IKEA-AOD_PIP_v20

PIP

Application Overview Document for PIP

Document Control
Revision History

Date	Version	Author	Description
10-03-2014	0.1	Rajeev	Initial Draft
14-03-2014	0.2	Mandakini	Updated Initial Draft
09-11-2016	0.5	Mandakini	Updated till PIP Release 12.1.2
04-05-2017	0.6	Mandakini	Updated CDS Premium details
04-09-2017	0.9	Manish	Updated Log and POSlog retention change for PIP 14.0
04-10-2017	0.10	Priyanka	Updated ES Invoice flow for 13.3
11-10-2017	0.11	Mandakini	Updated China JMS security 14.0
02-05-2018	0.12	Mandakini	Updated China JMS security 14.1 Added functionality for EBCJMSBOS01 for Return WS call
17-05-2018	0.13	Mandakini	Added ICM-PIP integration functionality for 15.0
17-05-2018	0.14	Priyanka	Added brief description about webservice for UpdateRange.
24-07-2018	0.15	Priyanka	Updated ICM interface details for Find Interface
02-08-2018	0.16	Priyanka	Added disintegration details of IBIS report.
07-02-2020	0.17	Mandakini	Added changes in PIP from 15.0 to 19.1.1
22-06-2020	0.18	Malashri	Added changes in PIP with respect to SAP BOT and PIP Helpdesk Tool (19.3Release)
23-06-2020	0.19	Sumanth	Added changes in PIP with respect to ItemImport(19.3 release)
25-06-2020	0.20	Akshaya	Added changes in PIP item import functionality (19.3 Release)
20-11-2020	0.21	Malashri	Added changes in BusinessDaySalesSummary, EndOfDay ,SERVER SPECIFICATIONS ,NEW STORE SETUP,NEW SERVICE OFFICE INSTALLATION (19.5 Release)
23-11-2020	0.22	Saroj Kumar Das	Added PubSub Module. Implementing Google Cloud Pub/Sub.
27-04-2021	0.23	Mandakini	Added PCM/BCM integration with PIP (19.6 Release)
28-04-2021	0.24	Malashri	Added details on Payment on Collect (19.6 Release)
28-04-2021	0.25	Maharana Kishore	Added IsNotPrepaid details for MHS flow (19.6 Release)
28-04-2021	0.26	Nabarupa Saha	Added Document transaction with new ReasonCode and FullServe Element in SAREC (19.6 Release)
28-04-2021	0.27	Shivangi Miglani	Added SalesOrder offline job (19.6 Release)

Distribution

Name	Title	Document Version	Date

Approval Signatures

Name	Title	Document Version	Signature	Date
Manu Abraham Varghese				14-03-2014

Table of Contents

OBJECTIVES
The objective of the Application Overview Document (AOD) is to provide members of the delivery team with an overview of the application. The AOD describes the function of the application, the structure of the applications, the application configuration and the technical environment. This document will refer to any existing related documentation.

The AOD will be used to support the application and will always be kept current throughout the life of the application.

1.**Business**

1.1 Business Overview

IPOS (IKEA Point of Sale) is the name of the service containing the POS system including the integration point (PIP), TPNet which is the POS system in the stores and TPAdmin which is the central configuration and administration tool.

TPNet is the POS system (Point Of Sales) used in all IKEA stores in Europe, North America and Asia pacific, except for restaurants and franchise stores. The main purpose of POSsystem is to handle payment in cash and other media and produce a receipt to the customers. A POS system normally consists of one or more tills and some kind of back office application (client server concept) in TPNets case both a local and a central BackOffice. TPNet supports three different types of workstations groups: Regular tills, Return tills and ECO till (Express Check Out). An ECO till is basically a kiosk-till for the customers to use them self, register items and also pay for the purchase. ECO till have limited payment modes, only cards are possible to use, iPay, Bank, credit cards and Local cards.

TPAdmin is a central back office application used for administrates the stores and tills from a central point, especially regarding campaigns, configuration and monitoring.

PIP (POS Integration Point) is an integration application residing on the TPNet server, handling all communication to other dependency systems within IKEA.

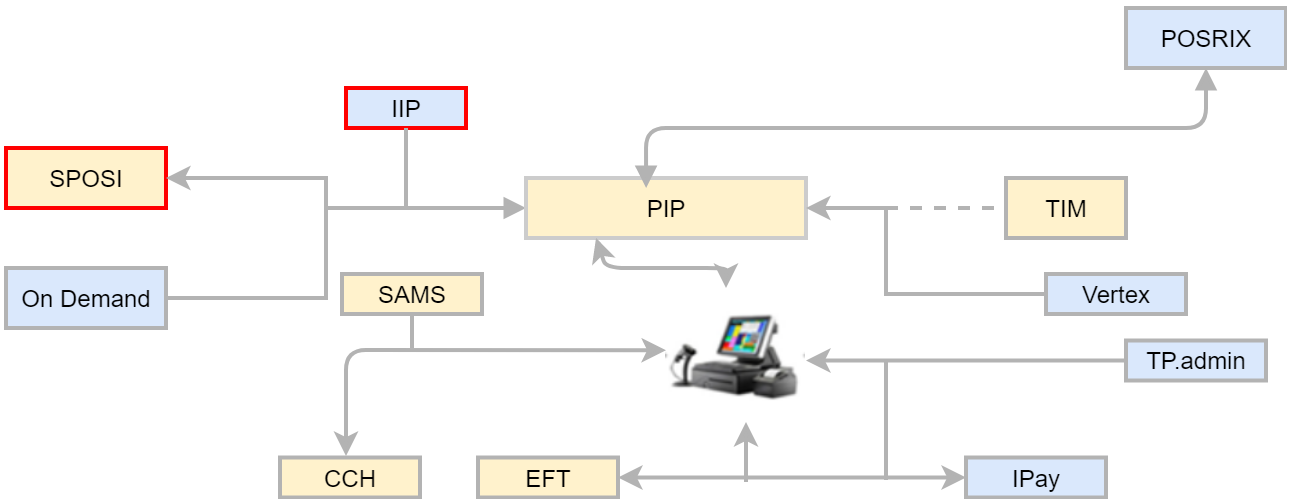
ISTX (IKEA Sales Tax) is an integration application residing on a Terminal server, handling all specific Canadian taxes in the SAMS returns flow.

RIX Adapter is a central application handling central additional article information to the TPnet backstore server.

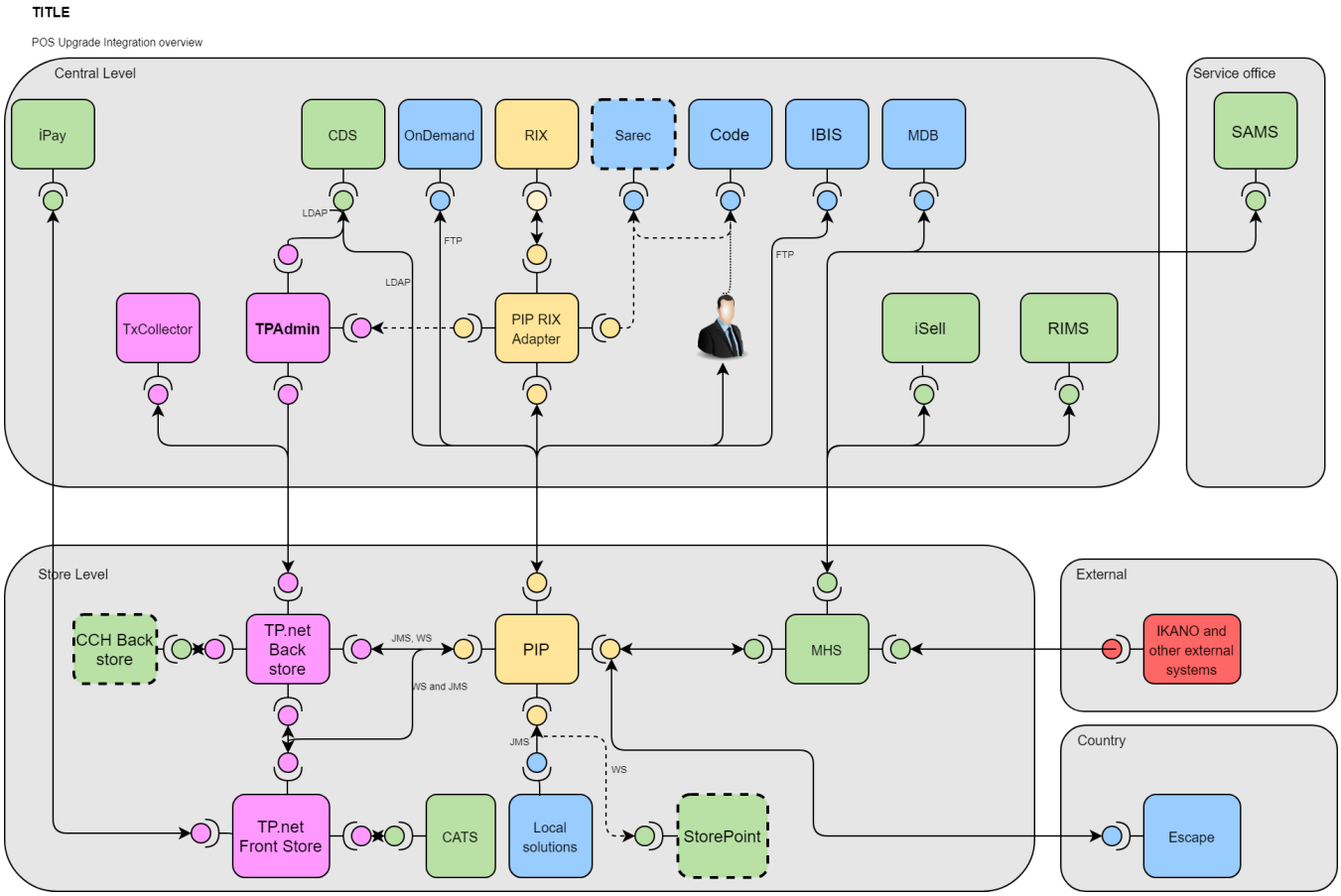
TX Collector is an application residing on the TP.net backstore server collecting sales data for TP Analyze.

PIP (POS Integration Point) an integration mechanisms that facilitate the communication between POS and IKEA legacy systems. This is the logical integration point not the physical and technical platform. The PIP is the store residing integration component which orchestrates almost all interactions within the Point Of Sale system and between the Points of Sale system and its dependency systems

1.2**Existing Payments Integration**



Business Processes
The purpose of the PIP, the Payment Integration Point, is to act as a store residing integration component. Within the PIP, no business logic is intended to exist. Instead, the PIP shall act as an orchestrator and data converter.
See the figure below for an overview of which systems the PIP interacts with. Note that not all systems and integrations are relevant for BV1.



1.3. Business Interrelationships

There are a number of Point Of Sale applications as well as other application iPay interacts with.

INTERFACE	PURPOSE
MHS	Store system handling functionality such as; Wharehouse management in SGF, C&C orders, Merchandise returns, Cash reconciliation, Sales space management.
TP.net Back store	TP.net Back store includes all functionality that is not done in the register. Typical functions that resides in the back store is for example Cash management, reporting, user administration.
RIX	Integration platform for the Retail area at central level
Store Point	Restaurant Application
CBD	Exchange Rate from the Global Accounting
STRLFORS & MDB	Import Range and Cards
CDS	Users Roles and Rights are stored in CDS and this is used by EBC
CDS Premium	CDS is getting upgraded to CDS Premium. Authentication of Employee discount will be validated by CDS Premium
SA.Payment	It collects and processes the valid and invalid transactions sent by TP.Net
CLA	Common Landing Application collects the poslog transactions and stores in database.
ICM	IKEA Customer Management stores the customer information.- To be decommissioned in Apr 2021
SAPP	Sends the price information to PIP.

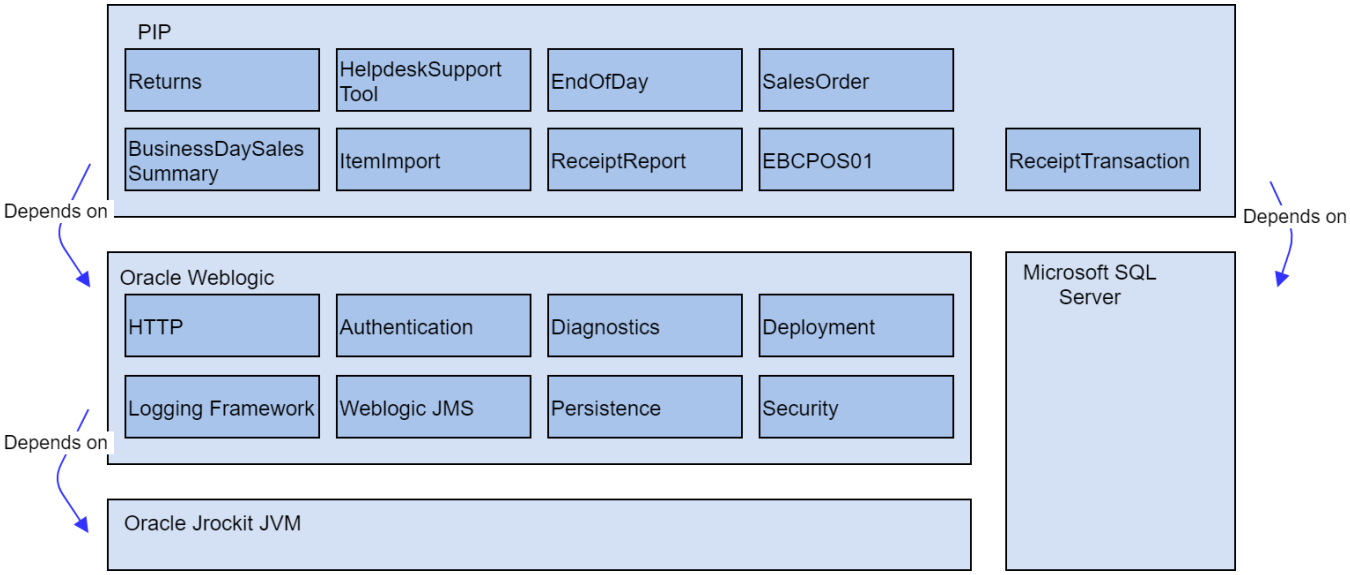
BCM	Known as Business Customer Master which stores the customer information. BCM Find API helps retrieving the customer information.
PCM	Known as Private Customer Master which stores the customer information. PCM Create/Update API helps creating/updating the customer information.

1.4.**Business Criticality**

2.**Functional**

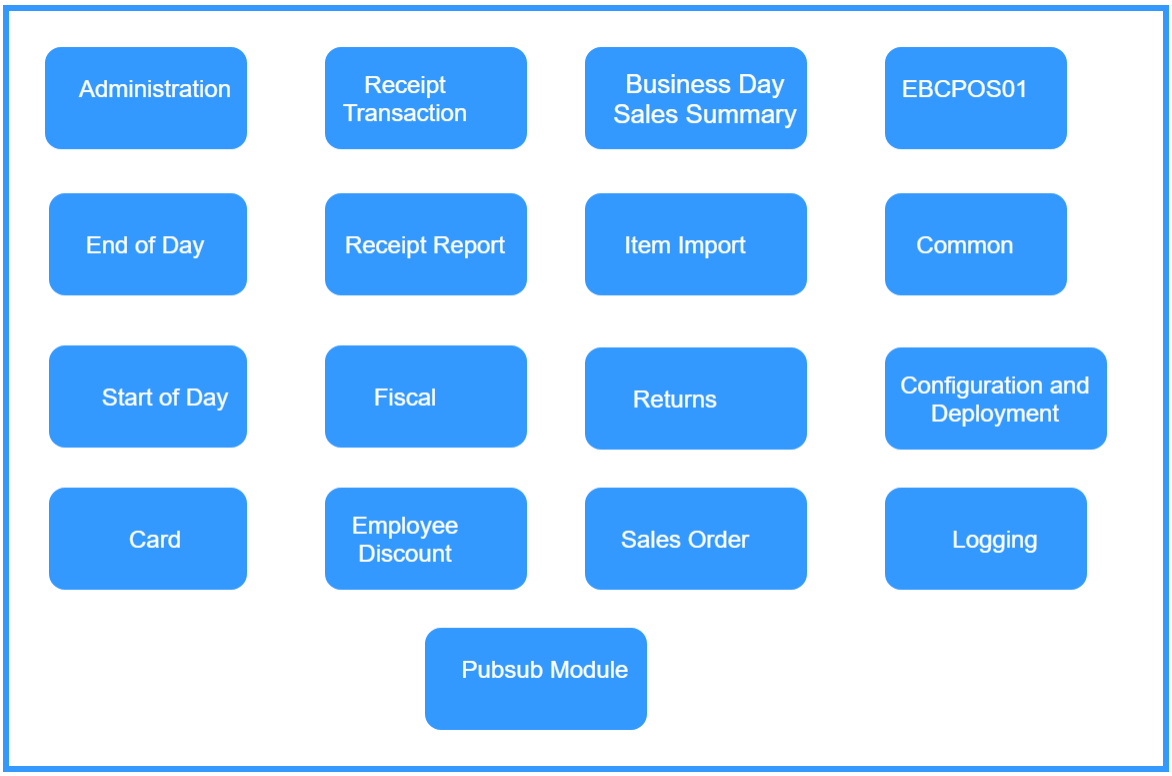
2.1. Functional Overview

The PIP is constituted of a number of applications and depends on a number of components and frameworks. The figure below gives a rough overview of the dependencies.



2.2.Functional Details

PIP has following major components.



Logical Component	Description
Start Of Day Signal	Enables the POS to signal to PIP that it has performed the Start Of Day operations.
Employee Discount	Provides the ability to check the existence and status of an employee in the CDS. Now CDS is getting upgraded to CDS Premium. Hence the ability to check the existence and status of an employee will be done through CDS Premium.
End Of Day Signal	Enables the POS to signal to other systems in the store that it has performed the End Of Day operations.
Returns	Enables the POS to retrieve information about returns from backend systems.
Sales Order	Enables the POS to retrieve information about sales orders from backend systems.
Receipt Reports	Creates reports from the batched receipt information and provides the result to backend systems. It also transmits the batched receipt information to the central archiving system for long term storage.
Receipt Transaction	Transmits the receipt information for each transaction to the backend system MHS It also transmits the food transactions to SPOSI. It also helps to send data to CLA DB. It also helps to send tender information to MHS. Receipt Line by Line - Transmits the receipt information for each article scanned to backend systems. It also helps sending valid and invalid transactions to SA.Payment system.
Cards	Enables to POS to check if the customers IKEA card is valid for the transaction.
Logging	Enables logging in a unified way for traceability and auditing.
Configuration and Deployment	Enables a unified, scriptable way of configuring and deploying the various PIP components.
Business Day Sales Summary	Creates the input for the SAREC by merging sales data from the two POS in the store.
PubSub Module	Misha has published the message using GCP topic. And PIP will be received message using GCP Subscription. And Stored in PIP database.
Item Import	Transmits the article information to the POS.
Common Utilities	Provides common utilities that are used by other PIP components.
Administration	Provides monitoring of the PIP.

PIP Fiscal 1. Vertex_RTE 1. Report generation 2. Invoice 3. Customer Search, Create and Update	Component in PIP that transfer RTE XML file from Vertex. Not in use currently as it is handled by TRC import. FTP to JMS topic (to which POS is listening). Component in PIP which generates different reports like MHS reports, Belgium, Slovakia, Hungary and Serbia fiscal reports. Receipt List MHS report is no longer used. Component in PIP that creates invoice request to third party system for Italy, Spain, Croatia. Component in PIP that helps to create, update and find customers in ICM. Create, Update and Find is only applicable for India and Slovenia. Remaining countries uses Find only. **ICM is going to be decommissioned in Apr 2021.** PIP is migrating to BCM and PCM instead of ICM. BCM/PCM are GCP services which are exposing APIs. BCM find API helps to search customer. PCM create/update API helps to create/update customer.

Please refer to the below link to find the detailed information about PIP:

IPOS BD documents update: https://iweof.sharepoint.com/teams/o365g_pipdocumentation_itsehbg/Shared%20Documents/Forms/AllItems.aspx?originalPath=aHR0cHM6Ly9pd2VvZi5zaGFyZXBvaW50LmNvbS86ZjovdC9vMzY1Z19waXBkb2N1bWVudGF0aW9uX2l0c2VoYmcvRWdqdnBzcHhMNnhGdnowWlRvOUxrY1VCcmlUa1QtQm5mc1Z5MWZkV09WR0lKdz9ydGltZT1pdHZyZjNJaDJFZw&id=%2Fteams%2Fo365g%5Fpipdocumentation%5Fitsehbg%2FShared%20Documents%2FBD&viewid=f7bf59e3%2D16d3%2D4fd4%2D96a7%2D7c34fe2882be

IPOS SLF documents update:
https://iweof.sharepoint.com/teams/o365g_pipdocumentation_itsehbg/Shared%20Documents/Forms/AllItems.aspx?originalPath=aHR0cHM6Ly9pd2VvZi5zaGFyZXBvaW50LmNvbS86ZjovdC9vMzY1Z19waXBkb2N1bWVudGF0aW9uX2l0c2VoYmcvRWdqdnBzcHhMNnhGdnowWlRvOUxrY1VCcmlUa1QtQm5mc1Z5MWZkV09WR0lKdz9ydGltZT1pdHZyZjNJaDJFZw&viewid=f7bf59e3%2D16d3%2D4fd4%2D96a7%2D7c34fe2882be&id=%2Fteams%2Fo365g%5Fpipdocumentation%5Fitsehbg%2FShared%20Documents%2FOperations%28SLF%29

PIP Helpdesk support tool wiki page:
https://wiki.ikea.com/webwiki/tiki-index.php?page=iPOS%20Helpdesk%20Support%20Tool%20-%20PIP

IPOS documentation wiki page https://wiki.ikea.com/webwiki/tiki-index.php?page=iPOS+Documentation

IPOS CC, CV, OPAM , EAB notes are placed here:
https://iweof.sharepoint.com/teams/o365g_solutionteampayments_irssemal/Shared%20Documents/Forms/AllItems.aspx?id=%2Fteams%2Fo365g%5Fsolutionteampayments%5Firssemal%2FShared%20Documents%2FiPOS%2F02%20%2D%20Operational%20Architecture

The excel CC is being to migrated to Algosec location: fo.ikea.com
Once the migration is complete, the excel sheets will be decommissioned.

Process to be followed:
The documents in the above location can be edited and uploaded for revisions by PIP team and send to Solution architect for further approving and publishing.
Solution Architect will publish the IPOS CC, CV, OPAM, EAB notes.

Below is the Published OpAM location:
\\ITSEELM-NT0001.ikea.com\Projects_A\IT-Op Arch Models\Published OpAMs\iPOS (POS Upgrade)

PIP Monitoring template can be found here:
https://iweof.sharepoint.com/teams/o365g_solutionteampayments_irssemal/Shared%20Documents/Forms/AllItems.aspx?id=%2Fteams%2Fo365g%5Fsolutionteampayments%5Firssemal%2FShared%20Documents%2FiPOS%2F03%20%2D%20System%20Overview%2F05%20%2D%20Configuration%2F01%20%2D%20Monitoring&viewid=286d5268%2D836a%2D4544%2D857a%2Dbf0c20905f24

PIP system overview documents are available here:
https://iweof.sharepoint.com/teams/o365g_solutionteampayments_irssemal/Shared%20Documents/Forms/AllItems.aspx?id=%2Fteams%2Fo365g%5Fsolutionteampayments%5Firssemal%2FShared%20Documents%2FiPOS%2F03%20%2D%20System%20Overview&viewid=286d5268%2D836a%2D4544%2D857a%2Dbf0c20905f24

For China JMS and Samba share lockdown functionality, please refer the below link for the solution
The code implementation of China JMS lockdown has been enhanced with generic implementation in PIP-R14.1.
The detailed information can be found below:
https://iweof.sharepoint.com/teams/o365g_solutionteampayments_irssemal/Shared%20Documents/Forms/AllItems.aspx?id=%2Fteams%2Fo365g%5Fsolutionteampayments%5Firssemal%2FShared%20Documents%2FiPOS%2F40%20%2D%20Releases%2FPIP%2DR14%2E0&viewid=286d5268%2D836a%2D4544%2D857a%2Dbf0c20905f24

Currently Cyber security has also been implemented for country Russia.
Also, Samba share lockdown functionality has been enabled globally.

