Database Related Issues

Backlog

Number	Notes	Relates To	Type	Assignee	Rank
~~POLICY-1821~~ - Persistence of run time policy state in Pdps CLOSED	Investigate ways of persisting policy state information with different structures at run time. One approach could be to use Avro & Apache Hive. [There is also a NoSQL/JSON option available in MariaDB and Postgres] Another option is the Apache Cassandra database. Datastax provide a Java Driver for Apache Cassandra. Writing plugins towards state information stored in CPS might be another approach that could work across all the PDPs.	POLICY-2898	Epic		5
~~POLICY-2898~~ - Policy should function in a multi-cluster environment CLOSED	Multi-cluster database support	POLICY-1821	Story		1
~~POLICY-2715~~ - Allow underlying database to be configured: MariaDB or Postgres CLOSED	Allow DB to be changed based on user needs e.g Postgres for MariaDB	POLICY-1787	Story		3 (xacml & drools need to be tested with postgres)
~~POLICY-1787~~ - Support mariadb upgrade /rollback functionality CLOSED	Upgrade/rollback of database tables	POLICY-2715	Story	Kevin Timoney	1 (follows backup /restore JIRA)
~~POLICY-2086~~ - Remove references to mariadb from resource files CLOSED	References to DB should be moved from persistence.xml to properties file to facilitate the use of alternate databases. We can modify the current code to read in the location of a properties file		Story		3

JDBC properties

```
-Djdbc.properties=/path/to/jdbc.properties
 String propertiesPath = System.getProperty(
"jdbc.properties" );

if ( propertiesPath != null )
{
     FileInputStream in = new FileInputStream
( propertiesPath );

     try
     {
         jdbcProperties = Properties.load( in
);
     }
     finally
     {
         in.close( );
     }
}
```

Kubernetes

There are four different ways that you can use a ConfigMap to configure a container inside a Pod:

- Inside a container command and args
- Environment variables for a container
- Add a file in read-only volume, for the application to read
- Write code to run inside the Pod that uses the Kubernetes API to read a ConfigMap

Kubernetes Configmap

☑ ~~POLICY-3059~~ - Fix name of target-database property in persistence.xml files CLOSED	Invalid target-database property in persistence.xml in apex pdp. Persistence.xml should use eclipselink.target-database e.g. <property name="eclipselink.target-database" value="MySQL" />		Task	Ajith Sreekumar - reassign to Kevin	3 (bug fix) **Done**
~~POLICY-1837~~ - Review transaction boundaries of models CLOSED	Currently, the models Provider classes manage transactions. Transaction management should be moved to client for better performance and atomicity. This will also eliminate the need for caching on the client side. Note: The transactional annotation can be used to group transactions together. Ideally transactions should be executed serially in order to avoid dirty reads, non-repeatable reads and phantom reads. Note: DatabasePolicyModelsProviderImpl in the models-provider package provides rest APIs such as createServiceTemplate, updateServiceTemplate and deleteServiceTemplate which in turn call the corresponding APIs from AuthorativeToscaProvider in the models-tosca package..		Story		5 (depends on Spring direction)
~~POLICY-2540~~ - Proper handling of data types in policy-models and policy-api CLOSED	How best to deal with CRUD of data types in policy-api and policy-models. It is possible only to create data types indirectly in policy type create requests. Update and delete of data types is not possible. SimpleToscaProvider contains the following methods not available in AuthorativeToscaProvider : getDataTypes, getCascadedDataTypes, createDataTypes, updateDataTypes and deleteDataType.		Story	Assign to Liam	5
~~POLICY-3231~~ - Implement backup and restore functionality for PolicyDB CLOSED ~~POLICY-3000~~ - Create an ability to recover if policy database gets corrupted CLOSED	Database backup and restore	MDEV-23119	Story		1 rename to backup/restore
~~POLICY-2717~~ - Multi-Tenancy support for Policy Framework CLOSED	What policy components needs to be centralized vs de-centralized and moved to tenant namespaces. (R8 ONAP to support multi-tenancy) Multi-tenancy support in Policy Framework Note: Eclipselink supports Using Table-Per-Tenant Multi-Tenancy		Story		8 (design is available)
~~POLICY-2997~~ - Clean up old records from various DB tables CLOSED	Clean up/Roll up of old DB data. Purge/Archive job should be created and run at a scheduled interval. MariaDB Event Scheduler Postgres pg_cron MariaDB allows for the partitioning of tables : Partitioning Overview		Story		4 (for purge) 6 (roll-up) create separate tickets for purge & roll-up, and separate tickets for operations history and statistics
~~POLICY-3153~~ - Fix Db connection issues in TOSCA control loop CLOSED	Concurrent DB access issues in control loop POC.		Bug	Ramesh Murugan Iyer	bug (related to transaction handling?)
~~POLICY-3209~~ - CLAMP Component Lifecycle Management using Spring Framework CLOSED	Investigate what's involved in switching to spring		Story	Liam Fallon	4
	Externalizing ONAP DBs to a separate namespace We would like to improve the current situation by implementing a two staged deployment: 1) Deploy all required DB engines (can be done using community charts or any user chart) 2) Deploy ONAP components & configure them to make use of those engines This would allow use to basically bring his own database for ONAP no matter if it is running in the same k8s cluster or it has been provided by some DBaaS solution. Additionally it makes our deployment more modular and configurable thus may result in significant footprint savings.				6
☑ ~~POLICY-3156~~ - Review the design of storing of PDP statistics in the DB CLOSED	How much statistics should be stored		Task		

☑ ~~POLICY-3404~~ - Rolling DB errors in log output for API, PAP, and DB components CLOSED	JPA table-creation errors		Task		2

Jira relationships