[](http://upload.wikimedia.org/wikipedia/en/f/f1/Vfc-l.gif) 

**System Maintenance Technical Document**

**Version 1.0**

**Supply Planning: i2 Supply Chain Planner and PSE for Outdoor**

Version Control

|  |  |  |  |
| --- | --- | --- | --- |
| Version | Date | Author(s) | Description |
| 1.0 | 11/25/13 | Gopal Sadige  Digbijoy Das | Initial Draft |
|  |  |  |  |
|  |  |  |  |

Approval

|  |  |  |  |
| --- | --- | --- | --- |
| Name | Project | Email | Date |
|  |  |  |  |
|  |  |  |  |
|  |  |  |  |

**Table of Contents**

*[1 Introduction 5](#_Toc373183943)*

*[1.1 Overview 5](#_Toc373183944)*

*[1.2 Scope 5](#_Toc373183946)*

*[1.3 Component Owners 5](#_Toc373183947)*

*[2 System Details 6](#_Toc373183948)*

*[2.1 Architecture 6](#_Toc373183949)*

*[2.2 Business Processes 9](#_Toc373183951)*

*[2.3 Sub systems 11](#_Toc373183953)*

*[2.3.1 i2 Supply Chain Planner (SCP) 11](#_Toc373183954)*

*[2.3.1.1 Master Data 12](#_Toc373183956)*

*[2.3.1.2 Demand 17](#_Toc373183958)*

*[2.3.1.2.1 Unshipped Orders 17](#_Toc373183959)*

*[2.3.1.2.2 Contracts 19](#_Toc373183961)*

*[2.3.1.2.3 Retail PO, Retail OH and Distros 22](#_Toc373183962)*

*[2.3.1.2.4 Unbooked Forecast 24](#_Toc373183963)*

*[2.3.1.2.5 Retail Unshipped Orders 25](#_Toc373183965)*

*[2.3.1.2.6 Need To Buy (NTB) 28](#_Toc373183967)*

*[2.3.1.3 Supply 30](#_Toc373183968)*

*[2.3.1.3.1 On Hand Inventory 30](#_Toc373183970)*

*[2.3.1.3.2 POs, PRs and STOs 32](#_Toc373183971)*

*[2.3.1.3.3 Frozen Planned Orders 34](#_Toc373183972)*

*[2.3.1.4 Combine and Prioritize Demand 36](#_Toc373183973)*

*[2.3.1.5 ADW tables and files 37](#_Toc373183974)*

*[2.3.1.6 Planning logic 45](#_Toc373183976)*

*[2.3.1.7 Outbound feeds 46](#_Toc373183978)*

*[2.3.1.8 RDW tables 47](#_Toc373183979)*

*[2.3.2 PSE 48](#_Toc373183982)*

*[2.3.2.1 Routings 48](#_Toc373183984)*

*[2.3.2.2 Capacity 50](#_Toc373183985)*

*[2.3.2.3 Minimum 51](#_Toc373183987)*

*[2.3.2.4 53](#_Toc373183988)*

*[2.3.2.5 Demand Spreading 53](#_Toc373183989)*

*[2.3.2.6 Planned Orders 54](#_Toc373183990)*

*[2.3.2.7 Technical Details 56](#_Toc373183992)*

*[2.3.2.7.1 Application Modules / Framework Components 56](#_Toc373183993)*

*[2.3.2.7.2 System and Technical Architecture 56](#_Toc373183994)*

*[2.3.2.7.3 Important PSE tables 58](#_Toc373183995)*

*[2.4 Servers, Databases and Important directories 59](#_Toc373183996)*

*[2.5 ESP Batch Schedule 60](#_Toc373183998)*

*[3 Processes 61](#_Toc373184005)*

*[3.1 Configuration Management 61](#_Toc373184006)*

*[3.2 Support 61](#_Toc373184008)*

*[3.3 Communication 61](#_Toc373184010)*

*[3.4 Miscellaneous 61](#_Toc373184012)*

*[4 Known Issues & Defects 61](#_Toc373184013)*

*[5 Glossary and Acronyms 61](#_Toc373184018)*

*[6 References 62](#_Toc373184019)*

# Introduction

## Overview

This document discusses architecture, business processes, functionality and maintenance procedures of VF`s Supply Planning system that involves the sub-systems - i2 Supply Chain Planner (SCP) and Planning Systems Enhancements (PSE). This document will enable production support team to carry out support and development activities effectively. It would also act as training material for new team members. It will be an evolving document which would undergo revisions based on new changes added to the system in future.

## Scope

i2 and PSE applications for Outdoor have undergone a major change as part of project Acadia. The document focuses on Acadia version of i2 and PSE applications that cater to the Outdoor coalition only.

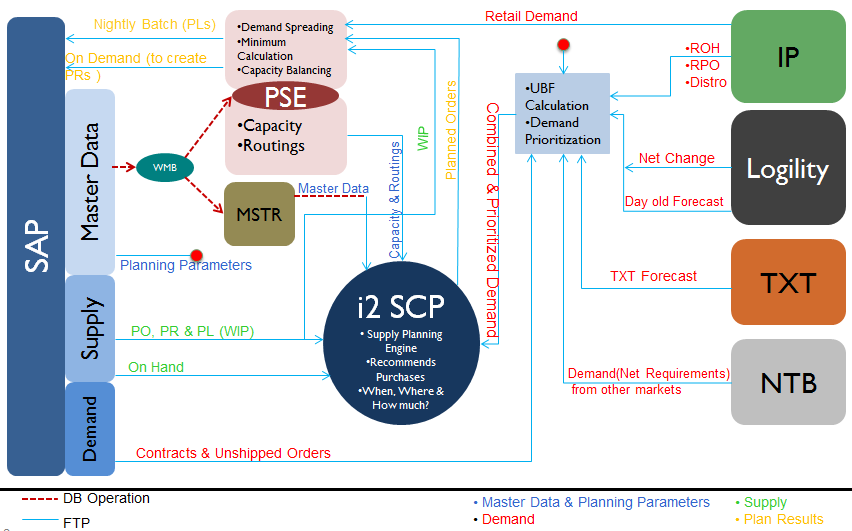
## Component Owners

|  |  |  |
| --- | --- | --- |
| **Component / Function / Module** | **Owner / SME** | **Email/phone** |
| i2 Supply Chain Planner and PSE | Gopal Sadige | [gopal\_sadige@vfc.com/](mailto:gopal_sadige@vfc.com/)  336-332-4213 |

# System Details

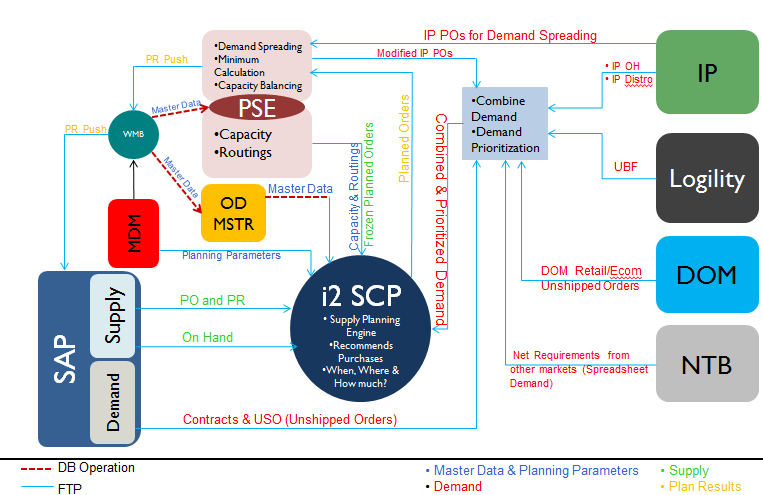
## Architecture

The following picture depicts detailed architecture, integration and data-flow of today`s Supply planning system. Connection points mentioned in red are demand elements, green are supply elements, blue are master data/planning parameters, amber are outbound elements from i2 SCP and PSE.



**Fig 2.1.1: Supply Planning system architecture (As-Is)**

The following picture depicts detailed architecture, integration and data-flow of Supply Planning system post implementation of project Acadia. Connection points mentioned in red are demand elements, green are supply elements, blue are master data/planning parameters, amber are outbound elements from i2 SCP and PSE.



**Fig 2.1.2: Supply Planning system architecture (post project Acadia)**

The details of the Supply, Demand & Master Data are mentioned below:

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **Data element** | **Type** | **Source** | **Target** | **Integration Type** |
| Inventory Policy | Planning Parameters | MDM | i2 | Hub and Spoke |
| Capacity Groups | Planning Parameters | MDM | i2 | Check table service/FTE |
| Planning Increments | Planning Parameters | MDM | i2 | Item Master Service->ODMSTR DB |
| Material Attribute | Master Data | MDM | i2 | Item Master Service->ODMSTR DB |
| Material Attribute Master | Master Data | MDM | i2 | Item Master Service->ODMSTR DB |
| Plant Attribute Master | Master Data | MDM | i2 | Vendor Master Service->ODMSTR DB |
| Product Size Master | Master Data | MDM | i2 | Item Master Service->ODMSTR DB |
| Lot Mfg Unit Master | Master Data | MDM | i2 | Item Master Service->ODMSTR DB |
| Style Color Attribute | Master Data | MDM | i2 | Item Master Service->ODMSTR DB |
| Product All Season Master | Master Data | MDM | i2 | Item Master Service->ODMSTR DB |
| Product Inventory Class Master | Master Data | MDM | i2 | Item Master Service->ODMSTR DB |
| Product Current Future Season Master | Master Data | MDM | i2 | Item Master Service->ODMSTR DB |
| Customer Attribute Master | Master Data | MDM | i2 | Customer Master Service->ODMSTR DB |
| Capacity | Master Data | PSE | i2 | Internal |
| Consumption | Master Data | PSE | i2 | Internal |
| Planning Routings | Master Data | PSE | i2 | Internal |
| Unbooked Forecast | Demand | Logility | i2 | FTE |
| Retail Unshipped Orders | Demand | DOM | i2 | Hub and Spoke |
| Unshipped Orders | Demand | SAP | i2 | FTE |
| Hard Contracts | Demand | SAP | i2 | FTE |
| Retail Purchase Orders (RPO) | Demand | IP | i2 | FTE |
| Retail Distro Orders (RDO) | Demand | IP | i2 | FTE |
| Retail On Hand (ROH) | Demand | IP | i2 | FTE |
| Need to Buy (NTB) Spreadsheet Demand | Demand | Other Markets | i2 | Upload through a portal |
| On Hand Inventory | Supply | SAP | i2 | FTE |
| Purchase Orders | Supply | SAP | i2 | FTE |
| Purchase Requistions | Supply | SAP | i2 | FTE |
| Frozen Planned Orders | Supply | PSE | i2 | FTE |
| Retail Purchase Orders (RPO) | Demand | IP | i2 | FTE |
| Retail Distro Orders (RDO) | Demand | IP | i2 | FTE |
| Retail On Hand (ROH) | Demand | IP | i2 | FTE |
| Material Master Data | Master Data | MDM | PSE | Item Master Service -> PSE DB |
| Plant Master Data | Master Data | MDM | PSE | Vendor Master Service -> PSE DB |

Note:

All the integration types are used to extract data from MDM tables.

Hub and Spoke:

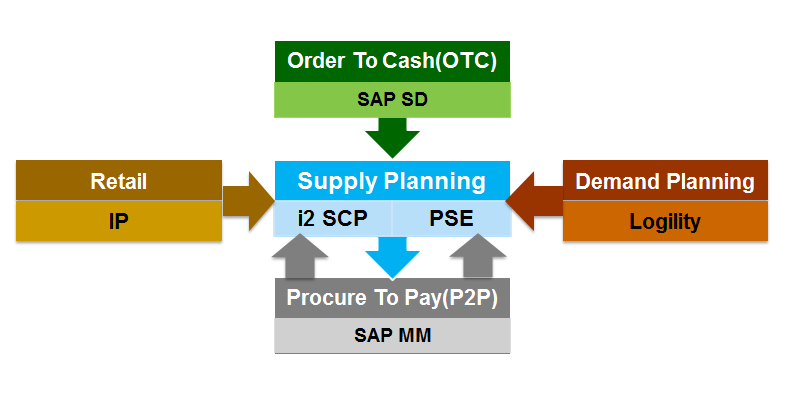
This integration type is used to extract data from MDM tables and transform the data and puts it in the file

Check Table Service:

This is a straight extract from MDM tables to files.

## Business Processes

From VF Corp`s business processes standpoint, Supply Planning is mainly integrated with *Order To Cash (OTC), Demand Planning, Retail* and *Procure To Pay (P2P)*. Business processes are explained in the context of Outdoor coalition, which is largely a purchasing model.



**Fig 2.2.1: Outdoor Coalition – Business Processes**

A brief description of these processes and business process flow involving them is as given below. It is explained in the context of VF Corp.

**Order To Cash (OTC):** A business process that starts at receiving a sales order from customer and ends with the cash received by customer. As part of this process, the following sub processes are executed:

* Create sales orders
* Perform ATP checks
* Allocate stock

SAP SD is used to perform the above mentioned sub-processes that are part of

OTC.

