**Interface Specification Document -**

**Channel Portal & Core Income Systems**

Author : Dileep Singh

Prepared Date : 08-May-2018

Document Version : 1.1b

Update Date : 18-Jun-2018

Approval form

|  |  |  |
| --- | --- | --- |
| **PREPARED BY:** |  |  |
|  | Dileep Singh | Dated |
|  | Title: Solution Architect, Majesco | 08-Feb-2018 |
|  |  |  |
|  |  |  |
| **UPDATED BY:** |  |  |
|  | <> | Dated |
|  | Title: <> |  |
|  |  |  |
|  |  |  |
| **REVIEWED BY:** |  |  |
|  | <> | Dated |
|  | Title: <> |  |
|  |  |  |
|  |  |  |
|  | <> | Dated |
|  | Title: <> |  |
|  |  |  |
|  |  |  |
|  | <> | Dated |
|  | Title: <> |  |
|  |  |  |
|  |  |  |
| **ACCEPTED & APPROVED BY:** |  |  |
|  | <> | Dated |
|  | Title: <> |  |
|  |  |  |
|  | <> | Dated |
|  | Title: <> |  |
|  |  |  |
|  | <> | Dated |
|  | Title: <> |  |
|  |  |  |
|  | <> | Dated |
|  | Title: <> |  |
|  |  |  |
|  |  |  |
|  |  |  |
|  |  |  |

Document History

**Assignment**

| Task No. | Task Name | Assigned By | Assigned To |
| --- | --- | --- | --- |
|  |  |  |  |
| 1 |  |  |  |

**Change Record**

| Date  (dd-Mon-yyyy) | Primary Author | Version | Change Reference |
| --- | --- | --- | --- |
|  |  |  |  |
| 08-May-2018 | Dileep Singh | 0.1 | Initial Draft |
| 15-May-2018 | Dileep Singh | 0.2 | Revised version with addition of sample data & falg for Incremental refresh |
| 16-May-2018 | Dileep Singh | 0.3 | Majesco Internal changes |
| 17-May-2018 | Dileep Singh | 0.4 | Revised approach from file based to table based. |
| 18-May-2018 | Dileep Singh | 1.0 | Baseline version |
| 13-Jun-2018 | Dileep Singh | 1.1 | Modified staging table structure, added Cash Value table, policy detail interface & MMO Interface |
| 14-Jun-2018 | Dileep Singh | 1.1a | Removed Cash value table and added web service for the same |
| 15-Jun-2018 | Dileep Singh | 1.1b | Added MMO fields in CP\_POL\_CLIENTS table, removed % & - from Policy Detail interface field names & added mockup 38.xlsx in Appendix. |

**Distribution**

| Date | Distributed To | Version | Department |
| --- | --- | --- | --- |
| 13 Sept 2004 | HODs and SMEs (GELS) | 1.2 | Customer Service |
|  |  |  |  |
|  |  |  |  |
|  |  |  |  |
|  |  |  |  |
|  |  |  |  |
|  |  |  |  |
|  |  |  |  |
|  |  |  |  |
|  |  |  |  |
|  |  |  |  |
|  |  |  |  |
|  |  |  |  |
|  |  |  |  |
|  |  |  |  |

Table of Content

[1. Introduction 6](#_Toc516768744)

[1.1 Overview 6](#_Toc516768745)

[1.2 Purpose 6](#_Toc516768746)

[1.3 Scope 6](#_Toc516768747)

[1.4 Definition/ Glossary of Terms 11](#_Toc516768748)

[2. Interface- Policy/Proposal, Customer Dues & Dashboard Summary 12](#_Toc516768749)

[2.1 Overview 12](#_Toc516768750)

[2.2 Reference Documents 12](#_Toc516768751)

[2.3 Interface Process Flow 12](#_Toc516768752)

[2.3.1 Policy/Proposal Summary 12](#_Toc516768753)

[2.3.2 Customer Outstanding dues 15](#_Toc516768754)

[2.3.3 Dashboard Summary 15](#_Toc516768755)

[2.4 Frequency 16](#_Toc516768756)

[2.5 Exception Handling 16](#_Toc516768757)

[2.6 Interface Definition Document 17](#_Toc516768758)

[2.7 Cross Reference 17](#_Toc516768759)

[2.8 Open Items 18](#_Toc516768760)

[2.9 Assumptions 18](#_Toc516768761)

[2.10 External System Dependency 18](#_Toc516768762)

[3. Interface- Policy/Proposal Detail Interface 19](#_Toc516768763)

[3.1 Overview 19](#_Toc516768764)

[3.2 Reference Documents 19](#_Toc516768765)

[3.3 Interface Process Flow 20](#_Toc516768766)

[3.4 Frequency 21](#_Toc516768767)

[3.5 Exception Handling 21](#_Toc516768768)

[3.5.1 Exception Handling 21](#_Toc516768769)

[3.5.2 Timeout Settings 22](#_Toc516768770)

[3.6 Interface Definition Document 22](#_Toc516768771)

[3.7 Cross Reference 22](#_Toc516768772)

[3.8 Open Items 22](#_Toc516768773)

[3.9 Assumptions 22](#_Toc516768774)

[3.10 External System Dependency 23](#_Toc516768775)

[4. Interface- Customer Detail Interface 24](#_Toc516768776)

[4.1 Overview 24](#_Toc516768777)

[4.2 Reference Documents 24](#_Toc516768778)

[4.3 Interface Process Flow 25](#_Toc516768779)

