor information needed for loading or interpreting the read layout, this block should be used in the RUN instead. For BAM loads it is not needed at all.

Modify GapDescriptorType

Some changes to the schema for the GapDescriptor have been implemented in order to better support Complete Genomics libraries. There are as yet no deposited experiments with GapDescriptor blocks in them, so this change will be benign.

Changes to Submission

The SUBMISSION/FILES block has been deprecated. Use the DATA_BLOCK/FILES instead.

Deprecated Fields

SRA 1.4 contains the following fields, branches, and options that should no longer be used in current submissions.

Field	Notes
`/STUDY/DESCRIPTOR/CENTER_NAME`	1
`/STUDY/DESCRIPTOR/PROJECT_ID`	2
`/EXPERIMENT/@expected_number_reads`	

Notes
1. Use document header attribute @center_name
2. Use STUDY/RELATED_STUDIES/RELATED_STUDY
3. n/a
4. Use TRANSCRIPTOMIC or METAGENOMIC or METATRANSCRIPTOMIC
5. Use AB 5500 Genetic Analyzer or AB 5500xl Genetic Analyzer
6. Use PIPELINE
7. Use PLATFORM/*/INSTRUMENT_MODEL
8. Use DATA_BLOCK/FILES/FILE/filetype, DATA_BLOCK/FILES/FILE/checksum

Table continues on next page...

Table continued from previous page.

Field	Notes
/EXPERIMENT/@expected_number_spots	
/EXPERIMENT/DESIGN/LIBRARY_DESCRIPTOR/LIBRARY_SOURCE[NON GENOMIC]	4
/EXPERIMENT/DESIGN/SPOT_DESCRIPTOR/SPOT_DECODE_METHOD	
/EXPERIMENT/DESIGN/SPOT_DESCRIPTOR/SPOT_DECODE_SPEC/ NUMBER_OF_READS_PER_SPOT	
/EXPERIMENT/LIBRARY/LIBRARY_DESCRIPTOR/LIBRARY_SOURCE[NON GENOMIC]	4
/EXPERIMENT/LIBRARY/SPOT_DESCRIPTOR/SPOT_DECODE_METHOD	
/EXPERIMENT/LIBRARY/SPOT_DESCRIPTOR/SPOT_DECODE_SPEC/ NUMBER_OF_READS_PER_SPOT	
/EXPERIMENT/PLATFORM/ABI_SOLID/COLOR_MATRIX	
/EXPERIMENT/PLATFORM/ABI_SOLID/COLOR_MATRIX_CODE	

Notes
1. Use document header attribute @center_name
2. Use STUDY/RELATED_STUDIES/RELATED_STUDY
3. n/a
4. Use TRANSCRIPTOMIC or METAGENOMIC or METATRANSCRIPTOMIC
5. Use AB 5500 Genetic Analyzer or AB 5500xl Genetic Analyzer
6. Use PIPELINE
7. Use PLATFORM/*/INSTRUMENT_MODEL
8. Use DATA_BLOCK/FILES/FILE/filetype, DATA_BLOCK/FILES/FILE/checksum

Table continues on next page...

Table continued from previous page.

Field	Notes
/EXPERIMENT/PLATFORM/ABI_SOLID/CYCLE_COUNT	
/EXPERIMENT/PLATFORM/ABI_SOLID/INSTRUMENT_MODEL[AB SOLiD 5500]	5
/EXPERIMENT/PLATFORM/ABI_SOLID/INSTRUMENT_MODEL[AB SOLiD 5500xl]	5
/EXPERIMENT/PLATFORM/ABI_SOLID/SEQUENCE_LENGTH	
/EXPERIMENT/PLATFORM/HELICOS/FLOW_COUNT	
/EXPERIMENT/PLATFORM/HELICOS/FLOW_SEQUENCE	
/EXPERIMENT/PLATFORM/ILLUMINA/CYCLE_COUNT	
/EXPERIMENT/PLATFORM/ILLUMINA/CYCLE_SEQUENCE	
/EXPERIMENT/PLATFORM/ILLUMINA/SEQUENCE_LENGTH	
/EXPERIMENT/PLATFORM/LS454/FLOW_COUNT	

Notes

1. Use document header attribute @center_name
2. Use STUDY/RELATED_STUDIES/RELATED_STUDY
3. n/a
4. Use TRANSCRIPTOMIC or METAGENOMIC or METATRANSCRIPTOMIC
5. Use AB 5500 Genetic Analyzer or AB 5500xl Genetic Analyzer
6. Use PIPELINE
7. Use PLATFORM/*/INSTRUMENT_MODEL
8. Use DATA_BLOCK/FILES/FILE/filetype, DATA_BLOCK/FILES/FILE/checksum

Table continues on next page...

Table continued from previous page.

Field	Notes
/EXPERIMENT/PLATFORM/LS454/FLOW_SEQUENCE	
/EXPERIMENT/PLATFORM/LS454/KEY_SEQUENCE	
/EXPERIMENT/PROCESSING/BASE_CALLS	6
/EXPERIMENT/PROCESSING/BASE_CALLS/BASE_CALLER	
/EXPERIMENT/PROCESSING/BASE_CALLS/SEQUENCE_SPACE	
/EXPERIMENT/PROCESSING/QUALITY_SCORES	6
/EXPERIMENT/PROCESSING/QUALITY_SCORES/@qtype[other]	
/EXPERIMENT/PROCESSING/QUALITY_SCORES/@qtype[phred]	
/EXPERIMENT/PROCESSING/QUALITY_SCORES/MULTIPLIER	
/EXPERIMENT/PROCESSING/QUALITY_SCORES/NUMBER_OF_LEVELS	

Notes
1. Use document header attribute @center_name
2. Use STUDY/RELATED_STUDIES/RELATED_STUDY
3. n/a
4. Use TRANSCRIPTOMIC or METAGENOMIC or METATRANSCRIPTOMIC
5. Use AB 5500 Genetic Analyzer or AB 5500xl Genetic Analyzer
6. Use PIPELINE
7. Use PLATFORM/*/INSTRUMENT_MODEL
8. Use DATA_BLOCK/FILES/FILE/filetype, DATA_BLOCK/FILES/FILE/checksum

Table continues on next page...

Table continued from previous page.

Field	Notes
/EXPERIMENT/PROCESSING/QUALITY_SCORES/QUALITY_SCORER	
/RUN/@instrument_model	7
/RUN/@run_file	
/RUN/@total_data_blocks	
/RUN/DATA_BLOCK/@format_code	
/RUN/DATA_BLOCK/@number_channels	
/RUN/DATA_BLOCK/@total_reads	
/RUN/DATA_BLOCK/@total_spots	
/RUN/PLATFORM/ABI_SOLID/COLOR_MATRIX	
/RUN/PLATFORM/ABI_SOLID/COLOR_MATRIX_CODE	

Notes

1. Use document header attribute @center_name
2. Use STUDY/RELATED_STUDIES/RELATED_STUDY
3. n/a
4. Use TRANSCRIPTOMIC or METAGENOMIC or METATRANSCRIPTOMIC
5. Use AB 5500 Genetic Analyzer or AB 5500xl Genetic Analyzer
6. Use PIPELINE
7. Use PLATFORM/*/INSTRUMENT_MODEL
8. Use DATA_BLOCK/FILES/FILE/filetype, DATA_BLOCK/FILES/FILE/checksum

Table continues on next page...

Table continued from previous page.

Field	Notes
/RUN/PLATFORM/ABI_SOLID/CYCLE_COUNT	
/RUN/PLATFORM/ABI_SOLID/INSTRUMENT_MODEL[AB SOLiD 5500]	
/RUN/PLATFORM/ABI_SOLID/INSTRUMENT_MODEL[AB SOLiD 5500xl]	
/RUN/PLATFORM/ABI_SOLID/SEQUENCE_LENGTH	
/RUN/PLATFORM/HELICOS/FLOW_COUNT	
/RUN/PLATFORM/HELICOS/FLOW_SEQUENCE	
/RUN/PLATFORM/ILLUMINA/CYCLE_COUNT	
/RUN/PLATFORM/ILLUMINA/CYCLE_SEQUENCE	
/RUN/PLATFORM/ILLUMINA/SEQUENCE_LENGTH	
/RUN/PLATFORM/LS454/FLOW_COUNT	

Notes

1. Use document header attribute @center_name
2. Use STUDY/RELATED_STUDIES/RELATED_STUDY
3. n/a
4. Use TRANSCRIPTOMIC or METAGENOMIC or METATRANSCRIPTOMIC
5. Use AB 5500 Genetic Analyzer or AB 5500xl Genetic Analyzer
6. Use PIPELINE
7. Use PLATFORM/*/INSTRUMENT_MODEL
8. Use DATA_BLOCK/FILES/FILE/filetype, DATA_BLOCK/FILES/FILE/checksum

Table continues on next page...

Table continued from previous page.

Field	Notes
/RUN/PLATFORM/LS454/FLOW_SEQUENCE	
/RUN/PLATFORM/LS454/KEY_SEQUENCE	
/RUN/SPOT_DESCRIPTOR/SPOT_DECODE_METHOD	
/RUN/SPOT_DESCRIPTOR/SPOT_DECODE_SPEC/NUMBER_OF_READS_PER_SPOT	
/SUBMISSION/ACTIONS/ACTION/HOLD/@HoldForPeriod	
/SUBMISSION/FILES	8

Notes
1. Use document header attribute @center_name
2. Use STUDY/RELATED_STUDIES/RELATED_STUDY
3. n/a
4. Use TRANSCRIPTOMIC or METAGENOMIC or METATRANSCRIPTOMIC
5. Use AB 5500 Genetic Analyzer or AB 5500xl Genetic Analyzer
6. Use PIPELINE
7. Use PLATFORM/*/INSTRUMENT_MODEL
8. Use DATA_BLOCK/FILES/FILE/filetype, DATA_BLOCK/FILES/FILE/checksum

Future Planned Revisions

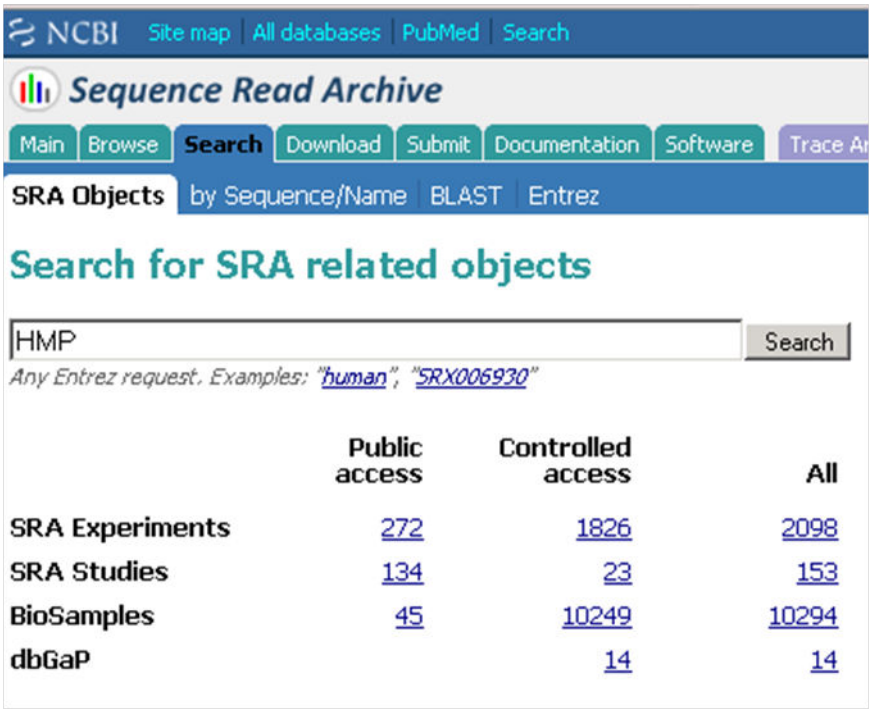
The next revision, SRA 1.5, will be contracting revision (one that potentially invalidates current documents). The main changes will be to remove deprecated fields. This will involve migration of data in anticipation of future schema changes. The SRA 1.5 schema release will follow soon after SRA 1.4 is deployed.

SRA Object Search function

Created: November 18, 2010.

Status	Active
Active Date	2010-11-18
Inactive Date	
Scope	NCBI SRA

It is now possible to submit to Entrez SRA a search time and have it return a table of hits organized by SRA object type. Here is an example with the general text search term "HMP":



This query returned hits in both the open (public access) and protected (controlled access) SRAs. Click through any of these hit links in order to access Entrez reports for each type of object.

Submitting PacBio Genome Modification Data

Created: July 17, 2013; Updated: December 12, 2013.

Status	Active
Active Date	2013-12-20
Inactive Date	
Scope	SRA, PathogenDetection

Overview

The SMRT(T) sequencing system from Pacific Biosciences is capable of measuring time points of polymerase incorporations[1]. Aberration in the incorporation rates (kinetics) can indicate a genome modification event such as methylation. Methylation sites and motifs are of interest to whole genome analysis and can be used to define the methylome of an organism. The output of the instrument can be analyzed and plotted as epigenetic markers on the finished genome sequence. The data can be viewed as a GFF track in whole genome browsers such as GBench.

As instances of secondary analysis of primary sequencing data, these datasets can be deposited into the SRA Analysis Archive and receive an accession. Unlike SRA data, SRA Analysis data are presented as they were deposited and are not checked except for verification of their corresponding file name, checksum, and file type. This application note describes the submission protocol for whole genome modification data into the SRA Analysis Archive.

Related Documents

Details about the general SRA Submission and file transfer protocols can be found here: http://www.ncbi.nlm.nih.gov/books/NBK49285/

GEO Archiving

Some bacterial epigenetic datasets have been deposited into the Gene Expression Omnibus, notably

http://www.ncbi.nlm.nih.gov/geo/query/acc.cgi?acc=GSE45178

and

http://www.ncbi.nlm.nih.gov/geo/query/acc.cgi?acc=GSE40133

Since then the archive policy has changed: as the kinetics data necessary for whole genome modification analysis is to be found in the raw sequencing data deposited in the SRA, it was considered appropriate to put modification datasets in the SRA Analysis Archive.

SRA Archiving

An example of a modification dataset deposited alongside a finished genome can be found at: http://www.ncbi.nlm.nih.gov/bioproject/?term=203445 (BioProject)

http://www.ncbi.nlm.nih.gov/biosample/?term=SAMN02179883 (BioSample)

http://www.ncbi.nlm.nih.gov/Traces/sra/sra.cgi?analysis=SRZ049709

Pre-requisites

An SRA submission account is necessary to proceed, see http://trace.ncbi.nlm.nih.gov/Traces/sra_sub/sub.cgi?view=submissions

The following information is required before preparing a genome modification submission:

1. Identity of the BioProject to which this analysis belongs
2. Identity of the BioSample target

Targets

This information identifies the "targets" of analysis, or the data components subjected to analysis.

1. Identity of the whole genome or draft genome. For complete genome sequences a list of accessions (with versions) of finished replicons (chromosomes and plasmids) is needed. For draft genomes the WGS Master record accession is sufficient.
2. A list of the PacBio SRA experiment accessions (no version). These will be returned by submission of the raw sequencing data to SRA. If the SRA deposits were delivered in h5.bas file format (the usual case), then the SRA archive will contain the kinetic data used for modification analysis.

