SMRT® Link Troubleshooting Guide (v6.0.0)

Introduction

This document discusses common SMRT Link v6.0.0 issues and how to troubleshoot them.

SMRT Link is the web-based end-to-end workflow manager for the Sequel® System. It includes software applications for designing and monitoring sequencing runs, and analyzing and managing sequence data. The applications include:

- **Sample Setup:** Calculate binding and annealing reactions for preparing DNA samples for use on the Sequel System.
- **Run Design**: Design runs and create and/or import sample sheets which become available on the Sequel System.
- **Run QC**: Monitor run progress, status and quality metrics.
- **Data Management**: Create Projects and Data Sets; manage access permissions for Projects and users; generate QC reports for Data Sets; view, import, or delete sequence, reference, and barcode files.
- **SMRT Analysis**: Perform multiple types of secondary analysis, including sequence alignment, variant detection, *de novo* assembly, structural variant calling, and RNA analysis.

Getting Support

Pacific Biosciences:

- **http://www.pacb.com/support/software-downloads/**
  PacBio web site includes SMRT Link software downloads, release notes, and documentation.
- **http://www.pacb.com/products-and-services/analytical-software/devnet/**
  PacBio Developer's Network web site includes tutorials and Data Sets.
- **http://www.pacb.com/support/technical-support/**
  PacBio Technical Support web site provides direct support for PacBio customers and Service Providers. Email **support@pacb.com** or call 1.877.920.PACB (7222). Current customers can submit an inquiry using our customer portal.

Third-Party Online Communities:

SeqAnswers.com (http://seqanswers.com/forums/forumdisplay.php?f=39) includes a PacBio forum that is actively monitored by the PacBio user community as well as some PacBio scientists and engineers.

Google Groups Mailing Lists:

- **SMRT_isoseq**: https://groups.google.com/forum/#!forum/smrt_isoseq
  Transcript isoform analysis with PacBio Iso-Seq® and/or third-party tools.
- **SMRT_SV**: https://groups.google.com/forum/#!forum/smrt_sv
  Analysis of structural variants.
- **SMRT_kinetics**: https://groups.google.com/forum/#!forum/smrt_kinetics
  Base modification analysis.
- **SMRT_denovo**: https://groups.google.com/forum/#!forum/smrt_denovo
  De novo genome assembly.

General Troubleshooting

SMRT Link Log Files

Installation/Upgrade

These files are useful for investigating and troubleshooting issues that occur during installation or upgrade:

```
$SMRT_ROOT/userdata/log/installer/install.log
$SMRT_ROOT/userdata/log/installer/upgrade.log
$SMRT_ROOT/userdata/log/installer/reconfig.log
```

SMRT Link Services

This file is useful for viewing error messages during routine operations:

```
$SMRT_ROOT/userdata/log/smrtlink-analysisservices-gui/secondary-smrt-server.log
```

SMRT Link Java Console

This file is useful for investigating major system failures, such as memory overflow:

```
$SMRT_ROOT/current/bundles/smrtlink-analysisservices-gui/current/private/pacbio/smrtlink-
analysisservices-gui/java.std*
```

WSO2

These files are useful for investigating issues with user rights and roles, such as authentication problems:

```
$SMRT_ROOT/current/bundles/smrtlink-analysisservices-gui/current/private/pacbio/smrtlink-
analysisservices-gui/wso2am-2.0.0/repository/logs/wso2carbon.log
```

```
$SMRT_ROOT/current/bundles/smrtlink-analysisservices-gui/current/private/pacbio/smrtlink-
analysisservices-gui/wso2am-2.0.0/repository/logs/wso2-apigw-errors.log
```

SMRT Analysis Job - Main Pipeline Logs

These files are useful for investigating any SMRT Analysis job failures; they are located in the `jobs_root` directory, for example `$SMRT_ROOT/userdata/jobs_root/004/004000/logs`.

```
└── logs
    ├── master.log
    └── pbsmrtpipe.log (Less verbose than master.log.)
```

SMRT Analysis Job - Task logs

These files are useful for investigating task-specific failures. Within the task directory is task-level output from individual task executions. Example: `$SMRT_ROOT/userdata/jobs_root/004/004000/tasks/pbcore-tools.tasks.filterdataset-0/`.

```
└── tasks
    └── pbalign.tasks.pbalign_ccs-1
        ├── cluster.stderr
        ├── cluster.stdout
        ├── stderr
        └── stdout
```

SMRT Analysis Job - Standard Error and OUT logs

Logs in the job top-level directory contain information returned from the shell or `pbscala`. These files are useful for investigating job failures at the SMRT Link level, for example database errors. The `*job.std*` files contain

STDERR/STDOUT from the main job execution script. Note that some of these files may be present for other job types, such as Data Set import or merging.

```
├── job.stderr
├── job.stdout
├── pbscala-job.stderr
└── pbscala-job.stdout
```

Sending Log Files to Technical Support

Troubleshooting information can be sent to Pacific Biosciences' Technical Support multiple ways. The following two methods **require** a connection to the PacBio Event Server and Update Server.

- From the SMRT Link menu: **About > Troubleshooting Information > Send**.
- From a SMRT Link "Failed" analysis Results page: Click **Send Log Files**.

