

BioMS Admin Guide

BioMS 1.0 Setup and Admin Guide

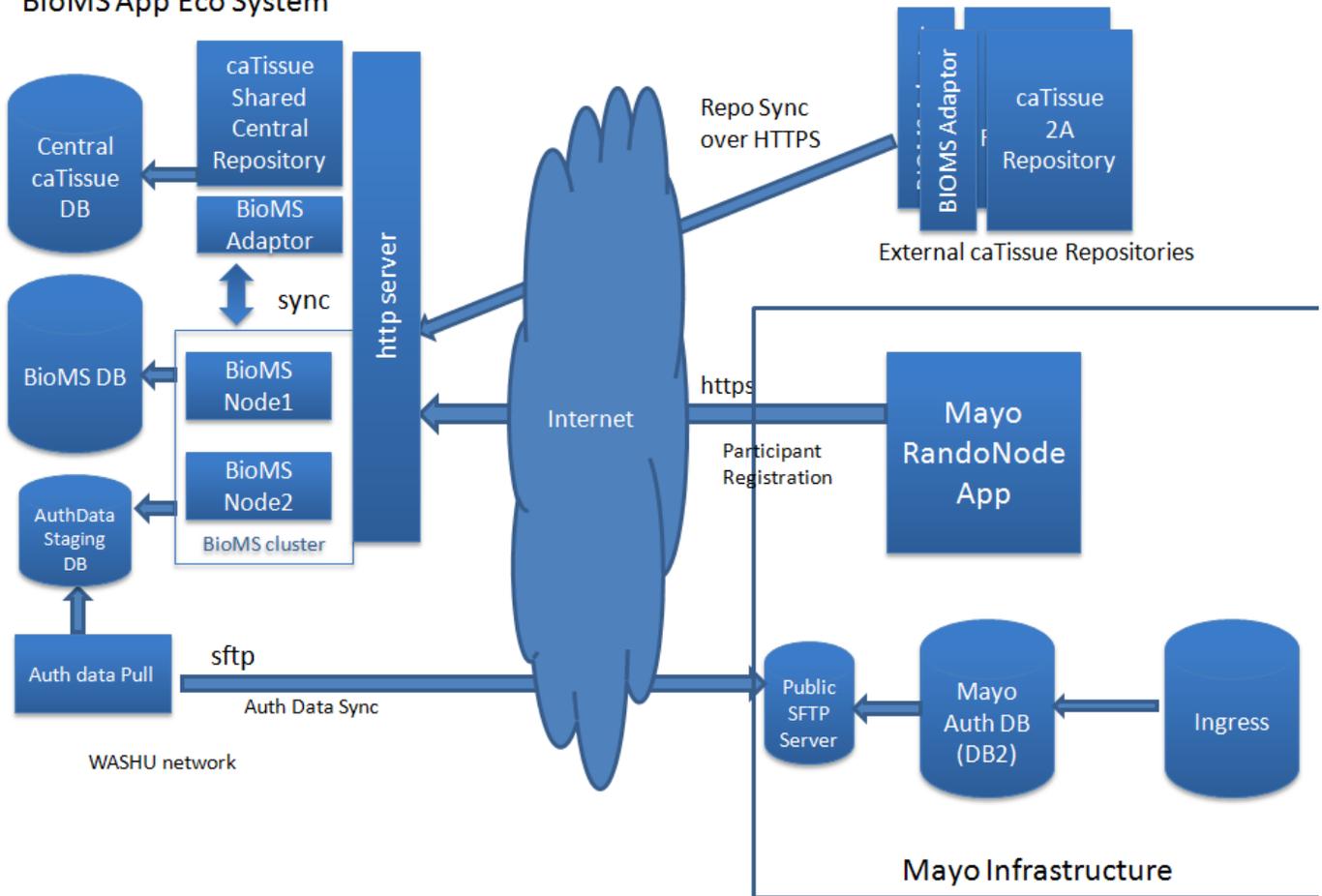
1. Purpose

This document is meant for the administrators of the BioMS application. This document describes the process of deploying the BioMS application and regular administration activities to be performed on the BioMS application.