Files

The PacBio SMRT analysis pipeline and motif finder software products produce the necessary output files. These three files per analysis are suggested (additional files can be added to the dataset):

modifications.csv – Comma separated file listing the genome modification locations

motifs.gff – GFF file of the modification sites, suitable for loading as a track

motifs_summary.csv – Comma separated file containing the nucleotide motifs

These files are archived without modification so the user would download the same files.

MD5 checksums for each file (using md5sum or similar program) should be computed for each file. The files do not need to be compressed.

Submission Protocol

1 Prepare the analysis xml file. Add analysis name, title, description, pipeline version, dependency references, and file information. One analysis xml file per genome is recommended. Using the xml features you can specify additional tag-value attributes or links. You can validate your file as follows:

xmllint –schema http://www.ncbi.nlm.nih.gov/viewvc/v1/trunk/sra/doc/SRA_1-5/SRA.analysis.xsd?view=co my.analysis.xml

2. Prepare the submission xml file. Only one submission xml file is required per submission batch (you can have multiple analysis xml files in a submission). A release date can be specified here. To validate this file, do

xmllint –S http://www.ncbi.nlm.nih.gov/viewvc/v1/trunk/sra/doc/SRA_1-5/SRA.submission.xsd?view=co my.submission.xml

3. Create a tar file from these files.

tar cvf my.submission.xml.tar my.submission.xml my.analysis.0.xml my.analyis.1.xml …

4. Transmit the files to the appropriate ftp or aspera destination of your submission account.

5. Once loaded, a single SRA analysis accessions (SRZxxxxxx) will be assigned to each analysis dataset. This accession can be used in a manuscript. There is no versioning of these objects.

6. You can specify a release date in the xml submission or you can write to NCBI to have the data released (or a new release date set).

Finding and Downloading Datasets

Under the current SRA system, the published analysis datasets are listed in a browsable form here: http://www.ncbi.nlm.nih.gov/Traces/sra/sra.cgi?view=analyses

SRA Analysis accessions can be downloaded from this location (must have the accession):

http://www.ncbi.nlm.nih.gov/Traces/sra/sra.cgi?view=download_analyses

Currently there is no linking with other objects in the NCBI Entrez system, so you will not see the analysis listed as one of the linked objects from BioProject, BioSample, or SRA. However, the browsable SRA analysis object does show these links.

Example

Accessions have been removed from these documents as the SRA will assign these on successful submission. Contact information has been removed.

One analysis xml document

```
<ANALYSIS xmlns="" alias="SAMN02179882_analysis" center_name="UCD-100K">
    <IDENTIFIERS>
        <SUBMITTER_ID namespace="UCD-100K">SAMN02179882_analysis</SUBMITTER_ID>
    </IDENTIFIERS>
    <TITLE>Base modification detection and motif analysis by SMRT Analysis</TITLE>
    <STUDY_REF accession="SRP022914">
        <IDENTIFIERS>
            <EXTERNAL_ID namespace="BioProject">PRJNA203445</EXTERNAL_ID>
        </IDENTIFIERS>
    </STUDY_REF>
    <DESCRIPTION>Strand-specific single-base resolution base modification and
                 pattern analysis
    </DESCRIPTION>
    <ANALYSIS_TYPE>
        <SEQUENCE_ANNOTATION>
            <PROCESSING>
                <PIPELINE>
                    <PIPE_SECTION>
                        <STEP_INDEX>1</STEP_INDEX>
                        <PREV_STEP_INDEX>NULL</PREV_STEP_INDEX>
                        <PROGRAM>SMRT_Analysis</PROGRAM>
                        <VERSION>1.4</VERSION>
                    </PIPE_SECTION>
                </PIPELINE>
            </PROCESSING>
        </SEQUENCE_ANNOTATION>
    </ANALYSIS_TYPE>
    <TARGETS>
        <TARGET sra_object_type="SAMPLE" accession="SAMN02179882"></TARGET>
        <TARGET sra_object_type="EXPERIMENT" accession="SRX317124"></TARGET>
        <IDENTIFIERS>
            <EXTERNAL_ID namespace="genbank">CP006004.1</EXTERNAL_ID>
            <EXTERNAL_ID namespace="genbank">CP006005.1</EXTERNAL_ID>
        </IDENTIFIERS>
    </TARGETS>
    <DATA_BLOCK>
      <FILES>
        <FILE filename="SAMN02179882/modifications.csv" filetype="csv"
         checksum_method="MD5" checksum="d0706df6b30b5eff17b3a215899d9b3c">
        </FILE>
        <FILE filename="SAMN02179882/motifs.gff" filetype="GFF"
         checksum_method="MD5" checksum="82b6cf00da93aec42a15accec73fe747">
        </FILE>
        <FILE filename="SAMN02179882/motif_summary.csv" filetype="csv"
         checksum_method="MD5" checksum="bb1a28e30e41b31f059cafd09f8b618e">
        </FILE>
      </FILES>
     </DATA_BLOCK>
</ANALYSIS>
```

A submission xml document supporting multiple analyses

```
<SUBMISSION xmlns="" alias="UCD-100K 2013-07-08" center_name="UCD-100K" lab_name="" >
    <IDENTIFIERS>
        <SUBMITTER_ID namespace="UCD-100K">UCD-100K 2013-07-08</SUBMITTER_ID>
    </IDENTIFIERS>
    <CONTACTS>
        <CONTACT name=""></CONTACT>
    </CONTACTS>
    <ACTIONS>
        <ACTION>
            <ADD source="SAMN02179882_analysis.xml" schema="analysis"></ADD>
        </ACTION>
        <ACTION>
            <ADD source="SAMN02179883_analysis.xml" schema="analysis"></ADD>
        </ACTION>
        <ACTION>
            <ADD source="SAMN02179884_analysis.xml" schema="analysis"></ADD>
        </ACTION>
        <ACTION>
            <HOLD HoldUntilDate="2014-07-08-08:00"></HOLD>
        </ACTION>
    </ACTIONS>
</SUBMISSION>
```

References

1. Lluch-Senar M, Luong K, Lloréns-Rico V, Delgado J, Fang G, et al. 2013; Comprehensive Methylome Characterization of Mycoplasma genitalium and Mycoplasma pneumoniae at Single-Base Resolution. PLoS Genet. 9(1):e1003191doiAbstract: http://www.ncbi.nlm.nih.gov/pubmed/?term=23300489 PubMed PMID: 23300489.

Aspera Keys

Created: January 29, 2014.

Status	Active
Active Date	2014-01-29
Inactive Date	
Scope	INSDC SRA

Notice

Reference herein to any specific commercial products, process, or service by trade name, trademark, manufacturer, or otherwise, does not necessarily constitute or imply its endorsement, recommendation, or favoring by the United States Government. The views and opinions of authors expressed herein do not necessarily state or reflect those of the United States Government, and shall not be used for advertising or product endorsement purposes.

Overview

The ascp program requires a private/public key pair for transfers. This guide describes how users can generate or convert their keys for use with ascp.

Scope

This document is intended for users transferring large data files from NCBI. It applies to the Sequence Read Archive (SRA), dbGaP, and other archives where Aspera download is enabled.

Generating New Keys

ascp Version

To find the version of ascp installed, run the program with "–A" or "--version"

```
../ascp -A
```

Versions 2.6 and newer of ascp

Linux/Unix and OS X users can use the ssh-keygen utility

Using ssh-keygen

```
ssh-keygen -f ./private.openssh
```

This will store a private key in the current working directory with the name 'private.openssh' as well as a public key with the name 'private.openssh.pub'

Using puttygen

Download the PuTTY software for UNIX

http://www.chiark.greenend.org.uk/~sgtatham/putty/download.html

For questions concerning PuTTY installation on UNIX, please see the README file provided in the downloaded source.

To generate a OpenSSH private key:

```
../puttygen -O private-openssh -t rsa -b 1024 -o private.openssh
```

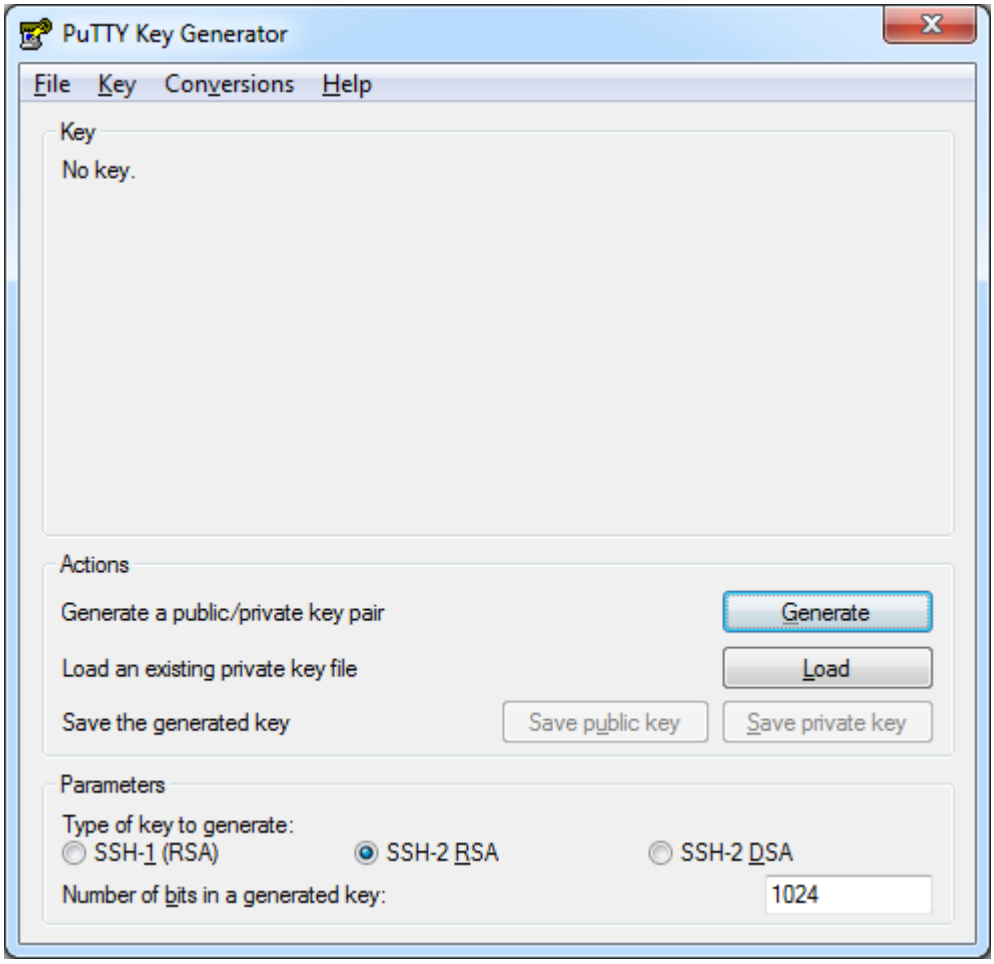
To generate an open-ssh public key from the private key:

```
../puttygen private.openssh -O public-openssh -o publicssh.pub
```

Microsoft Windows Users:

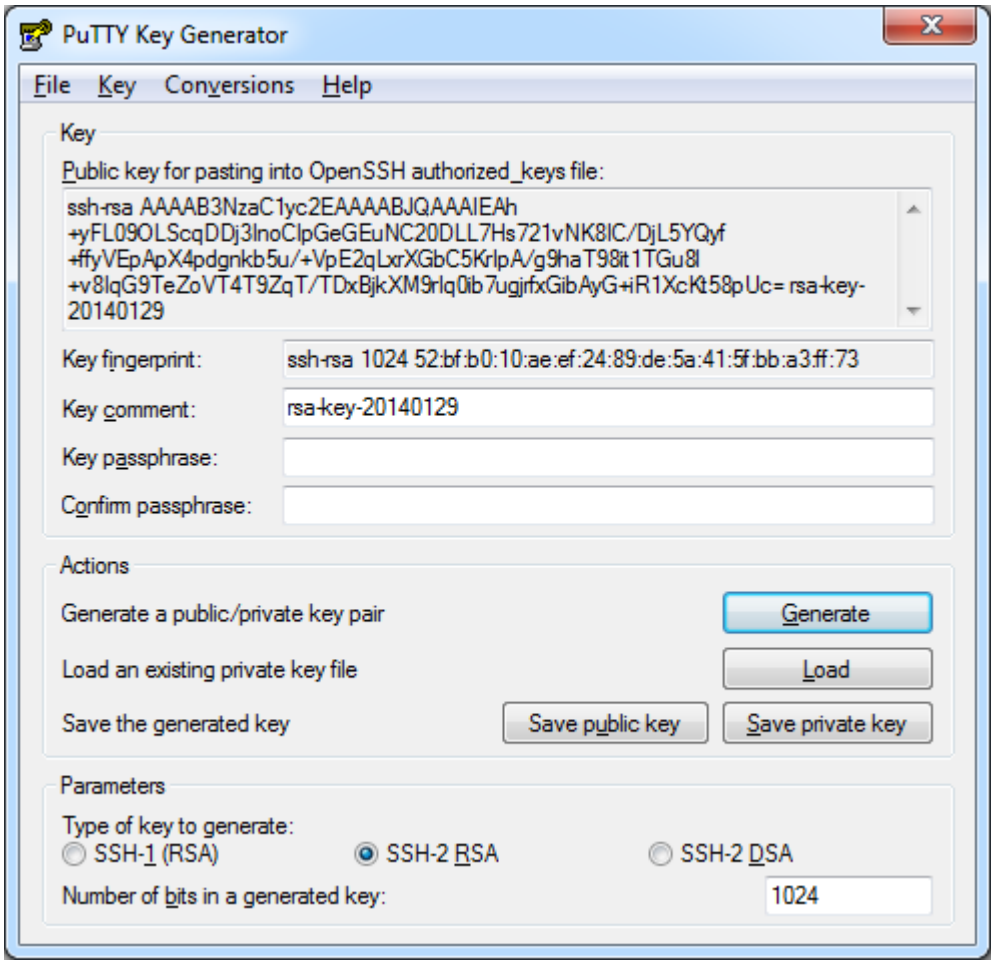
Download puttygen: http://the.earth.li/~sgtatham/putty/latest/x86/puttygen.exe

Run *puttygen.exe* to create an ssh key:



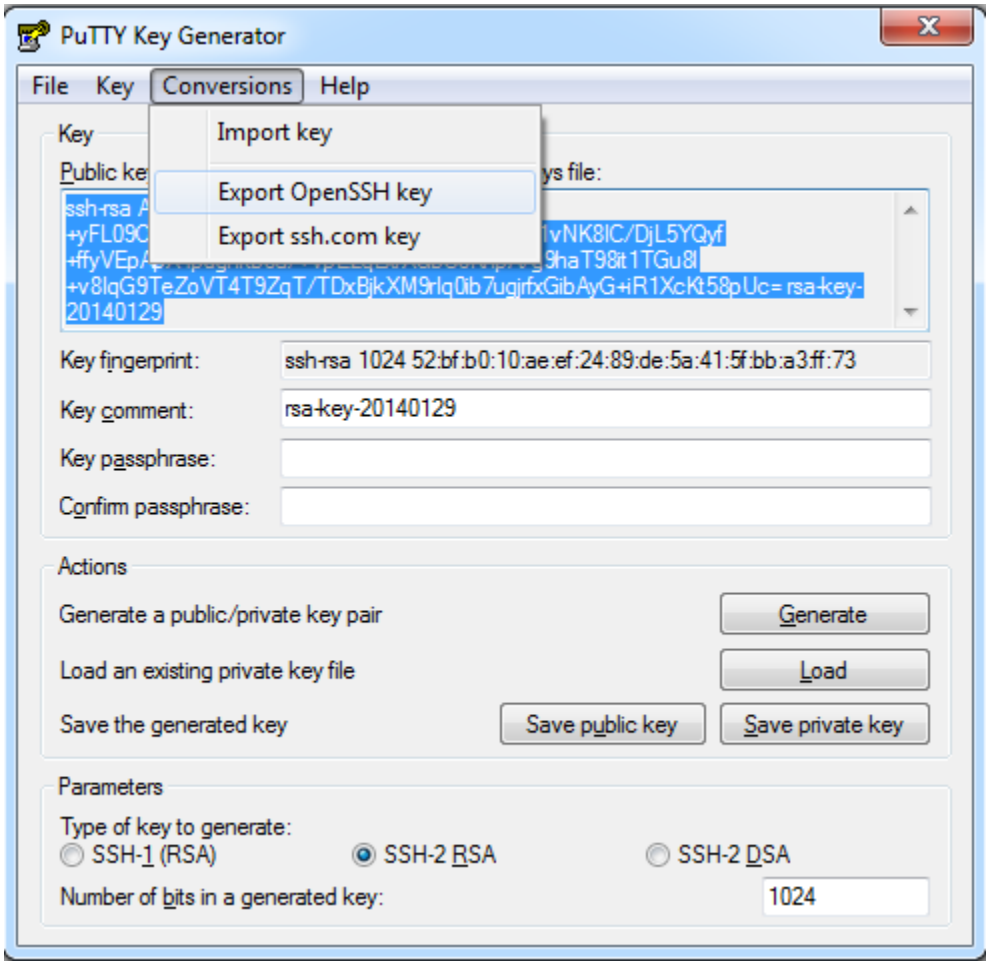
Make sure that SSH-2 RSA Parameter option is selected, and that the “Number of bits in a generated key” be set to 1024. Then press “Generate” (moving the mouse to generate a key).