[4.4 Frequency 26](#_Toc516768780)

[4.5 Exception Handling 26](#_Toc516768781)

[4.5.1 Exception Handling 26](#_Toc516768782)

[4.5.2 Timeout Settings 26](#_Toc516768783)

[4.6 Interface Definition Document 26](#_Toc516768784)

[4.7 Cross Reference 27](#_Toc516768785)

[4.8 Open Items 27](#_Toc516768786)

[4.9 Assumptions 27](#_Toc516768787)

[4.10 External System Dependency 27](#_Toc516768788)

[5. Interface- Agent Photo Interface 28](#_Toc516768789)

[5.1 Overview 28](#_Toc516768790)

[5.2 Reference Documents 28](#_Toc516768791)

[5.3 Interface Process Flow 29](#_Toc516768792)

[5.4 Frequency 30](#_Toc516768793)

[5.5 Exception Handling 30](#_Toc516768794)

[5.5.1 Using web service call 30](#_Toc516768795)

[5.6 Interface Definition Document 30](#_Toc516768796)

[5.7 Cross Reference 30](#_Toc516768797)

[5.8 Open Items 30](#_Toc516768798)

[5.9 Assumptions 30](#_Toc516768799)

[5.10 External System Dependency 31](#_Toc516768800)

[6. Interface- MMO Interface 32](#_Toc516768801)

[6.1 Overview 32](#_Toc516768802)

[6.2 Reference Documents 32](#_Toc516768803)

[6.3 Interface Process Flow 33](#_Toc516768804)

[6.4 Frequency 33](#_Toc516768805)

[6.5 Exception Handling 33](#_Toc516768806)

[6.5.1 Using web service call 33](#_Toc516768807)

[6.6 Interface Definition Document 33](#_Toc516768808)

[6.7 Cross Reference 33](#_Toc516768809)

[6.8 Open Items 34](#_Toc516768810)

[6.9 Assumptions 34](#_Toc516768811)

[6.10 External System Dependency 34](#_Toc516768812)

[7. Interface- LI Policy Cash Value 35](#_Toc516768813)

[7.1 Overview 35](#_Toc516768814)

[7.2 Reference Documents 35](#_Toc516768815)

[7.3 Interface Process Flow 36](#_Toc516768816)

[7.4 Frequency 36](#_Toc516768817)

[7.5 Exception Handling 36](#_Toc516768818)

[7.5.1 Using web service call 36](#_Toc516768819)

[7.6 Interface Definition Document 36](#_Toc516768820)

[7.7 Cross Reference 36](#_Toc516768821)

[7.8 Open Items 36](#_Toc516768822)

[7.9 Assumptions 37](#_Toc516768823)

[7.10 External System Dependency 37](#_Toc516768824)

[8. Appendix 38](#_Toc516768825)

# Introduction

## Overview

Income is an Insurance company which deals with multiple Line of Business. They have varied users who want to access their business portfolio and post sales needs. Channel portal is a one stop solution front end facing web application wherein the agent, manager and business supporting teams can login and see the dashboard which gives the overview of the business and task. The channel portal will have various menus and pages which will be channel and role based which the agent, manager and business supporting teams can access. The user is referred to as the Manager or the Agent or any authorized users who are granted access to the channel portal.

## Purpose

This document captures the Integration details required to interface Majesco Channel Portal system with other core income systems. Once integrated this new system will allow all Income agents to login into Channel Portal and have a consolidated view of all their clients, proposals, policies & dashboards irrespective of the line of business for these policies.

## Scope

The list of interfaces for Channel Portal Phase 1A is as listed below:

|  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- |
| **Sr. No.** | **Interface Description** | **Business Description** | **Source System** | **Target System** | **Integration Mechanism** | **Applicability for DMS (Yes/No)** | **Incremental**  **(Y/N)** | **Use Oracle ESB**  **(Y/N)** |
| 1 | Policy/Proposal Summary | This will fetch policy/proposal listing details from core systems and populate into Channel Portal database | All Core systems | Channel Portal | Table Based |  | Y | N |
| 2 | Customer Outstanding Dues | This will fetch Outstanding dues as per requirement for all the policies as on that day | eBaoLI | Channel Portal | Table Based |  | N | N |
| 2 | Dashboard Summary | This interface will capture dashboard summary box details at individual policy level ex. Policies Maturing, Failed GIRO transactions. | All Core systems | Channel Portal | Table Based |  | N | N |
| 3 | Policy/Proposal Detail | This will fetch complete policy/proposal details from core systems. These data will not be stored in Channel Portal database | All Core systems | Channel Portal | REST-JSON |  | - | Y |
| 4 | Customer Detail interface | This interface will capture customer details along with its contact info. | Entity | Channel Portal | SOAP |  | - | Y |
| 5 | Agent Photo | This web service will be used to fetch Agent profile photo that is stored in Income DB. | OnService Shared Service | Channel Portal | SOAP |  | - | N |
| 6 | Customer Age | This web service will be used to get policy holder age based on logic in individual core system | All core systems | Channel Portal | REST-JSON |  | - | Y |
| 7 | MMO Details | This web service will be used to get MMO details for Income customers | Entity Systems | Channel Portal | SOAP |  | - | Y |

**Table 1: Interface List**

Above list of interfaces will cater to below screens available in mockupv35.xlsx under Channel Portal FSD. Same mockupv35.xlsx is also attached in Appendix below for ready reference.

