
MAJESCO
The Future of Insurance Starts Here
Schemas, Metadata and
DS Resource Tables

AGENDA

- Introduction to Schemas
- Collaboration between different schemas
- Metadata tables
- DS Resource tables
- Transaction Tables

Introduction to Schemas

- There are 5 schemas in the database:
  1) The Policy studio Schema (Policy_Studio_Customercode)
  2) The Customer Schema (MIC_Customercode)
  3) Policy Schema (MIC_Policy_Customercode)
  4) Common Schema
  5) MIC_Admin Schema

- Please visit below link for reference

https://confluence.majesco.com/display/MPCL/Schemas#Schemas-CollaborationbetweendifferentSchemas

Introduction to Schemas

- There are 5 schemas in the database:
  1) The Policy studio Schema (Policy_Studio_Customercode)
  2) The Customer Schema (MIC_Customercode)
  3) Policy Schema (MIC_Policy_Customercode)
  4) Common Schema (Mic_Common_Customercode)
  5) MIC_Admin Schema (MIC_Admin)

- Please visit below link for reference

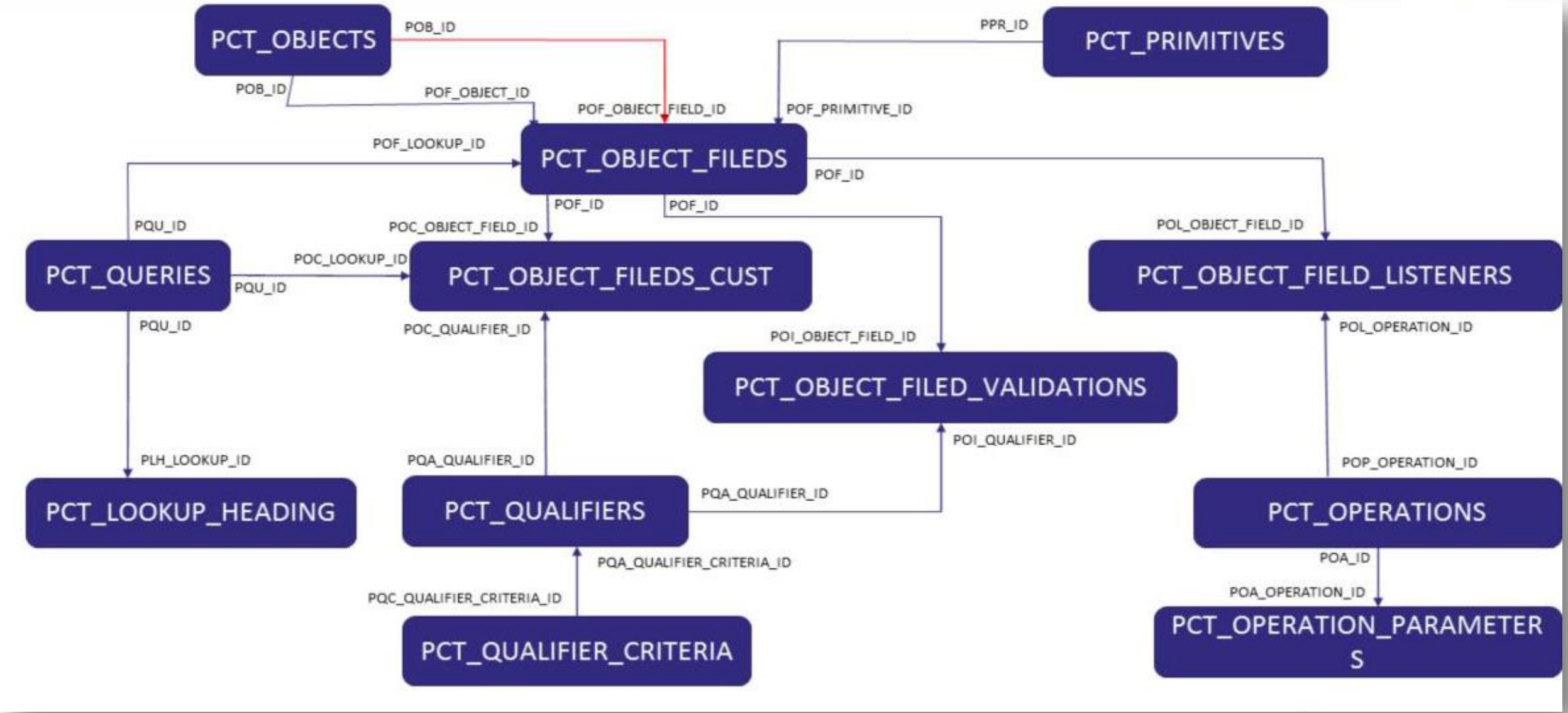
https://confluence.majesco.com/display/MPCL/Schemas#Schemas-CollaborationbetweendifferentSchemas