**Procure To Pay (P2P):** A business process that starts at finalizing the procurement requirement and ends at payment to the vendor. As part of this process, the following sub processes are executed:

* Create Purchase Requisitions (PRs)
* Make sourcing decisions
* Generate Purchase Orders (POs)

SAP MM is used to perform the above mentioned sub-processes that are part of P2P.

**Demand Planning:** Demand Planning is a process of arriving at a reliable forecast for future demand. In this process,

* A base line forecast is generated based on previous sales history using statistical forecasting models.
* The forecast is then adjusted by Sales and Marketing inputs before a final Demand Plan is generated.
* Demand plan will be then sent as an input to Supply Planning process.

Logility Demand Planning solution is used to perform the afore mentioned sub-processes that are part of Demand Planning

**Retail:** A Business process that deals with sub-processes that are related to the retail customers. Island Pacific houses variety of retail data – the Retail Purchase Orders and Retail On hand.

**Supply Planning:** A process of generating a feasible master plan to meet various

demand elements. Master plan can consist of a procurement plan or a production

plan based on the supply network of an organization. These plans can then be

taken up by MRP systems or Sourcing systems for further planning or execution.

* At VF Corp, i2 Supply Chain Planner (SCP) is used as a supply planning

engine

* OTC process feeds in Unshipped Sales Orders and Sales Contracts to Supply Planning process
* Supply planning process receives latest On Hand Inventory, Open Purchase Requisition(PR)s and Open Purchase Order(PO)s from P2P process
* Supply planning process gets Retail POs from Retail processes
* Demand planning sends forecast data to Supply planning process
* i2 SCP processes demands (Unshipped orders, Contracts, Retail POs and Forecasted demand) and generates a feasible supply plan based on the latest supply picture. Since VF Corp`s Outdoor coalition is a purchasing model, i2 SCP is modeled to generate Planned Orders
* Planned orders are then sent to PSE for further buy analysis and capacity balancing. Business users can perform various activities on the planned orders – combine, split, change vendors and dates. After some/all of these activities are performed based on the business requirements, finalized planned orders would be pushed to P2P process, where the planned orders get converted to Purchase Requisitions and eventually Purchase Orders

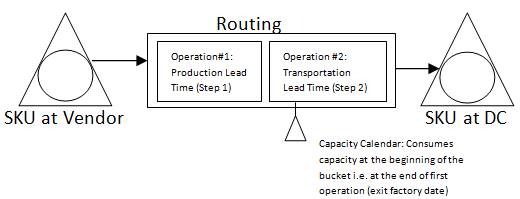
## Sub systems

### *i2 Supply Chain Planner (SCP)*

**Key characteristics of i2 SCP for Outdoor coalition:**

* i2 Supply Chain Planner has been modeled to recommend additional purchases and can produce a constraint based plan. It is a JIT model that focuses only on purchasing of finished good material.
* Planning unit is “SKU-DC” i.e. Planning is done at the at the individual SKU level to account for any size level imbalances.
* The i2 Supply Chain Planning System is updated and re-run nightly, so it reacts quickly to changes in supply or demand.
* Versions of i2 SCP engine and model are 6.3.2 and 5.2 respectively. The model is based on CDM architecture.
* i2 SCP`s MAP solver is implemented to do the Order By Order planning.

**Supply Chain Model for Outdoor:**



**Various components that are required to run i2 SCP:** 

#### Master Data

MSTR is the source of master data for i2 in today`s production. With MSTR

being decommissioned for Outdoor, a new data model (I2ODMSTR) has been

designed as part of Acadia. I2ODMSTR serves as a source master data for i2

going forward. It hosts the data related to Material, Plants, Vendors, Planning

Parameters and Costing. The ultimate source of these tables is MDM. Various

services in integration (WMB) would update/inserts data in these tables when

changes occur in MDM.

Following chart shows the tables in I2ODMSTR and their structure:

|  |  |  |  |
| --- | --- | --- | --- |
| **Table** | **Description** | **Structure** | **Updated by (Integration Service)** |
| MDM\_COLOR | Color values | COLOR\_CD VARCHAR2(8 BYTE) LONG\_COLOR\_NM VARCHAR2(40 BYTE) COLOR\_GROUP\_CD VARCHAR2(3 BYTE)  **Primary Key:** COLOR\_CD | Item Master |
| MDM\_PRODUCT\_TYPE | Product Type information | PRODUCT\_TYPE\_CD VARCHAR2(3 BYTE) PRODUCT\_TYPE\_NM VARCHAR2(40 BYTE) PROFIT\_CENTER\_CD VARCHAR2(10 BYTE)  **Primary Key:** PRODUCT\_TYPE\_CD | Item Master |
| MDM\_PRODUCT | All materials and their attributes | PRODUCT\_NBR VARCHAR2(18 BYTE) PRODUCT\_DES VARCHAR2(40 BYTE) STYLE\_NBR VARCHAR2(12 BYTE) COLOR\_CD VARCHAR2(8 BYTE) COLOR\_GROUP\_CD VARCHAR2(3 BYTE) MFG\_UNIT\_CD VARCHAR2(8 BYTE) MFG\_UNIT\_DESC VARCHAR2(30 BYTE) SBU\_CD VARCHAR2(3 BYTE) SBU\_NM VARCHAR2(30 BYTE) SUB\_SBU\_CD VARCHAR2(3 BYTE) SUB\_SBU\_NM VARCHAR2(30 BYTE) PRODUCT\_ROLE\_CD VARCHAR2(3 BYTE) SUB\_DIV\_NBR VARCHAR2(3 BYTE) AGE\_GROUP\_CD VARCHAR2(2 BYTE) MODEL\_CD VARCHAR2(5 BYTE) PRODUCT\_TYPE\_CD VARCHAR2(3 BYTE) MATERIAL\_TYPE\_CD VARCHAR2(4 BYTE) PACKAGE\_IND VARCHAR2(1 BYTE) OLD\_MATERIAL\_NBR VARCHAR2(18 BYTE)  **Primary Key:** PRODUCT\_NBR **Foreign Key:** COLOR\_CD->MDM\_COLOR::COLOR\_CD **Foreign Key:** PRODUCT\_TYPE\_CD->MDM\_PRODUCT\_TYPE::PRODUCT\_TYPE\_CD | Item Master |
| MDM\_PLANT | All plants (Vendors & DCs) | PLANT\_NBR VARCHAR2(6 BYTE) PLANT\_NM VARCHAR2(40 BYTE) PLANT\_ST\_LINE1\_ADR VARCHAR2(60 BYTE) PLANT\_CITY\_NM VARCHAR2(40 BYTE) PLANT\_ST\_PROV\_CD VARCHAR2(3 BYTE) PLANT\_POSTAL\_CD VARCHAR2(10 BYTE) PLANT\_COUNTRY\_CD VARCHAR2(3 BYTE)  **Primary Key:** PLANT\_NBR | Vendor Master. DCs are loaded manually (one-time) |
| MDM\_ PLANT\_INFO | Information related to Vendors & DCs | PLANT\_NBR VARCHAR2(6 BYTE) VENDOR\_NBR VARCHAR2(10 BYTE) LOCATION\_STAT\_CD VARCHAR2(2 BYTE) LOCATION\_CLASS\_CD VARCHAR2(2 BYTE) LOCATION\_CAT\_CD VARCHAR2(10 BYTE) COMPANY\_CD VARCHAR2(4 BYTE) NEED\_TO\_BUY\_IND VARCHAR2(1 BYTE)  **Primary Key:** PLANT\_NBR **Foreign Key:** PLANT\_NBR->MDM\_PLANT::PLANT\_NBR | Vendor Master. DCs are loaded manually (one-time) |
| MDM\_DELV\_PROG\_ASSN | Sales Orgs, Delivery Programs and their effectivity | SALES\_ORG\_CD VARCHAR2(4 BYTE) ORDER\_FROM\_DT DATE ORDER\_THRU\_DT DATE DELV\_FROM\_DT DATE DELV\_THRU\_DT DATE DELV\_PROGRAM\_CD VARCHAR2(3 BYTE) COLLECTION\_CD VARCHAR2(4 BYTE) THEME\_CD VARCHAR2(4 BYTE) | Item Master |
| MDM\_PRODUCT\_DELV\_PROG | Delivery program details | PRODUCT\_NBR VARCHAR2(18 BYTE) DELV\_PROGRAM\_CD VARCHAR2(3 BYTE) COLLECTION\_CD VARCHAR2(4 BYTE) THEME\_CD VARCHAR2(4 BYTE)  **Foreign Key:** PRODUCT\_NBR->MDM\_PRODUCT::PRODUCT\_NBR | Item Master |
| MDM\_RPT\_SEASON | Reporting Seasons | RPT\_SEASON\_CD VARCHAR2(4 BYTE) RPT\_SEASON\_YR VARCHAR2(4 BYTE) RPT\_SEASON\_NM VARCHAR2(20 BYTE)  **Primary Key:** RPT\_SEASON\_CD | Item Master |
| MDM\_PRODUCT\_RPT\_SEASON | Valid Product + Season combination | PRODUCT\_NBR VARCHAR2(18 BYTE) RPT\_SEASON\_CD VARCHAR2(4 BYTE) INVT\_CLASS\_CD VARCHAR2(3 BYTE) **Primary Key:** PRODUCT\_NBR,RPT\_SEASON\_CD | Item Master |
| MDM\_PRODUCT\_SALES\_AREA | Valid Product+Sales Org combinations | PRODUCT\_NBR VARCHAR2(18 BYTE) SALES\_ORG\_CD VARCHAR2(4 BYTE)  **Primary Key:** PRODUCT\_NBR,SALES\_ORG\_CD **Foreign Key:** PRODUCT\_NBR->MDM\_PRODUCT::PRODUCT\_NBR | Item Master |
| MDM\_PRODUCT\_SIZE | Products and their sizes | PRODUCT\_GRID\_NBR VARCHAR2(18 BYTE) DIM1 VARCHAR2(8 BYTE) DIM2 VARCHAR2(8 BYTE) DIM3 VARCHAR2(8 BYTE) SIZE\_GRID\_TXT VARCHAR2(25 BYTE)  **Foreign Key:** PRODUCT\_GRID\_NBR->MDM\_PRODUCT::PRODUCT\_NBR | Item Master |
| MDM\_PRODUCT\_SKU\_COST | Different costs at StyleColor/Size/DC level | PRODUCT\_NBR VARCHAR2(18 BYTE) SIZE\_GRID\_TXT VARCHAR2(25 BYTE) PLANT\_NBR VARCHAR2(4 BYTE) INITIAL\_COST\_AMT NUMBER(11,2) RELEASED\_COST\_AMT NUMBER(11,2) CURRENCY\_CD VARCHAR2(5 BYTE) LAST\_COST\_CHANGE\_DT DATE PREFERRED\_SKU\_IND VARCHAR2(1 BYTE) VALUATION\_TYPE\_CD VARCHAR2(10 BYTE) | Item Master |
| MDM\_PROD\_SKU\_RTL\_PRICE | Retail price at StyleColor/Size level | PRODUCT\_NBR VARCHAR2(18 BYTE) SIZE\_GRID\_TXT VARCHAR2(25 BYTE) COUNTRY\_CD VARCHAR2(4 BYTE) CURRENCY\_CD VARCHAR2(5 BYTE) RPT\_SEASON\_CD VARCHAR2(3 BYTE) MFG\_SUG\_RTL\_AMT NUMBER(9,2)  **Primary Key:** PRODUCT\_NBR,SIZE\_GRID\_TXT,COUNTRY\_CD,CURRENCY\_CD and RPT\_SEASON\_CD **Foreign Key:** PRODUCT\_NBR->MDM\_PRODUCT::PRODUCT\_NBR | Item Master |
| MDM\_PROD\_SKU\_WHSL\_PRICE | Wholesale price at StyleColor/Size level | COMPANY\_CD VARCHAR2(4 BYTE) PRODUCT\_NBR VARCHAR2(18 BYTE) SIZE\_GRID\_TXT VARCHAR2(25 BYTE) DELV\_PROGRAM\_CD VARCHAR2(3 BYTE) WHLS\_PRICE\_AMT NUMBER(11,2) CURRENCY\_CD VARCHAR2(5 BYTE) PRICE\_QTY NUMBER(5,0) PRICE\_UOM VARCHAR2(3 BYTE) PRICE\_DATE\_CD VARCHAR2(1 BYTE) PRICING\_DT DATE  **Primary Key:** COMPANY\_CD,PRODUCT\_NBR,SIZE\_GRID\_TXT **Foreign Key:** PRODUCT\_NBR->MDM\_PRODUCT::PRODUCT\_NBR | Item Master |
| MDM\_PLANNING\_INCREMENTS | Planning Parameters at StyleColor/Size/DC level- esp the min, mul, max & safety stock qtys | STYLE\_COLOR VARCHAR2(18 BYTE) DC VARCHAR2(4 BYTE) DIM1 VARCHAR2(8 BYTE) DIM2 VARCHAR2(8 BYTE) DIM3 VARCHAR2(8 BYTE) SIZE\_GROUP VARCHAR2(5 BYTE) SKU\_STATUS VARCHAR2(2 BYTE) INV\_POLICY VARCHAR2(8 BYTE) CAPACITY\_GROUP VARCHAR2(8 BYTE) MIN\_QTY NUMBER(5,0) MUL\_QTY NUMBER(5,0) MAX\_QTY NUMBER(5,0) SS\_QTY NUMBER(5,0) SIZE\_PCT NUMBER(5,0)  **Primary Key:** STYLE\_COLOR,DIM1,DIM2,DC **Foreign Key:** DC->MDM\_PLANT::PLANT\_NBR **Foreign Key:** STYLE\_COLOR->MDM\_PRODUCT::PRODUCT\_NBR | Item Master |
| MDM\_CUSTOMER | Complete list of Sold to customer | CUST\_NBR VARCHAR2(10 BYTE) CUST\_NM VARCHAR2(35 BYTE)  **Primary Key:** CUST\_NBR | Customer Master |

