Wyeth Austria

ALF

Austria Logistics and Finance System

Manual for Administrators

Table of contents

1 Introduction

1.1 Preface

This is the technical documentation for the ALF-System and is intended to support the helpdesk and other admin staff to understand the technical background of the system and to react properly if problems occur.

The **A**ustrian **Logistics** and **F**inance tool was designed for the following objectives:

Sales Reporting:
Reporting of sales (currently of our distributor Sanova) in various reports and views.

Logistics Reporting:
Displaying relevant Logistics information and verifying the correctness of the data.
Providing month end checks and month end closing procedures.

Accounting:
Automatically creating accounting records and providing downloadable files for JD Edwards file imports, finance month end checks and processing.

Targeting:
With the targeting module the proportional sales of the sales representatives are reported and the targets of the sales reps can be planned and online compared to their sales.

1.2 Process Overview and Logical Data Processing

The ALF system processes data from two external sources and data which is directly administrated in the system to fill the different reports and database.
Currently the two external sources are the daily-transmitted transmissions from the 3rd party distributor Sanova and the FORTE system of Wyeth.

Data transmitted:

Sanova
- Sales
- Stock
- Inventory Levels
- Customers
- Products

FORTE
- Budgets
- Products
- Products Groups
- Countries
- Currencies
- FX-Rates
- Cost of Goods
- Layout for the GM Report

The ALF system processes this data for different reports and exports. A part of the data is processed daily the other part is processed monthly.

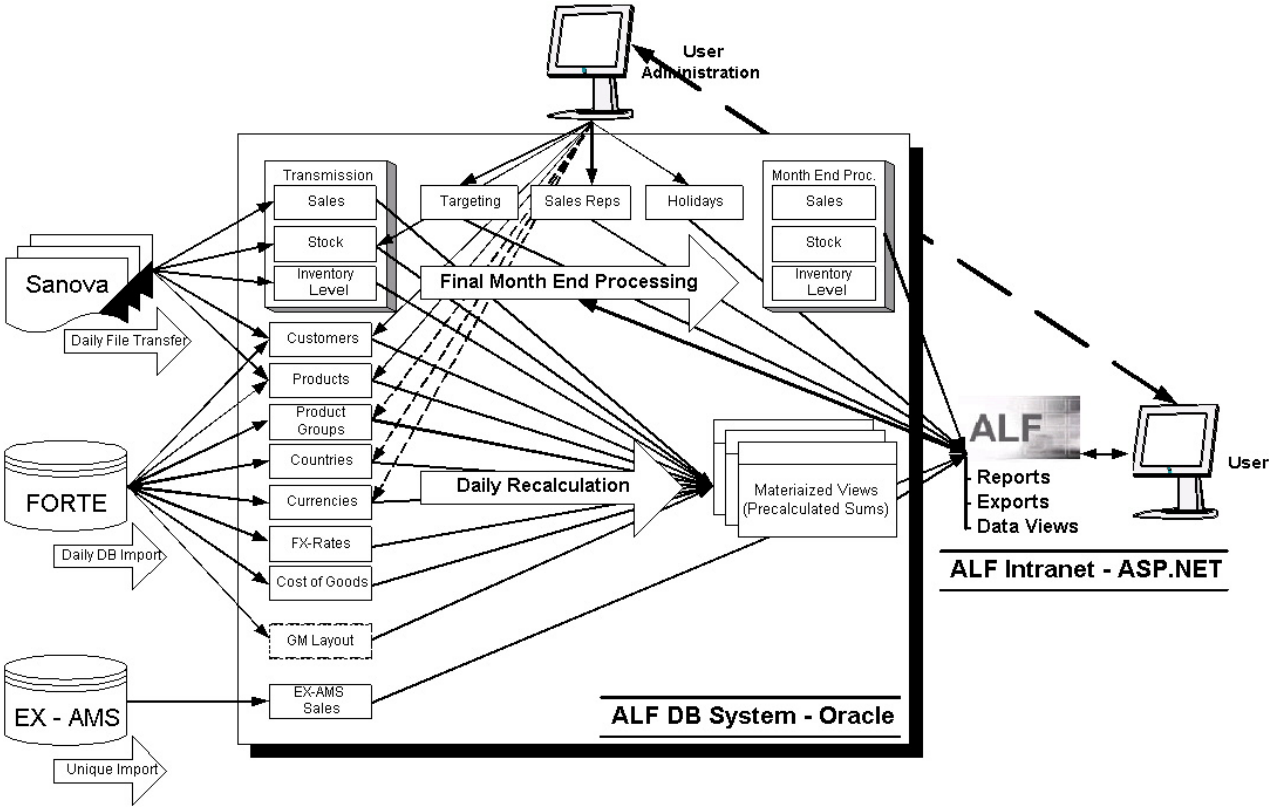
The system has the following outputs:

Outputs

Reports
- Sales Reports
- Logistics Reports
- Other Reports
- Targeting

Exports
- JDE – Export
- IRIS Exports (Sales & Stock)
- WWS Export

Data views
- Master Data
- Processing
- Targeting


User Administration
Transmission
Sales
Stock
Inventory Level
Targeting
Sales Reps
Holidays
Month End Proc.
Sales
Stock
Inventory Level
Final Month End Processing
Sanova
Daily File Transfer
Customers
Products
Product Groups
Countries
Currencies
FX-Rates
Cost of Goods
GM Layout
FORTE
Daily DB Import
Daily Recalculation
Materiaized Views (Precalculated Sums)
ALF
Reports
Exports
Data Views
User
ALF Intranet - ASP.NET
EX - AMS
Unique Import
EX-AMS Sales
ALF DB System - Oracle

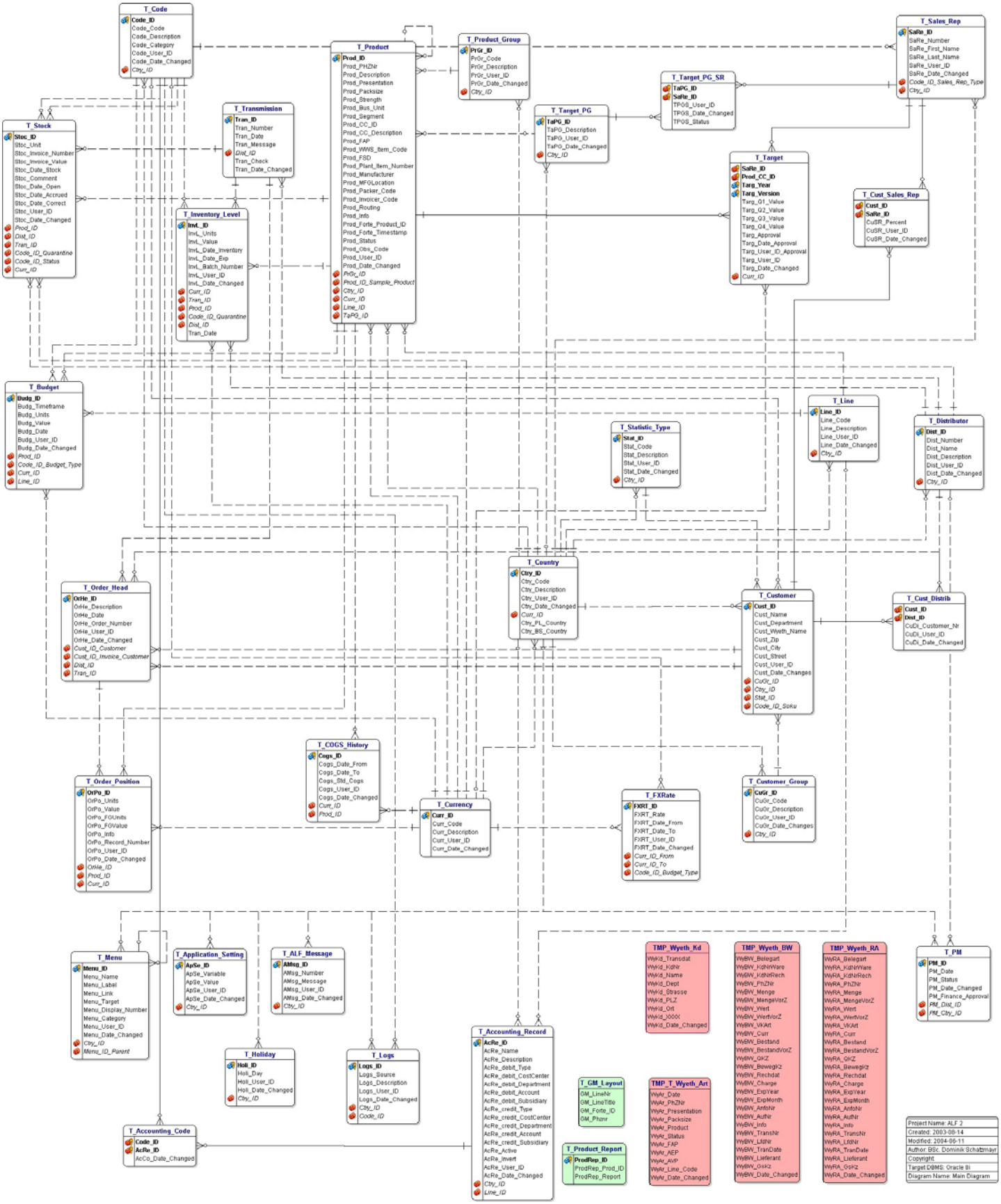
2 Data Model (Entity Relationship Diagram)

2.1 Introduction

This chapter shows the Entity Relationship Diagram and highlights the different areas of its usage.

As the whole model will be hard to understand the chapter Entity Areas will show the tables and their connection in the context of certain functions in the ALF-System.

2.2 Whole Model



2.2.1 Introduction

To understand in which tables which information is stored this chapter will shortly highlight tables and their information.

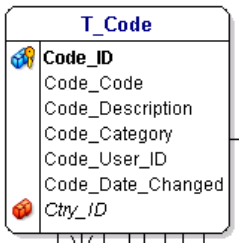
2.2.2 Entity Areas

2.2.2.1 Master Data

The following tables hold master data which need to be administered (**manually**) in the ALF-System. This data is not automatically imported or created!

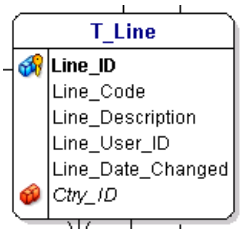
2.2.2.1.1 General

T_Code



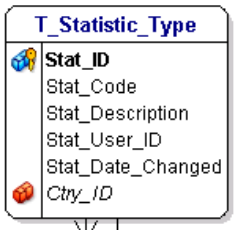
This table contains all codes which are used to highlight the special purpose of entries (i.e. the BewegKZ or the QKZ of the Sanova sales data or the budget types (BU or BE)...)

T_Line



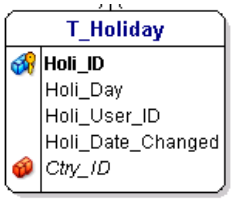
The line is a type of special code which marks Budget, Product and Accounting Record entries if they are (for) Actuals, Samples, Promotional or Clinical Studies purpose.

T_Statistic_Type



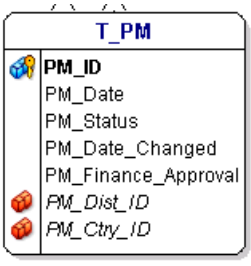
The statistic type was thought to flag the line of customers if it is a sales customer or a sample customer. This was the old system of the Sanova accounting system but during the implementation of the ALF-System Sanova changed to a new system, due to this the statistic type is now not used.

T_Holiday



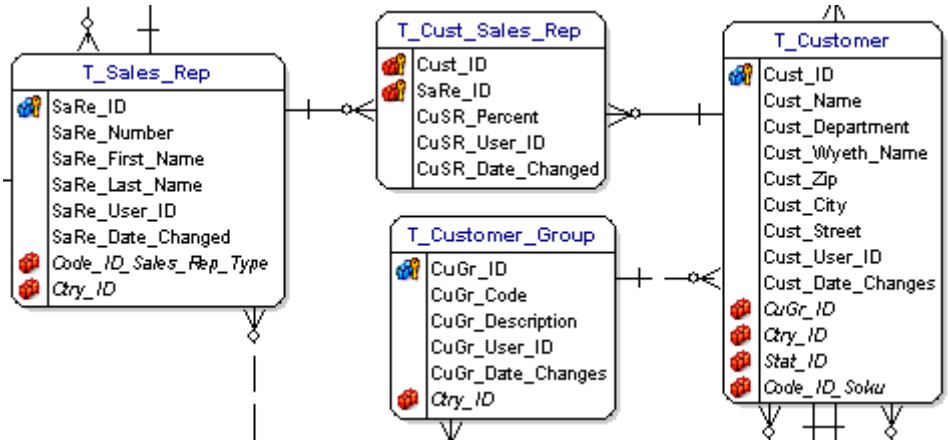
This table is needed for calculating the working days of the month and the working days of the whole year for several reports which linearly extrapolate the Year-To-Date sales to the whole year.
It holds all holidays and weekends of the year.
Currently, all holidays until 2016 are inserted. To insert more years, the database package **PKG_Calendar** can be used.

T_PM



PM stands for Processing Months. This table shows which months are already processed and holds date-flags for the different approvals (finance, Logistics and JDE approval) and information about the users who approved the month.

2.2.2.1.2 Relation Customer - Sales Rep



T_Sales_Rep

Sales Reps are stored in this table. The field SaRe_Number holds the User ID of the Intranet users table! This information is needed for the connex to their mobile phone numbers for the TPG Quick Statistics report, which provides the functionality to send the sales values as an SMS to the sales reps.

T_Customer

This table is described in the chapter 'Sales and Logistics Data from Sanova'.