Introduction to Schemas

1] The Policy studio Schema (Policy_Studio_Customercode)

- This is a primary schema for storing the meta-data. Also called as Metadata schema.
- There are several tables in this schema which are related to Dev Studio like the IEL table which are static tables, the PCT tables which contain the PCT meta data
- Following are the important metadata tables:
  - PCT_OBJECTS
  - PCT_OBJECT_FIELDS
  - PCT_PRIMITIVES
  - PCT_OBJECT_FIELDS_CUST
  - PCT_QUALIFIERS
  - PCT_QUALIFIER_CRITERIA
  - PCT_QUERIES
  - PCT_LOOKUP_HEADING
  - PCT_OBJECT_FIELD_VALIDATIONS
  - PCT_OBJECT_FIELD_LISTENERS
  - PCT_OPERATIONS
  - PCT_OPERATION_PARAMETERS

Introduction to Schemas



MAJESCO

Introduction to Schemas

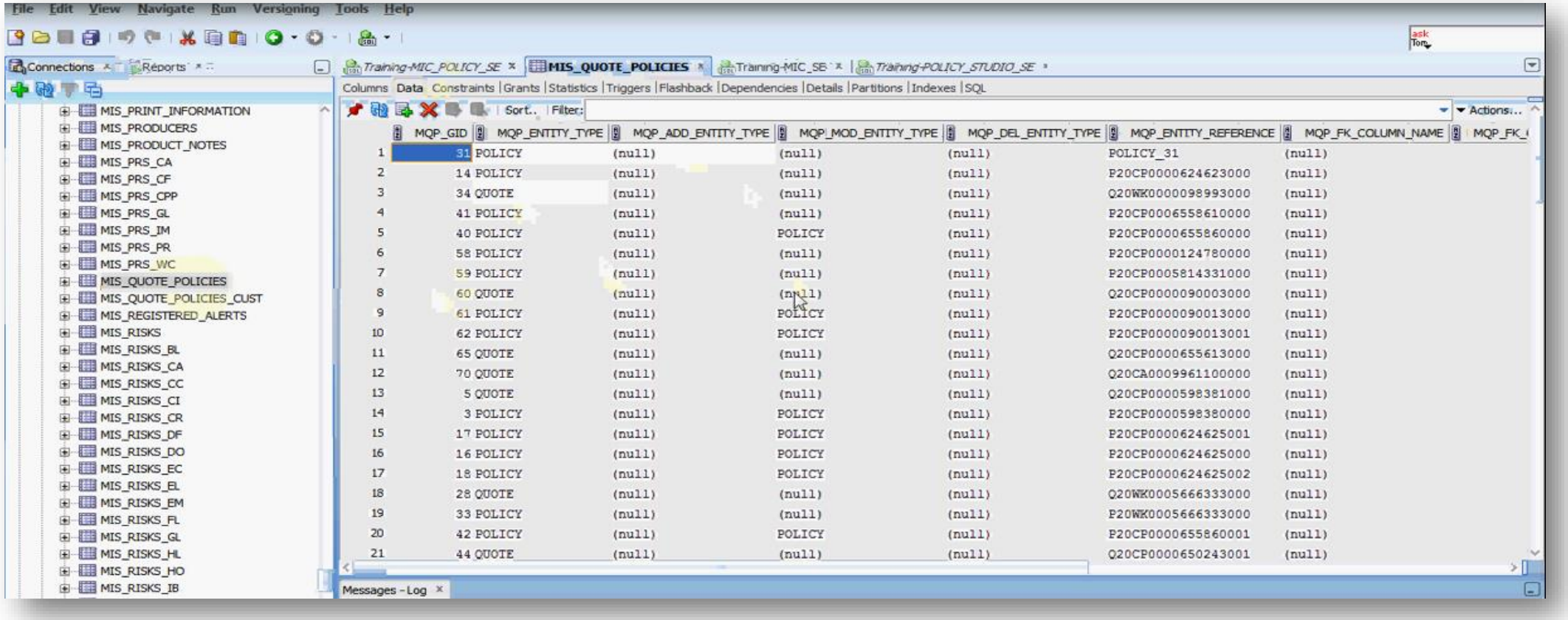
2] The Customer Schema (MIC_Customercode)

