EIS GROUP™

EIS Policy Core
Policy Premium Capping Design

Author	Zilvinas Vilutis
Status	Draft
Document Number	
Document Version	01.00.04
Last Revision Date	May 29th, 2015

Revision History

Name	Date	Reason For Changes	Version
Zilvinas Vilutis	May 12th, 2015	Initial Version	01.00.00
Zilvinas Vilutis	May 19th, 2015	Minor updates to the domain model. Sequence diagrams	01.00.01
Zilvinas Vilutis	May 26th, 2015	UI Mockup update. Sequence diagram added. UC2 and UC3 updated: No capping was applied on the previous term	01.00.02
Zilvinas Vilutis	May 27th, 2015	Removed assumption about new added risk items not being capped.	01.00.03
Zilvinas Vilutis	May 29th, 2015	Adding an extension point for configuring the restriction of capping override	01.00.04

Copyright

2014, Exigen Insurance Solutions, Inc. and/or affiliates. All rights reserved.

Table of Contents

Exigen Confidential

1. Introduction

1.1. Purpose of This Document

The purpose of this document is to provide requirements analysis and implementation design of the following JIRA ticket: EISDEV-102128 **Premium Capping Support**

The intended audiences for this document are the:

- Technical Team
- Engineering
- Quality Assurance
- Support
- Operations
- Training
- Technical Publications
- Business Team

1.2. Terms, Acronyms, and Abbreviations

Term	Definition
UC	Use Case
UI	User Interface
EIS	Exigen Insurance Suite Policy Administration Application
Premium	Insurance Policy Premium is a price for the policy that has to be paid over the term
Rating	A process of calculating Policy Premiums. In EIS Policy Core this process consists of sending a request to the Rating Engine where the actual rating happens
Rating Engine	A service (typically a web service) calculating Policy Premiums based on Policy data
Premium Capping	Limiting maximum Policy Premium increase or decrease during renewal
Premium Capping Percentage	Maximum premium change limit
Capping Factor	A factor (number) by which the Policy Premium is multiplied to reduce the premium change over the year(s).
Rate Effective Date	Date as of which the Rating factors are being used.
Policy Effective Date	Date when the Policy becomes active or *in effect*.
QCC	Quote Comparison Component
MCS	Manual Conversion Support

Exigen Confidential

2. Feature Requirements

This chapter covers the high-level requirements for Policy Premium Capping Support feature.

2.1. Functional Requirements

Policy Premium Capping is a concept of trying to apply premium changes incrementally to existing customer policies during renewal. Meaning if some factor change in the Rating Engine would mean a drastic premium change to a customer's policy during renewal then the insurer would choose reduce the change by applying Capping to the premium by allowing a maximum or premium change percentage called Capping Percentage.

Calculating Capping Factor

To calculate the Capping factor the following is being done:

1. Calculate Policy Premiums using the *legacy* Rating Engine OR using Rate Effective Date as the previous term Policy Effective Date
2. Calculate Policy Premiums using the *current* Rating Engine
3. Calculate the Premiums increase or decrease percentage
4. If the percentage is bigger than Capping Floor or Capping Ceiling values – calculate Capping Factor as: ("Policy Premiums – Old Rater" / "Policy Premiums – New Rater") * (1 + Ceiling)*

*Note: this algorithm is just a sample and might be different per customer, product or state.

Depending on the Customer requirements Capping Factor might be calculated at a Policy level, at Risk level or Coverage level.

Capping Factor Override

Capping Factor may be overridden manually in UI in case the agent wants to increase or decrease the capping factor based on some other conditions. When a Capping Factor is overridden – Policy Premiums must be re-calculated with the new Capping Factor provided.

See all Premium Capping Support requirements in the attachment in [Attachment 1](#) [EIS](#) [AAA](#) [Capping Requirements](#)

Capping Reset

For subsequent renewals Capping Factor will not be recalculated as the rating engine might be changed again. Instead the Capping Factor is increased (or decreased) by ceiling / floor percentage until it reaches 1.0 or 100%. There can be carriers which will decide to recalculate Capping Factor during each renewal.

Exigen Confidential

There is another process called Capping Reset when the system will decide that the Capping Factor is reset and has to be calculated again.

