inBloom – *Environment Specification*

Release 1.2.70

February 22, 2013

**Copyright © 2012 inBloom, Inc., and its affiliates.**

**inBloom is a trademark of inBloom, Inc.**

This document and the information contained herein is provided on an "AS IS" basis and inBloom, Inc. DISCLAIMS ALL WARRANTIES, EXPRESS OR IMPLIED, INCLUDING BUT NOT LIMITED TO ANY WARRANTY THAT THE USE OF THE INFORMATION HEREIN WILL NOT INFRINGE ANY RIGHTS OR ANY IMPLIED WARRANTIES OF MERCHANTABILITY OR FITNESS FOR A PARTICULAR PURPOSE.

1. Purpose 3

2. Summary 3

3. Infrastructure Configuration 3

3.1. MongoDB (Data Store Layer) 3

3.2. Ingestion 4

3.2.1. Landing Zones 4

3.2.2. GlusterFS 4

3.2.3. ActiveMQ 4

3.2.4. Staging DB 4

3.2.5. Ingestion servers 5

3.3. Portal 5

3.4. API 5

3.5. Data Browser 5

3.6. Admin Tools 6

3.7. Simple IDP 6

3.8. Search 6

4. Platform Migration Steps and Changes 7

4.1. Change: Search Indexer start script supports external configuration 7

4.2. Tomcat Configuration Change for Ingestion Servers 8

4.3. Developer Account Federation Support 8

4.4. Database Migration Steps 1.0.68 to 1.2.69 9

4.5. Release 70 Index Migrations 10

4.5.1. Additions 10

4.5.2. Deletions 10

4.5.3. Downtime 10

4.6. Enabling ‘Self’ Context & Roles for Teacher/Staff 10

4.6.1. Background 10

4.6.2. Description of Migration Script 11

4.6.3. Running of Migration Script 11

4.6.4. Output of Migration Script 11

4.6.5. Verification of Migration Script 11

5. Suggested Base Configuration 14

5.1. sli.properties 14

5.2. Portal Application sli.properties 19

5.3. portal-ext.properties 19

5.4. Admin config.yml 20

5.5. Data browser config.yml 22

5.6. Apache Tomcat 22

5.7. ActiveMQ 22

# Purpose

This document is intended to provide a minimum specification guideline for the deployment of the release into the inBloom™ Production and Sandbox environments, while also setting the expectation for what an environments components are expected to be in existence for the Release.

# Summary

This document provides minimum deployment footprints for this release, as well as an estimated footprint necessary for 5M students, and the Sandbox developer environment.  
  
Due to the expectation of a continued deployment within an Amazon AWS environment, this document has been geared as such.

# Infrastructure Configuration

## MongoDB (Data Store Layer)

Notice: Ingestion also requires separate MongoDB servers for staging purposes, which are covered within the Ingestion section of this document.

* Minimum Requirements
  + Three MongoDB Servers in a replica set (one shard)  
    This means that there is one master server and two replicated backups.
    - Recommended AWS Instance Class: High-Memory Quadruple Extra Large (m2.4xlarge) 68.4GB
    - Recommended storage: RAID10 EBS with PIOPS
  + Three MongoDB configuration servers  
    MongoDB specifies using exactly 3 config servers
    - Recommended AWS Instance Class: Large (m1.large) 7.5GB
    - Recommended storage: ephemeral
* The Data Store layer is scaled by adding replica sets (shards) where each replica set consists of 3 MongoDB servers as described above
* Requirements for 5M students  
  nine MongoDB replica sets (shards), for a total of twenty seven MongoDB servers.
* Recommendations for Sandbox  
  four MongoDB replica sets (shards), for a total of twelve MongoDB servers

## Ingestion

### Landing Zones

* Two Landing Zone Servers
  + Amazon Large (m1.large) instance providing SFTP services for Ingestion Users via ProFTPD (per Runbook).
  + Mount GlusterFS (per Runbook).

### GlusterFS

* Two GlusterFS instances providing scalable replicated distributed storage services to the Landing Zone servers and the ingestion servers.
* Minimum Requirements
  + Amazon m1.large instances
  + Gluster volume configured with a replica count of two, matching the number of initial storage bricks.
  + PIOPS EBS volume, 500GB size, 2000 IOPS, to be used as GlusterFS storage brick.
    - The storage size amount is purely an estimate based on existing testing in release candidate environments. It must be understood that this must be monitored, and potentially adjusted by the operator. It is very straightforward to add bricks to GlusterFS without incurring any downtime.
* Requirements for 5M  
  Amazon High I/O Quadruple Extra Large (hi1.4xlarge) instances
* Recommendations for Sandbox  
  Minimum requirements as described above

### ActiveMQ

* Two ActiveMQ Servers  
  One active, one secondary
  + Configuration details for ActiveMQ Clustering Configuration can be found at http://activemq.apache.org/networks-of-brokers.html.
  + Recommended AWS Instance Class: Large (m1.large) 7.5GB
  + Recommended storage: ephemeral

### Staging DB

* Three MongoDB Servers  
  One master and two replicated backups.
  + Recommended AWS Instance Class: High-Memory Quadruple Extra Large (m2.4xlarge) 68.4GB
  + Recommended storage: EBS with PIOPS

### Ingestion servers

* This is considered ingestion “standalone” mode as opposed to Maestro/Pit mode, which will be merged in the future releases.
* Minimum Requirements: two servers
  + Recommended AWS Instance Class: High-Memory Quadruple Extra Large (m2.4xlarge) 68.4GB
  + Recommended storage: ephemeral
  + Tomcat service should run as root user.
* Requirements for 5M  
  Four standalone ingestion servers
* Recommendations for 5M students  
  Seven standalone ingestion servers

## Portal

* Recommended Configuration
  + Utilize Amazon Elastic Load Balancer.
  + Two Tomcat web servers running the LifeRay product
    - Recommended AWS Instance Class: Large (m1.large) 7.5GB
    - Recommended storage: ephemeral
  + Use of Amazon RDS for back-end Portal MySQL Data storage
    - Recommended Large Database Instance making use of a Multi-AZ Deployment.