If there is **no** connection to the PacBio Event Server, run the following command to generate a .tgz file and email the file to **support@pacb.com** to file a case:

- `$SMRT_ROOT/admin/bin/tsreport-install --bundle`

SMRT Link Configuration Files

smrtlink.config

This is the master configuration file generated by the installer; it has the following format:

```
# smrtlink config

# 'install' settings
install__user='secondarytest';
install__group='Domain Users';
install__sluuid='__USE_DEFAULT__';              # Current val: '1e401734-a193-4863-bbae-
2e1294e77401'

# 'system' settings
system__memtotal='__USE_DEFAULT__';             # Current val: '70866960384'

# 'smrtlink' settings
smrtlink__dnsname='smrtlink-alpha.nanofluidics.com';
smrtlink__gui_port='8080';
smrtlink__services_port='8081';
smrtlink__gui_minmem='__USE_DEFAULT__';         # Current val: '3392'
smrtlink__gui_maxmem='__USE_DEFAULT__';         # Current val: '3392'
smrtlink__services_minmem='__USE_DEFAULT__';    # Current val: '16896'
smrtlink__services_maxmem='__USE_DEFAULT__';    # Current val: '16896'
smrtlink__mail_host='__USE_DEFAULT__';          # Current val: ''
smrtlink__mail_port='__USE_DEFAULT__';          # Current val: '25'
smrtlink__mail_user='__USE_DEFAULT__';          # Current val: ''
smrtlink__mail_password='__USE_DEFAULT__';      # Current val: ''
smrtlink__extended_cell_use_enable='__USE_DEFAULT__';  # Current val: 'false'
```

To update the system configuration, PacBio recommends that you modify this file and then run `$SMRT_ROOT/admin/bin/smrt_reconfig` to generate `smrtlink-system-config.json` and `preset.xml`.

- **Note**: If you edit either `smrtlink-system-config.json` or `preset.xml` manually, they will be **overwritten** by the next update or `smrt_reconfig` call; **only** `smrtlink.config` is preserved.

smrtlink-system-config.json

Use this file to view all the configurations used by SMRT Link services. As mentioned earlier, this is automatically generated from `smrtlink.config`, and local edits may be overwritten by future updates. However it is more readable and useful for debugging services than the installer configuration file.

```
{
  "smrtflow": {
    "server": {
      "port": 8081,
      "manifestFile": "/pbi/dept/secondary/siv/smrtlink/smrtlink-alpha/smrtsuite_170220/install/
smrtlink-release_5.1.0.21036/etc/pacbio-manifest.json",
      "eventUrl": "https://smrtlink-eve.pacbcloud.com:8083",
      "dnsName": "smrtlink-alpha.nanofluidics.com",
      "bundleDir": "/pbi/dept/secondary/siv/smrtlink/smrtlink-alpha/smrtsuite_170220/install/
smrtlink-release_5.1.0.21036/bundles/smrtlink-analysisservices-gui/current/private/pacbio/
smrtlink-analysisservices-gui/resources/pacbio-bundles"
    },
    "engine": {
      "maxWorkers": 24,
      "jobRootDir": "/pbi/dept/secondary/siv/smrtlink/smrtlink-alpha/smrtsuite_170220/userdata/
jobs_root",
      "pbsmrtpipePresetXml": "/pbi/dept/secondary/siv/smrtlink/smrtlink-alpha/smrtsuite_170220/
userdata/config/preset.xml"
    },
    "db": {
      "properties": {
        "databaseName": "smrtlinkdb",
        "user": "smrtlink_user",
        "password": "password",
        "portNumber": 8085,
        "serverName": "localhost"
      }
    }
  },
  "pacBioSystem": {
    "tomcatPort": 8080,
    "tomcatMemory": 1024,
    "smrtLinkServerMemoryMin": 16896,
    "smrtLinkServerMemoryMax": 16896,
    "smrtViewPort": 8084,
    "tmpDir": "/pbi/dept/secondary/siv/smrtlink/smrtlink-alpha/smrtsuite_170220/userdata/
tmp_dir",
    "logDir": "/pbi/dept/secondary/siv/smrtlink/smrtlink-alpha/smrtsuite_170220/userdata/log/
smrtlink-analysisservices-gui",
    "pgDataDir": "/pbi/dept/secondary/siv/smrtlink/smrtlink-alpha/smrtsuite_170220/userdata/
db_datadir",
    "enableCellReuse": false,
    "remoteBundleUrl": "http://smrtlink-update.pacbcloud.com:8084",
    "smrtLinkSystemRoot": "/pbi/dept/secondary/siv/smrtlink/smrtlink-alpha/smrtsuite_170220",
    "smrtLinkSystemId": "1e401734-a193-4863-bbae-2e1294e77401",
    "mailHost": null,
    "mailPort": 25,
    "mailUser": null,
    "mailPassword": null
  },
  "comment": "Created: Mon Dec 18 18:20:25 PST 2017"
}
```

preset.xml

Use this file to view the configuration presets for running `pbsmrtpipe` or when a SMRT Analysis job fails.
Example: `$SMRT_ROOT/smrtlink/userdata/config/preset.xml`:

```xml
<?xml version="1.0" ?>
<pipeline-template-preset id="MyPreset">

    <!-- Pbsmrtpipe Engine Options -->
    <options>
        <!-- Enable Distributed Mode -->
        <option id="pbsmrtpipe.options.distributed_mode">
            <value>True</value>
        </option>
        <!-- Enable file chunking -->
        <option id="pbsmrtpipe.options.chunk_mode">
            <value>True</value>
        </option>
         <!-- This will be disabled if pbsmrtpipe.options.distributed_mode is False -->
        <option id="pbsmrtpipe.options.cluster_manager" >
           <value>/opt/pacbio/smrtlink/userdata/generated/config/jms_templates</value>
        </option>

        <!-- Total Number of slocs/processors a pbsmrtpipe instance will use -->
        <option id="pbsmrtpipe.options.max_total_nproc" >
            <value>1000</value>
        </option>

        <!-- MAX Number of Processors per Task that will be used -->
        <option id="pbsmrtpipe.options.max_nproc">
            <value>16</value>
        </option>

        <!-- MAX Number of Chunks per Chunkable Task that will be used -->
        <option id="pbsmrtpipe.options.max_nchunks">
            <value>24</value>
        </option>

        <!-- MAX Number of Workers -->
        <option id="pbsmrtpipe.options.max_nworkers">
            <value>100</value>
        </option>

        <!-- Temporary directory (on the execution nodes) -->
        <option id="pbsmrtpipe.options.tmp_dir">
            <value>/opt/pacbio/smrtlink/userdata/tmp_dir</value>
        </option>

        <!-- Status Progress URL -->
        <option id="pbsmrtpipe.options.progress_status_url">
            <value></value>
        </option>

        <!-- Flag to immediately exit if a task fails -->
        <option id="pbsmrtpipe.options.exit_on_failure">
            <value>False</value>
        </option>

        <!-- Flag to enable DEBUG level in master.log and stdout -->
        <option id="pbsmrtpipe.options.debug_mode">
            <value>False</value>
        </option>

    </options>

    <!-- Default override for task options -->
    <task-options />

</pipeline-template-preset>
```