Capping Reset should be a part of OpenL rules

2.2. Non-Functional Requirements

The non-functional requirements for Premium Capping Support are:
- This feature must be made available to all future products so the support has to be implemented in EIS Policy Core.
- This feature must be easy extendable and configurable for client projects, as little as possible should be hard-coded and move as much as possible to OpenL

2.3. Assumptions and Constraints

- **Assumption:** Capping configuration will be stored in OpenL and some part of it will be retrieved to EIS Policy Core during a rating request only
- **Assumption:** A Copy of Capping Configuration will be stored in EIS Policy Core (when possible)
- **Assumption:** Capping Factor is calculated and applied during Rating in OpenL Rating Engine
- **Assumption:** Capped Premium, Non Capped Premium and Capping Factor Used will be returned in the Rating response
- **Assumption:** Once Capping Factor is calculated – it should not be re-calculated during mid-term transactions, but the same value to be used over the whole term
- **Assumption:** Decision, whether to re-calculate the capping factor will be performed in OpenL based on the policy data
- **Assumption:** In most cases Capping Factor will be applied to all subsequent renewals until it becomes 1.0 or 100 %, but this could also be different depending on the requirements
- **Assumption:** There may be different capping factors per variation in QCC so every variation will store separate Capping as well as Capping Configuration and Override information

In Scope:
- Policy Product Components for Capping UI and Capping Override
- Policy Services SPI for resolving Capping configuration

Out of Scope:
- OpenL WebStudio
- OpenL Rating Sheet configuration
- Reporting

See all Premium Capping Support requirements in the attachment in Attachment 1 EIS AAA Capping Requirements

Exigen Confidential

2.4. Use Cases

UC1: New Business Quote

Preconditions

- A new business quote has been created
- Required fields for rating have been entered
- User is authorized to view capping information

Main Flow

Actor	Action
User	1. Navigates to "Coverages & Premiums" tab 2. Performs "Calculate Premiums" action
System	1. Calculates Premiums 2. Displays Premium Screen

Expected results

"View Capping Details" button is disabled, no capping is applied

UC2: Renewal

Preconditions

- A renewal quote is created
- Required fields for rating have been entered
- User is authorized to view capping information
- No capping was applied on the previous term
- No or minimal rating factor changes were made and no capping is necessary

Main Flow

Actor	Action	Comments
User	1. Navigates to "Coverages & Premiums" tab 2. Performs "Calculate Premiums" action	
System	1. Calculates Premiums 2. Displays Premium Screen	"View Capping Details" button is disabled

Exigen Confidential

Expected Results

"View Capping Details" button is disabled

UC3: Renewal - Capping Applied the first time

This Use Case is valid for renewals when capping not applied in the previous term. Usually this is a second renewal

Preconditions

- A renewal quote is created
- Required fields for rating have been entered
- User is authorized to view capping information
- No capping was applied on the previous term
- Rating factor changes were significant enough to cause Premium Capping

Main Flow

Actor	Action	Comments
User	1. Navigates to "Coverages & Premiums" tab 2. Performs "Calculate Premiums" action	
System	1. Calculates Premiums 2. Displays Premium Screen	"View Capping Details" button is enabled
User	1. Clicks "View Capping Details" button	
System	1. Displays "View Capping Details" popup	

Expected Results

System displays Capping Information.

UC4: Renewal - Capping Override

Preconditions

- A renewal quote is created
- Required fields for rating have been entered
- User is authorized to view capping information
- Rating factor changes were significant enough to cause Premium Capping

Exigen Confidential

● Premiums are calculated and Capping is applied

Main Flow

Actor	Action	Comments
User	1. Navigates to "Coverages & Premiums" tab 2. Clicks "View Capping Details"	
System	1. Displays "View Capping Details" popup 2. Displays Premium Screen	
User	1. Enters manual capping factor and override reason 2. Clicks "Calculate"	
System	1. Calculates premiums with a manual capping factor	

Expected Results

Premiums are calculated using the manual capping factor

UC5: Renewal - Capping applied in Previous term

Preconditions

● A renewal quote is created
● Required fields for rating have been entered
● User is authorized to view capping information
● Capping **was** applied on the previous term

Main Flow

Actor	Action	Comments
User	1. Navigates to "Coverages & Premiums" tab 2. Performs "Calculate Premiums" action	
System	1. Calculates Premiums 2. Displays Premium Screen	"View Capping Details" button is enabled

Exigen Confidential

Actor	Action	Comments
User	1. Clicks "View Capping Details" button	
System	1. Displays "View Capping Details" popup	

Expected Results

- ● System displays Capping Information
- ● If previous capping factor was < 100%
  - ○ Capping Factor = MIN ((Previous Term Capping Factor + Ceiling %), 100%)
- ● If previous capping factor was > 100%
  - ○ Capping Factor = MAX ((Previous Term Capping Factor - Floor %), 100%)

UC6: Manual Conversion - New Business

This applies to a new business policy quote with source = "Manual Conversion". This means that the quote was imported using the MCS Import Tool and the agent is continuing to fill the rest of the details.