## API

* Minimum Requirements:
  + Utilize Amazon Elastic Load Balancer
  + Two Tomcat web servers  
    These run the Data Infrastructure software for the API servers
    - Recommended AWS Instance Class: Large (m1.large) 7.5GB
    - Recommended storage: ephemeral
* Requirements for 5M students  
  It is estimated that 5 million students will lead to a peak load of 55 pages/second (based on the API Load Estimate spreadsheet). This load will require 5 Tomcat web servers.
* Recommendations for Sandbox  
  For Sandbox, we recommend 3 API servers.

## Data Browser

* Scaling the Data Browser layer is accomplished by scaling Rails Web Servers. It is anticipated that these servers will be sufficient for 5M students and for Sandbox.
  + Minimum Requirements:
    - Utilize Amazon Elastic Load Balancer
    - Two Ruby on Rails web servers  
      These run the Data Infrastructure software for the Dashboard
      * Recommended AWS Instance Class: Large (m1.large) 7.5GB
      * Recommended storage: ephemeral

## Admin Tools

* Scaling the Admin layer is accomplished by scaling Rails Web Servers. It is anticipated that these servers will be sufficient for 5M students and for Sandbox.
  + Minimum Requirements:
    - Utilize Amazon Elastic Load Balancer
    - Two Ruby on Rails web servers  
      These run the Data Infrastructure software for the Dashboard
      * Recommended AWS Instance Class: Large (m1.large) 7.5GB
      * Recommended storage: ephemeral
  + Notice: The Admin Tools are dependent upon the API, and the OpenLDAP servers.

## Simple IDP

* Simple IDP is the Identity Provider for the Admin realms.   
  It is anticipated that these servers will be sufficient for 5M students and for Sandbox.
  + Minimum Requirements:
    - Utilize Amazon Elastic Load Balancer
    - 2 Simple IDP Servers
      * Recommended AWS Instance Class: Large (m1.large) 7.5GB
      * Recommended storage: ephemeral
    - 2 OpenLDAP Servers
      * Recommended AWS Instance Class: Large (m1.large) 7.5GB
      * Recommended storage: EBS

## Search

* Search Servers  
  These servers are used for the Elastic Search functionality.  
  The minimum requirements above will be sufficient for 5M students and for Sandbox.
  + Minimum Requirements:
    - Two Search Engine Servers
      * These instances will host the ElasticSearch infrastructure.
      * Recommended AWS Instance Class: Extra Large m2.2xlarge
      * Recommended storage: EBS
    - Two Index Servers
      * These instances will host the search-indexer.
      * Recommended AWS Instance Class: Large m1.xlarge
      * Recommended storage: EBS
* SARJE Servers  
  SARJE is infrastructure used by Search to run asynchronous jobs. Currently its sole use is for Search. It is anticipated that these servers will be sufficient for 5M students and for Sandbox.
  + Minimum Requirements:
    - Two ActiveMQ Servers
      * These are expected to be dedicated and separated from the Ingestion ActiveMQ instances.
      * Recommended AWS Instance Class: Large (m1.large)
      * Recommended storage: EBS
    - Two OpLog Agent servers
      * Recommended AWS Instance Class: Large (m1.large) 7.5GB
      * Recommended storage: EBS
    - Two Event Subscriber servers - Expected in Release 7.0
      * Recommended AWS Instance Class: Large (m1.large) 7.5GB
      * Recommended storage: EBS

# Platform Migration Steps and Changes

## API Grade Period Configuration

Production environments MUST be configured with the sli.security.gracePeriod set to a value of 0, where as sandbox environments must be set to a value of 2000.  
  
Production Configuration:  
 sli.security.gracePeriod = 0  
Sandbox Configuration:  
 sli.security.gracePeriod = 2000

## Database Migration Steps 1.2.69 to 1.2.70

1. Due to a number of index changes in Release 70, allow index addition to take place prior to the outage window as indicated in the *Release 70: Index Migrations, Additions* subsection below.
2. Beginning the Upgrade, ensure that all inBloom platform components that interact with the data store have been stopped, and ensure that backups have been performed prior to beginning this upgrade process.  These components would consist of the API, Ingestion, Search-Indexer, and SARJE.
3. If not already started, execute the *“Release 70: Index Migrations - Additions”* section below.
4. Execute the *“Release 70: Index Migrations - Deletions”* section below.
5. Execute the *“Release 70: App Auth Migration”* script detailed in the next sections.
6. Execute the *“Release 70: Enabling ‘Self’ Context & Roles for Teacher/Staff”* script detailed  one of the next sections.
7. Apply the *ingestion\_batch\_job.js* Index file to the ingestion\_batch\_job database to ensure indexes are in place for a collection that has been added.
8. Apply an update to the SLI database indexes:
   1. Open a Mongo CLI to the sli database. Example: “mongo localhost:27017/sli”
   2. Remove Indexes from the Application Collection, via command “db.application.dropIndexes()”
   3. Exit the Mongo CLI, and apply the sli\_indexes.js file to the sli database. Example: “mongo localhost:27017/sli <sli\_indexes.js”
9. Once the above steps have been executed, the inBloom platform components can be updated to Release 70.

## Release 70: Index Migrations

The migration of MongoDB indexes for release 70 is split into index additions, which must be executed before platform upgrade, and index deletions, which must be done after platform upgrade. This process avoids downtime.

### Additions

Index additions must be done before platform upgrade. The 10gen-recommended process for adding indexes is detailed at http://docs.mongodb.org/manual/core/indexes/#index-creation-background. The supplied script, 70IndexBefores.rb, takes two arguments:

1. A mongos address to read from the cluster.
2. A mongod address for the node to apply indexes.

The list of tenants must be read from the entire cluster, while indexes are added one node at a time.

### Deletions

Index deletions must be done after platform upgrade. The script, 70IndexAfters.rb, takes one argument:

1. A mongos address to read from the cluster.

Index deletions are performed via the entire cluster.

### Downtime

Using the provided scripts and the 10gen-recommended process for adding indexes, indexing migrations can be done with no downtime. If this is of no concern, the system can be taken offline, and the addition script (70IndexBefores.rb) can be run on the entire cluster, as opposed to node-by-node, followed by the deletion script (70IndexAfters.rb).

## Release 70: App Auth Migration Script

As part of the fix for DE2510, we needed to alter the schema for the application authorization collection in mongo.    
  
