MAJESCO

The Future of Insurance Starts Here

Context-Path (CP) Expressions

AGENDA

- Why CP Expression
- What are CP Expressions
- When Not to use CP Expression
- Applications of CP Expressions
- How to write CP Expression
  - Syntax
  - Examples
- Debugging CP Expression

Why CP Expressions?

- Developers need to write SQL queries while configuring application in Dev Studio.
- SQL statements may need developer to specify relationship between the tables
- CP is used for dependency creation.
  - CP knows it depends upon which columns, if value of any of the column used in CP changes, CP gets invalidated and runs again. When CP is resolved, we can see the dependent fields.
- SQL statements may return multiple records where only specific record is needed.
  - For example, consider following case:
  - Return "Line 1 of Insured's Address" for specific "Policy"
    - Need to specify relationship (join) between Policy and Insured, Insured and Address. This is Because Policy, Insured and Address object on screen have their respective tables. And hence data would be available in 3 different tables.

Why CP Expressions.. continued

- What may happen?
  - **Wrong Joins :** Developer may not write effective query. For example, instead of insured address, may write a query which results Agent address.
  - **Non Optimal Query :** Developer may not write optimal query. For example, developer may write wrong joins or too many joins that can hamper performance.
  - **Fetching multiple records** : Database may contain multiple records of Insured. Developer needs to write extra code for identifying specific record.
- How CP expressions resolves these issues?
  - **Context Path** uses object hierarchy instead of tables. Developers do not need to write any joins. Dev Studio will automatically convert CP to effective and optimal SQL query.
  - It is based on GID. GID is primary key of a transaction table. Using GID, CP knows record in the context and returns the desired record.

What are CP Expressions

- CP expression is alternative way to write SQL queries in Dev Studio.
- CP expression is converted to SQL queries using Dev Studio.
- It writes Optimal and Effective SQL queries.
- Very simple to use expression, developer need to focus on business logic rather than on SQL joins.
- These are DS specific expression, works only in Dev Studio
- CP expressions are tested for syntax using “cpExpQuickView.jsp”
- CP Expressions can be used while:
  - Writing “Advanced condition” UI in Rules
  - Configuring Data Sources
  - All places where queries can be written except Document Generation.

When Not to use CP Expression

- Think of “Does the query qualify to be converted to CP expression?”
  - If only the field and global expressions can be used, conversion to CP expression is not needed.
  - If the query does not contain MIS (transaction) tables, conversion to CP expression is not needed.
  - If the query only contains IEL tables, SDB tables, MIS_LOB_ASSOCIATIONS table, then conversion to CP expression is not needed
  - Document generation does not use CP.

Syntax

{CP:

o1 = /Object Path *[Condition]/~ alias = vmlb ;*

s1 = column1~ *alias = coverage*;

s2 = column2~ alias = coverage_code;

cond=condition;

cond=condition;

}

Note:

1. CP Expressions are case sensitive
2. Optional code in this syntax is shown in *italics.*
3. *Keywords:*

o: Object Path

s: Selection

cond: Condition

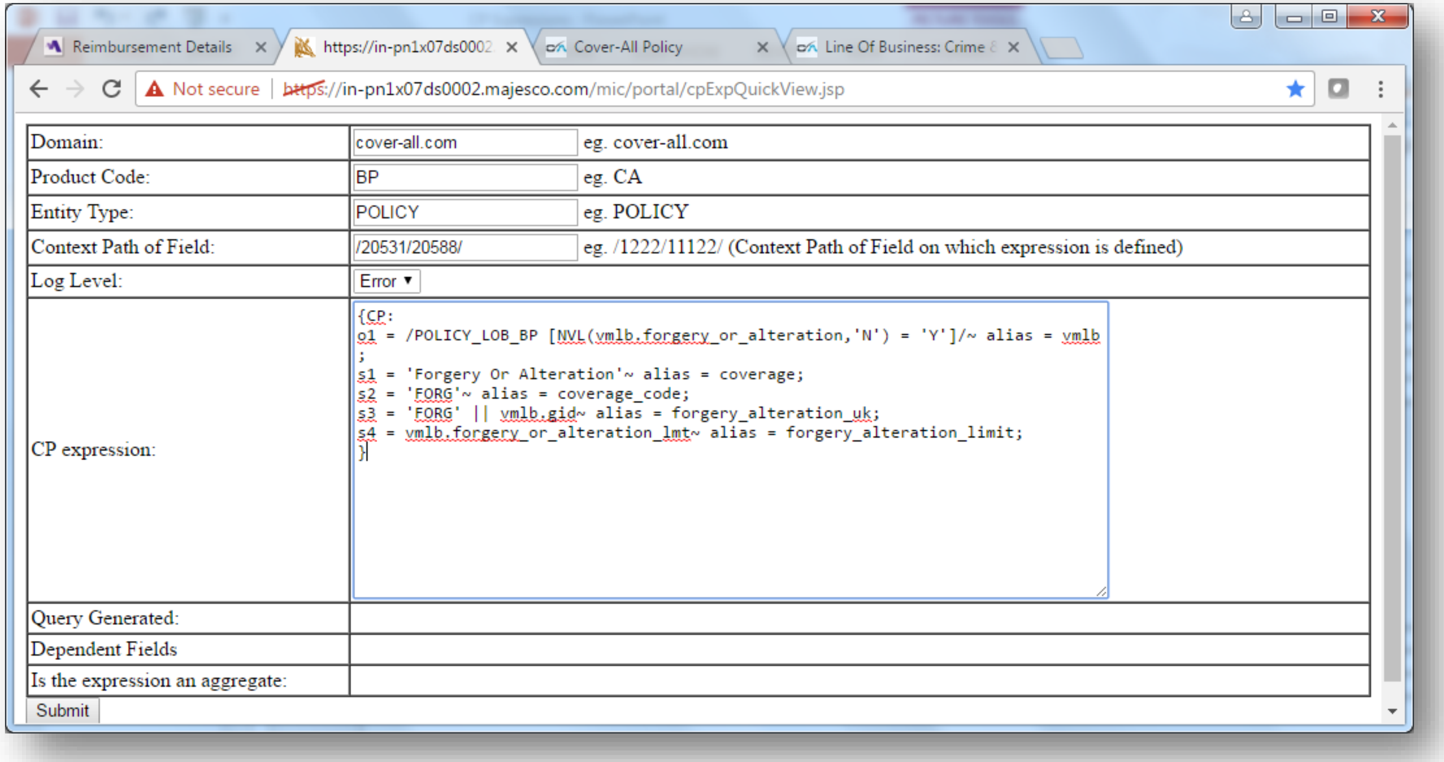
Steps in writing CP Expression

- Open JSP for writing CP expression:

https://in-pn1x07ds0002.majesco.com/mic/portal/cpExpQuickView.jsp

*This environment is used for training purpose. On projects, it will be your "Target" environment URL with /mic/portal/cpExpQuickView.jsp

It opens below page:



Domain:	cover-all.com	eg. cover-all.com
Product Code:	BP	eg. CA
Entity Type:	POLICY	eg. POLICY
Context Path of Field:	/20531/20588/	eg. /1222/11122/ (Context Path of Field on which expression is defined)
Log Level:	Error ▾	

