SMRT® Link Software Installation (v5.1.0)

Introduction

This document describes the procedure for installing **SMRT Link v5.1.0** on a customer system. This document is for use by Customer IT.

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. Additionally, SMRT Link provides support for **multiple** instruments.

SMRT Link is the primary access point for applications used by researchers, laboratory technicians, instrument operators, and bioinformaticians for various interactions with applications related to the Sequel platform. 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.

Overview

1. Install or upgrade the SMRT Link software. (See “Installation Summary” on page 4 and “Configuring LDAP” on page 18 for details.)
2. (**Optional**) Configure SMRT Link to use an SSL Certificate. (See “Configuring WSO2 in SMRT Link to Use a Signed SSL Certificate” on page 22 for details.)
3. (**Optional**) Add SMRT Link Users and Assign User Roles. (See “Adding SMRT Link Users and Assigning User Roles” on page 20 for details.)

COMPUTATIONAL AND STORAGE REQUIREMENTS

HPC Configurations / HPC Components	Targeted Applications HPC*	Targeted Applications PLUS HPC**	Large-genome *de novo* HPC***
Head Node			
64 GB RAM 32 Cores			
Compute Nodes			
Cores	18 physical or 36 hyper-threaded	96 physical or 192 hyper-threaded	192 physical or 384 hyper-threaded
RAM per Node (GB)	256	256	256
Local Disk per Node	100 GB	1 TB	1 TB
Intermediate Pipeline Storage	N/A	15 TB**** serving 1800 IOPS	30 TB**** serving 1800 IOPS
Long Term Data Storage			
	10 TB	38 TB	70 - 100 TB
Network			
10 GBE recommended (1 GBE required)			

* Targeted Sequencing applications (CCS, LAA, resequencing), assembly of bacterial genomes, Iso-Seq® application. Long-term storage is calculated based on moderate usage of Sequel® System per year.

** Targeted Sequencing applications as noted above plus occasional large-genome *de novo* assemblies. Long-term storage is calculated based on moderate usage of Sequel System per year.

*** For human-scale genomes with 50-fold coverage and target assembly time approximately 72 hours. Long-term storage is calculated for one Sequel System assuming 52 human genomes per year at 50-fold coverage.

**** Non-redundant storage dedicated to this compute environment - choose from NFS, Open Source or Commercial DFS, or CIFS.

Data storage

- The **SMRT Analysis software directory** (PacBio recommends `SMRTROOT=/opt/pacbio/smrtlink`) **must** have the same path and be **readable** by the `smrtanalysis` user across **all** compute nodes via **NFS**.
- The **SMRT Cell input directory** (PacBio recommends `$SMRT_ROOT/userdata/`) **must** have the same path and be **readable** by the `smrtanalysis` user across **all** compute nodes via **NFS**. This directory contains data from the instrument, and can either be:
  - A directory configured during instrument installation, or
  - A directory created when you received data from a core lab.
- The **SMRT Analysis output directory** (PacBio recommends `$SMRT_ROOT/userdata/jobs_root/`) **must** have the same path and be **writable** by the `smrtanalysis` user across **all** compute nodes via **NFS**. This directory is usually softlinked to a large storage volume.
- The **SMRT Analysis temporary directory** is used for fast I/O operations during run time. The software accesses this directory from `$SMRT_ROOT/userdata/tmp_dir` and you can softlink this directory manually or using the install script. This directory should be a **local** directory (**not** NFS-mounted) and be writable by the `smrtanalysis` user and exist as independent directories on **all** compute nodes.

Software Prerequisites: Server Operating Systems

- SMRT Link server software is supported on English-language CentOS 6.x; 7.x and Ubuntu 14.04; 16.04 64-bit Linux® distributions.
- SMRT Link is **not** guaranteed to work on Linux versions that are no longer supported by the Operating Systems' Vendors.
- SMRT Link server software **cannot** be installed on macOS® or Windows® systems.

Software/Hardware Prerequisites: Client Operating Systems/Web Browser

To use SMRT Link on a client, PacBio recommends:

- **Any** Mac or Windows host that meets the operating system requirements **and** is connected to the network.
- Microsoft Windows 7 or later.
- macOS X 10.7 or later.
- SMRT Link requires the Google® Chrome web browser, version 56 or later.
- SMRT Link requires a minimum screen resolution of 1600 by 900 pixels.
- 64-bit Java (Version 8 or later) installed on your local Windows or Macintosh host to run SMRT View.

Network Configuration

- Please refer to the **IT Site Prep Guide** provided with your instrument purchase for more details.
- For network connectivity considerations, see the network diagram in the **Computer Requirements** section of the **IT Site Prep Guide**.

SMRT Link Server Environment Assumptions

- A dedicated 64-bit Linux host with a `libc` version greater than 2.5.
- Installing as the **same** non-root user (`SMRT_USER`) that will be used to run the system.
- The `SMRT_USER` has full permissions in the file system in the `SMRT_ROOT` directory and in all linked directories for `tmp_dir` and `userdata`, and can lock files in these file systems. (Common problems include NFS setup problems, ACLs, and so on.)
- When running in distributed mode, all other nodes have the **same path** for `SMRT_ROOT` and for all linked directories.
- During the installation, no other daemons are running on the same ports.
- PacBio **highly recommends** that the system clock be synchronized to a public NTP time server.

Security Notes

- SMRT Link (and the Sequel System) are for **research use only** (RUO) and are **not** guaranteed to be secure.
- Pacific Biosciences **recommends** that you install the SMRT Link server on networks that are only accessible to trusted users, and discourages installing SMRT Link on public networks. If you require SMRT Link to be publicly-accessible, please contact Pacific Biosciences Technical Support.
- Do **not** install SMRT Link using `root` as the user.

Installation/Upgrade Checklist

Following is a list of items you should have ready **before** starting a new installation or upgrading an existing installation:

- Full path (**without** spaces) to the shared installation root directory. Used for the main installation root; see **Installation Directories** for details.
- Job Management System settings.
- Full path (**without** spaces) to a directory on the shared file system - the `jobs_root` directory.
- **(Optional)** LDAP Settings. See “Configuring LDAP” on page 18 for details.
- **(Optional)** SSL Certificate for WSO2. See “Configuring WSO2 in SMRT Link to Use a Signed SSL Certificate” on page 22 for details.

Installation Directories

SMRT Link v5.1.0 requires **two** installation root directories. These directories are a matching pair, with similar directory structures maintained by SMRT Link.

1. One of the root directories is on a **shared file system** available to **all** compute cluster nodes; it is used for files that need to be accessible by nodes on the compute cluster. This is the main installation directory, specified on the SMRT Link installer command line. (The path **cannot** include spaces.)
2. PacBio recommends that locations on the **local file system** should be used as destinations for the `db_datadir` and `tmp_dir` symbolic links. This is due to security and performance improvements inherent to local storage, and does **not** prevent the software from functioning correctly.

Note: The contents of the shared and local installation root directories **may change in future releases**.

Installation Summary

Following are the steps for installing SMRT Link v5.1.0. (See Page 8 for sample script output and more details.)

Step	Installation Summary - SMRT Link v5.1.0

1	Log in or switch to a non-root user to run the install script on the SMRT Link host:
	`SMRT_USER=smrtanalysis`
	`su $SMRT_USER`
	Notes on $SMRT_USER:
	This is the user that will perform the installation, all upgrades/patches, and own all running processes (web services, analysis services, and `pbsmrtpipe` workflow.)
	`$SMRT_USER` should have read/write permissions for `$SMRT_ROOT` (defined in Step 3 below) and submit privileges for the secondary analysis queue configured on the job scheduler. The convention is to use `smrtanalysis`.
	Ensure that `$SMRT_USER` has read/write access to the job data directory; if new job directories cannot be written, **all** import and analysis jobs will fail. This can happen when, for example, SMRT Link was originally installed as a privileged user (root) and then switched to another account.