Description of what the migration Script does:  
This script will change the applicationAuthorizations in the tenant specific databases to match the new schema where the applicationId is the primary lookup, mapping to an array of educationOrganizations which have allowed that app use of their data  
  
**How to run the migration script:**  
The migration script can be found at:

*opstools/migration/70ApplicationAuthorizationMigration.rb*  
Usage pattern is:

70ApplicationAuthorizationMigration.rb {mongo host}:{mongo port}  
Example:

ruby 70ApplicationAuthorizationMigration.rb localhost:27017  
  
**Output of the migration script:**  
$ ruby 70ApplicationAuthorizationMigration.rb localhost:27017  
updating db: 40b558db2acdbbe0d12e9e4dc84380b178d16017  
updating db: 49bf23dc9d2ee913bfc6aa086c0ea16ed553e010  
updating db: 4ab064f9f39d1b1a82f42807a9491abf18592ce6  
updating db: 6e6a6f2086bb5fe5dbfd17d8d5f502d48759834b  
  
**How to tell if the migration script was successful:**  
*Low level:* make the following mongo queries to a tenant specific database and verify the expectations.

The following should equal 0:

db.applicationAuthorization.find( { "body.authType" : "EDUCATION\_ORGANIZATION" } ).count()

The following should be greater than 0:

db.applicationAuthorization.find( { "body.applicationId" : { $exists : true } } ).count()  
  
*High level:* A user who was able to use an app in the 69 code base should be able to login to that same app in the 70(+) code base.

## Release 70: Enabling ‘Self’ Context & Roles for Teacher/Staff

### Background

As part of enabling self-context for aggregate viewers, teachers, and staff members, we have created a migration script for updating the ‘customRole’ collection in each tenant-specific database.

### Description of Migration Script

This script will add a ‘selfRights’ context to each custom roles document in the tenant specific databases. The ‘selfRights’ context is accessed when an actor (say, an Educator) makes an API call to access themselves (more specifically, /staff/{id}). If the actor has requested their entity, and only their entity, their READ authorities will be elevated. Details of the elevation can be found in the US5053 story description, found on Rally:  
<https://rally1.rallydev.com/#/4711755666d/detail/userstory/9834862716>

### Running of Migration Script

The migration script can be found here:

{sli\_home}/sli/opstools/migration/custom\_roles/70\_self\_rights\_migration.rb

To run the script, you must specify the hostname and port of the MongoDB instance to connect to, as seen below:

ruby 70\_self\_rights\_migration.rb {host}:{port}

An example used to connect to the mongod (or mongos) for updating the ‘customRole’ collections in each tenant-specific database is found under ‘Output of the migration script’. Failing to specify a hostname and port, without spaces, will halt script execution and output an error message of the following form:

Usage: 70\_self\_rights\_migration.rb <host>:<port>  
host - hostname for mongo instance  
port - port mongo is running on (27017 is common)

### Output of Migration Script

$ ruby 70\_self\_rights\_migration.rb [localhost:27017](http://localhost:27017/)

INFO -- : --------------------------------------

INFO -- :  Migration script for self-context

INFO -- : --------------------------------------

INFO -- : Adding self rights for custom roles in db: <tenant db hash>

INFO -- : Adding self rights for custom roles in db: <tenant db hash>

...

### Verification of Migration Script

First, verify that none of the following databases were altered by the script:  
[ 'admin', 'config', 'ingestion\_batch\_job', 'sli', 'local' ]  
  
This can be done by looking at the output of the migration script (in the console where it was run), and making sure that none of the databases were ‘visited’ during script execution.  
  
The self-rights migration script will alter each custom roles document as follows:

* The ‘Aggregate Viewer’ should have READ\_GENERAL and READ\_RESTRICTED authorities in self-context.
* The ‘Educator’ group should have READ\_RESTRICTED authority in self-context.
* The ‘Leader’ and ‘IT Administrator’ groups should have no elevation of authorities in self-context (and no ‘selfRights’ should appear on the group).
* Any additional custom roles defined in the ‘customRole’ document, not part of the 4 default roles listed above, should have READ\_GENERAL and READ\_RESTRICTED authorities in self-context.

To verify that the above alterations have been successfully completed, connect to any of the databases listed in the output, and run the following command:

db.customRole.findOne()

The entry for ‘Educator’ in the ‘customRole’ document should be updated from this:

{  
    "groupTitle": "Educator",  
    "isAdminRole": false,  
    "names": [  
      "Educator"  
    ],  
    "rights": [  
      "READ\_GENERAL",  
      "AGGREGATE\_READ",  
      "READ\_PUBLIC"  
    ]  
  }

to this:

  {  
    "groupTitle": "Educator",  
    "isAdminRole": false,  
    "names": [  
      "Educator"  
    ],  
    "rights": [  
      "READ\_GENERAL",  
      "AGGREGATE\_READ",  
      "READ\_PUBLIC"  
    ],  
    **"selfRights": [**  
**"READ\_RESTRICTED"**  
**]**  
  }

The entry for ‘Aggregate Viewer’ in the ‘customRole’ document should be updated from this:

{  
    "groupTitle": "Aggregate Viewer",  
    "isAdminRole": false,  
    "names": [  
      "Aggregate Viewer"  
    ],  
    "rights": [  
      "AGGREGATE\_READ",  
      "READ\_PUBLIC"  
    ]  
  }

to this:

  {  
    "groupTitle": "Aggregate Viewer",  
    "isAdminRole": false,  
    "names": [  
      "Aggregate Viewer"  
    ],  
   "rights": [  
      "AGGREGATE\_READ",  
      "READ\_PUBLIC"  
    ],  
    **"selfRights": [**  
**"READ\_GENERAL",**  
**"READ\_RESTRICTED"**  
**]**  
  }

Entries for any custom roles in the ‘customRole’ document should be updated from this:

{  
    "groupTitle": "CustomReadWrite",  
    "isAdminRole": false,  
    "names": [  
      "CustomReadWriteRole"  
    ],  
    "rights": [  
      "READ\_GENERAL",  
      "WRITE\_GENERAL",  
      "READ\_PUBLIC",  
      "READ\_RESTRICTED"  
    ]  
  }

to this:

  {  
    "groupTitle": "CustomReadWrite",  
    "isAdminRole": false,  
    "names": [  
      "CustomReadWriteRole"  
    ],  
    "rights": [  
      "READ\_GENERAL",  
      "WRITE\_GENERAL",  
      "READ\_PUBLIC",  
      "READ\_RESTRICTED"  
    ],  
**"selfRights": [**  
**"READ\_GENERAL",**  
**"READ\_RESTRICTED"**  
**]**  
  }

The entries for ‘Leader’ and ‘IT Administrator’ roles in the ‘customRole’ document should not be changed. Please verify that no ‘selfRights’ exist on the entries for ‘Leader’ or ‘IT Administrator’.