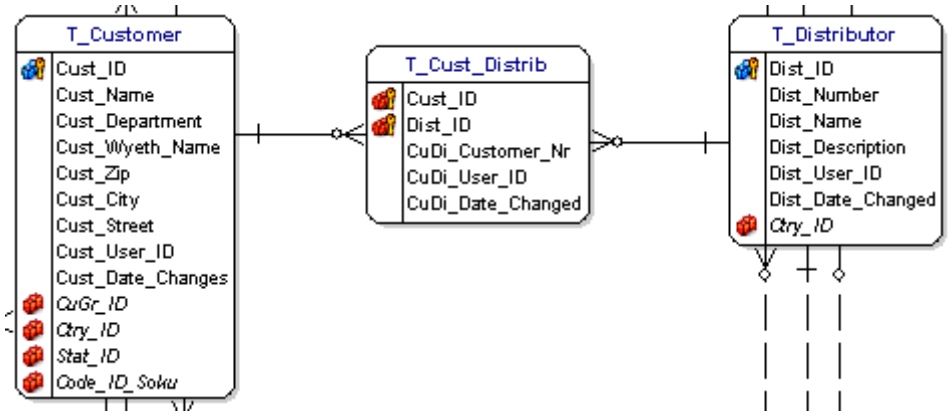
T_Cust_Sales_Rep

The T_Cust_Sales_Rep table is the link table between sales reps and customers and holds the percentage by which customers are allocated to sales reps. Normally, the customers are assigned for 100% to a single sales rep but specific customers (such as hospitals) are served by more than one sales rep and then a certain number of sales reps share this customer. No customer can be assigned for more or less then 100% - ALF will warn and show the actually assigned percentage.

T_Customer_Group

The customer groups shows if the customer belongs to the group of hospitals, pharmacy, doctors, ... etc.

2.2.2.1.3 Relation Customer - Distributor (Customer Number)



T_Distributor

The current (only) distributor is Sanova.
The system can handle indefinite customers, given their reported data is compatible with the ALF model.

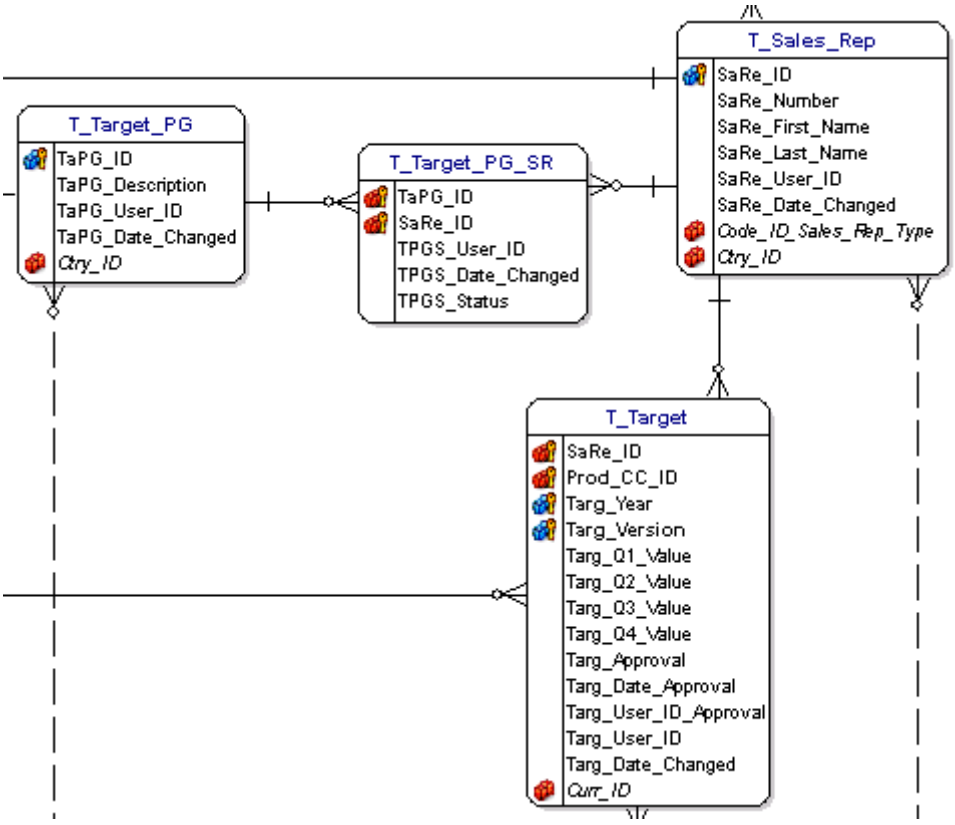
T_Cust_Distrib

If different distributors had the same customers, they would have different numbers for the customers, so the customer number is stored in this link table between customer and distributor.

T_Customer

This table is described in the chapter 'Sales and Logistics Data from Sanova'

2.2.2.1.4 Targeting



T_Target_PG

The Target Product Group is a virtual set of products (hospital classic, Enbrel, ...) which is used for reporting and targeting per se, providing business-defined product grouping.

T_Target_PG_SR

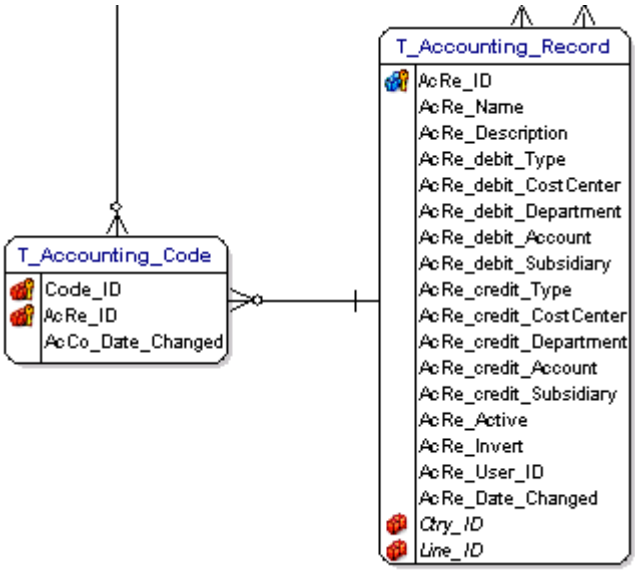
The Target Product Group Sales Rep table is the link table between Target Product Group and Sales Reps. The current situation is that each sales rep belongs to only one target product group, although the system would be able to handle multi-related sales reps.

T_Target

The target defined in ALF means the a sales target for a sales rep; sales meaning the proportional sales of the customers of a sales rep.

In one target entry there are 4 values (for each quarter) of a year. Due to the possibility of different versions, a version needs to be approved before it is displayed in reports.
The latest approved version is that one that counts and which is used for statistical reports.

2.2.2.1.5 Accounting



T_Accounting_Record

Accounting Records are the records extracted from the Finance Month End Report (former Stock & Cogs Report), which will be transferred to the JDE system. The following parameters are used to control the system:

Active: only active records will be exported.
Invert: if the sum values need to be multiplied with (–1) for credit / debit booking
Line_ID: which types of products should be used (also booking records of samples can be established)

Example accounting records:

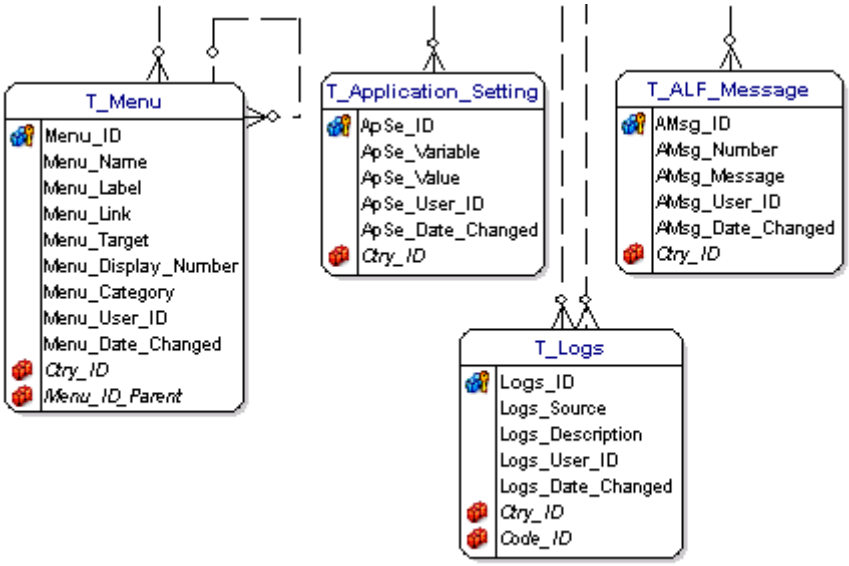
- Cost of Goods Sold
- Cost of Free Goods
- Accruals Open
- etc…

T_Accounting_Code

This is the link table between the accounting records and the codes (in this case the BewegKZs *), as accounting records can only receive values of BewegKZs from the Finance Month End report) that should be taken from the Stock & Cogs Report.

*) BewegKZs = "Bewegungskennzeichen" = type of booking entry of Sanova Logistics System

2.2.2.2 Application Tables



T_Menu

This table holds the menu of the ALF Intranet navigation.
The self-reference (Menu_ID_Parent) is used to define an entry as a child entry of a parent.

T_Application_Setting

The application-setting table holds parameters for the whole ALF-System, such as link, user and password for the FTP Sanova account, which is used by the ALF service to download the sales files, etc.

T_Logs

The T_logs table was originally intended to hold internal logs for the admins to have an insight of what is happening. But it is now also used to inform the users of new database entries which need to be administered or in case errors occurred during importing data.

2.2.2.3 Temp Tables

The TMP_ tables are used for the import of 3rd party information (currently only from Sanova) as described in the chapter Data Import.

TMP_Wyeth_BW

Used for the Wyeth_BW data file (as described in Data Import --> Transaction data (Bewegungsdaten))

TMP_Wyeth_RA

Used to archive entries of the TMP_Wyeth_BW with BewegKZ = RA and GSKZ = IF.

TMP_Wyeth_KD

Used for the Wyeth_KD data file (as described in Data Import --> Customer Data)

TMP_Wyeth_ART

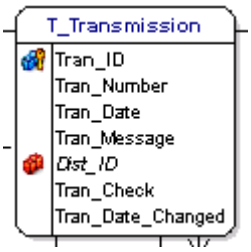
Used for the Wyeth_ART data file (as described in Data Import --> Product Data)

T_Product_Report

Used to parameterize different reports in order to control which products should be reported and which not. Currently, only the Premarin Report is parameterized. E.g. if you want an additional product on the Premarin Report you need to add this product ID in the T_Product_Report table with the ProdRep_Report = 'Premarin'.

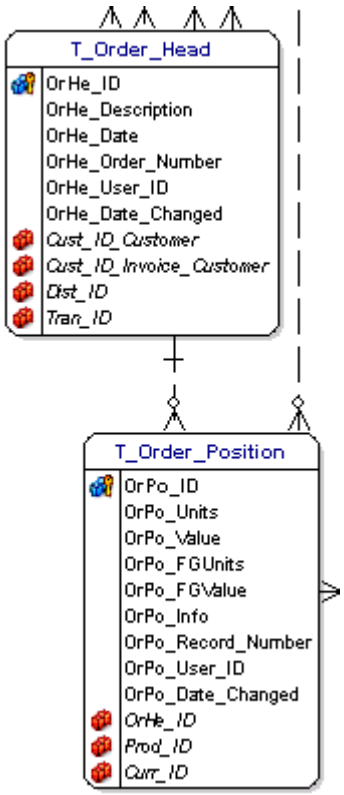
2.2.2.4 Sales and Logistics Data from Sanova

T_Transmission



Sanova provides the sales data in whole transmission files on their FTP-server. Every file should have a different transmission ID number. In order to delete a whole transmission and all its data distributed in the sales and Logistics tables, only the single transmission entry in this table needs to be deleted. All dependent information will be deleted automatically.

Orders



Each order is entered into two tables: T_Order_Head and T_Order_Position.

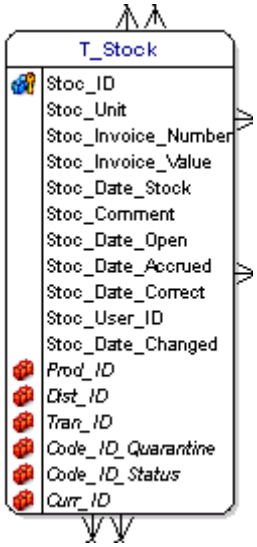
T_Order_Head

The order head is the information of an order which is used for all its positions, such as the customer, the order number, date, ... etc.

T_Order_Position

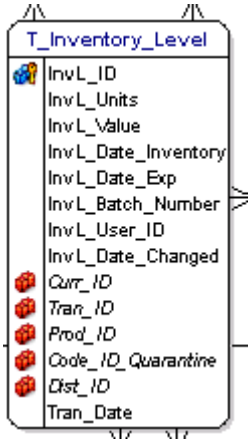
The order position is a link table between T_Order_Head and T_Product. It further holds information of quantity, value and FG (Free Goods) amount of the specific product order.

T_Stock



The stock table identifies stock movements (Lagerbewegungen) which are no sales, such as product delivery (Wareneingang), or damages, or changes due to the accounting processes, etc.

T_Inventory_Level



The inventory level shows the quantity of a specific product on stock. Errors encountered in the past were mainly due to Sanova failed to send the inventory level of ALL products, but only those which had a change in the inventory level. Therefore it is not possible to know if a product is out of stock or if there was no movement. The Stock Check report handles such inconsistencies without any problems.

ALF sets the inventory level for all products to zero, starting 2003-01-01 (initial date of the ALF-System) for all reports to run correctly.

2.2.2.5 Master Data from Sanova

T_Customer

In the Wyeth_KD FTP file of Sanova all customers relevant for the transmission are included. If a customer is missing in the file and therefore also in the T_Customer table and there is an order for this specific customer, an appropriate error message will be displayed at the start page of the ALF-System.