2. Introduction

Here is the complete BioMS application ecosystem.

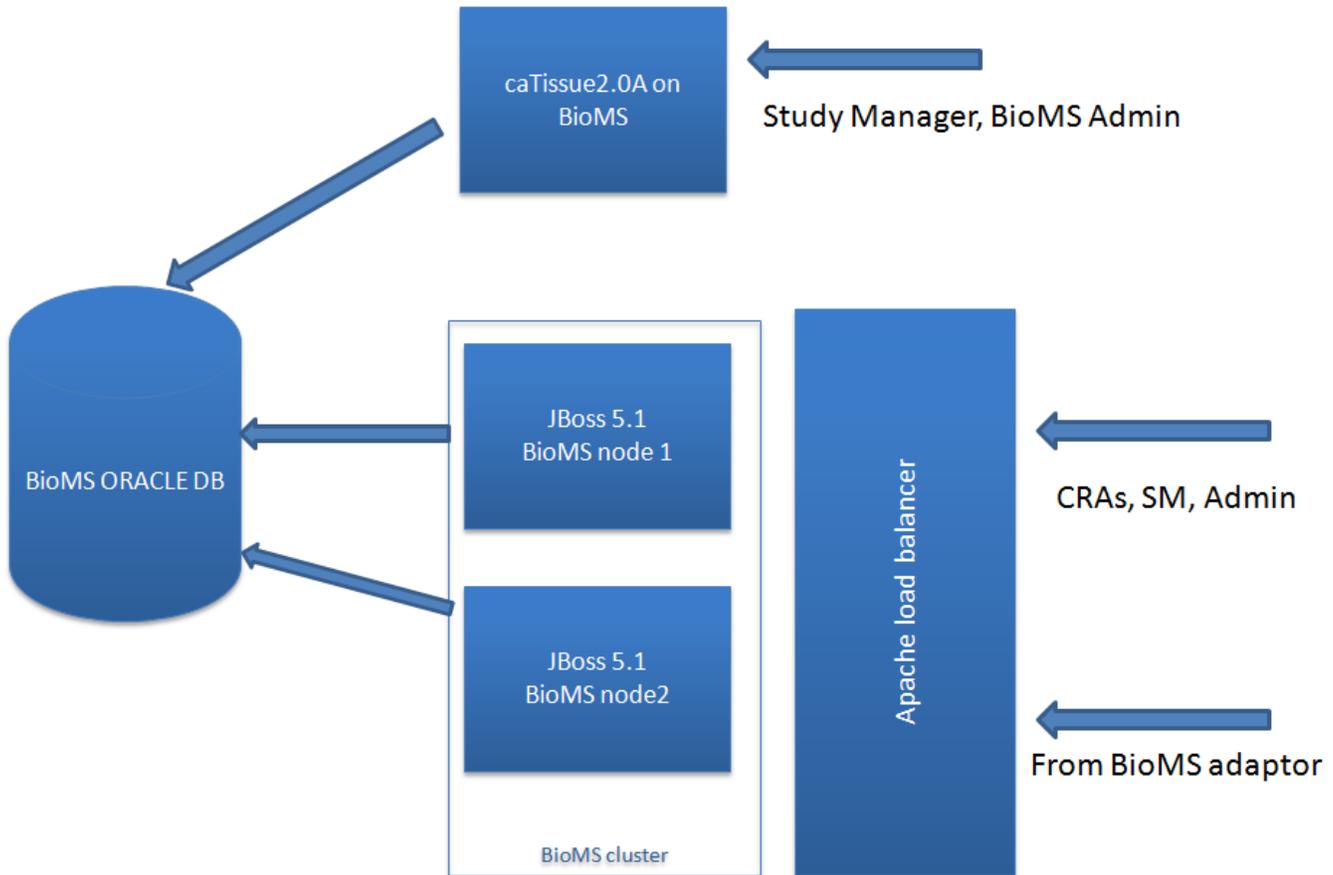
BioMS App Eco System



3. Deployment

The picture below shows the deployment model for the BioMS application with a two node Jboss cluster.