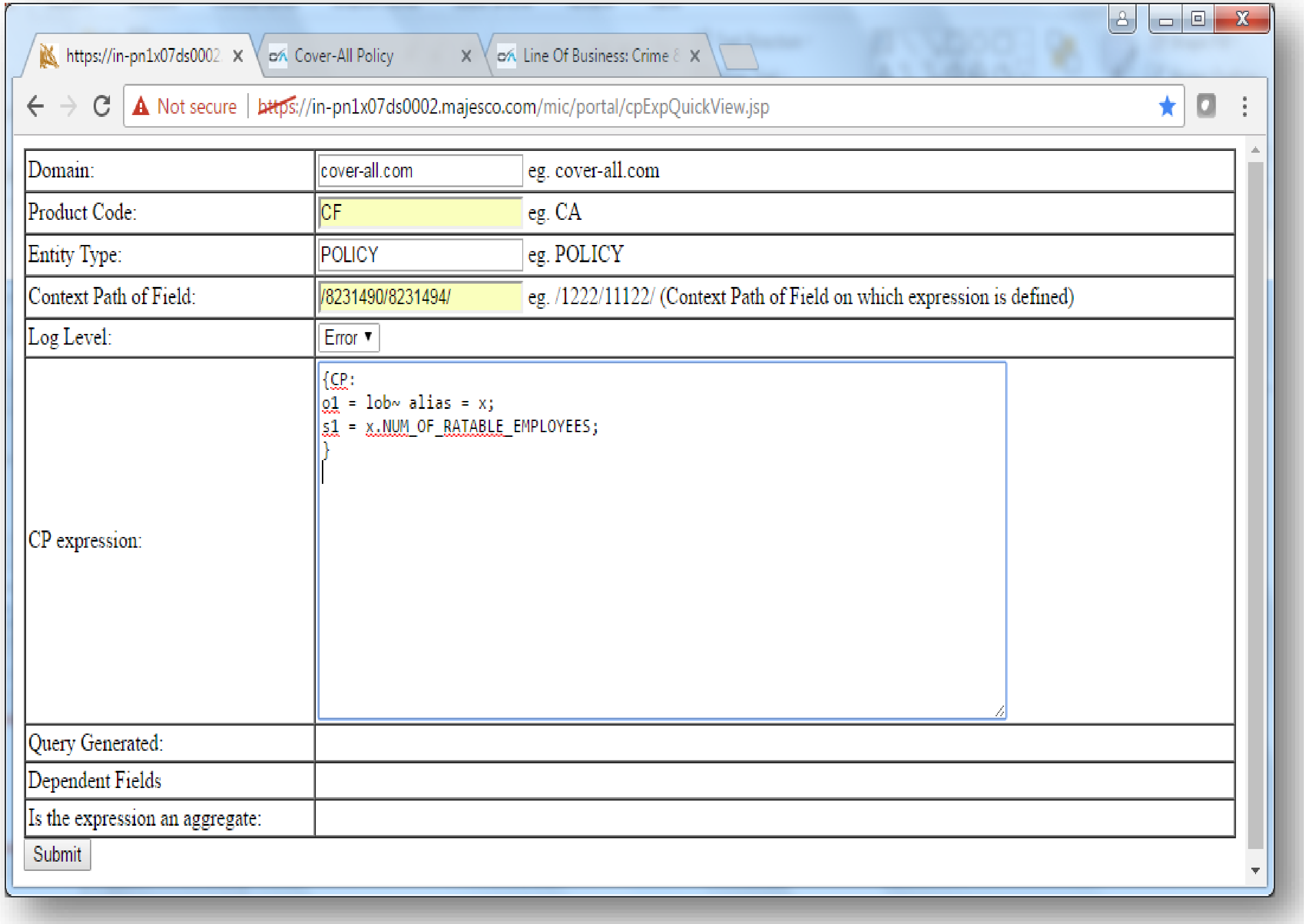
CP expression:

```
{CP:
o1 = /POLICY_LOB_BP [NVL(vmlb.forgery_or_alteration,'N') = 'Y']/~ alias = vmlb
;
s1 = 'Forgery Or Alteration'~ alias = coverage;
s2 = 'FORG'~ alias = coverage_code;
s3 = 'FORG' || vmlb.gid~ alias = forgery_alteration_uk;
s4 = vmlb.forgery_or_alteration_lmt~ alias = forgery_alteration_limit;
}
```

Query Generated:	
Dependent Fields	
Is the expression an aggregate:	

Submit

Understanding cpExpQuickView.jsp


https://in-pn1x07ds0002
Cover-All Policy
Line Of Business: Crime &
Not secure
https://in-pn1x07ds0002.majesco.com/mic/portal/cpExpQuickView.jsp

Domain:	cover-all.com	eg. cover-all.com
Product Code:	CF	eg. CA
Entity Type:	POLICY	eg. POLICY
Context Path of Field:	/8231490/8231494/	eg. /1222/11122/ (Context Path of Field on which expression is defined)
Log Level:	Error ▾	

CP expression:	{CP:
	o1 = lob~ alias = x;
	s1 = x.NUM_OF_RATABLE_EMPLOYEES;
	}	

Query Generated:		
Dependent Fields		
Is the expression an aggregate:		