- Also called as Security schema.
- This schema is used by the runtime MIC environment to connect to the database. All inserts/updates/deletes to the runtime data happen via the connection obtained for this user thus providing an explicit control over mutations to the actual policy data. There are no tables present in this schema. The tables reside in the policy and other schemas but are accessible via this schema on a need basis via synonyms and restricted access rights.
- The following is an example of the synonyms present in this schema

Introduction to Schemas

3] Policy Schema (MIC_Policy_Customercode) :

- This schema is used to store the run time data which is created by customer for a policy. Tables which are created from meta data are stored in this schema.
- The following is an example of run time data:



MAJESCO

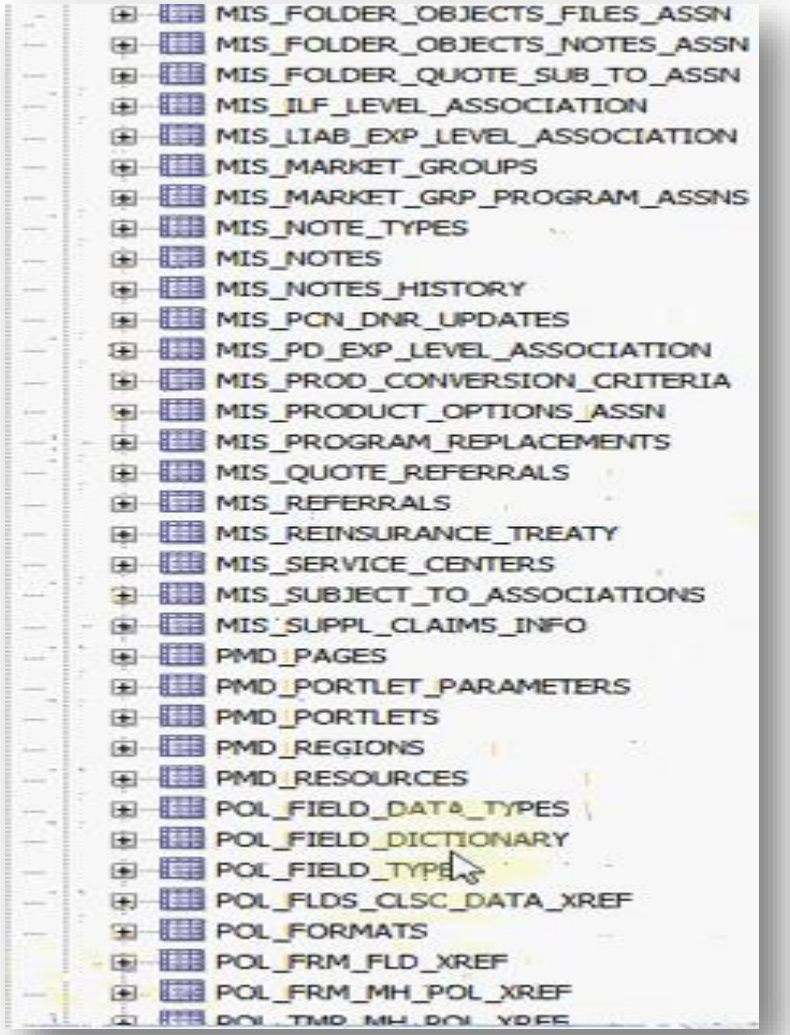
Introduction to Schemas

4] Common Schema (MIC_Common_Customercode)

- This schema contains Meta data related to the policy portals. The metadata contained in this schema relates to the following portal facets
  - Notes
  - Product and Global options
  - Referrals
  - Transactions
  - Producers
  - Portal events and activities
  - Workflow tasks
  - Web services
  - Rule set data

Introduction to Schemas

- Displayed below are some of the tables present in this schema:



MAJESCO

Introduction to Schemas

5) MIC_ADMIN Schema (MIC_ADMIN) :

- It is used to store information regarding the web server and the service box. During the policy creation the web servers are accessed. In the services box, the IP addresses of the services which have to be accessed are stored. The services can be started or stopped from here

Collaboration between different schemas

- An example stating collaboration between different schemas:
  - As the user logs in to the policy application the Security Schema i.e. the CUSTOMER schema is used for authentication and authorization of the user's credentials (user name and password).
  - Once the user is on the policy creation page the schema is accessed. POLICY STUDIO
  - When the user enters the data in the text fields in the policy screen the POLICY schema (Transactional schema) data is used to extract the run time information from the database.
  - Once the policy is complete the COMMON schema is accessed using which the workflows are executed.