T_Product (also is in the Master Data FORTE area)

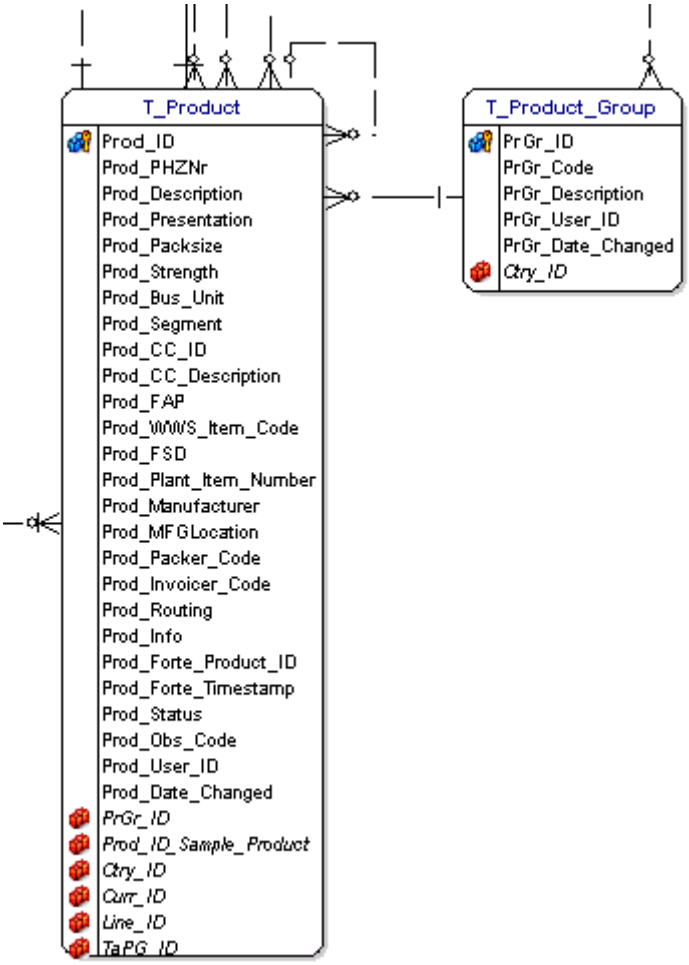
Following the same approach as for T_Customer, the Wyeth_ART FTP file includes all products which are relevant for the transmission.
Sanova does not send all information which is stored in the T_Product table (i.e. CostCenter information, OBS_Code, etc), the majority of product data is imported from the FORTE System.

2.2.2.6 Master Data from FORTE

FORTE Master Data is imported with the PKG_Import_FORTE which sequentially scans predefined FORTE Views for new information, and if something is new, imports it via a database link (FORTE.wyeth.labs.com). The development system also links to the FORTE live system, as there are no update statements, only select statements.

The following database objects and areas are all imported from the FORTE system:

T_Product and T_Product_Group



T_Product

T_Product holds products from the FORTE System and Sanova (both deliver different information). The following fields are important:

Phznr:	Product number which is unique (used to match FORTE and Sanova data)
Obs_Code:	With the Obs_Code (stands for obsolete) (from the FORTE system) products can be hidden.
CC_ID	The cost center is the base for planning targets (the 4 quarter targets are defined for a specific CC, sales rep and year)
Sample Prod:	The Prod_ID_Sample_Product shows the sample product(s) of a sales product.
FORTE TimeSt:	By means of the FORTE_Time_Stamp it is possible to determine whether the entry comes from the FORTE System (can also be an updated Sanova entry) or from Sanova (without update from FORTE)

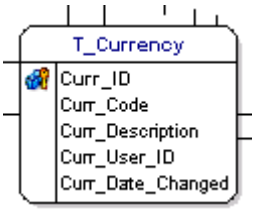
T_Product_Group

Holds defined product groups of the FORTE system. FORTE product groups are automatically imported, whereas for Sanova product data the product groups are set on 'XXX – undefined' and need to be administered manually.

T_Country

Since ALF is designed to serve many countries, the country entity is used to identify the origin of different database objects. Currently, the system only runs for Austria but it is designed for international use.

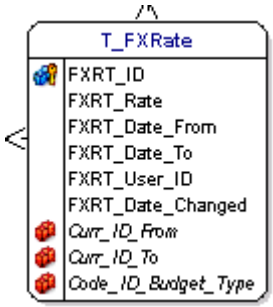
T_Currency



In an international context, all entries which hold (money) values have a link to the currency table in order to identify the currency of that value.

The GM_Report is currently the only multi-currency report.

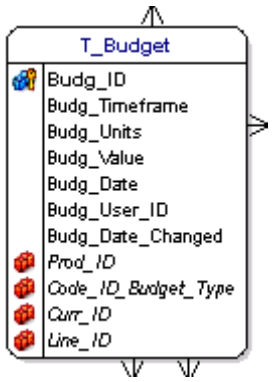
T_FXRate



For calculating values from a specific currency into another the FX-Rate table holds the different exchange rates (to be synchronous with the FORTE system).

The whole table holds the currency cross rates currently available in FORTE.

T_Budget



The budget table holds budget values for a specific product and month. The budget type tells (in general) if the budget value is a BE (Best Estimate) or a BU (Budget) value.

T_COGS_History

COGS (Cost of Goods) are IC costs taken from the "RPIS_Currentprices" view and holds data historically. As the "currentprices" view of RPIS only holds current prices, the ALF-System imports those and on change of the old entry creates a new and current entry with the new value.
For the import it is also possible to identify historic price changes and to import them correctly.

T_GM_Layout

Since the GM (General Manager) Report needs to follow a specific line item format, the T_GM_Layout table holds this format and shows which products are grouped in which line (LineNr) with which heading (title).

3 Data Input

3.1 Introduction

There are three possibilities of data input:

Through 3rd party distributor (currently only Sanova):

Automatically via FTP (this process could also be started manually by administrators if the data was not send via FTP).
The distributor data is primarily sales, and customer data.

Through the FORTE interface via database link:

Periodically (every night) the ALF-System checks the FORTE system and updates all changes of data (this process could also be started manually).

Through manual input into the ALF application

Data that is uniquely stored in the ALF-System (data which is not available from neither 3rd party nor FORTE) must be administered through the application. This part is accessed through Master Data and Admin menus and concerns data such as special additional information on products, customers, sales reps and targets, etc.

3.2 3rd Party Data Import

3.2.1 Introduction

The 3rd party data import includes three different types of data:

- Customer data
- Product data
- Sales and Logistics data, also called 'Bewegungsdaten' (transactional data)

Responsible for the import into the database is the ALF-Service (see ALF-Service) which is working with the SQL-Loader tool of Oracle and the stored procedure PKG_Import_Sanova (see Packages –> PKG_Import_Sanova)

3.2.2 FTP – Server

The data file of Sanova is located on the following FTP server of Sanova/Herba/Chemosan:

URL: ftp://ftp.herba-chemosan.at
User: wyethlp
Password: wy2002acc

The file of the current day is stored in this folder**: /ftp/hcftp/wyethlp/NEW_DATA**

On the next day (or upon creating new data files) the files of the folder will be moved into this folder: **/ftp/hcftp/wyethlp/NEW_DATA/ARCHIV** and the file extension will get a continuous number.
(i.e. the first will be extended with xxx.1 and the second file will have xxx.2 and so on)
One transmission for a day is below 200KB this means the transmission files for a whole year take less than 50MB size of the server.

3.2.3 Customer data

3.2.3.1 Introduction

The name of the data file for customer data on the FTP server is: **WYETH_KD.DAT**.

3.2.3.2 Data File Definition

The file is structured in the following way:

TRANSDAT	POSITION(1:8)	DATE(8)	"YYYYMMDD", *Transmission-Date*
KDNR	POSITION(10:14)	CHAR(5)	*Customer number as used by Sanova*
NAME	POSITION(16:43)	CHAR(28)	*Customer name*
DEPT	POSITION(45:72)	CHAR(28)	*Department (or Name2)*
STRASSE	POSITION(74:101)	CHAR(28)	Street
PLZ	POSITION(103:106)	CHAR(4)	*ZIP Code*
ORT	POSITION(108:131)	CHAR(24)	City
XXXXXX	POSITION(133:138)	CHAR(6)	***this field is not filled & not used***

The file is semicolon-separated.

3.2.3.3 Import Table

ALF-Service imports the file into the **TMP_T_WYETH_KD** table (following the filename WYETH_KD.DAT) without changing the layout. Only the data types of the different fields are set. (eg. the "YYYYMMDD" string of the date will be converted into a real date, etc.)

3.2.3.4 Data Table

The stored procedure **`PKG_IMPORT_SANOVA.F_Customers`** imports the customer data from the temporary import table into the **T_CUSTOMER** table and the **T_CUST_DISTRIB** table, where the customer number is stored (since different distributors could have the same customer with different customer numbers). For details see the Packages chapter)

3.2.4 Product data

3.2.4.1 Introduction

The name of the data file for customer data on the FTP server is: **WYETH_ART.DAT**.

3.2.4.2 Data File Definition

Structure:

```
TRANSDAT      POSITION(1:8)       DATE(8)       "YYYYMMDD", Transmission-Date
PHZNR         POSITION(9:14)      CHAR(6)
ARTBEZ        POSITION(15:54)     CHAR(40)
PACKMNG       POSITION(55:64)     CHAR(10)
KURZBEZ       POSITION(65:94)     CHAR(30)
STATUS        POSITION(95:95)     CHAR(1)       unused
FAP(DAP)      POSITION(96:102)    NUMBER(7)
AEP           POSITION(103:109)   NUMBER(7)
AVP           POSITION(110:116)   NUMBER(7)
LIEFCDE       POSITION(117:121)   CHAR(5)       Line (act, sam, pro) *)

*) "Lieferantencode"
```

LIEFCDE:

This abbreviation is standing for 'Lieferanten Code' and tells (in a code) if the product is an actual (salable), sample, promotional, etc. product.

Code	Description	Alf-code
34547	Salable goods	ACT
84547	Samples	SAM
64547	Promotional products	PRO
M2800	Clinical studies	CLS
84548	Test Samples ("Testplättchen")	TSM

3.2.5 Transaction data (Bewegungsdaten)

The name of the data file for the transaction data on the FTP server is: **WYETH_BW.DAT**.

3.2.5.1 Data File Definition

Structure:

```
BELEGART      POSITION(01:02)     CHAR(2)       document type
KDNRWARE      POSITION(03:07)     CHAR(5)       deliver customer
KDNRRECH      POSITION(08:12)     CHAR(5)       invoice customer
PHZNR         POSITION(13:27)     CHAR(14)      product number
MENGE         POSITION(28:36)     NUMBER(9)     units
MENGEVORZ     POSITION(37:37)     CHAR(1)       sign (+/-)
WERT          POSITION(38:50)     NUMBER(13)    value (need to div by 100)
WERTVORZ      POSITION(51:51)     CHAR(1)       sign (+/-)
VKART         POSITION(52:52)     CHAR(1)       type of sale: N=free goods
CURR          POSITION(53:55)     CHAR(3)       currency
BESTAND       POSITION(56:64)     NUMBER(9)     inventory level
BESTANDVORZ   POSITION(65:65)     CHAR(1)       sign (+/-)
QKZ           POSITION(66:66)     CHAR(1)       quarantine code
BEWEGKZ       POSITION(67:68)     CHAR(2)       transaction code
RECHDAT       POSITION(69:76)     DATE(8)       "YYYYMMDD", invoice date
CHARGE        POSITION(77:91)     CHAR(15)      batch number
EXPYEAR       POSITION(92:95)     CHAR(4)       expiring year
EXPMONTH      POSITION(96:97)     CHAR(2)       expiring month
ANFONR        POSITION(98:107)    CHAR(10)      unused Astra Zeneca field
AUFNR         POSITION(108:113)   CHAR(6)       job account number
```

```
INFO        POSITION(114:138)   CHAR(25)      unused additional info
TRANSNR     POSITION(139:146)   CHAR(8)       transmission number
LFDNR       POSITION(147:151)   CHAR(5)       sequential number in file
TRANSDAT    POSITION(152:159)   DATE(8)       "YYYYMMDD", transmission-Date
LIEFERANT   POSITION(160:164)   CHAR(5)       line (act, sam, pro)
GSKZ        POSITION(165:165)   CHAR(1)       credit advice code: G=credit
```

BELEGART (document type)

The field document type identifies the purpose of each record.

Code	Name	Description
FA	1) Sales	Invoice, credit, or rebooking via special customer
LB	2) Inventory Level	Current inventory level
LG	3) Stock (movement)	Movement of products which are no sales (i.e. product delivery into the stock, …)
RA	4) Arrear	Orders that could not be fulfilled… These have no relevance to the ALF-System but are archived in the **TMP_Wyeth_RA** table
SM	5) Sum month	A sum entry of the whole month (all send transmissions in the month) to check if the send data is consistent.
ST	6) Sum day *)	A sum entry of the current transmission to check if the send data is consistent.

*) 1) = Faktura, 2) = Lagerbestand, 3) = Lagerbewegung, 4) = Rückstandsauftrag, 5) = Summe Monat, 6) = Summe Tag

QKZ (quarantine code)

This code tells the stock status of the stock entry, i.e. if this product is locked in the stock, or not saleable, … but this information is not relevant for the import and only reported in the Stock for Iris Reporting.

BewegKZ (transaction code)

This code identifies the type of transaction (is it a positive, negative or neutral sale, etc.) of the particular sales or stock entry.

In the old system of Sanova there were Special Customers, which indicated the type of transaction, but in our new system there is only the transaction code. The systems changed while developing ALF, therefore all ‘old’ special customer data are mapped to the specific transaction code.
The ALF-System is currently only working with the transaction codes.