As part of nightly batch, master data extraction programs are run. They extract data from the afore-mentioned tables and generate flat files. These files will be used by i2 funnel programs to perform master data checks. Following are the list of shell scripts that are run as part of master data extraction. File that each of the script generates is also listed:

* i2od\_matl\_attrb\_extr.sh
  + $I2OD\_FROM\_MSTR/MATERIAL\_ATTRIBUTES\_MSTR.dat
* i2od\_product\_size\_extr.sh
  + $I2OD\_FROM\_MSTR/PRODUCT\_SIZE\_MSTR.dat
* i2od\_lot\_mfg\_unit\_extr.sh
  + $I2OD\_FROM\_MSTR/LOT\_MFG\_UNIT\_MSTR.dat
* i2od\_product\_sku\_cost\_extr.sh
  + $I2OD\_FROM\_MSTR/PRODUCT\_SKU\_COST.dat
* i2od\_prod\_sku\_rtl\_price\_extract.sh
  + $I2OD\_FROM\_MSTR/SKU\_RETAIL.dat
* i2od\_prod\_sku\_rtl\_price\_extract\_unique.sh
  + $I2OD\_FROM\_MSTR/SKU\_RETAIL\_unique.dat
* i2od\_prod\_sku\_whls\_price\_extract.sh
  + $I2OD\_FROM\_MSTR/SKU\_WHLS.dat
* i2od\_rdw\_all\_product\_season\_mstr.sh
  + $I2OD\_FROM\_MSTR/product\_all\_season\_master.dat
* i2od\_rdw\_product\_season\_invt\_class.sh
  + $I2OD\_FROM\_MSTR/product\_season\_invt\_class\_master.dat
* i2od\_current\_future\_season\_mstr.sh
  + $I2OD\_FROM\_MSTR/product\_current\_future\_season\_master.dat
* i2od\_rdw\_style\_color\_attrb\_ext.sh
  + $I2OD\_FROM\_MSTR/style\_color\_attrb.dat
* i2od\_size\_percent\_extr.sh
  + $I2OD\_FROM\_SAP/i2odsa\_size\_percent.dat
* i2od\_customer\_extr.sh
  + $I2OD\_FROM\_MSTR/CUSTOMER\_EXTR.dat
* i2od\_plant\_company\_mstr\_extr.sh
  + $I2OD\_FROM\_MSTR/PLANT\_COMPANY\_MSTR.dat
* i2od\_plant\_attrb\_mstr\_extr.sh
  + $I2OD\_FROM\_MSTR/PLANT\_ATTRB\_MSTR.dat
* i2od\_style\_color\_season\_extr.sh
  + $I2OD\_FROM\_MSTR/style\_color\_season.dat

#### Demand

##### Unshipped Orders

Unshipped Orders feed to i2 carries the Unshipped Sales Orders (USO) and Unshipped Allocated Sales Orders (USA), requested by various customers. Source of this feed is SAP. This is maintained at a SKU and DC level. Unshipped Orders and Unshipped Allocated Orders are tagged as USO & USA respectively in DEMAND\_ALL table. Layout, location, file name of this feed are described below:

* Location: $I2OD\_FROM\_SAP
* File Name: i2ods7\_unshipped\_orders.dat
* Layout of the detail record:

|  |  |  |  |
| --- | --- | --- | --- |
| **Order** | **Description** | **Type** | **Comments** |
| 1 | Record Type ('1' for Details and '0' for Header) | Char(1) |  |
| 2 | Owner Code ('W' for JW and 'O' for OD) | Char(2) | Will be left blank |
| 3 | Sales Org | Char(4) |  |
| 4 | Distribution Channel | Char(2) |  |
| 5 | Division | Char(2) |  |
| 6 | Company Code | Char(4) |  |
| 7 | Sold to customer number | Char(10) |  |
| 8 | Country Code | Char(3) |  |
| 9 | Account Class | Char(4) | Will be left blank |
| 10 | Shipping Location | Char(4) |  |
| 11 | Material | Char(18) |  |
| 12 | Base Grid Value | Char(8) |  |
| 13 | Dim1 (Size) | Char(8) |  |
| 14 | Dim2 | Char(8) |  |
| 15 | Dim3 | Char(8) |  |
| 16 | Confirm Date | Char(8) |  |
| 17 | Demand Type Indicator | Char(1) | Default “H” |
| 18 | Sales UOM | Char(3) |  |
| 19 | Base UOM | Char(3) |  |
| 20 | Open Qty | Number(14) |  |
| 21 | Sign1 | Char(1) | Default “+” |
| 22 | Allocated Qty | Number(14) |  |
| 23 | Sign2 | Char(1) | Default “+” |
| 24 | Delivered Qty | Number(14) |  |
| 25 | Sign3 | Char(1) | Default “+” |
| 26 | Style color | Char(18) |  |
| 27 | Document Category | Char(1) |  |
| 28 | Sales Document Type | Char(4) |  |
| 29 | Sales Order Number | Char(10) |  |
| 30 | Sales Doc Line Item | Char(6) |  |
| 31 | Sales Doc Schedule Line | Char(4) |  |
| 32 | Ship To Customer Number | Char(10) |  |
| 33 | Item Category | Char(4) |  |
| 34 | Contract Document Number | Char(10) |  |
| 35 | Contract Document Line Item | Char(6) |  |
| 36 | Sales Season | Char(3) |  |
| 37 | Sales Collection | Char(4) |  |
| 38 | Sales Theme | Char(4) |  |
| 39 | Creation Date | Char(8) |  |
| 40 | Last Change Date | Char(8) |  |
| 41 | Last Change By | Char(12) |  |
| 42 | Required Delivery Date | Char(8) |  |
| 43 | Cancel Date | Char(8) |  |
| 44 | Customer PO Number | Char(20) |  |
| 45 | Delivery Block | Char(2) |  |
| 46 | Delivery Status | Char(1) |  |
| 47 | Sales Rep ID | Char(10) | Will be Char(10) field as opposed to current Char(8) field |
| 48 | Sales Rep Name | Char(40) |  |
| 49 | Cust Rep Id | Char(10) | Will be Char(10) field as opposed to current Char(8) field |
| 50 | Cust Rep Name | Char(40) |  |
| 51 | Requirement Category | Char(16) |  |
| 52 | Stock Category | Char(16) | This needs to be considered as the “Category” in i2 & PSE |
| 53 | Filler | Char(51) |  |

* Header data would be in the following format:

Layout:

Char(1) => Record Type ("0")

Char(8) => SAP program number

Char(8) => Date

Char(6) => Time

Example: 0YSDO202220130306032540

* Control (trailer) record data would be in the following format:

Layout:

Char(1) => Record Type ("9")

Char(8) => Date

Char(6) => Time

Char(20) => Records Count

Char(19) => Sum of quantities

Char(1) => Sign ("+")

Example: 920130306032540000000000000036744370000000000014226524+

##### Contracts

Contracts feed to i2 carries the future hard projections (HPR) from SAP. This is maintained at a SKU and DC level. Layout, location, file name of this feed are described below:

* Location: $I2OD\_FROM\_SAP
* File Name: i2ods8\_contracts.dat
* Layout of the detail record:

|  |  |  |  |
| --- | --- | --- | --- |
| **Order** | **Description** | **Type** | **Comments** |
| 1 | Record Type ('1' for Details and '0' for Header, 9 for Control record) | Char(1) |  |
| 2 | Order Number | Char(10) |  |
| 3 | Sales Doc Number | Char(6) |  |
| 4 | Schedule Line | Char(4) |  |
| 5 | Sales Org | Char(4) |  |
| 6 | Company Code | Char(4) |  |
| 7 | Owner Code ('W' for JW and 'O' for OD) | Char(1) | Will be left blank |
| 8 | Distribution Channel | Char(2) |  |
| 9 | Sales Document Type | Char(4) |  |
| 10 | Sold to customer number | Char(10) |  |
| 11 | Ship to customer number | Char(10) |  |
| 12 | Customer Account Class | Char(4) | Will be left blank |
| 13 | Cust Country Code | Char(3) |  |
| 14 | Style color | Char(18) |  |
| 15 | Shipping Location | Char(4) |  |
| 16 | Contract Type ('H' for Hard and 'S' for Soft) | Char(1) | Always “H” for Outdoor & Action Sports |
| 17 | Micro Slices | Number(2) | Filler: Custom for Jeanswear |
| 18 | Build Ahead Days | Number(3) | Filler: Custom for Jeanswear |
| 19 | Build Late Days | Number(3) | Filler: Custom for Jeanswear |
| 20 | Contract Open Qty | Number(14) |  |
| 21 | Sign | Char(1) |  |
| 22 | Schedule Ship Date (Year -YYYY) | Char(4) |  |
| 23 | Schedule Ship Date (Month -MM) | Char(2) |  |
| 24 | Schedule Ship Date (Day -DD) | Char(2) |  |
| 25 | Base Grid Value | Char(8) |  |
| 26 | Allocated Status | Char(1) |  |
| 27 | Dim1 (Size) | Char(8) |  |
| 28 | Dim2 | Char(8) |  |
| 29 | Dim3 | Char(8) |  |
| 30 | Respread Size Percentage | Number(4.2) | Filler: Custom for Jeanswear |
| 31 | Allocated Qty | Number(14) |  |
| 32 | Sign2 | Char(1) |  |
| 33 | Delivery Program | Char(3) |  |
| 34 | Delivery Collection | Char(4) |  |
| 35 | Creation Date | Char(8) |  |
| 36 | Last Change Date | Char(8) |  |
| 37 | Last Change By | Char(12) |  |
| 38 | Required Delivery Date | Char(8) |  |
| 39 | Cancel Date | Char(8) |  |
| 40 | Customer PO Number | Char(20) |  |
| 41 | Delivery Block | Char(2) |  |
| 42 | Delivery Status | Char(1) |  |
| 43 | Sales Rep ID | Char(10) | Sales Rep Id will be Char(10) |
| 44 | Sales Rep Name | Char(40) |  |
| 45 | Cust Rep Id | Char(10) | Cust Rep Id will be Char(10) |
| 46 | Cust Rep Name | Char(40) |  |
| 47 | Requirement Category | Char(16) |  |
| 48 | Stock Category | Char(16) | Category that needs to go into DEMAND\_ALL file for contract orders |
| 49 | Filler | Char(14) |  |

* Header data would be in the following format:

Layout:

Char(1) => Record Type ("0")

Char(8) => SAP program number

Char(8) => Date

Char(6) => Timestamp

Example: 0YSDI\_CON20130410035327

* Control (trailer) record data would be in the following format:

Layout:

Char(1) => Record Type ("9")

Char(8) => SAP Program Number

Char(8) => Date

Char(8) => Timestamp

Char(14) => Records Count

Char(1) => Sign ("+")

Char(14) => Sum of quantities

Char(1) => Sign ("+")

Example:

9YSDI\_CON2013041003532700000000248197+00000009443479+

##### Retail PO, Retail OH and Distros

Island Pacific (IP) retail system sends three types of demands to i2. They are:

Retail Purchase Orders (RPO): All open PO for Retail are sent as RPOs. Layout, location, file name of this feed are described below:

* Location: $I2OD\_FROM\_IP
* File Names:
  + IP\_PO\_vla.dat
  + IP\_PO\_LUCY.dat
  + IP\_PO\_vans.dat
  + IP\_PO\_VANS\_CAN.dat
  + IP\_PO\_tnf.dat
  + IP\_PO\_TNF\_CAN.dat
  + IP\_PO\_TBL.dat
* Layout of the detail record:

|  |  |  |
| --- | --- | --- |
| **Order** | **Field** | **Length** |
| 1. | IP PO | Char(20) |
| 2. | SAP Customer | Char(10) |
| 3. | Required Delivery Date | Char(8) |
| 4. | SAP Plant | Char(4) |
| 5. | SAP Sales Org | Char(4) |
| 6. | Material (Style Color) | Char(18) |
| 7. | Size1 | Char(6) |
| 8. | Size2 | Char(6) |
| 9. | UPC | Char(12) |
| 10. | Order Qty | Char(7) |
| 11. | Total Received Qty | Char(7) |
| 12. | Totap Open Qty | Char(7) |
| 13. | IP Demand Create Date | Char(8) |
| 14. | IP Demand Change Date | Char(8) |
| 15. | IP Demand Cancel Date | Char(8) |
| 16 | Stock Category | Char(16) |

Retail On Hand (ROH): Retail On Hand position. Layout, location, file name of this feed are described below:

* Location: $I2OD\_FROM\_IP
* File Name:
  + IP\_ONHAND\_vans.dat
  + IP\_ONHAND\_VANS\_CAN.dat
  + IP\_ONHAND\_tnf.dat
  + IP\_ONHAND\_vla.dat
  + IP\_ONHAND\_TNF\_CAN.dat
  + IP\_ONHAND\_LUCY.dat
  + IP\_ ONHAND \_TBL.dat
* Layout of the detail record:

|  |  |  |
| --- | --- | --- |
| **Order** | **Field** | **Length** |
| 1. | Plant | Char(4) |
| 2. | Sale Org | Char(4) |
| 3. | Material(Style color) | Char(18) |
| 4. | Size1 | Char(6) |
| 5. | Size2 | Char(6) |
| 6. | UPC | Char(12) |
| 7. | On Hand Qty | Char(7) |
| 8. | SAP Customer Number | Char(10) |

Retail Distros (RDO): Distros released after the last interface and Not released distros will be sent to i2 SCP as retail demand. Layout, location, file name of this feed are described below:

* Location: $I2OD\_FROM\_IP
* File Name:
  + IP\_DISTRO\_vla.dat
  + IP\_DISTRO\_vans.dat
  + IP\_DISTRO\_TNF\_CAN.dat
  + IP\_DISTRO\_LUCY.dat
  + IP\_DISTRO\_tnf.dat
  + IP\_DISTRO\_VANS\_CAN.dat
  + IP\_DISTRO\_VANS\_CAN.dat
  + IP\_DISTRO\_TBL.dat
* Layout of the detail record:

|  |  |  |
| --- | --- | --- |
| **S.No.** | **Field** | **Length** |
| 1. | Plant | Char(4) |
| 2. | Sale Org | Char(4) |
| 3. | Order Type | Char(3) |
| 4. | Material(Style color) | Char(18) |
| 5. | Size1 | Char(6) |
| 6. | Size2 | Char(6) |
| 7. | UPC | Char(12) |
| 8. | Total Distro Qty | Char(7) |
| 9. | SAP Customer Number | Char(10) |

##### Unbooked Forecast

Unbooked Forecast is the output Logility Demand Planning solution. It is netted

forecast that is maintained at Stylecolor/Size/DC/Week level. Layout, location,

file name of this feed is described below. The assumption is that the fields

which are highlighted in “green” will be sent to i2 SCP and rest all will be sent

as “blank” values. Unbooked Forecast would be tagged as “UBF” in

DEMAND\_ALL table. Layout, location, file name of this feed are described below:

* Location: $I2OD\_FROM\_LOGILITY
* File Name: i2ods7\_LOGILITY\_UBF\_orders.dat
* Layout of the detail record:

|  |  |  |
| --- | --- | --- |
| **Field Description** | **Field Length** | **Comments** |
| **Record Type** | **Char(1)** | “1” for detail record |
| **Owner Code** | **Char(2)** | Default to  “O” |
| Sales Org | Char(4) |  |
| Distribution Channel | Char(2) |  |
| Division | Char(2) |  |
| Company Code | Char(4) |  |
| Sold to Customer Number | Char(10) |  |
| Country Code | Char(3) |  |
| Account Class | Char(4) |  |
| **Shipping Location** | **Char(4)** | DC |
| **Material** | **Char(18)** | Style Color |
| Base Grid Value | Char(8) |  |
| **Dim 1** | **Char(8)** | Size 1 |
| **Dim 2** | **Char(8)** | Size 2 |
| Dim 3 | Char(8) |  |
| **Confirm Date** | **Char(8)** | Fcst Date |
| **Demand Type Indicator** | **Char(1)** | Default value "F" |
| Sales UOM | Char(3) |  |
| **Base UOM** | **Char(3)** |  |
| **Open Qty** | **Char(14)** | Fcst qty |
| Sign1 | Char(1) | Default sign "+" |
| **Allocated Qty** | **Char(14)** | All Zeros |
| Sign2 | Char(1) | Default sign "+" |
| Delivered Qty | Char(14) |  |
| Sign3 | Char(1) | Default sign "+" or leave blank |
| Style color | Char(18) |  |
| Document Category | Char(1) |  |
| Sales Document Type | Char(4) |  |
| Sales Order Number | Char(10) |  |
| Sales Doc Line Item | Char(6) |  |
| Sales Doc Schedule Line | Char(4) |  |
| Ship To Customer Number | Char(10) |  |
| Item Category | Char(4) |  |
| Contract Document Number | Char(10) |  |
| Contract Document Line Item | Char(6) |  |
| Sales Season | Char(3) |  |
| Sales Collection | Char(4) |  |
| Sales Theme | Char(4) |  |
| Creation Date | Char(8) |  |
| Last Change Date | Char(8) |  |
| Last Change By | Char(12) |  |
| Required Delivery Date | Char(8) |  |
| Cancel Date | Char(8) |  |
| Customer PO Number | Char(20) |  |
| Delivery Block | Char(2) |  |
| Delivery Status | Char(1) |  |
| Sales Rep ID | Char(10) |  |
| Sales Rep Name | Char(40) |  |
| Cust Rep Id | Char(10) |  |
| Cust Rep Name | Char(40) |  |
| **Requirement Category** | **Char(16)** |  |
| Stock Category | Char(16) |  |
| **Filler** | **Char(51)** |  |

##### Retail Unshipped Orders

Layout, location and file name of the Ecom & Retail Unshipped Orders file from DOM would look like as mentioned below. The assumption is that the fields which are highlighted in “green” will be sent to i2 SCP and rest all will be sent as “blank” values. This type of demand is tagged as “RUA” in DEMAND\_ALL table. Layout, location, file name of this feed are described below:

* + Location: $I2OD\_FROM\_DOM
  + File Name: i2ods7\_DOM\_orders.dat
  + Layout of the detail record:

|  |  |  |
| --- | --- | --- |
| **Field Description** | **Field Length** | **Comments** |
| Record Type | Char(1) |  |
| Owner Code | Char(2) |  |
| Sales Org | Char(4) |  |
| Distribution Channel | Char(2) |  |
| Division | Char(2) |  |
| Company Code | Char(4) |  |
| Sold to Customer Number | Char(10) |  |
| Country Code | Char(3) |  |
| Account Class | Char(4) |  |
| **Shipping Location** | **Char(4)** |  |
| **Material** | **Char(18)** |  |
| Base Grid Value | Char(8) |  |
| **Dim 1** | **Char(8)** |  |
| **Dim 2** | **Char(8)** |  |
| Dim 3 | Char(8) |  |
| **Confirm Date** | **Char(8)** |  |
| **Demand Type Indicator** | **Char(1)** | Default value "H" |
| Sales UOM | Char(3) |  |
| **Base UOM** | **Char(3)** |  |
| Open Qty | Char(14) |  |
| Sign1 | Char(1) | Default sign "+" or leave blank |
| **Allocated Qty** | **Char(14)** |  |
| Sign2 | Char(1) | Default sign "+" or leave blank |
| Delivered Qty | Char(14) |  |
| Sign3 | Char(1) | Default sign "+" or leave blank |
| Style color | Char(18) |  |
| Document Category | Char(1) |  |
| Sales Document Type | Char(4) |  |
| Sales Order Number | Char(10) |  |
| Sales Doc Line Item | Char(6) |  |
| Sales Doc Schedule Line | Char(4) |  |
| Ship To Customer Number | Char(10) |  |
| Item Category | Char(4) |  |
| Contract Document Number | Char(10) |  |
| Contract Document Line Item | Char(6) |  |
| Sales Season | Char(3) |  |
| Sales Collection | Char(4) |  |
| Sales Theme | Char(4) |  |
| Creation Date | Char(8) |  |
| Last Change Date | Char(8) |  |
| Last Change By | Char(12) |  |
| Required Delivery Date | Char(8) |  |
| Cancel Date | Char(8) |  |
| Customer PO Number | Char(20) |  |
| Delivery Block | Char(2) |  |
| * + 1. Delivery Status | * + 1. Char(1) |  |
| Sales Rep ID | Char(10) |  |
| Sales Rep Name | Char(40) |  |
| Cust Rep Id | Char(10) |  |
| Cust Rep Name | Char(40) |  |
| **Requirement Category** | **Char(16)** |  |
| Stock Category | Char(16) |  |
| **Filler** | **Char(51)** |  |

* Header data would be in the following format:

Layout:

Char(1) => Record Type ("0")

Char(8) => DOM program number

Char(8) => Date

Char(6) => Time

Example: 0YSDO202220130306032540

* Control (trailer) record data would be in the following format:

Layout:

Char(1) => Record Type ("9")

Char(8) => Date

Char(8) => Time

Char(20) => Records Count

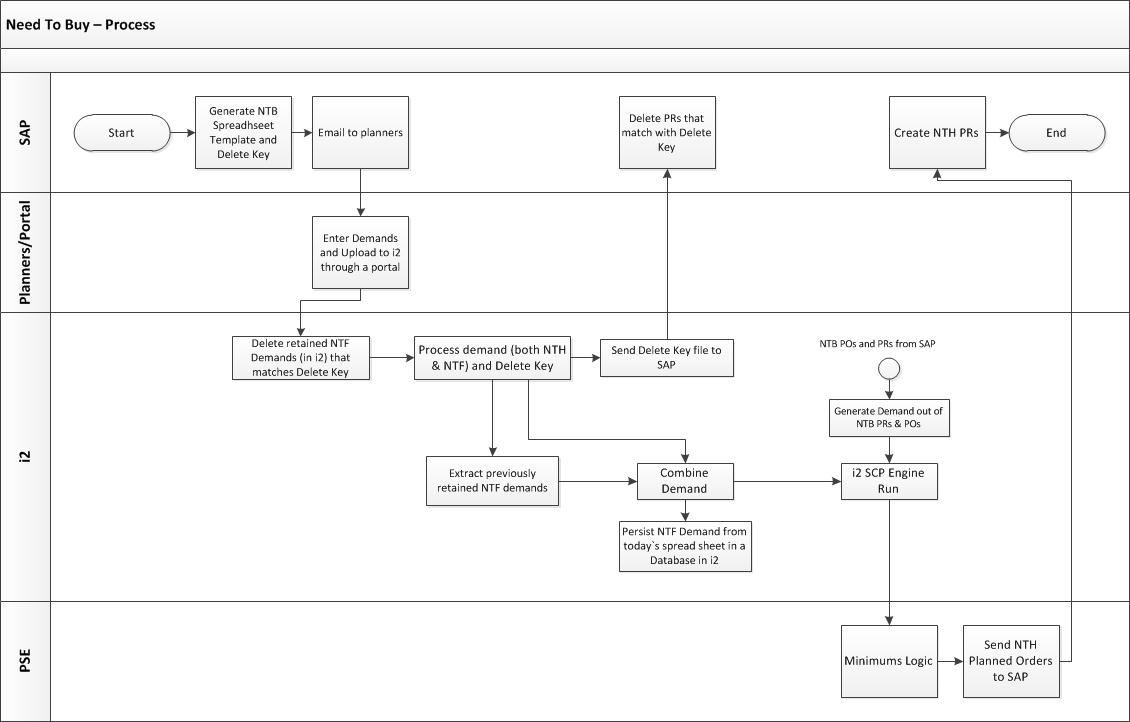
Char(19) => Sum of quantities

Char(1) => Sign ("+")

Example: 920130306032540000000000000036744370000000000014226524+

##### Need To Buy (NTB)

* Need to Buy (NTB) is a type of demand that is sent to i2 systems through a spread sheet.
* Markets that are not integrated with VF`s IT systems send their demands in form a spreadsheet mainly to enable a global view of sourcing report
* Following is the sequence of steps that are executed as part of “Need To Buy” process:



**NTB Template Generation**

* It starts with a template generation in SAP by a planner.
* Planner sets up search criteria – Division, SBU, Sub SBU, Product Line and Plant and generates a template that consists of SKUs and place holders (defined by time buckets – dates) for net requirements. The template would also have a delete key, which would be used as criteria to delete the PRs and PLs (in i2) that were generated previously. Below embedded is a sample NTB template



* Planner receives an email with a generated template

**Update Requirements and Upload**

* Planner then updates the spreadsheet with net requirements (buy requirements) and uploads it through the portal
* On successful upload, the spreadsheet would be available in i2 sever for processing

**Pre-process in i2 funnel**

* From the template, requirements from the first time bucket will be considered NTH (Need To Buy Hard) demands and the ones in second thru last time buckets are considered NTF (Need To Buy Future).
* While processing these demands, i2 funnel persists the NTF demands in a database.
* i2 funnel processes NTH and NTF requirements and converts them into demand line items in DEMAND\_ALL file.