2	Download SMRT Link from **http://www.pacb.com/support/software-downloads/**.

3	**Define $SMRT_ROOT and run the installer:**
	`SMRT_ROOT=/opt/pacbio/smrtlink`
	Notes on $SMRT_ROOT:
	SMRT Link should be installed on a file system shared between nodes in the cluster. The convention is to use `/opt/pacbio/smrtlink` for the path. The `$SMRT_ROOT` directory must **not** exist when the installer is invoked, as the installer will attempt to first check if it exists, and will **abort** the installation without overwriting an existing location.
	`smrtlink_5.1.0.26412.run --rootdir $SMRT_ROOT`
	If a previous installation was canceled or otherwise failed, the installer can be invoked **without** re-extraction. Rerun using the `--no-extract` option:
	`smrtlink_5.1.0.26412.run --rootdir $SMRT_ROOT --no-extract`
	See "Installation Details" on page 8 for additional information.

4	**Start the SMRT Link services:** `$SMRT_ROOT/admin/bin/services-start` `$SMRT_ROOT/admin/bin/services-status` **(Optional)**
5	**Validate the installation:** `$SMRT_ROOT/admin/bin/import-canneddata` **(Optional)** The SMRT Link installation comes with PacBio-provided data, including lambda virus genome data for resequencing, HIV HXB2 reference for Minor Variant Analysis, and barcode FASTA files for demultiplexing. `import-canneddata` is optional as it is **also** invoked when you perform Step 6 (Run the Site Acceptance Test workflow.) `$SMRT_ROOT/admin/bin/run-sat-services` **or** `$SMRT_ROOT/admin/bin/run-sat-local` **Note:** `run-sat-service` runs in a **distributed** environment, and `run-sat-local` runs **locally** in non-distributed mode. The Site Acceptance Test (SAT) application is used to validate all new PacBio systems upon installation.

Step	Installation Summary - SMRT Link v5.1.0
6	**Run the Site Acceptance Test workflow:** 1. From the Chrome browser, open SMRT Link (at `http://myhostname:9090/`) and login using `admin/admin`. The port will be forwarded to a hard-coded port `8243`. Check with your IT group to ensure that this port is open. 2. Click "Yes" on the message to notify PacBio Technical Support about the successful installation of SMRT Link and scheduling the ICS upgrade. 3. Select **SMRT Analysis** from the SMRT Link home page. 4. Select **+ Create New Analysis** and type `SAT_run_postinstall` in the **Name** field. 5. In the Analysis Applications list, click **Site Acceptance Test (SAT)**. 6. Select **View: BAM Data**. 7. Select the Data Set with the **lambda/007_tiny** Run Name. 8. Select **Lambda NEB** from the references droplist. 9. Click **Start**.
7	(**Optional**) Configure LDAP. See for "Configuring LDAP" on page 18 for details.
8	**(Optional)** Configure SMRT Link to use a Signed SSL Certificate. See "SMRT® Link and SSL Certificate Procedures" on page 21 for details.

Upgrading SMRT® Link

Supported Upgrade Path

- SMRT Link upgrades **must** be performed sequentially, that is: **3.1.0 > 3.1.1 > 4.0.0 > 5.0.1 > 5.1.0**.
- You **cannot** upgrade to SMRT Link from SMRT Analysis v2.3.0 or earlier. Additionally, analysis job directories and run history from SMRT Analysis v2.3.0 or earlier are **not** compatible with SMRT Link and **cannot** be imported.

Note: You **must** set your admin login name/password back to the default `admin/admin` **before** upgrading to v5.1.0. See "Changing the Password for the Admin Account" on page 18 for details.

Step	Upgrading SMRT Link
1	**Run the Upgrader:** Upgrades are handled by the script `smrtupdater` located in `$SMRT_ROOT/admin/bin/smrtupdater`. The script performs the following: 1. Confirms the valid non-root user that will own `pbsmrtpipe` jobs and daemon processes. 2. Checks for running services, and stops them if necessary. 3. Performs hardware, OS, and software prerequisite checks. 4. Transfers the computing configurations from the previous installation. 5. Confirms and validates symbolic links to TMP and USERDATA directories. PacBio recommends running `SMRT_ROOT/admin/bin/services-stop` **before** running the upgrade. `$SMRT_ROOT/admin/bin/smrtupdater smrtlink_5.1.0.26412.run` **Note**: Stopping and restarting `smrtlink` services (for example after updating SMRT Link) may result in some running `pbsmrtpipe` jobs with a status that does **not** update, and whose underlying Python processes continue blocking. These processes may need to be terminated from the command line. For best performance, schedule updates for a time when few or no jobs are running.
2	**Start the SMRT Link services:** `$SMRT_ROOT/admin/bin/services-start` `$SMRT_ROOT/admin/bin/services-status` **(Optional)**

Step	Upgrading SMRT Link
3	**Validate the installation:** `$SMRT_ROOT/admin/bin/import-canneddata` **(Optional)** The SMRT Link installation comes with PacBio-provided data, including lambda virus genome data for resequencing, HIV HXB2 reference for Minor Variant Analysis, and barcode FASTA files for demultiplexing. `import-canneddata` is optional as it is **also** invoked when you perform Step 6 (Run the Site Acceptance Test workflow.) `$SMRT_ROOT/admin/bin/run-sat-services` **or** `$SMRT_ROOT/admin/bin/run-sat-local` **Note:** `run-sat-service` runs in a **distributed** environment, and `run-sat-local` runs **locally** in non-distributed mode. The Site Acceptance Test (SAT) application is used to validate all new PacBio systems upon installation.
4	**Run the Site Acceptance Test workflow:** 1. From the Chrome browser, open SMRT Link (at `http://myhostname:9090/`) and login using `admin/admin`. The port will be forwarded to a hard-coded port 8243. Check with your IT group to ensure that this port is open. 2. Click "Yes" on the message to notify PacBio Technical Support about the successful installation of SMRT Link and scheduling the ICS upgrade. 3. Select **SMRT Analysis** from the SMRT Link home page. 4. Select **+ Create New Analysis** and type `SAT_run_postinstall` in the **Name** field. 5. In the Analysis Applications list, click **Site Acceptance Test (SAT)**. 6. Select **View: BAM Data**. 7. Select the Data Set with the **lambda/007_tiny** Run Name. 8. Select **Lambda NEB** from the references droplist. 9. Click **Start**.

Updating the SMRT Link Chemistry Bundle

SMRT Link Chemistry Bundle updates allow updating of SMRT Link features **without** having to reinstall the SMRT Link software. The update also updates **Sequel® Instrument Control Software (ICS).**

Note: Only SMRT Link users with the **Admin** role can perform this update.

1. In SMRT Link, choose **About** from the Main menu. (A red circle indicates that a Chemistry Bundle Update is available.)
2. Click the **Update** button.
3. Clear the browser cache: Choose **More Tools > Clear browsing data**, choose **the beginning of time** from the droplist, then check **Cached images and files**. Click **Clear browsing data**.
4. Restart the browser. The SMRT Link Chemistry Bundle for this installation of SMRT Link is updated.
5. On the instrument, choose **Admin** from the Main menu. (A red circle indicates that a Chemistry Bundle Update is available.)
6. Click the **Updates** tab, then click **Install**. The instrument software then restarts, which will take around 10 minutes.