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| **Sr.No.** | **Screens** | **Sections** | **Source** | **Interface Name** | **Remarks** |
|  | Dashboard | Summary Boxes | CP DB | - | CP DB will be populated by Dashboard Summary Interface |
|  | Dashboard | Submitted Proposals | CP DB | - | CP DB will be populated by Proposal Summary Interface |
|  | Dashboard | Recently Issued Policies for the past 3 months | CP DB | - | CP DB will be populated by Policy Summary Interface |
|  | Customer Listing - 1 | Portfolio by Line of Business | CP DB | - | CP DB will be populated by Policy Summary Interface |
|  | Customer Listing -2 | Client search, Portfolio | CP DB | - | CP DB will be populated by Policy Summary Interface |
|  | Customer Detail - Dashboard | 3 Dashboard boxes, Bar graph | CP DB | - | CP DB will be populated by Policy Summary Interface |
|  | Customer Detail - Dashboard | Outstanding Loan/APL | CP DB | - | CP DB will be populated by Policy Summary Interface |
|  | CM - Policy Summary Cards | All cards | CP DB | - | CP DB will be populated by Policy Summary Interface |
|  | CM - Policy Summary Rows | All cards | CP DB | - | CP DB will be populated by Policy Summary Interface |
|  | CM - Policy - Details – Life  CM - Policy - Details – DPS  CM - Policy - Details – IS  CM - Policy - Details – Eldersh  CM - Policy - Details – GI  CM - Policy - Details – Affinit  CM - Policy - Details – Group  CM - Policy - GI Insured  CM - Policy – Insured  CM - Policy – Transac  CM - Policy – Documents  CM - Policy - Fund Details  CM Detail - Customer Info  Company – Policy  CM - Policy - General Group  CM - Policy - Insured Group  CM - Policy - Transac Group | Customer Brief | CP DB | - | CP DB will be populated by Policy Summary Interface.  In this interface required customer details are also being fetched. |
|  | CM - Policy - Details - DPS | All except Customer Brief | REST JSON Service | Policy/Proposal Detail interface |  |
|  | CM - Policy/Proposal - Details – IS | All except Customer Brief | REST JSON Service | Policy/Proposal Detail interface |  |
|  | CM - Policy/Proposal - Details - Eldersh | All except Customer Brief | REST JSON Service | Policy/Proposal Detail interface |  |
|  | CM - Policy/Proposal - Details - GI | All except Customer Brief | REST JSON Service | Policy/Proposal Detail interface |  |
|  | CM - Policy/Proposal - Details - Affinit | All except Customer Brief | REST JSON Service | Policy/Proposal Detail interface |  |
|  | CM - Policy/Proposal - Details - Group | All except Customer Brief | REST JSON Service | Policy/Proposal Detail interface |  |
|  | CM - Policy/Proposal - GI Insured | All except Customer Brief | REST JSON Service | Policy/Proposal Detail interface |  |
|  | CM - Policy/Proposal - Insured | All except Customer Brief | REST JSON Service | Policy/Proposal Detail interface |  |
|  | CM - Policy/Proposal - Transac | All except Customer Brief | REST JSON Service | Policy/Proposal Detail interface |  |
|  | CM - Policy/Proposal - Documents | All except Customer Brief | REST JSON Service | Policy/Proposal Detail interface |  |
|  | CM - Policy/Proposal - Fund Details | All except Customer Brief | REST JSON Service | Policy/Proposal Detail interface |  |
|  | CM Detail - Customer Info | Customer Information tab | SOAP | Customer Detail interface |  |
|  | Customer Listing - Company | Customer search | CP DB | - | CP DB will be populated by Policy/Proposal Summary Interface.  In this interface required customer details are also being fetched. |
|  | Customer Listing - Company | Portfolio by Line of Business | CP DB | - | CP DB will be populated by Policy/Proposal Summary Interface |
|  | Customer Listing - Company pg 2 | Portfolio by Line of Business | CP DB | - | CP DB will be populated by Policy/Proposal Summary Interface |
|  | Company - Policy/Proposal | Group Business | CP DB | - | CP DB will be populated by Policy/Proposal Summary Interface |
|  | Company - Policy/Proposal | View Details | REST JSON Service | Policy/Proposal Detail interface | Will be real-time |
|  | CM - Policy/Proposal - General Group | All except Customer Brief | REST JSON Service | Policy/Proposal Detail interface | Will be real-time |
|  | CM - Policy/Proposal - Insured Group | All except Customer Brief | REST JSON Service | Policy/Proposal Detail interface | Will be real-time |
|  | CM - Policy/Proposal - Transac Group | All except Customer Brief | REST JSON Service | Policy/Proposal Detail interface | Will be real-time |
|  | Policy/Proposal Listing | Policy/Proposal Listing (Details) | CP DB | - | CP DB will be populated by Policy/Proposal Summary Interface |
|  | Policy/Proposal Listing | Policy/Proposal Listing (Details) 🡪 Policy/Proposal No | REST JSON Service | Policy/Proposal Detail interface | Will be real-time |
|  | Agent Photo | Agent Phot (Top Right Section) | SOAP | Agent Photo Interface | Will be real-time |