Generating a key will result in something like this:



Copy the text from the “Public Key for pasting into OpenSSH authorized_keys file” text box. The OpenSSH public key must look like the following example. Other formats can’t be used as the public key.

```
ssh-rsa AAAAB3NzaC1yc2EAAAABJQAAAIEAoQNz1WIxVOvdRL9fx
 … jVp9nc= rsa-key-20090113
```



Click “Export OpenSSH Key” in the “Coversions” menu to retain the private key in OpenSSH format. NOTE - leave “Key passphrase” and “Confirm passphrase” empty (otherwise, you will be prompted to enter the passphrase whenever you perform an Aspera transaction).

Keys for ascp versions prior to 2.6

Linux/UNIX Users

Puttygen - Download the PuTTY software for UNIX

http://www.chiark.greenend.org.uk/~sgtatham/putty/download.html

For questions concerning PuTTY installation on UNIX, please see the README file provided in the downloaded source.

To generate a putty private key:

```
../puttygen -O private -t rsa -b 1024 -o puttyprivate.ppk
```

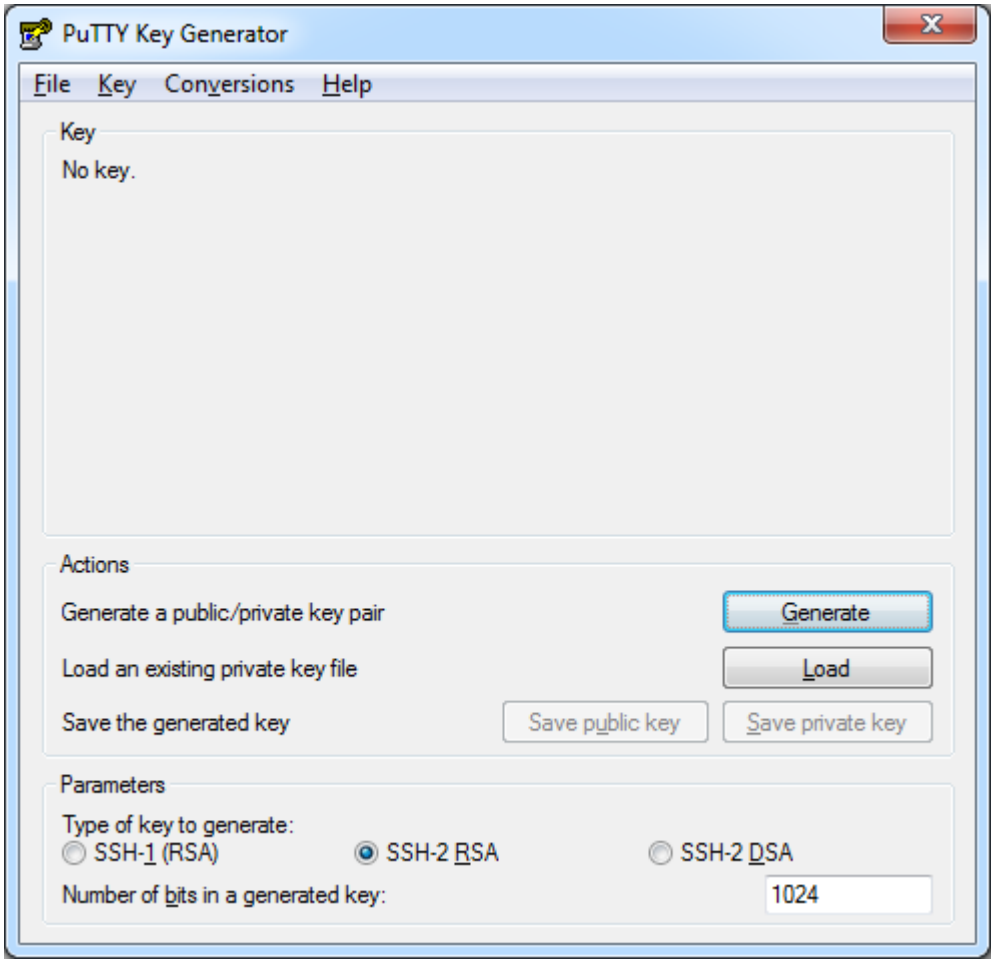
To generate an open-ssh public key from the private key:

```
../puttygen puttyprivate.ppk -O public-openssh -o publicssh.pub
```

Microsoft Windows Users:

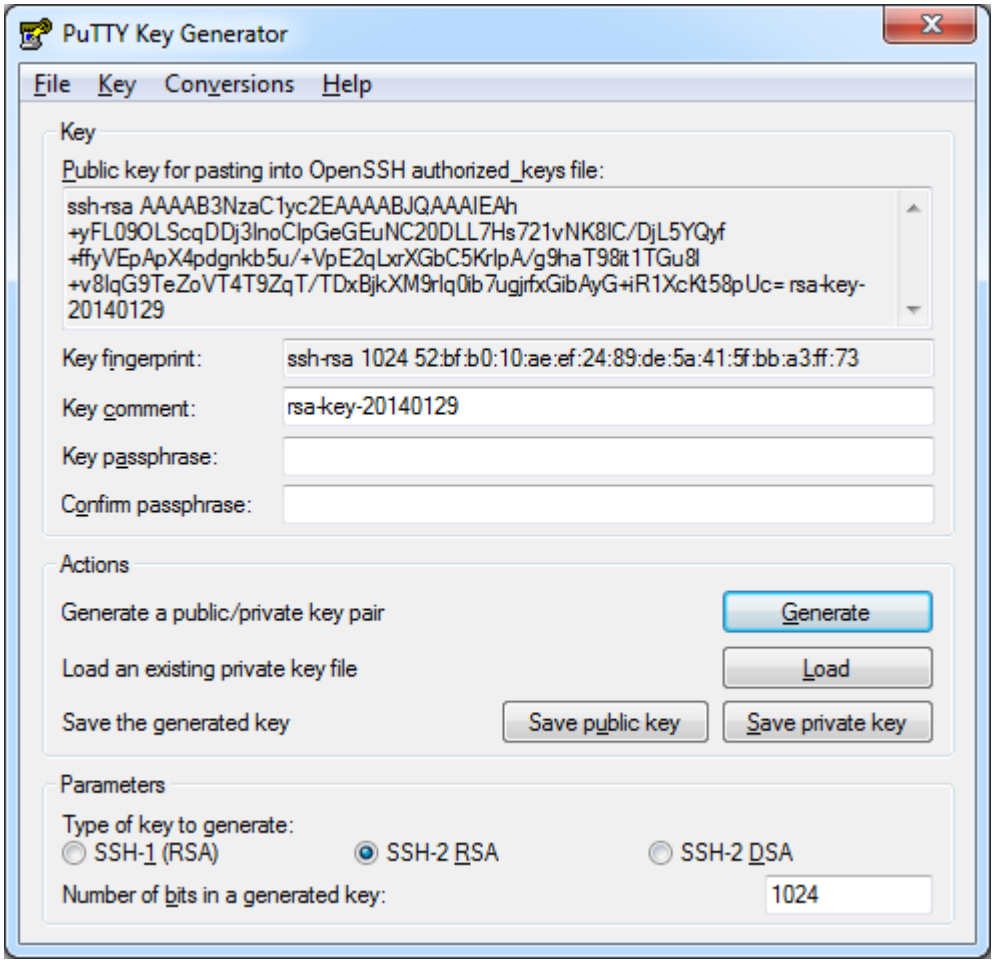
Download puttygen: http://the.earth.li/~sgtatham/putty/latest/x86/puttygen.exe

Run *puttygen.exe* to create an ssh key:



Make sure that SSH-2 RSA Parameter option is selected, and that the "Number of bits in a generated key" be set to 1024. Then press "Generate" (moving the mouse to generate a key).

Generating a key will result in something like this:



Click “Save Private Key” to retain the private key. NOTE - leave “Key passphrase” and “Confirm passphrase” empty (otherwise, you will be prompted to enter the passphrase whenever you do an Aspera transaction).

Copy the text from the “Public Key for pasting into OpenSSH authorized_keys file” text box. The OpenSSH public key must look like the following example. Other formats can’t be used as the public key.

```
ssh-rsa AAAAB3NzaC1yc2EAAAABJQAAAIEAoQNz1WIxVOvdRL9fx
 … jVp9nc= rsa-key-20090113
```

Converting Key Formats

PuTTY format keys (.ppk) will need to be converted to OpenSSH for use with the latest version of ascp.

Linux/UNIX Users

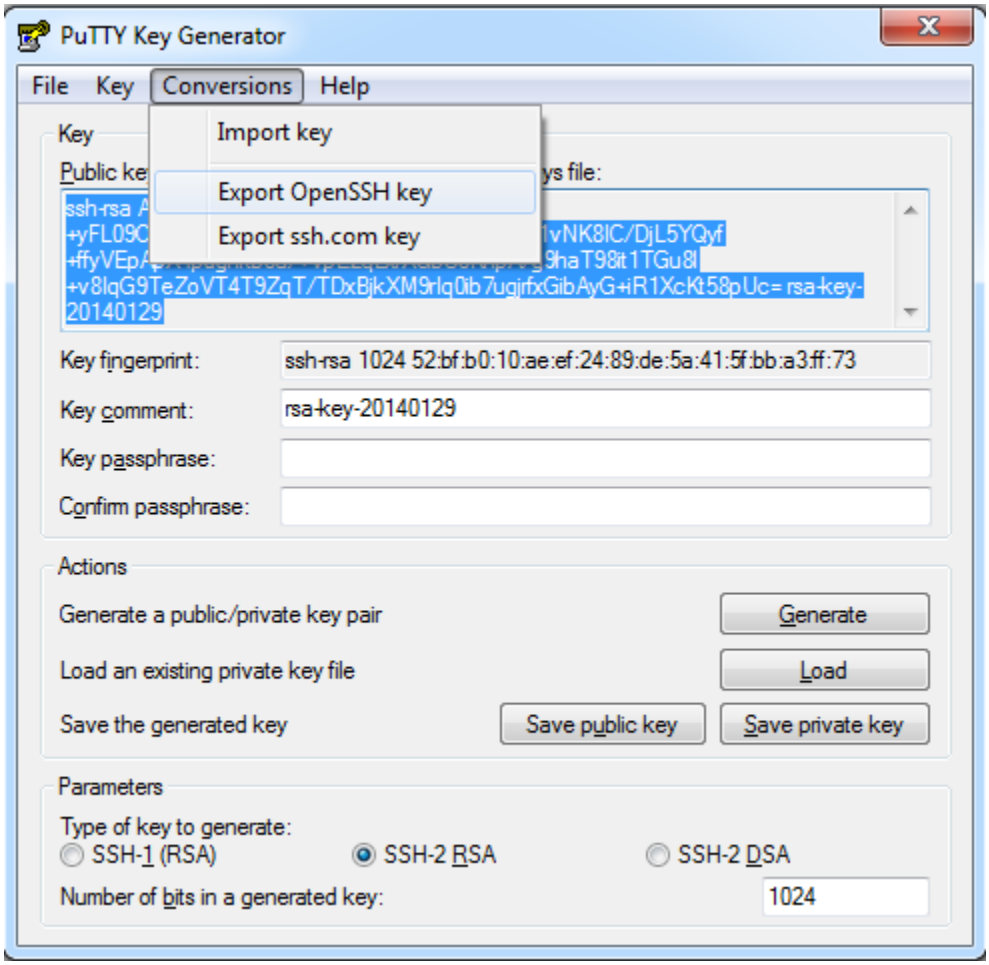
To convert a PuTTY format private key to a OpenSSH format private key with puttygen:

```
puttygen <original_key.ppk> -O private-openssh -o <new_key.openssh>
```

Microsoft Windows Users:

Run *puttygen.exe* to convert a PuTTY format key:

In the “File” menu select “Load private key”. Select the PuTTY key the needs to be converted from the file browswer.



Select “Export OpenSSH key” from the “Conversions” menu.

Deprecated Notes

SRA XML Specification Version 1.0 (Deprecated)

Created: April 9, 2009; Updated: March 10, 2010.

Status	Inactive
Active Date	2009-04-09
Inactive Date	2009-11-26
Scope	INSDC SRA

1 Overview

This document summarizes the proposed changes for Release 1.0 of the Short Read Archive (SRA) schemas governing XML metadata. Release 1.0 is a slight change over Release 0.8, which was in introduced in June 2008. The goal of this release is to patch the XML schema with needed changes while not invalidating current XML implementations.

A second release with deeper changes that will require migration of existing data and possible changes to client XML generation software is also planned. This release will include a period of public comment and ample time for adjustment and migration.

1.1 Related Documents

1.2 Revision History

16 Feb 2009 -27 Mar 2009 – Drafts A,B,C worked out by NCBI, EBI, DDBJ

2 Changes

2.1 Remove expected counts from Experiment and Run

These fields have proved misleading when bound by the submitter. These fields remain as optional attributes in EXPERIMENT and RUN, but deprecation warnings will be issued for new documents that have these fields bound.

2.2 Add center_name attribute to Experiment and Sample

This allows the submitter to establish ownership of the record and allows the Archive to correctly ascribe ownership.

2.3 Add new instrument models

New instrument values have been added to Experiment and Run:

- 454 XLR Titanium
- Illumina Genome Analyzer II
- AB SOLiD System 2.0
- AB SOLiD System 3.0

2.4 New Study type values

- RNASeq
- Other

2.5 Add new library selection values

New values for LIBRARY_SELECTION have been added to Experiment.