SMRT Link Administration

SMRT Link Recovery Installation

Use the following steps to recover a SMRT Link installation that is not working correctly or has suffered major corruption, such as files deleted, moved, and so on. The most recent database backup can be used to restore job history and LDAP settings to the new database.

Set $SMRT_USER, $SMRT_ROOT:

```
SMRT_ROOT=/opt/pacbio/smrtlink
SMRT_USER=smrtanalysis
su - ${SMRT_USER}
```

Stop Running Services. Terminate Processes if Necessary:

```
${SMRT_ROOT}/admin/bin/services-stop
${SMRT_ROOT}/admin/bin/services-start
```

Backup the Existing SMRT Link Installation Directory:

```
sudo mv ${SMRT_ROOT} ${SMRT_ROOT}.bak
```

This directory can be removed once the new installation has been verified and failure analysis completed.

Install SMRT Link: This installation command **preserves** the existing configuration from the backed-up installation. Make sure to install the **same** version of SMRT Link.

```
smrtlink_release-tarball.run \
--rootdir ${SMRT_ROOT} \
--configfile ${SMRT_ROOT}.bak/userdata/config/smrtlink.config \
--batch
```

To invoke the interactive installer and select new configuration settings, for example, when migrating to a new server, use:

```
smrtlink-x.x.x-release-tarball.run --rootdir ${SMRT_ROOT}
```

Restoring Databases from Backup

Stop the Database Server:

```
${SMRT_ROOT}/admin/bin/dbhelper --stop-db
```

Restore the SMRT Link Database:

```
${SMRT_ROOT}/admin/bin/dbhelper --restore smrtlinkdb \
--restore-file ${SMRT_ROOT}.bak/userdata/db_datadir/backups/upgrade/latest_smrtlinkdb.sql
```

Restore User Rights and Roles, LDAP Configuration Database (WSO2):

```
${SMRT_ROOT}/admin/bin/dbhelper --restore wso2am \
--restore-file ${SMRT_ROOT}.bak/userdata/db_datadir/backups/upgrade/latest_ws02am.sql
```

Start Services:

```
${SMRT_ROOT}/admin/bin/services-start
```

SMRT Link Processes Sometimes Exceed Soft User Limits

Symptom: Under normal usage, SMRT Link can spawn processes exceeding the user resource limits. The resulting errors can take a number of forms, such as the following:

```
-bash: fork: retry: Resource temporarily unavailable
su - smrtanalysis : cannot set userid: Resource temporarily unavailable.
java.lang.OutOfMemoryError: unable to create new native thread
```

Diagnosis: A snapshot of SMRT Link thread usage can be determined by using:

```
SMRT_USER="smrtanalysis"
ps -Lf -u ${SMRT_USER} | wc -l
```

Compare this with the current soft resource limit settings for `max user processes` and `open files`:

```
ulimit -u -n
```

Solution: The soft resource limit can be increased using:

```
ulimit -u 8192
ulimit -n 8192
```

We have found that a limit of 8192 for `max user processes` and `open files` seems to work in most cases.

```
$ ulimit -u -n
max user processes              (-u) 8192
open files                      (-n) 8192
```

The hard resource limit, which is the upper bound of the range a user can set for itself, is sometimes set lower than what is recommended for SMRT Link, preventing the soft resource limit from being set. Following is the output from `ulimit`:

```
$ ulimit -n 8192
-bash: ulimit: open files: cannot modify limit: Operation not permitted
```

In this case, the hard resource limit must be increased. Check the current limits:

```
ulimit -Hu -Hn
```

To change the hard resource limits, edit `/etc/security/limits.conf`, and add the following lines:

```
smrtanalysis - nproc 8192
smrtanalysis - nofile 8192
```

Log out of any user shells for the `$SMRT_USER` and log back in. Check the new `ulimit` settings using `ulimit -u -n`.

Stop, then restart SMRT Link services:

```
$SMRT_ROOT/admin/bin/services-stop
ps -ef | grep smrt
# kill any reported processes still running.
$SMRT_ROOT/admin/bin/services-start
```