**Table 2: CP Screens & Interface Mapping List**

Overall lifespan of the policies to be considered are as below:

|  |  |  |
| --- | --- | --- |
|  |  | **Channel Portal** |
| Life | Inforce | No validity period |
|  | Lapsed | No validity period |
|  | Terminated | No validity period |
| Health | Inforce | Show latest renewal only |
|  | Lapsed | Show latest policy. No validity period |
|  | Terminated | Show latest policy. No validity period |
| GI | Inforce | Show latest renewal only |
|  | Terminated | Show latest policy. No validity period |
| Group | Inforce | No validity period |
|  | Lapsed | Show latest policy. No validity period |
|  | Terminated | Show latest policy. No validity period |
| Affinity | Inforce | No validity period |
|  | Lapsed | Don’t show |
|  | Terminated | Don’t show |

**Table 3: Policy Lifespan**

## Definition/ Glossary of Terms

| **Terms/ Abbreviations /Acronyms** | Description |
| --- | --- |
| CP | Channel Portal |
| DB | Database |
| ESB | Enterprise Service Bus |

**Table 3: Glossary**

# Interface- Policy/Proposal, Customer Dues & Dashboard Summary

## Overview

Policy/Proposal & Dashboard summary will follow a database table based approach. Newly created policies or proposals along with their updates on them will be written to a staging table in each core system and the same will be consumed and stored in channel portal database system. On similar lines, even required dashboard details will be retrieved in day-end batch.

## Reference Documents

| **No.** | Title of Document | Version |
| --- | --- | --- |
| 1 | FSD - Income\_Channel\_Portal\_v1 0a.doc | 1.0a |
|  |  |  |

## Interface Process Flow

### Policy/Proposal Summary

Policy/Proposal summary interface will be used to capture basic field details of Policies & Proposals. These fields will generally be the listing details and card details that would be seen on CP screens. These details will directly be stored in CP database and will be used to display data on screens if there is any enquiry for any of these fields based on policy number/proposal number.

Process flow for population of these data in CP database is as depicted below:

DPS

T1/F1

Elder

Shield

Life, Annuity & Group

eBaoLI

eBaoGI

Life & Group

T2/F2

IncomeShield

Channel Portal

General

T3/F3

Data read

Incremental

Service/Tool

CP DB

DB Write

Health

T4/F4

Health

T5/F5

MCS

Health-Group

T6/F6

MHS

Health

T7/F7

* TX/FX are Time & Frequency for reading data from individual system that is to be decided.

**Fig 1 : Policy/Proposal Summary Interface**

As per the process flow depicted in Fig 1 above, individual core systems viz. Life, Health, General, Group and Affinity will dump their data in a table structure provided by Majesco. List of tables is as below:

|  |  |  |
| --- | --- | --- |
| **Sr. No.** | **Table Name** | **Remarks** |
| 1 | CP\_POLICY\_DTLS | Used to capture basic Proposal/Policy details |
| 2 | CP\_POLICY\_CLIENTS | Used to capture all customer types attached to a Proposal/Policy |
| 3 | CP\_POL\_CLIENT\_ADDR | Captures address details for the Policy/Proposal customers |
| 4 | CP\_POL\_CLIENT\_PHONE | Captures phone details for the Policy/Proposal customers |
| 5 | CP\_POL\_CLIENT\_EMAIL | Captures email details for the Policy/Proposal customers |

**Table 3: Table List**

Column level details for the above mentioned tables are attached under section 2.6 Interface Specification Documents.

All these tables need to be populated on incremental basis based on last population timestamp.

Once data in these tables are populated by core systems on their side, the same will be copied to the database of Channel portal system.

Transfer of data from staging tables in individual core system to CP staging table will be through a separate “Interface Manager”. The same is being discussed with Project One team and shall be finalized based on effort approval.

Also, individual core system should ensure that before populating these staging tables, they have purged the earlier data present in these tables.

Frequency & timing of these data transfer to be decided with core team and business users.

Additionally, note that CP is an enquiry system and will show whatever field values is made available to it. Hence, individual core system can populate selective fields as per their own system feature.

**Sample illustration 1:**

Core system has new transactions as tabulated below:

|  |  |  |  |
| --- | --- | --- | --- |
| **Sr.No.** | **Proposal/Policy** | **Action** | **Updation** |
| 1 | Prop1 | New proposal |  |
| 2 | Pol1 | New policy |  |
| 3 | Pol2 | Update on policy | Frequency : Monthly(12) 🡪 Quarterly(4) |
| 4 | Pol3 | Update on policy customer | 1. DOB : 14-Jan-1983 🡪 15-Feb-1984 2. Gender : Male (1) 🡪 Female(2) |

For this assumed core system process would be as listed below:

1. Insert into CP\_DATA\_TRANSFER\_LOG table with values in NTRANSFERLOGSEQ, NINTFTYPE & DTLOADSTRT with latest sequence number, 1 & Start time respectively.
2. Populate tables as below :
   1. For Prop1: CP\_POLICY\_DTLS (NISPOLICY=0), CP\_POL\_CLIENTS, CP\_POL\_CLIENT\_ADDR, CP\_POL\_CLIENT\_EMAIL & CP\_POL\_CLIENT\_PHONE.
   2. For Pol1: CP\_POLICY\_DTLS (NISPOLICY=1 & NISNEWPOLICY=1). No need to populate CP\_POL\_CLIENTS, CP\_POL\_CLIENT\_ADDR, CP\_POL\_CLIENT\_EMAIL & CP\_POL\_CLIENT\_PHONE unless there is any change in these data.
   3. For Pol2: CP\_POLICY\_DTLS (NISPOLICY=1 & NISNEWPOLICY=0) with corresponding values in all the columns along with value in column NPMTFREQ. Other tables need not be populated.
   4. For Pol3: A single record for this needs to be populated in CP\_POL\_CLIENTS table with values in all columns including updated value in DTBIRTH & NGENDERCD. Other tables need not be populated as only DOB & GENDER has been changed.
3. On successful population of these staging tables, update table in step 1 i.e. CP\_DATA\_TRANSFER\_LOG for columns DTLOADEND & NLOADSTATFLG with end time & 1 successfully.
4. Data will be transferred using a separate service that will read records from all tables against NTRANSFERLOGSEQ for which DTLOADEND is not null & NLOADSTATFLG = 1.