Preconditions

- ● A renewal quote is created
- ● Policy Source = "Manual Conversion"
- ● Required fields for rating have been entered
- ● User is authorized to view capping information
- ● Capping **was not** applied on the previous term in the legacy system

Main Flow

Actor	Action	Comments
User	1. Navigates to "Coverages & Premiums" tab 2. Performs "Calculate Premiums" action	
System	1. Calculates Premiums 2. Displays Premium Screen	"View Capping Details" button is enabled
User	1. Clicks "View Capping Details" button	

Exigen Confidential

System	1. Displays "View Capping Details" popup	

Expected Results

System displays Capping Information.

UC7: Manual Conversion - Capping Previously Applied

Same as UC6, but Capping was applied in the previous term in the legacy system and needs to be transferred

Preconditions

- A renewal quote is created
- Policy Source = "Manual Conversion"
- Required fields for rating have been entered
- User is authorized to view capping information
- Capping **was applied** on the previous term in the legacy system
- Previous Term Capping Factor was imported using the MCS

Main Flow

Actor	Action	Comments
User	1. Navigates to "Coverages & Premiums" tab 2. Performs "Calculate Premiums" action	
System	1. Calculates Premiums 2. Displays Premium Screen	"View Capping Details" button is enabled
User	1. Clicks "View Capping Details" button	
System	1. Displays "View Capping Details" popup	

Expected Results

- System displays Capping Information
- If previous capping factor was < 100%
  - ○ Capping Factor = MIN ((Previous Term Capping Factor + Ceiling %), 100%)
- If previous capping factor was > 100%

Exigen Confidential

○ Capping Factor = MAX ((Previous Term Capping Factor - Floor %), 100%)

UC8: Automated Conversion

An automated policy conversion from legacy systems was performed, which resulted in a "stub" policy and a renewal is being processed.

In general - it should act exactly the same as a regular renewal. If Capping was not applied previously - it should be calculated from scratch (UC3-UC4), otherwise - it should be adjusted (UC5, UC7)

2.5. Samples

Please see the working formulas in the excel spreadsheet: Premium Capping Samples

Given the following configuration

Ceiling	10%
Floor	-10%

Increase premium sample

Renewal #	Premium Old Rater	Premium New Rater	Premium Change %	Capping Factor	Capped Premium
1	$ 1,000.00	$ 1,500.00	50%	73.33%	$ 1,100.00
2	$ 1,500.00	$ 1,600.00	7%	80.67%	$ 1,290.67
3	$ 1,600.00	$ 1,500.00	-6%	88.73%	$ 1,331.00
4	$ 1,500.00	$ 2,000.00	33%	97.61%	$ 1,952.13
5	$ 2,000.00	$ 1,500.00	-25%	100.00%	$ 1,500.00
6	$ 1,500.00	$ 3,000.00	100%	100.00%	$ 3,000.00
7	$ 3,000.00	$ 1,000.00	-67%	100.00%	$ 1,000.00

Decreased premium sample

Renewal #	Premium Old Rater	Premium New Rater	Premium Change %	Capping Factor	Capped Premium
1	$ 1,000.00	$ 600.00	-40%	150.00%	$ 900.00
2	$ 600.00	$ 800.00	33%	135.00%	$ 1,080.00
3	$ 800.00	$ 750.00	-6%	121.50%	$ 911.25
4	$ 750.00	$ 1,000.00	33%	109.35%	$ 1,093.50

Exigen Confidential

5	$ 1,000.00	$ 900.00	-10%	100.00%	$ 900.00
6	$ 900.00	$ 700.00	-22%	100.00%	$ 700.00

In the samples above you can see that one the Capping Factor is not recalculated based on premium changes, but based on previous factor + floor or ceiling value instead until it reaches 100%. To recalculate the factor - a Capping Reset will have to be applied.

Exigen Confidential

3. Design

This chapter is meant to cover the implementation design for Policy Premiums Capping Support.