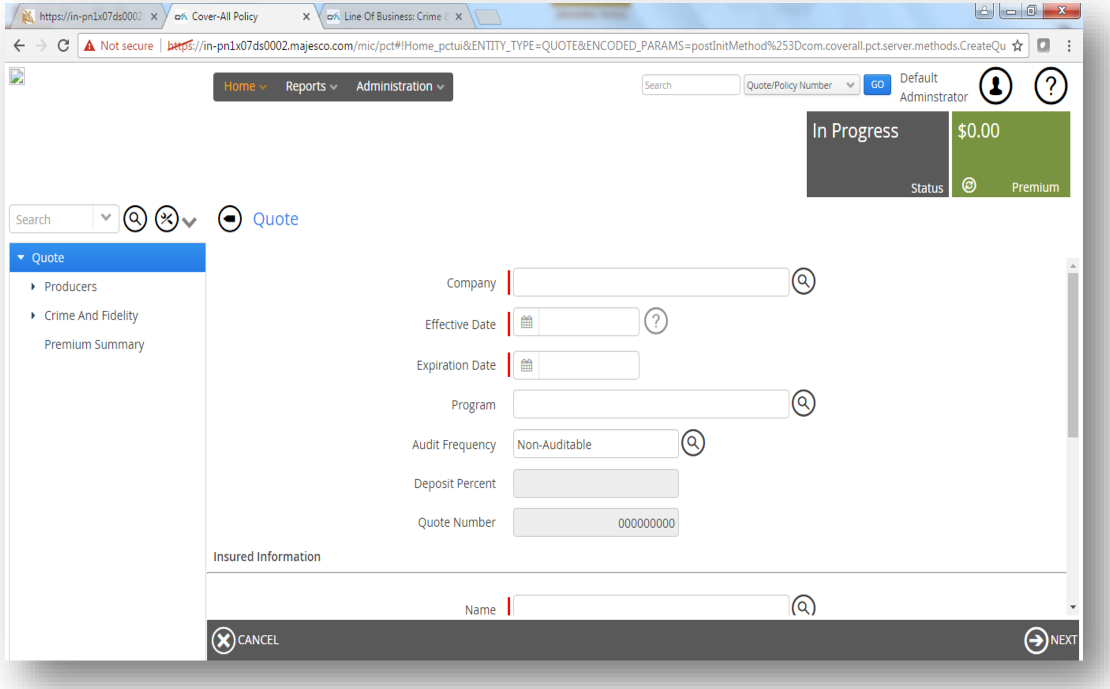
Submit

Note:

1. Product Code: Product code you are working in. Please note: CP expression do not support cross LOB queries.
2. Context path of field: CP of target field. It is field for which query result will be set.
3. CP Expression : It the source Entity/Object from which data needs to be selected by using CP Expression

Steps to find context path using Metapanel

- Open “Target” select any field
- Press CTRL+ALT+D
- Make applicable selections, click on submit
- Press CTRL+ALT+D again
- Click Meta_Panel …



In Progress
Status

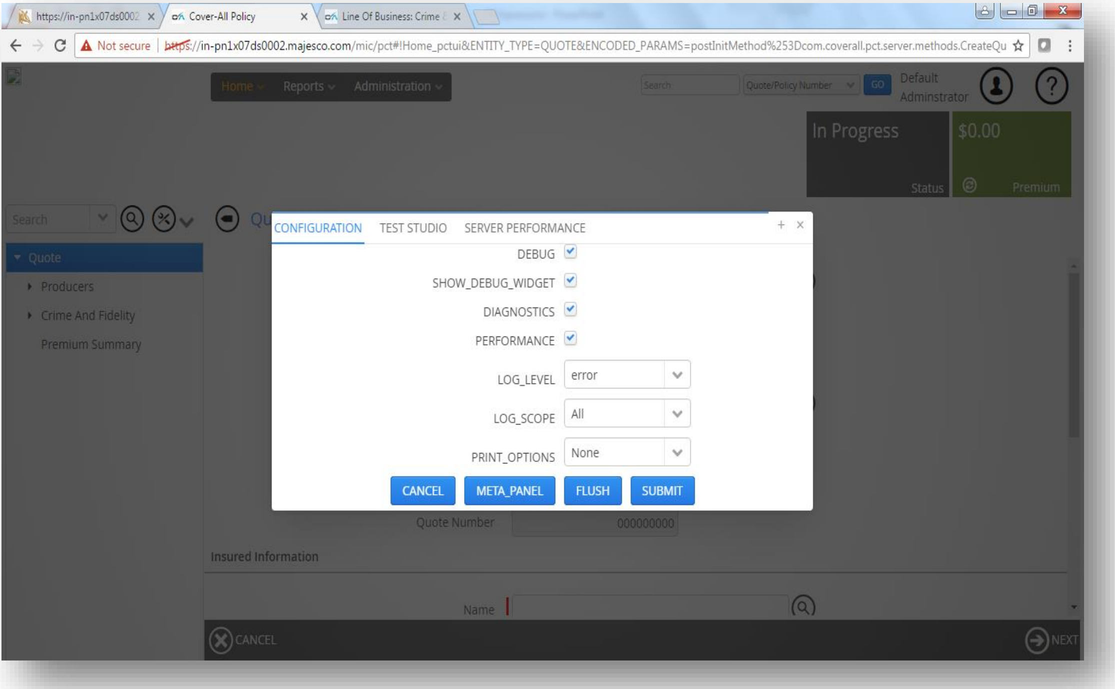
$0.00
Premium

Quote

Company	
Effective Date	
Expiration Date	
Program	
Audit Frequency	Non-Auditable
Deposit Percent	
Quote Number	000000000