BioMS 1.0 Deployment Model



BioMS 1.0 application is deployed with an instance of caTissue 2.0A (shown as caTissue 2.0A on BioMS on the diagram above) which run against the same database schema BioMS is running on. This instance of caTissue is used for building specimen forms for use in BioMS, for generating reports on the specimen data stored in the BioMS DB and for performing some other administration activities.

BioMS application should be deployed in a JBoss cluster with minimum 2 nodes and front-ended by Apache http load balancer. BioMS application requires Oracle 10.3 Enterprise or above for database.

The sections below describe the steps for deploying the BioMS application.

4. Pre-requisites

Make sure the following pre-requisites are satisfied before starting the deployment of BioMS.

1. One Oracle 10g/11g Schema with permissions for creating table and sequences
2. 4 Server machine with the following minimum configuration
 - a. Dual core, 4GB RAM, 50 GB HDD x 2 – for BioMS cluster nodes
 - i. Software required on these machines : RHEL 5.0+, Oracle Client (sqlldr, tnspring, sqlplus), ftp , p7zip, jboss 5.1
 - ii. Open the following ports for UDP traffic on both nodes. This is required for the bioms jboss cluster to work.
 1. 45688
 2. 45710
 3. 44401
 4. 44402

5. 7900

- b. Dual core, 4GB RAM, 50 GB HDDx1 -for caTissue 2.0A on BioMS, this machine should have oracle client installed on it. This is required for installing caTissue.
 - i. Software required on this machine : RHEL 5.0+, Oracle Client (tnsping, sqlplus)
- c. Dual Core, 1GB RAM, 50 GB HDD x 1 – for the apache load balancer
 - i. Software required on this machine : RHEL 5.0+,apache

5. Setup caTissue2.0A on BioMS

Get the caTissue 2.0A installer **caTISSUE_Suite_v2.0_Installable.zip** and unpack it on caTissue2.0A on BioMS server. Follow the [caTissue 2.0 installation instruction](#) to install caTissue to the BioMS Oracle DB.

A simple setup of caTissue requires change the following properties in the install.properties, for other properties usually the default values are good.

```
application.base.path.linux=/usr/local/bms/catissueBMS-webapp
```

```
# Required for local installs
```

```
database.re-create=false
```

```
database.system.user= (this property should be left empty)
```

```
database.system.password=(this property should be left empty)
```

```
database.system.url=<db connection url> (e.g. jdbc:oracle:thin:@orad1m1.cbmi.wucon.wustl.edu:1521:clindbd1)
```

```
# Require for all installs
```

```
database.drop-schema=true
```

```
database.type=oracle
```

```
database.server=orad1m1.cbmi.wucon.wustl.edu
```

```
database.port=1521
```

```
database.name=clindbd1
```

```
database.user=ALLIANCEMBMS
```

```
database.password=<DB PASSWORD>
```

```
database.url=jdbc:oracle:thin:@orad1m1.cbmi.wucon.wustl.edu:1521:clindbd1
```

```
oracle.tns.name=clindbd1
```

```
#CSM DATABASE CREDENTIALS
```

```
esm.database.type=oracle
```

```
esm.database.host=orad1m1.cbmi.wucon.wustl.edu
```

```
esm.database.port=1521
```

```
esm.database.name=clindbd1
```

```
esm.database.username=ALLIANCEMBMS
```

```
esm.database.password=<DB_PASSWORD>
```

```
gsid.isEnabled=false
```

```
ccts.integration.isEnabled=false
```

```
ctrp.isEnabled=false
```

```
gridgrouper.isEnabled=false
```

```
load.balancer.url=<url at which users will be accessing this catissue, e.g url of the loadbalancer apache server, or external url of the catissue>
```

To install run **ant install** from the installer unpack folder

To upgrade an existing installation to a newer build , update the install.properties as detailed above and run **ant upgrade:jboss** from the installer unpack folder. Make sure the database properties in the install.properties is pointing to the database where the current catissue is running.