EIS Policy Core part contains of the following pieces:

- View Capping UI
- Capping Override UI
- SPI for Premium Capping mapping to and from OpenL Rating Engine
- QCC Support for Capping

3.1. UI Components & Flow

Capping Information is for display only and cannot be modified. The only thing that can be modified is the Capping Factor

Capping Information Display

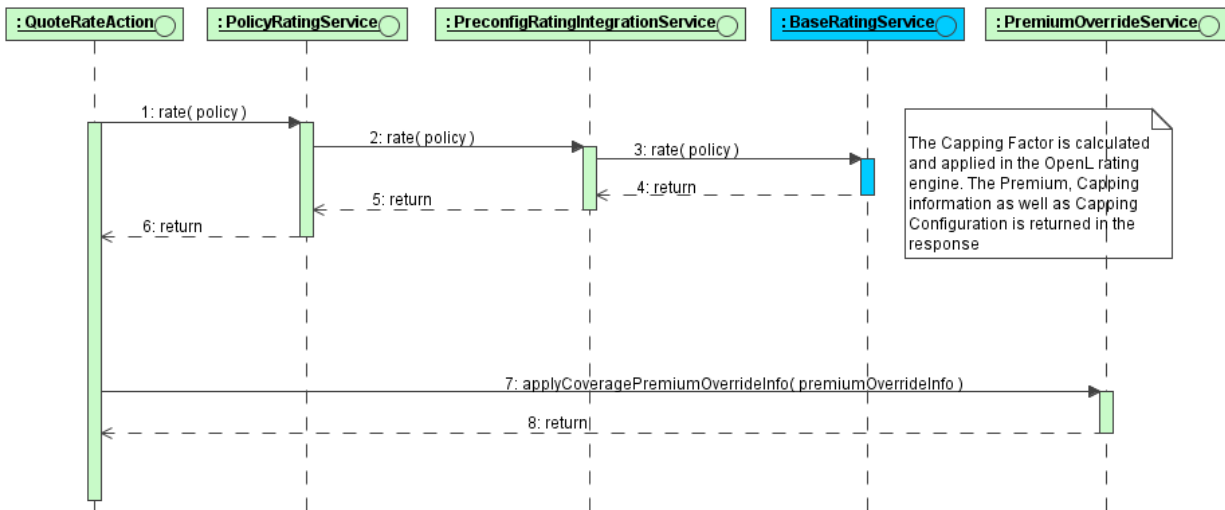**Exigen Confidential**

3.2. Integration with Rating Engine (OpenL)

Premium Capping is essentially one of the rating steps in the Rating process. When a premium is calculated for a renewal - Rating engine will have to determine the capping factor and then apply it when calculating the actual Premium for the Policy Coverages.

Sequence Diagram

Rating Request

Rating request will have to support input of Capping Factor override as well as all the attributes required to calculate the new Capping Factor. E.g., for subsequent renewals when a new capping factor is calculated - the previous term capping factor is used. Also *Capping Reset* indicator might need to be taken into account.

Rating Response

The Rating response will have to provide the following data at the most granular level at least these premium attributes:

- Premium - Old Rater
- Premium - New Rater
- System Calculated Capping Factor
- Capping Factor applied (in case of capping override)
- Capped Premium (which, in essence is regular premium)

Capping Configuration

Capping Configuration is going to be stored and in OpenL Rating Sheet, however it will be retrieved together with the Rating Response. This means that EIS Policy Core requires to store a copy of the configuration for display.

3.3. Domain

For different Products different capping levels may be used, therefore EIS Policy Core has to support all premium capping levels. The most granular level for Premium Capping may be Coverage so Premium Capping support will be required on:

- Policy level
- Risk Item level
- Policy Coverage level
- Risk Item coverage level

To support Premium Capping we are creating a *PremiumCapping* entity and adding it to each *PremiumHolder* level as well as the interface itself.