Please refer the below link to know PIP-ICM integration.
The code implementation of PIP-ICM integration can be found in PIP-R15.0

The detailed information can be found below:
https://iweof.sharepoint.com/teams/o365g_solutionteampayments_irssemal/Shared%20Documents/Forms/AllItems.aspx?id=%2Fteams%2Fo365g%5Fsolutionteampayments%5Firssemal%2FShared%20Documents%2FiPOS%2F40%20%2D%20Releases%2FPIP%2DR15%2E0&viewid=286d5268%2D836a%2D4544%2D857a%2Dbf0c20905f24

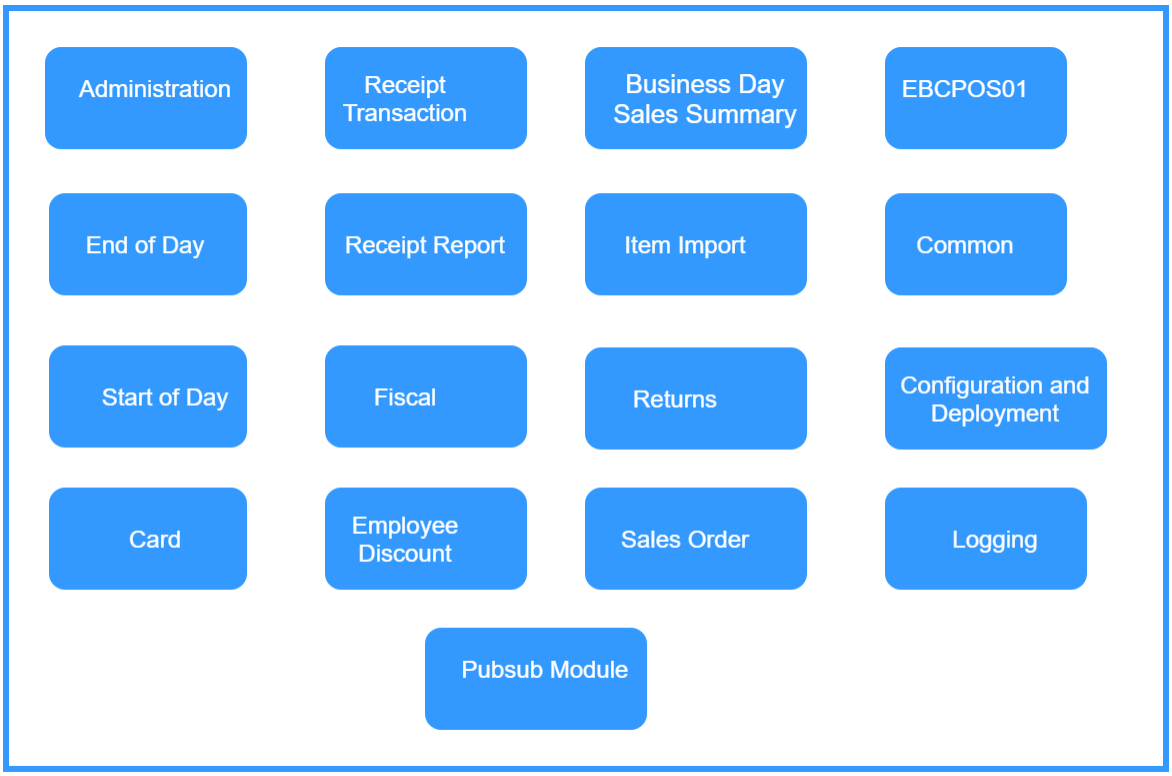
SLA document for PIP can be found below:
https://inside.ingka.com/supportservices/ApplicationsTools1/IL/Documents/gl_ipos_sla_FY20.pdf

Release 19.6 documents can be referred from Confluence below link:
https://confluence.build.ingka.ikea.com/pages/viewpage.action?pageId=279595929
3.**TECHNOLOGY RELATED**

3.1. Technical Overview

The goal of the PIP is to provide the POS with a unified interface to the different IKEA systems and thus requiring less IKEA specific customization of the third party product.
This is done by implementing multiple integration flows that perform quite unrelated business functions and may or may not integrate with different IKEA systems in different ways. Presently there are around ten such integrations but there is every possibility that this number will increase in the future.

The PIP is not developed and deployed as a single application. Instead the solution is modularized and split into multiple completely independent applications that can be individually installed on the application server and run in the same JVM.

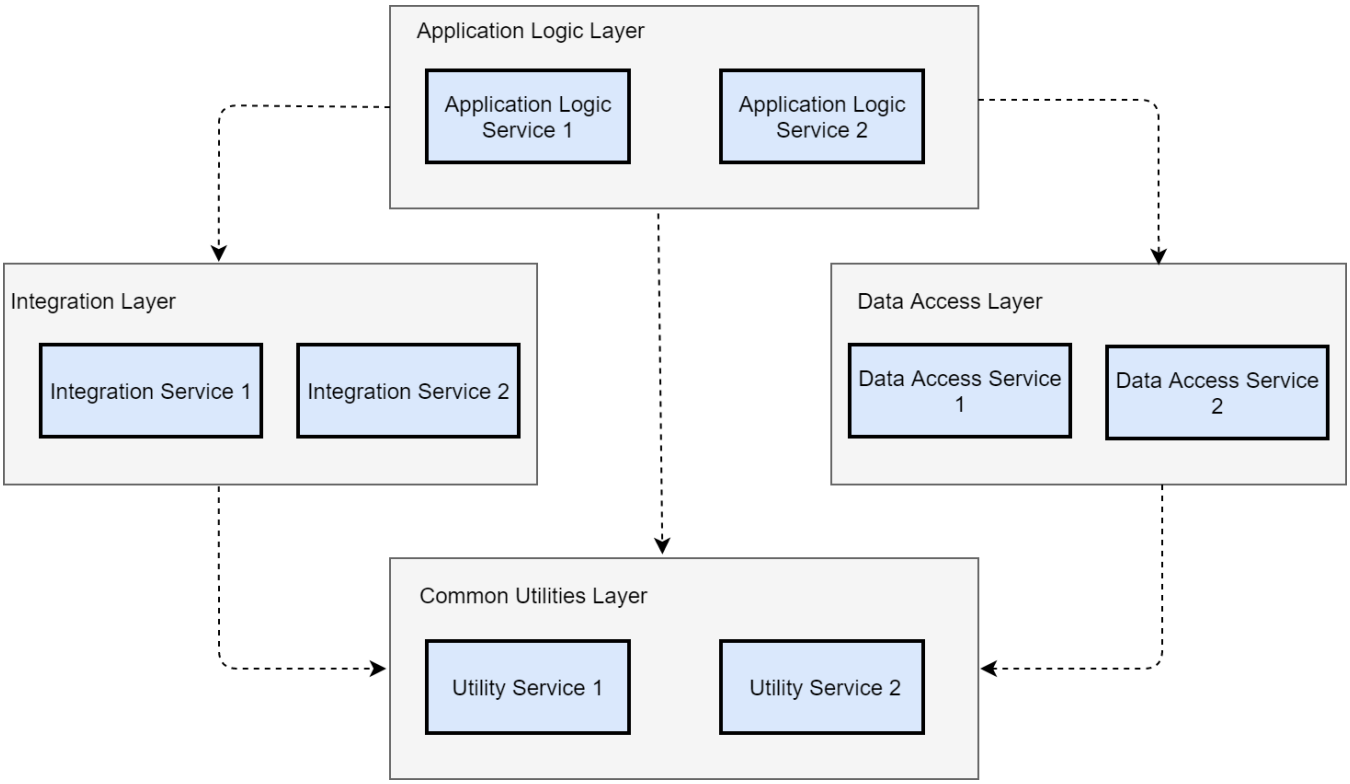


Weblogic Application Server

3.2 Application Architecture

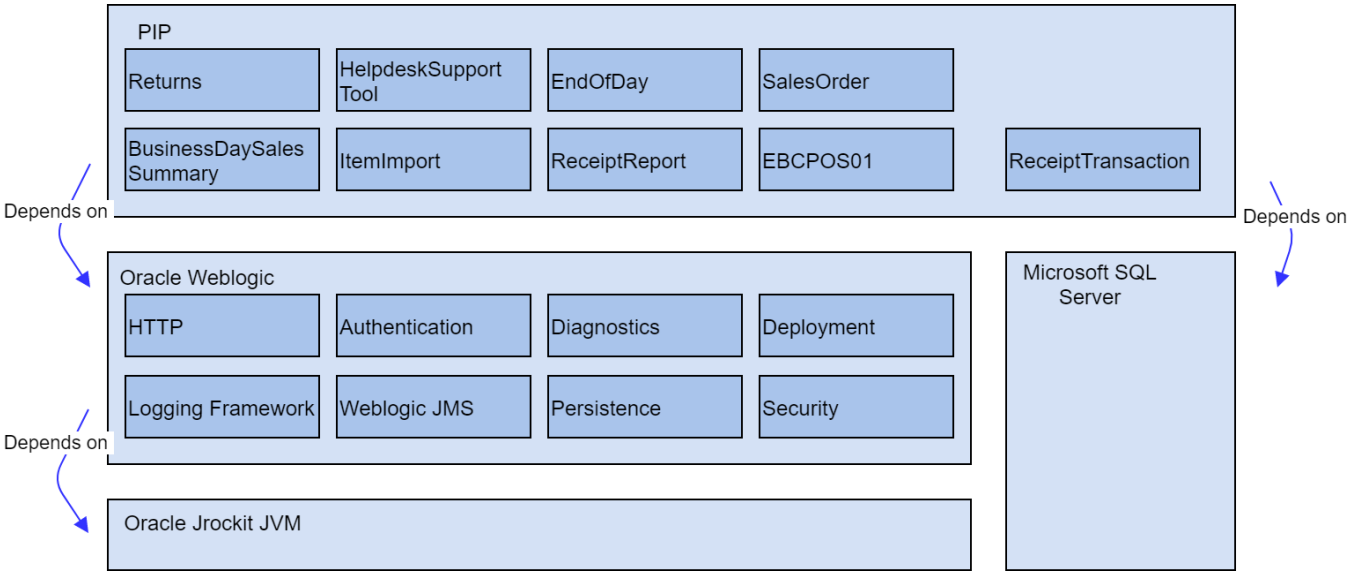
The PIP consists of multiple independent applications but they are structured in a similar way. Four types of layers exist:

- Application Logic Layer. This is where the logic for the various integration flows is located. Services in this layer make use of services in other layers to orchestrate the application logic.
- Integration Layer. This is where services that encapsulate how the communication with other systems is implemented.
- Data Access Layer. This is where services that encapsulate data access are located.
- Common Utilities Layer. Here we have services that provide common logic that can be used in other layers.



Please refer the below links in Sharepoint to find more system design view of PIP:
https://iweof.sharepoint.com/teams/o365g_solutionteampayments_irssemal/Shared%20Documents/Forms/AllItems.aspx?id=%2Fteams%2Fo365g%5Fsolutionteampayments%5Firssemal%2FShared%20Documents%2FiPOS%2F03%20%2D%20System%20Overview&viewid=286d5268%2D836a%2D4544%2D857a%2Dbf0c20905f24

3.3 Application Platform



3.3.1 Weblogic application server
The Weblogic application server is a standardized JEE application server, implying that any JEE applications can be deployed onto it, and implying that the PIP applications, which are JEE application, can be deployed on any other JEE compliant application server. The application server acts container for the PIP, and the PIP cannot execute without it. A certain level of knowledge about the what a server is and what is means for the PIP is therefore recommended. The application server provides a number of services and mechanisms which the PIP utilizes, and below is a description of a few of these.

3.3.2 The Weblogic admin application
The application server comes with an administration console, which is a web application running on that same server. The console provides a means of both configuring the application server and the resources and mechanisms provided, and it provides a means of managing running objects deployed onto it, of which the PIP is an example. This means that the administrator of the administration console is a very powerful user, having the authority to create, edit and delete configuration, deploy, undeploy, start and stop applications, and also to stop, start and delete the server itself.
Everything that can be done in the administration console can also be done programmatically towards the server, and there is also a python scripting framework which facilitates execution of a large number of operations on the server. As a note, it should be mentioned that both the administration console application and the jython framework actually utilize the programming API for managing the server in the end.

3.3.3 HTTP
Weblogic serves all HTTP and HTTPS traffic and is responsible for interpreting requests and relaying necessary information to the application(s). No dedicated HTTP server is therefore needed, even if it would be possible to redefine the topology so as to split the responsibility for the servicing of the applications, and the servicing of the HTTP requests.

3.3.4 JMS
JMS is in itself a JEE standard, providing a messaging mechanism. Depending on the need of a certain integration, the messaging service can be configured an various ways, e.g. should the topic/queue (the endpoint to which messages are pusblished and to which consumers listen) be durable, should it save the messages to a persistent store, should the messages be logged, etc.
PIP provides 33 topics or queues as per version Release 12.1.2, of which some are exposed to TP.Net and some are used internally by the PIP applications. An example is the pip.receipts.POSLogTransaction to which TP.Net places one message per receipt transaction, and which is configured as durable, implying that the PIP is guaranteed to pick it up, even if the ReceiptTransaction application should be down by the time the message is placed. Another example if the pip.range.ItemMaintenance, to which PIP published messages and TP.Net listens, also in a durable fashion. A third example is the v1.1 pip.receipts.POSLogLineByLine, to which the cash registers publish messages, but which is neither durable nor persistent, implying that if a consumer is not listening when the message is placed on the topic, it will never get a hold of that certain message.
Even if JMS itself is a standardized specification, the Weblogic JMS API is proprietary, which means that all JMS clients wanting to communicate with the PIP need to implement the Weblogic JMS client API.
In the PIP, some messages are published as xml strings and some as encoded messages.

3.3.5 Logging
The application server logs a configurable amount of information itself, and it also provides means for applications running on it to log to files defined in the application server. In the case of the PIP, the applications by-pass this mechanism and instead utilize the Log4J framework. See the troubleshooting guide [TROGUI] for more information.

3.3.6 Authentication and authorization
Another mechanism provided by the application server is authentication. In the server configuration, the authentication mechanism can be configured to use any arbitrary user repository, and it can also be configured to define users within the application server with corresponding authorization levels.
For the PIP, an authenticator (the pip.LdapAuthenticator) is configured towards the CDS, which is used by the administration console application for the authenticating console users.
Another example is the Helpdesk Support Tool, which utilizes the application server authenticator to authenticate support staff towards the CDS.
CDS Premium:

CDS is being upgraded to CDS Premium. For the PIP, an authenticator (the pip.LdapAuthenticator) is configured towards the CDS Premium.

It uses LDAPS Protocol. Port used is 636.

More information can be referred in the below location:
<ac:structured-macro ac:name="anchor" ac:schema-version="1" ac:macro-id="cf4b6d3d-918a-4aad-9079-36a336541882"><ac:parameter ac:name="" >_Toc313020040</ac:parameter></ac:structured-macro>https://iweof.sharepoint.com/teams/o365g_solutionteampayments_irssemal/Shared%20Documents/Forms/AllItems.aspx?id=%2Fteams%2Fo365g%5Fsolutionteampayments%5Firssemal%2FShared%20Documents%2FiPOS%2F03%20%2D%20System%20Overview%2F02%20%2D%20Systems%20and%20business%20services%2F04%20%2D%20Employee%20Discount%20Verification&viewid=286d5268%2D836a%2D4544%2D857a%2Dbf0c20905f24
and
https://iweof.sharepoint.com/teams/o365g_solutionteampayments_irssemal/Shared%20Documents/Forms/AllItems.aspx?id=%2Fteams%2Fo365g%5Fsolutionteampayments%5Firssemal%2FShared%20Documents%2FiPOS%2F40%20%2D%20Releases%2FPIP%2DR13%2E0&viewid=286d5268%2D836a%2D4544%2D857a%2Dbf0c20905f24

3.3.7 Deployment
The deployment mechanism in the application server is what enables an application to be registered in the server. For the PIP, the scripts used for deploying the applications are contained within the ICC4 module.
As an option to deploying the PIP through ICC4/IDEM, either of the applications can always be deployed independently of the others. This application can then be started without affecting the other applications already deployed and running. This feature is useful in a scenario where an urgent patch of one of the applications is required during business hours. If the entire PIP is deployed as an ICC4 module, this implies a service outage of all the applications and services, whereas if a single application is deployed, only this application is affected.

3.3.7 Persistence
In the application server, you're able to define persistent stores for utilizing the file system for persistence, and you're able to define database connections. The PIP utilizes both these mechanisms. A persistent store is configured for the durable topics, so that when the application server is stopped or crashes, the messages will still be available on the topics when the server comes back up again. A database connection is also created and configured, which is used by several of the PIP applications.

3.4 Source Code Control

CLM is the source code control for PIP.
URL:
https://clm.rat.itshost.se/ccm/auth/authrequired#action=com.ibm.team.workitem.viewWorkItem&id=249602
Connection to CLM can be found as below:

Now the source code control is Git for both PIP and EBCPOSRIX.
https://git.build.ingka.ikea.com/SolutionTeamPaymentsRS/PIP.git
https://git.build.ingka.ikea.com/SolutionTeamPaymentsRS/EBCPOSRIX.git

We are using Jira where the new Change Requests are placed.
Below is the Jira path for PIP.
https://jira.digital.ingka.com

3.5 Production, UAT and Development Locations

3.6 Hardware and Software Pre-requisites

Part of infrastructure team
GDBA, GWLS, GLINUS, Gnetwork.

3.7 Interfaces to other systems

Item import

The POS system needs to be loaded with articles before it can be operational. An article can be either be updated or deleted in the POS. Detailed information can be found in the solution design[SODITM]. This integration flow can have three different payload scenarios:
Full load containing all articles (non-destructive). Locally modified attributes should be excluded (not overwritten).
Add or update item (AddUpdate in Item Maintenance) add or update one or many items to the POS. If an item doesn't exists prior to being updated it will be added automatically.
Delete item delete one or many items in the POS
The items are stored in the MHS system. MHS is also the system that initiates the integration flow. The information is then pushed via PIP (POS Integration Point) to POS-Server. In the PIP the payload is converted to ARTS standard Item Maintenance format and put on a JMS topic. Article information that is not available from MHS is added (merged) with data from the central RIX system.

Protocol	Format
JMS	ARTS
LBB	CEM
EBB	CEM

Returns
Returns are initiated via the SAMS application, which creates a unique return identity. The actual return or negative sales transaction (receipt data) is handled via the normal receipt flow.
A return is initially created by a cashier using the SAMS interface GUI. The return is persisted in MHS and the cashier is then presented a unique return id. The cashier then enters this id in the POS register, upon which a request is sent to MHS via PIP that contains the return identity. The MHS system responds with all the items that are connected to the return. Return data from MHS only contains the items that shall be returned, i.e. the return does not contain the complete receipt.
Three flows related to returns are relevant for the PIP:
Get and lock return Once the cashier has retrieved a unique return id through the SAMS GUI, this is entered in the POS register, which calls the PIP BsGetAndLockReturn service, which in turn calls the BsGetAndLockReturn service.
To access BsGetAndLockReturn service EBCJMSPOS01 was called.
After PIP-R14.1 implementation, PIP will no longer call BsGetAndLockReturn service EBCJMSPOS01.
Instead PIP will call BsGetAndLockReturn service EBCJMSBOS01. Please refer the below link for more information:
https://iweof.sharepoint.com/teams/o365g_solutionteampayments_irssemal/Shared%20Documents/Forms/AllItems.aspx?id=%2Fteams%2Fo365g%5Fsolutionteampayments%5Firssemal%2FShared%20Documents%2FiPOS%2F40%20%2D%20Releases%2FPIP%2DR14%2E0&viewid=286d5268%2D836a%2D4544%2D857a%2Dbf0c20905f24

Save return When the cashier has locked the return, he or she will, in the normal case, save it in MHS via the TP.net Backstore and the PIP. This flow utilizes the Receipt Transaction interface. See section **Error! Reference source not found.**.
Unlock return - In the scenario where the cashier gets the return but for some reason decides not to actually save and complete the return, the cashier has the option to actively cancel it. In this case, the TP.net Backstore calls the BsUnlockReturn service in PIP, which in turn calls the BsUnlockReturn service in MHS.
User authentication
User authentication concept for POS differs pending on which domain the user wants to access. For enterprise access the user will be authenticated against CDS. Users that want to access the POS within the store will only be authenticated internally against the POS server. The main types are presented below.
POS Enterprise applications (TPAdmin) authentication against CDS
POS store applications (TP.net Backstore and Frontstore) only internal authentication
PIP user authentication against CDS

B2B Business card authorization
The Business Customer Card Authorization (IKEA Business) verifies that the customers' contract number is valid as well as the current available credit balance. This goes for the Private Customer Card Authorization (IKEA Handla) as well. The difference is that another source system is used for retrieving the authorization, e.g. Ikano instead of RIMS. Business to Business Credits is handled via the RIMS application. A quote from RIMS user guide, "RIMS stands for Receivables & Invoice Management System. RIMS is responsible for the creation and maintenance of Business to Business accounts, and the production of invoices."
Private labeled store card authorization
The private customer card authorization verifies that the customers' contract number is valid as well as current available credit balance (IKEA Handla.
As per MHS Transformation Program, PIP will directly call RIMS and IKANO for Authorizing card without sending request to MHS.
Also, RIMS system needs a report called Sales completion Report which will contain all the transactions which has been performed by RIMS card.
This report will be generated by PIP during End of Day.
PIP communicates to RIMS using ITF WSProxy. PIP communicates to the IIP Host using Secured SSL connection.

Protocol	Format
WS	PIP (CEM)

The integrations which iPOS today has with RIMS will be changed with BOT
i.e. Change credit check call from BSAuthorizePurchase to new API on API
Management platform. When PIP receives BSAuthorizePurchase request from
Tp.net then PIP will fetch the token code from Auth0 system and will append
the token to the request and will send to API connect Gateway system for
The card validation. The mode of communication is http and data exchange
is in JSON Format.

Sales orders
In order for customers to pay for sales orders previously created in iSell or MHS, the POS need to retrieve sales order data. A sales order can have many different forms, such as Cash & Carry order, Distribution order, Draft order or Quotation
As per MHS Transformation Program (Remove 87), PIP will call ISELL directly for getting the order details from ISELL if the orders are not present in MHS.
PIP will also ensure that PIP will send all order payment information to ISELL without fail.
PIP will try to resend the Payment information in case of delivery failure until ISELL EBC is reachable again.
PIP communicates to ISELL using ITF WSProxy.

Fiscal
Fiscal requirements tend to be country-specific. The purpose of this component is to keep country-specific Fiscal customizations in one place.

Invoice: The Invoice solution is generic and can be used for several countries.
Two operations are available: CreateInvoice and GetCustomer. It will always be possible to call the two operations, but a simple "Not implemented" message will be returned in all countries other than the ones where they are used. GetCustomer is expected to be used when a country needs to store /update/pick up customer data via the PIP, whilst CreateInvoice to create the invoice and communicate back what is to be printed on the receipt.
Invoices are delivered to the invoice system either in online mode (synchronous call just before receipt transaction is completed) or in offline mode (PIP delivers offline invoice to invoice system based on completed receipt transaction in POSLog).

Fiscal report:
Due to legal requirement few countries need to generate Fiscal report containing the transaction information for the day. It helps to provide fiscal report to store (Belgium, Slovakia, Hungary, Serbia)

RTE import
Provide tax rates and rules to POS (US only) No longer used

ICM Customer information: Find, Create and Update

PIP fiscal component helps to Find, Create and Update customer information from ICM (IKEA Customer Management).

TP.Net sends SOAP WS call like FindCustomer, CreateCustomer and UpdateCustomer to PIP. PIP will transform the SOAP message to JSON format and create REST WS call to ICM API gateway. ICM replies with the information to PIP. PIP transforms the reply to SOAP message and sends the information to TP.Net.
Currently PIP-ICM integration (FindCustomer interface) has to be enabled globally. CreateCustomer and Update Customer interfaces has been enabled for countries India and Slovenia.