**Planning i2 SCP & Post-processing**

* i2 SCP generates planned orders for NTH and NTF demand items and post them to PSE for buy analysis

**Push PRs to SAP**

* Per business requirements, only NTH demands need to be converted into PRs in SAP
* A scheduled bulk-push program would trigger after minimums logic is run in PSE. This program would push NTH planned orders to SAP for PR creation

#### Supply

Daily supply picture is required to be sent to i2 SCP in order for it to plan. i2 SCP tries to meet demand by using different supply elements – On Hand inventory and Work In Process inventory

##### On Hand Inventory

On Hand feed from SAP to i2 contains latest inventory picture at StyleColor/Size/DC level. This is maintained at a SKU and DC level. Layout, location, file name of this feed are described below:

* Location: $I2OD\_FROM\_SAP
* File Name: i2od\_SAP\_FG\_onhand\_inventory.dat
* Layout of the detail record:

|  |  |  |  |
| --- | --- | --- | --- |
| **Order** | **Description** | **Length** | **Comments** |
| 1 | File Record Type ('1' for Details and '0' for Header) | Char(1) |  |
| 2 | Record Sequence Number | Number(7) |  |
| 3 | Sequence Sign | Char(1) |  |
| 4 | Plant | Char(4) |  |
| 5 | WMS Transaction ID | Char(3) | In today`s world, we are getting BAL => Unrestricted  ERR => IDOC Errors  RTN => Returns  In addition, we will get PRD => Production in future  ??? => Received Blocked |
| 6 | Material Name | Char(18) |  |
| 7 | Stock Category Number | Char(16) |  |
| 8 | Grid Value | Char(8) |  |
| 9 | DIM1 | Char(8) |  |
| 10 | DIM2 | Char(8) |  |
| 11 | DIM3 | Char(8) |  |
| 12 | Storage Location | Char(4) |  |
| 13 | On Hand Quantity | Number(11) |  |
| 14 | Sign1 | Char(1) |  |
| 15 | Blocked Qty | Number(11) |  |
| 16 | Sign2 | Char(1) |  |
| 17 | Quality Inspection Qty | Number(11) |  |
| 18 | Sign3 | Char(1) |  |
| 19 | UPC | Char(18) | New field to be added |
| 20 | Filler | Char(16) | Stock Category will be added |
| 21 | Filler | Char(39) |  |

* Header data would be in the following format:

Layout:

Char(1) => Record Type ("0")

Char(8) => Sequence Number (“0000000”)

Char(3) => “ALL”

Char(4) => Blank

Char(13) => SAP Program Name

Char(3) => Blank

Char(8) => Date

Char(6) => Timestamp

Example: 00000000+ALL YSDINVBALANCE 20130410030730

* Control (trailer) record data would be in the following format:

Layout:

Char(1) => Record Type ("9")

Char(7) => Sequence Number (“0000000”)

Char(1) => Sign ("+")

Char(8) => Date

Char(8) => Timestamp

Char(14) => Records Count

Char(1) => Sign ("+")

Char(14) => Sum of quantities

Char(1) => Sign ("+")

Example: 90000000+201304100307300000001565733+00015011728+

On Hand Inventory file would be processed and formatted into ON\_HAND\_LVL.dat file, which will be used by i2 SCP engine. Different buckets on On-Hand feed to i2 will need to be processed as mentioned below:

|  |  |  |
| --- | --- | --- |
| **Bucket** | **Description** | **Expected Behavior** |
| BAL-UR | Unrestricted | On Hand - Same as today |
| BAL-QI | Quality Inspection | Need to be processed as WIP one day out |
| BAL-Blocked | Blocked | Need to be processed as WIP one day out |
| RTN | Returns | Need to be processed as WIP 7 days out |
| ERR | Failed IDOCs | Ignore - Same as today |
| Production | PRD | Ignore |

##### POs, PRs and STOs

i2 receives PO and PR feed from SAP. PO feed contains open POs and PR feed contains open PRs and Stock Transport Orders (STOs). The data is maintained at Stylecolor/Size/DC level. Layout, location, file name of this feed are described below:

* Location: $I2OD\_FROM\_SAP
* File Name:
  + i2openpo.dat
  + i2openpr.dat
* Layout of the detail record:

|  |  |  |  |
| --- | --- | --- | --- |
| **S.No.** | **Field** | **Characters** | **Comments** |
| 1 | Record Type | 1 | "0" for Header Record Line, "1" for Data Records & "9" Control Record Line |
| 2 | Doc Number | 10 |  |
| 3 | Line Item | 5 |  |
| 4 | Schedule Line Number | 4 |  |
| 5 | Material Number | 18 |  |
| 6 | Size 1 | 4 |  |
| 7 | Size 2 | 4 |  |
| 8 | Plant Number | 4 | Blank Value. This represents Vendor today. Use the "Vendor Plant" mentioned at the end, instead of this field |
| 9 | Storage Loc | 4 |  |
| 10 | Delivery Date | 2 |  |
| 11 | Delivery Date | 2 |  |
| 12 | Delivery Date | 4 |  |
| 13 | Plan Scheduled Qty | 11 |  |
| 14 | Plan Delivered Qty | 11 |  |
| 15 | Open Order Qty | 11 |  |
| 16 | Exit Factory Date | 2 |  |
| 17 | Exit Factory Date | 2 |  |
| 18 | Exit Factory Date | 4 |  |
| 19 | SAP Location | 4 |  |
| 20 | Delivery Complete flag | 1 |  |
| 21 | Unit of Measure | 3 |  |
| 22 | Data Record Type | 1 | "R" for PR "O" for PO "T" for STO (Stock Transport Orders) |
| 23 | Material Type | 1 | Blank Value |
| 24 | Document type(Order Type) | 4 |  |
| 25 | Sales Order Number | 10 |  |
| 26 | Sales Document Line Item | 6 |  |
| 27 | Sold To Party | 10 |  |
| 28 | Container Number | 13 | Blank |
| 29 | Shipment Number | 10 | Blank |
| 30 | Plant  Description | 30 |  |
| 31 | Vendor number | 10 |  |
| 32 | High Lvl.Vendor Description | 35 |  |
| 33 | Transport Mode | 10 |  |
| 34 | Buy Season | 4 |  |
| 35 | Buy Month | 3 |  |
| 36 | PR Status Code | 2 |  |
| 37 | Managing office description | 40 |  |
| 38 | Net price | 11 |  |
| 39 | Condition price | 11 |  |
| 40 | Price unit | 5 |  |
| 41 | Production Date | 8 |  |
| 42 | Category | 16 | New. Use this value to populate WIP\_updated.dat |
| 43 | Vendor Plant | 10 | Use this value for Vendor |

* Header data would be in the following format:

Layout:

Char(1) => Record Type ("0")

Char(8) => SAP program number

Char(8) => Date

Char(6) => Timestamp

Example: 0YMMO000703202013040432

* Control (trailer) record data would be in the following format:

Layout:

Char(1) => Record Type ("9")

Char(8) => SAP Program Number

Char(8) => Date

Char(8) => Timestamp

Char(3) => Number of Header recs

Char(9) => Number of Detail recs

Char(11) => Scheduled qty total

Char(11) => Ordered qty total

Char(11) => Open qty total

Example: 9YMMO000703202013040432 001000375659000304669820000060650000029860482

* Open PR & Open PO files will need to be processed as they are being processed today i.e. the records should be processed into WIP.dat and WIP\_updated.dat
* Category on PO & PR records should be populated on WIP\_updated.dat at the end of the record.
* Assumption is that i2openpo.dat (Open PO file) will contain the STO records. They are identified by the Data Rec Type = “T”, it should be treated as Stock Transport Orders (STOs) and these records should be processed into INSTRANSIT\_SHIPMENT.dat, which goes as an input to the engine.

##### Frozen Planned Orders

This enhancement is incorporated as part of project Acadia. The objective of

this functionality is to selectively retain certain planned orders that are generated

by i2 SCP. Since i2 SCP does not have a way of retaining (freezing) them,

planners can use Planned Order Execution (Buy Plan Report) in PSE would be

used to retain them will be used to retain them.

High-level process flow:

* i2 SCP would generate Planned Orders and transfer them to PSE.
* PSE will allow the users to select and freeze certain Planned Orders.
* These Planned Orders will be sent to i2 SCP as a WIP feed for the next day planning run.
* These will be treated as confirmed supply, until they get un-frozen in PSE.

The following process flow diagram depicts the to-be system behavior pictorially

Frozen WIP as PL

PLss(1)

**i2 SCP**

**PSE**

**SAP**

PRs

PRs/POs (WIP)

Frozen PLs as WIP

Layout, location, file name of this feed are described below:

* Location: $I2OD\_FROM\_I2SC
* File Name: Frozen\_Planned\_Order.dat
* Layout of the detail record (Frozen Planned Orders extracted from PSE is formatted to suit the current PO or PR file)

|  |  |
| --- | --- |
| **Frozen Planned Order data from PSE** | **Length** |
| Hard code it to "1" for the detail records | Char(1)=> File Record Type ('0' for Header record and '1' for Detail record) |
| This needs to be generated in code - a timestamp based key (or) hardcode 10-chars text | Char(10)=> Doc Number |
| Blank | Char(5)=> Doc line item |
| Blank | Char(4)=> Sched Line Number |
| Material from PSE table | Char(18)=> Material Number |
| DIM1 from PSE table | Char(4)=> Size1 |
| DIM2 from PSE table | Char(4)=> Size2 |
| Blank | Char(4)=> Plant Number |
| Blank | Char(4)=> STO Loc |
| DELIVERY\_END\_DATE from PSE tables | Char(2)=> Delivery Date (Month - MM) |
| DELIVERY\_END\_DATE from PSE tables | Char(2)=> Delivery Date (Day - DD) |
| DELIVERY\_END\_DATE from PSE tables | Char(4)=> Delivery Date (Year - YYYY) |
| BUY\_QUANTITY from PSE tables (for a size) | Number(8.3)=> Pln Scheduled Qty |
| Blank | Number(8.3)=> Pln Delivery Qty |
| Blank | Number(8.3)=> Open Order Qty |
| EXIT\_FACTORY\_DATE from PSE tables | Char(2)=> Exit Factory Date (Month - MM) |
| EXIT\_FACTORY\_DATE from PSE tables | Char(2)=> Exit Factory Date (Day - DD) |
| EXIT\_FACTORY\_DATE from PSE tables | Char(4)=> Exit Factory Date (Year - YYYY) |
| DC - from PSE tables | Char(4)=> SAP Loc |
| Blank | Char(1)=> Delivery CPLT Indicator |
| UNIT\_UOM\_TYPE from PSE tables | Char(3)=> UOM 3POS |
| Hard code it as "F" | Char(1)=> Data Rec Type ("O" for PO, "R" for PR & "T" for DC to DC) |
| Blank | Char(1)=> Material Type |
| Blank | Char(4)=> Order Type |
| Blank | Char(10)=> Sales Order Number |
| Blank | Char(6)=> Sales Doc Line Item |
| Blank | Char(10)=> Sold To Party |
| Blank | Char(13)=> Container |
| Blank | Char(10)=> Shipment |
| Blank | Char(30)=> Plant Description |
| Blank | Char(10)=> Vendor Code |
| Blank | Char(35)=> Vendor Description |
| Blank | Char(10)=> Transmode |
| SEASON from PSE tables | Char(4)=> Buy Seas |
| BUY\_MONTH from PSE tables | Char(3)=> Buy Month |
| Blank | Char(2)=> PR Status |
| Blank | Char(40)=> Managing Office |
| Blank | Number(9.2)=> Net Price |
| Blank | Number(9.2)=> Condition Price |
| Blank | Number(5)=> Price Unit |
| PRODUCTION\_DATE From PSE tables | Char(8)=> Production Date |
| Blank | Char(16)=> Category |
| Vendor Number from PSE table | Char (10) => Vendor Plant |

#### Combine and Prioritize Demand

i2 SCP`s MAP solver is implemented to do the Order By Order planning. Each demand line item is treated as an order. Demands from various source systems are combined and prioritized based on certain business rules and a single file, DEMAND\_ALL is generated. DEMAND\_ALL file goes as an input i2 SCP engine.

Following are different types of demand that i2 SCP tries to satisfy:

* USA - Allocated Order in SAP
* USO - Unshipped Order in SAP
* HPR - Hard Projection in SAP
* UBF - Un-booked Forecast from Logility
* DIR - Direct Ship Order in SAP
* NTH - Need to Buy (Hard)
* NTF - Need to Buy (Forecast)
* ROH - IP On Hand
* RPO - IP Purchase Order
* RDO - IP Non Processed Distro
* RUA – Retail Allocated Order in DOM

Demands are sorted based on the following elements:

* Safety Time Date
* Demand Type
* Number of Routings
* Demand Date
* Demand Quantity