- Hybrid Selection
- DNAse

2.6 Add new library strategy values

New values for LIBRARY_STRATEGY have been added to Experiment

- Bisulfite-Seq
- DNAse-Hypersensitivity

2.7 Add new library selection values

- Reduced Representation

2.8 Submission structure made more flexible

- Submissions may not need FILES section and no longer have to have one even if there are no files.
- Outgoing submission XML may be stripped of CONTACTS, ACTIONS, FILES data because they are not relevant to the user of the Archive.

2.9 Drop run_file from RUN

The RUN.run_file attribute has never been used effectively. It is not needed and the data in it can be dropped from the archive.

2.10 Add fields to Sample Name

The following fields have been added to SAMPLE.SAMPLE_NAME in order to create additional ways to unambiguously name a sample:

- SCIENTIFIC_NAME

2.11 Add Title to Sample

The Sample object now should have a title to make it easier to search. For example: “E. coli K-12 MG1665 genomic sample.” Titles need not be unique.

2.12 Move Sample Members Table to Experiment

This change means that sample pools will be specified at the level of experiment. Multiplexed sample experiments where each sample is distinguishable by a bar code are listed by sample and bar code. Pooled samples are listed by sample only. Sample can be identified by alias or accession. SAMPLE.MEMBERS has been removed from the schema. The optional DEFAULT_MEMBER identifies the sample to use when none of the specified bar codes matches the reads, usually due to sequencing error.

2.13 WITHDRAW to become SUPPRESS

This submission action is actually the GenBank SUPPRESS action.

2.14 Add a new action called PROTECT

TO support submission of short read data into protected databases like dbGaP.

2.15 Remove HoldUntilPublication

SUBMISSION.ACTIONS.ACTION.HOLD/HoldUntilPublication is to be removed.

This is too hard to implement.

2.16 Remove CURATE

SUBMISSION.ACTIONS.ACTION.CURATE to be removed, not used.

2.17 Remove SUBMISSION.handle

Not used.

2.18 Remove CLOSE

Not used.

2.19 Remove requestor, request_date

SUBMISSION.ACTIONS.requestor, SUBMISSION.ACTIONS.request_date,

Not needed as the submission system will record these.

2.20 Remove handle

SUBMISSION.handle is not used.

2.21 Add submission title

SUBMISSION.TITLE would be used in some cases by submitters who are referencing SRA/ERA accession in their publication.

2.22 Add broker_name

Identity of broker authority responsible for the submission. This is used to

determine ownership and editorial authority, hold and release control, and

future editing capability.

2.23 Remove EXCEPTIONS block

SUBMISSION.EXCEPTIONS to be replaced by a dedicated document (SRA.Receipt.xsd) for this purpose.

2.24 Rename submission_id to alias

Insert new attribute called SUBMISSION.alias to be consistent with other objects. Deprecate SUBMISSION.submission_id and remove later.

2.25 Add to all documents additional link type: XREF_LINK

This is another way to specify external links using the database and accession. This method relies on the archive to construct a proper link, but is less sensitive to changes in the way links are served in the external database.

2.26 Master Study for study

This is to encode the situation where the information of the study is derived from another record, and the values for the study record are derived from the master. Possible values are:

None – (if there is no master study)

INSDC – Genome Project id

GEO – Gene Expression Omnimus at NCBI

ArrayExpress – Array Express resource at EBI

dbGaP – Genotype and Phenotype resource at NCBI

Bioinvestigation Index – BioInvestigation resource at EBI

2.27 RUN.DATA_BLOCK

This section is made optional in order to redact the submission information from the Run record, in the case where the Run record is displayed to the user of the archive (in Entrez XML, or in the ERA). However, this field continues to be required to process a submission.

2.28 RUN.DATA_BLOCK.ADDRESS

This is a new section to describe the address of the run data provided with the submission, when (a) the portion to be loaded is a subset of the data provided, or (b) the data are provided in uncontained format (fastq or native) and the address of the reads needs to be communicated to the loader. The default behavior when this spec is not provided is to load all data encountered in the submitted run data file.

The previous tags that tracked this information, DATA_BLOCK name, sector, region, have been deprecated. The new scheme is specific to the platform but also can be queried in a database.

2.29 New ANALYSIS object

The ANALYSIS object will contain unstructured submissions of secondary analysis of sequence read objects, including assemblies, alignments, and clean sequence datasets appropriate for submission to dbEST.

2.30 RUN.SPOT_DESCRIPTOR.SPOT_DECODE_SPEC

This tag was removed because it was underspecified, and because the spot layout needs to be fully specified in every case.

2.31 SPOT_DESCRIPTOR moved to Run

The important section describing the layout of the spot sequence has been moved to Run for a variety of reasons:

- In some cases the layout is only known at run time
- In some cases the layout must be changed in response to QC
- In some cases the layout may be specific to a group of reads that define the run

Submissions containing EXPERIMENT.SPOT_DESCRIPTOR will continue to be processed for a time, but new submissions should switch to the Run based location of this block.

2.32 New section RUN.PLATFORM

The Run object now has a PLATFORM block, which includes information from the old EXPERIMENT.PLATFORM as well as new settings such as RAW_SEQUENCE_LENGTH for Illumina and SOLiD in order to indicate the number of bases for these fixed length platforms. New submitters should switch platform details from Experiment to Run. EXPERIMENT.PLATFORM will remain, but in a cut down version, as the highest level indicator of what sequencing platform the experiment is targeting.

The reasons for this change include:

- The instrument model needs to be defined in one place only

- Parameters such as number of flows or bases are run specific and only known at run time.
- Parameters such as flow sequence and color matrix may be static, but are closely related to the interpretation of the run data and therefore should be located there.

This change comes at the cost of redundancy with respect to the runs in the experiment. However, the experiment can be presented as the union of the platform settings, which should be consistent.

For 454, the following fields apply: KEY_SEQUENCE, FLOW_SEQUENCE, FLOW_COUNT.

For Illumina, the following fields apply: SEQUENCE_LENGTH, intended to be the number of bases in the raw sequence (including both mate pairs and any technical reads).

2.33 New section RUN.PROCESSING

The Run object now has a PROCESSING block, which includes information about what pipeline was used to process the run data. As this is not always known until run time, the data should be bound at the level of run. Precise details of the processing pipeline can be embedded in the run container file (SFF or SRF). Some aspects of the former EXPERIMENT.PROCESSING have been refactored into the RUN.PLATFORM block, such as RUN.SEQUENCE_SPACE and RUN.PLATFORM.QUALITY_MODEL.

The design for specifying the run processing pipeline is flexible, and allows for multiple pipe sections arrayed in a workflow. The workflow commands allow for any procedural workflow to be specified. The optional COMMAND tag can be used to specify a complete workflow (with branches and loops), otherwise the process is executed in the order specified by the PROCESS_INDEX tag.

Normally, a single entry will be sufficient, as the processing pipeline is the manufacturer's standard protocol.

3 Summary of Deprecated Fields

EXPERIMENT.expected_number_spots

EXPERIMENT.expected_number_reads

EXPERIMENT.expected_number_bases

RUN.DATA_BLOCKS.DATA_BLOCK.total_spots

RUN.DATA_BLOCKS.DATA_BLOCK.total_reads

RUN.instrument_model[Solexa 1G Genome Analyzer]

RUN.instrument_model[Solexa 1G Genome Analyzer]

EXPERIMENT.PLATFORM.LS454.instrument_model[GS 20]

EXPERIMENT.PLATFORM.LS454.instrument_model[GS FLX]

RUN.run_file

STUDY.PROJECT

RUN.total_spots

RUN.total_reads

RUN.number_channels

RUN.format_code

RUN.total_data_blocks

RUN.run_file

SAMPLE.members

DATA_BLOCK.name

DATA_BLOCK.sector

DATA_BLOCK.region

4 Summary of Required Fields

EXPERIMENT.TITLE – This field is optional in the schema but will eventually be required. As a business rule, new submissions must have this field set to a value.

SAMPLE.TITLE – This field is optional in the schema but will eventually be required. As a business rule, new submissions must have this field set to a value.

SUBMISSION.CONTACTS – This is optional in the schema and may not be reproduced by the Archive because of the private nature of the content. However, on submission the Archive will require a CONTACTS section.

SUBMISSION.ACTIONS – This is optional in the schema and may not be reproduced by the Archive because of irrelevance. However, on submission the Archive will require a ACTIONS section.

STUDY.MASTER_STUDY – This is optional in the schema but required in order to identify the source of the study record information, or None if there is no master record.

RUN.DATA_BLOCK – This is optional in the schema but required for a submission to be processed.

RUN.DATA_BLOCK.ADDRESS – This spec is required when the data are submitted in an uncontained format (such as fastq or a native format).

SRA XML Specification Version 1.1 (Deprecated)

Created: November 17, 2009; Updated: March 10, 2010.

Status	Inactive
Active Date	2009-11-17
Inactive Date	2010-10-31
Scope	INSDC SRA

1 Overview

This document summarizes the proposed changes for Release 1.1 of the Sequence Read Archive (SRA) schemas governing XML metadata. Release 1.1 is an expansion of Release 1.0, which was in introduced in April 2009. The goal of this release is to patch the XML schema with needed changes while not invalidating current XML implementations.

Major new features in this release are:

- Additional library choices needed for epigenomics
- Additional platform and instrument choices
- Bar code support for pooled and multiplexed samples
- Tightening of specification of run files to allow for improvement of loader programs
- Identification of disused features and options

A second release (SRA 1.2) will be organized with deeper changes that will require migration of existing data and possible changes to client XML generation software is also planned. This release will include a period of public comment and ample time for adjustment and migration.

The third release (SRA 1.3) should then introduce the next round of feature changes.

1.1 Related Documents

The SRA schema can be obtained from this site: http://www.ncbi.nlm.nih.gov/Traces/sra/sra.cgi?cmd=show&f=schema&m=doc&s=schema

1.2 Revision History

16 Feb 2009 – 04 Dec 2010 – Drafts A-K developed by INSDC partners

1.3 Release Plan

After a period of review existing documents can assume the new schema without change. Following deployment, the SRA capabilities will be built out to take full advantage of the new schema features. Finally, deprecated fields in existing documents will be migrated. The next revision will concentrate on removing deprecated features that have already been migrated from existing documents.

2 Explanation of Changes

2.1 Changes to All Documents

2.1.1 Remove counts from all documents

These fields have proved misleading when bound by the submitter. These fields remain as optional attributes in EXPERIMENT and RUN, but deprecation warnings will be issued for new documents that have these fields bound.

2.1.2 Add to all documents additional link type: XREF_LINK

This is another way to specify external links using the database and accession. This method relies on the archive to construct a proper link, but is less sensitive to changes in the way links are served in the external database.

2.1.3 Add to all documents additional link type: SRA_LINK

This is another way to specify local links visible within the scope of the Home Archive. The intent is that during submission and loading such links will be converted to XREF_LINK where possible.

2.1.4 Add ownership attributes to all documents

The following attributes have been added to all documents: center_name, broker_name.

These allow for establishing a namespace of the center for any document. In addition, refcenter_name has been added to allow for specification of the reference name space.

2.2 Add New Choices to Schema

2.2.1 Add new instrument models

New instrument values have been added to Experiment

- 454 XLR Titanium
- Illumina Genome Analyzer II
- AB SOLiD System 2.0
- AB SOLiD System 3.0

The use of instrument model in Run has been deprecated.

2.2.2 New Study type values

- RNASeq
- Other

2.2.3 Add new library selection values

New values for LIBRARY_SELECTION have been added to Experiment.

- Hybrid Selection
- DNAse

2.2.4 Add new library strategy values

New values for LIBRARY_STRATEGY have been added to Experiment

- Bisulfite-Seq
- DNAse-Hypersensitivity

In addition, BARCODE has been deprecated as it pertains to a pooling strategy and not a library strategy.

2.2.5 Add new library selection values

- Reduced Representation

2.3 Changes to Study

2.3.1 Add RELATED_STUDIES to SRA Study

The SRA study is usually a surrogate object that includes information from one or more studies in other repositories. The SRA_LINK or XREF_LINK mechanisms can be used to call out these dependencies. At some point in the future SRA Study will be migrated over to the new Project repository.

As PROJECT_ID is deprecated, use the RELATED_STUDIES mechanism instead.

2.4 Changes to Sample

2.4.1 Add fields to Sample Name

The following fields have been added to SAMPLE.SAMPLE_NAME in order to create additional ways to unambiguously name a sample:

- SCIENTIFIC_NAME

Scientific name of sample that distinguishes its taxonomy. Please use a name or synonym that is tracked in the INSDC Taxonomy database. Also, this field can be used to confirm the TAXON_ID setting.

- INDIVIDUAL_NAME - Individual name of the sample. This field can be used to identify the individual identity of a sample where appropriate (this is usually NOT appropriate for human subjects). Example: "Glennie" the platypus.

Documentation for all the sample name fields has been improved, giving better guidance to submitters:

2.4.2 Add Title to Sample

The Sample object now should have a title to make it easier to search. For example: “E. coli K-12 MG1665 genomic sample.” Titles need not be unique.

2.4.3 Move Sample Members Table to Experiment

This change means that sample pools will be specified at the level of experiment. Multiplexed sample experiments where each sample is distinguishable by a bar code are listed by sample and bar code. Pooled samples are listed by sample only. Sample can be identified by alias or accession. SAMPLE.MEMBERS has been removed from the schema. A SAMPLE_POOL_DESCRIPTOR has been defined as an option for EXPERIMENT.DESIGN instead of SAMPLE in order to support experiments conducted on sample pools (multiplexed or otherwise).

2.5 Changes to Submission

2.5.1 Submission structure made more flexible

- Submissions may not need FILES section and no longer have to have one even if there are no files.
- Outgoing submission XML may be stripped of CONTACTS, ACTIONS, FILES data because they are not relevant to the user of the Archive.

2.5.2 WITHDRAW to become SUPPRESS

This submission action is actually the GenBank SUPPRESS action.

2.5.3 Add a new action called PROTECT

TO support submission of short read data into protected databases like dbGaP.

2.5.4 Remove HoldUntilPublication

SUBMISSION.ACTIONS.ACTION.HOLD@HoldUntilPublication is to be removed because the feature is underspecified.

2.5.5 Remove CURATE

SUBMISSION.ACTIONS.ACTION.CURATE to be removed, not used.

2.5.6 Remove SUBMISSION.handle

This field is never used.

2.5.7 Remove requestor, request_date

The SUBMISSION.ACTIONS requestor and request_date tags have been deprecated because the submission system will record this info.

2.5.8 Add submission title

SUBMISSION.TITLE would be used in some cases by submitters who are referencing SRA/ERA accession in their publication.

2.5.9 Remove EXCEPTIONS block

SUBMISSION.EXCEPTIONS to be replaced by a dedicated document (SRA.Receipt.xsd) for this purpose.

2.5.10 Rename submission_id to alias

Insert new attribute called SUBMISSION.alias to be consistent with other objects. Deprecate SUBMISSION.submission_id and remove later.

2.5.11 Add links and attributes to Submission

Links and Attributes that are available to all documents are included with Submission. This is to allow for binding of information specific to the submission (but not the content or metadata) to the submission in a flexible way. This information is NOT intended to be used in indexing.

2.6 Changes to Run

The RUN.DATA_BLOCK is made optional in order to redact the submission information from the Run record, in the case where the Run record is displayed to the user of the archive (in Entrez XML, or in the ERA). However, this field continues to be required to process a submission.