  - MIC_ADMIN schema is used to store information regarding the web server and the service box. During the policy creation the web servers are accessed. In the services box, the IP addresses of the services which have to be accessed are stored.

Metadata tables

- Following are the important metadata tables and their significance
  - PCT_OBJECTS :
    - This is one of the most important metadata table that contains records of all Objects and Entities that exists in the system. Every Object/Entity have some properties that are set in this table. For e.g. Policy, Insured, LOB GENERAL LIABILITY, Professional Liability, LIABILITY COVERAGE etc are different objects present in this table.
  - PCT_OBJECT_FIELDS :
    - This metadata table contains list of all the fields/Object used in any particular Object/Entity. For example LOB GENERAL LIABILITY object has multiple fields and objects composed within it(Foreign Key).
    - All the fields and composed object details can be obtained from this table.
    - Also this table contains properties for every fields and composed object.
    - For example : Commercial Property LOB may have a field by name Company which needs to be non-editable. This Updatable property of Company filed can be obtained from this table.
  - PCT_PRIMITIVES :
    - This table contains record of all user defined data types along with their details like the base type, Length, default value for used defined data types, etc.
    - This user defined data types are used while creating new fields on any Object/Entity.
- Reference link: https://confluence.majesco.com/display/MPCL/Metadata

Metadata tables

- PCT_OBJECT_FIELDS_CUST :
  - This table contains records of all the customizations configured at field level.
  - It contains THEN part of configuration rule.
  - It is having relation with PCT_OBJECT_FIELDS table.
- PCT_QUERIES :
  - This table contains records of all the queries used at field level.
  - It is having relation with PCT_OBJECT_FIELDS_CUST, PCT_OBJECT_FIELDS, PCT_LOOKUP_HEADING.
- PCT_LOOKUP_HEADING :
  - This table contains column header name to be displayed for columns which are selected in query of customization.
  - The header is used to display some domain specific names to the columns retrieved in select query.
  - This table is related with PCT_QUERIES table via LOOKUP_ID.