Installing only SMRT Tools

To install **only** SMRT Tools, use the `--smrttools-only` option with the installation command, whether for a new installation or an upgrade. Examples:

```
smrtlink-*.run --rootdir smrtlink --smrttools-only
smrtlink-*.run --rootdir smrtlink --smrttools-only --upgrade
```

Installation Details

Following is an example of the prompts that display when running the Install script, with added explanations:

Part 1 of 10: SMRT Analysis User

PacBio recommends that you run this script as a designated SMRT Analysis user (e.g. `smrtanalysis`) who will own all `pbsmrtpipe` jobs and daemon processes.

```
Current user is 'smrtanalysis' (primary group: Domain Users)
Use the 'smrtanalysis' as the SMRT Analysis user? [Y/n]:
Using install user (computed): smrtanalysis
Using install group (computed): smrtanalysis
Using smrtlink uuid (computed default): 00001111-aaaa-bbbb-cccc-dddddeeeeeff
```

Part 2 of 10: SMRT Link Server DNS

The DNS name of the SMRT Link host server is used for creating SMRT Link URLs (such as `http://smrtlink.example.com`), ensuring that an SSL certificate will validate. If the Domain Name System (DNS) does **not** resolve to the expected address, then an IP address must be used.

```
Detecting DNS names...
Pick an option:
  1) smrtlinkhost.somedomain.com
  2) smrtlinkhost
  3) 10.1.220.142 (eth0)
  4) Specify an alternate DNS name
Choice [1]:
Using dnsname (selected interactively): smrtlinkhost.somedomain.com
```

Part 3 of 10: SMRT Link Setup

SMRT Link requires **two** ports for proper operation. These ports **cannot** be used for listening by any other processes. The SMRT Link GUI port serves to redirect from an unencrypted http connection to the login.

page using secure https.

```
  Enter the SMRT Link GUI (http) port [9090]:
  Enter the SMRT Link Services port [9091]:
```

Memory settings must be preset for SMRT Link. By default, PacBio sets the initial (`-Xms`) and maximum (`-Xmx`) Java heap sizes to the same values.

- The default for SMRT Link services is 25% of the total memory with a maximum of 32,768 MB.
- The default for the SMRT Link GUI is 5% of the total memory with a maximum of 8,192 MB. PacBio recommends using the defaults.

```
  Enter the SMRT Link Services initial memory (in MB) [25088]:
  Enter the SMRT Link Services maximum memory (in MB) [25088]:
  Enter the SMRT Link GUI initial memory (in MB) [4992]:
  Enter the SMRT Link GUI maximum memory (in MB) [4992]:

Using GUI port (default, accepted): 9090
Using services port (computed default, accepted): 9091
Using GUI initial memory (computed default, accepted): 4992
Using GUI maximum memory (computed default, accepted): 4992
Using services initial memory (computed default, accepted): 25088
Using services maximum memory (computed default, accepted): 25088
```

Part 4 of 10: SMRT View Server Setup

SMRT View Server requires one port for the SMRT View GUI Web server. For proper operation, this port **cannot** be used for listening by any other processes. Select a port number which does **not** conflict with any other programs.

```
  Enter the SMRT View Server (http) port [9094]:
```

Memory settings must be preset for SMRT View. By default, PacBio sets the initial (`-Xms`) and maximum (`-Xmx`) Java heap sizes to the same values.

The default for SMRT View is 20% of the total memory with a maximum of 32,768 MB. PacBio recommends using the defaults.

```
  Enter the initial memory SMRT View Server (in MB) [20096]:
  Enter the maximum memory SMRT View Server (in MB) [20096]:

Using SMRT View port (computed default, accepted): 9094
Using SMRT View initial memory (computed default, accepted): 20096
Using SMRT View maximum memory (computed default, accepted): 20096
```

Part 5 of 10: Database Setup

The SMRT Link server uses a database which will need access to a network port and a directory to store database data files.

- The port will **only** be used to access the database from the install host.
- The database data directory should be located on a **local** (not shared) disk partition.
- The default location will be in the SMRT Link Local File System Root Directory, specified above.

```
Enter the SMRT Link Database port [9095]:
Enter the full path to the 'dbdatadir' directory [/path/to/smrtroot/userdata/db_datadir.default]:
  Directory '/path/to/smrtroot/userdata/db_datadir.default' does not exist.  Create it? [Y/n]:

    Creating directory '/path/to/smrtroot/userdata/db_datadir.default'...

Using SMRT Link Database port (computed default, accepted): 9095
Using Database data dir (computed default in a previous install, accepted): /path/to/smrtroot/userdata/
db_datadir.default
```

Part 6 of 10: User-Specific Directories Setup

```
The following directories should be configured to point to the actual locations:
    jobs_root
    tmp_dir
```

jobs_root: This directory stores output from SMRT Analysis and needs to be large: >15 TB.

```
Enter the full path to the 'jobs_root' directory [/path/to/smrtroot/userdata/jobs_root.default]:
Directory '/path/to/smrtroot/userdata/jobs_root.default' does not exist.  Create it? [Y/n]:

    Creating directory '/path/to/smrtroot/userdata/jobs_root.default'...
```

tmp_dir: This directory is used for fast I/O operations, and should be a local directory (**not** NFS-mounted) and needs to be large for large genome assembly jobs (>500 GB minimum, 1 TB recommended). This directory will be automatically created, as needed, on compute cluster nodes.

The directory **must** exist on each cluster node and be writable to the SMRT Link user. If missing, SMRT Link will attempt to create this destination, if permissions permit.

```
Enter the full path to the 'tmp_dir' directory [/tmp/smrtlink]: /tmp/smrtlink:
```

```
Using jobs_root (computed default, accepted): /path/to/smrtroot/userdata/jobs_root.default
Using tmp_dir (default, accepted): /tmp/smrtlink
```

Part 7 of 10: Remote Service Setup

SMRT Link provides the following services, which require connection to PacBio servers:

- SMRT Link Event Service
- SMRT Link Update Service

The **SMRT Link Event Service** can be used to send information to the PacBio Technical Support Team to troubleshoot installation and analysis failures.

The **SMRT Link Update Service** will provide automatic notification and installation of available updates to components of the current version of SMRT Link software.

The SMRT Link server **must** be allowed outbound http/https connectivity on ports 8083 and 8084 for the SMRT Link Update and Event Services.

```
Enable SMRT Link Event Service? [Y/n]:
Enable SMRT Link Update Service? [Y/n]:

Test connectivity to the remote URLs? [Y/n]:

Checking remote service URLs...
    Checking SMRT Link Event Service URL... ok
    Checking SMRT Link Update Service URL... ok

All enabled remote service URLs are available.

Using 'SMRT Link Event Service' Enable (configured interactively): true
Using 'SMRT Link Update Service' Enable (configured interactively): true
```

Part 8 of 10: SMRT Link Analysis Job Email Notification

SMRT Link can be configured to send email notifications of completed analysis jobs to the user who launched the analysis (for both successful and failed jobs).

- SMRT Link only supports connections to SMTP Relays without encryption. Servers using basic authentication, SSL/TLS, or STARTTLS are **not** supported.
- Email notification is disabled if the outgoing mail server host is empty.

```
  Enter the SMRT Link notification outgoing mail server host []: mail.somedomain.com
  Enter the SMRT Link notification mail port [25]:
  Enter the SMRT Link notification mail user []:

    No mail user specified.  Enabling SMRT Link mail notification,
    but disabling mail server authentication.

  Send a test message? [N/y]: y
    Enter target email address: []: someuser@somedomain.com

      Sending test email address to 'someuser@somedomain.com'...

      Email to 'someuser@somedomain.com' sent successfully.
      Please check email for test message.

  Keep current email notification settings? [Y/n]:
```

Part 9 of 10: Distributed Computing Setup

PacBio has validated **Sun Grid Engine (SGE), PBS**, **LSF,** and **SLURM**. You may attempt to manually configure for alternate job management systems, but these are **not** guaranteed to work.

A Job Management System may be used to dispatch jobs to a distributed compute environment. If **no** Job Management System is specified, the system will run in Non-Distributed Mode, and **all** compute jobs will be run locally on the install host. Available Job Management Systems will be detected from the PATH environment variable, but may also be selected manually.