Exigen Confidential
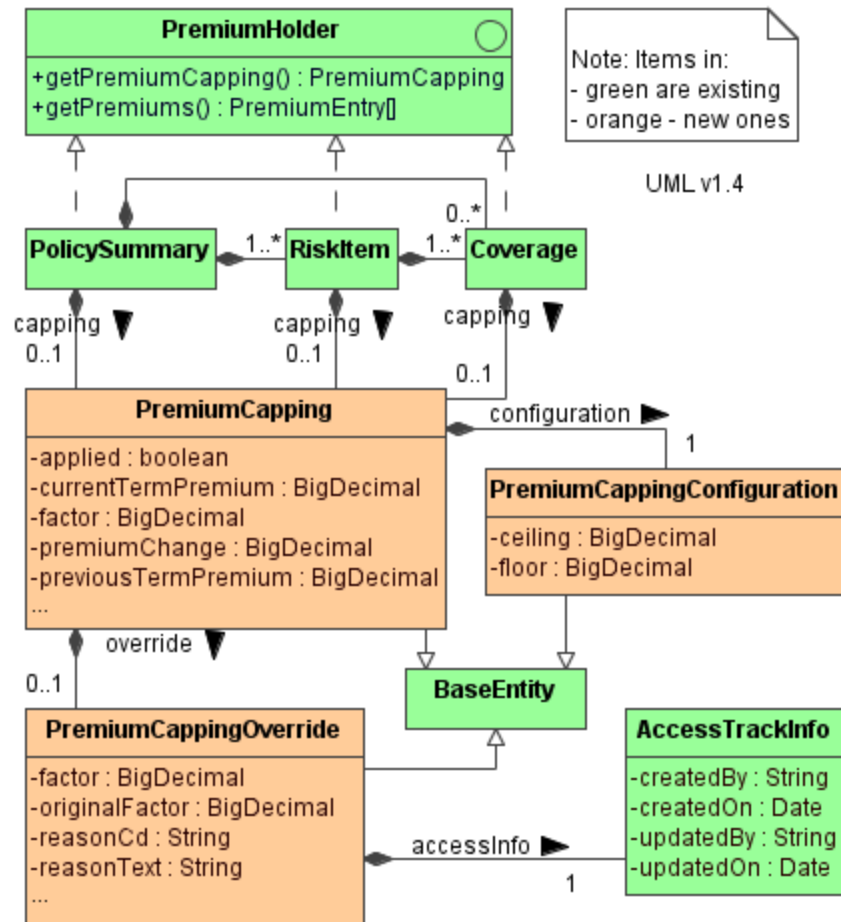
Premium Capping entity

PremiumCapping class extends BaseEntity class and contains the following attributes:

Attribute Name	Type	Description
applied	boolean	Indicates whether capping is applied
factor	BigDecimal	Capping Factor applied
previousTermPremium	BigDecimal	Premium Calculated with *previous* policy term effective date as rate effective date for the current policy
currentTermPremium	BigDecimal	Current Policy term premium before capping
premiumChangePct	BigDecimal	Premium Change percentage - how much much more (or less) premiums have changed comparing to the previous rater
calculationDate	Date	Date when the Capping Factor was calculated by the System

Exigen Confidential

PremiumCappingConfiguration entity

Attribute Name	Type	Description
floor	BigDecimal	Maximum premium decrease percentage
ceiling	BigDecimal	Maximum premium increase percentage

PremiumCappingOverride entity

Attribute Name	Type	Description
applied	boolean	Is Capping Override applied?
originalFactor	BigDecimal	System Calculated Capping Factor
factor	BigDecimal	Manually overridden Capping Factor
reasonCd	String	Override reason code (for a lookup)
reasonText	String	Manually entered override reason (when 'reasonCd' = 'other')
accessInfo	AccessTrackInfo	Capping Override audit information (created / updated on, created / updated by)

3.4. Services

Premium Capping Override

To override Premium Capping factor another call will have to be made to Rating Engine to apply the manually overridden Capping Factor.

Process is the following:

- Set the required attributes in *PremiumCappingOverride* entity
- call *applyCappingOverride* method which would perform the required logic, including re-calculating policy premiums (if required), or clear premiums data, depending on the requirements

Exigen Confidential

```
┌─────────────────────────────────────────────────────────────────────────┐
│               PremiumCappingOverrideService                          ◯   │
├─────────────────────────────────────────────────────────────────────────┤
│ +applyCappingOverride( root : RootPremiumHolder, premiumHolder : PremiumHolder ) : void  │
│ +removeCappingOverride( root : RootPremiumHolder, premiumHolder : PremiumHolder ) : void │
└─────────────────────────────────────────────────────────────────────────┘
```

Also the Capping Override will have to be removed during a list of Policy actions that has to be configurable.

3.5. Quote Comparison Component (QCC) support

There may be different capping factors per variation in QCC so every variation will store separate Capping as well as Capping Configuration and Override information.

Possible scenarios:

- Capping Factors may be different in each variation
- Some variation(s) may have Capping applied, but other(s) - not
- Some variation(s) may have Capping Override applied, but other(s) - not

In general this means that every variation will have to have a set of Premium Capping information at any level required.