Insured Information

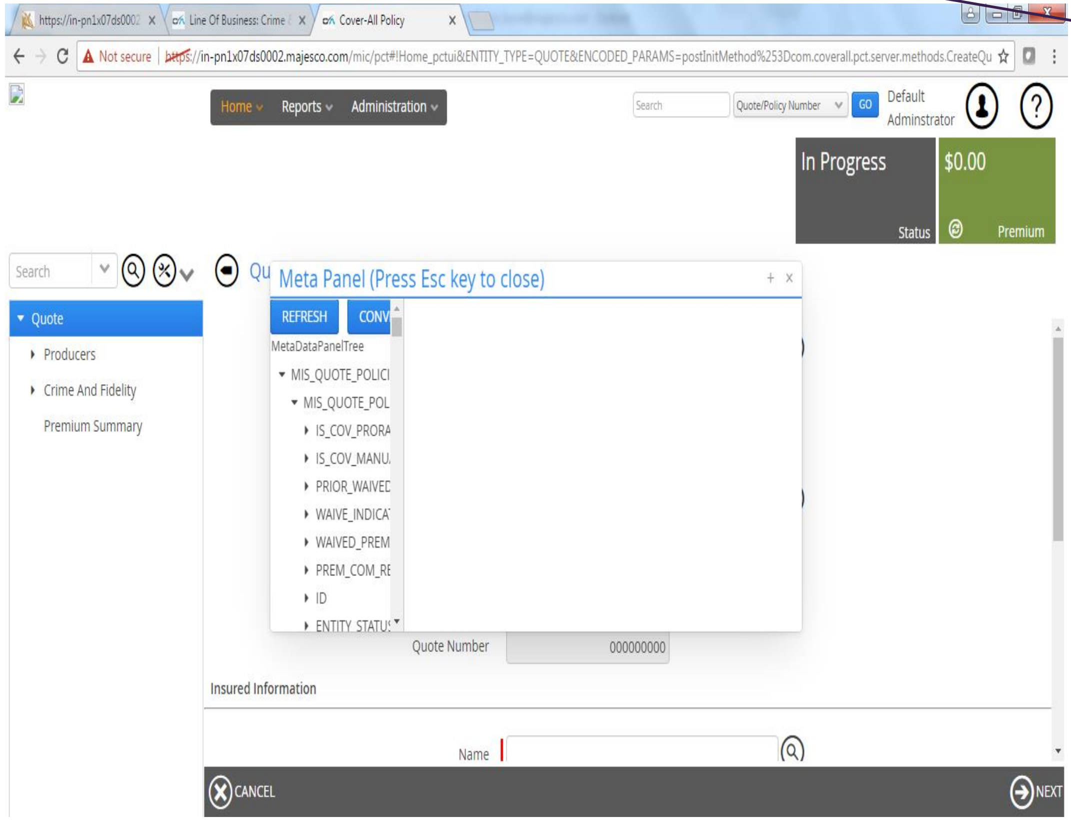
Name	

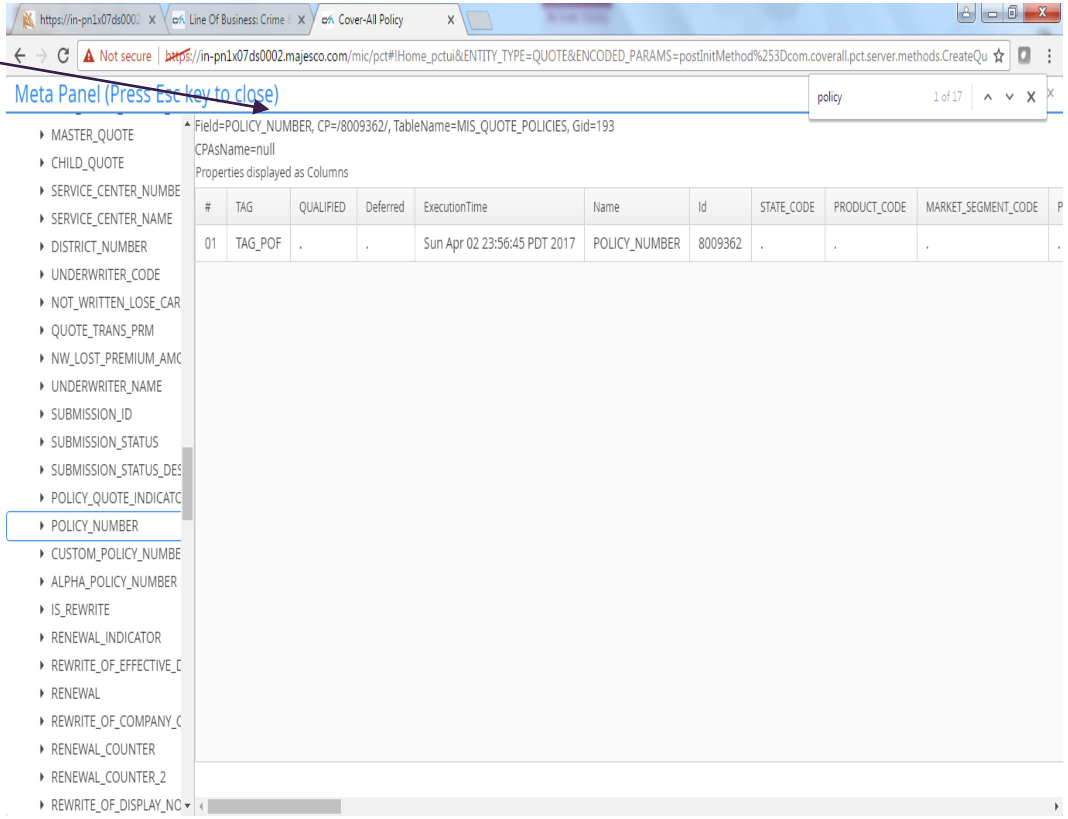


DEBUG	☑
SHOW_DEBUG_WIDGET	☑
DIAGNOSTICS	☑
PERFORMANCE	☑
LOG_LEVEL	error
LOG_SCOPE	All
PRINT_OPTIONS	None

Steps to find context path using Metapanel

- Copy CP from here and paste on “Context path on field” on slide 9.





Field=POLICY_NUMBER, CP=/8009362/, TableName=MIS_QUOTE_POLICIES, Gid=193

CPAsName=null

Properties displayed as Columns

#	TAG	QUALIFIED	Deferred	ExecutionTime	Name	Id	STATE_CODE	PRODUCT_CODE	MARKET_SEGMENT_CODE	P
01	TAG_POF	.	.	Sun Apr 02 23:56:45 PDT 2017	POLICY_NUMBER	8009362	.	.	.	.

Examples

Sample1:

```
{CP:
o1 = lob;
s1 = {field:NUM_OF_RATABLE_EMPLOYEES};
}
```

This expression works for the selected LOB selected on cpExpQuickView.jsp It retrieves value of field NUM_OF_RATABLE_EMPLOYEES. Following is SQL generated for above CP:

```
SELECT {field:NUM_OF_RATABLE_EMPLOYEES} FROM (