The **RUN.run_file** attribute has never been used effectively. It is not needed and the data in it can be dropped from the archive.

2.6.1 Changes to RUN.DATA_BLOCK

Several changes have been made to the DATA_BLOCK itself:

The **DATA_BLOCK** contains loading instructions about the data. It is not needed once the data have been successfully loaded, and does not need to be included in mirrored or downloaded metadata dumps. Therefore, **DATA_BLOCK** has been made technically optional, although it is required on submission.

The new **DATA_BLOCK.serial** attribute will allow for loading of multiple DATA_BLOCKs by indicating the load order. This specification is needed in order for loaders to work with multiple DATA_BLOCK loads.

The new **DATA_BLOCK.FILE.filetype.sra** choice has been added in anticipation of direct submission of sra objects (native SRA archive files).

The new, more specific choices for 454 native file types has been added:

DATA_BLOCK.FILE.filetype.454_native

DATA_BLOCK.FILE.filetype.454_native_seq

DATA_BLOCK.FILE.filetype.454_native_qual

The **DATA_BLOCK.FILE.filetype.Helicos_native** choice has been added to support restricted grammars for Helicos text data.

New options for Illumina native filetypes have been added in order to provide more specificity: **Illumina_native_seq**

Illumina_native_prb

Illumina_native_int

Illumina_native_qseq

Illumina_native_fastq

Illumina_native_scarf

More specific choices are offered for SOLiD native file types

DATA_BLOCK.FILE.filetype.SOLiD_native

DATA_BLOCK.FILE.filetype.SOLiD_native _csfasta

DATA_BLOCK.FILE.filetype.SOLiD_native _qual

Finally, a new **DATA_BLOCK.FILE.filetype.tab** file type allows for the specification of per-spot auxiliary sequence data where this is need for loading (for example alternative read_seg settings).

The new **DATA_BLOCK.FILE.READ_LABEL** allows you to associate a given file to named tag(s) in the spot descriptor (for example F1.qseq vs R1.qseq).

The new **DATA_BLOCK.FILE.DATA_SERIES_LABEL** allows you to associate a given file to one or more named data series in the spot descriptor (for example F1.csfasta vs F1.qual). The choices were taken from the SRA Toolkit documentation:

INSDC:read

INSDC:read_filter

INSDC:quality

INSDC:intensity

INSDC:signal

INSDC:noise

INSDC:position

INSDC:clip_quality_left

INSDC:clip_quality_right

INSDC:readname

INSDC:read_seg

Together, **DATA_BLOCK.FILE.READ_LABEL and DATA_BLOCK.FILE.DATA_SERIES_LABEL** should be able to specify most loader configurations we have encountered. Note that these are modeled as elements so that a file can file multiple data series or match multiple read labels.

DATA_BLOCK.FILE.checksum and **DATA_BLOCK.FILE.checksum_method** can be used to specify the checksum of the final component that will be presented to the loader.

The **DATA_BLOCK.FILE.quality_scoring_system** allows the submitter to specify that the incoming data is in log-odds form. This will allow the loader to not have to prescan the file in order to guess which scoring system is being used.

The **DATA_BLOCK.FILE.qualty_encoding** tells whether the quality string is ASCII character, decimal, or hexadecimal encoded. The **DATA_BLOCK.FILE.ascii_offset** allows for the specification of the representation of the basis value (the zero) in the quality data series. Together these parameters can interpret any character based or decimal based quality representation.

The new **DATA_BLOCK.member_name** allows an individual data block among several to be associated with a member of the sample pool. This is being introduced in anticipation of the introduction of sample bar coding support.

2.7 New ANALYSIS object

The ANALYSIS object will contain unstructured submissions of secondary analysis of sequence read objects, including assemblies, alignments, and clean sequence datasets appropriate for submission to dbEST.

2.8 Changes to Experiment

2.8.1 Deprecated SPOT_DECODE_SPEC, NUMBER_OF_READS_PER_SPOT

The SPOT_DECODE_METHOD tag is deprecated because it was underspecified, and because the spot layout needs to be fully specified in every case. The NUMBER_OF_READS_PER_SPOT tag has been deprecated as it is redundant with the READ_SPEC entries.

2.8.2 Decode options added to Spot Descriptor

A new read attribute READ_SPEC.READ_LABEL allows for the naming of tags (F3, R3).

The EXPECTED_BASECALL_TABLE can be used to lookup the combination of tags that can resolve a given spot's relationship with a set of samples in a sample pool.

Added a new READ_SPEC choice branch called RELATIVE_ORDER, which specifies that the read in question is to be found before or after the specified read.

2.8.3 Changes to EXPERIMENT.PLATFORM

For 454, the following fields apply: KEY_SEQUENCE, FLOW_SEQUENCE, FLOW_COUNT.

For Illumina and AB_SOLiD, the following field should be used from now on: SEQUENCE_LENGTH, intended to be the number of bases/colors in the raw sequence (including both mate pairs and any technical reads). CYCLE_SEQUENCE and CYCLE_COUNT are deprecated.

Added placeholder branches for COMPLETE_GENOMICS and PACBIO_SMRT platforms.

Added instrument_model choice for HELICOS (HeliScope).

2.8.4 Changes to EXPERIMENT.PROCESSING

Several unused fields are deprecated: QUALITY_SCORES.NUMBER_OF_LEVELS and QUALITY_SCORES.MULTIPLIER.

2.9 New SRA Package Object

A new schema SRA.package.xsd has been introduced in order to provide a container for any combination of SRA XML documents, and to allow for applications using SRA objects to aggregate them in any form. SRA packages are not now supported for submission, but eventually will be used in preference to tar archive files.

3 Summary of Deprecated Fields

EXPERIMENT.DESIGN.SPOT_DESCRIPTOR.NUMBER_OF_READS_PER_SPOT

EXPERIMENT.DESIGN.SPOT_DESCRIPTOR.SPOT_DECODE_METHOD

EXPERIMENT.expected_number_bases

EXPERIMENT.expected_number_reads

EXPERIMENT.expected_number_spots

EXPERIMENT.PLATFORM.ILLUMINA/AB_SOLID.CYCLE_SEQUENCE, CYCLE_COUNT

EXPERIMENT.PLATFORM.ILLUMINA.instrument_model[Solexa 1G Genome Analyzer]

EXPERIMENT.PLATFORM.LS454.instrument_model[GS 20, GS FLX]

EXPERIMENT.PROCESSING.NUMBER_OF_LEVELS. There should be only one entry for QUALITY_SCORES.

EXPERIMENT.PROCESSING.MULTIPLIER

RUN. DATA_BLOCK.total_spots

RUN.DATA_BLOCK.FILES.FILE.filetype[_seq.txt, _prb.txt, _sig2.txt, _qhg.txt]

RUN.DATA_BLOCK.format_code

RUN.DATA_BLOCK.number_channels

RUN.DATA_BLOCK.total_reads

RUN.instrument_model

RUN.run_file

RUN.total_data_blocks

RUN.total_data_blocks

RUN.total_reads

RUN.total_spots

SAMPLE.members

STUDY.PROJECT_ID

SUBMISSION.submission_id (use alias instead)

SUBMISSION.ACTIONS.ACTION.HOLD.HoldForPeriod

4 Summary of Required Fields

The following fields are optional only at the level of schema, and for the purpose of providing backward compatibility with old documents. New submissions should use these fields.

EXPERIMENT.TITLE – This field is optional in the schema but will eventually be required. As a business rule, new submissions must have this field set to a value.EXPERIMENT.PLATFORM.ILLUMINA.SEQUENCE_LENGTH – This field is optional in the schema but will be required for new submissions.

SAMPLE.TITLE – This field is optional in the schema but will eventually be required. As a business rule, new submissions must have this field set to a value.

SUBMISSION.CONTACTS – This is optional in the schema and may not be reproduced by the Archive because of the private nature of the content. However, on submission the Archive will require a CONTACTS section.

SUBMISSION.ACTIONS – This is optional in the schema and may not be reproduced by the Archive because of irrelevance. However, on submission the Archive will require a ACTIONS section.

SUBMISSION.ACTIONS.ACTION.HOLD – You must specify a date using the HoldUntilDate attribute.

STUDY.RELATED_STUDIES– This is optional in the schema but required in order to identify the source of the study record information.

RUN.DATA_BLOCK – This is optional in the schema but required for a submission containing run data to be processed.

5 Summary of Impending Changes

5.1 Impending Changes to SUBMISSION – SRA.submission.xsd

Changes expected in next major release

Remove HoldUntilPublication option
Remove submission_id, use alias instead
Remove deprecated fields

5.2 Impending Changes in SAMPLE – SRA.sample.xsd

Changes expected in next major release

Require SAMPLE.TITLE
Require SCIENTIFIC_NAME or TAXON_ID

5.3 Impending Changes in RUN – SRA.run.xsd

Changes expected in next major release

Remove deprecated fields

5.4 Impending Changes in EXPERIMENT – SRA.experiment.xsd

Changes expected in next major release

Require EXPERIMENT.TITLE
Allow only one copy of QUALITY_SCORES
Require ILLUMINA.SEQUENCE_LENGTH for Illumina platform choice

5.5 Other Changes

Add support for the CompleteGenomics, PacificBioSciences sequencing platforms.

6 Summary of Future Changes

Future work will address the following issues:

- RUN.SPOT_DESCRIPTOR specialization to allow for better aggregation of runs to libraries.
- EXPERIMENT.PLATFORM respecification
- EXPERIMENT.PROCESSING respecification
- EXPERIMENT links specification to allow for relationships between experiments.

Illumina HiSeq-2000 Address Transform (Deprecated)

Created: October 15, 2010; Updated: April 12, 2011.

Status	Inactive
Active Date	2010-09-24
Inactive Date	2011-04-12
Scope	INSDC SRA Illumina HiSeq-2000 data

Notice

Reference herein to any specific commercial products, process, or service by trade name, trademark, manufacturer, or otherwise, does not necessarily constitute or imply its endorsement, recommendation, or favoring by the United States Government. The views and opinions of authors expressed herein do not necessarily state or reflect those of the United States Government, and shall not be used for advertising or product endorsement purposes.

1 Overview

Treatment is required to Illumina HiSeq-2000 sequencing data prior to submission to SRA Archives. This treatment must be applied by submitters to their data. A future update will eliminate the need for this treatment.

1.1 Related Documents

- sequenceread tool for converting Illumina data
- SRF toolkit io_lib
- SRA file formats guide

2 Problem Statement

The SRA Illumina Genome Analyzer loaders rely on the GA and GA II address field convention (*flowcell:lane:tile:x:y*) to determine the order of spots in the run data file and to detect duplicates. To constrain the size of these fields, the SRA loader allows up to a maximum of 24,576 for X and Y. With HiSeq-2000 the addressing that reflected use of a discrete camera field per tile has given way to a continuous camera field covering much more area, so Y values up to 200,000 are commonly seen. All HiSeq-2000 runs have this problem.

There is no simple way to fix the SRA loaders to adapt to HiSeq-2000 data. Instead the SRA loaders will adopt a more general strategy of address determination and duplicate detection. In the meantime, submitters must transform their HiSeq-2000 data in such a way as to restore the GA addressing parameters without loss of data or information. This approach will be reversible if it later becomes possible to archive the native addresses.

Runs receiving this treatment should be specially marked. A fix in the SRA loader code will make this treatment unnecessary. The SRA addressing improvement is expected early 2011.

3 Meta Data Requirements

Ensure that the INSTRUMENT_MODEL specified in the Experiment XML is set to *Illumina HiSeq 2000*.

Additionally, depending on which data treatment has been applied, please add to the SRA RUN object RUN ATTRIBUTE of a *HISeq_address_sort* or *HISeq_address_transform* with value set to 1. For example,

```
<RUN_ATTRIBUTE><TAG>HISeq_address_sort</TAG><VALUE>1</
VALUE></RUN_ATTRIBUTE>
```

or equivalent in the Interactive Submission Tool.

4 Data Treatment – Preferred

We have incorporated a near-term modification to the Illumina SRF and Fastq/Qseq loaders to allow retention of the original HiSeq-2000 coordinates within the archive spot names. This modification requires you to order submitted spots by Tile, Y, and then X. If you are submitting SRF files, you will need to pull the spot information from the file, perform the requisite ordering, and then regenerate the SRF file. Ideally, if you have the original *qseq* files, you can re-order them and then regenerate the SRF file from the reordered *qseq* files. A linux sort command for a *qseq* file is:

```
sort -t "`/bin/echo -e '\t'`" -k 4,4n -k 6,6n -k 5,5n qseq.txt > qseq.ordered
```

The *qseq.ordered* file will need to renamed back to the *qseq.txt* format before submitting to the SRA or regenerating an SRF file. You can convert the reordered *qseq* files back into an *SRF* format using the most recent *illumina2srf* utility from sequenceread.

If you have *fastq* files, one method of ordering would be to convert the *fastq* to a *qseq* format and then apply the sort command provided above. If you have *SRF* files without the original *qseq* files, then you can dump *fastq* from the SRF files, convert the *fastq* to ordered *qseq* files, and then regenerate the *SRF* file. Dumping fastq from *SRF* files is accomplished using the most recent version of io_lib. Conversion to an ordered *SRF* via *fastq* dumping results in a loss of filter and 4-channel quality values.