Troubleshooting Analysis Services

Confirm that the services were started by running `services-start`:

```
$SMRT_ROOT/smrtsuite/admin/bin/services-start

Applying config

Completed applying config

Checking services daemon...

Checking GUI webserver daemon...

Starting services daemon...

Starting GUI webserver daemon...
```

```
Using CATALINA_BASE:   /home/UNIXHOME/fas/test/smrtsuite/install/smrtsuite-
fromsrc_3.0.4.173984,173940-173984-173628-173506/bundles/smrtlink-analysisservices-gui/install/
smrtlink-analysisservices-gui-fromsrc_3.0.4.173984/private/pacbio/smrtlink-analysisservices-gui/
apache-tomcat-8.0.26

Using CATALINA_HOME:   /home/UNIXHOME/fas/test/smrtsuite/install/smrtsuite-
fromsrc_3.0.4.173984,173940-173984-173628-173506/bundles/smrtlink-analysisservices-gui/install/
smrtlink-analysisservices-gui-fromsrc_3.0.4.173984/private/pacbio/smrtlink-analysisservices-gui/
apache-tomcat-8.0.26

Using CATALINA_TMPDIR: /home/UNIXHOME/fas/test/smrtsuite/install/smrtsuite-
fromsrc_3.0.4.173984,173940-173984-173628-173506/bundles/smrtlink-analysisservices-gui/install/
smrtlink-analysisservices-gui-fromsrc_3.0.4.173984/private/pacbio/smrtlink-analysisservices-gui/
apache-tomcat-8.0.26/temp

Using JRE_HOME:        /home/UNIXHOME/fas/test/smrtsuite/install/smrtsuite-
fromsrc_3.0.4.173984,173940-173984-173628-173506/admin/bin/../../../../current/bundles/smrtlink-
analysisservices-gui/current/private/thirdparty/jre/jre_8u40

Using CLASSPATH:       /home/UNIXHOME/fas/test/smrtsuite/install/smrtsuite-
fromsrc_3.0.4.173984,173940-173984-173628-173506/bundles/smrtlink-analysisservices-gui/install/
smrtlink-analysisservices-gui-fromsrc_3.0.4.173984/private/pacbio/smrtlink-analysisservices-gui/
apache-tomcat-8.0.26/bin/bootstrap.jar:/home/UNIXHOME/fas/test/smrtsuite/install/smrtsuite-
fromsrc_3.0.4.173984,173940-173984-173628-173506/bundles/smrtlink-analysisservices-gui/install/
smrtlink-analysisservices-gui-fromsrc_3.0.4.173984/private/pacbio/smrtlink-analysisservices-gui/
apache-tomcat-8.0.26/bin/tomcat-juli.jar

Using CATALINA_PID:    /home/UNIXHOME/fas/test/smrtsuite/install/smrtsuite-
fromsrc_3.0.4.173984,173940-173984-173628-173506/admin/bin/../../../../rwdir/run/catalina.pid

Tomcat started.

Completed starting the services and GUI webserver

Waiting for services to start...

Failed to make request to http://smrtlink-release:8080

Failed to make request to http://smrtlink-release:8081/status

Failed to make request to http://smrtlink-release:8081/secondary-analysis/datasets/subreads

Services started successfully.
```

The last line of output should indicate `Services started successfully`.

If logged into the host running SMRT Link, check that the services are up and running:
(**Note**: The following command returns **nothing** on success, and an error message on failure to connect.)

```
$SMRT_ROOT/smrtsuite/admin/bin/services-status
```

To check that the services can be reached from a web browser or other location:

```
curl http://smrtlinkhost:9091/status
{
  "uuid": "8111633f-8969-4e32-b0c0-1edda976b957",
  "version": "0.50.5-SNAPSHOT",
  "id": "pacbio.smrtservices.smrtlink_analysis",
  "uptime": 230469,
  "message": "Services have been up for 3 minutes and 50.469 seconds.",
  "user": "smrtanalysis"
}
```

Note: The URL can also be entered directly into a browser window.

Alternately, a more detailed service status can be displayed using `pbservice`:

```
$SMRT_ROOT/smrtsuite/smrtcmds/bin/pbservice status --host smrtlinkhost --port 9091
[INFO] 2016-02-26 21:42:20,639 Starting tool version 0.2.0
[INFO] 2016-02-26 21:42:20,670 Starting new HTTP connection (1): smrtlink-release
[INFO] 2016-02-26 21:42:20,680 Starting new HTTP connection (1): smrtlink-release
[INFO] 2016-02-26 21:42:20,706 Starting new HTTP connection (1): smrtlink-release
[INFO] 2016-02-26 21:42:20,732 Starting new HTTP connection (1): smrtlink-release
[INFO] 2016-02-26 21:42:20,754 Starting new HTTP connection (1): smrtlink-release
[INFO] 2016-02-26 21:42:20,775 Starting new HTTP connection (1): smrtlink-release
[INFO] 2016-02-26 21:42:20,792 Starting new HTTP connection (1): smrtlink-release
[INFO] 2016-02-26 21:42:20,816 Starting new HTTP connection (1): smrtlink-release
[INFO] 2016-02-26 21:42:20,833 Starting new HTTP connection (1): smrtlink-release
<ServiceAccessLayer http://smrtlinkhost:9091 >
Status Services have been up for 18 minutes and 55.387 seconds.
----------
Dataset Summary
----------
SubreadSets 0
HdfSubreadSets 0
ReferenceSets 0
AlignmentSets 0
----------
All Job types Summary
----------
import-dataset Jobs 0
----------
merge-datasets Jobs 0
----------
convert-fasta-reference Jobs 0
----------
pbsmrtpipe Jobs 0
----------
[INFO] 2016-02-26 21:42:20,851 exiting with return code 0 in 0.21 sec.
```

Listing ports opened for listening on SMRT Link host:

```
$ ssh smrtlinkhost 'netstat -n -l | grep 90'
tcp        0        0 0.0.0.0:9090            0.0.0.0:*              LISTEN
tcp        0        0 0.0.0.0:9091            0.0.0.0:*              LISTEN
tcp        0        0 127.0.0.1:9092          0.0.0.0:*              LISTEN
tcp        0        0 0.0.0.0:9093            0.0.0.0:*              LISTEN
tcp        0        0 0.0.0.0:9094            0.0.0.0:*              LISTEN
```

Upgrade: Ensuring SMRT Analysis Services are Stopped

Sometimes, the `$SMRT_ROOT/admin/bin/services-stop` script may leave residual processes related to SMRT Link. Following are a few things you can check to ensure that this has not happened, and what to do if services remain.