For more information on customizing all of the Job Management Systems, edit the environment variables located in the file `$SMRT_ROOT/userdata/user_jmsenv/user.jmsenv.ish`. Note that changes to this file will apply to **every** job submitted to the cluster.

```
Auto-detected the following Job Management Systems:
SGE         (From PATH: /usr/bin)
Pick an option:
    1) SGE
    2) Other JMS
    3) None (Non-Distributed Mode)
Choice [1]:
Using jmstype (selected interactively): SGE
```

SGE Configuration

If using Sun Grid Engine, or other *GE job schedulers, select Option 1 (SGE). The install script will attempt to discover `SGE_ROOT`, `SGE_CELL`, and `SGE_BINDIR` environment variables. If these are **not** found, type them in manually when prompted. The queue name and parallel environment must also be chosen from a discovered list. If more environmental variables need to be defined, enter them in the file `$SMRT_ROOT/userdata/user_jmsenv/user.jmsenv.ish`.

```
Detecting SGE setup (locations of binaries, SGE_ROOT, SGE_CELL)...
Detected the following settings:
      SGE_ROOT=/usr/share/gridengine
      SGE_CELL=default
      SGE_BINDIR=/usr/bin
Where detected:
      SGE_ROOT     (from 'SGE_ROOT environment variable')
      SGE_CELL     (from 'SGE_CELL environment variable')
      SGE_BINDIR   (from 'PATH environment variable, default')

Are these correct [Y/n]:
Using the following settings:
      SGE_ROOT=/usr/share/gridengine
      SGE_CELL=default
      SGE_BINDIR=/usr/bin
Select the queue to use for SMRT Analysis jobs:
Pick an option:
   1) default
   2) fast
   3) bigmem
Choice [1]: 3
Select the parallel environment to use for SMRT Analysis jobs:
Pick an option:
   1) smp
   2) mpi
```

```
Choice [1]:
Additional arguments to the SGE job submission command may be added in SGE_STARTARGS. The default job submission
command is:
   qsub -S /bin/bash -sync y -V -q ${QUEUE} -N ${JOB_NAME} \
        -o ${STDOUT_FILE} -e ${STDERR_FILE} \
        -pe ${PE} ${NPROC} ${CMD}

Specify extra 'qsub' args, SGE_STARTARGS []:

Using SGE_ROOT (SGE_ROOT environment variable): /usr/share/gridengine
Using SGE_CELL (SGE_CELL environment variable): default
Using SGE_BINDIR (PATH environment variable, default): /usr/bin
Using SGE_QUEUE (selected interactively): bigmem
Using SGE_PE (selected interactively): smp
Using SGE_STARTARGS (default, accepted):
Using use_settings_file flag (computed): false

Checking setting for the SGE SMRTAnalysis queue (bigmem, smp)...
Checking that the queue is valid...
Checking that the pe is valid...
Checking that the pe is in the queue pe_list...
Checking the pe allocation_rule...
Checking the queue hostlist...
```

PBS Configuration

If using the PBS Job Scheduler, select Option 1 (PBS). The install script will attempt to discover `PBS_BINDIR` and `PBS_QUEUE` environment variables. If these are **not** found, type them in manually when prompted. If more environmental variables need to be defined, enter them in the file `$SMRT_ROOT/userdata/user_jmsenv/user.jmsenv.ish`.

```
Auto-detected the following Job Management Systems:

PBS         (From PATH: /opt/pbs/bin)

Pick an option:
      1) PBS
      2) Other JMS
      3) None (Non-Distributed Mode)
Choice [1]:

Using jmstype (selected interactively): PBS

Detected the following settings:
      PBS_BINDIR=/opt/pbs/bin
      PBS_QUEUE=     (Use PBS default queue, currently: 'defqueue') PBS_STARTARGS=
Where detected:
      PBS_BINDIR        (from 'PATH environment variable, default')
      PBS_QUEUE         (from 'default')
      PBS_STARTARGS     (from 'default')

Are these correct [Y/n]: n
Specify PBS_BINDIR [/opt/pbs/bin]:

Select PBS_QUEUE:
Pick an option:
      1) defqueue      (PBS default queue)
```

```
      2) batch
      3) --NONE--      (Use PBS default queues, currently 'defqueue')
Choice [3]: 2


Additional arguments to the PBS job submission command may be added in PBS_STARTARGS. The default job submission
command is:
      qsw ${CMD} -S /bin/bash -V -q ${QUEUE} -N ${JOB_NAME} \
      -o ${STDOUT_FILE} -e ${STDERR_FILE} -l nodes=1:ppn=${NPROC} -PBS


Specify extra 'qsub' args, PBS_STARTARGS []:


Use the existing start and stop commands (i.e. qsw, qdel)? [Y/n]:


Using jmstype (selected interactively): PBS
Using PBS_BINDIR (PATH environment variable, default, accepted): /opt/pbs/bin
Using PBS_QUEUE (configured interactively): batch
Using PBS_STARTARGS (default, accepted):
Using PBS start cmd (default): qsw
Using PBS stop cmd (default): qdel
```

LSF Configuration

If using the LSF Job Scheduler, select Option 1 (LSF). The install script will attempt to discover `LSF_BINDIR`, `LSF_SERVERDIR`, `LSF_LIBDIR`, and `LSF_ENVDIR` environment variables. If these are **not** found, type them in manually when prompted. If more environmental variables need to be defined, enter them in the file `$SMRT_ROOT/userdata/user_jmsenv/user.jmsenv.ish`.

```
Auto-detected the following Job Management Systems:
LSF         (From PATH: /opt/lsf/bin)


Pick an option:
      1) LSF
      2) Other JMS
      3) None (Non-Distributed Mode)
Choice [1]:


Using jmstype (selected interactively): LSF


Detected the following settings:
      LSF_BINDIR=/opt/lsf/bin
      LSF_QUEUE=   (Use default queue, currently: 'defqueue')
      LSF_STARTARGS=
Where detected:
      LSF_BINDIR        (from 'PATH environment variable, default')
      LSF_QUEUE         (from 'default')
      LSF_STARTARGS     (from 'default')


Are these correct [Y/n]: n
Specify LSF_BINDIR [/opt/lsf/bin]:


Select LSF_QUEUE:
Pick an option:
      1) defqueue      (LSF default queue)
      2) normal
      3) interactive
      4) longrun
```

```
      5) --NONE--      (Use LSF default queues, currently 'defqueue')
Choice [5]: 2

Additional arguments to the LSF job submission command may be added in LSF_STARTARGS. The default job submission
command is:
      bsub -K -J ${JOB_NAME} -o ${STDOUT_FILE} -e ${STDERR_FILE} \
      -n ${NPROC} -q ${QUEUE} -R "span[hosts=1]" ${CMD}

Specify extra 'bsub' args, LSF_STARTARGS []:

Using jmstype (selected interactively): LSF
Using LSF_BINDIR (PATH environment variable, default, accepted): /opt/lsf/bin
Using LSF_QUEUE (configured interactively): normal
Using LSF_STARTARGS (default, accepted):
```

SLURM Configuration

If using the SLURM Job Scheduler, select Option 1 (SLURM). The install script does **not** attempt to discover environment variables. To configure SLURM, you **must** assign the following environment variables in the file `$SMRT_ROOT/userdata/user_jmsenv/user.jmsenv.ish`:

- `BINDIR` should be set to the directory that contains the `srun` and the `salloc` SLURM binaries.
- `PARTITION` is basically what is known as the "queue" in other JMS systems.
- `PRESTARTARGS` are for any additional arguments to `salloc`.
- `STARTARGS` are for any additional arguments to `srun`.

```
Auto-detected the following Job Management Systems:
Slurm       (From PATH: /opt/slurm/bin)

Pick an option:
      1) Slurm
      2) Other JMS
      3) None (Non-Distributed Mode)
Choice [1]:

Using jmstype (selected interactively): Slurm

Detected the following settings:
      SLURM_BINDIR=/opt/slurm/bin
      SLURM_PARTITION=     (Use Slurm default partition, currently: 'defqueue')
      SLURM_PRESTARTARGS=
      SLURM_STARTARGS=
Where detected:
      SLURM_BINDIR        (from 'PATH environment variable, default')
      SLURM_PARTITION     (from 'default')
      SLURM_PRESTARTARGS  (from 'default')
      SLURM_STARTARGS     (from 'default')

Are these correct [Y/n]: n
Specify SLURM_BINDIR [/opt/slurm/bin]:

Select SLURM_PARTITION:
Pick an option:
      1) defqueue      (Slurm default partition)
      2) normal
```

```
      3) fast
      4) long
      5) --NONE--      (Use Slurm default partition, currently 'defqueue')
Choice [5]: 2

Additional arguments to the Slurm job submission command may be added in SLURM_PRESTARTARGS (for salloc) and
SLURM_STARTARGS (for srun. The default job submission command is:

      salloc --jobname"${JOB_NAME} --nodes=1 --cpus-per-task=${NPROC} \
             --partition=${PARTITION} \
      srun   --cpus-per-task=${NPROC} \
             --ntasks 1 -o ${STDOUT_FILE} -e ${STDERR_FILE} \
             --partition=${PARTITION} ${CMD}

Specify extra 'salloc' args, SLURM_PRESTARTARGS []:
Specify extra 'srun' args, SLURM_STARTARGS []:

Using SLURM_BINDIR (PATH environment variable, default, accepted): /opt/slurm/bin
Using SLURM_PARTITION (configured interactively): normal
Using SLURM_PRESTARTARGS (default, accepted):
Using SLURM_STARTARGS (default, accepted):
```

For Other JMS Configurations:

When **no** JMS is automatically detected, select Option 2 (Other JMS).

```
Auto-detected the following Job Management Systems:
NONE        (No JMS Detected)
Pick an option:
      1) None (Non-Distributed Mode)
      2) Other JMS
Choice [1]: 2

Pick an option:
      1) SGE
      2) OGS
      3) UGE
      4) PBS
      5) TORQUE
      6) PBSPro
      7) LSF
      8) OpenLava
      9) Slurm
      10) OtherJMS    (Other/Unrecognized Third Party JMS)
      11) CustomJMS   (Custom JMS)
      12) None        (Non-Distributed Mode)
Choice [12]: 10

Using jmstype (selected interactively): OtherJMS__*

Detected the following settings:
      OTHERJMS_NAME=
      OTHERJMS_BINDIR=
      OTHERJMS_QUEUE=
      OTHERJMS_STARTARGS=
where detected:
```

```
      OTHERJMS_NAME          (from 'default')
      OTHERJMS_BINDIR        (from 'default')
      OTHERJMS_QUEUE         (from 'default')
      OTHERJMS_STARTARGS     (from 'default')

Could not determine OTHERJMS_NAME setting, please specify below.

Specify OTHERJMS_NAME []: prun
Specify OTHERJMS_BINDIR []: /opt/prun/bin
Select OTHERJMS_QUEUE: []: normal
Specify extra args, OTHERJMS_STARTARGS []:

Using OTHERJMS_NAME (configured interactively): prun
Using OTHERJMS_BINDIR (configured interactively): /opt/prun/bin
Using OTHERJMS_QUEUE (configured interactively): normal
Using OTHERJMS_STARTARGS (default, accepted):
```

Part 10 of 10: Distributed Computing Configuration Setup

Configure the following options for distributed computing:
`NWORKERS, NPROC, TOTAL_NPROC, CHUNKING, MAXCHUNKS`

NWORKERS: Specifies the maximum number of simultaneous `pbsmrtpipe` jobs that can be run by the SMRT Link server. This should be set to **no more** than the number of processors available on the SMRT Link server machine. The default is the minimum of 32 and number of processors on the system.

```
  Enter the max number of workers 'NWORKERS' [24]:
```

NPROC: Specifies the maximum number of slots available per task on each compute node. The suggested value is determined by the processor count of the SMRT Link system (assuming 1 core per slot), but should be set no greater than the lowest slot count on the available compute nodes.

```
  Enter the number of processors per task 'NPROC [23]:
```

TOTAL_NPROC: Specifies the maximum number of total processors/slots a `pbsmrtpipe` job will use concurrently. This value **only** has significance if set lower than the product of NCHUNKS and NPROC.

```
  Enter the total number of processors 'TOTAL_NPROC' [1000]:
```

CHUNKING: Specifies whether large files should be broken up into smaller chunks.

```
  Enable chunking 'CHUNKING' [Y/n]:
```

MAXCHUNKS: Specifies the maximum number of chunks when breaking up large files.

```
  Enter the max number of chunks 'MAXCHUNKS' [24]:

Using NWORKERS (computed default, accepted): 24
Using PROC (computed default, accepted): 23
Using TOTAL_NPROC (default, accepted): 1000
Using CHUNKING (configured interactively): true
Using MAXCHUNKS (default, accepted): 24

End of Log Output

Saving config...
```

```
Applying settings...
  Applying jms settings....
Creating user.jmsenv.ish file...
    Generating jmsenv.ish file...
    Generating jms template files...
  Applying dirlinks settings....
  Applying smrtslag preset.xml settings....
  Applying smrtslag config.json settings....
  Installing smrtlink-system-config.json...
  Validating smrtlink-system-config.json...
  Applying database settings....
  Running smrtlink-analysisservices-gui apply-config...
  Running smrtlink-analysisservices-gui upgrade...
Successfully Completed apply-config
Successfully completed running smrtflow.tools.apply_config 0.3.0 (smrtflow 0.10.0+19302.ffa7e32) in 1 sec.
  Applying smrtview settings....
  Running smrtview apply-config...

SMRT Link Install successful.
```

User-specific configurations can be injected into JMS commands in two ways:

1. During the installation or reconfiguration, specify the following extra `qsub` arguments:
   `SGE_STARTARGS []: "-l mem_free=2G, h_rt=120:0:0"`.
   Note that if performing a reconfiguration, the following steps are **required** to apply the changes:

   A) `$SMRT_ROOT/smrtlink/admin/bin/services-stop`

   B) `$SMRT_ROOT/smrtlink/admin/bin/smrt_reconfig`. This will regenerate the configuration files **without** performing a complete reinstall of the software.

   C) `$SMRT_ROOT/smrtlink/admin/bin/services-start`. New configuration settings will be automatically applied following a restart of SMRT Link Services.

2. By adding the configurations to `$SMRT_ROOT/smrtlink/userdata/user_jmsenv/user.jmsenv.ish`.
   Example: To specify resource request list for `mem_free` and `h_rt` to SGE `qsub`, define `qsuboptions` in the variable `SGE_STARTARGS`.

   To do so, 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"`

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

LDAP Integration

SMRT Link supports integration with LDAP for user login authentication. **Without** LDAP integration with SMRT Link, only **one** user (with the login `admin/admin`) is enabled.

If you are interested in configuring SMRT Link integration with your organization's LDAP, PacBio recommends that you consult your LDAP administrator to help determine the correct LDAP settings.

Note: Since SMRT Link v4.0.0, existing LDAP configurations are **automatically** migrated during upgrade.

Changing the Password for the Admin Account

Note: Do **not** change the `admin` password for the `admin/admin` account by **only** using the WSO2 API Manager. Please follow these steps carefully, otherwise you will **not** be able to access the built-in `admin` account.