4.1 Method for fastq Conversion/Ordering via qseq Format

You can convert from *fastq* to *qseq* using this Perl script:

```
#!/opt/perl-5.8.8/bin/perl
##############################################################################
#
#                              PUBLIC DOMAIN NOTICE
```

```
#               National Center for Biotechnology Information
#
#  This software/database is a "United States Government Work" under the
#  terms of the United States Copyright Act.  It was written as part of
#  the author's official duties as a United States Government employee and
#  thus cannot be copyrighted.  This software/database is freely available
#  to the public for use. The National Library of Medicine and the U.S.
#  Government have not placed any restriction on its use or reproduction.
#
#  Although all reasonable efforts have been taken to ensure the accuracy
#  and reliability of the software and data, the NLM and the U.S.
#  Government do not and cannot warrant the performance or results that
#  may be obtained by using this software or data. The NLM and the U.S.
#  Government disclaim all warranties, express or implied, including
#  warranties of performance, merchantability or fitness for any particular
#  purpose.
#
#  Please cite the author in any work or product based on this material.
#
###############################################################################

use strict;


if ( ( scalar @ARGV ) eq 0 && ( -t STDIN ) )
    {
        die "'fastq2qseq.pl < fastq file >' OR '| fastq2qseq.pl'\n";
    }


while (<>)
    {
        s/^@//; chomp;
        my $Seq = <>; chomp $Seq;
        <>;
        my $Qual = <>; chomp $Qual;


        my ($Read,$Barcode,$Y,$X,$Tile,$Lane,$Run);


        if ( m/\/(\d{1})$/ )
            {
                $Read = $1; s/\/\d{1}$//;
            }
        if ( m/\#(\S+)$/ )
            {
                $Barcode = $1; s/#\S+$//;
            }
        if ( m/[:_](\d+)$/ )
            {
                $Y = $1; s/[:_]\d+$//;
            }
```

```
        if ( m/[:_](\d+)$/ )
            {
                $X = $1; s/[:_]\d+$//;
            }
        if ( m/[:_](\d+)$/ )
            {
                $Tile = $1; s/[:_]\d+$//;
            }
        if ( m/[:_](\d+)$/ )
            {
                $Lane = $1; s/[:_]\d+$//;
            }
        if ( m/[:_](\d+)$/ )
            {
                $Run = $1; s/[:_]\d+$//;
            }


        print "$_\t$Run\t$Lane\t$Tile\t$X\t$Y\t$Barcode\t$Read\t$Seq\t$Qual\t1\n";
    }
```

You can convert from *qseq* back to fastq using this Perl script:

```
#!/opt/perl-5.8.8/bin/perl
##############################################################################
#
#                            PUBLIC DOMAIN NOTICE
#               National Center for Biotechnology Information
#
#  This software/database is a "United States Government Work" under the
#  terms of the United States Copyright Act.  It was written as part of
#  the author's official duties as a United States Government employee and
#  thus cannot be copyrighted.  This software/database is freely available
#  to the public for use. The National Library of Medicine and the U.S.
#  Government have not placed any restriction on its use or reproduction.
#
#  Although all reasonable efforts have been taken to ensure the accuracy
#  and reliability of the software and data, the NLM and the U.S.
#  Government do not and cannot warrant the performance or results that
#  may be obtained by using this software or data. The NLM and the U.S.
#  Government disclaim all warranties, express or implied, including
#  warranties of performance, merchantability or fitness for any particular
#  purpose.
#
#  Please cite the author in any work or product based on this material.
#
##############################################################################


use strict;


my ( $MACH, $RUN, $LANE, $TILE, $X, $Y, $BARCODE, $READ, $SEQ, $QUAL, $FILTER )
```

```
    = ( 0, 1, 2, 3, 4, 5, 6, 7, 8, 9, 10 );
my $sep = ':';


if ( ( scalar @ARGV ) eq 0 )
    {
        if ( -t STDIN )
            {
                print "qseq2fastq.pl <qseq file> <separator (opt, def=:)>\n";
                print "\tor\n| qseq2fastq.pl\n";
            }
        else
            {
                while (<>) { convertQseqLine ( $_ ); }
            }
    }
else
    {
        my $qseqFile = shift;
        if ( ( scalar @ARGV ) > 0 ) { $sep = shift; }
        open (QSEQ,$qseqFile) or die "Unable to open $qseqFile\n";
        while(<QSEQ>) { convertQseqLine ( $_ ); }
    }


sub convertQseqLine {
    my $line = shift;
    chomp $line;
    my $defline = "";
    my @tmp = split(/\t/,$line);
    if ( $tmp[$MACH] ne "" )
        {
            $defline .= "$tmp[$MACH]";
        }
    if ( $tmp[$RUN] ne "" )
        {
            $defline .= "$sep$tmp[$RUN]";
        }
    $defline .= "$sep$tmp[$LANE]$sep$tmp[$TILE]$sep$tmp[$X]$sep$tmp[$Y]";
    if ( $tmp[$BARCODE] ne "" )
        {
            $defline .= "#$tmp[$BARCODE]";
        }
    if ( $tmp[$READ] ne "" )
        {
            $defline .= "/$tmp[$READ]";
        }
    print "\@$defline\n$tmp[$SEQ]\n+\n$tmp[$QUAL]\n";
}
```

The success of these scripts is dependent on the similarity between your data and a typical Illumina fastq data set. Note that the filter value is lost when converting from *qseq* to *fastq*

but would not have been present in the original fastq file. You can combine the two scripts in a pipe as follows:

```
fastq2qseq.pl foo.fastq | \
sort -t "`/bin/echo -e '\t'`" -k 4,4n -k 6,6n -k 5,5n | \
qseq2fastq.pl > foo_sorted.fastq
```

This pipe will use the default separator, a colon. The second defline following the '+' will be lost if present in the original file but this will not affect loading at the SRA.

5 Alternative Data Treatment – Not Preferred

We have defined the following spot address transformation:

Spot_Tile* = (Spot_Tile x 100) + (Spot_Y / 20000)

Spot_Y* = Spot_Y % 20000

Along with this transformation, some of the transformed spots need to be relocated to later in the *qseq* files to ensure that tile addresses are contiguous, which is another requirement of the SRA loaders. For example, in a file that we received from a submitter s_*1_1_0001_qseq.txt*, is the following sequence of spots (line numbers indicated on the left):

242328:SL-HAG11811220419980

242329:SL-HAG11811221319996

242330:SL-HAG11811218720000

242331:SL-HAG11811229519756

242332:SL-HAG11811238019756

The address transformation will have this result:

242328:SL-HAG1181100220419980

242329:SL-HAG1181100221319996

242330:SL-HAG118110121870

242331:SL-HAG1181100229519756

242332:SL-HAG1181100238019756

In order for the transformed address on record 242330 to load successfully, it is relocated further into the file as record 245435:

242328:SL-HAG1181100220419980

242329:SL-HAG1181100221319996

242330:SL-HAG1181100229519756

242331:SL-HAG1181100238019756

…

245435:SL-HAG118110121870

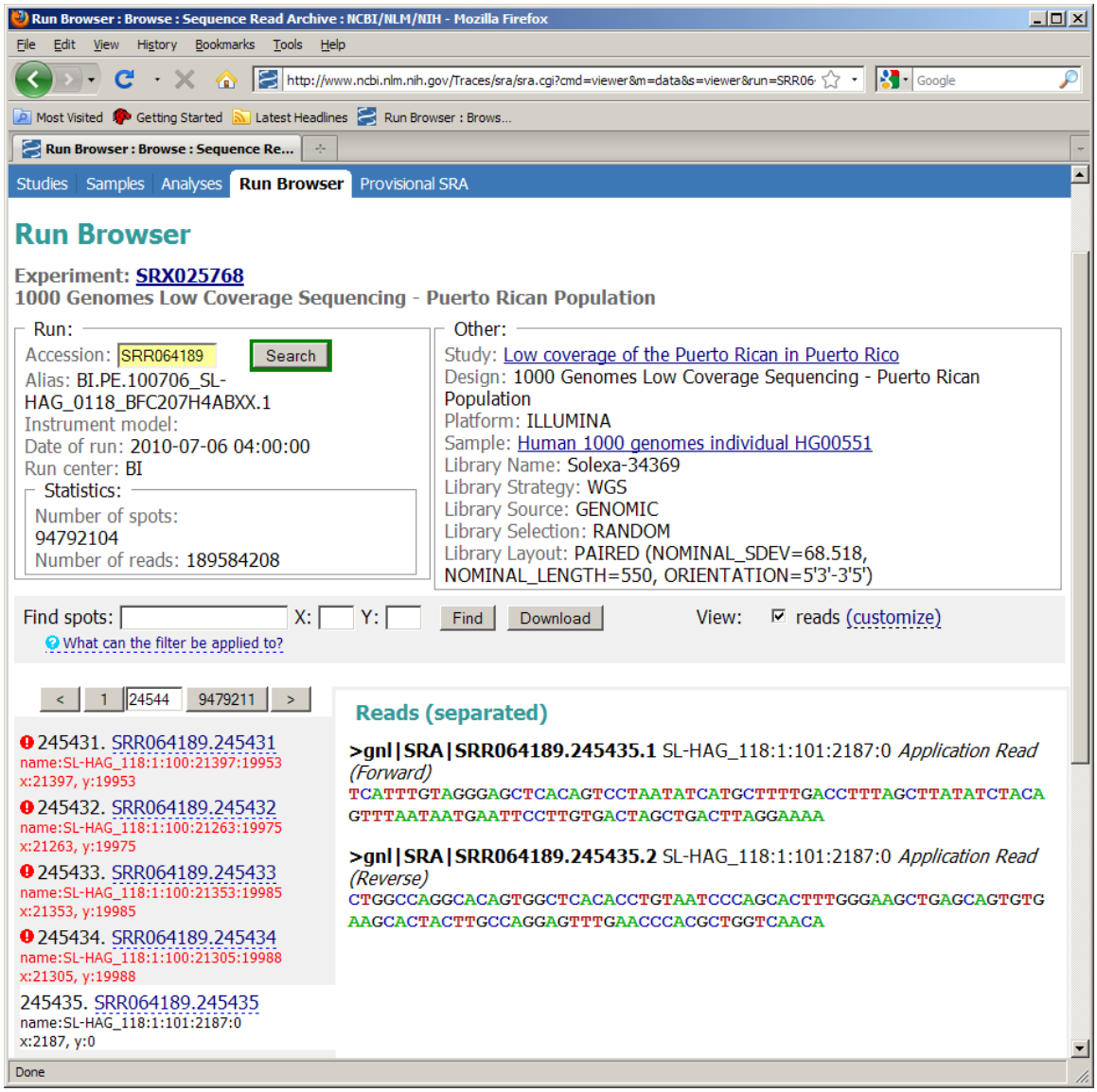
To accomplish this reordering, the transform must buffer spots that are assigned to a tile (101) not currently being input (100). Then, as the tile value changes on input (to 101), the currently buffered spots (tile 101) are written out and the cycle is repeated.

A perl script is provided below that performs this operation on qseq files that should be available in your Illumina run folder.

The SRA submission can contain either the modified *qseq* file(s) (filetype is fastq), or can be converted into SRF format using the *illumina2srf* utility from sequenceread. Be sure to use version 2.1.2 or later, as sequenceread-2.1.1 had problems with the modified tiles (e.g. a tile of 100 changes to 1).

6 Example

A screen capture of the modified and loaded run SRR064189 is presented below.



To view this interactively, please visit:

http://www.ncbi.nlm.nih.gov/Traces/sra/sra.cgi?cmd=viewer&m=data&s=viewer&run=SRR064189

7 Method

```
#!/opt/perl-5.8.8/bin/perl
# This is a code excerpt that performs address transformation on Illumina HiSeq-2000 reads
# in order to make them loadable by SRA loaders relying on GA and GA II spot addressing
# convention.
```

```
###############################################################################
#
#                             PUBLIC DOMAIN NOTICE
#               National Center for Biotechnology Information
#
#  This software/database is a "United States Government Work" under the
#  terms of the United States Copyright Act.  It was written as part of
#  the author's official duties as a United States Government employee and
#  thus cannot be copyrighted.  This software/database is freely available
#  to the public for use. The National Library of Medicine and the U.S.
#  Government have not placed any restriction on its use or reproduction.
#
#  Although all reasonable efforts have been taken to ensure the accuracy
#  and reliability of the software and data, the NLM and the U.S.
#  Government do not and cannot warrant the performance or results that
#  may be obtained by using this software or data. The NLM and the U.S.
#  Government disclaim all warranties, express or implied, including
#  warranties of performance, merchantability or fitness for any particular
#  purpose.
#
#  Please cite the author in any work or product based on this material.
#
###############################################################################



        use strict;


        die "\nHiSeq2000_2_SRA.pl  < qseq file >\n\n"
            if ( scalar @ARGV eq 0 && -t STDIN );


        my $TILE_INDEX = 3;
        my $Y_INDEX = 5;


        # Initialize using first line in qseq file


        my $firstLine = <>;
        my @spot = split(/\t/,$firstLine);
        adjustSpot ( \@spot );
        printSpot ( \@spot );
        my $currTile = $spot[$TILE_INDEX];
        my $nextTile = 0;
        my @nextTileSpots = ();


        # Continue to process qseq file
```

```
while (<>)
    {
        my @spot = split(/\t/,$_);
        adjustSpot ( \@spot );


        # Print spots in *$currTile*


        if ( $spot[$TILE_INDEX] eq $currTile )
            {
                printSpot ( \@spot );
            }


        # Determine first *$nextTile* value and start collecting
        # *$nextTile* spots into @nextTileSpots .


        elsif ( ! ( $nextTile) )
            {
                $nextTile = $spot[$TILE_INDEX];
                push @nextTileSpots,\@spot;
            }


        # After *$nextTile* is set, continue collecting *$nextTile*
        # spots into @nextTileSpots


        elsif ( $spot[$TILE_INDEX] eq $nextTile )
            {
                push @nextTileSpots,\@spot;
            }


        # If *$spot[$TILE_INDEX]* is not *$currTile* or *$nextTile*,
        # then set *$currTile* and *$nextTile* to new values.
        # Output spots collected in @nextTileSpots, and start
        # collecting a new set of *$nextTile* spots in @nextTileSpots.


        else
            {
                printTileSpots ( \@nextTileSpots );
                $currTile = $nextTile;
                $nextTile = $spot[$TILE_INDEX];
                @nextTileSpots = ();
                push @nextTileSpots,\@spot;
            }
    }
```

```
printTileSpots ( \@nextTileSpots );


################################################################
sub printTileSpots {
    my $nextTileSpotsRef = shift;
    foreach my $spotRef ( @$nextTileSpotsRef )
        {
            printSpot ( $spotRef );
        }
}


################################################################
sub adjustSpot {
    my $spotRef = shift;
    $$spotRef[$TILE_INDEX] = ( $$spotRef[$TILE_INDEX] * 100 ) + int ( $
$spotRef[$Y_INDEX] / 20000 ) ;
    $$spotRef[$Y_INDEX] = $$spotRef[$Y_INDEX] % 20000 ;
}


################################################################
sub printSpot {
    my $spotRef = shift;
    print join ( "\t", @$spotRef );
}
```

Illumina SRF Barcode Submissions (Deprecated)

Created: October 27, 2010; Updated: April 12, 2011.

Status	Inactive
Active Date	2010-10-27
Inactive Date	2011-04-12
Scope	INSDC SRA Illumina SRF barcodes

1 Overview

This application note describes the method for specifying metadata that will allow for loading of Illumina barcoded data delivered in SRF format. Future changes will deprecate this application note as well as simplify the submission process for Illumina barcoded data.

1.1 Related Documents

- SRA Barcoding Guide

2 Problem Statement

We have updated the current Illumina SRF loader to accept Illumina barcode submissions. We expect these submissions will be de-multiplexed such that each SRA Run is associated with a specific member name. One feature of this data is that the Illumina barcode sequence occurs between the two 'application' reads within each spot. Although this spot construction allows for the correct member name assignment, it can lead to an issue with dividing the spot into its individual reads. When an EXPECTED_BASECALL_TABLE is present and a member name assigned, our SRF loader will use the BASECALL value or values associated with that member name to partition the spot sequence from left to right using the criteria specified in the 'match_edge', 'max_mismatch', and 'min_match' attributes. A match found within the first 'application' read results in an incorrectly partitioned spot.

We have a solution on the way in the form of a schema update that will allow:

1. Specification of a BASE_COORD value at the same time as an EXPECTED_BASECALL_TABLE, and
2. Incorporation of a SPOT_DESCRIPTOR construct within the Run XML

Support for this schema update within the SRF loader will not be immediate. Until this support is in place, the near-term solution is to replace the EXPECTED_BASECALL_TABLE with a BASE_COORD value and to transfer information encapsulated in the EXPECTED_BASECALL_TABLE to the EXPERIMENT_ATTRIBUTES area of the Experiment XML. Additionally, you must remove the 'read_group_tag' within the READ_LABEL for each MEMBER within the

POOL construct. We expect integration of the new schema into the SRF loader will occur by early 2011.