# Suggested Base Configuration

*Note:* The configuration examples provided below detail a running system. Due to the rebranding flux, some properties may not have been able to be identified from a development point of view. Therefore, email addresses and URLs that are imbedded in to the configuration should be validated prior to go-live.

## sli.properties

api.perf.log.path = /var/log/tomcat  
sli.security.noSession.landing.url = https://rcapi.slidev.org/api/oauth/authorize?response\_type=code  
# sli.security.sp.issuerName is the URL of the API service.  
sli.security.sp.issuerName = https://rcapi.slidev.org  
sli.security.gracePeriod = 0  
sli.trust.certificates = /opt/tomcat/trust/trustedCertificates  
# Security events notices are sent to the following email address.  
sli.support.email = support@inBloom.org  
#The following settings govern user login session lengths, in milliseconds.  
sli.session.length = 1800000  
sli.session.hardLogout = 28800000  
# The following setting must be updated to match the cookie domain of the installation.  
sli.api.cookieDomain = .slidev.org  
# The following three settings govern Sandbox and API Application creation behavior.  
sli.sandbox.enabled = true  
sli.autoRegisterApps = true  
bootstrap.sandbox.createSandboxRealm = true  
# The following bootstrap settings will require the FQDN of service names to be updated.  
bootstrap.admin.realm.name = inBloom  
bootstrap.admin.realm.tenantId = SLI  
bootstrap.admin.realm.idpId = https://rcidp01ext.slidev.org/sliidp?realm=SLIAdmin  
bootstrap.admin.realm.redirectEndpoint = https://rcidp01ext.slidev.org/sliidp?realm=SLIAdmin  
bootstrap.sandbox.realm.uniqueId = SandboxIDP  
bootstrap.sandbox.realm.name = Sandbox Environment  
bootstrap.sandbox.realm.idpId = https://rcidp01ext.slidev.org/sliidp  
bootstrap.sandbox.realm.redirectEndpoint = https://rcidp01ext.slidev.org/sliidp  
bootstrap.app.conf = \${sli.conf}  
# sli.tenant.ingestionServers is a depricated setting.  
sli.tenant.ingestionServers = rcingest01  
# sli.tenant.landingZoneMountPoint governs the file path that the API will utilize when creating tenant collection entries for the Ingestion Service.  
sli.tenant.landingZoneMountPoint = /ingestion/lz  
sli.landingZone.server = rclz01.slidev.org  
sli.useraccount.maximum = 500  
# The following settings until sli.ingestion settings generally need to be unmodified with the exception of sli.api.ldap settings.  
sli.api.ldap.user = cn=Admin,dc=slidev,dc=org  
sli.api.ldap.pass = [LDAP\_ADMIN\_PASS]  
sli.application.buildTag = sli.app.buildTag  
sli.api.performance.tracking = false  
sli.api.security.context.paging = 100000  
sli.security.in\_clause\_size = 100000  
sli.sandbox.autoRegisterApps = false  
sli.mongodb.database = sli  
sli.mongodb.host = localhost:27017  
sli.mongodb.port = 27017  
sli.mongodb.user =   
sli.mongodb.pass =   
sli.mongodb.keyencoding = \%:\%25,\\.:\%2E  
sli.perf.mongodb.database = apiPerf  
sli.perf.mongodb.host = localhost  
sli.perf.mongodb.port = 27017  
sli.mongodb.connections = 20  
sli.stagingmongodb.connections = 20  
sli.batchjobmongodb.connections = 20  
# sli.ingestion settings require server names and potentially database names updated based on the environment configuration.  Generally the bulk of the settings here should remain unmodififed.  
sli.ingestion.staging.mongodb.database = is  
sli.ingestion.staging.mongodb.host = rcmongo10.slidev.org  
sli.ingestion.staging.mongodb.port = 27017  
sli.ingestion.staging.clearOnCompletion = true  
sli.ingestion.batchjob.mongodb.database = ingestion\_batch\_job  
sli.ingestion.batchjob.mongodb.host = rcmongo10.slidev.org  
sli.ingestion.batchjob.mongodb.port = 27017  
sli.ingestion.errors.tracking = true  
sli.ingestion.warnings.tracking = true  
sli.test.prop = ci DAL Context Test Property  
sli.ingestion.securityEvent.capSize =   
sli.ingestion.healthcheck.user = admin  
sli.ingestion.healthcheck.pass = admin  
sli.mongo.tracking = false  
sli.mongo.tracking.interval.seconds = 5  
sli.ingestion.mongotemplate.writeConcern = SAFE  
sli.ingestion.staging.mongotemplate.writeConcern = SAFE  
sli.default.mongotemplate.writeConcern = SAFE  
landingzone.inbounddir = /ingestion/lz/  
logging.path = /var/log/tomcat  
sli.ingestion.topic.command = activemq:topic:ingestion.command  
sli.ingestion.exception.message.log = true  
sli.ingestion.log.level = info  
sli.ingestion.queue.workItem.host = rcingest01.slidev.org  
sli.ingestion.queue.workItem.port = 61616  
sli.ingestion.queue.workItem.secondaryhost =   
sli.ingestion.queue.workItem.secondaryport =   
sli.ingestion.queue.options = keepAlive=true&jms.prefetchPolicy.queuePrefetch=0&wireFormat.maxInactivityDurationInitalDelay=60000  
sli.ingestion.queue.brokerUrl = tcp://${sli.ingestion.queue.workItem.host}:${sli.ingestion.queue.workItem.port}?${sli.ingestion.queue.options}  
sli.ingestion.queue.maxConnections = 25  
sli.ingestion.queue.maximumActive = 500  
sli.ingestion.queue.workItem.queueURI = seda:IngestionWorkItem  
sli.ingestion.queue.workItem.concurrentConsumers = 4  
sli.ingestion.queue.maestro.queueURI = activemq:queue:ingestion.maestro  
sli.ingestion.queue.maestro.consumerQueueURI = txActivemq:queue:ingestion.maestro  
sli.ingestion.queue.maestro.concurrentConsumers = 1  