SCNr: Special Customer Number

Code	Description	Reason	Stock	SCNr.
BR	Damages	Damages in the stock	**yes**	07601
AZ	Wyeth samples	Indicates that sales goods where made to sample good	**yes**	07602
UM	Samples stock	Contra entry to the entry above (new samples into the stock)	**yes**	07603
AW	Valueless goods	Accounting for goods without value (expired goods,…) (no contra entry!)	**yes**	07604

IN	Inventory difference	Inventory accounting	**yes**	07605
DN	Pro-Depot/ emergency stock	Special Sanova accounting	**yes**	07606
Sales	Exchanged goods	Special Customer without impact	**yes**	07607
	Exchanged goods Wyeth	Obsolete;	**n/a**	07608
n/a	Return goods	Back goods into stock		07609
WU	Goods in transit	Goods in transit	**yes**	07610
TS	Transport damage	Transport damage of delivery	**yes**	07612
Sales	Expired goods from customers	Expired goods from customers – not into the stock	**no**	07617
HM	Evidence sample	Sales to Wyeth quality management	**yes**	07618
KO	Adjustment	Adjustment of sales goods (i.e. repacking of 10s to 1s units)	**yes**	07619
Sales	Hospital without deduction	Goods back from hospitals (not into stock)	**no**	07621
AL	Valueless goods	Same as AW	**yes**	-
WE	Goods delivery	Goods delivery from distributor	**yes**	-

GSKZ (credit advice code / "Gutschriftenkennzeichen")

The credit advice code field identifies the type of credit advice granted.

Code	Description	Stock	Reason
R3	Price correction	**yes**	Wrong price charged
R2	Wrong order acceptance	**yes**	Wrong product or customer, etc…
G6	Cancellation due to customer error	**yes**	Cancellation due to customer error
G9	Expired goods with 40% deduction	**yes**	Expired goods from whole sales
G7	Expired goods with 0% deduction	**yes**	Expired goods from hospitals
G2	Recall	**yes**	Batch recall
G5	First inventory filling	**yes**	Redemption from first inventory filling
G4	Others	**yes**	Standard code for credit advice
IF	Information	**n/a**	Dummy entry, stored in **TMP_Wyeth_RA**

3.2.5.2 Import Table

The ALF-System imports this file into the **TMP_T_WYETH_BW** table.

3.2.5.3 Data Table

Using the PL-SQL stored procedure **`PKG_IMPORT_SANOVA.F_BW`** the data is inserted into **T_Transmission, T_Order_Head, T_Order_Position, T_Stock, T_Inventory_Level and TMP_Wyeth_RA** (where rows with the document type (Belegart) = RA and the GSKZ = IF are archived).

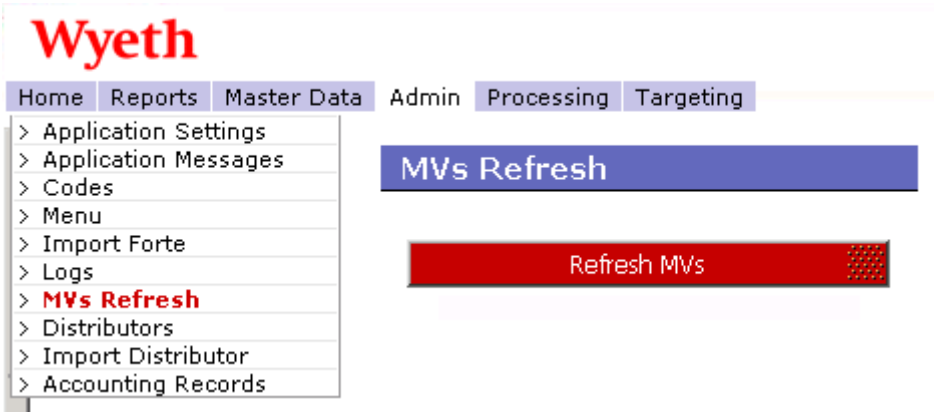
In case there is uninsertable data, e.g. if the specific code, product or customer is missing in the database, there is a second stored procedure **`PKG_IMPORT_SANOVA.F_add_broken_BW`** for inserting the remaining data after the missing element has been added. This process needs to be started manually! This is not working automatically!

3.2.6 Materialized Views behavior

3.2.6.1 Introduction

After the import process the filled tables hold new information, and in order to update the reports (which are based on aggregated information) the materialized views need to be refreshed. The ALF-Service running on the server starts this process automatically, but in case they need to be refreshed, this could be done in two ways:

- Directly in Oracle, starting the **`PKG_IMPORT_SANOVA.F_Refresh_MVs.`**
- Or through the application in the menu **Admin --> MVs Refresh**



3.3 FORTE Import

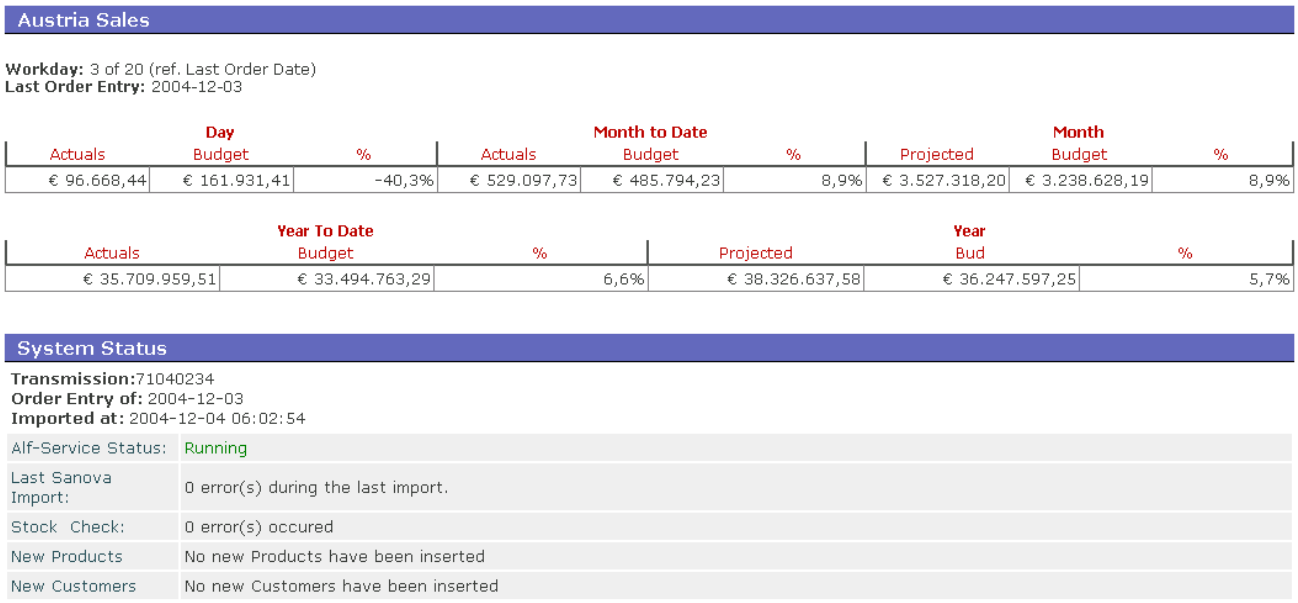
For details see Data Models –> Whole Model –> Entity Areas –> Master Data from FORTE.

4 Start page

4.1 Introduction

The start page of the ALF system gives a short overview of the most important sales and maintenance conditions of the system.

If any new data has been imported which needs user administration the system will highlight this on the start page and give links directly to the master data interface to administer those entities.



Austria Sales

Workday: 3 of 20 (ref. Last Order Date)
Last Order Entry: 2004-12-03

Day			Month to Date			Month		
Actuals	Budget	%	Actuals	Budget	%	Projected	Budget	%
€ 96.668,44	€ 161.931,41	-40,3%	€ 529.097,73	€ 485.794,23	8,9%	€ 3.527.318,20	€ 3.238.628,19	8,9%

Year To Date			Year		
Actuals	Budget	%	Projected	Bud	%
€ 35.709.959,51	€ 33.494.763,29	6,6%	€ 38.326.637,58	€ 36.247.597,25	5,7%

System Status

Transmission:71040234
Order Entry of: 2004-12-03
Imported at: 2004-12-04 06:02:54

Alf-Service Status:	Running
Last Sanova Import:	0 error(s) during the last import.
Stock Check:	0 error(s) occured
New Products	No new Products have been inserted
New Customers	No new Customers have been inserted

The start page is divided into the following two areas:

- Austria Sales
- System Status

4.1.1 Austria Sales

The Austria Sales area shows detailed sales information of the sales and the budget for…

- Today
- The current sales of the month
- The projected sales for the whole month
- The current sales of the year
- The projected sales for the whole year

The last order entry shows the date of the last order registered in the system.
The header "Workday" shows how many days are already gone by in the respective month. This information is needed to calculate the day-budget and the month-to-day-budget.

4.1.2 System Status

The system status area provides important maintenance information, with instantly shows if any relevant errors occurred or not.

Transmission:	The number of the last transmission from Sanova.
Order Entry of:	Date of the last order in the last transmission.
Imported at:	Date and time of when the last transmission was imported.
Alf-Service Status:	Status of the ALF Service, if it is running or not.
Last Sanova Import:	Shows if errors occurred while importing.
Stock Check:	Shows if the Stock Check found stock inconsistencies.
New Products	Shows if new products, which need data administration, where imported, with direct links to the product master data interface.
New Customers	Shows if new customers, which need data administration, where imported, with direct links to the customer master data interface.

5 Reports

5.1 Introduction

This chapter will describe all reports currently available in the ALF-System, detailed in their technical aspects.

The structure of the reports will be specified, describing which column holds which data and how different calculations (e.g. percentages) are made.

The construction of the reports will be identified, explaining which materialized views, tables and regular views used, in order to be able to know which part of the system to touch if troubles occur with a report.

5.2 How is the asp.net Data Grid filled?

The data grid receives its data from a stored procedure in the database. The .aspx logic starts a specific procedure (each report has its own stored procedure) with one reference cursor parameter (which holds the data which will be presented) and the parameters set from the user (e.g. reporting month), line (ACT, SAM), etc.

The stored procedure transmits a select statement to the database, which in return opens and fills the reference cursor.

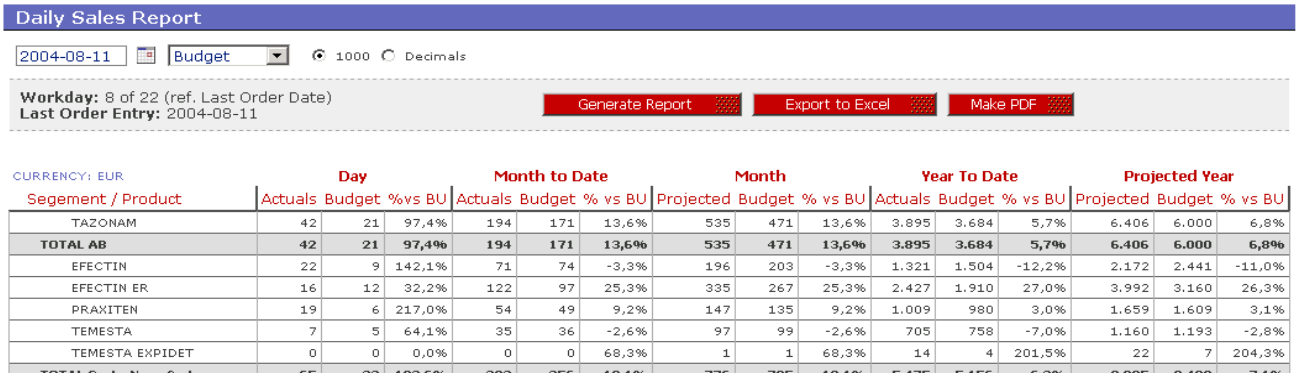
In general the database select statements for all reports, except stock reports (found in **PKG_Stock**) and targeting reports (found in **PKG_Targeting**) can be found in the **PKG_Reports** database package.

5.3 Sales Reports

5.3.1 Daily Sales Report

5.3.1.1 Introduction

The Daily Sales Report displays detailed sales information on a daily updated basis. It is by far the most important report for the Austrian sales organisation!



CURRENCY: EUR	Day			Month to Date			Month			Year To Date			Projected Year		
Segement / Product	Actuals	Budget	%vs BU	Actuals	Budget	% vs BU	Projected	Budget	% vs BU	Actuals	Budget	% vs BU	Projected	Budget	% vs BU
TAZONAM	42	21	97,4%	194	171	13,6%	535	471	13,6%	3.895	3.684	5,7%	6.406	6.000	6,8%
TOTAL AB	**42**	**21**	**97,4%**	**194**	**171**	**13,6%**	**535**	**471**	**13,6%**	**3.895**	**3.684**	**5,7%**	**6.406**	**6.000**	**6,8%**
EFECTIN	22	9	142,1%	71	74	-3,3%	196	203	-3,3%	1.321	1.504	-12,2%	2.172	2.441	-11,0%
EFECTIN ER	16	12	32,2%	122	97	25,3%	335	267	25,3%	2.427	1.910	27,0%	3.992	3.160	26,3%
PRAXITEN	19	6	217,0%	54	49	9,2%	147	135	9,2%	1.009	980	3,0%	1.659	1.609	3,1%
TEMESTA	7	5	64,1%	35	36	-2,6%	97	99	-2,6%	705	758	-7,0%	1.160	1.193	-2,8%
TEMESTA EXPIDET	0	0	0,0%	0	0	68,3%	1	1	68,3%	14	4	201,5%	22	7	204,3%

5.3.1.2 Structure

Report Layout:

Day: Sales and budget (BE) of the specific working day. As the Budget values in the database are only month values, the month value is divided through the working days of the chosen month.

Month to Date: Sum of sales value up to the selected day of month. Budget daily values up to the selected date (the monthly budget value divided by the number of days in the month, and multiplied with number of days, month to day)

Month: Projected Sales Value (Month to Date Sales divided by the number of days to-date, multiplied with the number of days in the month). The budget value is derived from the FORTE system.

Year to Date: Same as the Month to Date but not the summed up values of the month of the selected date but the summed-up values of the whole year up to the selected date.

Projected Year: Same as the Month values but not projected for the month of the selected date but for the whole year. The budget value is the summed up value of the budget values of the year.