To achieve this - a new entity *CappingVariation* needs to be created and added to each level.

Capping Variation Entity

Attribute Name	Type	Description
quoteVariation	QuoteVariation	A reference to a QuoteVariation entity
capping	PremiumCapping	A variation of PremiumCapping information for the given Quote Variation

Exigen Confidential

Domain Class Diagram
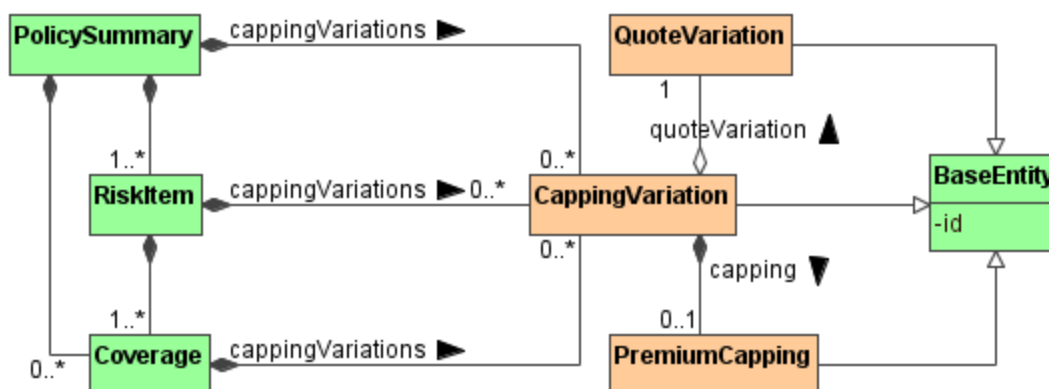
Create New Variation

When creating a new variation for a quote with policy plan is selected:

● All Capping Information (including configuration and override) is cleared / removed
● New Capping Information to be created during rating

When a non-plan variation is added:

● All Capping Information is cloned to the new variation

Select Variation

On Select Variation action - Capping information should be switched.

Clear Variation

When clear variation action is used - all Capping information should be cleared as well.

3.6. Extension points

Capping Components

Premium Capping Components should be regular PF components so extension should made in the same fashion as any other component.

Capping UI (Pop-up)

Since the Premium Capping UI is in a pop-up - some explicit action has to be taken to make it extendable.

Exigen Confidential

The popup XHTML code should be separated into parts that should not be overridden and parts that potentially could.

The popup contains the following parts:

- Button to show popup
- Popup contents:
    - Header
    - Input fields (with BLS rules applied)
    - Action buttons:
        - Apply
        - Cancel

It would make sense to create a separate XHTML file for each *Header* and *Input Fields* parts to allow easy customization of the popup, however I would recommend to keep the action buttons non-customizable.

Also, the "button to show popup" should be in a separate file for extension as some customers might require a different restriction e.g. different privilege, or transaction type or date. Some may not allow a mid-term Premium Capping Override.

There is a good example from Premium Override context: PremiumOverridePopup.xhtml

QCC Support for Capping

The whole Premium Capping object structure is meant to be created, switched & cleaned for each policy plan variation operation so no explicit support should be required.

4. Impact to other systems

4.1. Performance

We are adding multiple entity relationships at multiple levels so policy save, load & update operations will take a little longer.

4.2. Policy Product Configuration

New UI components are created and they will have to be configured in the product(s) if the feature is about to be used in the Product.

Exigen Confidential

4.3. Policy Premium Override

Premium Override is applied on top of the premium returned from the (OpenL) Rating Engine meaning on top of Capped Premium.

4.4. Policy Premium Prorating

No impact

4.5. Preconfigured / Existing Product(s)

To test the feature - we will need to configure one of the Preconfig products to use Capping UI

4.6. Policy XML Serialization / Deserialization

As new components are introduced they will have to be taken into account when creating a Policy XML for import & etc.

4.7. Conversion

Even for conversion the same process should be followed. Capping information may be imported over XML as regular component data, so if a continuous capping needs to be applied on the next renewal - regular process may be used.

4.8. Taxes, Fees & Commissions

Taxes, Fees & Commissions are applied on top of the prorated premiums so they will receive the capped premium implicitly.

4.9. Ledger

No impact

4.10.   Reporting

A new report is expected to be created for Capping, requirements should be provided separately.

Exigen Confidential

Attachments

1. [EIS_AAACappingRequirements_07](#)

Attachment 1 EIS AAA Capping Requirements

Exigen Confidential

Page **25 of 25**