### Customer Outstanding dues

Customer outstanding dues, is applicable only for eBaoLI system. Unlike, above interface, staging table “CP\_CLIENT\_OSDUES” will be populated only during day end and in full. Population of this table would only be applicable for those policies that have any outstanding due, loan or APL.

For policies that don’t have any of the dues outstanding, need not be populated. Before any population, core team would need to ensure that this table has been completely purged.

CP system will read this table and populate corresponding table on CP side with previous day data completely purged on its side as well.

Only one record per policy is expected in this staging table else an error will be logged.

### Dashboard Summary

Dashboard Summary data will exactly follow the same process as Customer Outstanding Dues interface. It will populate data in CP\_POL\_DASHBORAD\_DTLS table.

Dashboard table data population will be an end of day batch and will be in full. It will be applicable for those policies only that have Outstanding dues, Giro transactions, Renewal dues and Maturing policies. For policies that don’t have any of these parameters, need not be populated. Once the core system batch completes, required data needs to be populated in its table post purging of previous day data.

Fields from 9-21, provided in Interface Definition Document under section 2.6, are derived fields and individual core team would need to write logic as per details provided in “Channel Portal Fields v12.xlsx” attached herewith in the Appendix section below.

Core team would need to ensure that the table contains only one record per policy. In case of multiple records, CP will log an error for the policy in CP DB and details for such policies will not be updated in CP actual tables i.e if a policy is due for maturity & also has failed GIRO transaction, still system should log a single policy in the table and update separate columns for both the transactions.

## Frequency

| **No.** | Business Events | Frequency | Timings |
| --- | --- | --- | --- |
| 1 | Life system (eBao-LI) | Daily | TBD |
| 2 | Health system (WBIS-IS, J2EE-DPS/MHS, FFA- Eldershield) | Daily | TBD |
| 3 | Group system (J2EE-MCS, eBao-LI- EB/GB/GTL) | Daily | TBD |
| 4 | General system (eBao-GI) | Daily | TBD |
| 5 | Affinity (eBao-LI) | Daily | TBD |
| 6 | Customer outstanding dues from Life system (eBao-LI) | Daily | Day-end |
| 7 | Dashboard Summary | Daily | Day-end |

**Table 4: Interface Frequency**

## Exception Handling

Exception handling mechanism will be as below:

1. Exception occurred during transfer of data from individual core systems to CP DB.
   1. Technical error: CP will log a generic error with appropriate error description in CP error table and/or application log.
   2. Functional/Business error: CP will log error at individual Policy/Proposal level with appropriate error description in CP error table.
2. Exception occurred while inserting/updating data in actual portal tables :
   1. Technical error: CP will log a generic error with appropriate error description in CP error table.
   2. Functional/Business error: CP will log error at individual Policy/Proposal level with appropriate error description in CP error table.

In addition to above handling, an email will also be sent separately to support teams and other concerned persons for the status of each run. If any error, corresponding error details will also be shared.

## Interface Definition Document

Interface document for the Policy/Proposal Summary interface for staging tables are as attached below:

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **Sr. No.** | **Interface Name** | **Table Name** | **Table Structure** | **Incremental (Y/N)** |
| 1 | Policy/Proposal Summary | CP\_POLICY\_DTLS |  | Y |
| 2 | Policy Client details | CP\_POL\_CLIENTS |  | Y |
| 3 | Policy Client Address details | CP\_POL\_CLIENT\_ADDR |  | Y |
| 4 | Policy Client Phone details | CP\_POL\_CLIENT\_PHONE |  | Y |
| 5 | Policy Client Email details | CP\_POL\_CLIENT\_EMAIL |  | Y |
| 6 | Customer Outstanding Due details | CP\_CLIENT\_OSDUES |  | N |
| 7 | Dashboard Summary details | CP\_POL\_DASHBORAD\_DTLS |  | N |

**Table 5: Table structure details**

## Cross Reference

| **No.** | Module | FSD Section No | Description |
| --- | --- | --- | --- |
| 1 | Customer & Policy/Proposal | FSD - Income\_Channel\_Portal\_v1 0a.doc – Appendix A – No.1 | Screen Layout Channel Portal v1.0a |
| 2 | Customer & Policy/Proposal | FSD - Income\_Channel\_Portal\_v1 0a.doc – Appendix A – No.3 | Channel Portal fields mapping created by Benny |

## Open Items

| **No.** | Item Description | Requirement Log Reference # |
| --- | --- | --- |
| - | - | - |

## Assumptions

* Individual core system will populate required tables on their side.
* Income team will transfer these data to CP database.
* Core team members are able to identify new as well as updated policies/proposals.