Metadata tables

- PCT_QUALIFIERS：
  - This table is used for configuration rule and validation.
  - It stores data for the WHEN section of any configuration rule.
  - WHEN section of rule contains data like Control Date, Product, Company, State, Control by Adoption, etc.
  - It is related to PCT_OBJECT_FIELDS_CUST via QUALIFIER_ID.
- PCT_QUALIFIER_CRITERIA：
  - This table contains query specified at ADVANCED CONDITION of any configuration rule.
  - The result of this query should return Y if customization rule needs to be executed.
  - It is related to PCT_QUALIFIERS via QUALIFIER_CRITERIA_ID.
- PCT_OBJECT_FIELD_VALIDATIONS ：
  - This table contains THEN part of validation rule written on field of object.
  - It contains TRIGGERING_POINT of this validation rule, ERROR_TYPE and ERROR_MESSAGE to be displayed on successful execution of this rule.
  - This table is related with PCT_QUALIFIERS table via QUALIFIER_ID and related to PCT_OBJECT_FIELDS via OBJECT_FIELD_ID.

Metadata tables

- PCT_OBJECT_FIELD_LISTENERS：
  - This table stores the ACTION rule details of every field.
  - It stores the listener type i.e. ON_CHANGE, ON_CLICK of the field.
  - It is related to PCT_QUALIFIERS table via QUALIFIER_ID and to PCT_OBJECT_FIELDS via OBJECT_FIELD_ID.
- PCT_OPERATIONS：
  - This table contains THEN part of ACTION rule.
  - It specifies what action need to be performed on particular listener Type(Event).
  - It is related to PCT_OBJECT_FIELD_LISTENERS via OPERATION_ID.
- PCT_OPERATION_PARAMETERS：
  - This table contains parameters required for execution of any OPERATION specified in PCT_OPERATIONS .

DS Resource tables

- Dev Studio tables starts with DS and stores activities performed on DEV STUDIO during Product configuration:
  - **DS_RESOURCE:**
    - This table captures record for every resource created on Dev Studio.
    - Resource can be defined as any thing created on Dev Studio. It can be Issue(Ticket), Document, Configuration rule, Validation Rule, Action Rule, Object, Build, Install, etc. identified by RESOURCE_TYPE column.
    - It specifies whether the resource was created for any specific LOB or specific PRODUCT.
    - RESOURCE_OWNER column specifies if the resource was created in BASE or it was created by any CUSTOMER. Null values refers to resource creation in Base.
    - It contains a unique ID called GID which is related to unique ID of PCT tables. For e.g. if a resource of Type Object is created then an entry would be made in DS_RESOURCE(having GID)also an entry would be made in PCT_OBJECTS(having POB_ID). Here GID of DS_RESOURCE will be related to POB_ID of PCT_OBJECTS.
    - It has one more column named DSR_UUID which contains unique ID and has same value in SOURCE as well as TARGET environment. Whereas GID can differ because of different sequence no on different environment.
  - Reference link : https://confluence.majesco.com/display/MPCL/DS+Resource+Tables