6. Setup BioMS on Bioms-node1

1. Get the BioMS 1.0 distribution from files.cbmi.wucon.wustl.edu/files/bioms/1.0/RC1/bioms-1.0.zip and unpack into a folder (lets call this folder BIOMS_INSTALL_HOME)
2. Get JBoss 5.1 from [here](#) and unpack into a folder (lets call this JBOSS_HOME).

3. Copy the folder JBOSS_HOME/server/all/ to JBOSS_HOME/server/bioms-node1
4. Copy the contents of BIOMS_INSTALL_HOME/jboss-overlay onto JBOSS_HOME/server/bioms-node1/
5. Copy BIOMS_INSTALL_HOME/war/bioms.war to JBOSS_HOME/server/bioms-node1/deploy
6. Edit JBOSS_HOME/bin/run.conf and replace the JAVA_OPTS variable with the following

```
JAVA_OPTS="-Xms128m -Xmx2048m -XX:MaxPermSize=1024m -Dorg.jboss.resolver.warning=true
-Dsun.rmi.dgc.client.gcInterval=3600000 -Dsun.rmi.dgc.server.gcInterval=3600000
-Dgov.nih.nci.security.configFile=$HOME/.bioms/conf/ApplicationSecurityConfig.xml"
```

7. Edit <jboss-home>/server/bioms-node1/deploy/cluster/jgroups-channelfactory.sar/META-INF/jgroups-channelfactory-stacks.xml and add line

```
bind_port="44401"
```

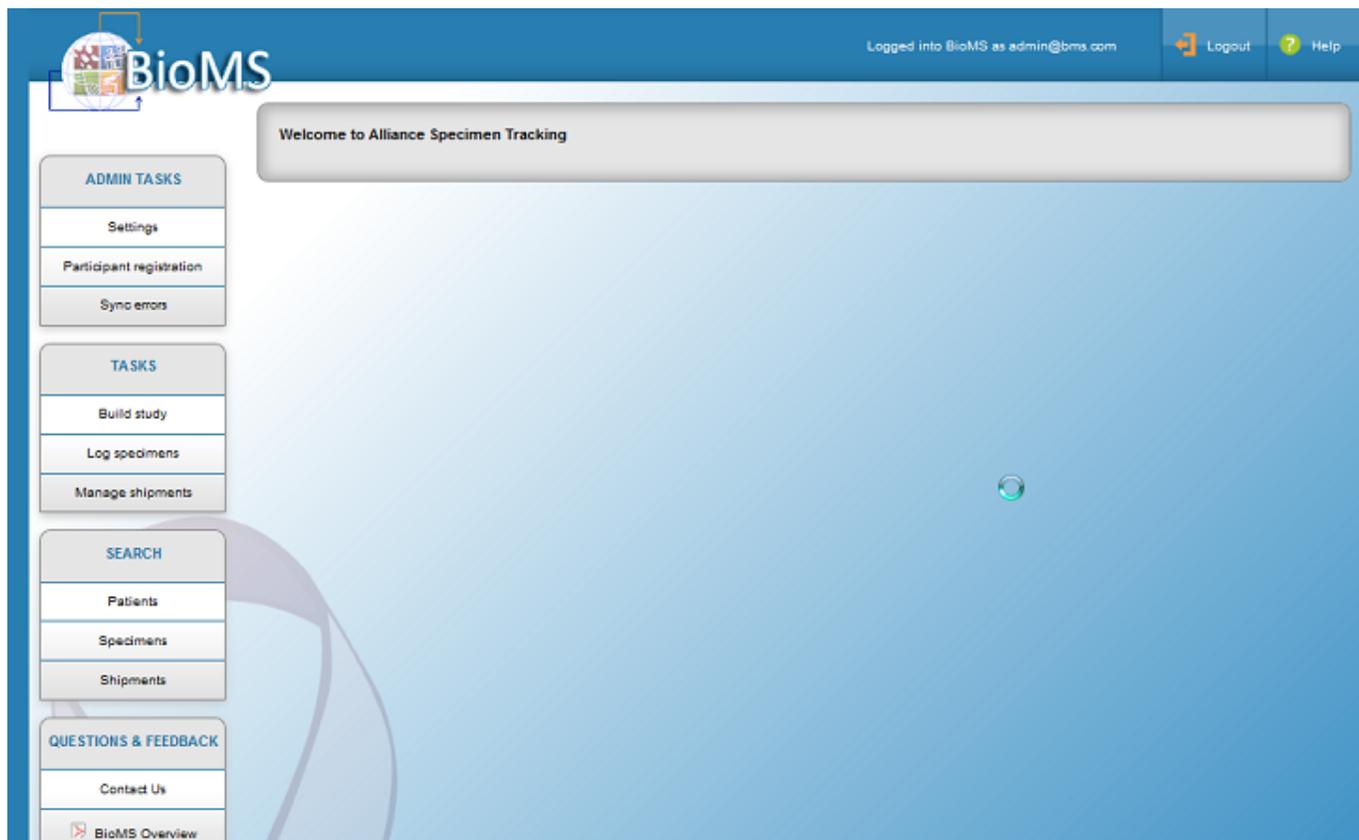
after the lines

```
<UDP
singleton_name="shared-udp"
mcast_port="{jboss.jgroups.udp.mcast_port:45688}"
mcast_addr="{jboss.partition.udpGroup:228.11.11.11}"
```