1. Always ensure that you have attempted to stop the services using the normal script. (You may safely run the script repeatedly.)

   ```
   $SMRT_ROOT/admin/bin/services-stop
   ```

2. Check to see if any SMRT Link related processes remain:

   ```
   ps -ef | grep "smrt"
   ```

If any SMRT Link processes remain, the command will return more than just the `grep` command. With all processes successfully stopped, the output will **only** include the `grep` command.

```
$ ps -ef | grep "smrt"

smrtana+  77970      1  0 Aug09 ?        00:00:28 /home/smrtanalysis/v5-test/smrtlink/install/
smrtlink-release_5.0.0.6792/bundles/smrtlink-analysisservices-gui/install/smrtlink-analysisservices-
gui-release_5.0.0.6752/private/thirdparty/postgresql/postgresql_9.6.1/bin/postgres -D /home/
smrtanalysis/v5-test/smrtlink/userdata/db_datadir/dbstore -p 55560 -h localhost
```

```
smrtana+  78251     1  0 Aug09 ?        00:00:00 sh /home/smrtanalysis/v5-test/smrtlink/install/
smrtlink-release_5.0.0.6792/bundles/smrtlink-analysisservices-gui/install/smrtlink-analysisservices-
gui-release_5.0.0.6752/private/pacbio/smrtlink-analysisservices-gui/wso2am-2.0.0/bin/wso2server.sh

smrtana+ 123086  77970  0 09:08 ?        00:00:00 postgres: smrtlink_user smrtlinkdb 127.0.0.1(2138)
idle

smrtana+ 123088  77970  0 09:08 ?        00:00:00 postgres: smrtlink_user smrtlinkdb 127.0.0.1(2139)
idle

smrtana+ 123090  77970  0 09:08 ?        00:00:00 postgres: smrtlink_user smrtlinkdb 127.0.0.1(2140)
idle

smrtana+ 123349 123317  0 09:18 pts/0    00:00:00 grep smrt
```

If you receive any output besides your `grep` process, first attempt to shut down the processes cleanly with Signal 15. This can be done with either `pkill` or `kill`. (The `-15` is the Signal 15, which will attempt to shut the process down cleanly. The `-f` in both `pkill` and `pgrep` ensures that the entire command-line is matched against the string `smrt`.)

```
pkill -15 -f "smrt" or kill -15 $(pgrep -f "smrt")
```

Check again to see if any SMRT Link processes remain:

```
ps -ef  | grep "smrt"
```

If any processes are returned (besides the `grep` process again), repeat the above step. If processes still remain, send a Signal 9 to terminate those processes:

```
pkill -9 -f "smrt" or kill 9 $(pgrep -f "smrt")
```

Check again to see if any SMRT Link processes remain:

```
ps -ef  | grep "smrt"
```

If you still see SMRT Link processes from the output, those processes may be hung. At this point the best course of action is a reboot.

Please be aware that using `grep` to search for the string `smrt` will result in all processes returned if the chosen user name contains the string `smrt`, such as the recommended `smrtanalysis`. The example below accounts for that particular case, so adjust as necessary. Though, the `pgrep` and `pkill` examples do **not** match against username.

```
ps -ef | grep -E "^smrt.*smrt"
```

Cannot Create an Iso-Seq 3 Analysis After SMRT Link v6.0.0 Upgrade

Symptom: Users are prevented from creating a new Iso-Seq 3 analysis after upgrading to SMRT Link v.6.0.0.

Diagnosis: After upgrading SMRT Link to v6.0.0, attempting to create a new analysis using the Iso-Seq 3 or Iso-Seq 3 with Mapping applications may result in the following error message in the UI:

```
Error: Uncaught (in promise): Unable to find dataSet with id 4d1ae852-b6b3-83b3-6006-9e63cc5b7723
(404) Resource Not Found: Unable to find dataSet with id 4d1ae852-b6b3-83b3-6006-9e63cc5b7723
```

This error does **not** occur when using any other analysis application.

Solution: The following step must be performed during the upgrade, as documented in Step 6 on page 5 of the document **SMRT Link Software Installation (v6.0.0)**.

```
${SMRT_ROOT}/admin/bin/import-canneddata
```

This command imports PacBio-provided data files required to run several SMRT Link analysis applications. The files include reference genome files (lambda virus genome data for resequencing, HIV HXB2 reference for Minor Variant Analysis, and human HLA) and barcode FASTA files for demultiplexing.

Configuration: Advanced Distributed Computing Configuration

PacBio supports distributed job submission in SMRT Link using the following job schedulers:

- GridEngine: SGE, OGE, UGE (**Note**: GridEngine variants are for the most part interoperable.)
- PBS /Torque
- LSF
- SLURM

Configuring Job Distribution

Job distribution is configured interactively at installation. Configuration files are generated at installation and reconfiguration, and in most cases do not need additional modification.

To review and/or modify configuration settings for SMRT Link job distribution at any time, use:

```
$SMRT_ROOT/admin/bin/smrt_reconfig
```

New configuration settings will be automatically applied following a restart of SMRT Link Services:

```
$SMRT_ROOT/admin/bin/services-stop
$SMRT_ROOT/admin/bin/services-start
```

Switching Distributed Computing On or Off in SMRT Link

The recommended way to enable/disable distributed computing is to run `$SMRT_ROOT/admin/bin/smrt_reconfig` and select `NONE` for the `jms_type`.