## External System Dependency

| **No.** | External System Name | Dependency Description | Income CCM, SM |
| --- | --- | --- | --- |
| 1 | eBaoLI (LI, Affinity, EB, GTL) | To fetch required incremental policy & proposal details | Man Cheng Cheng |
| 2 | eBAOGI | To fetch required incremental policy & proposal details | Christoper Lim |
| 3 | FFA | To fetch required incremental policy & proposal details | Balachandar Mahendran |
| 4 | DPS,MHS,MCS | To fetch required incremental policy & proposal details | Subramaniyam Hemalatha |
| 5 | IncomeShield | To fetch required incremental policy & proposal details | Mahesh Reddy/Thant Zin |

# Interface- Policy/Proposal Detail Interface

## Overview

Policy/Proposal detail interface will be invoked whenever an agent clicks on any Policy/Proposal number. This interface will be used to fetch complete details for a Policy/Proposal as required by channel portal screens. List of screens that will consume this interface is tabulated in Table 2 above.

## Reference Documents

| **No.** | Title of Document | Version |
| --- | --- | --- |
| 1 | FSD - Income\_Channel\_Portal\_v1 0a.doc | 1.0a |

## Interface Process Flow



Initial Login (1)

Response (13)

Policy/Proposal

Service

Channel Portal

Policy/Proposal

**Channel**

**Portal**

Request (3)

Initial Listing (2)

Response (12)

Request (4)

CP DB

Policy/Proposal

Web Service



Oracle ESB

Service

Request (5)

**Oracle ESB**

Response (11)

Core system Web service

Policy/Proposal

Web Service

Response (10)

Request (6)

Response (9)

Income Core Systems

**Core System**

Request (7)

Request/Response (8)

Core System DB

This interface will be a REST service and will be used to fetch data in real time from Income core systems. Whenever a user wants to see complete details of any Policy/proposal from any screen, user will click on that particular policy. On clicking the policy, CP will internally invoke a service i.e. Request 3 that would pass request for details to Oracle ESB (Request 4 & 5). This request will be in JSON format and will contain values for Policy number and Line of Business (LOB). Based on the received Policy number & LOB, Oracle ESB in turn internally need to communicate with corresponding core system (based on LOB i.e. Request 6 & 7) to fetch the required details of passed in policy.

Based on the policy number, core system will revert to Oracle ESB with required details like General Info for Main Plan & Rider, Insured Info for the policy and Transactions on the policy etc. by fetching data from its own database (Request/Response 8). These details will be passed on back to Oracle ESB (Response 10 & 11). Once received, Oracle ESB will transfer the data back to channel portal service & in turn to Channel Portal screen (Response 12 & 13).

This entire REST service communication will be seamless and transparent to end user.

For the exact list of screen that will be based on this REST web service, please refer to Table 2 above.

Additionally, note that CP will call ESB service as per the system under which policy is tagged i.e. there will be one service per Income core system. However, the JSON format for each of this service will remain the same. Variables of JSON will be populated as per the system feature.

Document details required in JSON need to be shared only by eBaoLI. Other systems need not populate document related details. As part of interface, it is expected that LI system will pass the URL for the document and the same will be displayed in CP. User will click on this link to open the document in the browser. As of now, the document can open only in IE 11. If a user needs to open this document, then he will have to login in Channel Portal using IE11. CP does not provide interoperability between browsers i.e. login into CP using Chrome and then opening the document in IE11.

## Frequency

| **No.** | Business Events | Frequency | Timings |
| --- | --- | --- | --- |
| 1 | Policy Proposal Detail Interface | Real Time | - |

**Table 7: Glossary**

## Exception Handling

### Exception Handling

Every response message will have a Result node containing attributes as errCode & errMessage.

Whenever there is any exception these nodes need will be populated as tabulated below (sample exception) :

|  |  |  |
| --- | --- | --- |
| errCode (Error Code) | errMessage (Error Message) | Remark |
| 1 | Validation Error: Please pass the proper input details | Usually for missing mandatory fields or incorrect data type. |

### Timeout Settings

Timeout settings for these service will be set to 90 seconds in case Oracle ESB fails to respond.

## Interface Definition Document

Please refer to sheets “Service Summary” & “Mapping Details” in excel attached below.



## Cross Reference

| **No.** | Module | FSD Section No | Description |
| --- | --- | --- | --- |
| 1 | Channel Portal | FSD - Income\_Channel\_Portal\_v1 0a.doc | This document details Channel Portal screens & its fields. |

## Open Items

| **No.** | Item Description | Requirement Log Reference # |
| --- | --- | --- |
| - | - | - |

## Assumptions

* All web-service communication with core systems will be through Oracle ESB.
* Channel Portal will expose a single service and Oracle ESB will communicate with core system based on the line of business passed in the request body.
* Channel Portal is only consumer of the ESB service and hence any transformations required will happen at the Oracle ESB level.
* Since basic fields of policies/proposals are populated using DB interface, there might be mismatch in the values of certain fields.

## External System Dependency

| **No.** | External System Name | Dependency Description | Income CCM, SM |
| --- | --- | --- | --- |
| 1 | eBaoLI (LI, Affinity, EB, GTL) | To fetch required incremental policy & proposal details | Man Cheng Cheng |
| 2 | eBAOGI | To fetch required incremental policy & proposal details | Christoper Lim |
| 3 | FFA | To fetch required incremental policy & proposal details | Balachandar Mahendran |
| 4 | DPS,MHS,MCS | To fetch required incremental policy & proposal details | Subramaniyam Hemalatha |
| 5 | IncomeShield | To fetch required incremental policy & proposal details | Mahesh Reddy/Thant Zin |
| 6 | Oracle ESB | Would be the intermediate layer between CP & Core system for web service communication | Sunil Nallamilli |

# Interface- Customer Detail Interface

## Overview

Customer detail interface as name suggests would be used to fetch customer details from Income Entity system using web service.