8. Edit JBOSS_HOME/server/bioms-node1/conf/login-config.xml and update the database connection details under the application-policy element with name 'bms' to point to the BioMS database schema.
9. Edit JBOSS_HOME/server/bioms-node1/deploy/bioms-ds.xml and update the database connection details
10. Copy BIOMS_INSTALL_HOME/.bioms to users home folder (\$HOME)
11. Edit \$HOME/.bioms/ApplicationSecurityConfig.xml and replace \${user.home} with the home directory path of the user.
12. Edit \$HOME/.bioms/bioms-config.groovy and update the following properties
 - a. datasource section with the database connection details for the BioMS database schema
 - b. set bms.baseURL to the base URL at which the application will be accessed by users. This should be set to the base load balancer URL. If the application is configured for external access then this url should be the external URL of the application. (e.g. <http://biomstest.wuslt.edu/bioms>)
12. Start the JBoss Server bioms-bioms-node1

```
run.sh -b0.0.0.0 -cbioms-node1 -u239.255.100.100 -gBioMSPartition -Djboss.messaging.ServerPeerID=1
```

This will start the server and the BioMS application deployed. Try accessing the BioMS application at url <http://<bioms-node1-server>:8080/bioms>. Login to the application using credentials admin@bms.com/Passw0rd and we should see a page similar to the following snapshot.



7. Setup BioMS on Bioms-node2

Follow the same steps give in Section 3.3 substituting all bioms-node1 with bioms-node2 to setup second node of the BioMS cluster on server BioMS node2

In node2 perform the following additional configuration.

1. Edit Edit `$HOME/.bioms/bioms-config.groovy` and update the following configuration

In the quartz section set `autoStartup=false`

Start the BioMS Node with the command

```
run.sh -b0.0.0.0 -cbioms-node2 -u239.255.100.100 -gBioMSPartition -Djboss.messaging.ServerPeerID=2
```

Make sure the -u and -g options values are matching with the values used in starting node1 and ServerPeerID is different

Once the bioms-node2 server is started make sure bioms-node2 bioms is accessible at `http://<bioms-node2-server>:8080/bioms`

8. Setup the Apache load balancer

Install apache HTTP server and configure mod_jk module and use the following mod_jk worker configuration

```

# Define list of workers that will be used
# for mapping requests
worker.list=loadbalancer,status

# Define Bioms-node1
# modify the host as your host IP or DNS name.
worker.bioms-node1.port=8009
worker.bioms-node1.host=<bioms-node1-host>
worker.bioms-node1.type=ajpl3
worker.bioms-node1.lbfactor=1
worker.bioms-node1.cachesize=10

# Define Bioms-node2
# modify the host as your host IP or DNS name.
worker.bioms-node2.port=8009
worker.bioms-node2.host=<bioms-node2-host>
worker.bioms-node2.type=ajpl3
worker.bioms-node2.lbfactor=1
worker.bioms-node2.cachesize=10

# Load-balancing behaviour
worker.loadbalancer.type=lb
worker.loadbalancer.balance_workers=bioms-node1,bioms-node2
worker.loadbalancer.sticky_session=1

# Status worker for managing load balancer
worker.status.type=status