1. Open **WSO2 API Manager** and login as `admin`.
2. Under **Users and Roles**, Click **List > Users > Change Password for admin user**.
3. Click **Sign-out**, and shut down SMRT Link using `$SMRT_ROOT/admin/bin/services-stop`
4. Change the password in the following files:

   Line 26 in `$SMRT_ROOT/current/bundles/smrtlink-analysisservices-gui/current/private/pacbio/smrtlink-analysisservices-gui/wso2am-2.0.0/repository/conf/user-mgt.xml`

   Lines 19 and 21 in `$SMRT_ROOT/current/bundles/smrtlink-analysisservices-gui/current/private/pacbio/smrtlink-analysisservices-gui/wso2am-2.0.0/repository/conf/jndi.properties`

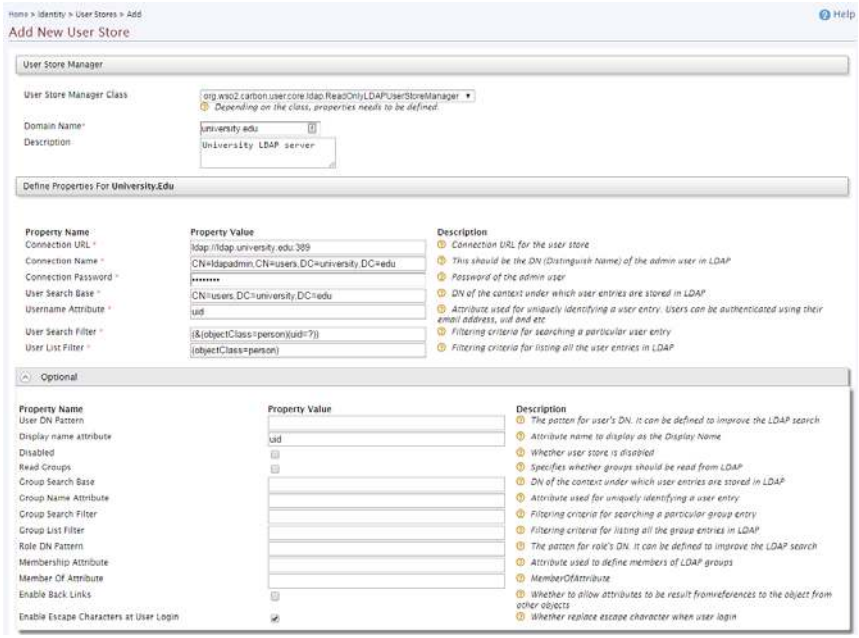
5. Enter `$SMRT_ROOT/admin/bin/set-wso2-creds --user 'admin' --password 'newpassword'`
6. Start SMRT Link services again using `$SMRT_ROOT/admin/bin/services-start`

Configuring LDAP

- LDAP is configured **after** SMRT Link v5.1.0 is installed, using the **WSO2 API Manager** software, as shown below.
- You must **first** configure LDAP **before** you can enable a network user to be a SMRT Link user, and specify their role.

1. Enter the following in your browser: `https://<hostname>:9443/carbon/` where `<hostname>` is the host where SMRT Link is installed.
2. Login using `admin/admin`.
3. Click **User Stores > Add**.

4. Edit the fields as necessary for your site.



The following fields are **required**. (**Note**: Values provided in the example above are listed below for clarity. Actual values should be provided by your LDAP administrator):

- User Store Manager Class: `org.wso2.carbon.user.core.ldap.ReadOnlyLDAPUserStoreManager`
- Domain Name: `university.edu`
- Connection URL: `ldap://ldap.university:389`
- Connection Name: `CN=ldapadmin,CN=users,DC=university,DC=edu`
- Connection Password: `<password>`
- User Search Base: `CN=users,DC=university,DC=edu`
- Username Attribute: `uid`
- User Search Filter: `(&(objectClass=person)(uid=?))`
- User List Filter: `(objectClass=person)`
- Display name attribute: `uid`

For more information on LDAP, consult the following web pages:

https://en.wikipedia.org/wiki/Lightweight_Directory_Access_Protocol
https://en.wikipedia.org/wiki/LDAP_Data_Interchange_Format
https://msdn.microsoft.com/en-us/library/ms677605%28v=vs.85%29.aspx

Problems with the LDAP server may be debugged by looking at the log file located here:

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

SMRT® Link User Roles

SMRT Link provides three user roles: **Admin**, **Lab Tech**, and **Bioinformatician**. These have different user privileges, as shown below:

Tasks/Privileges	Admin	Lab Tech	Bioinformatician
Add/Delete SMRT Link Users	Y	N	N
Assign roles to SMRT Link users	Y	N	N
Update SMRT Link software	Y	N	N
Access Sample Setup Module	Y	Y	N
Access Run Design Module	Y	Y	N
Access Run QC Module	Y	Y	Y
Access Data Management Module	Y	Y	Y
Access SMRT Analysis Module	Y	Y	Y

PacBio recommends the following role assignments:

- Assign **at least** one user per site the **Admin** role. That individual is responsible for enabling and disabling SMRT Link users, as well as specifying their roles. The **Admin** can also access all SMRT Link modules, as well as every file in the system. (SMRT Link supports **multiple** users with the **Admin** role per site.)
- Assign users who work in the lab preparing samples and performing runs the **Lab Tech** role. **Lab Tech** can also access all SMRT Link modules.
- Assign users who work **only** on data analysis the **Bioinformatician** role. **Bioinformatician** can **only** access the Run QC, Data Management and SMRT Analysis modules; this is the lowest access level.

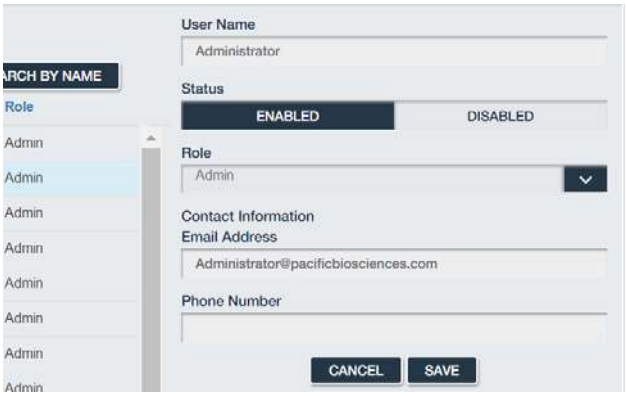
Adding SMRT Link Users and Assigning User Roles

- You must **first** configure LDAP **before** you can manage users and assign SMRT Link roles to users.
- After LDAP is configured, if you do **not** assign a SMRT Link role to a user, that user will **not** be able to login to SMRT Link.

1. Access **SMRT Link**: Enter `http://<hostname>:9090`, where `<hostname>` is the host where SMRT Link is installed.
2. Choose **Configure** from the SMRT Link menu and click **User Management**.
3. There are 2 ways to find users:

- **To display all SMRT Link users**: Click **Display all Enabled Users**.
- **To find a specific user**: Enter a user name, or partial name and click **Search By Name**.

4. Click the desired user. If the Status is **Enabled**, the user has access to SMRT Link; **Disabled** means the user **cannot** access SMRT Link.
   - To **add** a SMRT Link user: Click the **Enabled** button, then assign a role. (See Step 5.)
   - To **delete** a SMRT Link user: Click the **Disabled** button.
5. Click the **Role** field and select one of the three roles. (A **blank** role means that this user **cannot** access SMRT Link.)
6. Click **Save**. The user now has access to SMRT Link, based on the role just assigned.



SMRT® Link and SSL Certificate Procedures

SMRT Link v5.1.0 uses SSL (Secure Sockets Layer) to enable access via HTTPS (HTTP over SSL), so that your SMRT Link logins and data are encrypted during transport to and from SMRT Link. SMRT Link includes an Identity Server, which can be configured to integrate with your LDAP/AD servers and enable user authentication using your organizations' user name and password. To ensure a secure connection between the SMRT Link server and your browser, the SSL Certificate can be installed **after** completing SMRT Link installation.