Checking JMS Configuration

Use the following command to see the currently-configured job submission command:

```
$SMRT_ROOT/admin/bin/runjmscmd --start --test
```

Output:

```
noexec: LOGNAME='smrtanalysis' PATH='/usr/bin:/usr/bin:/bin'
SGE_CELL='default'
SGE_ROOT='/var/lib/gridengine'
TERM='xterm'
USER='smrtanalysis'
/usr/bin/qsub -S /bin/bash -sync y -V -q prod66
-N J112233.runjmscmd_test_jobname
-o /path/to/stdout.log -e /path/to/stderr.log
-pe smp 1 'tmpcmd.sh
--arg1 --arg2 "string arg with spaces"'
```

Customizable JMS Configuration File

To specify resource request lists for `mem_free` and `h_rt` to SGE `qsub` (for example, to define `qsub` options in the variable `SGE_STARTARGS`), add the following line to `$SMRT_ROOT/smrtlink/userdata/user_jmsenv/user.jmsenv.ish`:

```
SGE_STARTARGS="-l mem_free=2G, h_rt=120:0:0"
```

To make changes to other JMS environment variables:

```
JMS_TYPE="SGE"
export SGE_ROOT="/opt/gridengine";
export SGE_CELL="default";
QUEUE="high_mem";
PE="mpi";
```

Restarting SMRT Link Services is **not** needed.

Generated JMS Configuration Files

The following configuration files are generated by the installer. Do **not** edit them as they will **not** persist between upgrades. Customizations should be added in `$SMRT_ROOT/smrtlink/userdata/user_jmsenv/user.jmsenv.ish`.

This file contains JMS environment variables:

`$SMRT_ROOT/smrtlink/userdata/generated/config/jmsenv/jmsenv.ish`

This file is a command-line template for submitting jobs:

`$SMRT_ROOT/smrtlink/userdata/generated/config/jms_templates/start.tmpl`

This file is a command-line template for stopping jobs:

`$SMRT_ROOT/smrtlink/generated/config/jms_templates/stop.tmpl`:

Moving the SMRT Link Jobs Directory to a New Location

This procedure is for moving the SMRT Link jobs directory from the default location to an alternate location.

- Make sure **all** currently-running analyses are completed before performing the following steps.
- Perform the steps while logged in as the `$SMRT_USER`.

1. Stop SMRT Link Services:
   ```
   ${SMRT_ROOT}/admin/bin/services-stop
   ```

2. Move SMRT Link jobs to the new file system location:
   ```
   NEW_JOBS_ROOT=/path/to/new/jobs_root/
   mkdir -p ${NEW_JOBS_ROOT}
   mv ${SMRT_ROOT}/userdata/jobs_root.local/* ${NEW_JOBS_ROOT}
   ```

3. Redirect the symlink to the new location:
   ```
   ln -sfn ${NEW_JOBS_ROOT} ${SMRT_ROOT}/userdata/jobs_root
   ```

4. Start SMRT Link Services:
   ```
   ${SMRT_ROOT}/admin/bin/services-start
   ```

Run Design

Run Design Save/Import Failure - Postgresql Encoding Issue

Symptom: You cannot save or import Run Designs, and there is no visible output or error message when using the graphical user interface.

Diagnosis: This is caused by the Postgres database being encoded in an ISO-8859 format (or LATIN[0-9] as Postgres reports it), while the SMRT Link instance expects **UTF-8** format.

Steps to Confirm that the Locale is Incorrectly Set:

1. Check the `secondary-smrt-server.log` file and look for warnings that certain character translations have gone wrong.
2. Check the output of `locale` to ensure that `LANG=*.UTF-8` and that all `LC_*` variables match. `LC_ALL`, should be blank:
   ```
   smrtanalysis$ locale
   LANG="en_US.UTF-8"
   LC_COLLATE="en_US.UTF-8"
   LC_CTYPE="en_US.UTF-8"
   LC_MESSAGES="en_US.UTF-8"
   LC_MONETARY="en_US.UTF-8"
   LC_NUMERIC="en_US.UTF-8"
   LC_TIME="en_US.UTF-8"
   LC_ALL=
   ```
3. Check the encoding of the database:
   ```
   cd $SMRT_ROOT/current/bundles/smrtlink-analysisservices-gui/current/private/thirdparty/
   postgresql/postgresql_9.6.1/binwrap
   ./psql -p 9095 -h localhost -U smrtlink_user -d smrtlinkdb -c '\l'
   ```
4. Check the locale of the PG clients:
   ```
   ./psql -p 9095 -h localhost -U smrtlink_user -d smrtlinkdb -c 'SHOW client_encoding;'
   ```
5. Check the PG client autodetection when `stdout` or `stdin` is a terminal:
   ```
   ./psql -p 9095 -h localhost -U smrtlink_user -d smrtlinkdb -c 'SHOW client_encoding;' | cat
   ```
6. Check the Linux distribution:
   ```
   lsb-release -d
   ```

Solution:

Reinstall SMRT Link after setting the following environment variables:

1. First, set the environment to a valid setting.
   ```
   export LANG=en_US.UTF-8
   unset LC_ALL
   export PGCLIENTENCODING=UTF-8
   ```
2. Reinstall SMRT Link.
3. Look through the files below for things set oddly. Set the proper values in configuration files to make the solution persistent across reboots.

   ```
   /etc/locale.conf and ~/.locale.conf
   /etc/bash.bashrc, ~/.bashrc, /etc/bash.bash_profile, ~/.bash_profile
   /etc/profile, /etc/profile.d, and ~/.profile
   /etc/environment, /etc/environment.d, and ~/.environment
   ```

   If all else fails, set in `~/.bashrc` at the end because this file is parsed last.

Run QC

Recovering from a Failed Data Set Import

Symptom: Automatic import or multiple manual import attempts failed - the Data Sets **cannot** be seen in Data Management or Run QC. The failed import job was deleted by the user.