sli.ingestion.queue.maestro.uriOptions = &transferExchange=true  
sli.ingestion.queue.pit.queueURI = activemq:queue:ingestionPit  
sli.ingestion.queue.pit.consumerQueueURI = txActivemq:queue:ingestionPit  
sli.ingestion.queue.pit.concurrentConsumers = 30  
sli.ingestion.queue.pit.uriOptions = &transferExchange=true  
sli.ingestion.nodeType = standalone  
sli.ingestion.tenant.loadDefaultTenants = true  
sli.ingestion.tenant.tenantPollingRepeatInterval = 5s  
sli.ingestion.cache.type = inmemory  
sli.ingestion.cache.servers =   
sli.ingestion.cache.opTimeout = 10  
sli.ingestion.split.chunk.size = 500  
sli.ingestion.split.threshold.percentage = 0.3  
sli.ingestion.referenceSchema.referenceCheckEnabled = false  
sli.ingestion.errorsCountPerInterchange = 10000  
sli.ingestion.warningsCountPerInterchange = 10000  
sli.ingestion.totalRetries = 5  
sli.ingestion.dataset.sample = {"small":["SmallSampleDataSet.zip"],"medium":["MediumSampleDataSet.zip"]}  
sli.ingestion.file.timeout = 600000  
sli.ingestion.file.retryinterval = 30000  
sli.ingestion.queue.landingZone.keystore = /opt/tomcat/encryption/tomcat.keystore  
sli.ingestion.queue.landingZone.keystorePassword = [DAL\_KEYSTORE\_PASS]  
sli.ingestion.queue.landingZone.queueURI = activemq:queue:ingestion.landingZone  
sli.ingestion.queue.landingZone.concurrentConsumers = 1  
api.client = apiClient  
# api.server.url and security.server.url MUST to be set to the FQDN of the API.  
api.server.url = https://rcapi.slidev.org/  
security.server.url = https://rcapi.slidev.org/  
# Portal.footer.url and portal.header.url are Dashboard settings that are utilize to help render the GUI with Portal integration.  NOTICE, FQDN must be modified to match the environment.  
portal.footer.url = https://rcportal.slidev.org/headerfooter-portlet/api/secure/jsonws/headerfooter/get-footer  
portal.header.url = https://rcportal.slidev.org/headerfooter-portlet/api/secure/jsonws/headerfooter/get-header  
dashboard.log.level = warn  
# The oauth. settings are utilized by Dashboard for OAUTH.  
oauth.client.id = [DASHBOARD\_CLIENT\_ID]  
oauth.client.secret = [DASHBOARD\_CLIENT\_SECRET]  
oauth.redirect = https://rcdashboard.slidev.org/dashboard/callback  
panel.config.driver.dir = config  
panel.config.custom.dir = custom  
# Dashboard settings generally do not require modification.  
dashboard.google\_analytics.id = [GOOGLE\_ANALYTICS\_ID]  
dashboard.WSCall.timeout = 3000  
dashboard.minify.js = true  
dashboard.cache.disable = false  
dashboard.encryption.keyStorePass = [DAL\_KEYSTORE\_PASS]  
dashboard.encryption.dalKeyAlias = dalKey  
dashboard.encryption.dalKeyPass = [DAL\_KEY\_PASS]  
dashboard.encryption.keyStore = ../data-access/dal/keyStore/ciKeyStore.jks  
# Bootstrap settings are below.  URL FQDNs, Client IDs and Secrets must be updated.  
# bootstrap.app.keys is a list of the applications to be bootstrapped upon API Start=up.  
bootstrap.app.keys = admin,portal,dashboard,databrowser  
bootstrap.app.admin.template = applications/admin.json  
bootstrap.app.admin.name = Admin Apps  
bootstrap.app.admin.description = The inBloom Administration Application allows you to change a variety of system settings.  
bootstrap.app.admin.version = 0.0  
bootstrap.app.admin.authorized\_for\_all\_edorgs = true  
bootstrap.app.admin.allowed\_for\_all\_edorgs = true  
bootstrap.app.databrowser.template = applications/databrowser.json  
bootstrap.app.databrowser.name = inBloom Data Browser  
bootstrap.app.databrowser.description = The inBloom Data Browser allows developers and administrators to access all available information in the inBloom datastore.  
bootstrap.app.databrowser.version = 0.0  
bootstrap.app.databrowser.authorized\_for\_all\_edorgs = true  
bootstrap.app.databrowser.allowed\_for\_all\_edorgs = true  
bootstrap.app.dashboard.name = inBloom Dashboards  
bootstrap.app.dashboard.description = The inBloom Dashboards allow you to see information about students in lists and profiles.  
bootstrap.app.dashboard.template = applications/dashboard.json  
bootstrap.app.dashboard.version = A.0  
bootstrap.app.dashboard.authorized\_for\_all\_edorgs = true  
bootstrap.app.dashboard.allowed\_for\_all\_edorgs = true  
bootstrap.app.portal.name = Portal  
bootstrap.app.portal.description = The inBloom Portal application is the primary access portal.  
bootstrap.app.portal.version = 0.0  
bootstrap.app.portal.template = applications/portal.json  
bootstrap.app.portal.url = https://rcportal.slidev.org/portal  
bootstrap.app.portal.client\_id = [PORTAL\_CLEARTEXT\_CLIENT\_ID]  
bootstrap.app.portal.client\_secret = [PORTAL\_CLEARTEXT\_CLIENT\_SECRET]  
bootstrap.app.portal.authorized\_for\_all\_edorgs = true  
bootstrap.app.portal.allowed\_for\_all\_edorgs = true  
bootstrap.app.vendor = inBloom  
bootstrap.app.admin.client\_secret = [ADMIN\_CLEARTEXT\_CLIENT\_SECRET]  
bootstrap.app.admin.client\_id = [ADMIN\_CLEARTEXT\_CLIENT\_ID]  
bootstrap.app.admin.url = https://rcadmin.slidev.org  
bootstrap.app.databrowser.client\_secret = [DATABROWSER\_CLEARTEXT\_CLIENT\_SECRET]  
bootstrap.app.databrowser.client\_id = [DATABROWSER\_CLEARTEXT\_CLIENT\_ID]  
bootstrap.app.databrowser.url = https://rcdatabrowser.slidev.org  
bootstrap.app.dashboard.client\_secret = [DASHBOARD\_CLEARTEXT\_CLIENT\_SECRET]  