It is important to note that PacBio will **not** provide a Signed SSL Certificate. However, once your site has obtained a Signed SSL Certificate, PacBio's tools can be used to install it and configure SMRT Link to use it. (**Note**: PacBio recommends that you consult your IT administrator about obtaining an SSL Certificate.) You will need a certificate issued by a Certificate Authority (CA, sometimes referred to as a "certification authority"). PacBio has tested SMRT Link with certificates from the following certificate vendors: VeriSign, Thawte and DigiCert.

If your site does **not** provide an SSL Certificate, SMRT Link v5.1.0 will use a PacBio self-signed SSL Certificate. If you use the self-signed SSL Certificate, **each** user will need to accept the browser warnings related to access in insecure environment. You can also have your IT administrator configure desktops to **always trust** the provided self-signed Certificate. Note that SMRT Link is installed within your organization's secure network, behind your organization's firewall.

See "Using SMRT Link with a PacBio Self-Signed SSL Certificate" on page 25 for details on how to handle the security warnings when accessing SMRT Link.

Use the following procedures **only** if your site provides an SSL Certificate. These procedures are **not** applicable if you are using PacBio's Self-Signed SSL Certificate.

Note: If you have **already** setup an SSL Certificate in SMRT Link v4.0.0, those settings will be carried over **automatically** when upgrading to SMRT Link v5.1.0.

Prerequisites

Please consult your system administrator if you need the following programs installed:

1. `openssl`: This common package is available on all of the major distributions through their package installers.
2. `keytool`: Part of the standard Java Runtime.

Note: If you already have a complete `.jks` file, including the signed certificate, see "Installing an Existing Certificate" on page 24. Note that the `.jks` file needs to be generated using the appropriate password/alias.

Configuring WSO2 in SMRT Link to Use a Signed SSL Certificate

Deploying a signed SSL certificate to avoid the browser warning when using SMRT Link requires updates to the third-party WSO2 API Manager software, which handles authentication and manages user rights and roles. This process requires several steps, most of which need to be run on the command line:

1. Purchase the signed certificate from the certificate authority.
2. Generate a private/public key pair and a Certificate Signing Request (CSR).
3. Create a new `truststore` file that includes the new key.
4. Update the WSO2 configuration to use the signed certificate key. (This part is automated using the included script `install_ssl_cert.sh`.)
5. If LDAP has already been configured, reenter the password in the WSO2 Administrative interface.

Key file and Certificate Signing Request (CSR) Generation

For clarity, this document uses variables for some of the subsequent steps:

```
SMRT_ROOT="/path/to/smrtlink"
FQDN="hostname.domain.com"
KEYPW="password"
KEYNAME="hostname_domain_com"
KEYSTORE="${KEYNAME}.jks"
TRUSTSTORE="client-trustore.jks"
```

Edit `FQDN` and `KEYPW` as appropriate for your site.

Step 1: Generate a certificate-signing request (`.csr`) and a keystore (`.jks`) file.

At PacBio, we used DigiCert to obtain certificates for our internal SMRT Link servers. We also provide a tool to generate the appropriate command for creating the key and keystore files. If your organization's chosen certificate authority does **not** provide this information, below is an example of the commands necessary:

```
$ keytool -genkey -alias server -keyalg RSA -keysize 2048 -keystore hostname_domain_com.jks -dname
"CN=hostname.domain.com, O=Company Name, L=City, ST=CA, C=US"

$ keytool -certreq -alias server -file hostname_domain_com.csr -keystore hostname_domain_com.jks

Enter keystore password: <password>

Re-Enter new password: <password>

Output hostname_domain_com.csr,

hostnamet_domain_com.jks (private key)
```

Notes on Passwords and Aliases

Many of the following steps require entry of a new password. This password:

- **Must** be the **same** as the one entered at the command-line for WSO2.
- Should be **unique** for this purpose and **not** be reused anywhere else, although it may be shared across multiple `smrtlink` instances that use the same SSL certificate.
- Should be **non-obvious**; note that it will be stored in plain text in multiple configuration files.

The **alias** needs to be explicitly set to `server` everywhere for the certificate configuration.

Now that you have the certificate-signing request, you can use it to request a new certificate from DigiCert or any other certificate authority. If using DigiCert, the **SSLPlus** product is suitable for a single SMRT Link instance, but wild card certificates for an entire domain are available at a higher price. This process is very fast.

Step 2: Download the new certificate in `.p7b` format from the DigiCert website `hostname_domain_com.p7b`.

Step 3: Combine the certificate and the keystore files:

```
$ keytool -import -trustcacerts -alias server -file ${KEYNAME}.p7b -keystore ${KEYNAME}.jks
Enter keystore password:
Certificate reply was installed in keystore
```

Step 4: Generate an intermediate file in `.pem` format:

```
$ keytool -export -alias server -keystore ${KEYNAME}.jks -file ${KEYNAME}.pem
Enter keystore password:
Certificate stored in file <hostname_nanofluidics_com.pem>
```

Step 5: Generate the WSO2 truststore `client-truststore.jks` file using the `.pem` file:

```
$ keytool -import -alias server -file ${KEYNAME}.pem -keystore client-truststore.jks -storepass
$KEYPW

<Miscelaneous keytool output>

Trust this certificate? [no]:  y
Certificate was added to keystore
```

Step 6: Stop the services by entering `${SMRT_ROOT}/admin/bin/services-stop`.

Step 7: Install the new `.jks` files and update the configuration files:

```
${SMRT_ROOT}/admin/bin/install_ssl_cert.sh ${FQDN} ${KEYSTORE} ${TRUSTSTORE} ${KEYPW}
```

This script will install a signed SSL certificate to SMRT Link, removing the browser warnings that occur when using the default certificate. To run this script, you will need two files in Java Key Store (`.jks`) format:

- One containing the SSL keys and certificate.
- A separate `client-truststore.jks` required by the authentication manager.

Usage: `install_ssl_cert.sh $FQDN $KEYSTORE $TRUSTSTORE $KEYPW` where:

- `$FQDN` is the fully-qualified domain name appropriate to the signed SSL Certificate, such as `smrtlink.university.edu.`
- `$KEYSTORE` is the path to the keystore file generated from the SSL Certificate (`.jks` extension); this will be copied to the SMRT Link installation.
- `$TRUSTSTORE` is the path to `client-truststore.jks.`
- `$KEYPW` is the password used for generating keys.

The FQDN must match the `dnsname` specified in the installer The shorthand (such as "`smrtlinkhost`") will **not** work because the certificate is for a domain name, **not** an unqualified hostname. When running the SMRT Link installer, do this by passing the arguments `--dnsname $FQDN`.

Also note that if you are using LDAP authentication, you may need to reenter the password for the LDAP connection in the WSO2 administrative interface once SMRT Link has been started again, for example: `https://smrtlink.pacb.com:9443/carbon.`
This is because the password is stored encrypted with the SSL certificate key, which has now changed.

Step 8: Start SMRT Link services by entering `$SMRT_ROOT/admin/bin/services-start`.

Step 9: Final Check:

Go to `http://hostname:9090` and login as `admin/admin` (if LDAP is not enabled). Note that SSL is **not** used on the UI port (i.e. 9090) because this only serves static content; the actual login credentials are sent to port 8243 which only uses SSL.

You will be redirected to `https://hostname.domain.com:8243/sl/#/welcome`, and should see a padlock sign in front of the URL which indicates that the site is secure.