#### ADW tables and files

Before i2 SCP engine is run, various data elements required for planning are

Consolidated, loaded into ADW tables for integrity checks and the ones that pass integrity checks are extracted into files. i2 SCP engine processes the files as part of planning run. ADW tables are built with referential integrity among data elements – Item, Facility, Facility Item, Demand All, On Hand, WIP etc. The following table lists

|  |  |  |  |
| --- | --- | --- | --- |
| **ADW table** | **Engine Model file** | **Description** | **Fields** |
| PLN | PLN.dat | Includes the planning horizon | description horizon\_start\_date\_time horizon\_end\_date\_time plan\_current\_date\_time  supply\_chain\_name : "Outdoor-VFCorp" plan\_name : "VFO\_PLAN" plan\_id : "1" ctrl\_id : "1" |
| DMN | DMN.dat | Domain information. In this model, the domain is OUTDOOR\_CORP | is\_managed description role domain\_name supply\_chain\_name : "Outdoor-VFCorp" ctrl\_id : "1" |
| TRANSP\_MODE | TRANSP\_MODE.dat | Not used in this model | NA |
| SC | SC.dat | Contains Name of Supply Chain (model) | description enterprise\_name supply\_chain\_name : "Outdoor-VFCorp" ctrl\_id : "1" |
| BUCKETS | BUCKETS.dat | Defines the name for a list of time buckets used for planning. In this model, it just contains "FISCAL\_CAL" |  |
| BUCKETS\_DETAIL | BUCKETS\_DETAIL.dat | Defines individual bucket, having a specific start and end time. In this model, it contains values for Fiscal Calendar bucket (FISCAL\_CAL) |  |
| ATTR\_EXT | ATTR\_EXT.dat | Contains a listing of all user-defined fields, across all entities and their data types |  |
| ENT\_ATTR\_ASN | ENT\_ATTR\_ASN.dat | Entity that is used to identify the entity to which the extended attribute belongs to |  |
| CAL | CAL.dat | Calendar models the working pattern of a facility, a resource or a division in terms of time. This table/file contains Calendar Name and the entity that it represents | calendar\_name description category sub\_category ctrl\_id : "1" |
| CAL\_ENTRY | CAL\_ENTRY.dat | Lists out the calendar entries for all the calendars defined in the CAL. This calendar\_entry\_name attribute defines the name of the period of time over which a particular pattern applies. The pattern itself is specified via the calendar\_entry\_pattern attribute with values like EVERYDAY, WEEKENDS, DAYS\_OF\_WEEK etc. **Capacity: Start and end dates for the calendar (capacity) entries** | calendar\_name calendar\_entry\_name start\_date\_time end\_date\_time daily\_start daily\_end entry\_rank calendar\_entry\_pattern ctrl\_id : "1" |
| CAL\_ENT\_PAT\_PARAM | CAL\_ENT\_PAT\_PARM.dat | Each calendar\_entry\_pattern (e.g., WEEKDAYS, DAYS\_OF\_WEEK, DAY\_OF\_WEEK\_OF\_MONTH) may need zero or one or more parameters to define the pattern. For “days of the week” pattern needs one parameter - which day of the week. Everyday pattern does not need any parameter to define it. Key attributes for this entity are calendar\_name, calendar\_entry\_name, parameter. | calendar\_name calendar\_entry\_name parameter parameter\_value ctrl\_id : "1" src\_dttm sync\_ind |
| C\_ENT\_TYPE | C\_ENT\_TYPE.dat | **Capacity quantities ( in pieces ) by location, manufacturing unit by calendar entry name.** | calendar\_name calendar\_entry\_name entry\_type entry\_quantity entry\_quantity\_uom ctrl\_id : "1" |
| ITEM\_MASTER | ITEM\_MASTER.dat | Lists all the items i.e. StyleColor-Size1-Size2-Size3 (optional) | item\_name description category sub\_category artificial ctrl\_id : "1" |
| DOMAIN\_ITM\_ASN | DOMAIN\_ITM\_ASN.dat | Items that are valid for domain OUTDOOR\_CORP | item\_name domain\_name : "OUTDOOR\_SITE" ctrl\_id : "1" |
| FCLTY | FCLTY.dat | Contains all facilities - DCs and Vendors that are part of VF Supply Chain Network | facility\_name category sub\_category domain\_name : "OUTDOOR\_SITE" ctrl\_id : "1" |
| FCLTY\_ITEM | FCLTY\_ITEM.dat | Contains Items valid at a facility. Item-DC and Item-Vendor | facility\_name item\_name style\_name category sub\_category in\_cost domain\_name : "OUTDOOR\_SITE" ctrl\_id : "1" |
| VEN\_SOURCING | VEN\_SOURCING.dat | Not used in this model | NA |
| ITEM\_BOD\_ASGN | ITEM\_BOD\_ASGN.dat | Not used in this model | NA |
| BOD | BOD.dat | Not used in this model | NA |
| APPR\_SHPMNT | APPR\_SHPMNT.dat | Not used in this model | NA |
| VEN\_MASTER | VEN\_MASTER.dat | Not used in this model | NA |
| BOM | BOM.dat | Bill Of Material (BOM)s: Items and components that it is made up of | facility\_name bom\_name  ctrl\_id : "1" |
| COMPNT\_SUBST | COMPNT\_SUBST.dat | Not used in this model | NA |
| BOM\_COMPNT | BOM\_COMPNT.dat | A Bill of Material (BOM) may be viewed as a logical grouping of components that are consumed together, to produce an item. | facility\_name bom\_name consumed\_item\_name consumed\_quantity consumed\_quantity\_uom consumed\_facility\_name domain\_name : "OUTDOOR\_SITE" ctrl\_id : "1" |
| RES\_GRP\_MSTR | RES\_GRP\_MSTR.dat | This table lists the resource groups that will be used by the planning engines. Every resource or resource group that is represented in the planning engines needs to be defined in this entity. | resource\_group\_name description ctrl\_id : "1" |
| DOM\_RES\_GRP\_ASN | DOM\_RES\_GRP\_ASN.dat | Association of ResourceGroups to the Domains in the supply chain. This assocation defines the ResourceGroups that can be managed by the specified Domain. | resource\_group\_name domain\_name : "OUTDOOR\_SITE" ctrl\_id : "1" |
| RES\_DETAIL | RES\_DETAIL.dat | Lists the resources and their properties to be used by the planning engines. | resource\_name facility\_name load\_policy category sub\_category resource\_representation resource\_type efficiency\_policy ctrl\_id : "1" |
| OP\_RES\_GRP\_ASN | OP\_RES\_GRP\_ASN.dat | This table declares the association of operations and the resource/resource groups that they load. Where applicable, the substitute resource\_group is also declared. | resource\_group\_name substitute\_resource\_group cdm\_type facility\_name routing\_name step\_number capacity\_flow\_type flow\_quantity ctrl\_id : "1" |
| RES\_CAL\_LIST | RES\_CAL\_LIST.dat | This entity is used to associate calendars (defined in the Calendar entity) with attributes of a Resource that have time varying profiles | resource\_name resource\_calendar\_attribute\_name calendar\_name ctrl\_id : "1" |
| INV\_STYLE | INV\_STYLE.dat | Planning Parameters like Safety Stock (Min On Hand), Minimum Qty, Multiple Qty & Max Quantiy are defined Inventory Style table. These parameters drive planning. | style\_name flow\_policy category sub\_category multiple\_quantity quantity\_range\_min quantity\_range\_max default\_min\_time default\_min\_time\_uom default\_max\_time default\_max\_time\_uom default\_excess\_on\_hand default\_min\_on\_hand domain\_name : "OUTDOOR\_SITE" ctrl\_id : "1" |
| ITEM\_DLVRY | ITEM\_DLVRY.dat | This contains delivery information of an item. This is not used much in this model. | transport\_mode\_name facility\_name item\_name domain\_name : "OUTDOOR\_SITE" ctrl\_id : "1" |
| ROUTING | ROUTING.dat | Sequence of operations. | facility\_name routing\_name effective\_start\_date\_time effective\_end\_date\_time ctrl\_id : "1" |
| OPERATION | OPERATION.dat | An activity that is required to transform or a transport an item.  In this mode, **Lead Time information (from Planning Routings) for the operations (Vendor lead time & transit time)** | facility\_name routing\_name step\_number operation\_name operation\_type category sub\_category operation\_time operation\_time\_uom yield interruptible slack\_policy operation\_calendar ctrl\_id |
| ITEM\_BOM\_RTNG | ITEM\_BOM\_RTNG.dat | The ItemBomRouting entity lists the association between the BOM and the Routing to be used in the planning for one produced item. In this model, **Prioritized Routings from Planning Routings** | routing\_facility\_name bom\_name routing\_name produced\_facility\_name produced\_item\_name produced\_quantity produced\_quantity\_uom priority priority\_criterion domain\_name : "OUTDOOR\_SITE" ctrl\_id : "1" |
| INTRANSIT\_SHPMNT | INTRANSIT\_SHPMNT.dat | Lists all the instransit shipments supply. In this model, it contains Stock Transport Orders (STOs) | intransit\_shipment\_id item\_name destination\_facility quantity quantity\_uom state\_spec receipt\_date\_time released : "Y" locked : "Y" domain\_name : "OUTDOOR\_SITE" ctrl\_id : "1" plan\_id : "1" |
| SCH\_RCP | SCH\_RCP.dat | Not used in this model | NA |
| DEMAND | DEMAND.dat | Table/file that holds "types of demand" |  |
| ON\_HAND\_LVL | ON\_HAND\_LVL.dat | This table/file carries the latest On Hand inventory picture across Material/DCs | receipt\_id facility\_name item\_name on\_hand\_date\_time on\_hand\_quantity on\_hand\_quantity\_uom lots\_tracked domain\_name : "OUTDOOR\_SITE" ctrl\_id : "1" plan\_id : "1" |
| WIP | WIP.dat | Table/file that contains Work In Process inventory i.e. PO & PRs | lot\_number produced\_facility\_name produced\_item\_name  routing\_facility\_name  routing\_name  step\_number  alternate\_facility\_name  alternate\_item\_name  produced\_quantity  produced\_quantity\_uom  consumed\_quantity  consumed\_quantity\_uom  wip\_start\_date\_time  wip\_end\_date\_time  state\_spec  consume\_capacity  utilize\_yield  generate\_operation\_plan  attach\_state  locked  explode\_inhibitor  operation\_name  identifier  domain\_name "OUTDOOR\_SITE" ctrl\_id "1" plan\_id "1" |
| DEMAND\_ALL | DEMAND\_ALL.dat | Lists prioritized orders of demands (orders) | request desc of request delivery\_request due\_date\_start adjusted\_due\_date item\_request request\_item quantity request\_loc ba\_limit bl\_limit priority |

ADW tables and corresponding files that are extracted and used in planning run.

Following table lists a complete set of ADW tables and corresponding files extracted from them along with the fields that are part of them.

#### Planning logic

Each demand item in DEMAND\_ALL file (order from i2 SCP standpoint) is planned one at a time. Once a demand item is planned, it is locked from being changed. SCP would not re-look at the planned demand line item that is already planned, to satisfy other demands.

i2 SCP`s planning sequence is as explained below:

*for each demand line item in DEMAND\_ALL*

*{*

* *Try to satisfy the demand with Finished Goods Inventory*
* *If it is not satisfied fully/partially:* 
  + *Try Open Purchase Orders*
* *If it is not satisfied fully/partially:* 
  + *Try Open Purchase Requisitions*
* *If necessary, system then attempts to satisfy demand by recommending buying more.*
  + - *The highest priority Planning Routing is used first.*
    - *System attempts to deliver supply to the dc based on the Demand Date and Build Ahead setting.*
  + *Capacity is checked.*
  + *Other routings are tried automatically (if necessary) to satisfy the demand & minimize lateness.*
  + *Plan is “locked”*
  + *Process the next demand*

*}*

In some cases demands cannot be satisfied - Demand Not Planned Condition. It may be because “*Plan No More*” Setting, “*Build Late Limit*” was reached, or Capacity not available

Planning logic and capacity search is as explained below:

Build Ahead Limit

Build Late Limit

Original Due date (DC Date)

Adjusted Due Date

Build Ahead Limit

Build Late Limit

The Adjusted Due Date is defined as Actual Due Date offset by Safety Time. The Build Ahead Limit (BAL) and Build Late Limit (BLL) will be defined for a material/DC and will be available on a demand line item.

* + - Adjusted Due Date = Original Due Date – Safety Time
    - Adjusted Due Date must be greater than or equal to Plan Current Date

i2 SCP engine starts planning at Safety Time Date and if there is no capacity available, the engine tries to plan early first. In case there is no capacity available, it continues to check for capacity till it hits “Build Ahead Limit”. If it does not find capacity between Safety Time Date and Build Ahead Limit, it starts moving out from Safety Time Date and tries to plan late.

#### Outbound feeds

After i2 SCP`s planning run, i2 post processing programs build pegging tables, stocking plans extensions and build outbound feeds. IP Due In report is one of such outbound feeds.