```

Configure the worker uri map in Apache load balancer. Add the following uriworkermap.properties file in apache config

```

/bioms=loadbalancer

/bioms/*=loadbalancer

```

9. Mayo Authorization data Sync Setup

BioMS application would get Study, Site and allowed user information from the Mayo SSS system. BioMS application syncs authorization data nightly from the Mayo system. The bioms distribution contains a scripts that download the auth data bump from mayo and load into bioms database. This script should be configured to run nightly using cron in one of the bioms nodes. Follow the steps below to configure the sync process.

In Bioms Node1 edit `~/bioms/mayo-auth-sync/sync-mayo-auth-data.sh` and set the following properties to point to the BioMS db

BMS_DB_TNSNAME - TNS Name of the BioMS DB. The TNS name should be defined in the `tnsnames.ora` file.

BMS_DB_USERNAME - DB user name

BMS_DB_PASSWORD - DB password

To configure nightly mayo auth data sync cron job run 'crontab -e' command and add the following to the end of the file and save

```

0 22 * * * $HOME/.bioms/mayo-auth-sync/sync-mayo-auth-data.sh >
$HOME/.bioms/mayo-auth-sync/log/mayo-auth-sync.log.`date`
+%\%y\%m\%d-\%H\%M\%S` 2>&1

```

This will run the Mayo auth data sync every night at 10PM and output of the run would be logged to a file of the `$HOME/.bioms/mayo-auth-sync/log/mayo-auth-sync.log.YYMMDD-HHmmSS`

To test the sync job and trigger an immediate sync of the auth data run

```
$HOME/.bioms/mayo-auth-sync/sync-mayo-auth-data.sh >
$HOME/.bioms/mayo-auth-sync/log/mayo-auth-sync.log.`date
+\%y\%m\%d-\%H\%M\%S` 2>&1
```

Once the execution is complete check the log file at `$HOME/.bioms/mayo-auth-sync/log/mayo-auth-sync.log`.nnnnnnnn and make sure there are no errors in the log.

10. CTEP Authentication Setup

In QA and Production setup of BioMS users of BioMS will be authenticated against the CTEP user registry. To enable CTEP authentication perform the following steps in both bioms nodes

1. Set the following environment variable in `$HOME/.bashrc`

```
export CTSU_HOME=$HOME/.bioms/conf/CTEP
```

2. Edit `$HOME/.bioms/bioms-config.groovy` and in `ctep` section set `enabled=true`
3. Restart the bioms JBOSS server.

To verify CTEP authentication is enabled, go to the bioms home page and it should show the CTEP login screen.

Alliance Login Screen

Welcome to the Alliance Login Screen.

Username*:

Password*:

IAM-SSO

You are accessing a U.S. Government information system, which includes (1) this computer, (2) this computer network, (3) all computers connected to this network, and (4) all devices and storage media attached to this network or to a computer on this network. This information system is provided for U.S. Government-authorized use only.

Unauthorized or improper use of this system may result in disciplinary action, as well as civil and criminal penalties.

By using this information system, you understand and consent to the following.

You have no reasonable expectation of privacy regarding any communications or data transiting or stored on this information system. At any time, and for any lawful Government purpose, the government may monitor, intercept, record, and search and seize any communication or data transiting or stored on this information system.

Any communication or data transiting or stored on this information system may be disclosed or used for any lawful Government purpose.

[Forgot Password?](#) [Reset Password?](#) [Annual Registration](#) [Request New Account](#)

11. Participant Registration Sync Setup

Participant study registrations are pushed to bioms from the Mayo Rando Node application. Mayo Rando node invokes a BioMS url to push registration to BioMS. The invocation is protected by an authorization key shared with the Rando Node application. The authorization key for the rando node invocation is set by the property `registration.webservice.user` configured in the `~/bioms/bioms-config.groovy`.