Percentages: The percentage values show the difference between sales and budget. E.g. if the percentage is 0% then sales are equal to the budget; if the percentage is 50% then the sales are 50% higher than the budget, e.g. actuals 150 at a budget of 100) and if the percentage is –50% then the sales are 50% lower than the budget (actuals 50, budget 100).

5.3.1.3 Construction

The stored procedure **`PKG_REPORTS.GetDailySales`** is started with the following parameters:

- `Date (of the reporting day)`
- `Workdays (of the month)`
- `WorkdaysUntilToday (of the month)`
- `Budget (BU or BE)`

The procedure calculates the following variables to project the values for the whole year:

- `workdaysUntilToday_Year`
- `workdays_Year`

The columns are used in the following hierarchy of the MVs, views and tables:

- **MV_NET_SALES**
  - ↳ **MV_DAILY_SALES5**
    - ↳ **MV_SALES_PER_DAY2**
      - ↳ **V_SALES_PER_DAY2**
        - ↳ T_LINE
        - ↳ T_PRODUCT_GROUP
        - ↳ T_ORDER_POSITION
        - ↳ T_ORDER_HEAD
      - ↳ **V_TRANSMISSION_PRODUCT**
        - ↳ T_TRANSMISSION
        - ↳ **MV_ACTIVE_PRODUCT**
          - ↳ T_PRODUCT

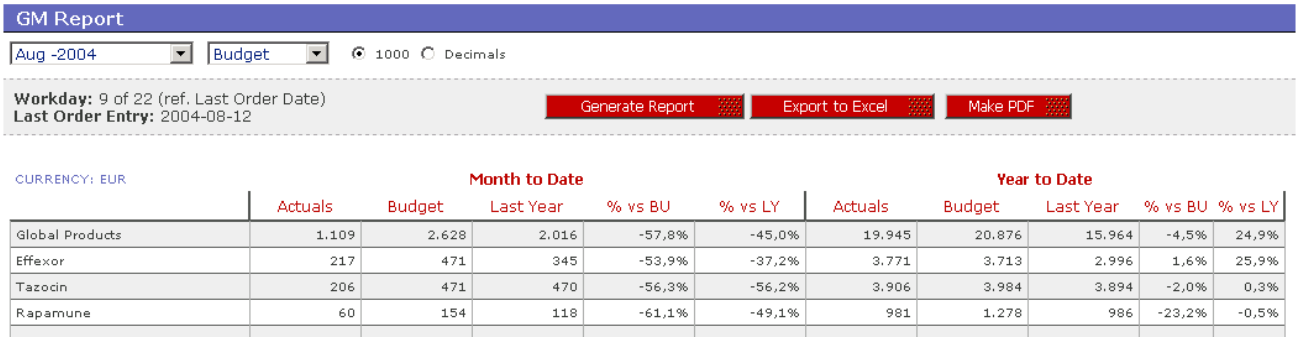
```
                    ↳ T_ORDER_POSITION
                    ↳ T_TRANSMISSION
                    ↳ T_ORDER_HEAD
                    ↳ T_INVENTORY_LEVEL
                    ↳ T_STOCK
                    ↳ T_BUDGET
                ↳ T_CUSTOMER
                ↳ V_FXRATE_EUR_USD_ACT
                    ↳ T_FXRATE
                    ↳ T_CODE
                    ↳ T_CURRENCY
                ↳ T_COGS_HISTORY
        ↳ V_BUDGET_FOR_ALL_PROD_DESC
            ↳ V_BUDGET_FOR_ALL_PROD
                ↳ V_TRANSMISSION_PRODUCT_MONTH
                    ↳ T_TRANSMISSION
                    ↳ MV_ACTIVE_PRODUCT
                        ↳ T_PRODUCT
                        ↳ T_ORDER_POSITION
                        ↳ T_TRANSMISSION
                        ↳ T_ORDER_HEAD
                        ↳ T_INVENTORY_LEVEL
                        ↳ T_STOCK
                        ↳ T_ BUDGET
                ↳ V_BUDGET_BU
                    ↳ T_BUDGET
                    ↳ T_CODE
                    ↳ T_LINE
                ↳ V_BUDGET_BE
                    ↳ T_BUDGET
                    ↳ T_CODE
                    ↳ T_LINE
                ↳ V_FXRATE_EUR_USD_BU
                    ↳ T_FXRATE
                    ↳ T_CODE
                    ↳ T_CURRENCY
    ↳ T_TRANSMISSION
• MV_DAILY_SALES5
    ↳ V_BUDGET_FOR_ALL_PROD_DESC
            ↳ V_BUDGET_FOR_ALL_PROD
                ↳ V_TRANSMISSION_PRODUCT_MONTH
                    ↳ T_TRANSMISSION
                    ↳ MV_ACTIVE_PRODUCT
                        ↳ T_PRODUCT
                        ↳ T_ORDER_POSITION
                        ↳ T_TRANSMISSION
                        ↳ T_ORDER_HEAD
                        ↳ T_INVENTORY_LEVEL
                        ↳ T_STOCK
                        ↳ T_ BUDGET
                ↳ V_BUDGET_BU
                    ↳ T_BUDGET
                    ↳ T_CODE
                    ↳ T_LINE
                ↳ V_BUDGET_BE
                    ↳ T_BUDGET
                    ↳ T_CODE
                    ↳ T_LINE
                ↳ V_FXRATE_EUR_USD_BU
                    ↳ T_FXRATE
                    ↳ T_CODE
                    ↳ T_CURRENCY
        ↳ MV_SALES_PER_DAY2
            ↳ V_SALES_PER_DAY2
                ↳ T_LINE
                ↳ T_PRODUCT_GROUP
                ↳ T_ORDER_POSITION
                ↳ T_ORDER_HEAD
            ↳ V_TRANSMISSION_PRODUCT
                    ↳ T_TRANSMISSION
                    ↳ MV_ACTIVE_PRODUCT
                        ↳ T_PRODUCT
                        ↳ T_ORDER_POSITION
                        ↳ T_TRANSMISSION
                        ↳ T_ORDER_HEAD
                        ↳ T_INVENTORY_LEVEL
                        ↳ T_STOCK
```

- T_BUDGET
- T_CUSTOMER
- V_FXRATE_EUR_USD_ACT
  - T_FXRATE
  - T_CODE
  - T_CURRENCY
- T_COGS_HISTORY

- **MV_SALES_PER_DAY_PROD_DESC**
  - **MV_SALES_PER_DAY2**
    - **V_SALES_PER_DAY2**
      - T_LINE
      - T_PRODUCT_GROUP
      - T_ORDER_POSITION
      - T_ORDER_HEAD
    - **V_TRANSMISSION_PRODUCT**
      - T_TRANSMISSION
      - **MV_ACTIVE_PRODUCT**
        - T_PRODUCT
        - T_ORDER_POSITION
        - T_TRANSMISSION
        - T_ORDER_HEAD
        - T_INVENTORY_LEVEL
        - T_STOCK
        - T_BUDGET
      - T_CUSTOMER
      - **V_FXRATE_EUR_USD_ACT**
        - T_FXRATE
        - T_CODE
        - T_CURRENCY
      - T_COGS_HISTORY
- T_BUDGET
- T_CODE
- T_PRODUCT

5.3.2 GM Report

5.3.2.1 Introduction


GM Report

Aug -2004 | Budget | 1000 | Decimals

Workday: 9 of 22 (ref. Last Order Date)
Last Order Entry: 2004-08-12

Generate Report | Export to Excel | Make PDF

CURRENCY: EUR

	Month to Date					Year to Date				
	Actuals	Budget	Last Year	% vs BU	% vs LY	Actuals	Budget	Last Year	% vs BU	% vs LY
Global Products	1.109	2.628	2.016	-57,8%	-45,0%	19.945	20.876	15.964	-4,5%	24,9%
Effexor	217	471	345	-53,9%	-37,2%	3.771	3.713	2.996	1,6%	25,9%
Tazocin	206	471	470	-56,3%	-56,2%	3.906	3.984	3.894	-2,0%	0,3%
Rapamune	60	154	118	-61,1%	-49,1%	981	1.278	986	-23,2%	-0,5%

The GM (General Manager) Report in the ALF-System follows the GM Report from the FORTE system. It contains exactly the same information and has the same design.

In order to be able to group the information the same way the FORTE system does, the table T_GM_Layout holds the grouping layout from FORTE. The table is updated (deleted and new inserted) through the **`PKG_IMPORT_FORTE.F_GM_Layout,`** which runs every night, together with all other FORTE import procedures.

5.3.2.2 Structure

The GM Report is a monthly report which holds full monthly information. Note that the current month shows month-to-date values, meaning that data only gives partial monthly views until the end of the month.

The header “Workday” shows how many days are already gone by in the respective month.

Report Layout:

Month to Date: Actuals / sum of actual sales of month (respectively, up to the current day of the current month).

Budget: Monthly budget / BE values for the selected month (in case of the current month, the value for the whole month is displayed; no calculations for daily budget values are made.

Last Year: Last Year columns show the sales for the same month of the last year. If reports for 2003 (the ALF-System only holds sales values since January 2003) are selected, the Last Year values show the sales values of the Month to Date (this means the selected month).

Percentage: The Percentage values are the same way implemented as in the Daily Sales Report and show the difference between sales and budget, i.e. the difference between selected month sales and last year month sales.

Year to Date: Actuals / same as the Month to Date but summed up values of the whole year up to the selected month.

Last Year: The Last Year columns also hold Year to Date values of the previous year; e.g. if the selected month is Apr. 2004 than the YtD sales value hold the sales of Jan, Feb, Mar and Apr 2004 and the YtD LY values holds the sales for Jan, Feb, Mar and Apr 2003.)

5.3.2.3 Construction

The stored procedure **`PKG_REPORTS.GetGMReport`** is started with the following parameters:

- `Date (of the reporting month)`

The procedure selects columns of only one MV the **MV_GM_Report2**.

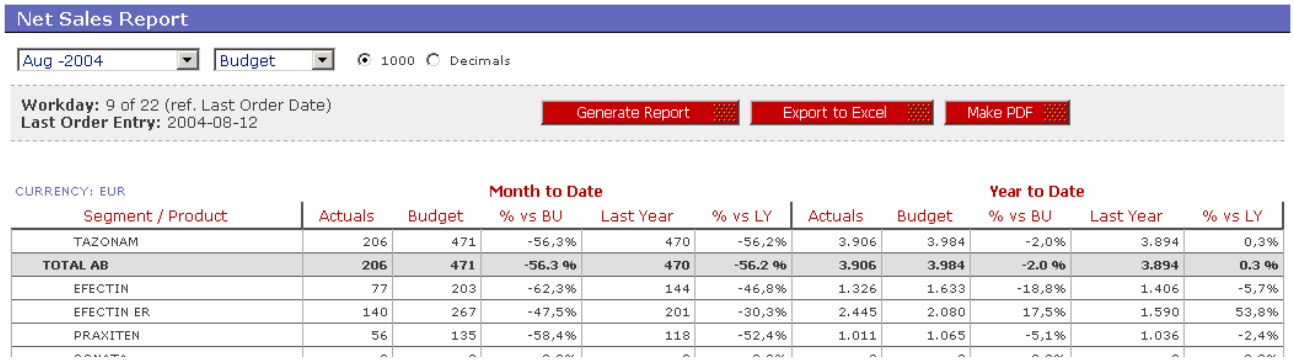
The columns are taken in the following hierarchy of the MVs, views and tables:

- **MV_GM_REPORT2**
  - **V_GM_REPORT_BUDGET2**
    - **V_BUDGET2**
      - **V_BUDGET_FOR_ALL_PROD**
        - **V_TRANSMISSION_PRODUCT_MONTH**
          - T_TRANSMISSION
          - **MV_ACTIVE_PRODUCT**
            - T_PRODUCT
            - T_ORDER_POSITION
            - T_TRANSMISSION
            - T_ORDER_HEAD
            - T_INVENTORY_LEVEL
            - T_STOCK
            - T_ BUDGET
        - **V_BUDGET_BU**
          - T_BUDGET
          - T_CODE
          - T_LINE
        - **V_BUDGET_BE**
          - T_BUDGET
          - T_CODE
          - T_LINE
        - **V_FXRATE_EUR_USD_BU**
          - T_FXRATE
          - T_CODE
          - T_CURRENCY
  - **V_GM_REPORT_PRODUCT2**
    - **MV_SALES_PER_DAY2**
      - **V_SALES_PER_DAY2**
        - T_LINE
        - T_PRODUCT_GROUP
        - T_ORDER_POSITION
        - T_ORDER_HEAD
        - **V_TRANSMISSION_PRODUCT**
          - T_TRANSMISSION
          - **MV_ACTIVE_PRODUCT**
            - T_PRODUCT
            - T_ORDER_POSITION
            - T_TRANSMISSION
            - T_ORDER_HEAD
            - T_INVENTORY_LEVEL
            - T_STOCK
            - T_BUDGET
        - T_CUSTOMER
        - **V_FXRATE_EUR_USD_ACT**
          - T_FXRATE
          - T_CODE
          - T_CURRENCY
        - T_COGS_HISTORY
    - **V_GM2**
      - T_GM_LAYOUT
      - **MV_ACTIVE_PRODUCT**
        - T_PRODUCT
        - T_ORDER_POSITION

```
            ↪ T_TRANSMISSION
            ↪ T_ORDER_HEAD
            ↪ T_INVENTORY_LEVEL
            ↪ T_STOCK
            ↪ T_BUDGET
      ↪ T_TRANSMISSION
```

5.3.3 Net Sales Report

5.3.3.1 Introduction



CURRENCY: EUR

	Month to Date					Year to Date				
Segment / Product	Actuals	Budget	% vs BU	Last Year	% vs LY	Actuals	Budget	% vs BU	Last Year	% vs LY
TAZONAM	206	471	-56,3%	470	-56,2%	3.906	3.984	-2,0%	3.894	0,3%
TOTAL AB	**206**	**471**	**-56.3 %**	**470**	**-56.2 %**	**3.906**	**3.984**	**-2.0 %**	**3.894**	**0.3 %**
EFECTIN	77	203	-62,3%	144	-46,8%	1.326	1.633	-18,8%	1.406	-5,7%
EFECTIN ER	140	267	-47,5%	201	-30,3%	2.445	2.080	17,5%	1.590	53,8%
PRAXITEN	56	135	-58,4%	118	-52,4%	1.011	1.065	-5,1%	1.036	-2,4%

The Net Sales Report holds monthly sales information; the current month data is always shown as Month to Date (as in the GM Report).