India and Slovenia will have create, update functionalities enabled. PIP has added feature in FindCustomer API to send **full** response for countries India and Slovenia based on parameter set in configuration file.
For remaining countries PIP has added feature in FindCustomer API to send **limited** response based on parameter set in configuration file.
PIP helpdesk tool will display only those ICM URLs for which functionalities are enabled in PIP.

ICM will be decommissioned during Apr 2021, hence **PIP will be migrating to GCP services BCM and PCM.**

PIP fiscal component helps to find customer information using BCM (Business Customer Master) and helps to create and update customer information using PCM (Private Customer Master).

TP.Net sends SOAP WS call like FindCustomer, CreateCustomer and UpdateCustomer to PIP.

The BCM and PCM API's will be authenticated using Google IAM system.

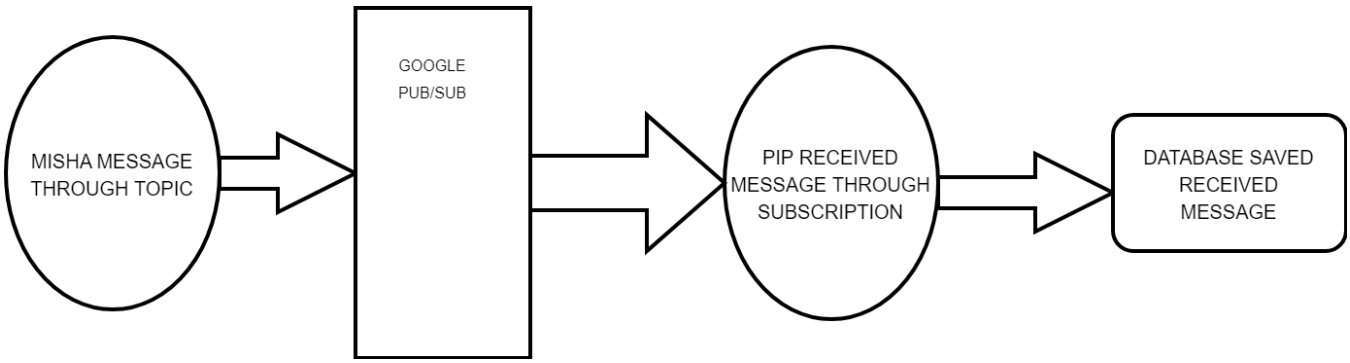
Whenever PIP receives WS call from TP.Net, it will first check if access token exist. If valid access token is existing, PIP will reuse the token and will send API call to BCM/PCM.

If access token is not valid, PIP will request Token system (IAM) for new token. Using this new token PIP will send API call to BCM/PCM.

PIP will transform the SOAP message to JSON format and create REST WS call to BCM API gateway (for find WS) and PCM API gateway (for create /update WS) . BCM/PCM replies with the information to PIP. PIP transforms the reply to SOAP message and sends the information to TP.Net.

More details can be referred from the link:
https://confluence.build.ingka.ikea.com/pages/viewpage.action?pageId=249061197
https://confluence.build.ingka.ikea.com/pages/viewpage.action?pageId=249061122

PubSub Module:
Misha has published the message using Google cloud Pubsub Topic.PIP will be received the message using Google Cloud Pub sub subscription. As per format Misha has published the message in JSON fomat.PIP will be subscribe the message and store in database.



Pubsub Module Diagram

3.8 Documentation

Please refer the below links to get detailed information on PIP components:
https://iweof.sharepoint.com/teams/o365g_solutionteampayments_irssemal/Shared%20Documents/Forms/AllItems.aspx?id=%2Fteams%2Fo365g%5Fsolutionteampayments%5Firssemal%2FShared%20Documents%2FiPOS&viewid=286d5268%2D836a%2D4544%2D857a%2Dbf0c20905f24

3.9 Network Specific Issues

N/A

3.10 Third Party Tools

N/A

4. Technical Environment

4.1 Server Names

Production Servers:

TEST Environment Servers:

4.2**Web URLs**

4.2.1 PIP_Helpdesk_Tool
Refer the links to PIP_Helpdesk_Tool for the stores in production in the below file:

4.2.2 Weblogic console
Refer the links to Weblogic console for the stores in production in the below file:

4.3**On-Line Region**

4.4**Libraries**

pip nexus:
http://itseelm-nt4696.ikea.com:8081/

4.5.**Software Requirements**

Oracle Weblogic 12C,
Spring,
Maven,
SQL,
EBBRuntime_x64_2.8.EN,
JavaDevelopmentKit_x64_6.0.11.EN,
JavaRuntime_6.0.24.EN,
EBCDevelopmentEnvironment_1.1.EN
CLM
Notepad++
Jenkins
Nexus

4.6**Server/Application access requirements**

Same as Vdi credentials

4.7.**Login IDs on Production and Development**

Same as Vdi credentials.

4.8.**Test data and Test environment**

4.9.**Change Control Procedure**

4.10.**Forms**

4.11.**Reports**

The below attached file contains information about all the reports generated by PIP

4.12.**Tools**

IKEA Helpdesk Tool
Weblogic 12.1.3.0.0
TNSNames_OID_1.0.EN
SQLDeveloper_x64_2.1.EN
EBB_config_1.0.EN
BSTester_6.0.EN
EBCSimulator_1.2.EN
7zip_9.20_x64.EN
Putty_0.6.EN
WinSCP_418.EN
UltraEdit_17.30.EN
Notepad++
4.13.**Methodologies**

4.14.**Complexity**

4.15.**Unusual coding practice**

4.16.**Commonly used routines**

5.**Datasets & Databases**

5.1. Overview

Database Schema Name : PIP
Landing schema for IIP : PIP_TEMP
USER for IIP: U_TMR

5.2. Tables

PIP.BATCH;
PIP.CONNECTION_TEST;
PIP.ERROR;
PIP.GENERATOR_TABLE;
PIP.RETAIL_ITEM;
PIP.SALES_TAX;
PIP.VALUE_ADDED_TAX;
PIP.WL_LLR_WLS_LIP1
PIP.POSLOG
PIP.RECEIPT
PIP.TRANSACTION_LOG
PIP.RU_TMR_MISHA
PIP.PIP_RU_NTIN
PIP.RU_TMR_PACE

5.3. Stored Procedures :

NA

5.4. User-Ids

PROD database for all store level :- **r_pip**

5.5. Performance Requirements

NA

6.**Application Components**

6.1 Framework and component

EBB

The EBB framework is an infrastructural integration component, developed and maintained internally by Ikea. The framework provides a unified way of integrating between Ikea legacy systems. A part of the framework is the EBC, which is a component which exposes one or several business services to the Ikea network.
EBB supports two modes which are relevant for iPOS; local mode and global mode. Global mode implies communication with a centrally residing system, e.g. iSell, whereas local mode implies communication between towards a local system, e.g. MHS. As a result, PIP calls MHS in local mode and OutputManagement in global mode. Third party products are generally not expected to adopt the EBB framework, which is why TP.Net never utilizes it, but rather calls the PIP by the exposed web services and JMS topics.
The EBCPOS01 is the "PIP EBC", which as per version 1.0 exposes 3 interfaces; the BsUpdateRange used in the item import process, and the BsGetSalesTax and BsUpdateSalesTax, both used for tax administration.

Spring Core and Spring Integration

The PIP applications are heavily based on the Spring framework, specifically the Core and Integration APIs.

Log4J

The PIP applications utilize the Log4J logging framework. The log levels, formats and log file destinations are configured in files <*application*>-log4j.properties, found in the pip config directory on the Backstore server. If log4j configuration is changed, the application to which the file applies needs to be restarted.

JMS

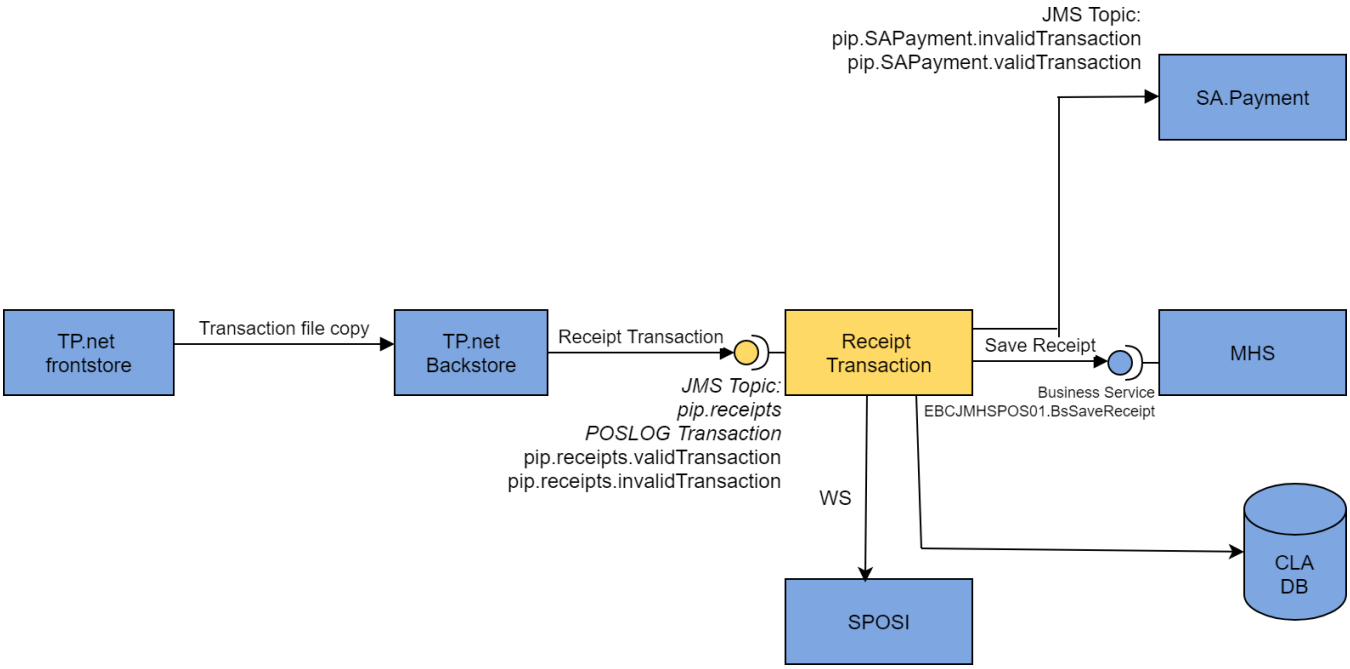
JMS stands for Java Messaging Service, and is a standardized messaging technology. In iPOS, we have chosen the Weblogic JMS implementation, since it is likely to operate well together with the Weblogic Application Server. A number of JMS topics and queues are configured in the Weblogic app server, on which applications place messages, and (other) applications read from. A topic or queue can be configured to persist messages, to guarantee delivery, to require authentication, etc., and the choices made for the iPOS entities are based on the requirements of the certain context in which it is used.
6.2 Applications
All PIP applications are packaged and deployed as EAR files, even though wrapped into ICC4 modules when deploying the PIP in its entirety. This means a single application can be deployed independently of the rest of the PIP.
The below is an introductory description of the PIP applications.

PIP Flow Diagram:
https://wiki.ikea.com/webwiki/tiki-download.php?attId=7911

ReceiptTransaction

The Receipt Transaction receives sales transactions from TPNet and passes them on to MHS. The main flow of events involved in a receipt transaction is:

1. The cash register copies an xml file containing items, tenders, cash register, and cashier information, onto a shared folder on the TPNet Backstore server
2. TPNet Backstore picks the file up from the shared folder, validates it, transforms it and passes the transformed xml on to the PIP JMS topic pip.receipts.POSLogTransaction in the POSLog format.
3. PIP validates the xml, transforms it, and passes it on to the MHS business service BsSaveReceipt.
4. PIP also inserts the receipt information to CLA database.
5. PIP also sends the Food transactions to SPOSI.
6. PIP also sends the valid and invalid transactions to SA.Payment team.



ReceiptReport

The Receipt report application is responsible for relaying the POSLog reports to a number of legacy archiving systems; IBIS, Escape and OutputManagement. The relaying process is triggered by the TP.Net Backstore application as part of the End of Day procedure. The main flow of events involved is:
As a part of CR 211186: PIP Development WW - Stop integration between PIP and IBIS, PIP has stopped sending ibis report.

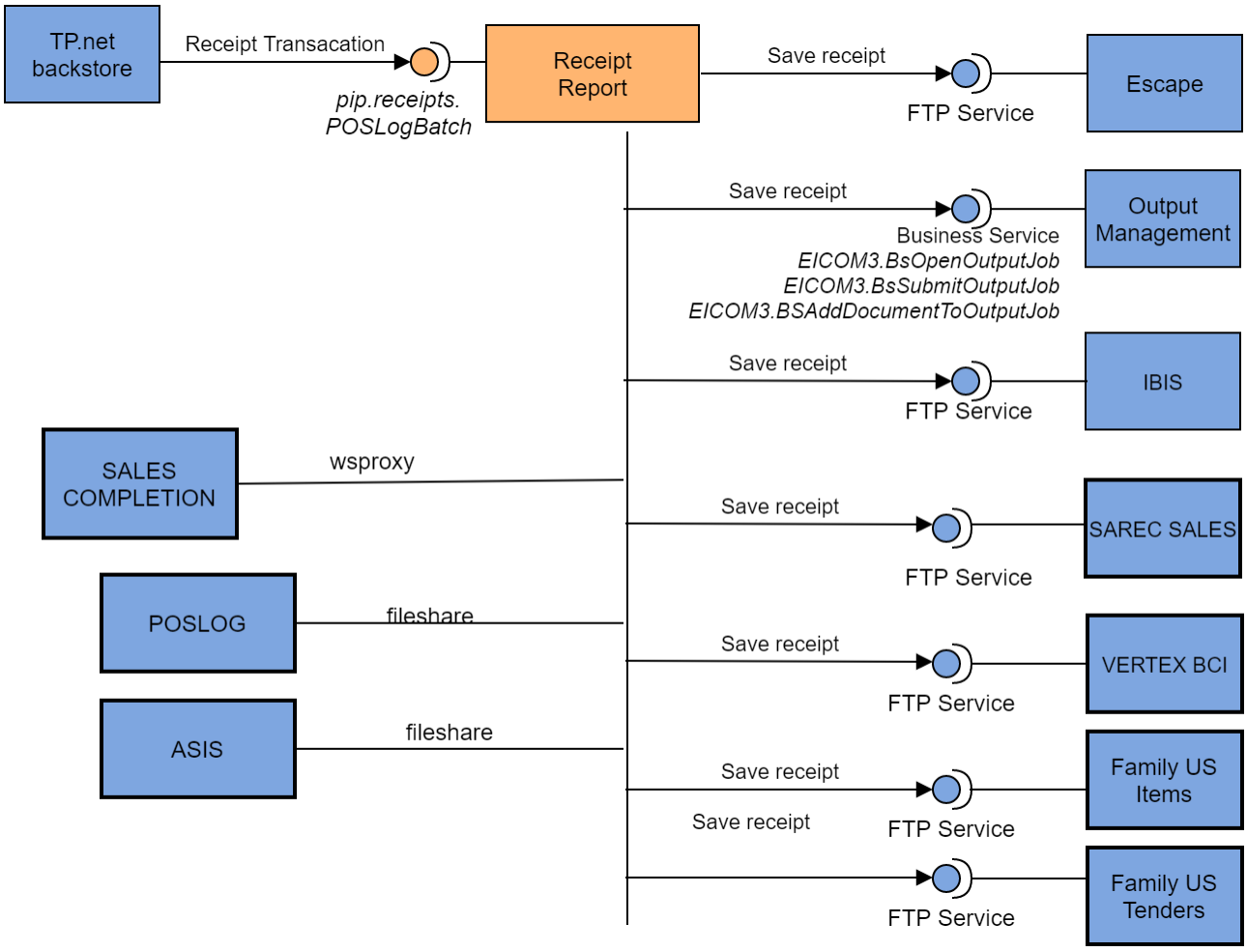
1. The TP.Net Backstore component sends POSLog receipt transaction batch messages to the JMS topic pip.receipts.POSLogBatch.
2. The application aggregates the payloads and splits them up into legacy system specific JMS topics.
3. The application then reads from the JMS topics and sends the reports to the corresponding system.

For Escape, Family US Items, Family US Tenders, Vertex BCI, SAREC SALES, the reports are transferred over FTP.

For OutputManagement, a number of business service calls are performed to the BsOpenOutputJob, BsSubmitOutputJob and BsAddDocumentToOutputJob in the EICOM3 EBC using ITF WSPROXY.

For Sales completion Report business service :RegisterSalesComp is called using WSPROXY where the SOAP message is delivered to RIMS application.

Also, the POSLOG batch and ASIS is generated and will be placed in the SERVER shared location



In RU country for TheoreticalSales Report,

1) New transaction type is added which is DocumentTransaction type having functionality similar to RetailTransaction with two different transaction link reasoncode

i) <TransactionLink ReasonCode="IKEA:FullServeHandOut"> - For IKEA:OrderType= 'SALESORDER'

ii)<TransactionLink ReasonCode="IKEA:PaymentOnCollectHandout"> - For IKEA:OrderType= 'DISTR'

More details can be found in the below link:
PAYM-2666(https://confluence.build.ingka.ikea.com/pages/viewpage.action?pageId=268808956)

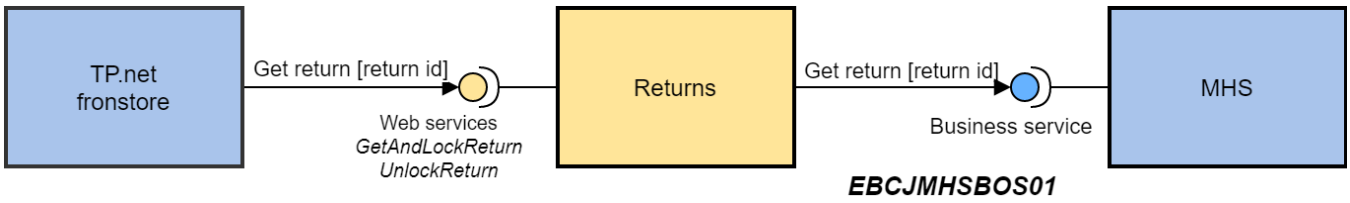
2) New element<IKEA:FullServe>Element is added in poslog and needs to be added in Sarec File for OrderType= 'SALESORDER' for transaction type retail and document.

More details can be found in the below link:
PAYM-2667(https://confluence.build.ingka.ikea.com/pages/viewpage.action?pageId=268806749)

Returns

The Return application is responsible for handling SAMS returns. This is not to be confused with voids, which are a type of return created instantly during the creation of a receipt transaction, i.e. while the customer is still in the cash lane. In the case of a void, the ReceiptTransaction is responsible. The flow of events is as follows:

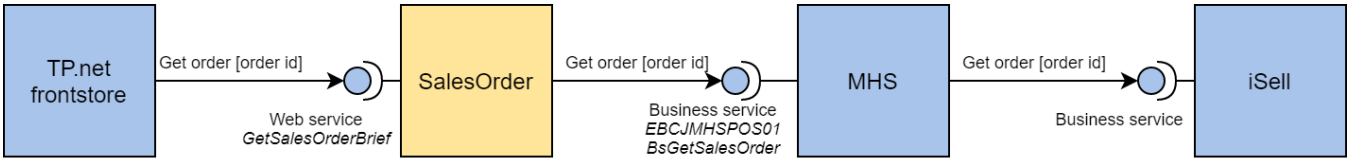
1. A return cashier creates a return in SAMS. SAMS saves the return in MHS and provides a Return ID as response.
2. The same cashier enters this Return ID in the POS application.
3. The POS application sends the ID to the PIP Return application.
4. PIP fetches the return in MHS by calling the BsGetAndLockReturn business service, providing the Return ID.
5. PIP responds with the return contents to the POS application, and the customer is returned tenders accordingly.



SalesOrder

The SalesOrder application is responsible for fetching already existing orders in iSell through MHS and provide them to the POS register. The order will then be paid using the receipt Receipt Transaction application. The flow of events:

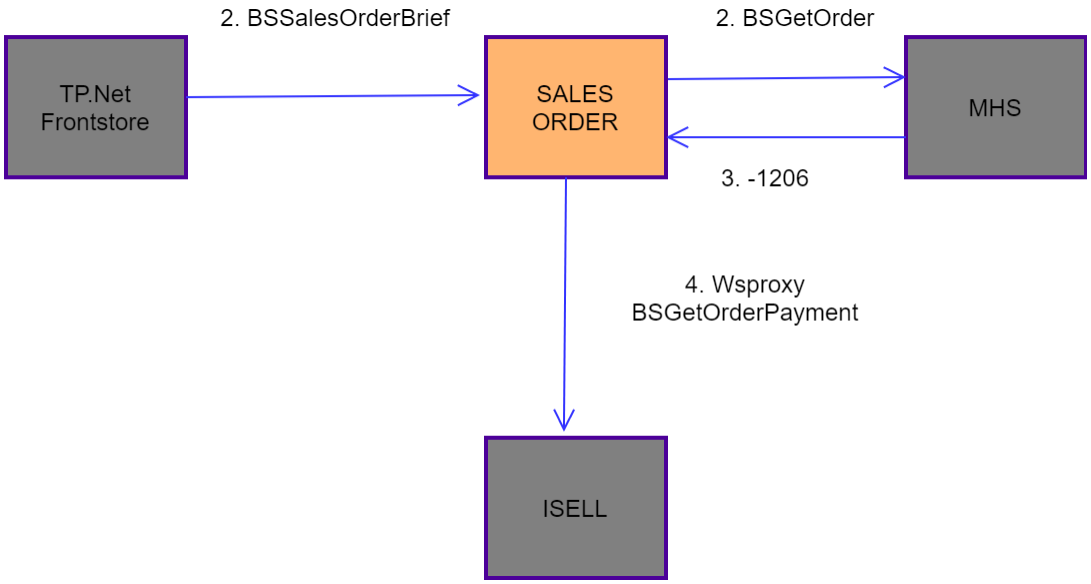
1. An order is created in iSell, typically by a sales person helping a customer assembling a number of articles for a kitchen.
2. Tp.Net frontstore fetches the already created order by calling the GetSalesOrderBrief web service in PIP, providing the iSell order number.
3. The PIP SalesOrder application fetches the order contents in MHS by calling the MHS business service BsGetSalesOrder.
4. MHS in its turn fetches the order in iSell and returns it to the SalesOrder application.
5. The SalesOrder application returns the contents of the order to the POS register.

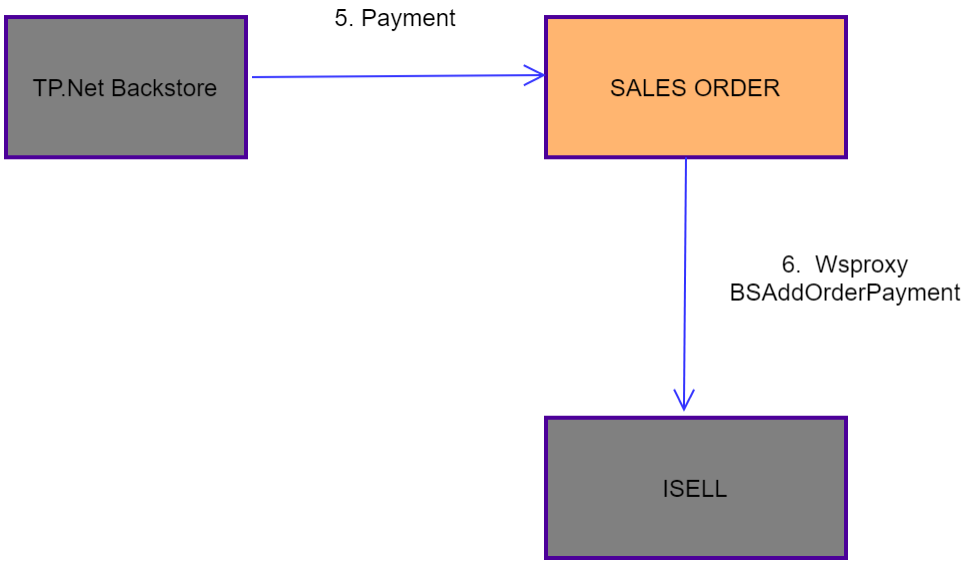


6. In case the order is not found in MHS, MHS will send error code -1206. PIP will call directly ISELL EBC using WSPROXY.