**IP Due-In report**

IP Due In report (*IP\_PO\_DUE\_IN.dat*) is generated under $I2OD\_TO\_IP directory. Layout of this file is as mentioned below:

|  |  |
| --- | --- |
| **Field** | **Length** |
| IP PO (from IP\_PO.dat) | Char(20) |
| SAP PO (WIP-PO) | Char(10) |
| Material (from IP\_PO.dat) | Char(18) |
| Size1 (from IP\_PO.dat) | Char(6) |
| Size2 (from IP\_PO.dat) | Char(6) |
| UPC (from IP\_PO.dat) | Char(12) |
| Order Qty (from IP\_PO.dat) | Char(7) |
| Received Qty (Planned Output) | Char(7) |
| Total Open Qty (IP Open Qty from IP\_PO.dat) | Char(7) |
| Due In Qty | Char(7) |
| Demand Planned Date | Char(8) |
| SAP Sales Org | Char(4) |
| WIP Flag | Char(1) |

#### RDW tables

Important RDW tables and their fields are:

|  |  |
| --- | --- |
| **RDW Table** | **Description/Fields** |
| WORKORDER\_PLAN | The i2 recommended Supply Plan after the planning run. Includes On hand, WIP and i2 Plans.  Data includes: style color, size, vendor, dc, dates, quantities and motive Note: "Motive" is an i2 term and tags records as STATE ( work in process & on hand inventory ) or PRODUCE ( new plans) to satisfy demand not covered by on hand inventory and/or purchase requisitions & purchase orders. |
| DEMAND\_RDW | Status of demands after the planning run: planned (satisfied) or not, on time or late, and if late how late. Data includes: Style color, size, dc, demand type, demand date, planned date, requested quantity & planned quantity Demand at the order line level, therefore, demand dates are day specific. |
| DEMAND\_SUPPLY\_NEW | Pegs (matches) all demand types to supply. Also contain details of both demand and supply. |
| DEMAND\_SUPPLY\_NEW\_ALT | Similar to Demand\_Supply\_New with Category Assignments - source for Category Logic |
| DEMAND\_SUPPLY\_XUBF | Same as DEMAND SUPPLY NEW but without demand type UBF (Un-booked forecast). |
| DEMAND\_SUPPLY\_XUBF\_ALT | Similar to Demand Supply XUBF with category Assignments |
| STOCKING\_PLANS | Time phased inventory projections. Includes Begin on hand, Consumed Qty, Produced Qty, Ending on Hand and several net position & coverage calculations. |
| STOCKING\_PLANS\_EXT | Stocking plans at Material / Size / DC / Week |
| STYLE\_STKPLN\_EXT | Stocking plans at Material / DC / Week |
| STYLE\_STKPLN\_MONTHLY\_EXT | Stocking plans at Material / DC / Month |
| UNIT\_STKPLN\_EXT | Stocking plans at Capacity Group / DC / Week |
| PDTLINE\_STKPLN\_EXT | Stocking plans at Product Line / DC / Week |
| SUBSBU\_STKPLN\_EXT | Stocking plans at Sub SBU / DC / Week |
| SBU\_STKPLN\_EXT | Stocking plans at SBU / DC / Week |
| PROFITCNTR\_STKPLN\_EXT | Stocking plans at Profit Center / DC / Week |
| RES\_PLAN | Resource Plan shows Capacity Utilization. Data includes Capacity, Capacity Load and Remaining Capacity by Capacity Group by Vendor by Week |
| RESOURCE\_ASSIGN | Resource Assignment contains details of Capacity load contained in the Res\_Plan table.  Data includes: style color, size, work center, planned end date, capacity consumption quantity, work order #, routing name Note: "work center" is the combination of vendor and capacity group |
| STYLE\_COLOR\_ATTRB | Style Color, Description, SBU code & name, Sub SBU code & name, Product line code & name, |
| VF\_CALENDAR | Date, Year, Week, Week Begin & End, Month Begin & End |
| SKU\_ATTRB | Item Name, Size 1, Size 2, SKU Quantity ( example: NF000A14606K-XXL-REG, XXL, REG, 1) sku quantity indicates the total quantity for "pre-pack" sizes |
| STYLE\_COLOR\_SEASON | Style Color, Season, Season Description, Season Year, Company, Inventory Class Code.(example: NF000AAHYJK3, 113, F13, 2013, TNF, 10) Note: To get accurate results it is important to set a limit one season when joining with other table(s). |
| CUSTOMER\_ATTRIBUTE | Customer Number and Customer Name ( example: 0000016582 Dicks Sporting Goods ) |
| MFG\_UNIT\_ATTRIBUTE | Capacity Groups with attributes - description, capacity owner |
| PRODUCT\_SKU\_COST | Standard Cost information by material, size and dc |
| PRD\_SKU\_WHSL\_PRICE | Pricing information |
| PRD\_SKU\_RTL\_PRICE | Manuf Suggested Retail Price (MSRP) information by season |
| PRD\_SKU\_RTL\_PRICE\_UNIQUE | Manuf Suggested Retail Price (MSRP) information for the latest season |
| INVENTORY\_POLICY | Material, DC, Policy Code Assignment |

### *PSE*

Planning System Enhancements (PSE) is a supplementary application that works along with i2 Supply Chain Planner (SCP). PSE provides user interface for planners to work on the planned orders generated by i2 SCP. It also houses master data for i2 SCP. Following are five major functionalities that PSE offers:

1. **Routings:** This module is used to create and manage planning routings
2. **Capacity:** Provides a way to define capacity values for vendors and capacity groups
3. **Minimums:** It allows planners to maintain vendor minimums that are used in identifying the buy-type of the planned orders
4. **Demand Spreading:** IP POs are sent to i2 in bulk quantities. Demand Spreading feature allows the planners to set up the split rules based on which the POs are split before they are processed in i2 SCP engine
5. **Planned Orders:** Highly used by Outdoor coalition planners to analyze planned orders, make buy decisions and trigger creation of PRs in SAP

#### Routings

Routings application enables planners to add & maintain the routes at a style color Level with any exception. Business benefits of Routings application are:

* Significant improvement in planner productivity
* Effectively able to manage SKU exceptions for routes
* Effectively able to manage phase-in and phase-out at style color level
* Improvement in planner productivity and accuracy with data maintenance

Following actions can be performed using Routing module:

* Routing process facilitates in planning how a finished material can reach a DC (Distribution Center). So, routing process may pass through a sequence of steps & locations before it reaches the target DC. Route is always referred as a combination of Material and DC.A material can typically be tagged in to one or more set of DCs.
* Typically the majority of the users who use Routing are “Planners” who actually log on to the application and set the routes between material and DC. They can add or delete the steps in any route.
* The modules of the Routing application are: Routing Maintenance, Mass Maintenance, Trans LT Maintenance, Change Log, Template and Material w/o Routes.
* In Routing Maintenance, the user can do the following three things: Routing Maintenance, Copy routing and Linked DCs.
* In the Routing Maintenance to maintain a route the user has to enter Material and DC to search any route for the combination:
* To search a Menu, there are various search filter provided in tree structure to help the user.
* While maintaining the route the user may specify location/vendor and lead time.
* Lead time refers to the time that would be required for an operation on a material to be complete and reach a location with a specific mode of transport.
* To create a new route, the user may make use of the pre-existing templates present in the application
* The user can specify the LT time, operation, location while adding the step to define the route.
* The last step of the route should always be “TRANSPORT” and the lead time for that can’t be modified for the user.
* Start Date and End Date indicate the only period in which the location/vendor will be active. If it is not mentioned, then it indicates the location is available throughout the year.
* Status indicates whether the location in route is active or not. “Y” refers to “Yes” where “N” denotes inactive.
* Priority value indicates the priority of the step in the route. Lower the value, higher the priority.
* Linked DCs helps to enable sharing the same route between different DCs and same material combination.
* Copy Routing provides the ability to copy the same route to a different material but same DC.
* Mass Maintenance helps in updating the LT time or Effectivity for the operation involved in the step of a route. For ODW the only possible lead time that can be modified for an operation is “VENDOR” but for IMW it may have “CUT”, “SEW” etc except for “TRANSPORT”.
* “TRANSPORT” LT time can’t be modified using Mass Maintenance screen but using Trans LT maintenance.
* Using “EFFECTIVTY”, the start date and end date of a location can be modified globally. If it is modified for a location, then the modified data will be reflected in all the routes in which it is part of.
* Trans LT Maintenance helps in modifying the LT time of various modes of “TRANSPORT” for a “From Location” to “To Location”. Once it is done, it will be reflected in all “TRANSPORT” step of the routes having the location combination and the mode.
* Change Log helps us to view the details of the update on Material and /or location.
* Change Log displays information such as Table Name, fields modifies, updated by, updated date, the previous value etc of the routing information.
* Material w/o Routes module helps the user to find out the materials that are not attached to any route for a DC.

#### Capacity

Capacity application enables planners to maintain capacity at plant - planning unit or sub plant - planning unit or directly at planning unit level. Following are business benefits of Capacity module in PSE:

* Ability to maintain the capacity in dozens at plant level
* Enables business users to balance the capacity between Plant/Sub Plant and planning unit
* Constraint Reporting capability to manage capacity violations
* Improvement in planner productivity through intuitive interfaces and accuracy with calculations & data maintenance

Following actions can be performed using Routing module:

* Plant/Sub-Plant Maintenance
* Maintain capacity at sub-plant and allocate to planning units
* Maintain capacity at plant and allocate to planning units
* Maintain capacity directly at planning unit level
* Capacity maintained at weekly or monthly level
* Capacity maintained in Minutes or Units
* Add Sub-Plant to plant
* Associate/De-associate planning units to the Plant/Sub-Plant
* Define max operator and review the operator usage violation
* Maintain sub-Constraint capacity and review sub-constraint violations

#### Minimum

Minimum module facilities the user to maintain the minimum for a material or style or fabric in vendor level. It also helps the user to create style or material or fabric level groups. Minimumsmodule has a provision to ensure the planned buy order meets the vendor minimum by the vendor defined or not .If the planned Buy order not meets the vendor minimum then marked this records as not valid, then ask the user to re plan the buy order to meet the vendor minimum.Minimum module also helps in capacity balancing.

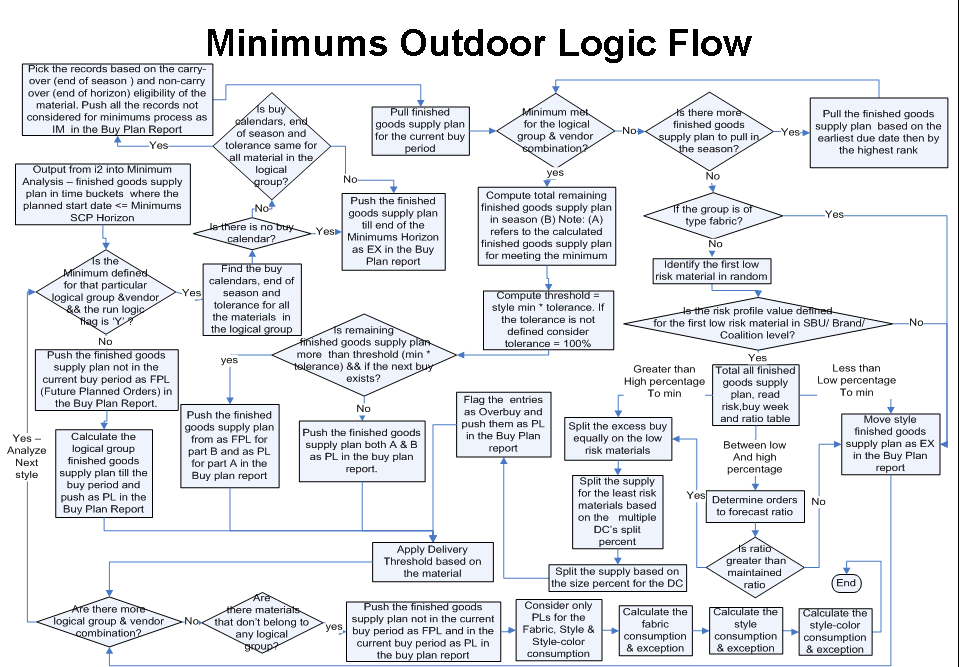
Following are the business benefits of Minimums module:

* Significant improvement in planner productivity
* Enable users to perform minimums processing based on the logical groups defined by the users at the style-level, material-level, fabric-level and other-level
* Effectively able to calculate the reporting minimums based on the changes
* Provides an ability to maintain Fabrics & BOM
* Scalable Centralized application.

Following actions can be performed using minimums module in PSE:

* Definition & Minimum Maintenance Screen helps the user to update the group minimum and run logic flag values.
* Material Association & De-association Screen helps the user to Associate and disassociate the material with already available groups.
* Material Minimum Maintenance Screen is used to update the vendor minimum value for vendor and Material combinations.
* Style Minimum Maintenance Screen is used to update the vendor minimum value for vendor and style combination.
* Fabric Minimum & Maximum Maintenance Screen is used to update the vendor minimum & vendor Maximum value for vendor and fabric combination.
* Material Report Screen is used to display the vendor minimum value & Consumption for vendor and Material combinations.
* Style Report Screen is used to display the vendor minimum value & Consumption for vendor and Style combinations.
* Fabric Report Screen is used to display the vendor minimum & vendor Maximum value for vendor and fabric combination.
* Delivery Threshold Maintenance Screen is used to update the Delivery Threshold value for vendor and Material combinations.
* Carry over Eligibility Maintenance Screen is used to update the Carry Over Flag value for Material.
* Mass Upload Screen is used to update the data in a bulk manner. Ability to De-associate materials easily from this screen (from an already associated group).

Flow chart explaining the minimums logic flow:



#### Demand Spreading

Island Pacific (IP) system sends Retails Purchase Orders to i2 SCP. Though the demand is spread out throughout the month the POs are created with bulk quantities with a single date of a month on them. This will result in other demands being not satisfied. Due to some technical issues, retail POs cannot be spread within IP system. To solve this, PSE enables a user interface, where the business users can set up split rules on a PO. These rules are used by a nightly batch program to spread the demand before it is sent to i2 SCP.

The Demand spreading maintenance is at material level and this module is to provide the split rules for materials.

1. Visibility to Original IP PO quantity and Balance Quantity in both quantity and LUM (lowest unit of measure)
2. Ability to re-date and or split multiple records (Island Pacific PO’s) at one time.
3. Flexibility to manage splits by percentage or quantity.
4. Ability to add or delete splits.
5. Summary level information displayed to understand how much was changed in total.
6. Demand spreading is valid only for Island Pacific PO’s with remaining balance quantities.
7. The split rule information is automatically purged when the IP PO Balance Quantity goes to zero.
8. Island Pacific PO data is refreshed in the Demand Spreading Module once per day during the nightly batch schedule.
9. Therefore, new Island Pacific PO’s created today, will be available to split and/or re-date using the Demand Spreading Module tomorrow.