bootstrap.app.dashboard.client\_id = [DASHBOARD\_CLEARTEXT\_CLIENT\_ID]  
bootstrap.app.dashboard.url = <https://rcdashboard.slidev.org/dashboard>  
sli.encryption.ldapKeyAlias: [LDAP\_KEY\_ALIAS]  
sli.encryption.ldapKeyPass: [LDAP\_KEY\_PASSWORD]  
sli.encryption.keyStorePass = [DAL\_KEYSTORE\_PASSWORD]  
sli.encryption.dalKeyAlias = dalKey  
sli.encryption.dalKeyPass = [DAL\_KEY\_PASSWORD]  
sli.encryption.dalInitializationVector = aabbccddeeff11223344556677889900  
sli.wildcard.x509certificate.alias = wildcard  
# sli.simple-idp.issuer-base is the base SimpleIDP URL  
sli.simple-idp.issuer-base = https://rcidp01ext.slidev.org/sliidp  
# sli.simple-idp.cot contains the information defining the IDP Circle of trust.  The FQDN of the API must be updated.  
sli.simple-idp.cot = https://rcapi.slidev.org=https://rcapi.slidev.org/api/rest/saml/sso/post  
# The next five settings do not need to be modified.   
sli.simple-idp.sandbox.users= SmallDatasetUsers,Small Sample Dataset,MediumDatasetUsers,Medium Sample Dataset  
sli.simple-idp.userSearchAttribute = uid  
sli.simple-idp.userObjectClass = inetOrgPerson  
sli.simple-idp.groupSearchAttribute = memberUid  
sli.simple-idp.groupObjectClass = posixGroup  
sli.simple-idp.sliAdminRealmName = SLIAdmin  
# sli.simple-idp.sandboxImpersonationEnabled must be set to true for Sandbox logins to be enabled.  
sli.simple-idp.sandboxImpersonationEnabled = false  
# sli.simple-idp.ldap.urls is the list of LDAP servers that the Simple IDP can utilize for authenticaiton.  
sli.simple-idp.ldap.urls = ldaps://rcldap01.slidev.org/  
# sli.simple-idp.ldap.base is the search base for the SimpleIDP.  
sli.simple-idp.ldap.base = ou=rcEnvironment,dc=slidev,dc=org  
########## Settings below this line do not need to be edited under normal circumstances.  
sli.dev.subdomain = ERRORNOTUSED.slidev.org  
sli.log4j.rootLogger = INFO, out  
sli.log4j.logger.org.apache.camel = WARN  
sli.log4j.logger.org.apache.activemq = WARN  
sli.log4j.logger.org.springframework = WARN  
log.path = /var/log/tomcat/  
# sli.sample configuration options are utilized by the Sample Application, and do not need to be edited for any production use.  
sli.sample.apiUrl = http://local.slidev.org:8080/  
sli.sample.callbackUrl = http://local.slidev.org:8081/sample/callback  
sli.sample.clientId = fm67sH6vZZ  
sli.sample.clientSecret = sb70uDUEYK1IkE5LB2xdBkTJRIQNhBnaOYu1ig5EZW3UwpP4  
# sli.sif-agent confiugraiton is not presently utilized.  
sli.sif-agent.agentId = test.subscriber.agent  
sli.sif-agent.agentPort = 25100  
sli.sif-agent.zoneId = TestZone  
sli.sif-agent.zoneUrl = http://local.slidev.org:8087/mock-zis/zis  
sli.sif-agent.adk.logFile = sif-openadk.log  
sli.sif-agent.idmap = default-idmap.csv  
sli.sif-agent.zonemap = default-zonemap.csv  
# bootstrap.app.sif settings are presently not required.  
bootstrap.app.sif.name = SIF Agent  
bootstrap.app.sif.description = SIF Agent  
bootstrap.app.sif.template = applications/sif.json  
bootstrap.app.sif.version = 0.0  
bootstrap.app.sif.url = http://local.slidev.org:1338/  
bootstrap.app.sif.apiUrl = http://local.slidev.org:8080/  
bootstrap.app.sif.callbackUrl = http://local.slidev.org:8081/  
bootstrap.app.sif.guid = 2ad39ff1-65f8-4a16-8912-b49872f1ee97  
bootstrap.app.sif.token = e4e9d71c-d674-11e1-9ea4-f9fc6188709b  
bootstrap.app.sif.client\_id = [redacted]  
bootstrap.app.sif.client\_secret = [redacted]  
bootstrap.app.sif.authorized\_for\_all\_edorgs = true  
bootstrap.app.sif.allowed\_for\_all\_edorgs = true  
# The configuration below needs to be added to API and Search Indexer config file.  
sli.search.url = http://[PRIVATE\_ELASTICSEARCH\_LOADBALANCER]:9200  
sli.search.username = [SEARCH\_USERNAME]  
sli.search.password = [SEARCH\_USER\_PASSWORD]  
sli.search.encryption = true  
sli.search.embedded = false  
sli.search.service.timeout = 3000  
sli.search.indexer.log.path = logs  
sli.search.indexer.log.level = INFO  
sli.search.indexer.dir.inbox = data/inbox  
sli.search.indexer.dir.extract = data/tmp  
sli.search.indexer.extract.schedule = 0 0 0 ? \* SUN  
sli.search.indexer.extract.runOnStartup = false  
sli.search.indexer.service.port = 10024  
sli.search.indexer.broker.uri = tcp://[ACTIVEMQ\_SERVER\_NAME]:61616  
sli.search.indexer.broker.username = system  
sli.search.indexer.broker.password = manager  
sli.search.indexer.queue.name = search  
sli.search.indexer.sarje.sub.topic = /topic/oplog\_subscribe  
sli.search.indexer.sarje.sub.req.queue = /queue/subscription/poll  
sli.search.indexer.env = local  
#Items below this line are added to the API configuration which are for the new Application Developer Production Realm account federation feature.  
bootstrap.developer.realm.name = inBloom App Developers  
bootstrap.developer.realm.uniqueId = DeveloperIDP  
bootstrap.developer.realm.idpId = https://sandboxidp.slidev.org/sliidp?realm=SLIAdmin&developer=true  
bootstrap.developer.realm.redirectEndpoint = <https://sandboxidp.slidev.org/sliidp?realm=SLIAdmin&developer=true>  
# This item was added for use by API and Ingestion.  
sli.security.writeValidation = true  
sli.ingestion.queue.parser.queueURI: activemq:queue:IngestionParser  
sli.ingestion.queue.parser.concurrentConsumers: 8  
sli.ingestion.queue.parser.uriOptions:

## Portal Application sli.properties

# The security.server.url and api.server.url settings point to the root of the REST API server.  The FQDN must be updated to match the environment.  
security.server.url=https://rcapi.slidev.org/  
api.server.url=https://rcapi.slidev.org/  
# The portal.oauth.client.id and portal.oauth.client.secret settings are the same settings that the portal application is bootstrapped with via the API sli.properties file.  
portal.oauth.client.id=[PORTAL\_CLIENT\_ID]  
portal.oauth.client.secret=[PORTAL\_CLIENT\_SECRET]  
# The portal.oauth.encryption setting tells the application if the above settings are encrypted utilizing the encryption tool.  
portal.oauth.encryption=true  
  
portal.oauth.redirect=https://rcportal.slidev.org/portal/login  
log.path = /opt/portal  
log.level = INFO  
sli.google\_analytics.id = [GOOGLE\_ANALYTICS\_ID]  
sli.domain = slidev.org  
  
api.client=apiClient  
# These settings have been depricated, and renamed to the portal.oauth settings.  
oauth.encryption=true  
oauth.client.id=[REDACTED]  
oauth.client.secret=[REDACTED]  
oauth.encryption=true  
oauth.redirect=https://rcportal.slidev.org/portal/login

## portal-ext.properties

# The following settings define the JDBC Parameters for connecting Liferay to a database.  The url, username, and password will need to match the environment configuration.  
jdbc.default.driverClassName=com.mysql.jdbc.Driver  
jdbc.default.url=jdbc:mysql://rcmysql01.slidev.org/lportal?emulateLocators=true&useUnicode=true&characterEncoding=UTF-8&useFastDateParsing=false&createDatabaseIfNotExists=true&useSSL=true&requireSSL=true  
jdbc.default.username=[LIFERAY\_DB\_USER]  
jdbc.default.password=[LIFERAY\_DB\_PASS]  
# The setting below governs if the two settings above are encrypted or not.  The settings can be encrypted utilizing the encryption toolkit that is provided with the software packages.  
jdbc.default.encrypted.password=false  
# The following setting governs if Liferay is clustered with multiple servers via Tomcat Clustering.  Set to false if the server is a stand-alone server without Tomcat clustering.  
cluster.link.enabled=true  
# The following timeout is the Portal Session Timeout in minutes.  
session.timeout=30  
mail.session.mail.smtp.host=email-smtp.us-east-1.amazonaws.com  
mail.session.mail.smtp.port=587  
mail.session.mail.smtp.auth=true  
mail.session.mail.smtp.user=[ENCRYPTED\_MAIL\_USERNAME]  
mail.session.mail.smtp.password=[ENCRYPTED\_MAIL\_PASSWORD]  
mail.session.mail.transport.protocol=smtp  
mail.session.mail.smtp.credential.encryption=true  
# sli.cookie.domain is the setting for at what context the application should issue cookies for browser storage.  
sli.cookie.domain=.slidev.org  
# The following setting informs Portal if the application is in Sandbox mode.  This setting MAY no longer be utilized.  
is\_sandbox = false  
# The following settings requires that the sli.encryption.properties variable is defined on the Tomcat Command line.  This is utilized to decrypt encrypted configuration.  
include-and-override=\${sli.encryption.properties}  
########## The following setting below this line should not need to be modified.  
portal.ctx=/portal  
users.reminder.queries.enabled=false  
users.reminder.queries.custom.question.enabled=false  
users.reminder.queries.required=false  
system.roles=SLI Administrator, Educator  
setup.wizard.enabled=false  
portlet.event.distribution=layout-set  
portlet.public.render.parameter.distribution=ALL\_PORTLETS  
org.slc.sli.login.servlet.filter.sso.SLIFilter=true  
users.screen.name.validator=com.liferay.portal.security.auth.LiberalScreenNameValidator  
users.email.address.required=false  
terms.of.use.required=true  
sli.sso.logout.on.session.expiration=true  
sso.login.error.page=/portal/web/guest/error  
sli.role.itadmin=IT Administrator  
sli.role.sliadmin=SLI Administrator  
sli.role.educator=Educator  
sli.role.admin=LEA Administrator,Realm Administrator,IT Administrator,SEA Administrator,SLC Operator,Application Developer,Ingestion User,SEA Super Administrator,LEA Super Administrator,Realm Admin,App Developer  
sli.role.liferayadmin=SLC Operator  
wsrp.page=/portal/web/guest/wsrp  
iframe.page=/portal/web/guest/iframe  
template.processing.enabled=false  
image.menu\_arrow=menu\_arrow1.png  
image.arrow=arrow.png  
image.arrow\_w=arrow\_w.png  
image.sli\_logo=inBloomBlack.png  
layout.user.public.layouts.enabled=false  
layout.user.public.layouts.modifiable=true  
layout.user.public.layouts.auto.create=false  
# IMPORT LAR APPROACH  
layout.lar.file.name=layout.lar  
auto.deploy.deploy.dir=/opt/deploy  
# Set path for error page  
layout.friendly.url.page.not.found=/web/guest/error  
org.quartz.jobStore.isClustered=true  
net.sf.ehcache.configurationResourceName=/ehcache/hibernate-clustered.xml  
ehcache.multi.vm.config.location=/ehcache/liferay-multi-vm-clustered.xml  
dl.store.impl=com.liferay.portlet.documentlibrary.store.JCRStore  
lucene.replicate.write=false   
lucene.store.jdbc.auto.clean.up.enabled=true  
lucene.store.type=jdbc  
layout.user.private.layouts.enabled=false  
layout.user.private.layouts.modifiable=true  
layout.user.private.layouts.auto.create=false  
auto.deploy.copy.log4j=false  
browser.cache.signed.in.disabled=true  
http.header.version.verbosity= partial  
session.timeout.warning=0