IsNotPrepaid from MHS
In RU, there is new tax slabs has been introduced where a article's price is differentiated based on whether the article is prepaid or not.
So as a part of CR PAYM2659, PIP is sending IsNotPrepaid flag to Tp.Net as a part of response while fetching order details from MHS
https://confluence.build.ingka.ikea.com/pages/viewpage.action?pageId=269817294

Remove 87 enable
As per MHS Transformation Program, PIP will send detailed order information to TP.Net.
Also, MHS will no more send the Payment information to ISELL. Instead PIP will send the Payment information to ISELL once Remove 87 is activated. PIP will try to resend the Payment information to ISELL unless ISELL acknowledge it.
TP.Net FrontstoreSALES ORDERISELL4. Wsproxy
BSGetOrderPaymentMHS2. BSGetOrder2. BSSalesOrderBrief3. -1206
TP.Net BackstoreSALES ORDERISELL6. Wsproxy
BSAddOrderPayment5. Payment





Payment on Collect
In Russia it is a legal requirement to offer customers the option of paying with Cash for online orders, therefore for the CFB launch POD for Collect at Store should be offered.
To enable this functionality the below stories has been ordered in PIP:
https://confluence.build.ingka.ikea.com/display/IPOSCCHPIP/PAYM-1294-RU+~~PIP RU+Payment+on+Collect+~~+iSELL+order+request+changes

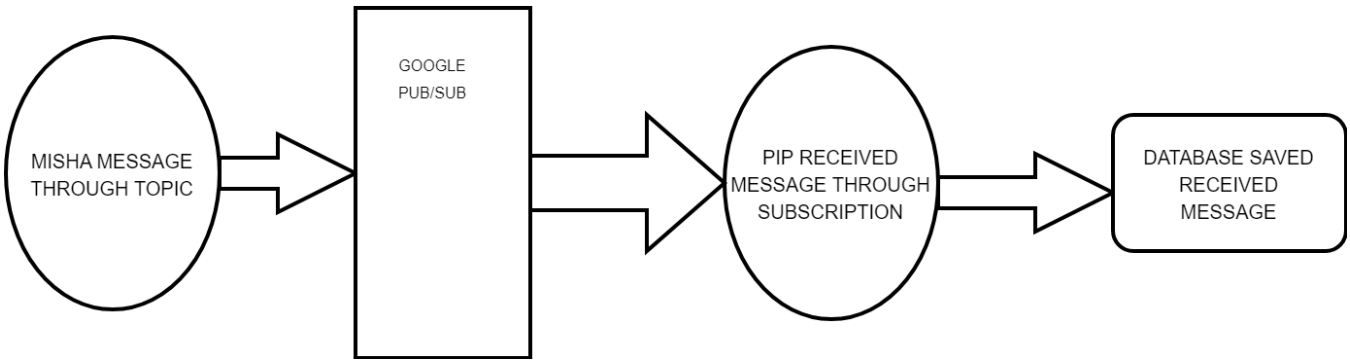
https://confluence.build.ingka.ikea.com/display/IPOSCCHPIP/PAYM-1568-RU+~~PIP RU+Payment+on+Collect+~~-----+iSell+Response+Changes

https://confluence.build.ingka.ikea.com/pages/viewpage.action?pageId=280340896

https://confluence.build.ingka.ikea.com/display/IPOSCCHPIP/PAYM-1452-PIP+Analysis+~~RU+Payment+on+Collect+~~+Changes+to+AddOrderPayment+request

PubSubModule:

Misha has published the message using Google cloud Pubsub Topic. PIP will be received the message using Google Cloud Pub sub subscription. As per format Misha has published the message in JSON fomat.PIP will be subscribe the message and store in database.



Pubsub Module Diagram
For more details on the solution, please refer: https://jira.digital.ingka.com/browse/PAYM-1305
6. ItemImport

The ItemImport is responsible for the loading of article data from MHS and RIX to the PIP and to TP.Net. An import is triggered by MHS. The flow of events:

1. MHS calls the PIP EBCPOS01 business service BsUpdateRange. The call contains the updated article information.
2. The EBCPOS01 application passes the data on to the ItemImport application via the pip.ebcpos.RetailItemLoadMHS JMS QUEUE.
3. Incase RIX flag is negative or RIX tariff code/Local number is not available for any article, then PIP will try for a fall-back scenario to EBCPOSRIX, where the PIP hits EBCPOSRIX and tries to fetch the details of that item. If the article is available in EBCPOSRIX, the same will be updated in PIP DB before updating MHS article details. Once details are updated, Source Flag Rix turns +ve.
4. The ItemImport application fetches complementing data from the RIXAdapter business service BsGetRetailItems and attaches it to the article data objects.
5. The ItemImport application splits the article data into chunks of max 1000 articles and places them in the pip.range.ItemMaintenance JMS topic.
6. While sending the item information to TP.Net in item maintenance file, PIP configures the item short name to different values. By default: 15 characters but for countries like Poland it is configured as 20 characters.
7. The TP.Net backstore picks the article data up from the topics and enters it in its database.

LOP Local Pricing
As part of MHS Transformation Program, Local Pricing is implemented where parts of article information will be sent by IIP to PIP instead of MHS.

SAPP will send Regular Price and Family Price to PIP instead of MHS.

Note: During IIP import incase RIX flag is negative or RIX tariff code/Local number is not available for any article, then PIP will try for a fall-back scenario to EBCPOSRIX, where the PIP hits EBCPOSRIX and tries to fetch the details of that item. If the article is available in EBCPOSRIX, the same will be updated in PIP DB before updating IIP article details. Once details are updated, Source Flag Rix turns +ve.

For country RUSSIA as per TMR Law, PIP has to send the TMR items to TP.Net.

Here, PACE and Misha systems will have TMR information available and will be sent to PIP.

PACE data will be sent to SAPP through a batch and SAPP will insert the data into PIP Postgre database.

Misha on the other hand will publish the information to Google PUB SUB topic and PIP will subscribe to this topic.

PIP will insert the data into PIP Postgre database whenever new data arrives in PUB/SUB.

PIP is supposed to self trigger item import daily using cron job and validate if any item has been qualified for TMR criteria.

If yes, these items will be self triggered as item import and will validate TP.Net criteria .

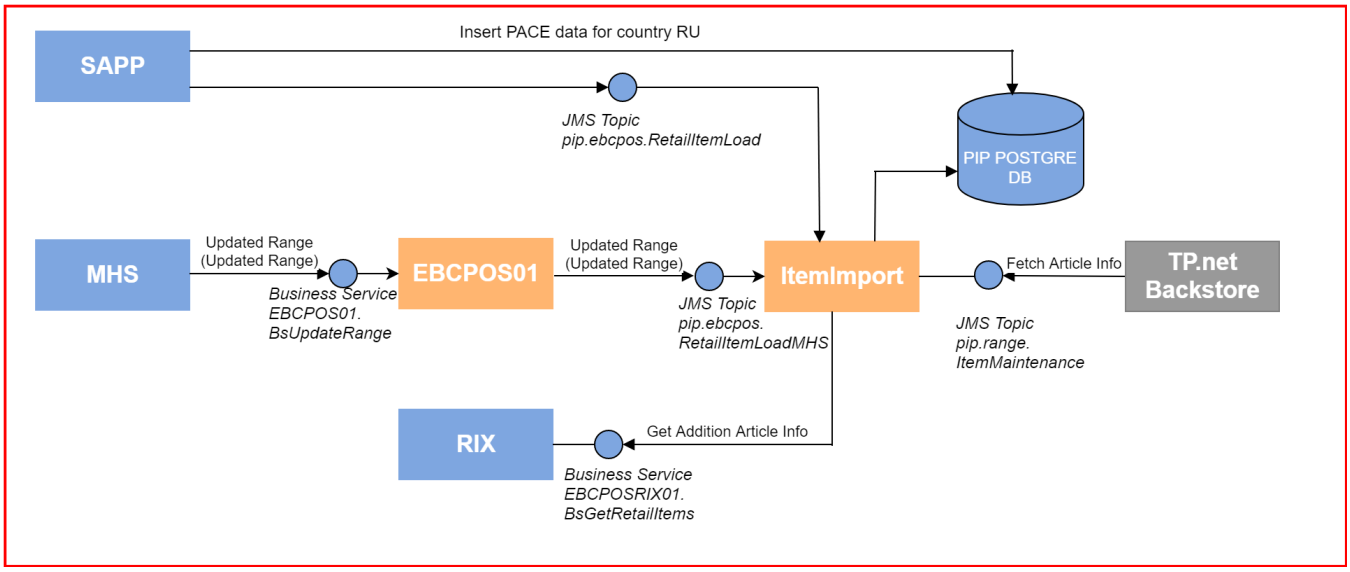
In case both TMR and TP.Net criteria is being met, PIP will send the TMR information to TP.Net.

During normal item import for country RUSSIA PIP will first validate TP.Net criteria and later TMR criteria. In case only TP.Net criteria is met, PIP will send article information to TP.Net without TMR data.

In case both TP.Net criteria and TMR criteria is met, PIP will send article information to TP.Net along with TMR data.

For this implementation, PIP is creating new user "u_tmr" which will be shared with IIP team so that they can insert PACE records for RUSSIA TMR.

PIP is also granting necessary access to the use so that it will only be able to access table "RU_TMR_PACE"
For more details: please refer
https://jira.digital.ingka.com/browse/PAYM-1157
https://jira.digital.ingka.com/browse/PAYM-1305



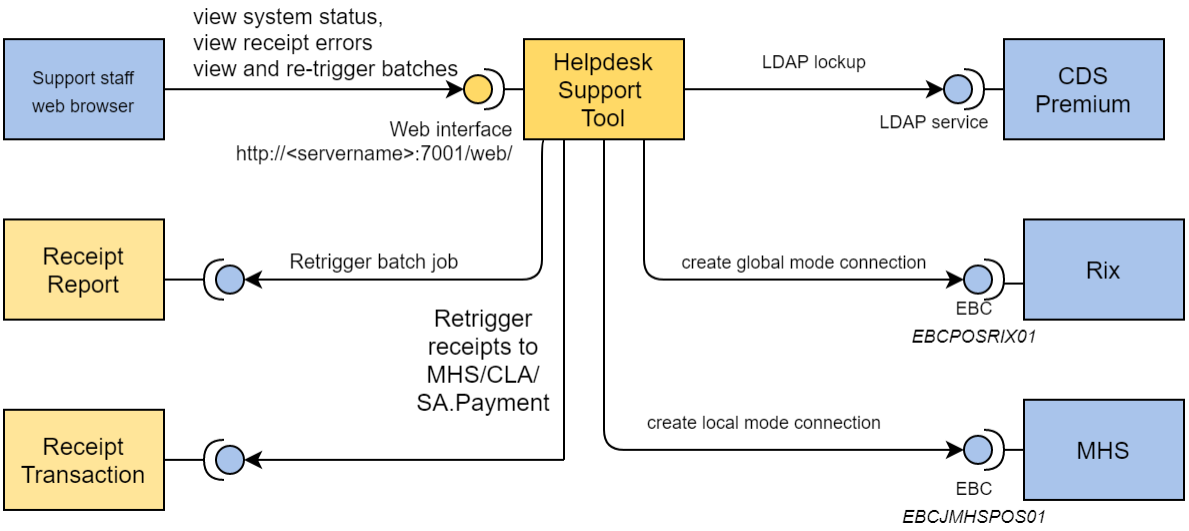
7. HelpdeskSupportTool

The purpose of the Helpdesk Support Tool application is to provide support staff with a means of viewing the status of a number of systems, to view possible receipt errors, and to view the outcome of a number of batch jobs and possibly re-trigger them.
IPOS for Food has gone live globally. Helpdesk Support Tool application also provides status of IPOS for Food receipts and connection to SPOSI.
Apart from this, it displays the cron schedule (like item check, item import, tax import), activation of profiles, activation of parameters.

In Order to get access to PIP helpdesk tool we have created a list of groups at CDS and configured roles specific to these groups in WebLogic Server. Basically these roles control access to PIP Helpdesk tool activities like trigger "item import",TRC upload, Deleting/editing NAK'ED receipts and send to downstream.
To have a read only Access to PIP helpdesk tool user needs to be added to the CDS Group- PIP_HelpdeskTool_Read_Access.

Sales Order Offline Job

After EOD is done a cron is introduced which fetches the failed orders from the PIP Error table having the error type 'SALESORDER_PAYMENT' for each store and will log that info in the Administrator log. Based on the info , an alert pattern is logged as -pip_failed_orders_errors_job Order numbers [..] and infrastructure events will be triggered .
To enable this functionality, the story has been ordered in PIP:
https://confluence.build.ingka.ikea.com/display/IPOSCCHPIP/PIP-PAYM-1386_Create+a+new+alert+for+sales+order+error+payloads+in+the+DB+as+part+of+the+EOD



8. EndOfDay

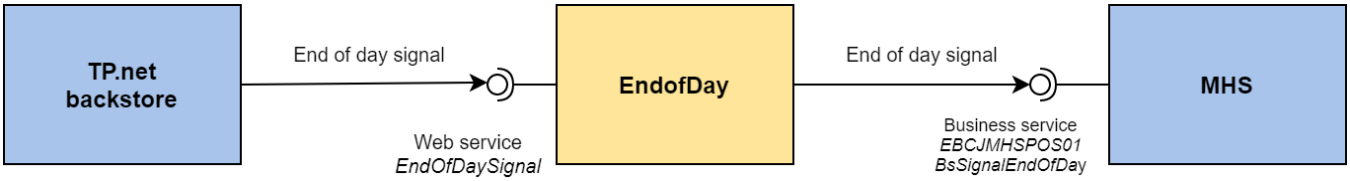
The EndOfDay application is responsible for relaying the End of Day signal to MHS. The flow of events:

1. TP.Net Backstore calls the PIP EndOfDaySignal web service.
2. The EndOfDay application checks for erroneous receipts in the PIP database and for undelivered receipt transactions on the pip.receipts.POSLogTransaction JMS topic. If none exist, the MHS business service BsSignalEndOfDay is called.
3. There are two types of erroneous receipts. RECEIPT_WARN and RECEIPT_ERROR.

End of Day will fail if there is more than 20 occurrence of RECEIPT_WARN. End of Day will fail if there is 1 occurrence of RECEIPT_ERROR.

DFSR Report:

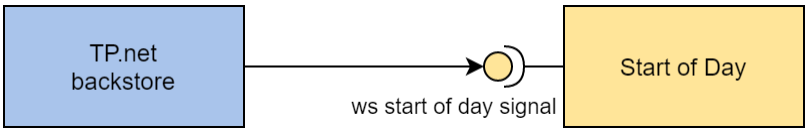
1. Part of EOD, DFSR report will get generated which is based on undelivered and delayed Transactions from PIP DB -Transaction LOG Table.
2. Change done in DFSR Report generation part of PIP 19.5 Release and logging unique PIA Items in PIP logs which are missing OCAS mapping. with this detail store wise mail sending to Foods BA Workgroup functionality has been introduced.



9. StartOfDay

The StartOfDay application is responsible for relaying the Start of Day signal to MHS. The flow of events:

1. TP.Net Backstore calls the PIP StartOfDaySignal web service.
2. Currently PIP is not using this signal for further processing.



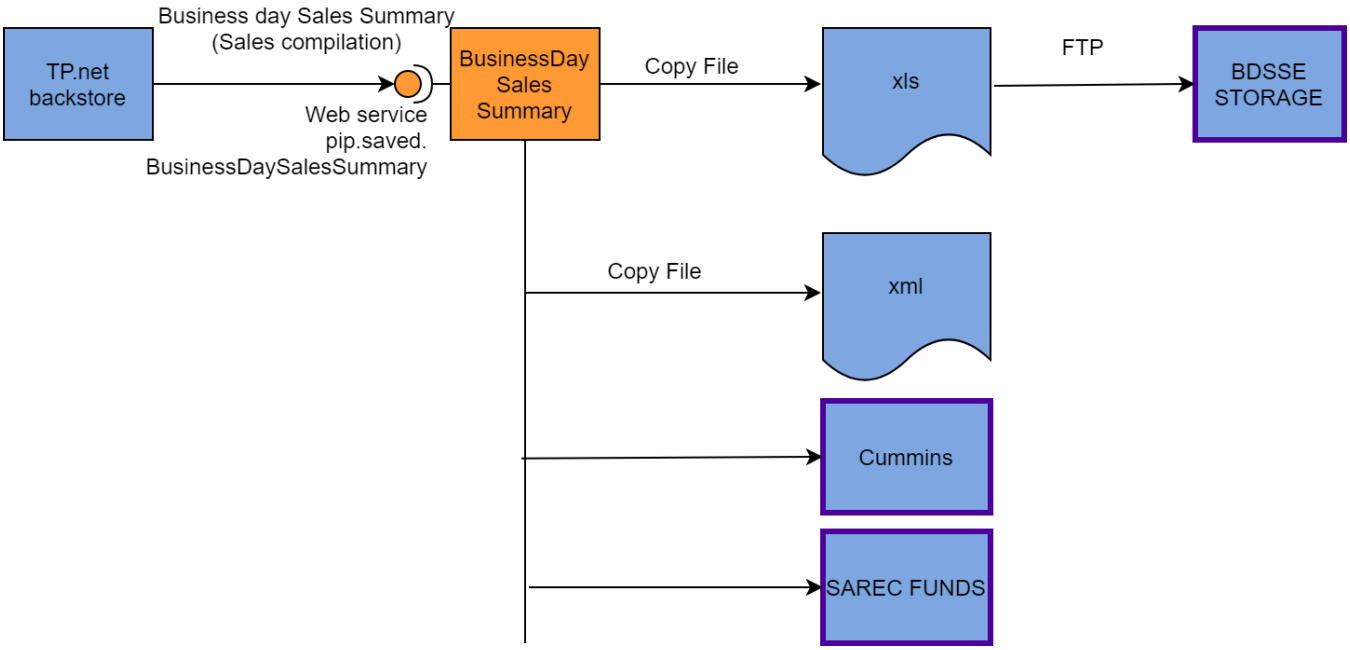
10. BusinessDaySalesSummary

The BusinessDaySalesSummary application is responsible for producing the daily sales reconciliation reports. The flow of events:

The TP.Net Backstore sends daily sales summary messages to the JMS topic pip.sarec.BusinessDaySalesSummary.

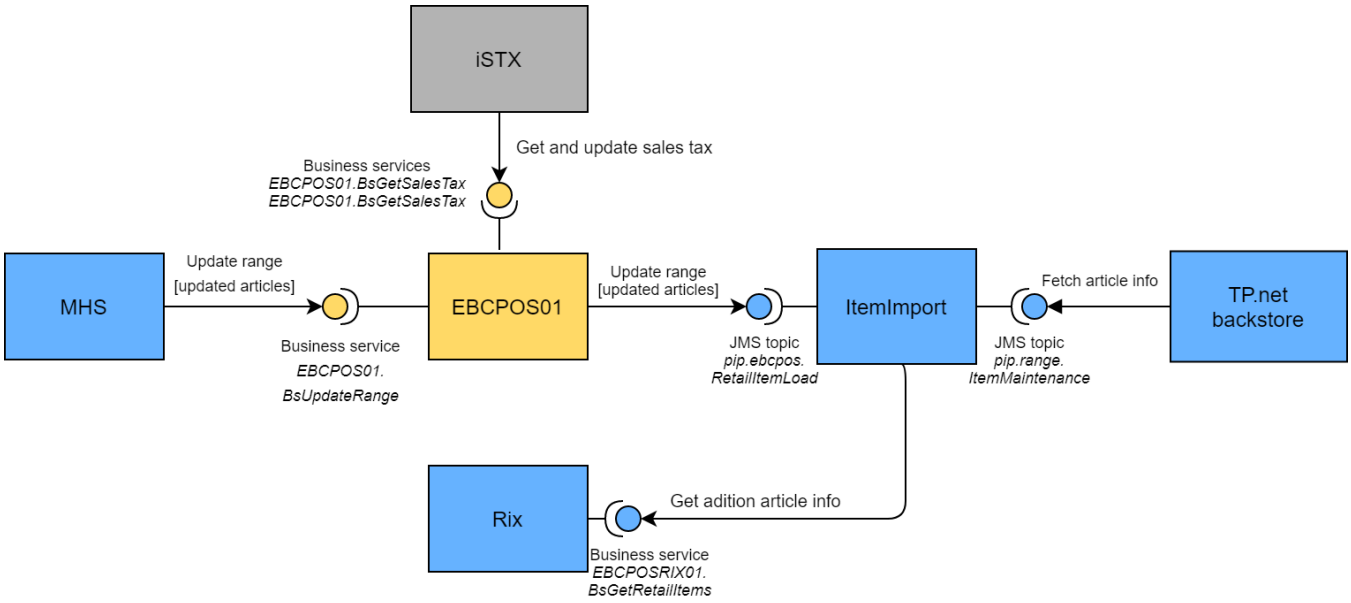
1. The BusinessDaySalesSummary application transforms the data into one xml file and one xls file and simply puts them on the file system.
2. For Norway, the xls is also sent to a FTP location.
3. BusinessDaySalesSummary application transforms the data into Declared Funds report and sends to FTP location.
4. For US, BusinessDaySalesSummary application transforms the data into Cummins report and sends to FTP location.
5. For Integrated stores new FOODs integrated template has been introduced to generate BDSSE report which is global. The change is only in xls file to introduce integrated safes details.

FTP



11. EBCPOS01

The EBCPOS01 is an EBC standardized application, with two purposes; it receives and passes on the updated article information to the ItemImport application, and it acts server to the iSTX tax administration application.



When the MHS sends updated article information to the PIP, the BsUpdateRange business service in EBCPOS01 is called. The EBCPOS01 then simply passes the information on to the pip.ebcpos.RetailItemLoadMHS to which the ItemImport application listens.

As a part of MHS TP, PIP has to update this EBB communication channel.
PIP has introduced new interface WSPOS01 to receive item information from MHS. PIP will accept the article information using webservice inbound gateway and send it to internal topic "pip.ebcpos.RetailItemLoadMHS" so that PIP can proceed with the information as it is doing in current implementation.

For detailed interface description please refer below document
https://iweof.sharepoint.com/teams/o365g_solutionteampayments_irssemal/Shared%20Documents/Forms/AllItems.aspx?id=%2Fteams%2Fo365g%5Fsolutionteampayments%5Firssemal%2FShared%20Documents%2FiPOS%2F03%20%2D%20System%20Overview%2F01%20%2D%20Interfaces%2FWS%20%2D%20Update%20Range%20MHS&viewid=286d5268%2D836a%2D4544%2D857a%2Dbf0c20905f24
The flow of events involved in the tax administration is as follows:

1. The iSTX requests data from the PIP database by calling the BsGetSalesTax business service in EBCPOS01.
2. The tax administrator edits the tax percentages, code mappings, and exemptions.
3. The iSTX then saves the updated data in the PIP database by calling the BsUpdateSalesTax business service in EBCPOS01.
4. MHS sends articles information to PIP EBCPOS01 using BSUpdateRange.
5. As part of MHS Transformation Program, IIP will be sending Family Price and Regular Price to EBCPOS01 instead of MHS.

Please note the value added tax information available in ISTX has been removed now.

PIP handles this in code level.

Also, it has been planned to handle sales tax for CA via CA TRC file.