The functionality of the following buttons on the screen are: -

*Modify Split:* To change multiple records at one time.

*Delete Split*: To delete split rules on multiple records at one time.

*Save:* To save changes

*Cancel:* to revert back to last save

#### ned

This module contains mainly two kinds of views: Planned Order View and Planned Order Execution. They are mainly used for generating the reports and at the same time they allow to perform modifications in the existing Planned Order so as to trigger Purchase Orders in SAP.

**Planned Order Execution:** There are various functionalities this screen - user can do a split, combine, change vendor and then push the modified planned to SAP to create a PR. The following are salient features of this functionality:

* 1. **Combine**: Allows planners to combine multiple Planned Orders [provided Materials, Vendor & DC are same for all the selected planed orders.] The combine functionality also involved an advance logic which involves adding “minimum delivery quantity" and "delivery intervals" as optional parameters when using the planned order combine feature. This automatically split planned orders if necessary to meet the minimum delivery quantity. The optional parameters are user entered when using the Combine Function (no pre-determined master data set-up).

Advance combine would provide the ability to:

* + 1. Combine the planned orders such that the resultant planned orders created with a certain interval apart
    2. Combine the planned orders such that the resultant planned orders is created with the selected quantity
  1. **Split**: Allows the planners to split a Planned Order based on the split quantity provided. When the quantity gets split at Material/DC level, level it will split the material proportionately.
  2. **Change Vendor:** Vendor on a planned order can be changed in order to balance capacity. On changing the vendor Exit factory date and Start date will change based on the lead time of new vendor.
  3. **Mass change Dates:** The mass change can be applied on the production date, exit factory date and DC date. It can also set the Buy month and Buy Season. A material can be associated to one or more season. Only the seasons that are associated with the material on the planned order will be allowed to be applied. Any other season cannot be applied on the material.

* 1. **Push to SAP:** After performing these modifications , Planned Orders would be then submitted to SAP system where new Purchase Requests are created against these Planned Orders

**Planned Order View:** This is a new GUI that was built as part of project Acadia. It is similar to YM49N transaction SAP 4.6. This view allows the PSE users to view the planned orders generated by i2 SCP, in a pivot view where the planned orders are categorized by their “Buy Type” as tagged by minimums logic and bucketed into weekly/monthly view (based on user selection). Users can select the planned orders from this view and perform the following activities:

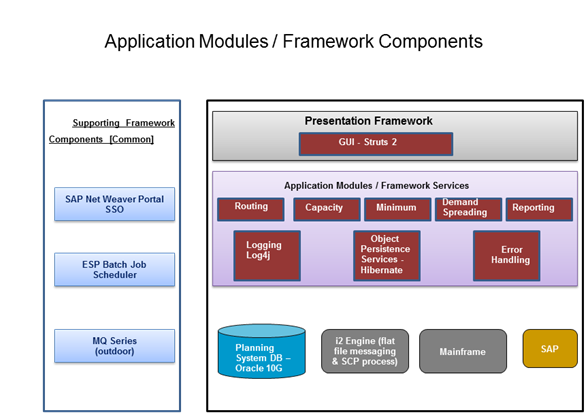
Following actions can be performed by planners on this screen:

1. Search the planned orders by providing search criteria on various data fields. The search criteria should be similar to what is currently available in “Planned Order Execution” screen of PSE.
2. Choose from list of view options- Weekly or Monthly.
3. Select the level of reporting – By Material, By Material/DC or By Material/DC/Vendor
4. View a horizontal buy plan view of planned orders entering the search criteria. This screen would allow users to select certain planned orders, perform combine/split/change vendor/change quantity and push them to SAP as PRs
5. It allows users to select certain planned orders and push them to SAP as PRs. As part of this process, users need the ability to perform following activities on the proposed view:
   * Select the planned orders by clicking a grid value
   * View the planned orders that represent the grid selected
   * Select certain planned orders from the list of planned orders and combine them
   * Split the planned orders
   * Push the combined/split planned orders to SAP as PRs.
6. **Push Report:** Status of PRs that are pushed to SAP is displayed on Push Report screen. It shows status of each push. If a PR is created successfully, it displays the PR number. Otherwise, it displays the error description.

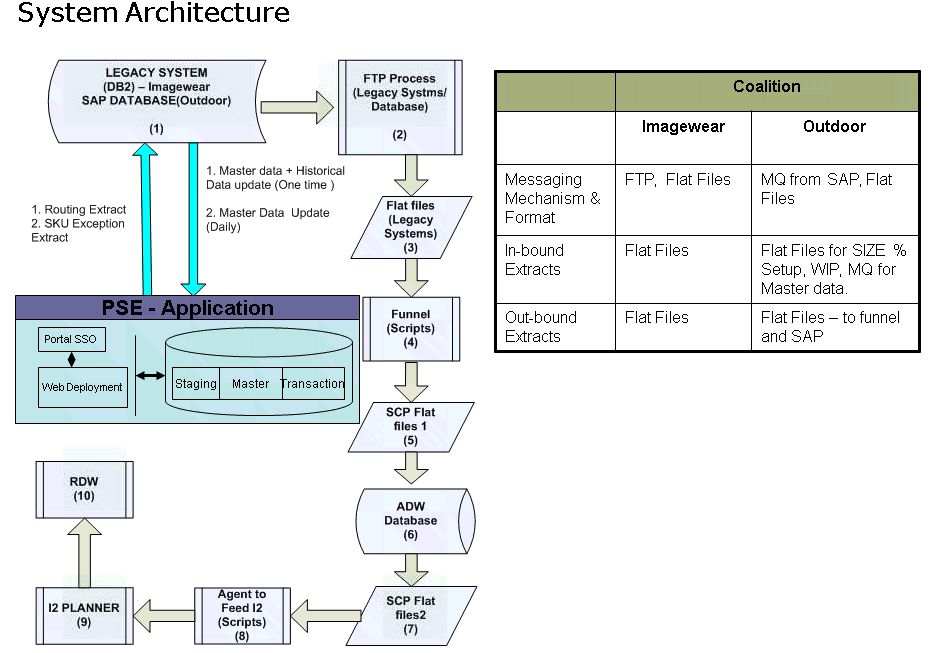
#### Technical Details

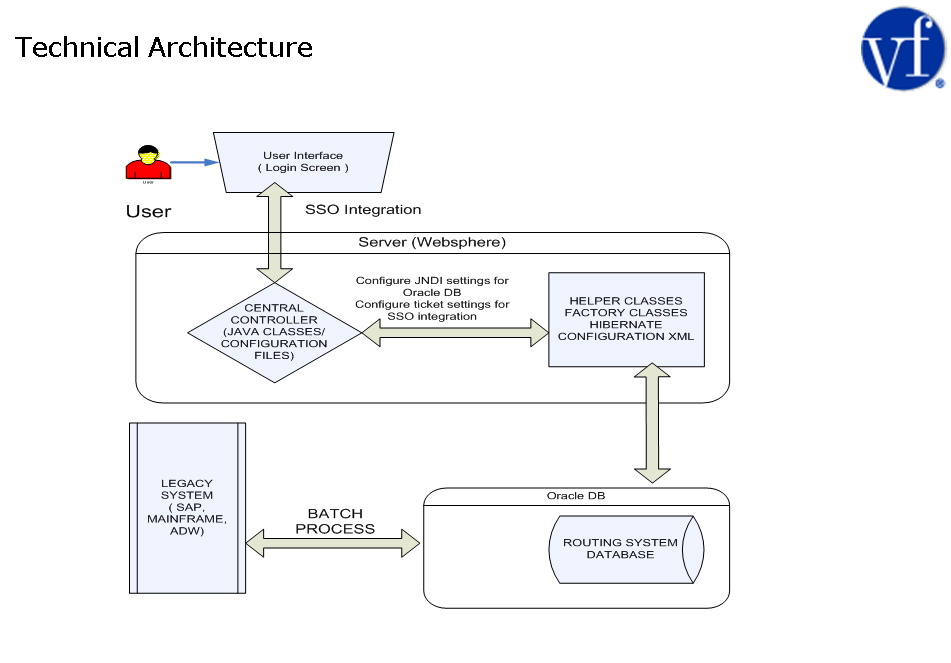
Technical details of PSE application are covered in this section:

##### Application Modules / Framework Components



##### System and Technical Architecture





##### Important PSE tables

|  |  |
| --- | --- |
| **Important DB tables** | **Purpose** |
| MINS\_RECORD\_OD | Buy Plan Report is built on this table |
| WORKORDER\_PLANNED\_OD  WORKORDER\_PLANNED\_OD\_FLAGS | Contains the Planned Order Information from i2 |
| CAPACITY  PLANNED\_CAPACITY  UNIT\_CAPACITY | Contains the details of Capacity Module |
| LOCATION  LOCATION\_ATTRIBUTES | Contains location details of Vendor and DC |
| MATERIAL  PLAN\_GRID  PLAN\_GRID\_STG | Material details |
| TRANS\_MODE  TRANS\_LEAD\_TIME | Details related to transportation time from Vendor to DC |
| DIVISION  SBU  SUB\_SBU  PROD\_LINE | Details of Brand , coalition  SBU is the next level after DIVISION  SUB\_SBU is the next level after SBU |
| BUY\_CALENDAR  SCREEN | Details of application navigation path |
| USER\_ROLES\_ACCESS USER\_ROLES\_ACCESS\_SCREEN | Details of Roles and Screen mapping |

## Servers, Databases and Important directories

|  |  |  |
| --- | --- | --- |
| **Item** | **i2 SCP** | **PSE** |
| Production Server | Not available yet | Not available yet |
| Sysd Server | itgcurs43 | itgcurs43 |
| Production DB (ORACLE\_SID, Port, Schema & Password) | Not available yet | Not available yet |
| Sysd DB (ORACLE\_SID, Port, Schema & Password) | itgcurs43.vfc.com:1526:I2OD65PP Username: i2vfo Password:vfo | itgcurs43.vfc.com:1526:I2SC65PP Username: i2vfcdev Password:acdppi2\_ |
| Production ESP Batch Schedule(s) | Not available yet | Not available yet |
| Sysd ESP Batch Schedule(s) | Not available yet | Not available yet |
| **On Production server** | Not available yet | Not available yet |
| Shell scripts directory | Not available yet | Not available yet |
| Cobol scripts directory | Not available yet | Not available yet |
| Perl scripts directory | Not available yet | Not available yet |
| SQL scripts directory | Not available yet | Not available yet |
| Datasets directory | Not available yet | Not available yet |
| Engine(or equivalent planning/forecasting program) installation directory | Not available yet | Not available yet |
| Engine related files (import, export, run files etc.) location | Not available yet | Not available yet |
| PSE Admin console | NA | <http://itgcurs43.vfc.com:9443/ibm/console/login.do> |
| PSE Application Access | NA | <http://itgcurs43:13000/AcadiaPP> |
| **On Sysd server** |  |  |
| Shell scripts directory | /u11/scp/I2OD65PP/test/scripts | /u11/scp/I2OD65PP/test/scripts |
| Cobol scripts directory | /u11/scp/I2OD65PP/test/src | NA |
| Perl scripts directory | /u11/scp/I2OD65PP/test/perl | /u11/scp/I2OD65PP/test/perl |
| SQL scripts directory | /u11/scp/I2OD65PP/test/sql /u11/scp/I2OD65PP/test/rdw | /u11/scp/I2OD65PP/test/sql |
| Datasets directory | /usr/rhythm/I2OD65PP/datasets | /usr/rhythm/I2OD65PP/datasets |
| Engine installation directory | /opt/i2/scp/6.3.2 | NA |
| Engine related files (import, export, run files etc.) location | /usr/rhythm/I2OD65PP/datasets/SCP\_CDM\_TBL/<<BRAND) | NA |
| PSE Admin console | NA | Not available yet |
| PSE Application Access | NA | Not available yet |
| Code base details | NA | O:\## Wipro Knowledge Transfer\PSE\Acadia |
| Dimension Path | NA | I2OM\_INIT\_WS \ I2OM \ PSE |

## ESP Batch Schedule

<<TODO>>

# Processes

## Configuration Management

<<<Describe how and where different documents are stored and how to access them (from offshore/onsite). Describe version control methods. For VDM projects; Multi-site Configuration management tools should be used>>>

## Support

<<<Describe mechanisms to resolve problems like servers going down, link going down, environment failure, communication failure, login access problems, email/voicemail problems etc>>>

## Communication

<<<Describe various communication channels. Look at both onsite-onsite as well as onsite-offshore situations. Look at status reporting, reviews, meetings, escalations, discussions and communication tools like telephones, VOIP, video, chat, email, voice-mail etc. as modes of communication. Detail processes to bridge any time-zone issues. Identify specific templates for communication, if required. For VDM projects, describe inter-location aspects of communication>>>>

## Miscellaneous

<<<Include any processes that cannot be covered under the above sections. Maybe something like backup procedures, disaster management procedures, audits etc. For VDM Projects, document them keeping multi-location execution into consideration.>>>

# Known Issues & Defects

List all the reports in the system that is maintained to track the measures defined in the Analysis Section, including the Non-Functional Requirements.

# Glossary and Acronyms

# References

Any documents, journals, books referenced while understanding the system.

|  |  |  |  |
| --- | --- | --- | --- |
| Document Name | Document Location | Version # | Last Updated |
|  |  |  |  |