Securely share the value of this property and the registration service url which is `<bms.baseURL>/integration/participantRegistration` (eg.

https://biomstest.wustl.edu/integration/participantRegistration)

12. Repository Sync

BioMS application run integrated with tissue repository caTissue application. When the BioMS application is integrated with repository caTissue application it automatically synchronizes specimen data and shipment data between BioMS and caTissue to keep the data in sync across the two application. BioMS application can be configured to sync with multiple repository caTissue instances.

More details on the repository sync can be found [here](#).

13. Administration

This section describes various ongoing administrative tasks to be performed on the BioMS application.

13.1. Setup sync with a repository caTissue 2.0A

Given below are steps for linking a new repository caTissue instance to BioMS for data synchronization.

1. Assign a unique name for the repository (<repo-name>)caTissue instance. E.g pco-repo.
2. Assign a secret authorization key (<authkey>) for the repository
3. Identify people who need to be informed when this repository connection to bioms breaks and the comma separated list of the emails of these people need to be set in the CONTACT column should below (<contact_emails>), e.g. admin@bioms.com, admin@pco.com

Add new row to the BMS_REMOTE_REPOSITORY table in the BioMS database with the following data.

ID	NAME	JMS_QUEUE_NAME	AUTHKEY	STATUS	CONTACT	
1 (max (id) +1)	<repo-name>	<repo-name>	<authkey>	1	<contact_emails>	

Here is the SQL for the inserting the repository data into the DB

```
Insert into BMS_REMOTE_REPOSITORY
( ID,NAME,JMS_QUEUE_NAME,AUTHKEY,STATUS, LAST_HEART_BEAT_TIME,CONTACT) values
((select max(id)+1 from
BMS_REMOTE_REPOSITORY), '<repo-name>', '<repo-name>', '<authkey>', 1,SYSDATE, '
<contact_emails>');
```

Securely share the <repo-name> and <authkey> and the bioms application url (the loadbalancer url) with the repository caTissue admin. They would need to update the bioms-adaptor.properties with these details.

Once the bioms-adaptor is setup properly at the repository caTissue and started you should see message like

controller.RepoSyncMessageController 2012-10-08 09:59:11,996 Sending 204 (no message available) to repo <repo-name>
in the bioms log.

Also messages like the following would be there on the BioMS adaptor log file.

5:01:14,745 INFO [STDOUT] 2012-10-08 15:01:14 SyncMessageReceiver [DEBUG] No mssage received, sleep for 10sec before trying again...

13.2. Setup a new Repository Site

Once the repository caTissue instance is linked with BioMS as described above the follow the steps below to setup a new Repository Site for sync. A single caTissue instance can host multiple independent repository sites and BioMS can handle that.

1. Request the caTissue Admin to create Repository Site along with the coordinator user in the caTissue and note the identifier of the Repository Site.
2. Get the details of the Repository Site and the Coordinator user Name from the repository caTissue admin.
3. Create a Site of type **Repository** in BioMS via the 'caTissue2.0A on BioMS' caTissue instance and select the coordinator user created by the caTissue admin as the coordinator (this user would have been synced to BioMS automatically). Note the id of the site as <bms_repo_site_id> just created from caTissue2.0A on BioMS.
4. Link the repository site created above with the caTissue remote repository created in Step 13.1. For this insert a new row into the BMS_REMOTE_REPOSITORY_SITE table as shown below.