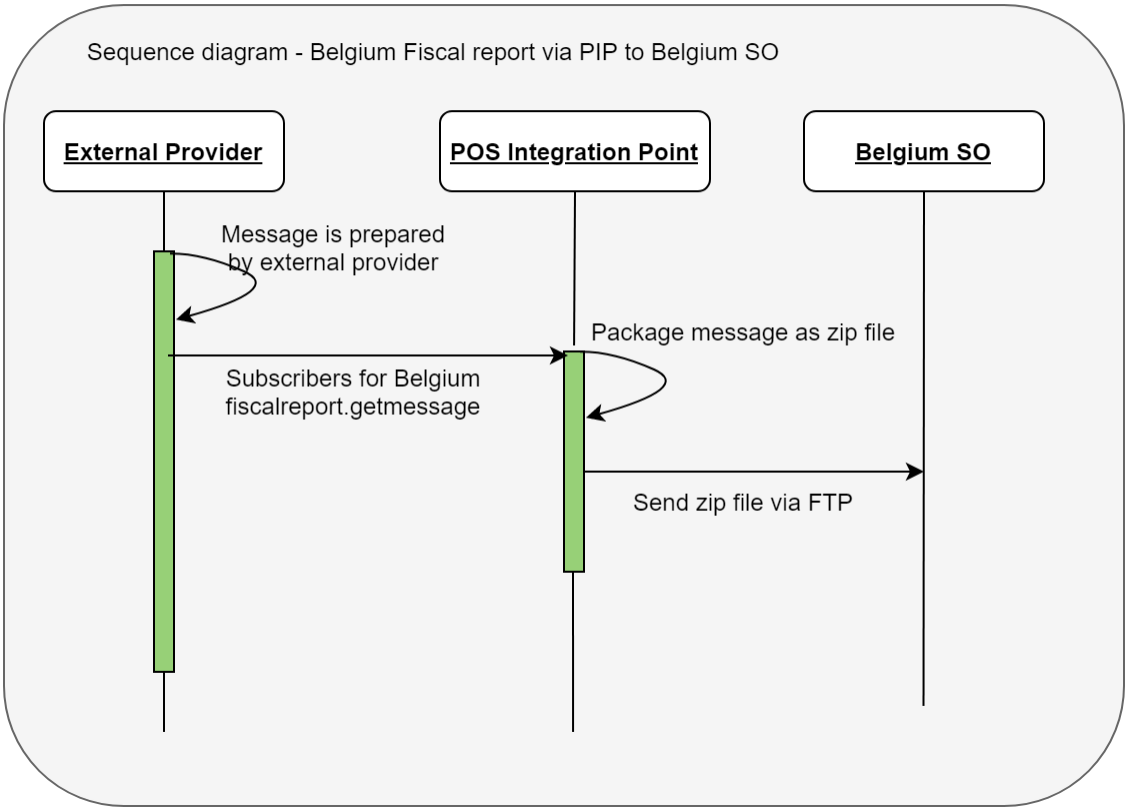
More details can be found in the below link:
https://clm.rat.itshost.se/ccm/web/projects/Solution%20Team%20Payment%20RS%20(Change%20Management)#action=com.ibm.team.workitem.viewWorkItem&id=201344

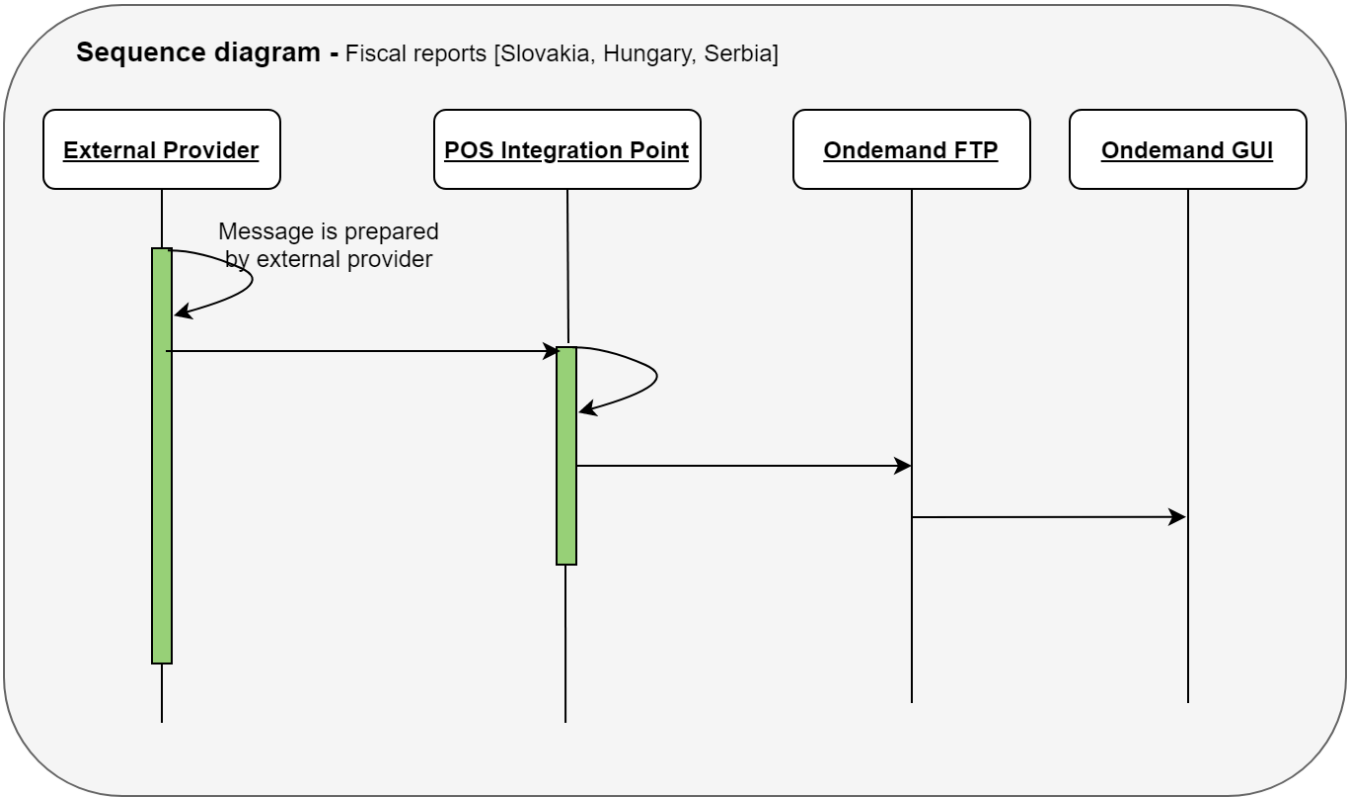
12. Fiscal

The fiscal service is designed in order to provide the POS with country-specific fiscal information.
The integration point (PIP) will mediate between the country-specific fiscal integrations.

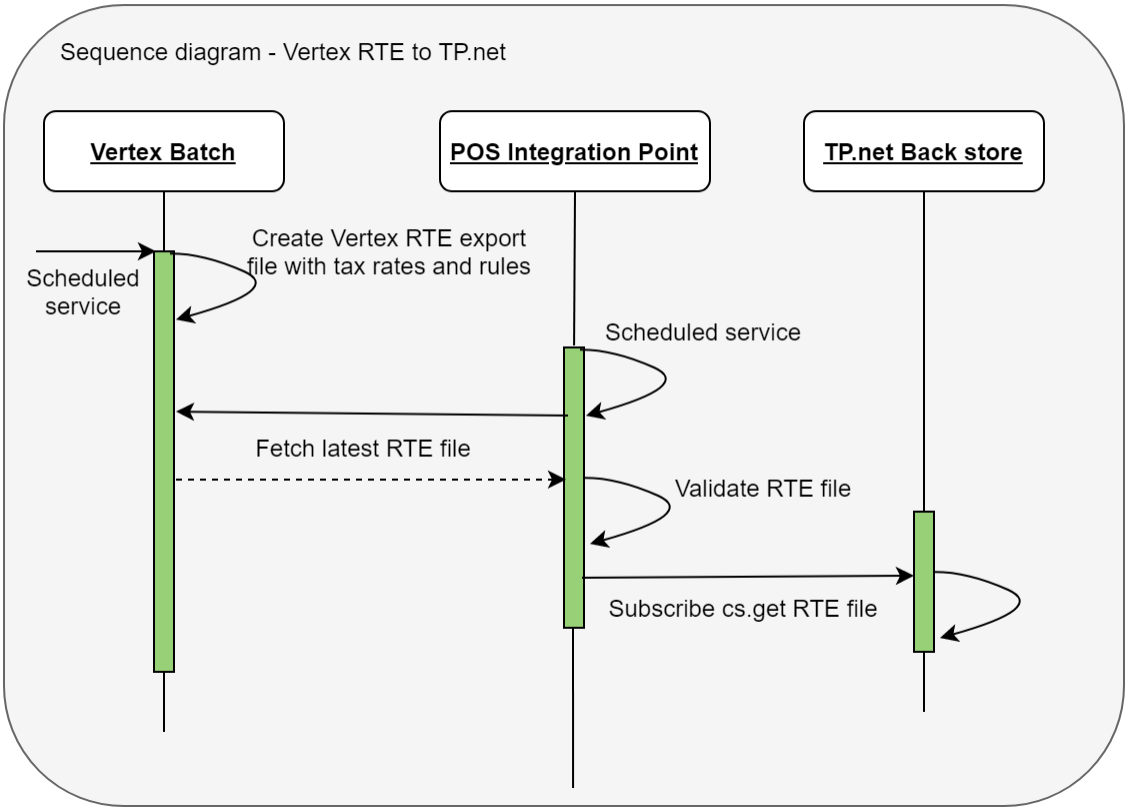
1. Belgium Fiscal Report
2. Slovakia Fiscal Report
3. Hungary Fiscal Report
4. Serbia Fiscal Report
5. Vertex RTE import No longer used.
6. MHS report
7. Customer information Find/Create/Update

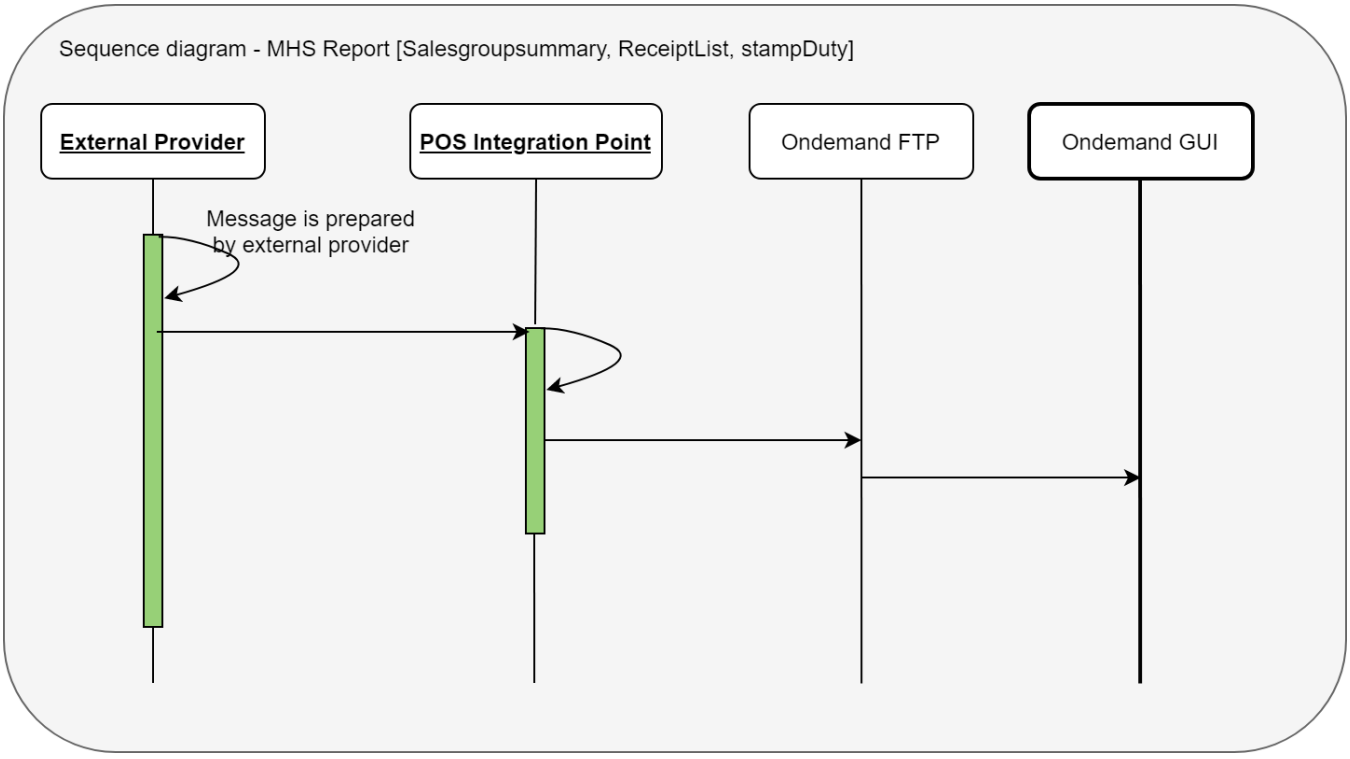
Belgium Fiscal Services



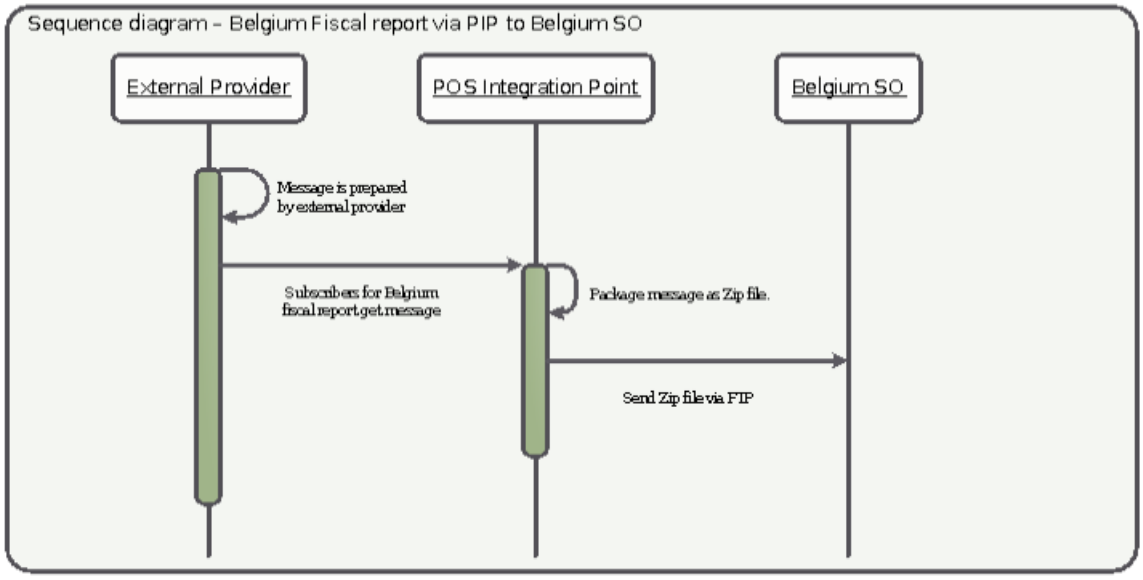


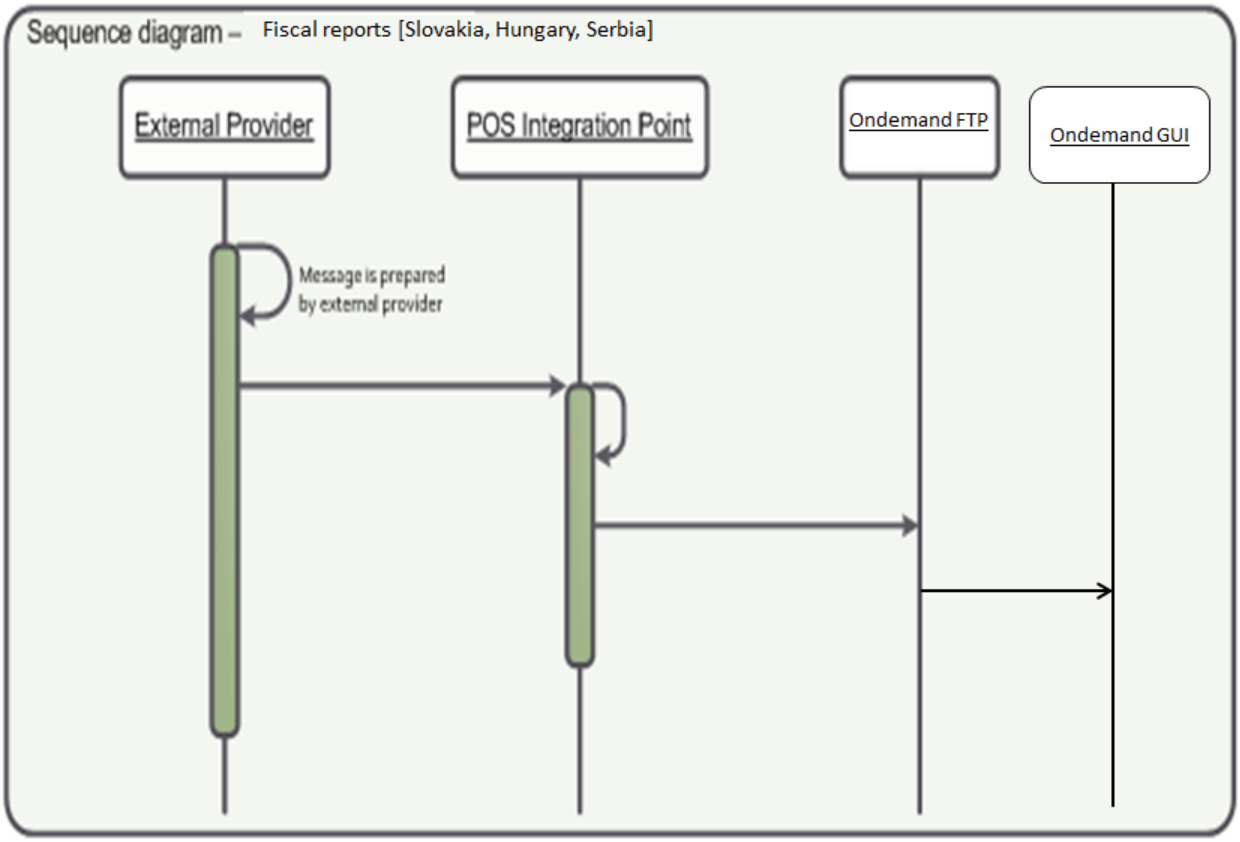
Vertex Fiscal Service



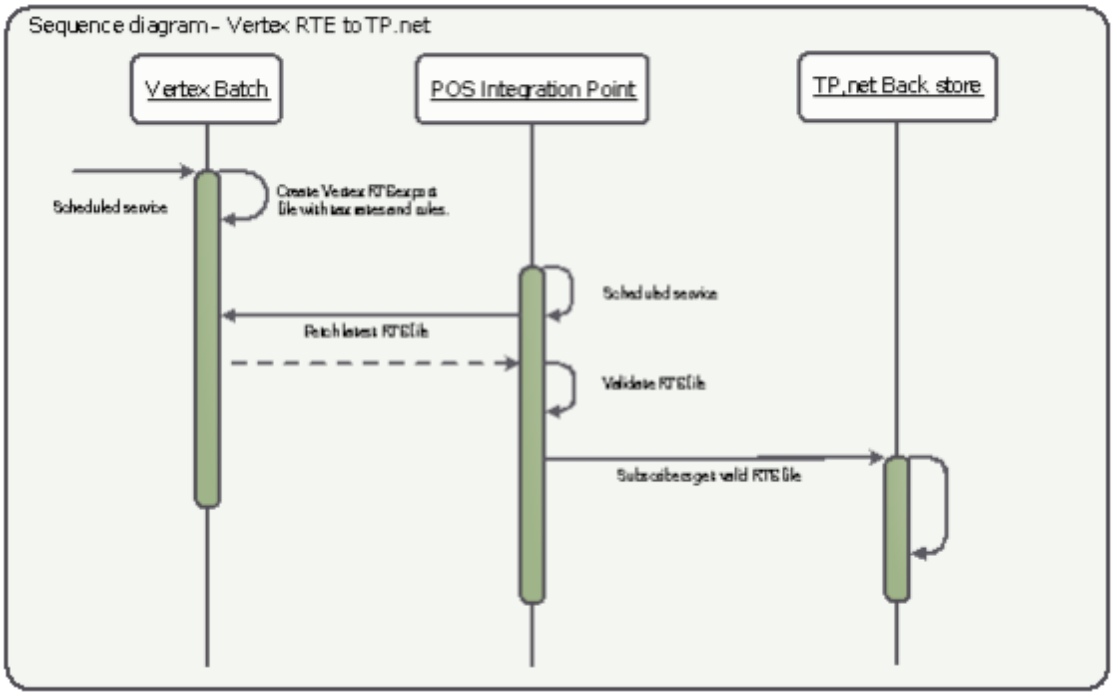


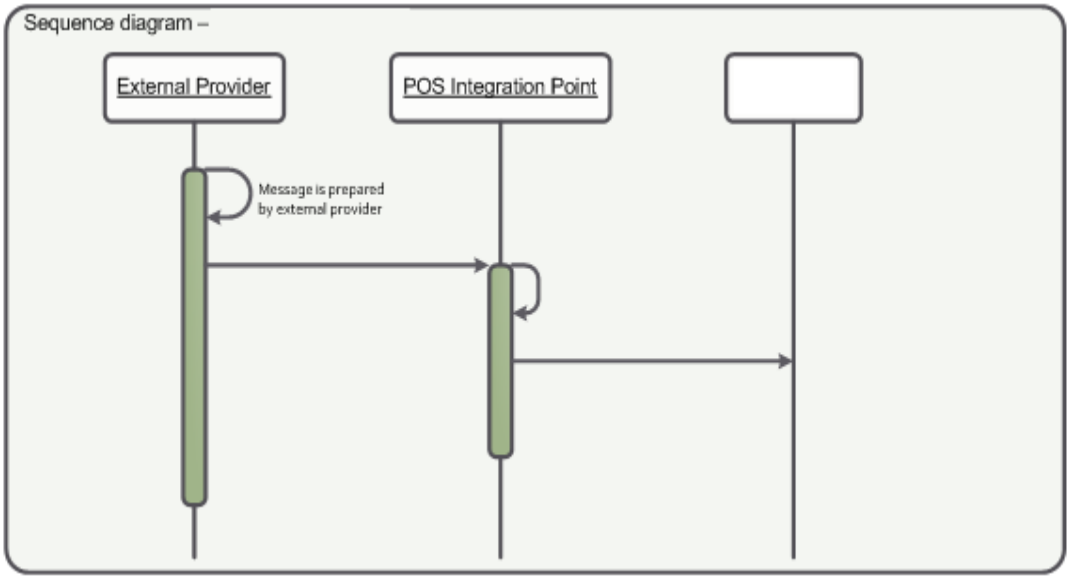
Belgium Fiscal Services





Vertex Fiscal Service





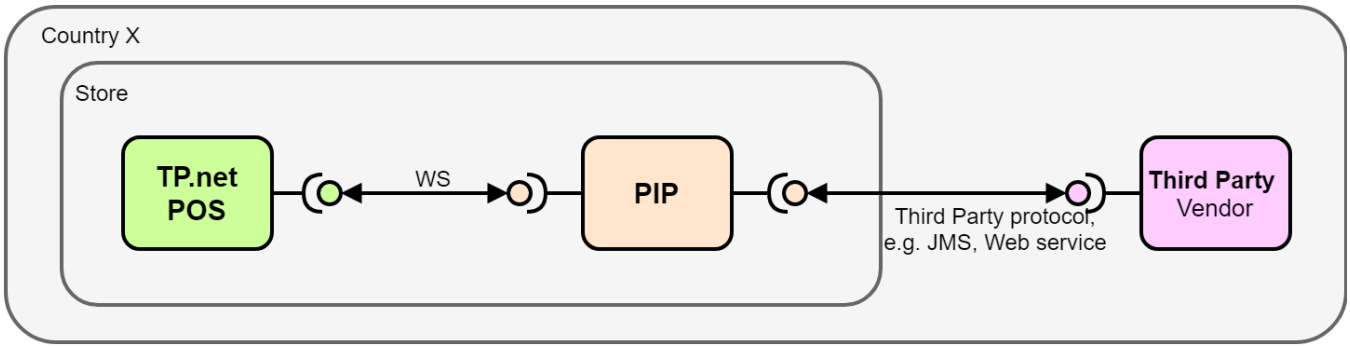
MHS reports

[Salesgroupsummary, ReceiptList, Stampduty] Ondemand FTPOndemand GUI
Fiscal also provides invoice service as legal requirement for some countries like ES, IT, HR.
The Invoice solution is generic and can be used for several countries.
The solution must be able to support the following;

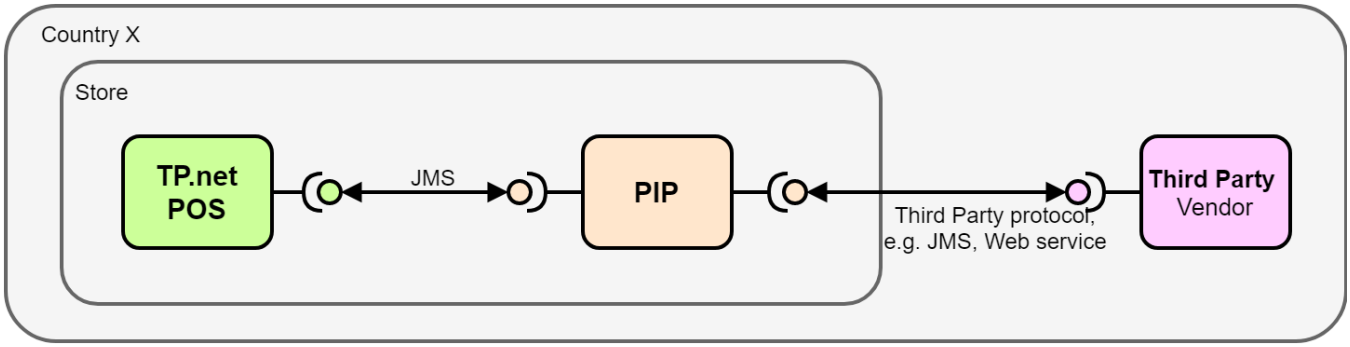
- Translate invoice information from TP.Net format to third party custom format and send to third party end point.
- Retrieve response with invoice from third party and translate and send back to TP.Net. Response will contain the invoice information that is to be printed on the receipt.
- For Spain CreateInvoice request, PIP will receive Invoice Number from third party i.e. TPInvoice and send this Invoice number to Tp.Net. So that Tp.Net will use this invoice number to create the barcode on receipt. This barcode will save cashier's efforts to enter invoice number manually at the time of return.

