# SDK 4.x CQL to HQL translator dependency on domain model

Currently, the CQL2ParameterizedHQL class takes the following parameters in its constructor:

* DomainTypesInformation
  + Provides information on the Java data type of each attribute of each class involved in the service’s data model.
* RoleNameResolver
  + Resolves role names of associations when they are not provided directly in the CQL query – role name is an optional attribute of Association
* ClassDiscriminatorResolver
  + An interface which provides a means to determine the value which must be passed to the special .class property of data types involved in an inheritance hierarchy
* caseInsensitive Boolean
  + Indicates that the generated HQL queries should make use of the lower() operator to force processing of queries without regard to the case of attribute values.

## Domain Types Information

This is a special XML document (deserialized into Axis beans) which contains information about each class and its attributes, and is used to make decisions about what Java data type to create for attribute parameters used in generated HQL queries. This document is generated by the data service creation style and wizard using the domain model as its original source of information.

## Role Name Resolver

This is a utility class which is constructed using the service’s domain model, and uses information contained in it to determine the role name of associations which don’t explicitly indicate one. In the case of non-existent or ambiguous associations (i.e. more than one association between the parent and associated class), it will return an appropriate error.

## Class Discriminator Resolver

This is an *interface* which determines the value which should be used for a given data type’s .class attribute in HQL. A concrete implementation of this interface exists which uses the HBM files generated by the caCORE SDK to determine the discriminator value, but also requires the caGrid domain model to understand class hierarchies.