Diagnosis:

1. Restart SMRT Link Services and verify that Data Sets cannot be seen in Data Management or Run QC despite multiple services-start attempts.
2. Run `pbservice get-dataset` and inspect the job status. The job summary field should show `INACTIVE/DELETED`.

```
 *DATASET SUMMARY*
              id: 40
            uuid: ba28e17f-f6e0-4dab-9bb7-980add05f04f
            name: HG04217_LIB4_30pM-Cell4
      numRecords: 565322
     totalLength: 4960024160
           jobId: 39
             md5: fa9ab0cb4992e7abd510a8f8afc5a0d3
       createdAt: 2017-09-15T21:48:19.762Z
       updatedAt: 2017-09-21T23:16:04.012Z
            tags: subreadset
            path: /igm/runs/IGM_Seq03/sequel/r54201_20170914_145022/4_D01/
   m54201_170915_064139.subreadset.xml
         (INACTIVE/DELETED)

     *JOB SUMMARY* (INACTIVE/DELETED)
                id: 39
              uuid: f04981e4-f8c9-4298-bef3-1f712a201e89
              name: Job import-dataset
             state: SUCCESSFUL
        project id: 1
         jobTypeId: import-dataset
         is active: false
         createdAt: 2017-09-15T21:44:30.001Z
         updatedAt: 2017-09-20T20:59:47.339Z
          run time: 429318 sec
        SL version: 5.0.1.9585
        created by: admin
           comment: Importing DataSet
              path: /igm/projects1/smrtlink_userdata/jobs_root/000/000039
```

Solution:

1. Run `$SMRT_ROOT/smrtcmds/bin/dataset newuuid` on the new `subreadset.xml` file. Note that this will update the UniqueId for `MetaType="PacBio.DataSet.SubreadSet"` in the `subreadset.xml` file.
2. Import using `$SMRT_ROOT/smrtcmds/bin/pbservice import-dataset`.

   Usage:
   ```
   pbservice import-dataset --host your-smrtlink-host --port 9091 --debug /path/to/dataset.xml
   ```

Run QC Empty for a Run

Symptom: Run QC metrics are empty and no plots can be viewed.

Diagnosis: Go to the SMRT Link Data Management page and verify that a Data Set for that run does **not** exist.

Solution:

1. Manually import the data:
   - Graphical User Interface option: Go to **SMRT Link Data Management > View or Import Sequence Data > Import > Sequel Sequence Data**.

   - Command-line option:
     ```
     $SMRT_ROOT/smrtcmds/bin/pbserviceimport-dataset --host < smrtlink-host> --port 9091 --debug /
     path/to/dataset.xml
     $SMRT_ROOT/smrtcmds/bin/pbserviceimport-dataset --host <smrtlink-host> --port 9091 /opt/smrtlink/
     userdata/data_root/r6789768976/to import multiple cells from a run
     ```

2. Verify that the import was successful:
   - Graphical User Interface option: Go to the SMRT Link Run QC page and verify that metrics are no longer empty for the run of interest.

   - Command-line option: `curl http://<smrtlink-host>:9091/smrt-link/datasets/ subreads?showAll=true`

Root Cause: Run QC can be empty if a prior import attempt failed due to any of the following reasons:

- Intermittent network connectivity failure
- Intermittent `smrtlink` services down

Data Management

Relative Path Dataset XML Quick Fix

Use the following workaround for relative-path dataset XMLs which cause analysis application errors such as the following from CCS2:

```
terminate called after throwing an instance of 'std::runtime_error'
what(): could not open BAM file: .../00002/tasks/pbccs.tasks.ccs-0/
m54001_160218_185027.subreads.bam
```

Compare the `ResourceId` attribute in the following examples.

Relative-Path Dataset XML:

```
<pbbase:ExternalResource
    Description="Points to the subreads bam file."
    MetaType="PacBio.SubreadFile.SubreadBamFile"
    Name="subreads bam"
    ResourceId="m54001_160322_221501.subreads.bam"
    TimeStampedName="pacbio_subreadfile_subreadbamfile-160323_03424927"
    UniqueId="53d3ff91-0ff4-4250-8850-2011e2d22031"
    Version="3.0.1">
```

Absolute-Path Dataset XML:

```
<pbbase:ExternalResource
    Description="Points to the subreads bam file."
    MetaType="PacBio.SubreadFile.SubreadBamFile"
    Name="subreads bam"
    ResourceId="/data/runs/54001/r54001_20160322_220424/1_A01/m54001_160322_221501.subreads.bam"
    TimeStampedName="pacbio_subreadfile_subreadbamfile-160323_03424927"
    UniqueId="53d3ff91-0ff4-4250-8850-2011e2d22031"
    Version="3.0.1">
```

Relative-Path Dataset XML Quick Fix commands:

These commands will convert relative-path dataset XMLs to absolute-path dataset XMLs:

```
mv subreadset.xml subreadset.bak.xml
dataset create --type SubreadSet subreadset.xml subreadset.bak.xml
```

Subread Data Set from Instrument is not Displayed in SMRT Link After Import

Symptom: Subread Data Sets (the `subreadset.xml` file) are **not** visible in the SMRT Link graphical user interface following import.

Diagnosis: Rerun the import:

```
$SMRT_ROOT/current/bundles/smrttools/smrtcmds/bin/pbservice import-dataset --host <smrtlink_host>
--port 9091 --debug  /path/to/subreadset.xml
```

A message indicating successful import, or reporting that the Data Set is already found should be displayed.