Viewing a Java Keystore File

The keystore files for SSL certificates are binary files. Use the following command to verify if the same password was used in the SSL certificate generation and install process. If the same password was **not** used in the certificate installation process, this command will give an error. To list the contents of a Java keystore file, use the `keytool -list` command, as shown below:

Usage: `keytool -list -v -keystore keystore.jks`

Example: `keytool -list -v -keystore smrtlink-test_nanofluidics_com.jks`

```
Enter keystore password:
 Keystore type: JKS
 Keystore provider: SUN
 Your keystore contains 1 entry
 Alias name: server
 Creation date: Feb 13, 2017
 Entry type: PrivateKeyEntry
 Certificate chain length: 3
 Certificate[1]:
 Owner: CN=smrtlink-release-test.nanofluidics.com, O="Pacific Biosciences of California, Inc.",
L=Menlo Park,   ST=CA, C=US
 Issuer: CN=DigiCert SHA2 Secure Server CA, O=DigiCert Inc, C=US
```

Errors/logs related to certificate installation can be found here:

```
$SMRT_ROOT/current/bundles/smrtlink-analysisservices-gui/current/private/pacbio/smrtlink-
analysisservices-gui/
```

Installing an Existing Certificate

If you **already** have a complete `.jks` file (suitable for Apache Tomcat, for example), including the signed certificate, you just need to change the alias of the keystore/certificate to `server` using the `keytool` command (`-keyclone` or `-changealias` subcommands).

Set the password to whatever you will supply to the install script in SMRT Link. Then, follow the instructions in "Adding the public key to client-truststore.jks" in **https://docs.wso2.com/display/IS500/Creating+New+Keystores** again with the same changes.

If you already have the SSL key in a `.jks` file and have obtained a certificate for this key in PKCS #7 Certificate format (`.p7b`), run this command to combine them:

```
$ keytool -import -trustcacerts -alias server -file star.university.edu.p7b -keystore
star.university.edu.jks
```

Then follow the instructions above to generate the `client-truststore.jks` file, and run the install script.

Recovering from the SSL Certificate Installation

It may sometime be necessary to uninstall the user-provided SSL certificate and restore the default certificate. The following steps will revert changes made by `$SMRT_ROOT/admin/bin/install_ssl_cert.sh`:

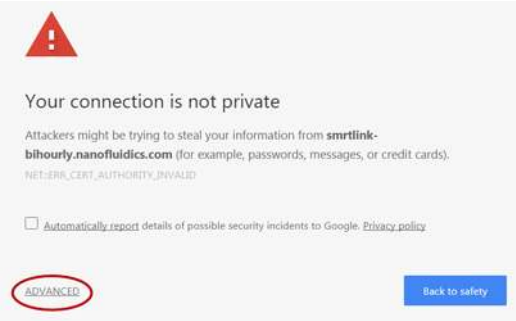
1. Stop SMRT Link services:
   `$SMRT_ROOT/admin/bin/services-stop`
2. Check that all SMRT Link processes have terminated by running `ps -ef | grep smrtlink`. Remaining processes should be terminated with kill <PID> or kill -9 <PID>.
3. Restore backup settings:
   ```
   cd ${SMRT_ROOT}/current/bundles/smrtlink-analysisservices-gui/current/private/pacbio/smrtlink-analysisservices-gui/wso2am-2.0.0/repository
   mv conf conf.new
   mv conf.orig conf
   mv resources/security/client-truststore.jks.orig resources/security/client-truststore.jks
   ```
4. Start SMRT Link services:
   `${SMRT_ROOT}/admin/bin/services-start`

Using SMRT Link with a PacBio Self-Signed SSL Certificate

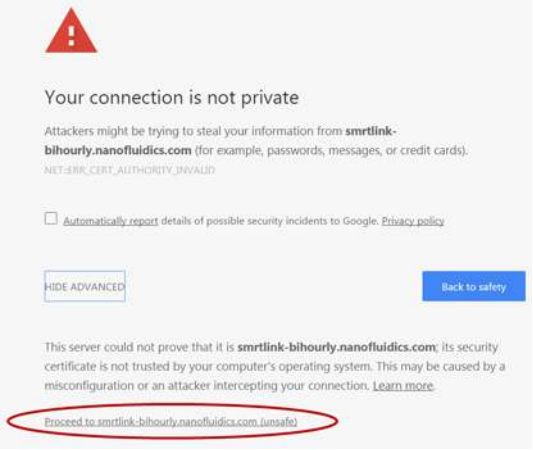
SMRT Link v5.1.0 ships with a PacBio self-signed SSL Certificate. If your site does **not** have a Signed SSL Certificate **and** you use the self-signed SSL Certificate, **each** user will need to accept the browser warnings related to access in insecure environment. You can also have your IT administrator configure desktops to **always trust** the provided self-signed Certificate. Note that SMRT Link is installed within your organization's secure network, behind your organization's firewall.

Security messages display when users try to login to SMRT Link for the **first time** using the Chrome browser. These messages may also display **other times** when accessing SMRT Link. **Each** SMRT Link user in your organization should address these browser warnings following the procedure below.

1. The first time you start SMRT Link after installation, you see the following text. Click the **Advanced** link.



2. Click the **Proceed...** link. (You may need to scroll down.)



3. Close the window by clicking the **Close** box in the corner.

4. The **Login** dialog displays, where you enter the User Name and Password. The next time you access SMRT Link, the Login dialog displays **directly**.

Client Software: Installing 64-bit Java 8 to run SMRT® View

SMRT View is a genome browser that visualizes sequencing data generated by the Sequel System.

- **Note**: To run SMRT View, 64-bit Java (Version 8 or later) **must** be installed on your local Windows or Macintosh host.

Installing 64-bit Java 8 on Windows

1. Use **Control Panel > Programs and Features** to check for and uninstall **all** existing versions of Java.
2. Go to http://www.java.com/en/download/manual.jsp.
3. Click **Windows Offline (64-bit)**. This downloads a `x64.exe` file. (**Note**: Other Java versions are 32-bit, and will **not** work with SMRT View.)
4. Double-click the `.exe` file to start the Java installer, and follow the installer directions.
5. After the installation is finished, restart the browser.

Installing 64-bit Java 8 on macOS

Note: This requires macOS 10.7.3 or later.

1. Use the Finder to search for **all** existing versions of Java, then drag them to the Trash to uninstall.
2. Go to http://www.java.com/en/download/manual.jsp.
3. Click **Mac OS X**. This downloads a `x64.dmg` file.
4. Double-click the `.dmg` file to mount the installer volume.
5. Double-click the Java icon to start the Java installer, and follow the installer directions.
6. After the installation is finished, restart the browser.

Importing Data into SMRT® Link

If you have a Sequel System installed and it is linked to the SMRT Link software during the instrument installation, your Sequel System data will be **automatically** imported in SMRT Link.

You can **manually** import the following types of files directly, using the SMRT Link GUI:

- **Reference sequence files** - FASTA files containing a reference sequence.
- **Sequel sequence data** - A file (`.subreads.xml`) containing information about Sequel sequence data, such as paths to the BAM files.
- **RS II sequence data** - A file (`.metadata.xml`)containing information about PacBio RS II sequence data from one cell.
- **Barcodes**/**Barcodes (FASTA)** - `.xml` or FASTA-format files containing barcodes.

You can also import data in SMRT Link using the `pbservice` command-line utility, as shown below.

- The host and port for the Analysis Services are optional and default to `localhost:9090`. You can change these settings using the `--host` and `--port` arguments.

Importing	Commands

BAM Data Sets Generated by the Sequel System	**Import individual SubreadSet XML files:**
	`$> pbservice import-dataset --host $HOST --port $PORT /path/to/subreads.subreadset.xml`
	Import a directory of SubreadSet XML files:
	`$> pbservice import-dataset --host $HOST --port $PORT /path/to/tree/containing/subreadssets.xml/`

PacBio RS II Data created with SMRT Analysis versions prior to v3.0.0	**Import a Dataset XML file (Subreads, reference sequences, or barcode files):** `$> pbservice import-dataset --host smrtlink-release --port 9091 /path/to/dataset.xml`
A FASTA Reference	**Creating a ReferenceSet XML file from a FASTA file:** `fasta-to-reference hg38.fasta /opt/smrtlink/references hg38 --organism Homo_sapiens > fasta2ref.log 2>&1`

P/N 100-749-900 Version 15 (February 2018)