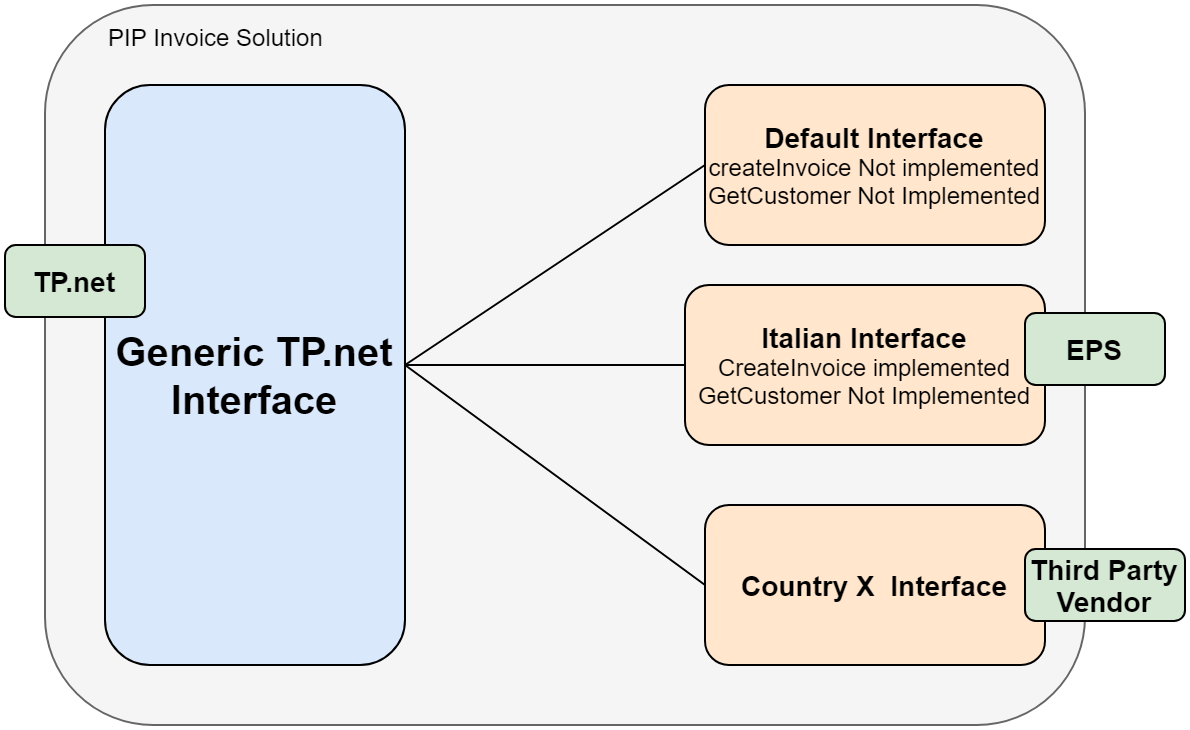
This change comes with a backward compatibility, if Tp.Net is not ready to accept invoice number then PIP will not send it. This configuration is maintained in pip.properties file

Generic Invoice solution - online

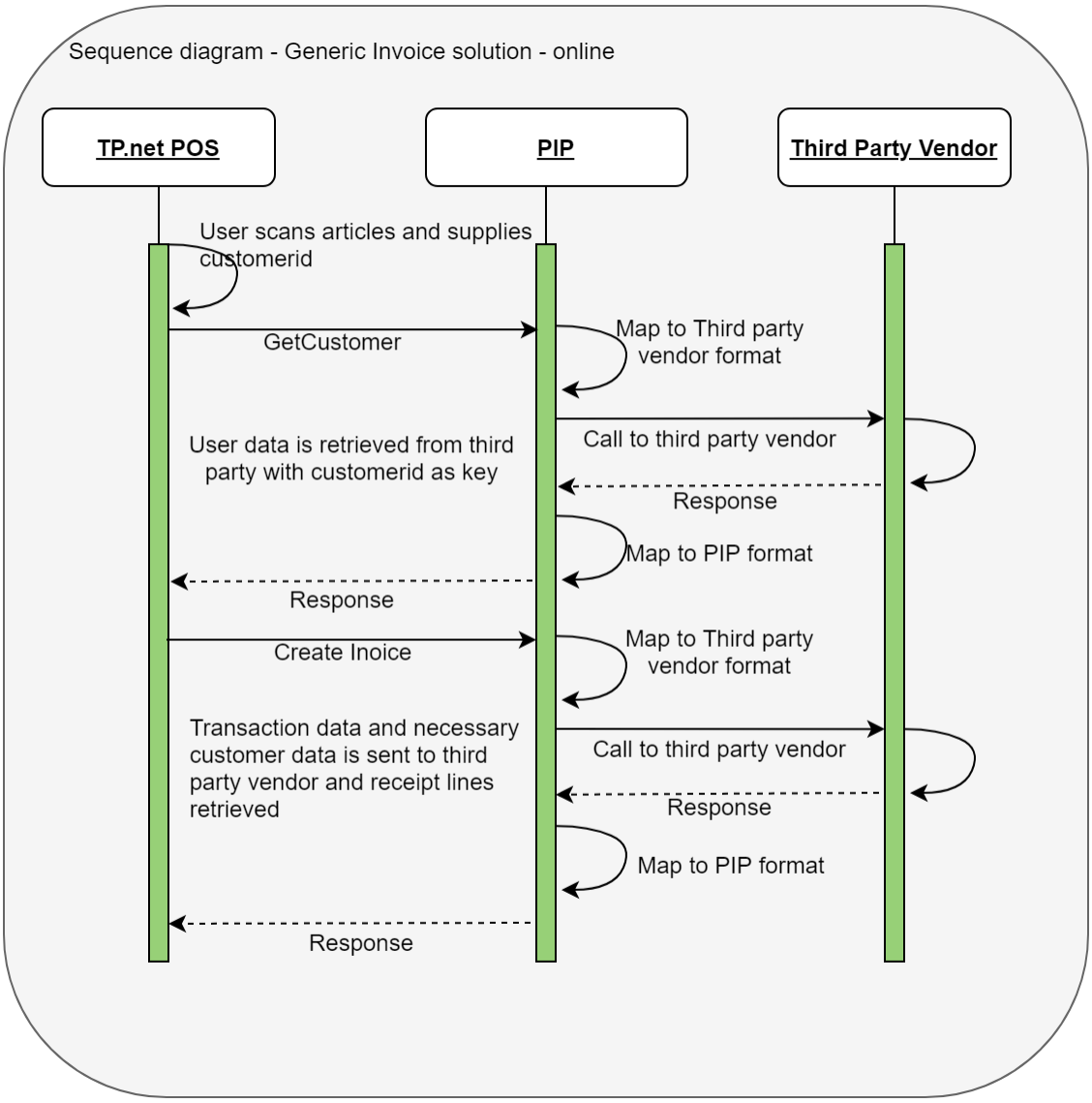


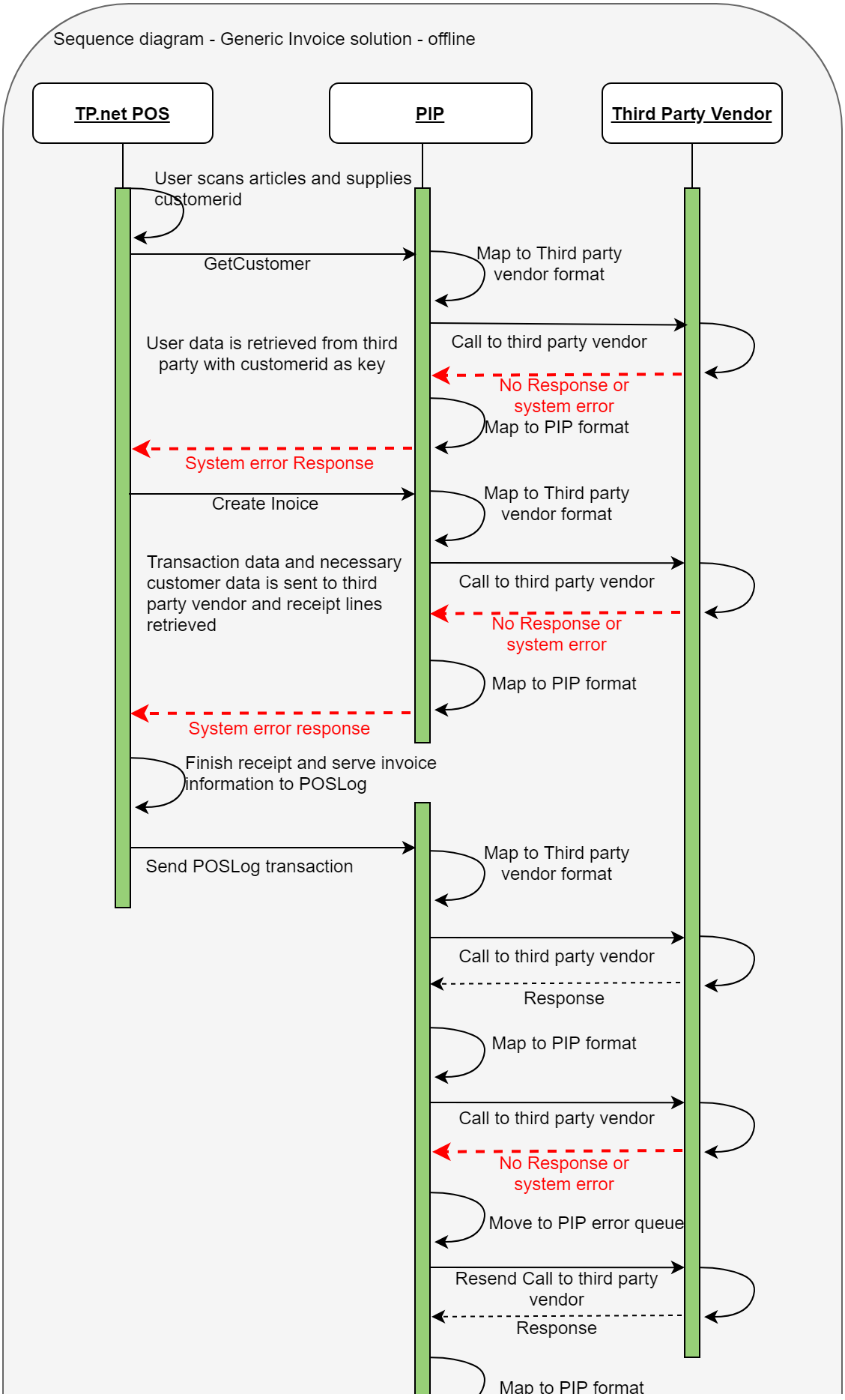
Generic Invoice solution - offline

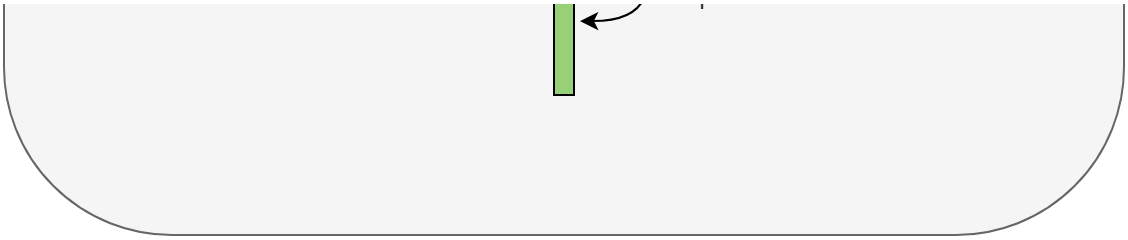




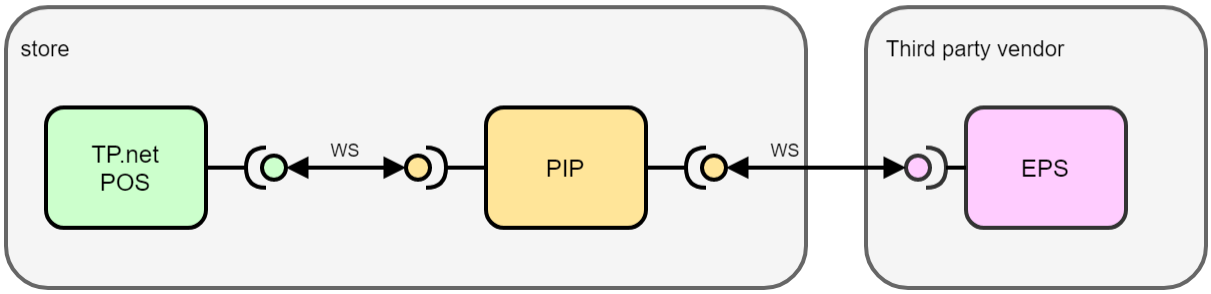
Two operations are available: CreateInvoice and GetCustomer. It will always be possible to call the two operations, but a simple "Not implemented" message will be returned in all countries other than the ones where they are used.


Sequence diagram - Generic Invoice solution - online
TP.net POS
PIP
Third Party Vendor
User scans articles and supplies customerid
GetCustomer
Map to Third party vendor format
Call to third party vendor
User data is retrieved from third party with customerid as key
Response
Map to PIP format
Response
Create Inoice
Map to Third party vendor format
Transaction data and necessary customer data is sent to third party vendor and receipt lines retrieved
Call to third party vendor
Response
Map to PIP format
Response

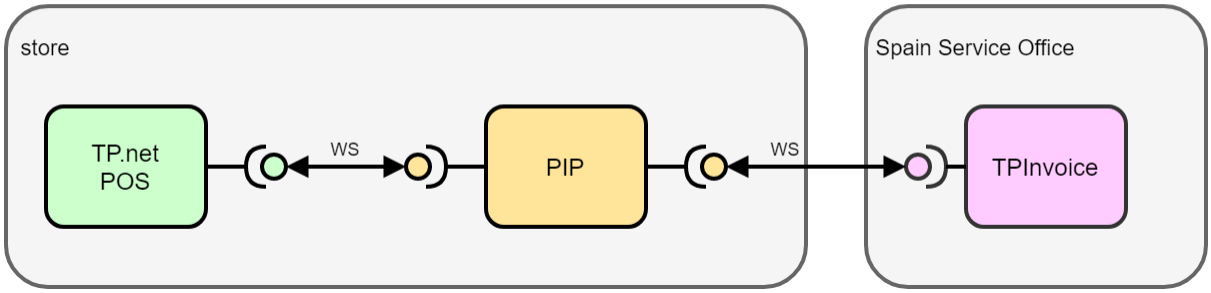

Sequence diagram - Generic Invoice solution - offline
TP.net POS
PIP
Third Party Vendor
User scans articles and supplies customerid
GetCustomer
Map to Third party vendor format
Call to third party vendor
User data is retrieved from third party with customerid as key
No Response or system error
Map to PIP format
System error Response
Create Inoice
Map to Third party vendor format
Transaction data and necessary customer data is sent to third party vendor and receipt lines retrieved
Call to third party vendor
No Response or system error
Map to PIP format
System error response
Finish receipt and serve invoice information to POSLog
Send POSLog transaction
Map to Third party vendor format
Call to third party vendor
Response
Map to PIP format
Call to third party vendor
No Response or system error
Move to PIP error queue
Resend Call to third party vendor
Response
Map to PIP format