5.3.3.2 Structure

Report Layout:

Month to Date: Actual sales and budget of the selected month. If the selected month is the current month (just like in the above screenshot, 12th August) in which not all workdays are over the report shows Month to Date (so from the beginning of the Month until today). If the selected month is historic, the Month to Date rows hold information for the whole month.

Year to Date: Actual sales and budget from the beginning of the year until today (if the selected month is the current one) or the last of the selected historic month.

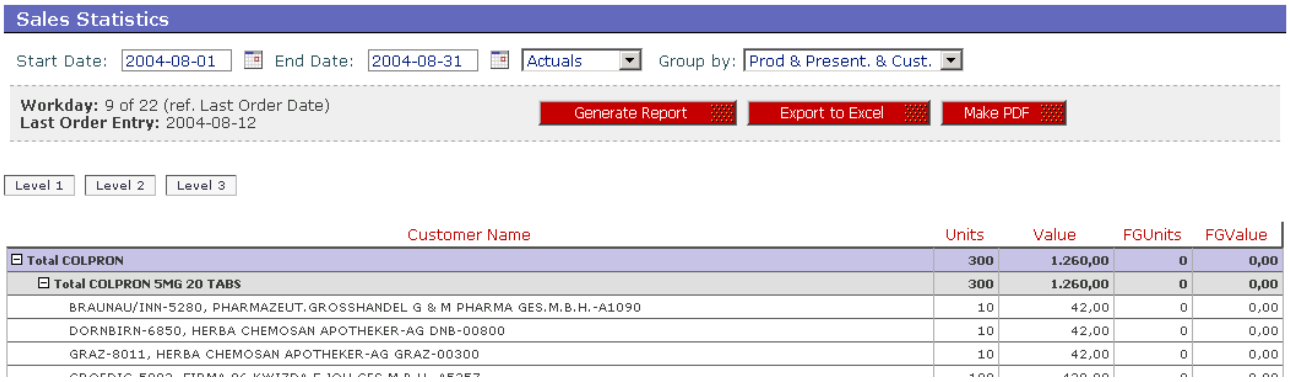
5.3.3.3 Construction

The stored procedure **PKG_REPORTS.GetNetSales** fills the cursor with all fields of the MV_NET_SALES for the selected month and budget type.

- MV_NET_SALES
  - MV_DAILY_SALES5
    - MV_SALES_PER_DAY2
      - V_SALES_PER_DAY2
        - T_LINE
        - T_PRODUCT_GROUP
        - T_ORDER_POSITION
        - T_ORDER_HEAD
        - V_TRANSMISSION_PRODUCT
          - T_TRANSMISSION
          - MV_ACTIVE_PRODUCT
            - T_PRODUCT
            - T_ORDER_POSITION
            - T_TRANSMISSION
            - T_ORDER_HEAD
            - T_INVENTORY_LEVEL
            - T_STOCK
            - T_BUDGET
        - T_CUSTOMER
        - V_FXRATE_EUR_USD_ACT
          - T_FXRATE
          - T_CODE
          - T_CURRENCY
        - T_COGS_HISTORY
    - V_BUDGET_FOR_ALL_PROD_DESC
      - V_BUDGET_FOR_ALL_PROD
        - V_TRANSMISSION_PRODUCT_MONTH
          - T_TRANSMISSION
          - MV_ACTIVE_PRODUCT
            - T_PRODUCT
            - T_ORDER_POSITION
            - T_TRANSMISSION
            - T_ORDER_HEAD
            - T_INVENTORY_LEVEL
            - T_STOCK
            - T_ BUDGET
        - V_BUDGET_BU
          - T_BUDGET
          - T_CODE
          - T_LINE
        - V_BUDGET_BE
          - T_BUDGET
          - T_CODE
          - T_LINE
        - V_FXRATE_EUR_USD_BU
          - T_FXRATE
          - T_CODE
          - T_CURRENCY
  - T_TRANSMISSION

5.3.4 Sales Statistics

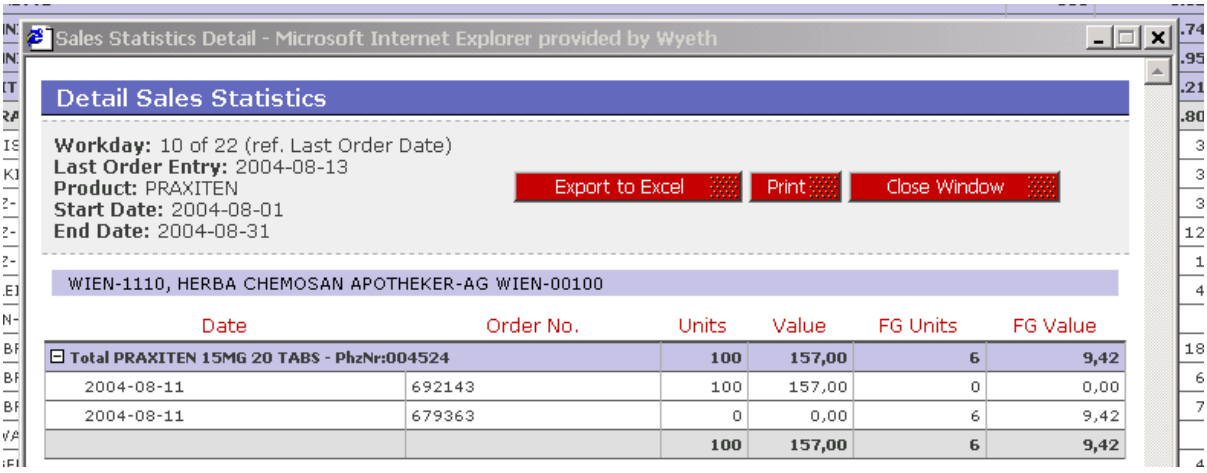
5.3.4.1 Introduction



The Sales Statistics report provides means to track detailed sales information. By clicking on each detail line (product and customer) a pop-up will open, giving detailed information of the orders of the customer and that particular product.

With the "Group by" field it is possible to control the layout and grouping of the displayed data. You can choose between three gourping alternatives:

- Prod. & Present. & Cust.: 1st sums are on the product, then presentation then customer
- Prod. & Cust. & Present.: product – customer – presentation
- Cust. & Prod. & Present.: customer – product – presentation



5.3.4.2 Structure

Report Layout:

Customer Name: The customer name string consists of the following connected database fields (from the T_CUSTOMER table).

CUST_CITY & ', ' & CUST_ZIP & ' ' & CUST_NAME & ' ' & CUST_DEPARTMENT & '-' & CUDI_CUSTOMER_NR

The '&' symbol is the string connection operator.

Units / Value: Sum of Units and Values of the orders of the customer on the product.

FG: FG stands for Free Goods and shows number of units and value of Free Goods attached to the order.

Columns Pop Up:

Date: Date of the Order.

Order No.: Number (AuftragsNr.) of Order

Units / Value: Units and Value of Order.

FG: Free Goods of Order.

5.3.4.3 Construction of the Report

The stored procedure **`PKG_REPORTS.GetSalesStatCustomer2`** is started with the following parameters:

- `StartDate`
- `EndDate`
- `line_id` (Actuals, Samples,…)

The procedure only selects columns of MV_SALES.

- **MV_SALES**
  - ↳ **T_CUSTOMER**
  - ↳ **T_CUSTOMER_GROUP**
  - ↳ **T_CUST_DISTRIB**
  - ↳ **T_ORDER_HEAD**
  - ↳ **T_TRANSMISSION**
  - ↳ **T_ORDER_POSITION**
  - ↳ **T_PRODUCT**
  - ↳ **T_PRODUCT_GROUP**
  - ↳ **T_LINE**
  - ↳ **T_COUNTRY**
  - ↳ **T_CURRENCY**
  - ↳ **T_TARGET_PG**
  - ↳ **T_CODE**
  - ↳ **T_COGS_HISTORY**

5.3.4.4 Construction of the Pop Up

The stored procedure **`PKG_REPORTS.GetSalesCustomerStatDetail`** is started with the following parameters:

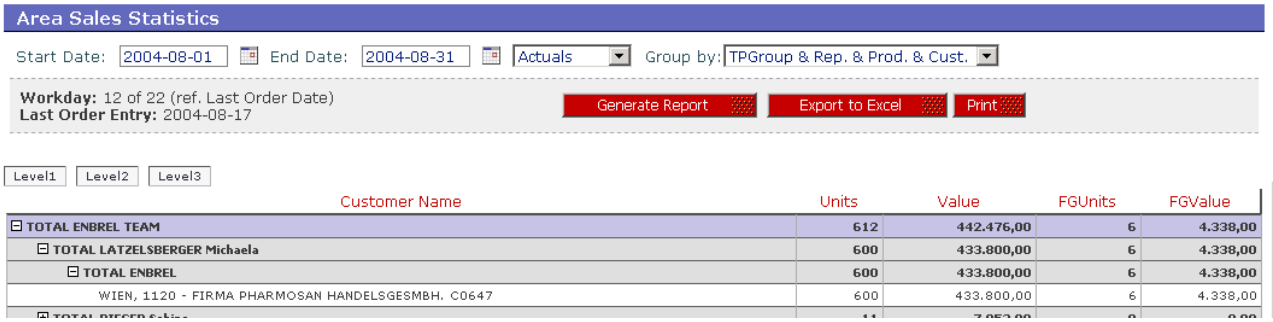
- `StartDate`
- `EndDate`
- `line_id` (Actuals or Samples)
- `prod_id` (Product ID)
- `cust_no` (Customer Number)
- `ctry_id` (Country ID)

The procedure only selects columns of MV_SALES.

- **MV_SALES**
  - ↳ **T_CUSTOMER**
  - ↳ **T_CUSTOMER_GROUP**
  - ↳ **T_CUST_DISTRIB**
  - ↳ **T_ORDER_HEAD**
  - ↳ **T_TRANSMISSION**
  - ↳ **T_ORDER_POSITION**
  - ↳ **T_PRODUCT**
  - ↳ **T_PRODUCT_GROUP**
  - ↳ **T_LINE**
  - ↳ **T_COUNTRY**
  - ↳ **T_CURRENCY**
  - ↳ **T_TARGET_PG**
  - ↳ **T_CODE**
  - ↳ **T_COGS_HISTORY**

5.3.5 Area Sales Statistics

5.3.5.1 Introduction



The Area Sales Statistics was designed to display all sales of the identified and asigned sales areas (sales representatives).

Definition of area sales:

- Products are grouped in so called Target Product Groups (TPG; e.g. Enbrel Team).
- Sales Reps are exclusively assigned to one of these Target Product Groups
- Customers are assigned to Sales Reps

Due to these assignments it is possible to create connections from products over sales reps to the customers. And this is what is displayed in this report.

As not every product is assigned to a TPG and not every customer is assigned to a sales rep, there are 'undefined' entries, which together with the TPG (assigned) sales add-up to 100% of net sales. Because of this, the Area Sales Statistics could be cross-checked to the other sales reports and will display the same values.

TOTAL Undefined sales are the sales for products which are not assigned to a TPG.
Undefined entries in a TPG are the sales of a product in the TPG of a customer who is not assigned to a sales rep.

With the "Group by" field it is possible to control the layout and grouping of the displayed data. You can choose between four alternatives. The order of the fields (TPG – Rep – Prod – Cust) shows their hierarchy. The following layouts and groupings can be chosen:

- TPGroup. & Rep. & Prod. & Cust.

- TPGroup. & Prod. & Rep. & Cust
- TPGroup. & Rep. & Cust & Prod.
- Prod. & Cust & Rep.

5.3.5.2 Structure

Report Layout:

Customer Name: The customer name string consists of the following connected database fields (from the T_CUSTOMER table).
CUST_CITY & ', ' & CUST_ZIP & ' ' & CUST_NAME & ' ' & CUST_DEPARTMENT & '-' & CUDI_CUSTOMER_NR
The & symbols the string connection operator.

Product: Name of the product.

Sales Rep: Name of the Sales Rep
(SARE_LAST_NAME & ' ' & SARE_FIRST_NAME from T_Sales_Rep)

Units / Value: Sum of Units and Values of the orders of the customers of the sales reps on the product in their TPG, according to the **percentage** of the assignment of the sales rep to their customers!

FG: FG stands for Free Goods and shows number of units and value of Free Goods attached to the order.