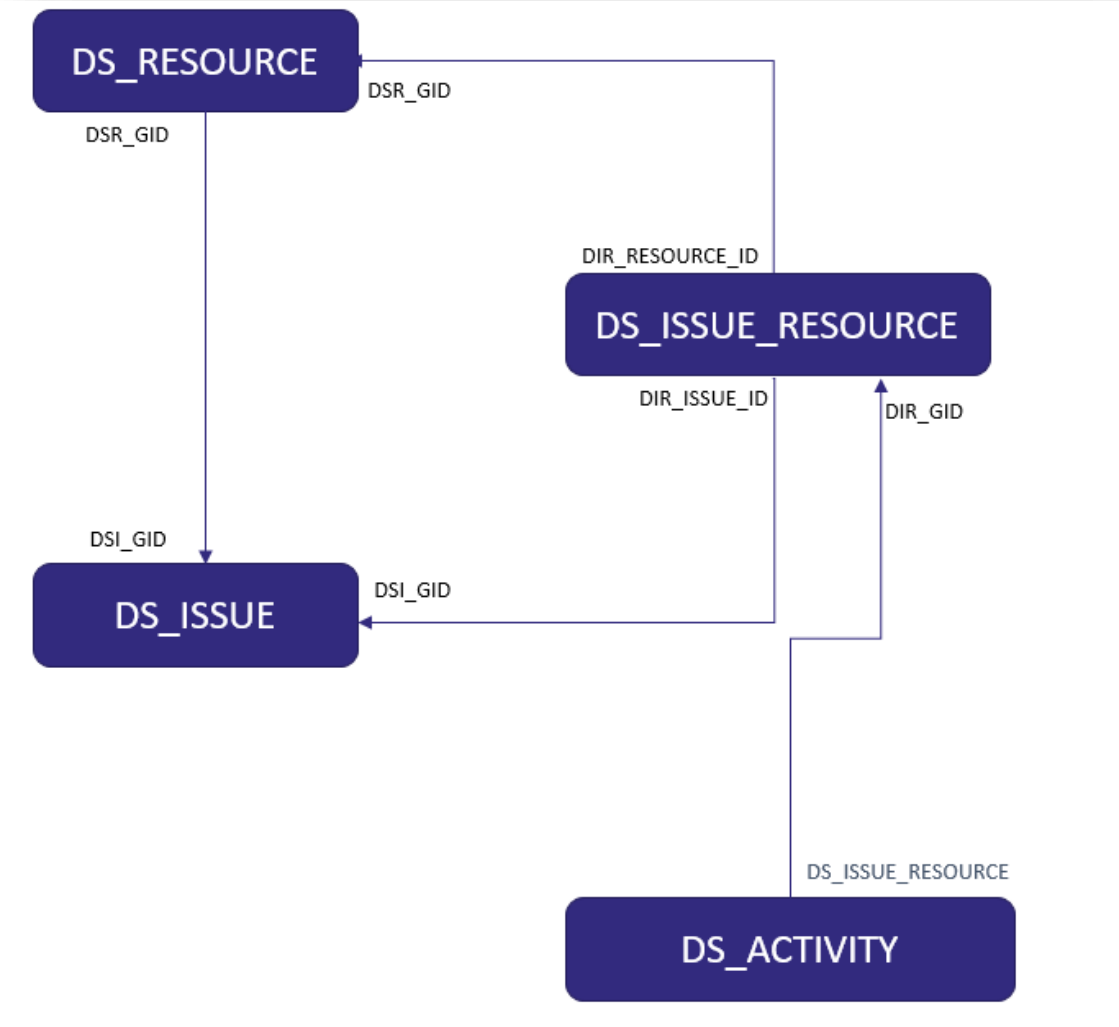
DS Resource tables

- DS_ISSUE :
  - This table contains list of all the issues(tickets) created on Dev Studio.
  - If a new resource is to be created or an existing resource is to be modified then Dev Studio permits it only after taking that resource in issue(Ticket).
  - It is related to DS_RESOURCE table via GID.