 SELECT
  {field:NUM_OF_RATABLE_EMPLOYEES}
 FROM
    VW_MIS_LOB_CF o1
 WHERE 1 = 1
  AND ('SEPARATOR' = 'SEPARATOR')
  AND (o1.gid = {field:gid})
  AND ('SEPARATOR' = 'SEPARATOR')
```

Examples

- Sample 2 : added “alias”

```
{CP:
o1 = lob~ alias = x;
s1 = x.NUM_OF_RATABLE_EMPLOYEES;
}
```

GENERATED SQL:

```
SELECT NUM_OF_RATABLE_EMPLOYEES  FROM (
  SELECT
    x.NUM_OF_RATABLE_EMPLOYEES
  FROM
      VW_MIS_LOB_CF  x
  WHERE 1 = 1
    AND ('SEPARATOR' = 'SEPARATOR')
    AND (x.gid = {field:gid})
    AND ('SEPARATOR' = 'SEPARATOR')
)
```

Examples

- Sample 3:

```
{CP:
o1 = lob;
s1 = o1.NUM_OF_RATABLE_EMPLOYEES;
}
```

- Please observe change in s1.

Examples

- Sample 4:

```
{CP:
o1 = /~ alias = x;
s1 = x.EFFECTIVE_DATE;
}
```

- To retrieve information irrespective of LOB’s, “/” can be used. It’s root policy screen.
- This will allow us to get information from QUOTE_POLICIES table.