## Reference Documents

| **No.** | Title of Document | Version |
| --- | --- | --- |
| 1 | FSD - Income\_Channel\_Portal\_v1 0a.doc | 1.0a |

## Interface Process Flow



Initial Login (1)

Response (13)

Client

Service

Channel Portal

Client

**Channel**

**Portal**

Request (3)

Initial Listing (2)

Response (12)

Request (4)

CP DB

Customer

Web Service



Oracle ESB

Service

Request (5)

**Oracle ESB**

Response (11)

Core system Web service

Customer

Web Service

Response (10)

Request (6)

Response (9)

Entity Systems

**Core System**

Request (7)

Request/Response (8)

Core System DB

This interface will be a SOAP web service and will be used to fetch data in real time from Income core system called Entity. When an agent logs into Channel Portal and clicks on any one of his searched customers, CP will internally invoke a service i.e. Request 3 that would pass request for details to Oracle ESB (Request 4 & 5). This request will contain Customer id (i.e. NRIC number) in its Request body. Based on the received Id, Oracle ESB in turn internally will need to communicate with Entity system (Request 6 & 7) to fetch the required details of the customer.

Based on the customer id, core system will revert to Oracle ESB with required details like Gender, DOB, Address, Email etc. by fetching data from its own database (Request/Response 8). These details will be passed on back to Oracle ESB (Response 10 & 11). Once received, Oracle ESB will transfer the data back to channel portal service & in turn to Channel Portal screen (Response 12 & 13).

This service will be used in CM Detail - Customer Info🡪Customer Info. Same has also been mentioned in Table 2 above.

## Frequency

| **No.** | Business Events | Frequency | Timings |
| --- | --- | --- | --- |
| 1 | Customer Detail Interface | Real Time | - |

**Table 8: Customer Interface Frequency**

## Exception Handling

### Exception Handling

Every response message will have a Result node containing attributes as errCode & errMessage.

Whenever there is any exception these nodes need will be populated as tabulated below (sample exception) :

|  |  |  |
| --- | --- | --- |
| **errCode (Error Code)** | **errMessage (Error Message)** | **Remark** |
| 1 | Validation Error: Please pass the proper input details | Usually for missing mandatory fields or incorrect data type. |

### Timeout Settings

Timeout settings for these service will be set to 90 seconds in case Oracle ESB fails to respond.

## Interface Definition Document

Please refer to sheets “Service Summary” & “Mapping Details” in excel attached below.



## Cross Reference

| **No.** | Module | FSD Section No | Description |
| --- | --- | --- | --- |
| 1 | Customer |  | To display customer details |

## Open Items

| **No.** | Item Description | Requirement Log Reference # |
| --- | --- | --- |
| - | - | - |

## Assumptions

* Web-service communication with core systems will be through Oracle ESB.
* Channel Portal will fetch details through Oracle ESB which in turn will communicate with Entity system based on customerIdType (Identity Type) and customerID (Identity Number).
* CP will consume service from Oracle ESB and hence any transformations, if required, would be at Oracle ESB level.

## External System Dependency

| **No.** | External System Name | Dependency Description | Income CCM, SM |
| --- | --- | --- | --- |
| 1 | Oracle ESB | Would be the intermediate layer between CP & Core system for web service communication | Sunil Nallamilli |

# Interface- Agent Photo Interface

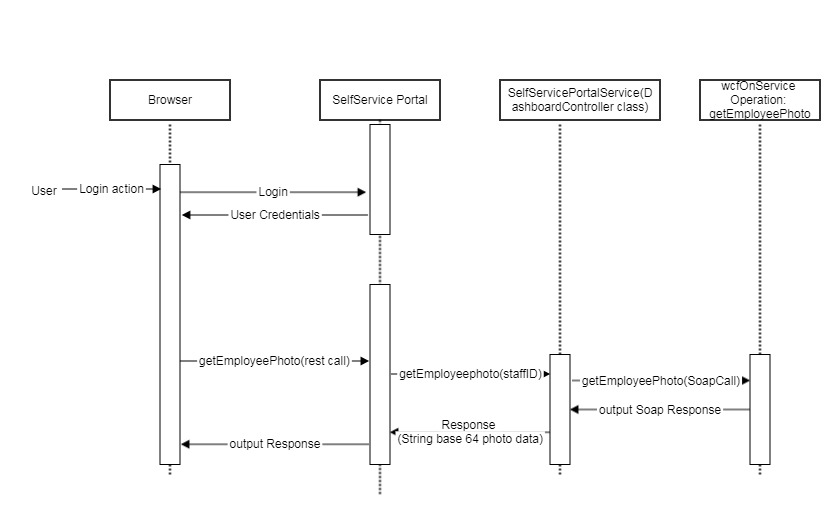
## Overview

Income has WcfOnService that allows to retrieve agent photo uploaded by HR. CP will use the same service esp. its component GetEmployeePhoto to get employee photo Income repository via soap web service call.

## Reference Documents

| **No.** | Title of Document | Version |
| --- | --- | --- |
| 1 | FSD - Income\_Channel\_Portal\_v1 0a.doc | 1.0a |
| 2 | wcfOnService\_Function Spec - WebAPI | - |

## Interface Process Flow



As per the above flow the User will login to the self-service portal. On successful login the user credential will be acquired. The portal will give a post rest call to the getEmployeePhoto in the application service layer and will pass the employee staff code and from this layer the SOAP web service WcfOnService operation getEmployeePhoto will be invoked via a soap call.

