Contents

[Summary 3](#_Toc34835464)

[Configuration 3](#_Toc34835465)

[System 3](#_Toc34835466)

[Test 3](#_Toc34835467)

[Execute Writers 3](#_Toc34835468)

[ExecuteCAQ 3](#_Toc34835469)

[ExecuteWPD 3](#_Toc34835470)

[ExecuteCalculator 3](#_Toc34835471)

[ExecuteXBroker 3](#_Toc34835472)

[Paths 3](#_Toc34835473)

[ProjectRootPath 3](#_Toc34835474)

[TestPath 4](#_Toc34835475)

[Scheme Settings 4](#_Toc34835476)

[LOBName 4](#_Toc34835477)

[LineOfBusiness 4](#_Toc34835478)

[Insurer 4](#_Toc34835479)

[SchemeInsurerName 4](#_Toc34835480)

[InsurerID 4](#_Toc34835481)

[SchemetableID 4](#_Toc34835482)

[CommissionPercent 4](#_Toc34835483)

[RangePrefix 4](#_Toc34835484)

[wpdFileName 4](#_Toc34835485)

[Calculator Settings 4](#_Toc34835486)

[Quantity of Narrations 4](#_Toc34835487)

[PremiumSections 5](#_Toc34835488)

[RateStartDateTime 5](#_Toc34835489)

[Project Levels 5](#_Toc34835490)

[LimitsDenormalised 5](#_Toc34835491)

[LoadDiscountsDenormalised 6](#_Toc34835492)

[SchemeCalculatorParameterType 6](#_Toc34835493)

[QuestionSetBuilder 6](#_Toc34835494)

[CAQLOBWriter 6](#_Toc34835495)

[WPDWriter 6](#_Toc34835496)

[SQLLOBWriter 7](#_Toc34835497)

# Summary

Reads the configured LOB Risk structure from TGSL Screen designer xml “.tst “ files held in the LOB. Writes

A content.xml containing question text for use with Xbroker.

The Product database data for adding the LOB to CAQ.

A LOB view for use with Bordereaux.

A Scheme calculator “.wpd” ,a TGSL install script and Scheme Calculator SQL Object Scripts.

# Configuration

## System

Values =-LIVE:UAT :DEV

Governs which environment to release scripts and apply database changes.

Setting UAT will apply CAQ LOB database changes to the UAT Product database held in the {ProductUAT} connectionString setting.

## Test

True:False

If true Scheme, LOB and Xbroker files will be output to an alternative location held in {TestPath}.

## Execute Writers

### ExecuteCAQ

True:False

If true will update the Product database QuestionSet for LOB {LOBName}logic and questions text from the TGSL Screen designer forms.

### ExecuteWPD

True:False

If true will output a TGSL “.wpd“ scheme calculator file for the LOB and scheme based on the configuration.

### ExecuteCalculator

True:False

If true will output

A set of SQL Objects for the Calculator database as configured by Calculator settings.

An AddScheme script to add a new scheme to TGSL as configured by Scheme Settings.

A set of synonyms for reading TGSL tables.

A TGSL Risk table View for reading for use in bordereaux.

### ExecuteXBroker

True:False

If true will update the Website (Xbroker) Content.xml with the question text from the TGSL Screen designer forms.

## Paths

### ProjectRootPath

Local full path Location of repository TGSL\LineOfBusiness. Files will be read from and written to this location.

### TestPath

Alternative write path to ProjectRootPath. If {Test} is true the files will be written to this location.

## Scheme Settings

### LOBName

TGSL Shortname for Line Of Business e.g. MLIAB for tradesman Liability.

### LineOfBusiness

Fully descriptive name of the product/Line Of Business e.g. “Tradesman Liability”

### Insurer

Full name of the insurer as described in the System\_Insurer

### SchemeInsurerName

For Moorhouse products backed by a Finance the name Lyndon has dreamed up. E.g. “Companion”.

### InsurerID

Insurer\_ID from table System\_Insurer matching {Insurer}.

### SchemetableID

Scheme\_Table\_ID From RM\_Scheme matching the SchemeName : {Insurer} – LineOfBusiness}. Only available after a scheme has been added. Used for configuring the “.wpd”

### CommissionPercent

Standard Partner commission for NB ,MTA and renewal. Used as a default value for the commission level in the AddScheme script.

### RangePrefix

5 letter prefix for the range groups created by the AddScheme script.

### wpdFileName

Name to give the “.wpd” file. Usually 2 to 3 letters from Insurer + 2to3 letters from Scheme + 01.

## Calculator Settings

### Quantity of Narrations

Number of Narrations of each type to generate in the “.wpd” and calculator functions . These can be tailored later.

#### NumberOfDecline

#### NumberOfRefers

#### NumberOfProductDetails

#### NumberOfExcesses

#### NumberOfSummarys

#### NumberOfBreakdowns

#### NumberOfEndorsements

### PremiumSections

Comma delimited list of Policy Sections to be calculated. Corresponding to values in TGSL table List\_Policy\_Section. E.g. “LIABPREM,EMPLPREM”

### RateStartDateTime

Date rates are due to start for the scheme. Used in AddScheme for schemestartdate. Assumptions table. For example Insert statements for default calculator tables.