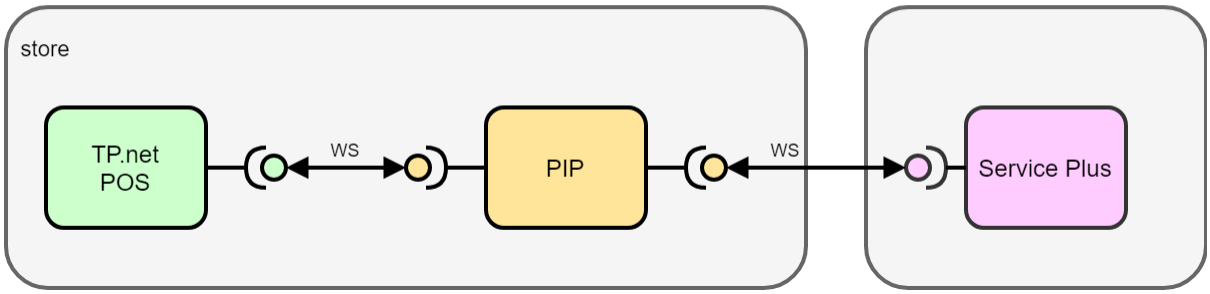
Italien Invoice



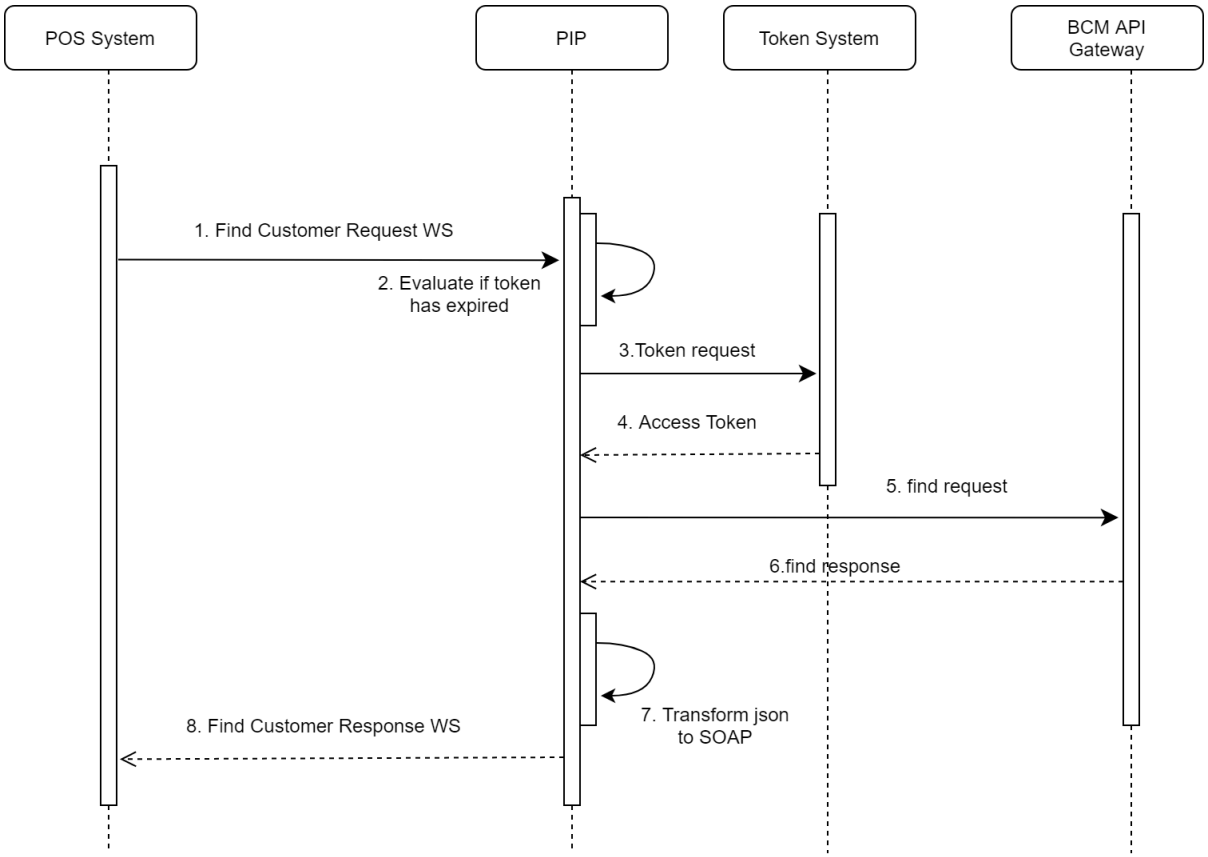
Spain Invoice



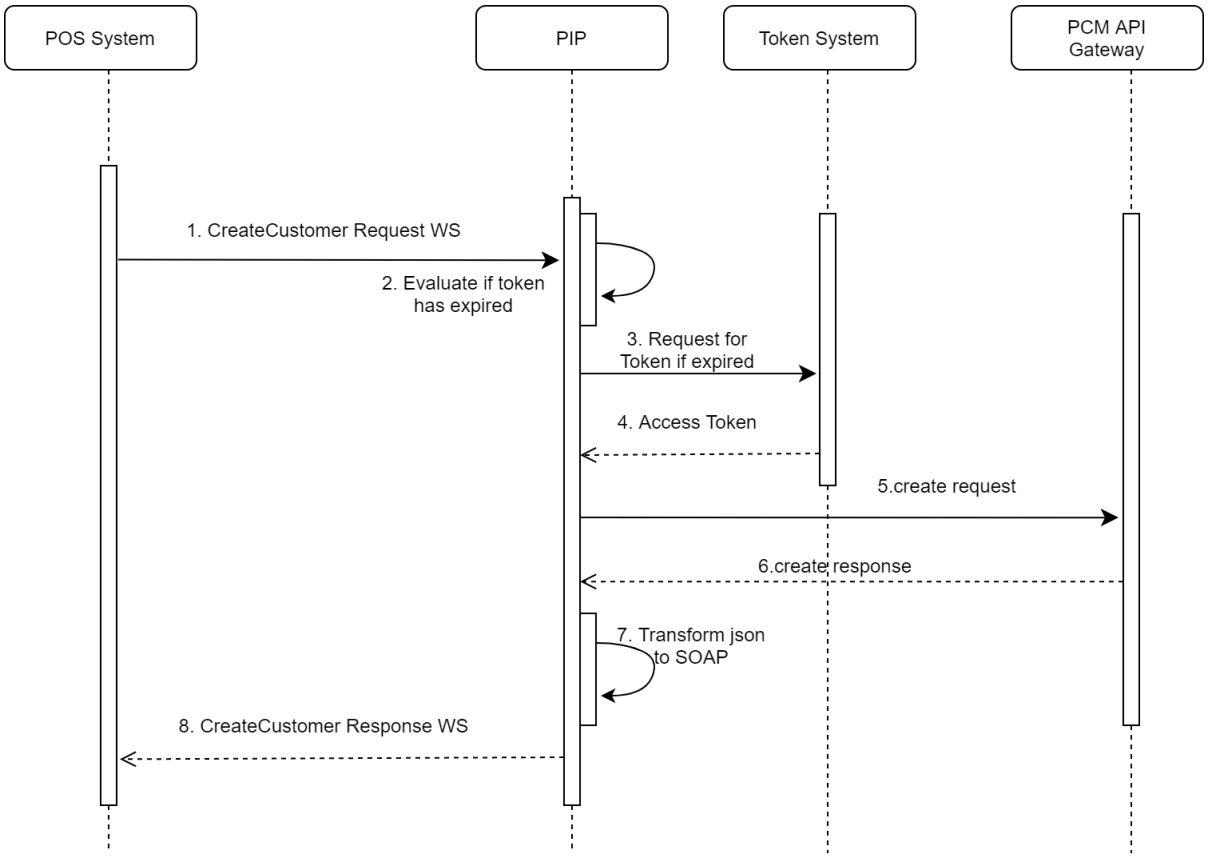
Croatia Invoice

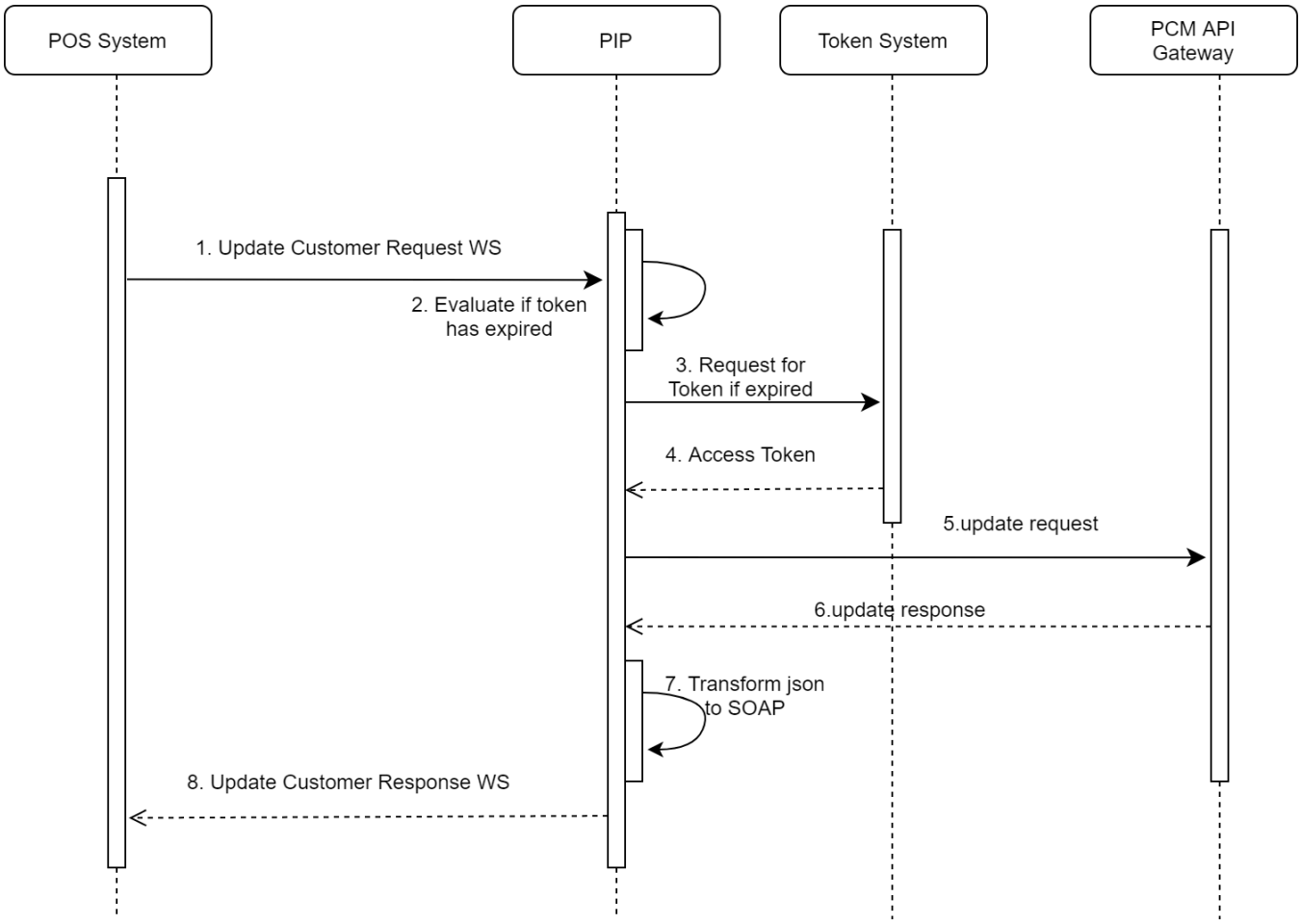


BCM flow:


POS System
PIP
Token System
BCM API Gateway
1. Find Customer Request WS
2. Evaluate if token has expired
3.Token request
4. Access Token
5. find request
6.find response
7. Transform json to SOAP
8. Find Customer Response WS

PCM flow:


POS System
PIP
Token System
PCM API Gateway
1. CreateCustomer Request WS
2. Evaluate if token has expired
3. Request for Token if expired
4. Access Token
5.create request
6.create response
7. Transform json to SOAP
8. CreateCustomer Response WS



13. Employee Discount

CDS is a corporate directory available for hundreds of services and applications within IKEA.
Corporate Directory Services (CDS) is a very important part of the IKEA security concept, aiming at increased security and simplified user administration for our Helpdesks.

The main task of CDS is to act as a Single Identity Provider, as CDS is the main source within IKEA for identity. Another key function in CDS is to tie co-worker data to given roles/ privileges, hence giving users the correct access level of business applications.

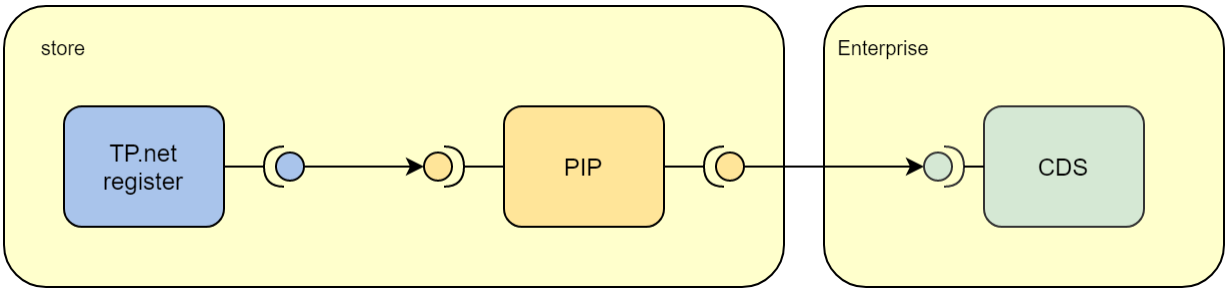
CDS stores information about more than 135 000 IKEA co-workers and the number is increasing.

Automatic employee discount verification will create efficiency in the receive payment process and reduce manual administration. This will also improve the accuracy in who is actually granted an employee discount, since the verification will done against IKEAs CDS system. Currently any employee discount needs to be verified manually in the store. This option will also remain with the new setup, but it will be reduced to a manual backup routine if the automatic approach fails.

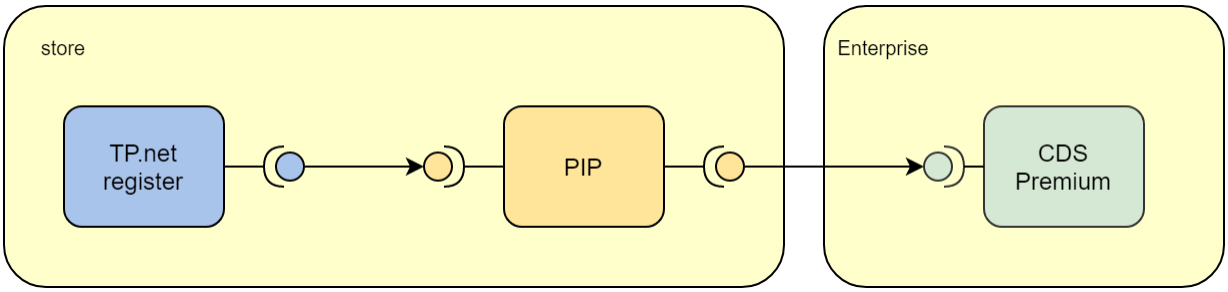
The solution must be able to support the following;

- Expose a Web Service from the POS Integration Point to be consumed by POS for Employee Discount Verification
- Secure the backend communication towards CDS to support the exposed Web Service mentioned above

The PIP will expose a Web Service for the discount verification. Registers will have the possibility to consume this service when a customer prompts the cashier for employee discount.



Now, CDS being upgraded to CDS Premium, the same functionality will be achieved by CDS Premium.



14. Cards

The Business Customer Card Authorization (IKEA Business) verifies that the customers' contract number is valid as well as current available credit balance. This goes for the Private Customer Card Authorization (IKEA Handla) as well. The difference is that another source system is used for retrieving the authorization, e.g. IKANO instead of RIMS.

Business to Business Credits is handled via the RIMS application. A quote from RIMS user guide, "RIMS stands for Receivables & Invoice Management System. RIMS is responsible for the creation and maintenance of Business to Business accounts, and the production of invoices."

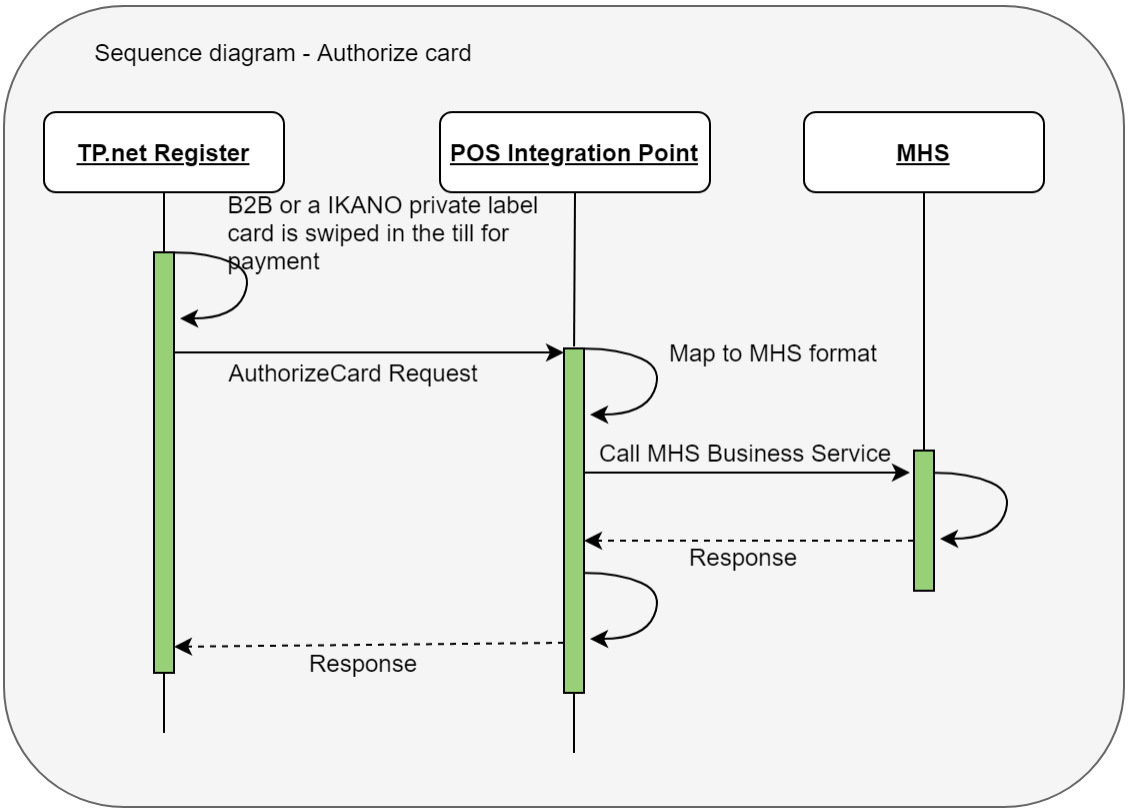
The private customer card authorization verifies that the customers' contract number is valid as well as current available credit balance

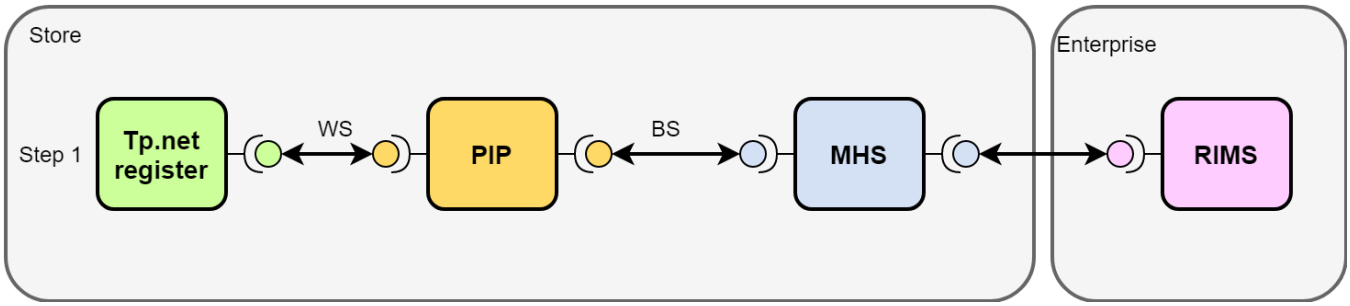
Since MHS is end of life and will be closed down, the capabilities performed by MHS will be moved to other systems. In RIMS B2B case, the MHS capabilities will be moved to PIP.PIP will call new RIMS service "BsAuthorizePurchase", EBCName: EBCRIMS1 using WSProxy. This business service is used to authorize the credit card.It is used when a RIMS card is swiped in POS in order to authorize a purchase.

In the next step of the MHS TP IKANO is being moved from MHS to IIP which needs to be handled in the POS Integration Point. Earlier Ikea Business cards were handled by MHS but as this functionality id being moved to remove dependency on MHS no PIP will validate IKANO cards with IKANO directly and register the sales as well.

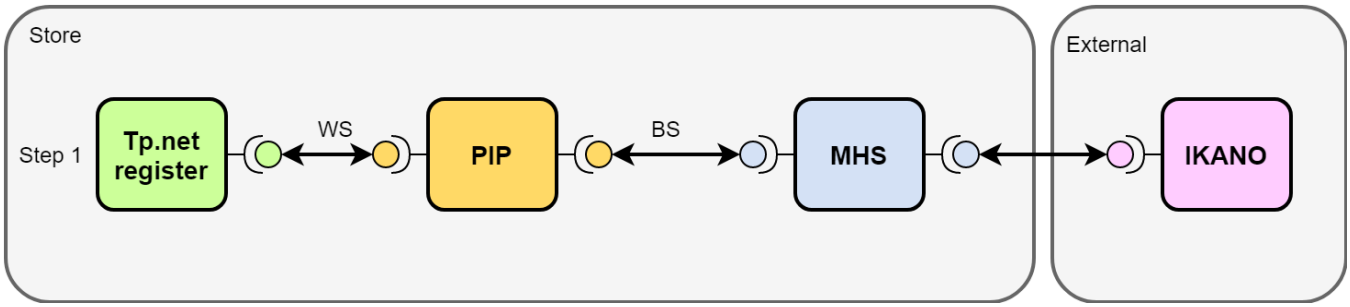
IKANO cards are for countries NO, DE, SE

RIMS cards are used for rest of the countries in IKEA.

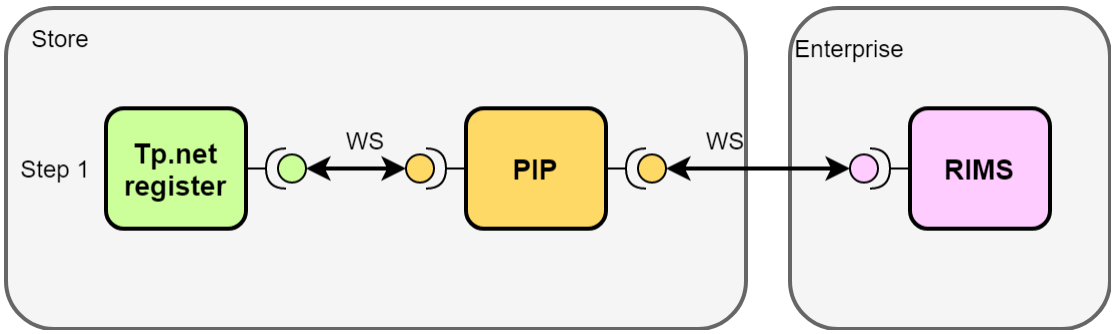




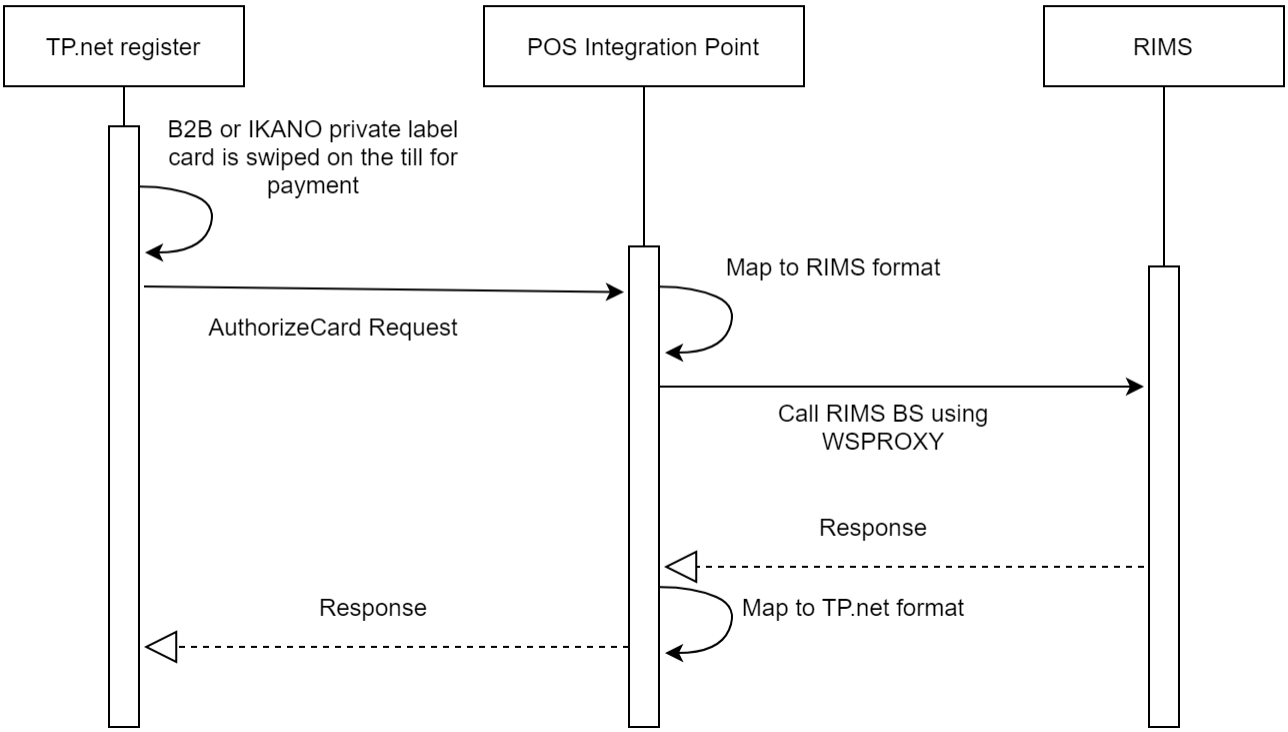
- Figure 1 depicts the overall information flows included in the Authorize card for B2B (RIMS) using MHS



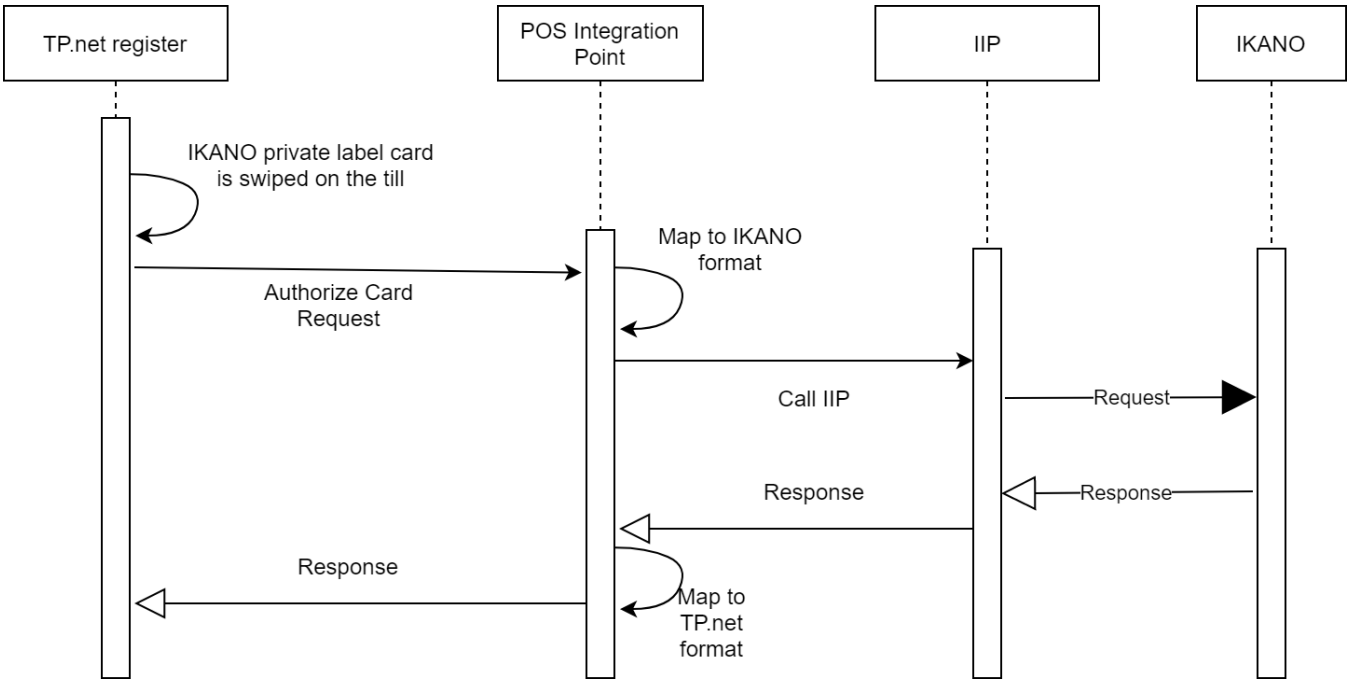
- Figure 2 depicts the overall information flows included in the Authorize card for Private Label cards (IKANO) using MHS



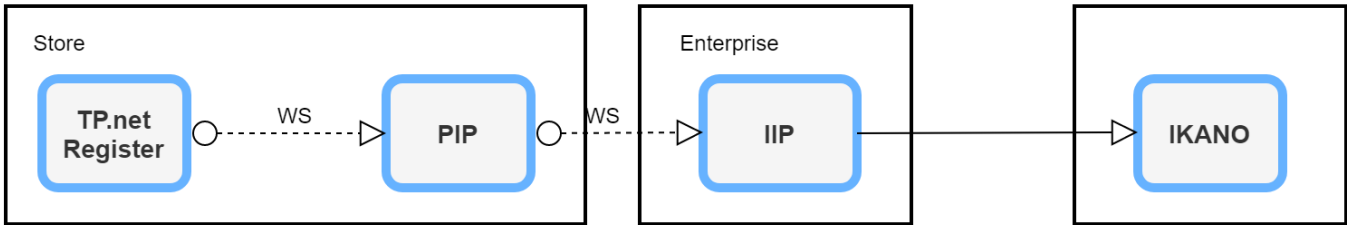
Sequence Diagram - AuthorizeCard



- Figure 3 depicts the overall information flows included in the Authorize card for B2B (RIMS) using WSProxy.



Card Authorization



- Figure 4 depicts the overall information flows included in the Authorize card for Private Label Cards (IKANO) using IIP Web Service.