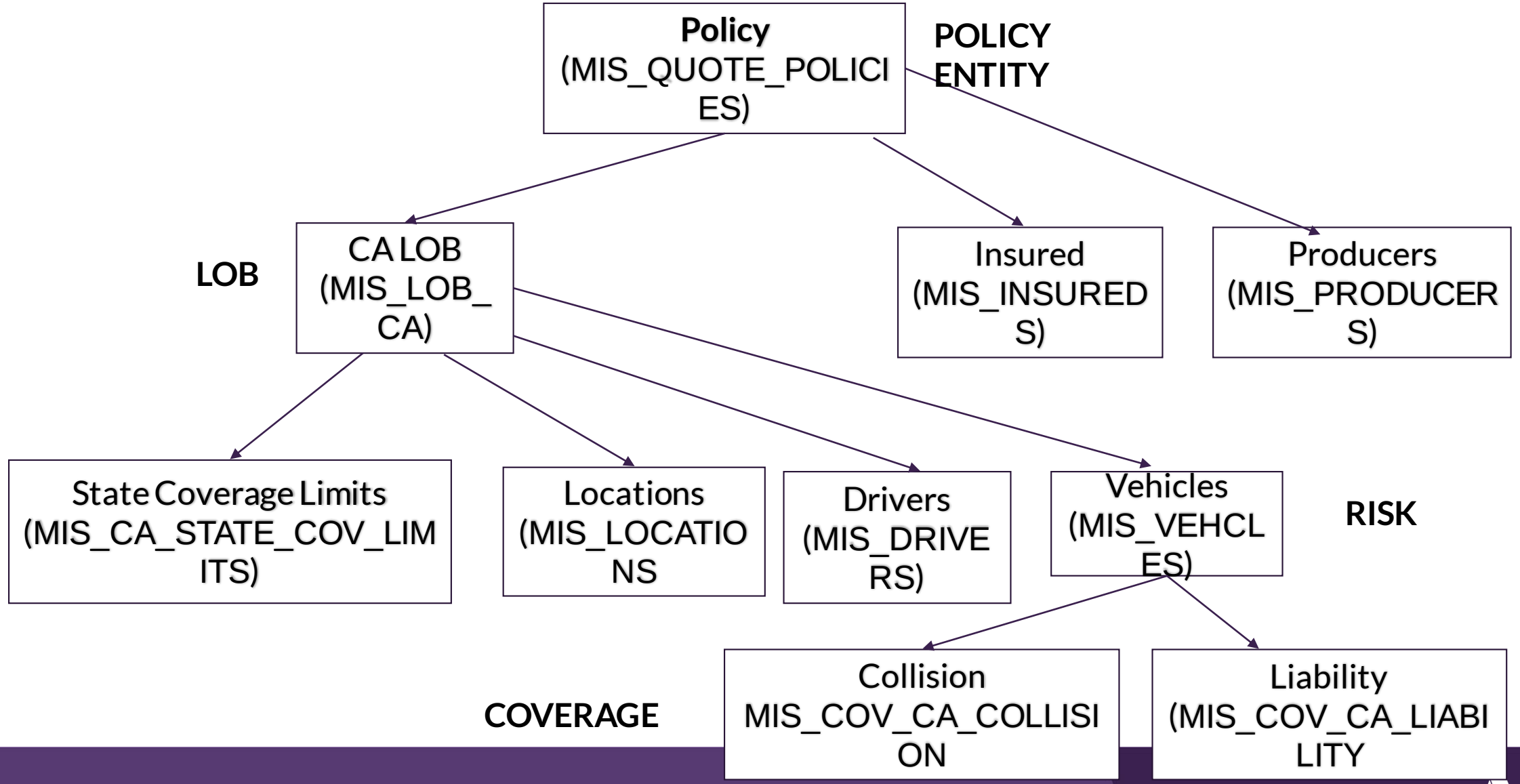
DS Resource tables

- DS_ISSUE_RESOURCE:
  - This table contains the mapping of issue(ticket) and corresponding resources.
  - The resources used in any tickets can be traced using this table.
  - It is related to DS_ISSUES table via DIR_ISSUE_ID and GID of DS_ISSUE table and related to DS_RESOURCE via DIR_RESOURCE_ID and GID of DS_RESOURCE.
- DS_ACTIVITY :
  - This table specifies the activity performed on any particular resource under any issue(ticket).
  - The activity can be defined as modification under a rule configuration, addition of a new rule,modification of field, addition of new field, etc.
  - This table also specifies the modified/added resource under DTA_RESOURCE_NAME column and DTA_DESCRIPTION gives information regarding the activity performed.
  - It is related to DS_ISSUE_RESOURCE table via DTA_ISSUE_RESORCE and DIR_GID of DS_ISSUE_RESOURCE.

DS Resource tables



Transactions Table Hierarchy



MAJESCO

MAJESCO

THANK YOU!