Examples

- Sample 5:

```
{CP:
o1 = lob;
s1 = o1.ANNUAL_PREMIUM;
s2 = o1.ID;
s3 = o1.USER_CREATED || ' ' || o1.FK_COLUMN_VALUE;
}
```

- Note: Observe which database table we are referring to

Examples

- Sample 6:

```
{CP:
o1 = /POLICY_INSURED/INSURED_ADDRESS;
s1 = o1.COUNTRY_CODE~alias=COUNTRY;
s2 = o1.COUNTRY;
s3 = o1.LINE_1;
s4 = o1.LINE_2;
s5 = o1.CITY;
s6 = o1.STATE_CODE;
s7 = o1.STATE_DESC;
s8 = o1.ZIP_CODE;
}
```

- Multiple Columns example.

Examples

- Sample 7:

```
{CP:
o1 = LOB_CA_LOCATIONS/;
s1 = o1.NAME
}
```

Examples

- Sample 8:

```
{CP:
  o1 = /~alias = vmqp;
  s1 = 'Y'~alias = flag;
  cond = UPPER(vmqp.transaction_code) IN ('04', '04S' , '04A') ;
}
```

- Condition example.

Relative & Actual context path of objects

- If the objects used in CP expression are under the LOB, provide the context path relative to the LOB (instead of full context path), otherwise it will not work when the LOB is used in a package.

- Instead of specifying as following:

o1=/POLICY_LOB_CA/LOB_CA_VEHICLES/VEHICLE_COVERAGE_INFO/VEH_UNINSURED_MOTORIST_COV/;

Specify it as:

o1=LOB_CA_VEHICLES/VEHICLE_COVERAGE_INFO/VEH_UNINSURED_MOTORIST_COV/;

Note: Relative context path of object does not start with a forward slash.

The objects context path would be something like:

/PACKAGE_CPP/POLICY_LOB_CA/LOB_CA_VEHICLES/VEHICLE_COVERAGE_INFO/VEH_UNINSURED_MOTORIST_COV/

So use relative path, unless absolute path is absolutely necessary.

Examples

- Sample 9
- Referring to nested objects

```
{CP:
o1 = /POLICY_INSURED/INSURED_ADDRESS;
s1 = o1.ZIP_CODE;
}
```

- Accessing composite objects from policy screen.

Examples

- Sample 10

```
FOR CP /8009362/ (Policy Number)
{CP:
o1 = /POLICY_AUDIT_DETAILS/AUDIT_DET_REPORTING_PERIOD/~alias=audreptime;
s1 = audreptime.REPORTING_START_DATE;
s2 = audreptime.REPORTING_END_DATE;
}
```

Examples

- Sample 11
- Writing conditions:
    - GET COUNT OF POLICIES WHERE STATE=”NEW YORK”
        - Approach 1:

```
{CP:
o1 = /[o1.RISK_STATE='NEW YORK'];
s1 = count(*);
}
```

- Approach 2:

```
{CP:
o1 = /;
s1 = count(*);
cond = o1.RISK_STATE='NEW YORK'
}
```

Examples

- Sample 12:

```
{CP:
o1 = lob~alias=mylob;
s1 = count(*);
cond = mylob.RISK_STATE='NEW YORK';
}
```

- Sample 13:

Writing multiple conditions

```
{CP:
o1 = lob~alias=mylob;
s1 = count(*);
cond = mylob.RISK_STATE='NEW YORK';
cond = mylob.COVERAGE is not null;
}
```

Getting count of Records

- Sample 14:

```
{CP:
o1 = lob~alias=mylob;
s1 = count(*);
cond = mylob.RISK_STATE='NEW YORK';
cond = mylob.COVERAGE is not null;
cond = mylob.NUM_OF_RATABLE_EMPLOYEES >5;
}
```

Getting count of Records

Usage of OR

Sample 15:

```
{CP:
o1 = lob~alias=mylob;
s1 = count(*);
cond = mylob.RISK_STATE='NEW YORK';
cond = mylob.COVERAGE is not null or mylob.NUM_OF_RATABLE_EMPLOYEES
>5;
}
```

Getting count of Records

Better way to write code:

```
{CP:
o1 = lob~alias=mylob;
s1 = count(*);
cond = mylob.RISK_STATE='NEW YORK';
cond = mylob.COVERAGE is not null or mylob.NUM_OF_RATABLE_EMPLOYEES
>5;
}
```