3 Treatment

In order to apply our near-term solution the following MEMBER contained within a sample POOL:

```
<MEMBER accession="SRS066103" refcenter="BI" refname="35956.0"
member_name="tagged_109_ACAGGTAT">
  <READ_LABEL read_group_tag="tagged_109">barcode</READ_LABEL>
</MEMBER>
```

Must have the 'read_group_tag' attribute removed:

```
<MEMBER accession="SRS066103" refcenter="BI" refname="35956.0"
member_name="tagged_109_ACAGGTAT">
  <READ_LABEL>barcode</READ_LABEL>
</MEMBER>
```

Additionally, this EXPECTED_BASECALL_TABLE for the second read within a SPOT_DESCRIPTOR:

```
<EXPECTED_BASECALL_TABLE>
  <BASECALL match_edge="full" max_mismatch="1" min_match="7"
read_group_tag="tagged_109">ACAGGTAT</BASECALL>
  <BASECALL match_edge="full" max_mismatch="1" min_match="7"
read_group_tag="tagged_110">ACAGTTGA</BASECALL>
  <BASECALL match_edge="full" max_mismatch="1" min_match="7"
read_group_tag="tagged_117">ACCAACTG</BASECALL>
</EXPECTED_BASECALL_TABLE>
```

Transforms into this BASE_COORD specification:

```
<BASE_COORD>69</BASE_COORD>
```

An example of the EXPERIMENT_ATTRIBUTE to add in order to capture information contained in the EXPECTED_BASECALL_TABLE is:

```
<EXPERIMENT_ATTRIBUTE>
  <TAG>EXPECTED_BASECALL_TABLE</TAG>
  <VALUE>
match_edge="full" max_mismatch="1" min_match="7"
read_group_tag="tagged_109" member_name="tagged_109_ACAGGTAT" ACAGGTAT
match_edge="full" max_mismatch="1" min_match="7"
read_group_tag="tagged_110" member_name="tagged_110_ACAGTTGA" ACAGTTGA
match_edge="full" max_mismatch="1" min_match="7"
read_group_tag="tagged_117" member_name="tagged_117_ACCAACTG" ACCAACTG
  </VALUE>
</EXPERIMENT_ATTRIBUTE>
```

Note that we added 'member_name' to permit reinstating the original EXPECTED_BASECALL_TABLE and to maintain the association between this barcode

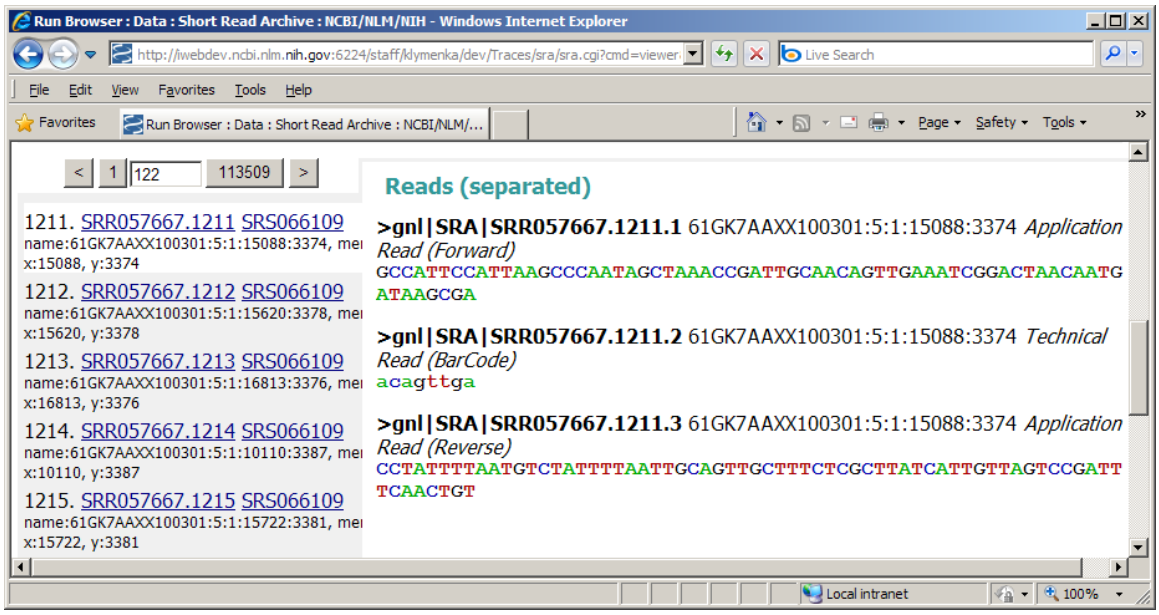
sequence and the member name. Once you have made these modifications to the Experiment XML, the Run should load successfully. A caveat to this near-term solution is that a single BASE_COORD value may not be appropriate for all Runs associated with an Experiment. In this situation, you will need to create additional Experiments, each of which represents a different BASE_COORD value.

Please add a RUN_ATTRIBUTE, *ExpectedBasecallTableInAttribute*, with a value of *yes*, to the Run xml in order to record the presence of an EXPECTED_BASECALL_TABLE attribute.

When all Runs associated with an Experiment are loaded, you can reinstate the EXPECTED_BASECALL_TABLE in the Experiment XML with the qualification that a reload of one of these Runs could cause the underlying spots to partition incorrectly.

4 Example

A screen capture of the desired result for SRR057667 is below.



Note that the barcode sequence, 'acagttga', occurs in the first application read and would have resulted in an incorrectly partitioned spot if the EXPECTED_BASECALL_TABLE were present.

>gnl|SRA|SRR057667.1211.1 61GK7AAXX100301:5:1:15088:3374 *Application Read (Forward)*

GCCATTCCATTAAGCCCAATAGCTAAACCGATTGCAACAGTTGAAATCGGACTAACAATG ATAAGCGA

TCGA Submission Protocol (Deprecated)

Martin Shumway

Created: April 16, 2009; Updated: July 14, 2011.

Status	Inactive
Active Date	2009-04-16
Inactive Date	2012-5-18
Scope	NCBI dbGaP SRA

1 Overview

This document describes the submission protocol for raw sequencing data and primary reference genome alignments for the Cancer Genome Atlas Project (TCGA), a NIH Roadmap study sponsored by the NCI and NHGRI. TCGA sequencing and alignment data come from human clinical samples and are considered identifying. In order to implement research use guidelines and enforce patient privacy rights, these data are accessed by users through the dbGaP authorized access distribution mechanism. Submitters of data to TCGA need to follow similar security procedures by submitting through the protected SRA interface, which deposits data into the dbGaP system. Excerpts of de-identified meta-data are exported to the public SRA and are available for search through the NCBI Entrez system.

1.1 Related Documents

General submission guidelines for the SRA including instructions for using aspera upload: SRA Submission Guide

Submission instructions for BAM files: SRA Analysis Submission Guide

To get help at NCBI, please write to sra@ncbi.nlm.nih.gov .

TCGA Project Home Page at NCI: http://cancergenome.nih.gov/

TCGA Data Guide: https://wiki.nci.nih.gov/display/TCGA/TCGA+Data+Primer

dbGaP TCGA Study Page: http://www.ncbi.nlm.nih.gov/gap?term=phs000178

SRA TCGA Study Page: http://trace.ncbi.nlm.nih.gov/Traces/sra/sra.cgi?study=SRP000677

1.3 Notices

Reference herein to any specific commercial products, process, or service by trade name, trademark, manufacturer, or otherwise, does not necessarily constitute or imply its endorsement, recommendation, or favoring by the United States Government. The views and

opinions of authors expressed herein do not necessarily state or reflect those of the United States Government, and shall not be used for advertising or product endorsement purposes.

2 Data Scope

2.1 Study

The SRA study for TCGA is SRP000677. This study points to the current dbGaP study (phs000178). The current version of the study is v3, but this changes every few months to reflect updates to the phenotypes and sample membership visible to the users of the study.

2.2 Samples

SRA samples are prepared by dbGaP and exported to the public Entrez BioSamples resource in abbreviated form. Here is an example record whose SRA accession is SRS061071:

http://www.ncbi.nlm.nih.gov/biosample/80112

2.3 BAM Files

Along with raw sequencing data. it is now typical to submit primary reference alignments of sequence reads.

The TCGA has mandated submission of primary sequencing data in the form of binary sequence Alignment/Mapping (BAM). The payload of this file contains both the sequencing data (in bases, quality scores,and read names produced by the instrument) and read placements with annotations about strand, alignment, and quality features. BAM files are sufficient to meet the submission needs of this project.

A requirement of BAM submission is that the reference genome be precisely specified. Please see SRA Analysis Submission Guide for specific requirements of BAM files.

2.4 ArchiveBAM Submission (Future)

A replacement for BAM that is suitable for archiving of both raw sequencing data and primary read placements is under design. This will allow for consolidated submissions of sequencing and read placements within the SRA Run object, eliminating much of the complexity associated with BAM file submission. Introduction of this service is expected in 2011.

2.5 SRA Run Submission (Legacy)

BAM files that use runs already archived in the SRA. In order to relate a read_group label to an existing archived SRA run, submitters should include reference to the run in the analysis XML submission, for example:

```
<RUN accession="SRR018666" read_group_label="A"/>
```

3 Submission Modalities

3.1 Submitting Center

You must have a center designation in order to submit sequencing data. Current TCGA centers are:

Center	center_name
Baylor College of Medicine	BCM
BC Cancer Agency Michael Smith Genome Sciences Centre	BCCAGSC
Broad Institute	BI
Harvard Medical School - Raju Kucherlapati Lab	HMS-RK
Johns Hopkins University – University of Southern California collaboration	JHU-USC
University of North Carolina at Chapel Hill - Lineberger Comprehensive Cancer Center	UNC-LCCC
Washington University, Genome Sequencing Center	WUGSC

3.2 Protected SRA

You must upload to the center specific protected host address, for example

gap-upload@ncbi.nlm.nih.gov:/asp-mycenter/protected. You must identify your submission as a protected submission, as follows:

```
<SUBMISSION …>
```

…

```
<ACTIONS>
..
<PROTECT/>
```

3.3 Aspera

You must use the aspera utility. The ftp and secure https protocols are not appropriate for data of this magnitude and are not supported. You must use encryption when transmitting data to NCBI.

3.4 XML

Submission metadata must be rendered in SRA XML. Spreadsheets, tab files, bare BAM files are not sufficient to complete the archiving process. There is no interactive submission tool available for protected SRA submissions.

3.5 Tracking

Submitters should track the progress of their SRA submissions at NCBI.

Entrez SRA is not yet aware of public analysis objects (SRZ accessions). However, you can track submission of analysis objects in one of three ways:

- Using the interactive submission tool, which also highlights problems with submission metadata or files
- Using the display of analysis objects released to SRA, including those accessible only through dbGaP, sorted by accession:

http://trace.ncbi.nlm.nih.gov/Traces/sra/sra.cgi?cmd=table&f=analysis&m=data&s=analysis

- Using the SRA Telemetry feature

SRA telemetry includes SRA xml for each deposited object and a tab file showing the status of each object by its NCBI accession. Deposits of metadata objects including analysis objects (SRZ) can be tracked for each depositor by downloading the SRA_Accessions tab file generated each day for each submitter.

For example,

Accession	SRZ000300
Submission	SRA025062
Status	Live
Updated	2010-10-21T18:19:06Z
Published	2010-10-21T18:19:06Z
Received	2010-10-13T21:08:02Z
Type	ANALYSIS
Center	BI
Visibility	controlled_access
Alias (of the SRA Analysis object, NOT the bam file)	G1743.TCGA-06-0188-01A-01D-0373-08.bam
Md5sum (of the XML object, NOT the bam file)	9324c67f9b47311e9973f50ec9ada3f7

A full description of this tab file can be found in the SRA Submission Guide http://www.ncbi.nlm.nih.gov/books/NBK47532/

3.6 Exchange Area

For most of 2009-2010, the “exchange area”, or anteroom, was utilized as a mechanism for collaborative exchange of TCGA BAM files. This area was needed while NCBI constructed submission pathways for BAM files.

The TCGA Exchange area is being taken out of service to be replaced by regular authorized access distribution.

- The ./asp-mycenter/exchange/TCGA directory has been removed
- The ./asp-mycenter/exchange/TCGA_phs000178 directory is no longer writable. Please do not deposit any new files into this area. BAM files and SRA metadata tar packages should be deposited into the regular ./asp-mycenter/protected area.
- Checksums (md5) should be included in the SRA Analysis metadata xml, rather than in a separate file.
- NCBI will compute a BAM index file (bai) on receipt of the data.

3.7 Release and Publication

The TCGA operates on a **rolling submission policy** meaning that each submission is released immediately. Please specify this in your submission XML:

```
<SUBMISSION …>

…

<ACTIONS>

..

<RELEASE/>
```

Release of project data is the responsibility of dbGaP. dbGaP follows a **periodic release policy** that corresponds to sample phenotype submission, quality control, and release.

4 Data Preparation

4.1 Study

Submitters do not create a SRA study for their submission. Rather, the SRA experiment is set to reference the SRA study. Here are the available studies:

SRA Study	dbGaP study	Genome project id	Title
SRP000677	phs000178	41443	The Cancer Genome Atlas (TCGA)

This binding can be expressed in XML as:

```
<STUDY_REF accession="SRP000677" />
```

4.2 Samples

Sample records are also created by NCBI for use by submitters. Each SRA analysis object and SRA experiment object references one or more sample records. This binding can be expressed in SRA experiment XML as:

```
<SAMPLE_REF accession="SRS096084" />
```

and in SRA analysis XML as:

```
<TARGET accession="SRS061581" sra_object_type="SAMPLE" />
```

You can test the existence of a BioSample record by looking up the TCGA aliquot id or using the SRA Sample accession. For example, try

http://www.ncbi.nlm.nih.gov/biosample/?term=TCGA-06-0876-10A-01D-1003-01

http://www.ncbi.nlm.nih.gov/biosample?term=SRS096084

To find BioSample records in bulk, it is currently necessary to obtain from NCBI a lookup table of dbGaP sample names to BioSample accessions for each dbGaP study that is being submitted. Samples may be in a loaded state or have been received and awaiting phenotype data and are therefore unreleased. It is also possible that samples have been withdrawn from the study. To confirm that the sample names you have correspond to those tracked at dbGaP, and to ensure that the samples for which you intend to submit data are still active in the database, please write NCBI for the latest sample lookup table that relates BioSample records (SRS) to TCGA aliquot barcodes (for example, TCGA-06-0876-10A-01D-1003-01).

TCGA will be migrating to UUIDs as sample names during 2011. dbGaP intends to participate in this migration. Use of BioSample ids (SRSs) will help make this conversion transparent to submitters and for a time will provide a lookup table based on both aliquot bar codes and UUIDs.

4.3 Metadata Preparation

The SRA requires that there exist predefined project (SRP) and sample (SRS) records for each submission to succeed. A TCGA submission consists of one or more experiments (SRX), and one or more runs (SRR), and one or more analysis objects (SRZ). The information content of these respective metadata objects is described in the SRA Analysis Submission Guide.

Do not combine xml metadata with BAM files. Please combine the xml metadata data files into a tar file, for example

```
tar cvf mysubmission.xml.tar A.submission.xml
A.experiment.xml A.run.xml A.analysis.xml
```

The SRA metadata will be made public. Consequently, do not include identifying information in the XML metadata. If information is restricted to the library preparation and run conditions this will not be an issue.

Public metadata are indexed, visible in Entrez, and dumped for bulk access. TCGA short read datasets will appear as normal deposits in every respect except that you cannot see or

download the run or analysis genotyping data. Instead, a message will appear that the user is asked to apply to the relevant Data Use Committee to gain access.

4.4 Run Data Preparation

SRA run data are extracted from the BAM files delivered with the SRA analysis objects. The BAM file is also called out as the run data file. For details please see the SRA Analysis Submission Guide.

