SHRINE 3.0.0 Chapter 8.4 - Configuring a Hub

> **Hub Admins Only**
>
> This section is intended for hub administrators only. For the ACT network, the hubs have already been set up and hosted at Harvard Catalyst, and remote sites should skip this section.

Here is a sample shrine.conf file for a system running SHRINE 3.0.0 , for a node supporting researchers and distributing queries. Note that the adapter section sets create = false; most real hubs do not have collections of patient data.

```
shrineHubBaseUrl = "https://localhost:6443" //The shrine hub's URL as observed from this tomcat server
i2b2BaseUrl = "http://i2b2.example.com:9090" //The local i2b2's URL as observed from this tomcat server
i2b2Domain = "exampleDomain"
i2b2ShrineProjectName = "SHRINE"

shrine {

  nodeKey = "somethingHub" //node key to get information from the hub about itself as a node.

  messagequeue {
    blockingqWebApi {
      enabled = true  //run shrine's MoM system at the hub.
    }
  }

  //shrineDatabaseType = "mysql" // "mysql" by default. It can be "sqlserver" "mysql" or "oracle"

  webclient {
    domain = ${i2b2Domain}
    name  = ${i2b2ShrineProjectName}
    siteAdminEmail = "shrine-admin@example.com"
  }
  pmEndpoint {
    url = ${i2b2BaseUrl}/i2b2/services/PMService/getServices
  }

  ontEndpoint {
    url = ${i2b2BaseUrl}/i2b2/services/OntologyService/
  }

  hiveCredentials {
    domain = ${i2b2Domain}
    username = "demo"
    password = "changeit"
    crcProjectId = "Demo"
    ontProjectId = ${i2b2ShrineProjectName}
  }//hiveCredentials

  breakdownResultOutputTypes {
    PATIENT_AGE_COUNT_XML {
      description = "Age patient breakdown"
    }

    PATIENT_RACE_COUNT_XML {
      description = "Race patient breakdown"
    }

    PATIENT_VITALSTATUS_COUNT_XML {
      description = "Vital Status patient breakdown"
    }

    PATIENT_GENDER_COUNT_XML {
      description = "Gender patient breakdown"
    }
  }//breakdownResultOutputTypes
```

```
  hub {
    create = true
    client {
      serverUrl = ${shrineHubBaseUrl}
    }

    //This part of the configuration is only used if no network is found in the hub's database
    //If no network is foundIt is loaded into the database when the hub starts.
    //To update the network or nodes, use the appropriate curl commands, not this config section.
    //Specified nodes are added only if the node key does not already exist in the hub's database.
    //todo put a reference to those commads as part of SHRINE-3032
    ifNoNetwork {
      network {
        name = "Shrine Dev Test Network" //Name of your network
        hubQueueName = "hub" //queue used to send messages to the hub, different from the queue used to send
messages to a QEP and an adapter colocated with the hub
        adminEmail = "yourname@example.com"
      }
      //Nodes in this network
      //In ifNoNetwork - nodes are added only if the node key does not already exist in the hub's database.
      nodes = [
        {
          name = "Shrine Dev1", //human-readable name for this node
          key = "somethingHub", //machine-friendly key used to identify this node. Never change this.
          userDomainName = "shrine-dev1.catalyst", //domain name for users from this node.
          adminEmail = "yourname@example.shrine-dev1.com", //the email address for the admin of this node
          queueName = "shrineDev1", //queue used to send messages to the qep and adapter at this node. This
field is optional, defautls to the key if not specified
          sendQueries = "false" //true to send queries to an adapter at this node. An optional field, true by
default.
        },
        {
          name = "shrine-dev2",
          key = "shrineDev2",
          userDomainName = "shrine-dev2.catalyst",
          adminEmail = "yourname@example.shrine-dev2.com"
        }
      ]
    }//ifNoNetwork

    networkHealth {
      email {
        networkName = "Enter network name"
        networkSignature = "Enter network signature"
        grantDescription = "Enter description of grant"
        helpContactName = "Enter help contact name"
        helpContactEmail = "yourname@example.com" //The email address for the node admin to contact for help if
there are questions about an error that occurred after running the connectivity test
      }
      webpage {
        networkName = "Enter network name"
        webPageNote = "This is a note" //Optional alert message to display at the bottom of the network health
webpage
        alertMessage = "This is an alert message"  //Optional alert message to display at the top of the
network health webpage
      }
      connectivityTest {
        queryId = "123456789"  //The query Id of a previously run query that will be used to create a new query
for the connectivity test
        interval = "24 hours" //How often to run the connectivity test
        timeLimit = "10 minutes"  //How long to wait for the query to finish running. If the query does not
finish by this time, it will be marked as an error.
        delay = "5 seconds" //The initial delay for starting the connectivity test. If the delay is "5 seconds"
then the connectivity test will run 5 seconds after server startup.
      }
    }//networkHealth
  }//hub

  adapter {
    create = false
```

```
  }//adapter

  keystore {
    file = "/opt/shrine/shrine.keystore"
    password = "changeit"
    privateKeyAlias = "shrine-hub"
    keyStoreType = "JKS"
    caCertAliases = ["shrine-ca"]
  }//keystore

  steward {
    createTopicsMode = Approved //the default is Pending - the most secure - but most sites use Approved

    emailDataSteward {
      //provide the email address of the shrine node system admin, to handle bounces and invalid addresses
      from = "shrine-admin@example.com"
      //provide the email address of the shrine node system admin, to handle bounces and invalid addresses
      to = "shrine-steward@example.com"
      //provide the externally-reachable URL for the data steward
      externalStewardBaseUrl = ${shrineHubBaseUrl}/shrine-api/shrine-steward
    }
  }//steward
}//shrine
```

It is rare but possible to have a set of patient data at the hub. Simply include the adapter section of shrine.conf from Chapter 8 in your shrine.conf .