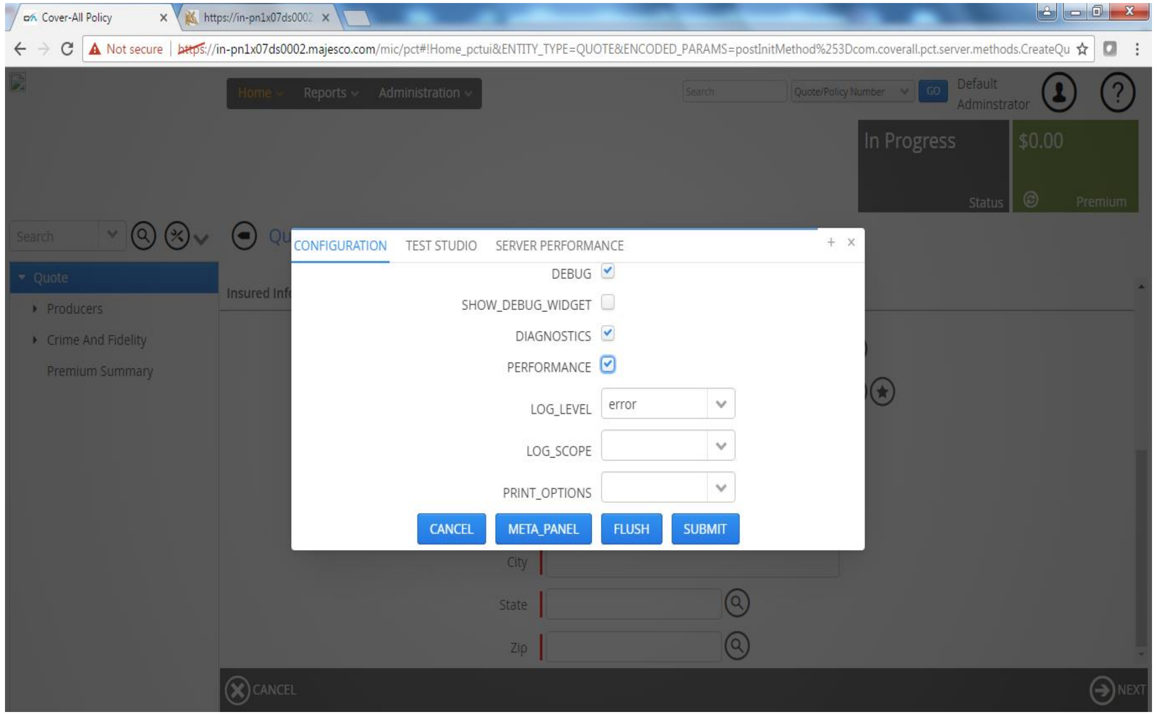
Nested CP Expression

- Code counting records for New York State and Company Name as ‘Cover-all Company’

```
{CP:
o1 = lob~alias=mylob;
s1 = count(*);
cond = mylob.RISK_STATE='NEW YORK';
cond = mylob.COVERAGE is not null;
cond = mylob.NUM_OF_RATABLE_EMPLOYEES >
    {CP:
    o1=/~alias=root;
    s1=count(*);
    cond=root.MQP_COMPANY_NAME='Cover-all  Company';
    };
}
```

Debugging CP expression

- Using CP Logs:
  - CP logs are stored in PCT_INFO_LOGS table
- Configuration options for Meta Panel:



CONFIGURATION

DEBUG ☑

SHOW_DEBUG_WIDGET □

DIAGNOSTICS ☑

PERFORMANCE ☑

LOG_LEVEL error

LOG_SCOPE

PRINT_OPTIONS

Checkbox	Usage
DEBUG	On Screen Debug (eg. Rules processing)
SHOW_DEBUG_WIDGET	Required in Policy2013. Not used in Policy 2015
DIGNOSTICS	Used for checking Server Performance for “show lookup” query
PERFORMANCE	Used for checking Server Performance for “show lookup” query
Note: DIAGNOSTICS and PERFORMANCE are used in combination and works only if Pull Modelling is OFF. If Pull modelling is on, first need to use “rating” and then debug panel.	

Best Practices

- Do not provide join between the objects(tables) or conditions using FK columns, the expression will make such joins if needed:
  - lib.reg_plates_liab_cov = um.reg_plates_unins_moto_cov (cp expression will join the tables)
  - o1.gid = o2.DRV_UNINSURED_MOTORIST_COV (cp expression will join the parent and child tables)
  - veh_uninsured_motorist_cov IS NOT NULL (such a use of FK column is wrong and will not give desired results. Test such a query to find out.)
- Always use CPA-name in object path don’t use context path.

MAJESCO

Thank You !