4.5 Alignment Preparation

BAM files should follow the requirements of BAM file submission for NIH projects. For details please see SRA Analysis Submission Guide. BAM files should not be compressed or wrapped into another archive container.

4.6 Probes and Capture Arrays

Where appropriate NCBI would like to define probe sets for capture arrays or techniques. These can be simply defined (list of targets and their coordinates is sufficient). These can be provided in spreadsheet form or *bed* file, and NCBI can create accessions in ProbeDB for these data, and attach them to the submitted experiments.

5 Submission Protocol

1. Prepare the submission xml with the new PROTECT action. This tells the SRA that the data are intended for dbGaP.
2. Use your established ssh key pairs for transmission with NCBI. Key pairs provide more security. More than one key pair can be defined, you may wish to dedicate one to the transaction of protected data.
3. Transmit submission articles (xml and data files) to a special server dedicated to delivery of protected datasets:

`ascp -l400m -Q files` asp-XXX@gap-upload.ncbi.nlm.nih.gov:protected/

where XXX is one of asp-<center_name>, for example asp-bi. Note that the –T option is NOT specified so that the data will remain encrypted during transmission.

4. Metadata and data can be delivered asynchronously, one or the other will wait in the protected area until the submission is complete.
5. Inspect the ./outgoing area for annotated XML for objects that have been processed. There will be some delay before the appearance of annotated XML files.
6. Files will be cleaned up automatically as they are processed and moved to dbGaP.

The submission process is the same as that for the open SRA but conducted in isolation. Currently BAM files are processed in the following manner:

1. Each BAM file has its md5 checksum computed. The BAM file name, its checksum, and the submitting center are compared to the stated name, checksum, and center_name provided with the SRA analysis xml.
2. An index file (.bai) is generated from the bam file, which requires scanning the entire file.
3. The BAM header is dumped for internal use.
4. Each BAM file corresponds to a sample in the project. If this sample is not contained in the subject-sample mapping table in dbGaP, the entire submission is rejected.
5. Runs are NOT extracted from the BAM files at this time. SRRs are left in an unloaded state. A future version of the SRA will load these runs from their associated BAM files.
6. The BAM file as delivered along with its index file is added to the list of currently "loaded" analysis objects. At the next periodic release, dbGaP provides these files through the dbGaP authorized access interface.

6 Updates and Withdrawals

Updates of metadata can be handled through the normal SRA channel. Please see the SRA Submission Guide for details about how to update metadata through modify xml submissions.

There does not exist a mechanism to automatically withdraw (suppress) objects in the SRA. Please write to the SRA helpdesk to request suppression of objects. This request is handled by a curator. Suppressed objects remain in the SRA database but are not indexed and not returned in any query. Run and analysis objects that have been suppressed are not available for download from the dbGaP authorized access channel.

7 Example Submissions

Example protected SRA submissions can be found in: ftp://ftp.ncbi.nlm.nih.gov/sra/examples/SRA029111

The files can be downloaded using this command:

```
wget ftp://ftp.ncbi.nlm.nih.gov/sra/examples/SRA029111/*.xml
```

The following files give an example XML package (SRA029111a.xml.tar) package containing four SRA documents to add to the database. A screenshot of the interactive submission tool following successful submission can be seen in this file: SRA029111a.pdf .

SRA029111a.analysis.xml

SRA029111a.experiment.xml

SRA029111a.pdf

SRA029111a.run.xml

SRA029111a.submission.xml

SRA029111a.xml.tar

The following files give an example XML package (SRA029111m.xml.tar) package containing four SRA documents that will modify existing documents already added to the database.

SRA029111m.analysis.xml

SRA029111m.experiment.xml

SRA029111m.run.xml

SRA029111m.submission.xml

SRA029111m.xml.tar

SRA Usability Changes 2010-11-17 (Deprecated)

Status	Inactive
Active Date	2010-11-18
Inactive Date	2012-05-18
Scope	NCBI SRA

1 Overview

The SRA web site http://www.ncbi.nlm.nih.gov/Traces/sra has changed in important ways.

- Static fastq dumps for every run have been removed from the site. Instead users are asked to download the run in archive format and execute a dump of data on their local system.
- The web site Study, Sample, Analysis report pages have been changed
- The Run Browser has been changed
- A new SRA Object search function has been added.

2 Static fastq dumps removed

NCBI has been provisioning text dumps of SRA runs for the past two years. These dumps are executed on a scheduled basis and provide an Archive view of the data, including splitting of mate pair reads into separate files.

The size of this service has now outstripped the ability to store it so it has been removed. From now on, users are asked download runs of interest and execute dumps into the desired format using the SRA SDK toolkit available at http://www.ncbi.nlm.nih.gov/Traces/sra/sra.cgi?cmd=show&f=software&m=software&s=software .

The advantage of this approach is that users can select which format to dump to, can be assured that the data are available soon after loading, and can be assured that the submitter corrections have been applied to their datasets.

Data sets that are available for download through aspera or ftp can be reached this way from the Entrez search result:

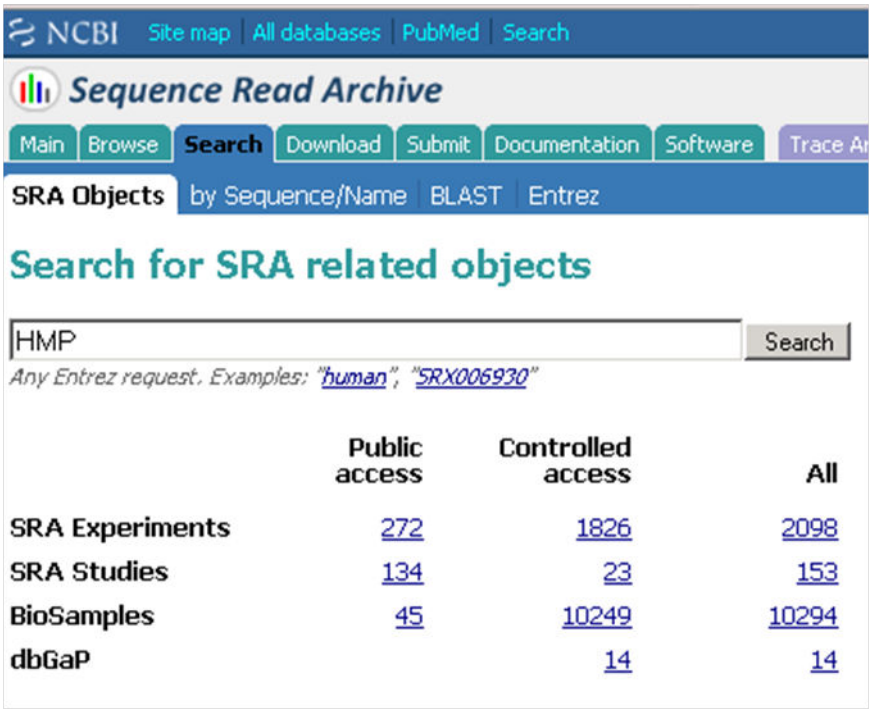
```
Download reads for this experiment in sra or sra-lite
formats

SRA data are provisioned in full "sra" format and "sra-lite"
format objects. With "sra-lite" it is possible to extract
sequence and qualities (for example, fastq format).
Intensity scores are provided only in the full "sra" format.
Extraction of SFF format should begin with full "sra"
```

formatted SRA objects so that the flowgram can be included in the SFF file.
To get "fastq" format see Converting SRA format data into FASTQ in the SRA Handbook.

3 Summary of Web Site Changes

3.1 New SRA Object search function

It is now possible to submit to Entrez SRA a search time and have it return a table of hits organized by SRA object type. Here is an example with the general text search term "HMP":



This query returned hits in both the open (public access) and protected (controlled access) SRAs. Click through any of these hit links in order to access Entrez reports for each type of object.

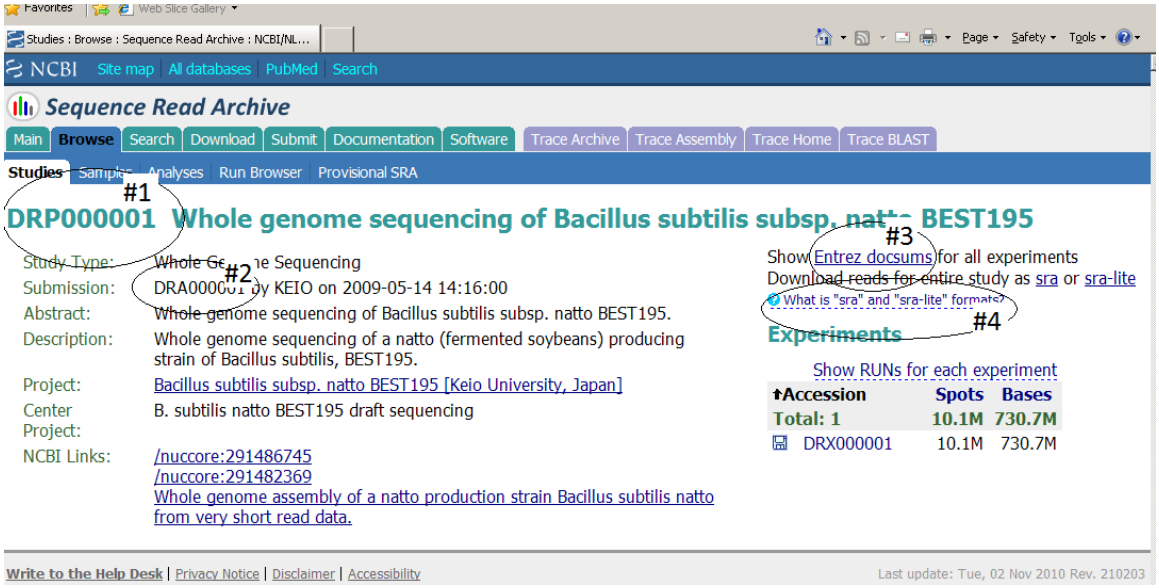
3.2 Browse/Studies, Samples, Analyses tables updated

1. Removed link in title (no hyperlink)
2. Replaced the link under Accession with the hyperlink previously under Title.
3. Removed all FASTQ download links

3.3 Browse/Studies, Samples, Analyses separate item

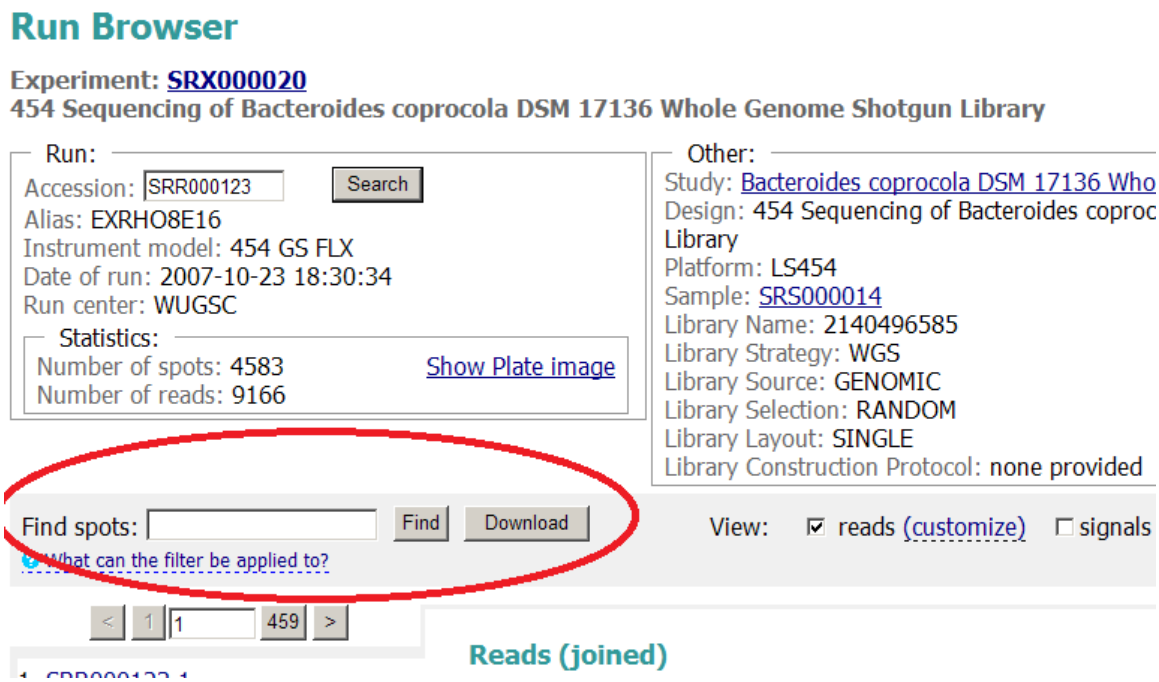
1. Removed links from Accession on title line for Studies, Samples and Analyses

2. Submission link has been removed
3. Added Entrez Docsum
4. Added help line/link



3.4 Browse/Run Browser updates

1 Removed X and Y coordinate search from Run Browser result page



4 Bulk Downloads

4.1 Linux Aspera Bulk Download

It is possible to download a run, experiment, sample, or study using the fasp protocol in Linux. A command template for Aspera using the 'ascp' utility is:

```
ascp -i <key> -QT -L <logdir> -l 100m anonftp@ftp-trace.ncbi.nlm.nih.gov:<src> <dest>
```

Where:

```
key    = <aspera install directory>/aspera/etc/asperaweb_id_dsa.putty
logdir = directory that contains 'aspera-scp-transfer.log' which can be used for
         resumption of an interrupted transfer


src    = source directory or file being downloaded
dest   = destination directory for download
```

An example command for Linux is:

```
ascp -i /opt/aspera/etc/asperaweb_id_dsa.putty -QT -L . –l 100m
anonftp@ftp-trace.ncbi.nlm.nih.gov:/sra/sra-instant/reads/ByExp/
litesra/SRX/SRX000/SRX000007/SRR000001 /tmp
```

4.2 Windows Aspera Bulk Download

It is possible to use Aspera to download a run, experiment, sample, or study using the fasp protocol on Windows. The template for Aspera using the 'ascp' utility is the same except you need a complete path to the 'ascp' utility:

```
"C:\Program Files\Aspera\Aspera Connect\bin\ascp" -i <key> -QT -L
<logdir> -l 100m anonftp@ftp-trace.ncbi.nlm.nih.gov:<src> <dest>
```

Where:

```
key    = "<aspera install directory>\Aspera\Aspera Connect\etc\asperaweb_id_dsa.putty"
```

See 'Linux Aspera Bulk Download' for explanation of logdir, src, and dest.

An example command for Windows is:

```
"C:\Program Files\Aspera\Aspera Connect\bin\ascp" -i "C:\Program Files
\Aspera\Aspera Connect\etc\asperaweb_id_dsa.putty"
 -QT -L "C:\Program Files\Aspera\Aspera Connect\var\log"
-l100m
anonftp@ftp-trace.ncbi.nlm.nih.gov:/sra/sra-instant/reads/ByExp/
litesra/SRX/SRX000/SRX000007/SRR000001 C:\Temp
```

4.3 FTP Download

You can use ftp-trace.ncbi.nih.gov to transfer runs using the ftp protocol. For example, to download 'SRR000001.lite.sra' after logging into ftp-trace.ncbi.nih.gov, use this directory:

```
sra/sra-instant/reads/ByExp/litesra/SRX/SRX000/SRX000007/SRR000001
```