Oracle Enterprise Repository

CMF Entry Type API

Overview

Description

CMF Entry Types describe metadata that may be attached to assets. CMF Entry Types are identified by an id and a single name string.

Definitions

Validation - When saving a CMF Entry Type, Oracle Enterprise Repository currently validates that:

- The CMF Entry Type name length is in bounds
- The CMF Entry Type name is unique
- When updating a CMF Entry Type, a CMF Entry Type ID is present

Related subsystems

A CMF *Entry* is linked to an asset from the perspective of the asset. CMF *Entry Types* define parameters for these entries.

Additional Import(s) Required

import com.flashline.registry.openapi.entity.MetadataEntryTypeSummary;
import com.flashline.registry.openapi.query.MetadataEntryTypeCriteria;

Use Cases

Use Case: Manipulating CMF Entry Types

Description

- Adding a new CMF Entry Type to Oracle Enterprise Repository.

- Assigning an existing CMF Entry Type to an asset.

Sample Code:

```java
package com.flashline.sample.metadataentrytypeapi;

import java.net.MalformedURLException;
import java.net.URL;
import java.rmi.RemoteException;

import javax.xml.rpc.ServiceException;

import com.flashline.registry.openapi.base.OpenAPIException;
import com.flashline.registry.openapi.entity.AuthToken;
import com.flashline.registry.openapi.entity.MetadataEntryTypeSummary;
import com.flashline.registry.openapi.query.MetadataEntryTypeCriteria;
import com.flashline.registry.openapi.service.v300.FlashlineRegistry;
import com.flashline.registry.openapi.service.v300.FlashlineRegistryServiceLocator;

public class MetadataEntryTypes {
  public static void main(String pArgs[]) throws OpenAPIException, RemoteException,
      ServiceException {
    try {

      /////////////////////////////////////////////////////
      // Connect to Oracle Enterprise Repository
      /////////////////////////////////////////////////////
      URL lURL = null;
      lURL = new URL(pArgs[0]);
      FlashlineRegistry repository = new FlashlineRegistryServiceLocator().getFlashlineRegistry(lURL);

      // /////////////////////////////
      // Authenticate with OER
      // /////////////////////////////
      AuthToken authToken = repository.authTokenCreate(pArgs[1],pArgs[2]);

      // -----------------------------------------
      // Create a new CMF Entry Type
      String newMetadataEntryTypeName = "Sample MetadataEntryType";
      MetadataEntryTypeSummary newMetadataEntryType = repository.metadataEntryTypeCreate(authToken, newMetadataEntryTypeName);
      System.out.println("The new MetadataEntryType id =\"" + newMetadataEntryType.getID() + "\"");

      // -----------------------------------------
      // Find a CMF Entry Type
      MetadataEntryTypeCriteria criteria = new MetadataEntryTypeCriteria();
      criteria.setNameCriteria("Sample");
      MetadataEntryTypeSummary[] metadataEntryTypes = repository.metadataEntryTypeQuery(authToken, criteria);
      long myMetadataEntryTypeID = metadataEntryTypes[0].getID();

      // -----------------------------------------
      // Read a CMF Entry Type
      MetadataEntryTypeSummary readMetadataEntryType = repository.metadataEntryTypeRead(authToken, myMetadataEntryTypeID);
      System.out.println("The MetadataEntryType name =\"" + readMetadataEntryType.getName() + "\"");

      // -----------------------------------------
      // Delete a CMF Entry Type
      repository.metadataEntryTypeDelete(authToken, myMetadataEntryTypeID);
    } catch (OpenAPIException lEx) {
      System.out.println("ServerCode = " + lEx.getServerErrorCode());
      System.out.println("Message    = " + lEx.getMessage());
      System.out.println("StackTrace:");
```

```
      lEx.printStackTrace();
    } catch (RemoteException lEx) {
      lEx.printStackTrace();
    } catch (ServiceException lEx) {
      lEx.printStackTrace();
    } catch (MalformedURLException lEx) {
      lEx.printStackTrace();
    }
  }

}
```