Next, check that the web services return the Data Set by type - this is how the SMRT Link graphical user interface gets its list:

```
curl http://<smrtlink-host>:9091/secondary-analysis/datasets/subreads
```

If SMRT Link does not display the Data Set, then try the query by UUID (found in the SubreadSet XML `UniqueId` attribute.)

```
curl http://<smrtlink-host:9091>/secondary-analysis/datasets/<UniqueId>
```

If the Data Set can be found by UUID, but **not** by type `subreads`, then confirm the existence of a line defining the run name in the `subreadset.xml` file:

```
grep 'pbmeta:Name' /path/to/subreadset.xml
```

If this line does **not** exist, then follow the steps below to edit the dataset XML.

Solution: If the dataset XML is not importing correctly and the `<pbmeta:Name></pbmeta:Name>` tag is missing from the file, backup and edit the file, then follow these steps to edit and reimport the dataset XML.

Backup the `subreadset.xml` file:

```
cp /path/to/subreadset.xml /path/to/subreadset.xml.bak
```

Make the following edits to `/path/to/subreadset.xml`:

1. Change the dataset XML UUID using the PacBio-provided `dataset` tool:
   ```
   $SMRT_ROOT/smrtcmds/bin/dataset newuuid /path/to/subreadset.xml
   ```

2. Add a `<pbmeta:Name></pbmeta:Name>` tag within `<pbmeta:RunDetails></pbmeta:RunDetails>`.

   Example:
   ```
   <pbmeta:RunDetails>
           <pbmeta:Name>RUN_NAME</pbmeta:Name>
           <pbmeta:TimeStampedName>r54099_20160304_202916</pbmeta:TimeStampedName>
           <pbmeta:CreatedBy>String</pbmeta:CreatedBy>
           <pbmeta:WhenCreated>0001-01-01T00:00:00</pbmeta:WhenCreated>
           <pbmeta:WhenStarted>2016-03-04T20:35:18.118914Z</pbmeta:WhenStarted>
   </pbmeta:RunDetails>
   ```

The dataset XML can then be imported:

```
$SMRT_ROOT/current/bundles/smrttools/smrtcmds/bin/pbservice import-dataset --host <smrtlink_host>
--port 9091 --debug --log2stdout
```

SMRT Analysis

HGAP Fails with Insufficient Memory Allocation Error

Diagnosis: HGAP fails on block size errors on the `daligner` step. The `analysis.log` file may contain one of the following errors:

- `"Error: Insufficient memory allocation (16.0Gb, reduce block size or increase allocation" in the analysis.log`
- `Warning: Block size too big, index occupies more than 1/4 of desired memory allocation (16.0Gb)`

Solution: Based on the genome size and complexity, this may involve fine tuning the `falcon` configuration override parameters.

Try reducing the block size (`-s`) option by pasting the following options for `falcon` configuration override in SMRT Link's HGAP 4 **Advanced Analysis Parameters** dialog:

    pa_DBsplit_option = -x500 -s100; ovlp_DBsplit_option = -x500 -s100

Note that the defaults values in HGAP are `-x500 -s200`.

Pbalign Failure after SMRT Link v6.0.0 Upgrade

Symptom: Resequencing, SAT or HGAP analyses may fail in the `pbalign` task. The `pbalign` error happens during the BAM sorting and indexing step, where `pbalign` appears to run out of memory. This error was seen at sites where users upgraded to v6.0.0 on a **single** server (that is, using non-distributed mode) using configuration settings from the previous **distributed** installation. This issue can also appear with installations on a compute cluster, where the memory per core is less than the recommended 4 GB.

Diagnosis: The following error can be seen in the `pbsmrtpipe.log` or `master.log` file for a failed analysis. The root cause of this error results from the different configuration of the latest `samtools` version, which requires much more memory than before, and results in `pbalign` failures, especially when using more threads.

    /opt/pacbio/smrtlink/userdata/jobs_root/000/000478/tasks/pbalign.tasks.pbalign-1/stderr'.)
    Extracted from stderr
    BamPostService returned a non-zero exit status 1:

The command `samtools sort -m ${m} --threads ${nproc}` allocates `${m}* ${nproc}` memory in the beginning of the run, resulting in an out-of-memory error during BAM sorting.

Solution:
1. Replace `$SMRT_ROOT/bundles/smrttools/install/smrttools-release_6.0.0.47835/private/pacbio/pythonpkgs/pbalign/lib/python2.7/site-packages/pbalign/bampostservice.py` with the PacBio-provided `bampostservice.py` file. We recommend that you **first** create a backup of the original `bampostservice.py` file.

If the installation is on a **single-server non-cluster system**, follow the recommendations below:

2. Stop services by running `$SMRT_ROOT/admin/bin/services-stop`.
3. Run `$SMRT_ROOT/admin/bin/smrt_reconfig` to lower `NPROC` and `MAXCHUNKS` according to the number of cores on the installation host such that memory per core is **at least** 4 GB (6 GB is safer), and limit `NWORKERS` to <=2.
4. Start services by running `$SMRT_ROOT/admin/bin/services-start`.
5. Run the Site Acceptance Test application again as validation.

Structural Variant Calling Application Fails

Symptom: The Structural Variant Calling application fails for some combinations of submitted Data Sets and reference genomes where the reference contains small contigs < ~200 KB.

Diagnosis: The **Analysis Overview > Status** section of the results page displays a status of **FAILED: Task pbsv.tasks.call-0 FAILED**. The failure is due to an incompatibility in how the application chunks computations when running on a compute cluster.

Solution: Replace the `pbsv` binary with a PacBio-provided `pbsv` binary that was fixed.

1. Backup the `pbsv` binary included with the SMRT Link release:
    ```
    cp $SMRT_ROOT/smrtcmds/bin/pbsv  $SMRT_ROOT/smrtcmds/bin/pbsv.bak
    ```

2. Replace the `pbsv` binary with the PacBio-provided `pbsv` version:
    ```
    cp /path/to/updated/pbsv  $SMRT_ROOT/smrtcmds/bin/
    ```