In the output from the WcfOnService the photo base 64string will be extracted and will be passed on to the front end to render the image on the browser. The prefix for the base 64 string passed to the Selfservice portal will be applied a the application end the final base 64 image string will be ‘**data:image/jpeg;base64,**’ + (the photo string in the <photo> tag returned from the getEmployeePhoto SOAP service).

This flow will be applicable for both individual agent and manager view.

For agents that don’t have their corresponding photos in the system, like non-staff agents, an image without any photo will be seen as shown below.





**Male Female**

## Frequency

| **No.** | Business Events | Frequency | Timings |
| --- | --- | --- | --- |
| 1 | Agent Photo Interface | Real Time | - |

## Exception Handling

### Using web service call

During web service call if there is any Business/Technical failure in the Web service call then the exceptions are logged in the application server logs. These logs are used to further verify with the web service owner for the root cause.

## Interface Definition Document



## Cross Reference

| **No.** | Module | FSD Section No | Description |
| --- | --- | --- | --- |
| 1 | Agent Summary | 3.1.2,4.5 | Agent Information section, Profile |

## Open Items

| **No.** | Item Description | Requirement Log Reference # |
| --- | --- | --- |
| - | - | - |

## Assumptions

* The Photo displayed will be fetched from the WcfOnService web service at run time Self-service portal will not be saving the image.
* If the photo is updated at the source end self-service portal will only be showing the image that is fetched from the WcfOnService.

## External System Dependency

| **No.** | External System Name | Dependency Description | Income CCM, SM |
| --- | --- | --- | --- |
| 1 | WcfOnService | Service to retrieve employee detail | Yip Kok Wei |

# Interface- MMO Interface

## Overview

Marketing Material Option (MMO) is captured in Entity system. An interface for CP needs to be designed that can retrieve various options under MMO.

## Reference Documents

| **No.** | Title of Document | Version |
| --- | --- | --- |
| 1 | MMODetails.docx (From Income ESB team) | - |

## Interface Process Flow

Income has an existing SOAP based service **MMODetails** containing an operation ***getMMODetails*** that helps to retrieve required MMO information. Input to this request is primarily Customer id i.e NRIC/FIN/Passport/UEN number that uniquely identifies an Income customer. Data regarding MMO is fetched from Income entity system. This service returns flag Y (Yes)/ N (No) for fields like eMail, Mail, Phone, SMS & Physical MMO. IDD for the interface is as attached under section 7.6.

CP screens that will reflect MMO details are as under:

* CM Detail - Customer Info
* Policy Listing

## Frequency

| **No.** | Business Events | Frequency | Timings |
| --- | --- | --- | --- |
| - | - | - | - |

## Exception Handling

### Using web service call

During web service call if there is any Business/Technical failure in the Web service call then the exceptions are logged in the application server logs. These logs are used to further verify with the web service owner for the root cause.

## Interface Definition Document



## Cross Reference

| **No.** | Module | FSD Section No | Description |
| --- | --- | --- | --- |
| - |  |  |  |

## Open Items

| **No.** | Item Description | Requirement Log Reference # |
| --- | --- | --- |
| - | - | - |

## Assumptions

* CP will consume existing Income MMO service.
* CP will communicate with Oracle ESB which in turn will retrieve data from Entity system.
* Entity has data for all customers available in all other Income core systems.

## External System Dependency

| **No.** | External System Name | Dependency Description | Income CCM, SM |
| --- | --- | --- | --- |
| - |  |  |  |

# Interface- LI Policy Cash Value

## Overview

Channel Portal needs to show real time cash value for all eligible LI policies. For this an existing web service in Entity will be used. Details of the same has been mentioned below.

## Reference Documents

| **No.** | Title of Document | Version |
| --- | --- | --- |
| 1 | eBAOLIPolicyDetails.docx (From Income ESB team) | - |

## Interface Process Flow

Income has an existing REST service **getEBaoLIPolicyDetails** that returns most of the parameters required for a policy from LI system. Of all the parameters returned by this service, Channel Portal specifically requires to display cash value. To retrieve this value, CP will be using this service.

ISD for the same is as attached under section 7.6.

## Frequency

| **No.** | Business Events | Frequency | Timings |
| --- | --- | --- | --- |
| 1 | Policy Summary Card View | Real Time | - |
| 2 | Policy Summary Row view | Real Time | - |

## Exception Handling

### Using web service call

During web service call if there is any Business/Technical failure in the Web service call then the exceptions are logged in the application server logs. These logs are used to further verify with the web service owner for the root cause.

## Interface Definition Document



## Cross Reference

| **No.** | Module | FSD Section No | Description |
| --- | --- | --- | --- |
| - |  |  |  |

## Open Items

| **No.** | Item Description | Requirement Log Reference # |
| --- | --- | --- |
| - | - | - |

## Assumptions

* Cash value is only applicable to LI policies.
* Existing ESB service returns Policy cash value.
* Logic for calculating cash value will be in LI system.

## External System Dependency

| **No.** | External System Name | Dependency Description | Income CCM, SM |
| --- | --- | --- | --- |
| 1 | eBaoLi | To get cash value | Man Cheng Cheng |
| 2 | ESB | An interface between CP & LI systems | Sunil Nallamilli |

# Appendix

| **No** | Description | Revision Date | Attachment |
| --- | --- | --- | --- |
| 1 | Channel Portal Mapping v12 |  |  |
| 2 | Channel Portal Mockup screen |  |  |