REMOTE_REPOSITORY_ID	SITE_ID
<id of the remote repo entry for the remote repo>	<bms_repo_site_id>

Here is the SQL command for this insert:

```
Insert into BMS_REMOTE_REPOSITORY_SITE (REMOTE_REPOSITORY_ID,SITE_ID)
values ((select id from BMS_REMOTE_REPOSITORY where
name=<repo-name> ),<bms_repo_site_id>);
```

5. **(To be performed on the caTissue repository)** Map the new Site created in BioMS with the corresponding repository Site created in caTissue per step 1. For this request the repository caTissue admin to insert the following data into BMS_CATISSUE_ENTITY table in the repository caTissue data base

ID	ENTITY_TYPE	BMS_ID	CATISSUE_ID
BMS_CATISSUE_ENTITY_MAP_SEQ.nextval	edu.wustl.catissuecore.domain.Site	<bms_repo_site_id>	<id_of_repo_site_in_catissue>

Here is the SQL insert command for the same.

```
Insert into BMS_CATISSUE_ENTITY_MAP
(ID,ENTITY_TYPE,BMS_ID,CATISSUE_ID) values
(BMS_CATISSUE_ENTITY_MAP_SEQ.NEXTVAL, 'edu.wustl.catissuecore.domain.S
ite', <bms_repo_site_id>, <catissue_repo_site_id>);
```

We should now be able to create studies with the new repository as the ship to site for specimen and sync the studies. When the study is synced study should show up in the caTissue.

13.3. Building and rolling out new Study

Note: This section needs to be revised based on the business process
This section describes the steps for building and rolling out a new Study in BioMS.

1. Identify the specimen form requirements for the study and if there is any specimen form required, build the form using the Local Extensions feature in the caTissue2A on BioMS. The form needs to be added to the form group with name **bmsforms**. Only these forms in this group will be available for selection in BioMS study builder.

The current version of BioMS has some restrictions on form fields and constraints and doesn't support all the capabilities of forms that could be created in caTissue. Given below is the summary of what is supported in the BioMS specimen forms.

Textfield (string ,numeric,URL datatypes and width, max length, min and max constraints are supported)

TextArea (width, max length and rows constraints are supported)

Checkbox

Radio Button

ListBox

ComboBox

Date picker (Date only)

File picker is not supported at this time. Also form inheritance and forms associations are not supported.

2. Once the form is built in the caTissue 2A on BioMS, send the exact form details, like the name of the form, form attribute names, data types and constraints to the admins of those repositories, to which atleast one specimen from the study would be shipped. The repository caTissue admin should build the exact same form with the details provided by the BioMS admin.

3. While step 2 is happening, BioMS Study manager can continue building the study in BioMS study builder. The Study activityStatus should be set to Disabled while it is getting built.

Note: Study Manager can build only those studies with the ids that were synced from the Mayo System via the Mayo authorization sync process. Mayo System source of record for the Study ids and the study titles. While the Mayo Authorization sync is getting setup, admin can manually insert Study IDs and study descriptions to MAYO_STUDY. Only those study ids inserted in the MAYO_STUDY table would be available for building in BioMS Study builder

4. Once the study is completely built SM clicks on the Synchronize Study to sync the study to all the repositories where specimen from the study would be shipped to.

5. When the Study is ready to accept registrations, change the study status to Active in BioMS.

13.3. Registering participants to Study

Participant registrations usually comes from Mayo Rando Node applications. But until the Mayo Rando Node integration is enables, BioMS admin can register participants to study via the participant registration page provided in BioMS. The participant registration page can be accessed from the ADMIN TASKS menu group when logged in as BioMS Admin.

13.4. Watching for Sync issues

BioMS admin can see if there are any happening during syncing of data between BioMS and repository caTissue from the Sync Errors menu in the ADMIN TASKS menu group accessible only for BioMS admins. This page lists all the errors occurred during the sync with the sync message and the error details including any stack trace.

BioMS admin should review this list of error at least once in a day to make sure sync functions are working properly.

13.5. CALGB Incremental Registration load

The CALGB incremental registration load is configured in alliance bioMs (dev) server.

The load script `/usr/local/bms/.bioms/calgb-reg-sync/load-incr-calgb-regdumps.sh` is configured to run at 15 minutes on the hours 7AM-8PM EST Monday-Friday.

The script downloads the incremental registrations dump files from <ftp.mayo.edu> and attempts to load them into bioms QA deployment.

The logs from each execution of the incremental load goes into the folder `alliancebms@/usr/local/bms/.bioms/calgb-reg-sync/log`