SAP-BOT:
INGKA is introducing a new solution for Business Credit Customers based on an SAP platform. The integrations which iPOS today has with RIMS will be changed with BOT i.e. Change credit check call from BSAuthorizePurchase to new API on API Management platform. Sales Completion report will not be generated for the stores where we enable this SAP-BOT Solution.

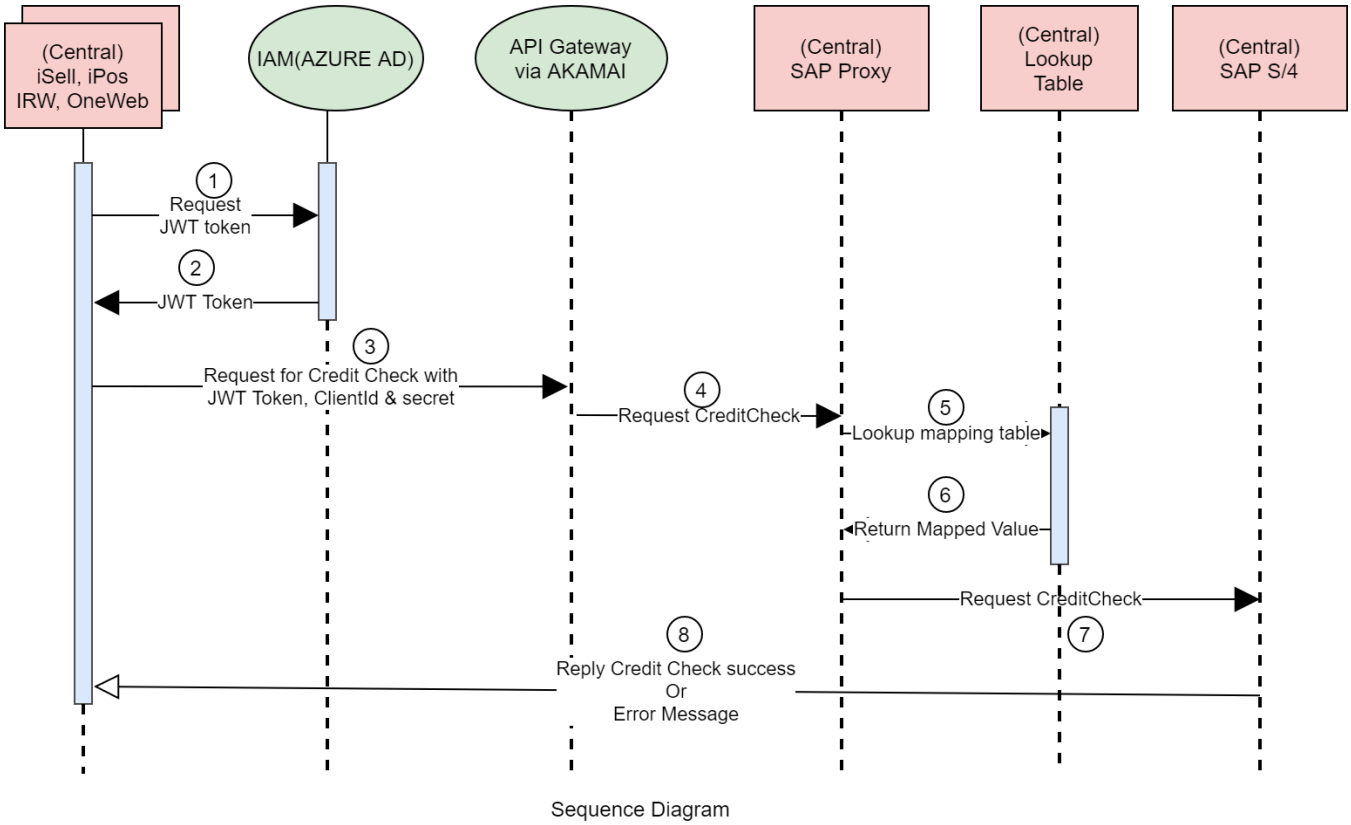


Figure 5 depicts the sequential flow of data included in the Authorize card for B2B (BOT) using IAM system (AUth0).

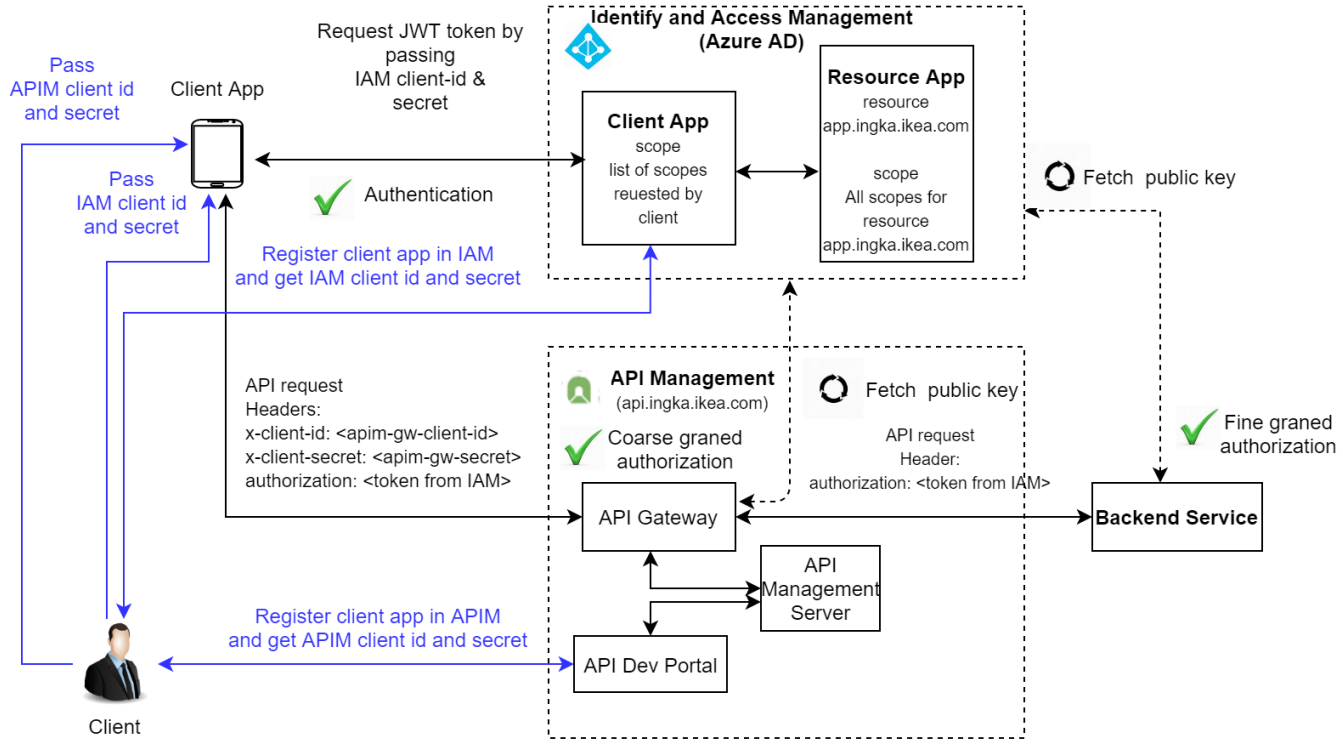
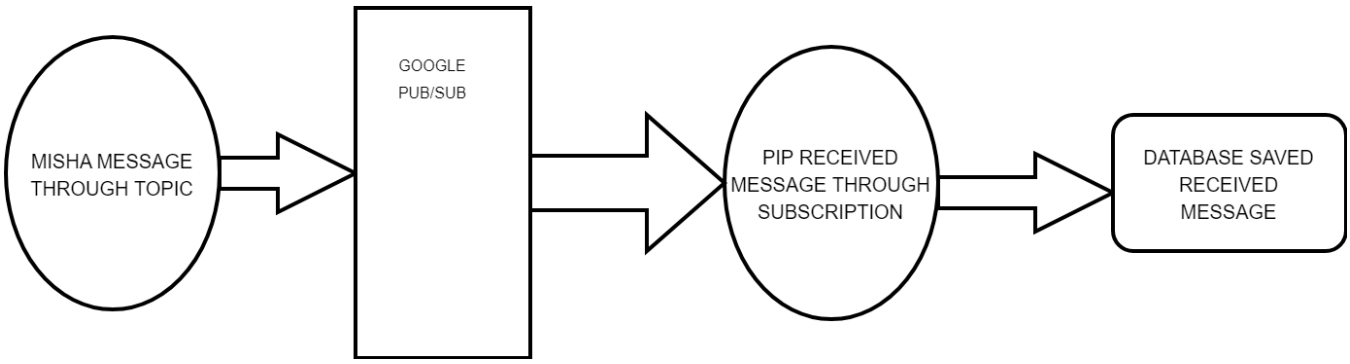


Figure 6 depicts the overall information flows included in the Authorize card for B2B (BOT) using IAM system (AUth0).

PubSub Module:
Misha has published the message using Google cloud Pubsub Topic.PIP will be received the message using Google Cloud Pub sub subscription. As per format Misha has published the message in JSON fomat.PIP will be subscribe the message and store in database.



1.2.1. Application Log Retention

PIP needs to implement a change so that all the below PIP application log will be available for 6 months (185 days approx) on the location "/ikea/pip/logs" only for China :

1. pip-administration.log
2. pip-cards.log
3. pip-employee.log
4. pip-endofday.log
5. pip-fiscal.log
6. pip-itemimport.log
7. pip-itemprice.log
8. pip-taximport.log
9. pip-receiptreports.log
10. pip-receipts-duplicate.log
11. pip-receipts-food.log
12. pip-receipts-inflow-check.log
13. pip-receipttransaction.log
14. pip-return.log
15. pip-salesorder.log
16. pip-salesorder-addPayment.log
17. pip-salesreconciliation.log
18. pip-fiscal-customer.log
19. pip-cla-receipts.log
20. pip-receipt-transaction-valid.log
21. pip-receipt-transaction-invalid.log
22. pip-pubsub.log

PIP has implemented the change in such a way that Number of Days for storing log file is configured in PIP configuration file and will be set in "pip-log-cleanup.sh" file while installing the PIP build. Below files has been modified where the static value is removed from all Log4j files.

SN	Location of the change (Program / Module / Database / Document Name)	Type of Change ⭐	Brief Description of change
1	pip-administration-log4j.properties pip-cards-log4j.properties pip-fiscal-customer-log4j.properties pip-fiscal-log4j.properties pip-itemimport-log4j.properties pip-itemprice-log4j.properties pip-receipt-transaction-invalid-log4j.properties pip-receipt-transaction-valid-log4j.properties pip-receiptreports-log4j.properties pip-receipts-cla-log4j.properties pip-receipts-duplicate-log4j.properties pip-receipts-food-log4j.properties pip-receipts-inflow-check-log4j.properties pip-receipttransaction-log4j.properties pip-return-log4j.properties pip-salesorder-addPayment-log4j.properties pip-salesorder-log4j.properties pip-salesreconciliation-log4j.properties pip-startofday-log4j.properties pip-taximport-log4j.properties pip-pubsub-log4j.properties	M	To remove the static value for storing the log.
2	pip-install/conf/pip-log-cleanup.sh	M	To set the dynamic value configured in PIP configuration file.
3	Installation script	M	To convert pip.properties from Dos to unix format.

Below configuration need to be done in pip.properties :

6.2.2 POSLog Retention

PIP Needs to implement a change so that Poslog will be available for 6 months (185 days approx) on the location "/ikea/pip/sarec/poslog" only for China.
PIP will implement a change in such a way that Number of Days for storing POSlog file will be configurable from PIP configuration file.

Server Space Information:

Country Code	Store No	Total POSLog Size in MB for 30 Days	Total POSLog Size in MB for 185 Days	Space allocated for folder "/ikea"
US	211	176	1056	20 GB
BE	179	222	1332	20 GB
GB	185	300	1800	20 GB

PIP has implemented the change in such a way that Number of Days for storing POSlog file is configured in PIP configuration file and will be set in "pip-log-cleanup.sh" file while installing the PIP build.

SN	Location of the change (Program / Module / Database / Document Name)	Type of Change ⭐	Brief Description of change

1	pip.properties	A	#Number Of Days for Retention of POSLog Batch
			CN.PIP_NO_OF_DAYS_POSLOG=185
			PIP_NO_OF_DAYS_POSLOG=30

SN	Location of the change (Program / Module / Database / Document Name)	Type of Change ⭐	Brief Description of change
1	pip-install/conf/pip-log-cleanup.sh	M	To set the dynamic value configured in PIP configuration file.
2	Installation script	M	To convert pip.properties from Dos to unix format.

The CleanUp script "pip-log-cleanup.sh" is schedule on the linux server to be executed on everyday. Below files on server contains this configuration :
/etc/cron.daily :
The cron expression to execute the "cron.daily" is configured in "anacrontab" file in the "/etc" folder.

7.OPERATIONAL REQUIREMENTS

7.1. Application Administration

<<List down any administration tasks here & other related information>>

7.2. DR Process & documentation

<<Provide the disaster recovery process and path to documentation>>

7.3. Daily, Weekly, Monthly, Quarterly, Monthly Checks/Activities

7.3.1 DAILY CHECKS

S No.	Task
1	NAK Check
2	EOD Check
3	Invoice fail check
4	Monitoring Store report batch

7.3.2 WEEKLY CHECKS

7.3.3 MONTHLY CHECKS
7.3.4 NEW STORE ROLLOUT
iPOS migration in stores

7.4. On Demand/Request

On Demand request comes from the Business (Global Accounting) for specific reports.
The SQLs have been created for commanly requested reports.

7.5. Backup Information

<<Please provide all the backup process, timings and location>>

8.PROCESS RELATED

8.1 Change Control

PIP Configuration Manager: Gillis Johnsson
Backup: Sudarshan K, Shilpa S

8.2. Problem Management

PIP Problem Manager: Madhulika Singh

8.3. Release Management

Release history and details are available in CLM in the below link:
https://clm.rat.itshost.se/ccm/web/projects/Solution%20Team%20Payment%20RS%20(Change%20Management)#action=com.ibm.team.workitem.runSavedQuery&id=_XP-dgFjeEeikXs2uXvKdaw

8.4.
Size/Complexity/Criticality

Size:
Complexity: High
Criticality: High

8.5. Coding Standards

Code is self-documenting.
Errors are detected by code-rule checking.
Code comments are up-to-date.
Proper Info and Debug comments are placed to help identify the functionality.
The operation name helps to identify what the operation does.
Code is easy to understand and maintain.
CONTACT INFORMATION

8.6. Business Contacts

Service Owner: Henrik Wifvesj
Service Manager: Juliann Veeger
Onshore Manager: Shafeek Pareed
Offshore Manager: Amrita Chowdhury
Offshore Senior Manager: Ravindra Kumar

8.7. Support Team Contacts

9.2.1 GWLS:
IDESK Group: GWLS - Weblogic Support
Support group mail id: dl.gwls@ingka.com
Distribution List: GWLS
Jasneet Singh Dutta (External - HCL Technologies Ltd),
Prasana Manne (External - HCL Technologies Ltd),
Rekha Guvvakallu (External - HCL Technologies Ltd),
Pujitha Vura (External - HCL Technologies Ltd).

9.2.2 GDBA:
IDESK Group: GDBA - Global Oracle DBA
Support group mail id: pudla.dbaseext@ingka.com; production.oracledba.se@ikea.com
Dinesh Adhikari (External - HCL Technologies Ltd),
Boopathy Ramamoorthy (External - HCL Technologies Ltd)

9.2.3 MHS:
IDESK Group: CAP-RM-MHS-SL1
Support group mail id: dlmhs.slfteam@ingka.com
Chandu Prasad (External - Capgemini India Pvt. Ltd),
Aparajita Chanda (External - Capgemini India Pvt. Ltd),
Dasari KalyanReddy (External - Capgemini India Pvt. Ltd)

9.2.4 RIX:
IDESK Group: CAP-RM-RIX-SL2
Support group mail id: dl.rix.slf@ingka.com
Distribution List: RIX SUPPORTTEAM
Poonam Shukla (External - Capgemini India Pvt. Ltd)

9.2.5 IBIS:
IDESK Group: IBM-CMB-BRS-SL2
Support group mail id: dl.bi.pfu.in@ingka.com; dl.bi.obp.in@ingka.com
Amritangshu Saadu (External - IBM India Pvt. Ltd)

9.2.6 HELPDESK:
IDESK Group: GIDESK - Global iDesk Support
Support group mail id:csan@ingka.com

9.2.7 Ondemand:
IDESK Group: IBM-SCS-ONDEMAND-SL2
Support group mail id: dl.ondemand.scs.in@ingka.com
Vainu Kulkarni (External - IBM India Pvt. Ltd)

9.2.8 IIP:
IDESK Group: GINT - Global Integration support Team
Support group mail id: dl.gint@ingka.com
Ajay Kumar Dubey (External - HCL Technologies Ltd)
Aarti Saini (External - HCL Technologies Ltd)

9.2.9 RIMS:
IDESK Group: IBM-SCS-rims-SL2
Support group mail id: ikea4.nrim@ingka.com
Souradeep Bhattacharya (External - IBM India Pvt. Ltd)
Raghavendra Bhatta (External - IBM India Pvt. Ltd)

9.2.10 IKANO:

9.2.11 LIP2:
IDESK Group: GINT - Global Integration support Team
Support group mail id: dl.gint@ingka.com
Ajay Kumar Dubey (External - HCL Technologies Ltd)
Aarti Saini (External - HCL Technologies Ltd)

9.2.12 Invoice support team:
IDESK Group: GWRETAIL - Windows Server Retail

9.2.13 GLINUX:
IDESK Group: GLINUX - Global Linux support group

9.2.14 GNETWORK:
IDESK Group: GNETWORK - Network support

9.2.15 Local IT:
IDESK Group: Backline-LOCALIT-AT, Backline-LOCALIT-IT

9.2.16 EPS: Bedussi Paolo p.bedussi@netswgroup.it

9.2.17 Service Plus:
Support group mail id: support@serviceplus-it.com Helpdesk hot line: +385 91 3654 284

9.2.18 ICM:
IDESK Group: CAP-ECOM-ICM-SL1
SL1 Phone Number : +46(0) 72 88 64 884

9.2.19 Auth0: IDESK Group: GDNA - Global Support Team Directories
& Authentic
Escalation /Support Group Manager: Joakim Prahl
Escalation Number +46 723 52 74 47GDNA

9.2.20 API Connect Gateway: Group DL in iDesk:APIM - API Management

8.8. Third Parties Contact Information

Third party vendor name: DIEBOLD NIXDORF:
Support group mail id: support.ikea@wincor-nixdorf.com

8.9**Client Support**

9.4.1 Project Team IKEA:
Jip Koudijs: jip.koudijs@ingka.ikea.com
Henrik Wifvesj: henrik.wifvesjo@ingka.ikea.com
Juliann Veeger: juliann.veeger@ingka.ikea.com

8.10.**Infrastructure (Server/Network)**

1. GWLS:

IDESK Group: GWLS - Weblogic Support
Support group mail id: dl.gwls@ingka.com
Distribution List: GWLS
Jasneet Singh Dutta (External - HCL Technologies Ltd),
Prasana Manne (External - HCL Technologies Ltd),
Rekha Guvvakallu (External - HCL Technologies Ltd),
Pujitha Vura (External - HCL Technologies Ltd).

8.11. DBAs

IDESK Group: GDBA - Global Oracle DBA
Support group mail id: pudla.dbaseext@ingka.com; production.oracledba.se@ikea.com
Dinesh Adhikari (External - HCL Technologies Ltd),
Boopathy Ramamoorthy (External - HCL Technologies Ltd)
8.12.**Other contacts**

9.7.1 IIP/GINT:
IDESK Group: GINT - Global Integration support Team
Support group mail id: dl.gint@ingka.com
Ajay Kumar Dubey (External - HCL Technologies Ltd)
Aarti Saini (External - HCL Technologies Ltd)

9.**PROBLEM REPORTING**

9.1.Helpdesk details

IDESK Group: GIDESK - Global iDesk Support
Support group mail id: csan@ingka.com

9.2. Vendor Details

Third party vendor name: DIEBOLD NIXDORF:
Support group mail id: support.ikea@wincor-nixdorf.com

10.**APPLICATION HISTORY**

10.1 Typical Problems

10.2. Tips & Tricks

10.3. Application Release History

Release history and details are available in CLM:
https://clm.rat.itshost.se/ccm/web/projects/Solution%20Team%20Payment%20RS%20(Change%20Management)#action=com.ibm.team.workitem.runSavedQuery&id=_XP-dgFjeEeikXs2uXvKdaw
Naming Standards

10.4.**Batch**

10.5.**On-Line**

REFERENCE DOCUMENTS

10.6. Templates

10.7.**Application Profile _XXX.doc**

10.8.

10.9. Data Flow Diagram

10.10.**ER Diagram._xxxx**

10.11.**Infrastructure Diagram_XXX**

10.12.**Interface Specs - Legacy_XXX**

10.13.**Process Decomposition Table_XXXX**

10.14.

10.15.RunBook Template

10.16.

10.17. Server Specifications

PIP servers follow below format for IKE stores. For 4- and 5-digit Store Numbers PIP server format has been revised as part of PIP 19.5 Release. For more details refer JIRA ID: PAYM-2535
3 digits: RET @@ ### -lxNNNN.ikea.com
4 and 5 digits: R @@ ##### -lxNNNN.ikea.com
Where;@ is the country code
is the store Number
NNNN will be like 4010,4030 etc.

10.18.

10.19.**Ticket Log_XXX**

10.20.**New Store Setup**

Slovenia country new stores 933 and 528 (Slovenia Service office ~~933) set up done part of PIP 19.5 Release. For more details please refer JIRA ID~~ PAYM-1435

10.21. XXX_support_log

10.22.

10.23. XXXXX RACI

<<Please provide a list of additional reference documents>>
11.**IPOS team tasks**

Please see the excel sheet in which the Tasks performed by IPOS team is mentioned.

12.**GOOD TO KNOW**

12.1.1. ITF

ITF service endpoint URL:
https://confluence.build.ingka.ikea.com/display/ITF/ITF+WS+Proxy+Service+Endpoints#ITFWSProxyServiceEndpoints-ITFWSProxyinChina(andRussia)
Refer-ServiceEndpoints section

PIP has different SERVICE account for different environments.

PIP-PPE web proxy Service account: S-EBCPOS01-U-PPSEELM
PIP-PTE web proxy Service account: s-ebcpos01-u-ptseelm
PIP-CTEF web proxy Service account: s-ebcpos01-u-ikeadt
PIP-CTEF2 web proxy Service account: s-ebcpos01-u-ikeadt
PIP-PROD web proxy Service account: s-ebcpos01-u-global

In case the password is reset then it will throw exception as below:

com.sun.xml.ws.fault.ServerSOAPFaultException: Client received SOAP Fault from server: ERR error code=1013 os status=0, EBB ERROR: LDAP protocol error. Check configuration, This error will occur when either password is wrong or username does not **exist**.

12.1.2. Jenkins

http://itseelm-nt4018.ikea.com:8080/
New build server:
http://itseelm-nt4138.ikea.com:8080/

12.1.3. Nexus

http://itseelm-nt4696.ikea.com:8081/

12.1.4. New Service Office installation:

1. Copy the existing tendermapping file.
2. Edit the tendermapping file with the changes provided by TP.Net.
Check for ExternalId in PIP.
Example of File for RSSO installation which we have received from TP.Net:

3. Edit PIP.properties
e.g Item import - Local time of Serbia between 1 - 5 A.M.
4. Edit Stores.csv
5. Reports/profiles which needs to be configured.
poslog,sarecsales,outputmanagement,ibis,asis,bdsse,sarecfunds
6. Execute UpdateSalesTax BS to update the tax in EBCPOSRIX for RSSO
7. Go to RIX whether they have range for Serbia in test environment
8. Update the doc for PPE and send to AJPC via incident.
9. Test server ask Configuration Manager to deploy the latest PIP build with new properties
10. Get confirmation from PIP and MHS test team that store XXX is the test mhs.
11. When PIP needs to be installed on any new service office store , GWLS needs to update BUCode and MHS_HOST details in server file pip_env. properties and for this task should be raised to them .Since currently service office installation not taken care by PIP installation scripts.