5.3.5.3 Construction

The stored procedure **`PKG_REPORTS.GetSalesAreaStatTAPG`** is started with the following parameters:

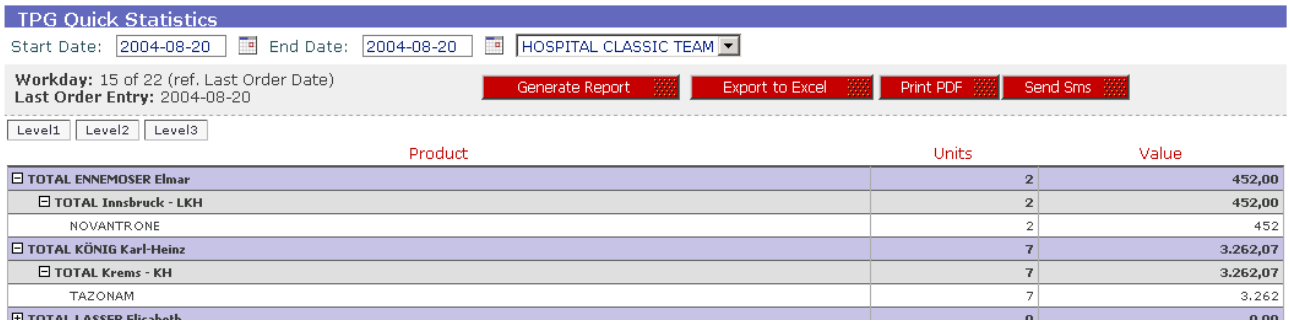
- `StartDate`
- `EndDate`
- `line_id` (Actuals, Samples, …)

The procedure selects columns from:

- **MV_SALES**
  - ↳ **T_CUSTOMER**
  - ↳ **T_CUSTOMER_GROUP**
  - ↳ **T_CUST_DISTRIB**
  - ↳ **T_ORDER_HEAD**
  - ↳ **T_TRANSMISSION**
  - ↳ **T_ORDER_POSITION**
  - ↳ **T_PRODUCT**
  - ↳ **T_PRODUCT_GROUP**
  - ↳ **T_LINE**
  - ↳ **T_COUNTRY**
  - ↳ **T_CURRENCY**
  - ↳ **T_TARGET_PG**
  - ↳ **T_CODE**
  - ↳ **T_COGS_HISTORY**
- **T_PRODUCT**
- **T_TARGET_PG**
- **T_SALES_REP**
- **T_TARGET_PG_SR**
- **T_CUST_SALES_REP**

5.3.6 TPG Quick Statistics

5.3.6.1 Introduction



The TPG Quick Statistics Report displays the sales of the sales reps without undefined sales. There are no tools to change the hierarchy of the displayed data, it only shows the sales of the selected Target Product Group.

The special functionality in this report is the possibility to send the sales entries directly as SMS to the sales reps by pressing the 'Send SMS' button. Every sales rep will receive the actual sales data via SMS on the mobile phone. The mobile phone numbers are retrieved from the Intranet user profile.

5.3.6.2 Structure

Report Layout:

Product: Name of the product.

Units / Value: Sum of Units and Values from the orders by customers and the sales reps grouped by their TPG, following the **percentage** of the customers assigned to the reps.

FG: FG stands for Free Goods and shows number of units and value of Free Goods attached to the order.

5.3.6.3 Construction

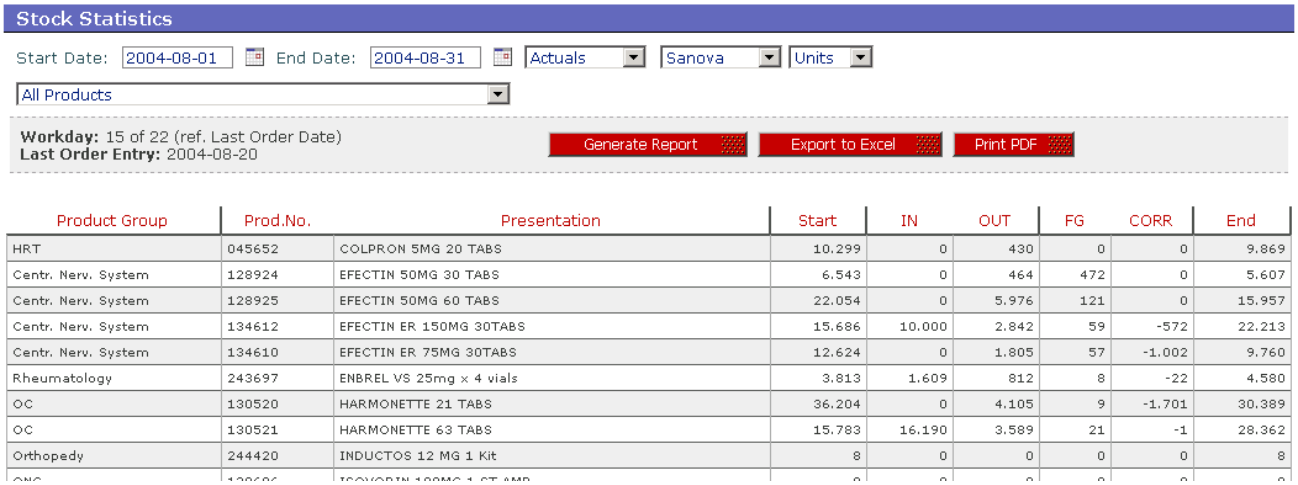
- MV_SALES
  - ↳ T_CUSTOMER
  - ↳ T_CUSTOMER_GROUP
  - ↳ T_CUST_DISTRIB
  - ↳ T_ORDER_HEAD
  - ↳ T_TRANSMISSION
  - ↳ T_ORDER_POSITION
  - ↳ T_PRODUCT
  - ↳ T_PRODUCT_GROUP
  - ↳ T_LINE
  - ↳ T_COUNTRY
  - ↳ T_CURRENCY
  - ↳ T_TARGET_PG
  - ↳ T_CODE
  - ↳ T_COGS_HISTORY
- T_PRODUCT
- T_TARGET_PG

- **T_SALES_REP**
- **T_TARGET_PG_SR**
- **T_CUST_SALES_REP**

5.4 Logistics Reports

5.4.1 Stock Statistics

5.4.1.1 Report



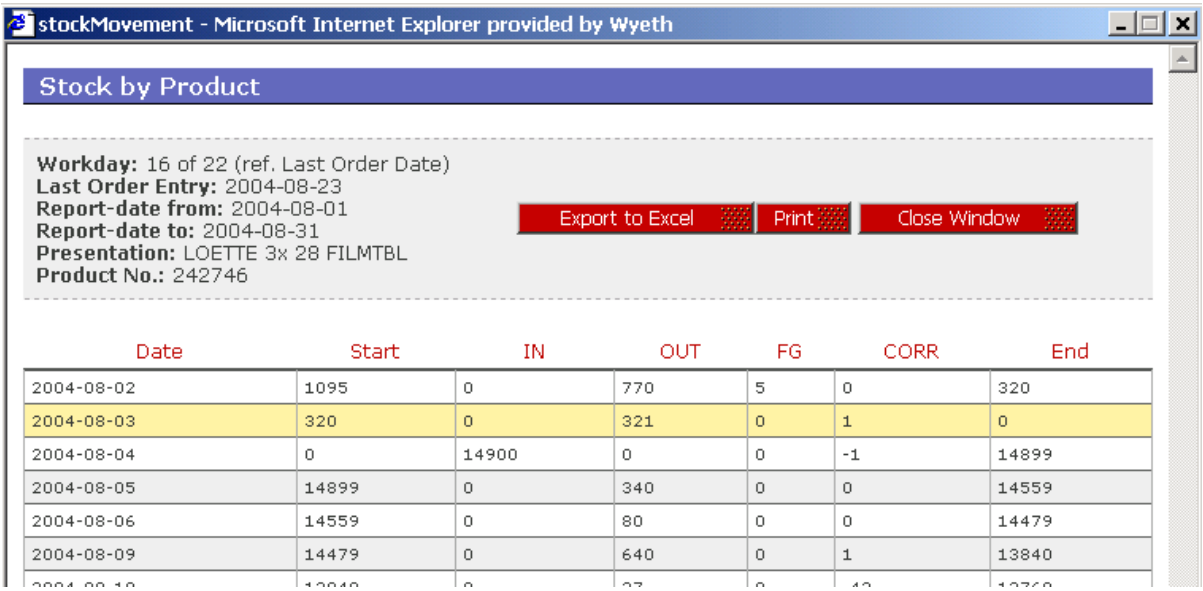
Product Group	Prod.No.	Presentation	Start	IN	OUT	FG	CORR	End
HRT	045652	COLPRON 5MG 20 TABS	10.299	0	430	0	0	9.869
Centr. Nerv. System	128924	EFECTIN 50MG 30 TABS	6.543	0	464	472	0	5.607
Centr. Nerv. System	128925	EFECTIN 50MG 60 TABS	22.054	0	5.976	121	0	15.957
Centr. Nerv. System	134612	EFECTIN ER 150MG 30TABS	15.686	10.000	2.842	59	-572	22.213
Centr. Nerv. System	134610	EFECTIN ER 75MG 30TABS	12.624	0	1.805	57	-1.002	9.760
Rheumatology	243697	ENBREL VS 25mg x 4 vials	3.813	1.609	812	8	-22	4.580
OC	130520	HARMONETTE 21 TABS	36.204	0	4.105	9	-1.701	30.389
OC	130521	HARMONETTE 63 TABS	15.783	16.190	3.589	21	-1	28.362
Orthopedy	244420	INDUCTOS 12 MG 1 Kit	8	0	0	0	0	8

The report shows rolled up stock information for the selected time period by SKU (stock keeping unit).
It displays inventory level, stock and sales data and checks stock movements.
This is done by analyzing he reported starting balance (inventory level), considering all incoming, outgoing sales and correction records, e.g. expired products taken out of the stock, for each product and calculates an end balance. This end balance is checked against the starting balance of the next day. Both values must be identical. If there are any differences, the data reported or booked by Sanova must be wrong.

After inserting a new transmission, the data of the transmission is checked for stock consistency. If there are any differences an error message will highlight the inconsistency on the start page of the ALF-System. By clicking on the error message users are directly transferred to the stock statistics report and the product concerned.

For better information delivery, the report offers more details, therefore it is possible to click every column to get detailed information of the underlying transactional data.

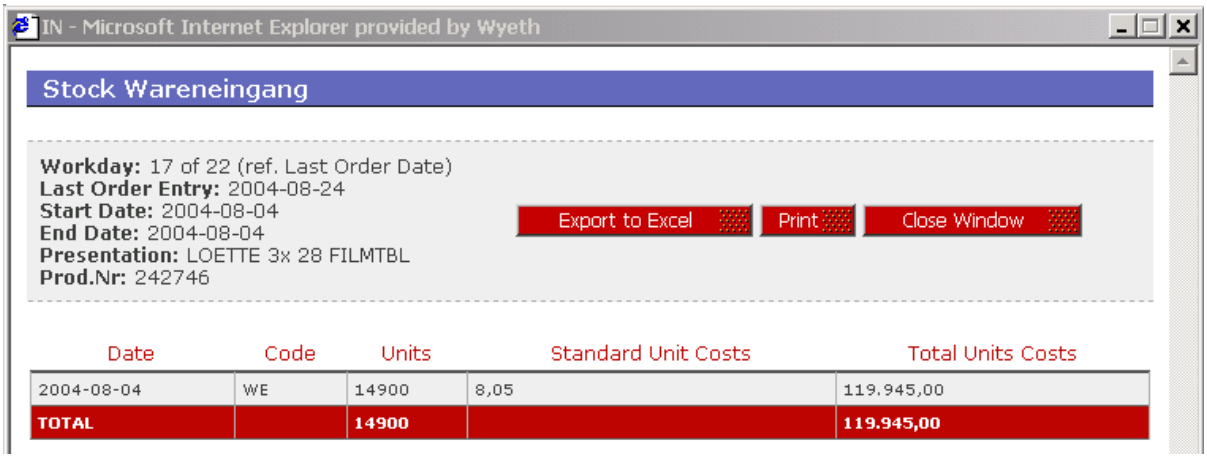
5.4.1.1.1 Product Stock Pop Up (Stock Movement)



Date	Start	IN	OUT	FG	CORR	End
2004-08-02	1095	0	770	5	0	320
2004-08-03	320	0	321	0	1	0
2004-08-04	0	14900	0	0	-1	14899
2004-08-05	14899	0	340	0	0	14559
2004-08-06	14559	0	80	0	0	14479
2004-08-09	14479	0	640	0	1	13840

By clicking on the column "Presentation" or the column "Start" of the report, this pop-up opens and shows the Start, In, Out, FG, Corr and End information for every single day in the selected time period of the main report.

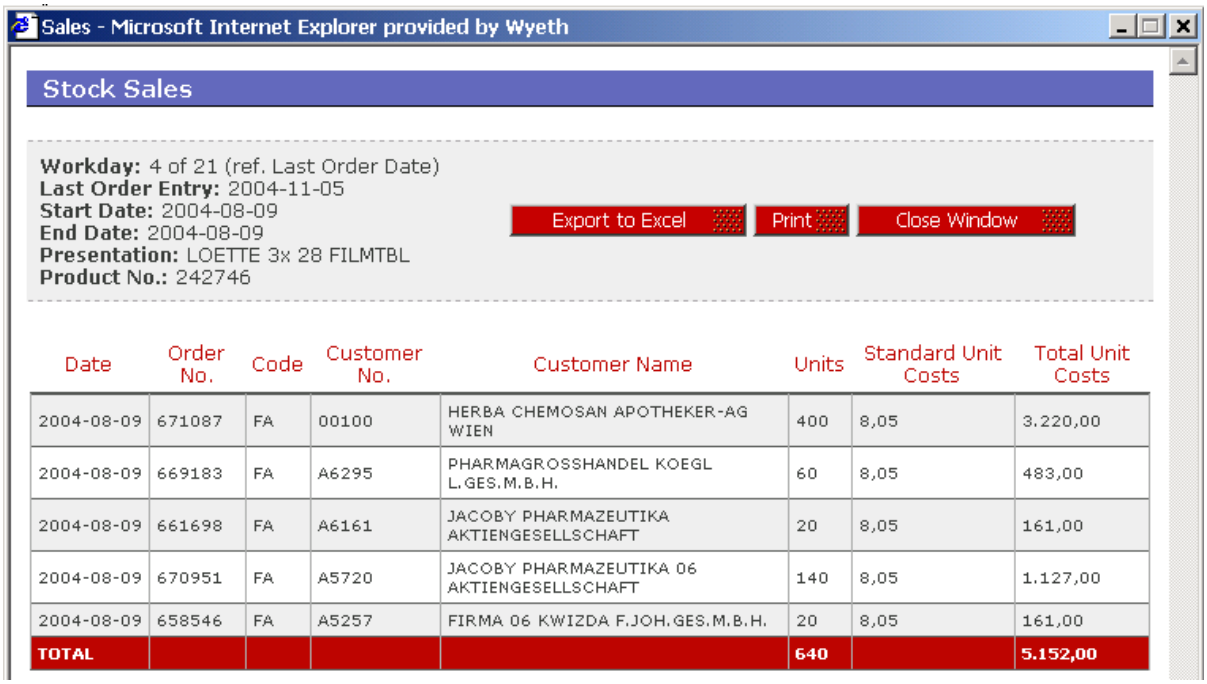
5.4.1.1.2 Incomings Stock Pop Up



Date	Code	Units	Standard Unit Costs	Total Units Costs
2004-08-04	WE	14900	8,05	119.945,00
TOTAL		**14900**		**119.945,00**

By clicking on the "IN" column in the Stock Statistics Report or the Product Pop-Up, this pop-up opens and shows detailed information on the stock incomings (BewegKZ = 'WE' in T_STOCK) of the selected day.