### Project Levels

Project:LOB:Scheme:None

Qualifies the names of SQL objects so that they can be shared by all schemes in the same LOB. Objects should generally be set at LOB and then tailored to a scheme if required.

Project :- Applies to all LOBS e.g. Shared Lists

LOB :-Applies at the LOB level e.g. List\_Link values.

Scheme :- Applies to an individual Scheme. E.g. Create a separate Table for endorsements.

None :- Do not create this object

#### ProjectLevelAssumptions

#### ProjectLevelClaims

#### ProjectLevelLimits

#### ProjectLevelLoadDiscounts

#### ProjectLevelRates

#### ProjectLevelSynonyms

#### ProjectLevelCalculator

#### ProjectLevelSchemeDispatch

#### ProjectLevelAddScheme

#### ProjectLevelEndorsements

#### ProjectLevelTrades

#### ProjectLevelExcesses

#### ProjectLevelWpd

#### ProjectLevelRiskTableTypes

#### ProjectLevelRiskBordereaux

#### ProjectLevelFormulae

### LimitsDenormalised

True:False

If true then each limit type has its own column in a table and a tvf is generated to bring back one record for the required dates.

If False then a table will hold a type column and a value column and an svf pulls back the individual value for the required limit type .

### LoadDiscountsDenormalised

True:False

If true then each Discount type has its own column in a table and a tvf is generated to bring back one record for the required dates.

If False then a table will hold a type column and a value column and an svf pulls back the individual value for the required discount type .

### SchemeCalculatorParameterType

Tables:Variables

Access Risk parameters from tables or individual variables.

This feature is now redundant as tables were settled on in version 3.0.

# QuestionSetBuilder

Reads the set of TGSL xml files (.tst) which describe a Line of business created by the Screen Designer into a QuestionSet Object.

The QuestionSet stores this form/RiskScreen data in memory as a list of Forms.

A form has a list of Child forms and a List of Controls.

A control describes a question, e.g. Question text, enabled, Type, TGSL Table and field name etc..

A controls Question text comes from the label whose description corresponds to the controls answerName

A controls answertype is parsed into a Product.QuestionSet AnswerTypeID.

Controls of type AnswerTypeID = 1 ie List types store List Table and Column names.

A list of reference tables (List tables) is maintained along with aliases for repeatedly used lists.

A control has a list of EnableControls. Controls whose enablement depends on this controls value.

An enableControl describes a control and the criteria required to enable/disable it.

# CAQLOBWriter

Maintains a set of SQLCommands and functions for updating the Product database LOB tables ie the [QuestionSet] Schema from the QuestionSet instance provided by QuestionSetBuilder.

Removes the existing LOBQuestionSet if it exists and creates a new question set as a single transaction which rolls back if it errors.

# WPDWriter

Transforms a QuestionSet provided by QuestionSetBuilder into a wpd scheme file which is generic for a LOB and tailored with the Schemename, Underwriter , SchemeTableID.

It Writes Transactor scheme code to

Read the Risk from the scheme {foreach} elements into a single XML string called RiskXML.

A TGSL TableApply function provided with the

LOB calculator usp Name.

RiskXML

Policy Start Date

Policy quote Stage

Postcode

SchemeTableId

AgentName

SubagentID

And expecting return values as described in the config along with the calculated Premiums and commissions, and whether the scheme has refered or declined.

Populate the output narrations from the ApplyTable outputs.

Populate Refers and Declines from the ApplyTable outputs.

Send Premiums and commissions to the associated Outputs.

Sum the Premiums and output as the NetPremium.

# SQLLOBWriter

Writes an AddDefaultScheme Script to populate Transactor\_Live with the new Scheme from the configuration parameters.

Scripts

Synonyms to the Transactor List tables referenced by the risk tables.

Create of the User defined TableTypes representing the Risk tables

Depending on the project level (LOB, Scheme, None), it writes database Scripts for database objects named using the Moorhouse Calculator convention for LOB and Schemes.

Excess - table,

ListEndorsement - data Insert statement

Limits - Table, data insert, svfSelect

Assumptions - Table, data insert, svfSelect, tvfAssumptions refers for assumptions

LoadsAndDiscount – Table, Data Insert, svfSelect

Rates – Table, Data Insert, svfSelect

Trades – Tble, tvfTrade

Claims – tvfClaims Summary of claims data

uspCalculator – Sets up a calculator framework to translate the LOB risk xml to tables and return outputs. For scheme level it will also perform lookups, calculate and return outputs. At Lob level it will call the tvf Scheme dispatcher then collect and shred the return values.

tvfDispatcher – Selects the tvfSchemecalculator to run based on SchemeTableID or refer. Passes back the calculated outputs to a calling uspLOBCalculator.

tvfSchemeCalculator – Acceptds the risk data, Performs all look ups and calculations, returns the outputs to tvfDispatcher.

Bordereaux View – A view of the join of all LOB risk tables and referenced List tables.

Formulae – Registers the Cover Text , Value and Premium formulae for each premium section configured. You need to add the Premium section data collection for this LOB to the procedure uspPremiumCoverSelect manually.

# XBrokerWriter

Writes the Full question text from the QuestionSet into the content.xml of the website files produced by Screen Designer.