## Admin config.yml

production:  
   # NOTICE: The vast majority of these settings will need to be modified to meet the environment that is being utilized.  
   # api\_base is the URL to the REST API  
   api\_base: https://rcapi.slidev.org/api/rest  
   # client\_id and client\_secret are the credentials utilized for   
   client\_id: [ADMIN\_CLIENT\_ID]  
   client\_secret: [ADMIN\_CLIENT\_SECRET]  
   # This is the redirect URL from the IDP to the application.  Only the FQDN of the URL should be updated.  
   redirect\_uri: https://rcadmin.slidev.org/callback  
   # The following settings are the settings for connecting to the LDAP server.   
   ldap\_host: rcldap01.slidev.org  
   ldap\_port: 636  
   ldap\_use\_ssl: true  
   ldap\_base: ou=SLIAdmin,ou=rcEnvironment,dc=slidev,dc=org  
   # The user defined by ldap\_user and ldap\_pass MUST NOT be the directory manager.  It must be a standard account with elevated permissions via access/olcAccess.  
   ldap\_user: cn=Admin,dc=slidev,dc=org  
   ldap\_pass: [LDAP\_PASSWORD]  
   # is\_sandbox defines if the Admin applicaiton is running in Sandbox mode or Production mode.  
   is\_sandbox: false  
   # auto\_approve is a setting that controls if Developer Registration requests are automatically approved or not.  
   auto\_approve: true  
   # email\_sender\_address\_user\_reg\_app is the email address that a user registration email is sent from.  
   email\_sender\_address\_user\_reg\_app: [USER\_REGISTRATION\_EMAIL]  
   # The following settings are the API keys for Google Recaptcha  
   recaptcha\_pub: [RECAPTCHA\_KEY\_ID]  
   recaptcha\_priv: [RECAPTCHA\_PRIVATE\_KEY]  
   # The password\_policy setting is the text returned to the user upon the LDAP server responding that the supplied password is insufficent.  This MUST be changed to a text string that defines the implemented password policy.  
   password\_policy: ["password must satisfy password policy"]  
   # The following settings are settings governing the application sending electronic email for notices.  
   support\_email: [SUPPORT\_EMAIL\_ADDRESS]  
   email\_sender\_name: inBloom Administrator  
   email\_sender\_address: [SUPPORT\_EMAIL\_ADDRESS]  
   email\_host: email-smtp.us-east-1.amazonaws.com  
   email\_port: 587  
   email\_username: [AMAZON\_SES\_USER]  
   email\_password: [AMAZON\_SES\_PASS]  
   email\_tls: true  
   ############ Review settings below this point, however changes MAY not be necessary.    
   # The following URLs are utilized for links within the Admin application, or for links embedded in emails.  They may need to be modified.  
   portal\_url: https://rcportal.slidev.org/portal   
   email\_replace\_uri: https://rcadmin.slidev.org  
   admin\_documentation\_link: https://rcadmin.slidev.org/documentation/link  
   app\_dev\_documentation\_link: http://dev.inBloom.org/getting-started  
   redirect\_slc\_url: http://www.inBloom.org  
   sample\_data\_url: http://dev.inBloom.org/getting-started/sandbox  
   sample\_data\_url: http://www.inBloom.org  
   terms\_url: http://dev.inBloom.org/legal/terms-of-use  
   privacy\_policy\_url: http://dev.inBloom.org/legal/privacy  
   ############ Settings below this point should not need to be changed under normal circumstances.  
   # Setting recaptcha\_disable to true disables captcha validation.  
   recaptcha\_disable: false  
   # Do not modify the admin\_realm text unless it exactly matches the API settings.  
   admin\_realm: Shared Learning Infrastructure  
   email\_sender\_name\_user\_reg\_app: inBloom Administrator  
   reset\_password\_lifespan: 86400  
   # maximum\_user\_count is the maximum number of registered users that can be permitted into Sandbox.  
   maximum\_user\_count: 500  
   encryption\_keyfile: /path/to/rcRailsKey  
   encryption\_iv: [IV\_VALUE\_FROM\_KEY\_GENERATION  
   devslc\_edu: [http://dev.inbloom.org](http://dev.inbloom.org/)

## Data browser config.yml

production:  
   #Notice: All of the settings below MUST be modified to meet the settings of the environment.  
   # api\_base must be set to the URL to access the REST API.  
   api\_base: https://rcapi.slidev.org/api/rest/v1  
   # The client\_id and client\_secret settings are the settings utilized for application/API interaction.  You would utilize the values set as part of bootstrapping the Databrowser application from the sli.properties file.  
   client\_id: [DATABROWSER\_CLIENT\_ID]  
   client\_secret: [DATABROWSER\_CLIENT\_SECRET]  
   # The redirect\_uri setting should be static with the exception of the FQDN of the application.  It is the URL that is passed to the IDP for the redirect back to the application.  
   redirect\_uri: https://rcdatabrowser.slidev.org/callback  
   # The portal\_url is the setting for the Portal Header/Footer.  Only the FQDN of the server name should change in this setting.  
   portal\_url: https://rcportal.slidev.org/headerfooter-portlet/api/secure/jsonws/headerfooter

## Apache Tomcat

For ingestion Tomcat instances, Tomcat’s command line needs to have garbage collection settings changed to ensure reliable batch job runs. Below is an example addition of these properties on the java command line:

-Xms40G -Xmx40G -XX:+UseParallelGC -XX:PermSize=512m -XX:MaxPermSize=512m

Additionally, Tomcat connectors for the API should have the following setting appended to the connector statement:

maxHttpHeaderSize="16384"

## ActiveMQ

ActiveMQ Configuration Blocks from the default configuration, conf/activemq.xml file. The bolded lines below contain the suggested settings.

Note: Between Release 6.5 and 6.8, the producerFlowControl was changed from “true” to “false”.

<destinationPolicy>

<policyMap>

<policyEntries>

**<policyEntry topic=">" producerFlowControl="false" memoryLimit="750mb">**

<pendingSubscriberPolicy>

<vmCursor />

</pendingSubscriberPolicy>

</policyEntry>

**<policyEntry queue=">" producerFlowControl="false" memoryLimit="750mb">**

</policyEntry>

</policyEntries>

</policyMap>

</destinationPolicy>

<systemUsage>

<systemUsage>

<memoryUsage>

**<memoryUsage limit="1gb"/>**

</memoryUsage>

<storeUsage>

<storeUsage limit="100 gb"/>

</storeUsage>

<tempUsage>

<tempUsage limit="50 gb"/>

</tempUsage>

</systemUsage>

</systemUsage>

<transportConnectors>

<transportConnector name="openwire" uri="tcp://0.0.0.0:61616"/>

**<transportConnector name="stomp" uri="stomp://0.0.0.0:61613"/>**

</transportConnectors>