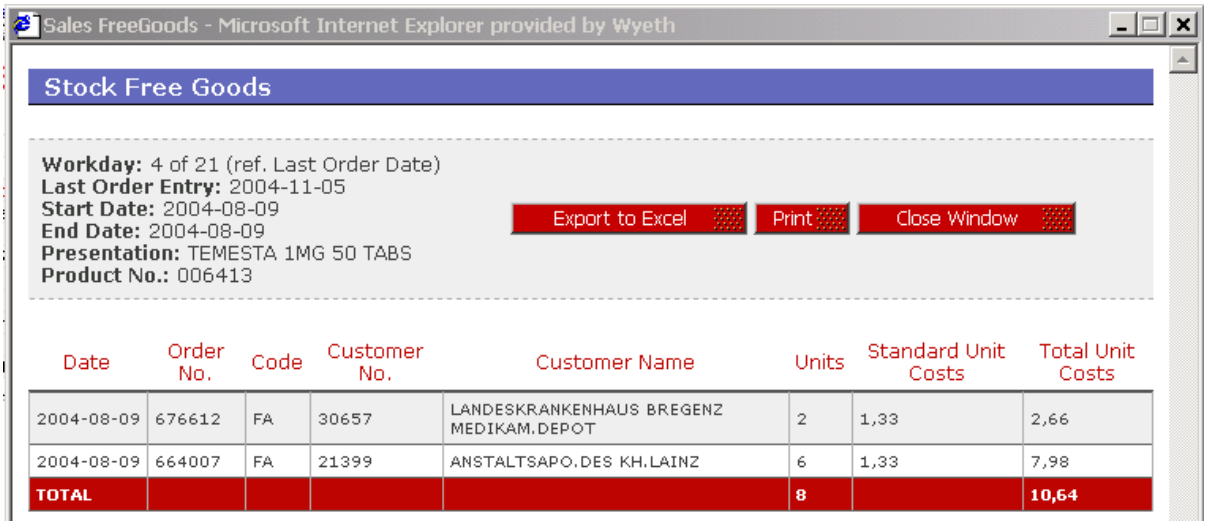
5.4.1.1.3 Outgoings Stock Pop Up



Date	Order No.	Code	Customer No.	Customer Name	Units	Standard Unit Costs	Total Unit Costs
2004-08-09	671087	FA	00100	HERBA CHEMOSAN APOTHEKER-AG WIEN	400	8,05	3.220,00
2004-08-09	669183	FA	A6295	PHARMAGROSSHANDEL KOEGL L.GES.M.B.H.	60	8,05	483,00
2004-08-09	661698	FA	A6161	JACOBY PHARMAZEUTIKA AKTIENGESELLSCHAFT	20	8,05	161,00
2004-08-09	670951	FA	A5720	JACOBY PHARMAZEUTIKA 06 AKTIENGESELLSCHAFT	140	8,05	1.127,00
2004-08-09	658546	FA	A5257	FIRMA 06 KWIZDA F.JOH.GES.M.B.H.	20	8,05	161,00
TOTAL					**640**		**5.152,00**

By clicking on the "OUT" column in the Stock Statistics Report or the Product Pop-Up, this pop-up opens and shows detailed information on the Sales of the selected date.

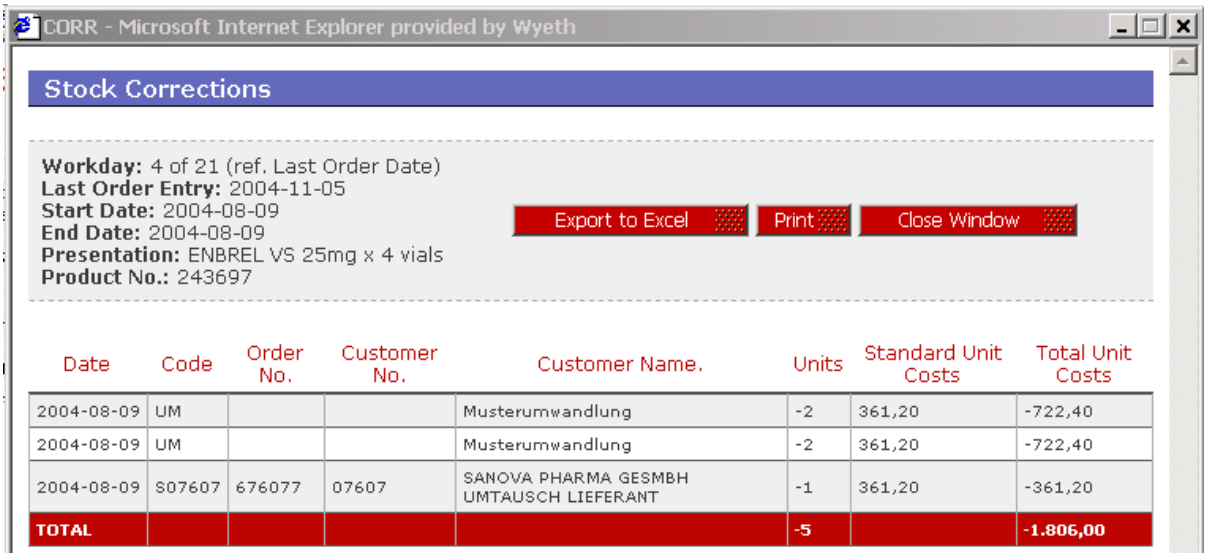
5.4.1.1.4 Stock Free Goods Pop Up



Date	Order No.	Code	Customer No.	Customer Name	Units	Standard Unit Costs	Total Unit Costs
2004-08-09	676612	FA	30657	LANDESKRANKENHAUS BREGENZ MEDIKAM.DEPOT	2	1,33	2,66
2004-08-09	664007	FA	21399	ANSTALTSAPO.DES KH.LAINZ	6	1,33	7,98
TOTAL					**8**		**10,64**

By clicking on the FG column in the Stock Statistics Report or in the Product Pop-Up, this pop-up opens and shows detailed information on the Free Goods granted on the selected date range.

5.4.1.1.5 Stock Corrections Pop Up



Date	Code	Order No.	Customer No.	Customer Name.	Units	Standard Unit Costs	Total Unit Costs
2004-08-09	UM			Musterumwandlung	-2	361,20	-722,40
2004-08-09	UM			Musterumwandlung	-2	361,20	-722,40
2004-08-09	S07607	676077	07607	SANOVA PHARMA GESMBH UMTAUSCH LIEFERANT	-1	361,20	-361,20
TOTAL					**-5**		**-1.806,00**

By clicking on the "CORR" column in the Stock Statistics Report or in the Product Pop-Up, this pop-up opens and shows detailed information on corrective stock movements during the selected date range.

5.4.1.2 Structure

Report Layout:

Product Group: Name of the product group.

Prod. No.: Product number (Sanova: Phznr = "Pharmazentralnummer").

Presentation: Product presentation.

Start: Inventory level value of the Start Date (in the Stock Movement Pop Up: Inventory level value of the date in the row).

IN: Rolled up value of the incoming stock of the selected time period or the given date in the row.

OUT: Rolled up value of the sales in the selected time period or the given date in the row.

FG: Rolled up value of the free goods granted in the selected time period or the given date in the row.

CORR: Rolled up value of the corrective stock movements in the selected time period or the given date in the row.

End: This is the calculated ending balance of the day as long as it is equal to the starting balance of the next day. If this is not the case, the line is framed red and shows two entries: a Sanova end value (the starting value of the next day) and a Wyeth value, which is calculated using the starting value of the current day plus all incomings, sales, free goods and plus/minus corrective stock movements.

Code: Code of the stock movement, denoting the type of special movement.

Standard Unit Costs: COGS (Cost Of Goods Sold) of the reported product.

Total Unit Costs: COGS * Units

Units: Units of the displayed stock movement.

5.4.1.2.1 Stock Check Failure

WARNING! Vaccines	134713	PREVENAR 0,5ml, 1 Vial AT	738	0	23	0	-6	Sanova:714 Wyeth:709

If a Stock Check fails for a specific product the row is highlighted in red (like the example above) to indicate the error. By clicking on the row entry, users will get more information in the pop-up which then opens.

The Sanova value is the reported inventory level of the next day.
The Wyeth value is calculated (Start + IN – OUT – FG + CORR = End).

5.4.1.3 Construction

5.4.1.3.1 Report

The stored procedure **`PKG_STOCK.GetStockStat`** is started with the following parameters:

- `StartDate`
- `EndDate`
- `line_id` (Actuals, Samples, …)
- `dist_id` (Distributor ID (Sanova))
- `prod_id` (product ID (if the ID is 0 the statement selects all products))

The procedure selects columns from:

- **V_STOCK_STAT**
  - **MV_STOCK**
    - **V_INVENTORY_LEVEL_PRODID**
      - **T_INVENTORY_LEVEL**
      - **V_TRANSMISSION_PRODUCT**
    - **V_MV_STOCK**
      - **T_CODE**
      - **V_STOCK**
        - **T_STOCK**
        - **T_PRODUCT**
        - **T_TRANSMISSION**
        - **T_PRODUCT_GROUP**
        - **T_CODE**
        - **T_LINE**
        - **T_CUST_DISTRIB**
        - **T_CUSTOMER**
    - **MV_SALES_PER_DAY2**
      - **V_SALES_PER_DAY2**
        - T_LINE
        - T_PRODUCT_GROUP
        - T_ORDER_POSITION
        - T_ORDER_HEAD
        - **V_TRANSMISSION_PRODUCT**
          - T_TRANSMISSION
          - **MV_ACTIVE_PRODUCT**
            - T_PRODUCT
            - T_ORDER_POSITION
            - T_TRANSMISSION
            - T_ORDER_HEAD
            - T_INVENTORY_LEVEL
            - T_STOCK
            - T_BUDGET
        - T_CUSTOMER
        - **V_FXRATE_EUR_USD_ACT**
          - T_FXRATE
          - T_CODE
          - T_CURRENCY
        - T_COGS_HISTORY

5.4.1.3.2 Incoming Stock Pop-Up

The stored procedure **`PKG_STOCK.GetStockWE`** is started with the following parameters:

- `StartDate`
- `EndDate`
- `line_id` (Actuals, Samples, …)
- `prod_id` (product ID (if the ID is 0 the statement selects all products))

The procedure selects columns from:

- **`V_STOCK`**
  - ↳ **`T_STOCK`**
  - ↳ **`T_PRODUCT`**
  - ↳ **`T_TRANSMISSION`**
  - ↳ **`T_PRODUCT_GROUP`**
  - ↳ **`T_CODE`**
  - ↳ **`T_LINE`**
  - ↳ **`T_CUST_DISTRIB`**
  - ↳ **`T_CUSTOMER`**

5.4.1.3.3 Outgoing Stock Pop-Up

The stored procedure **`PKG_STOCK.GetStockOUT`** is started with the following parameters:

- `StartDate`
- `EndDate`
- `line_id` (Actuals, Samples, …)
- `prod_id` (product ID (if the ID is 0 the statement selects all products))

The procedure selects columns from:

- **`MV_SALES`**
  - ↳ **`T_CUSTOMER`**
  - ↳ **`T_CUSTOMER_GROUP`**
  - ↳ **`T_CUST_DISTRIB`**
  - ↳ **`T_ORDER_HEAD`**
  - ↳ **`T_TRANSMISSION`**
  - ↳ **`T_ORDER_POSITION`**
  - ↳ **`T_PRODUCT`**
  - ↳ **`T_PRODUCT_GROUP`**
  - ↳ **`T_LINE`**
  - ↳ **`T_COUNTRY`**
  - ↳ **`T_CURRENCY`**
  - ↳ **`T_TARGET_PG`**
  - ↳ **`T_CODE`**
  - ↳ **`T_COGS_HISTORY`**

5.4.1.3.4 Free Goods Pop-Up

The stored procedure **`PKG_STOCK.GetStockOUT_FG`** is started with the following parameters:

- `StartDate`
- `EndDate`
- `line_id` (Actuals, Samples, …)
- `prod_id` (product ID (if the ID is 0 the statement selects all products))

The procedure selects columns from:

- **`MV_SALES`**
  - ↳ **`T_CUSTOMER`**
  - ↳ **`T_CUSTOMER_GROUP`**
  - ↳ **`T_CUST_DISTRIB`**
  - ↳ **`T_ORDER_HEAD`**
  - ↳ **`T_TRANSMISSION`**
  - ↳ **`T_ORDER_POSITION`**
  - ↳ **`T_PRODUCT`**
  - ↳ **`T_PRODUCT_GROUP`**
  - ↳ **`T_LINE`**
  - ↳ **`T_COUNTRY`**
  - ↳ **`T_CURRENCY`**
  - ↳ **`T_TARGET_PG`**
  - ↳ **`T_CODE`**
  - ↳ **`T_COGS_HISTORY`**

5.4.1.3.5 Corrections Pop-Up

The stored procedure **`PKG_STOCK.GetStockKORR`** is started with the following parameters:

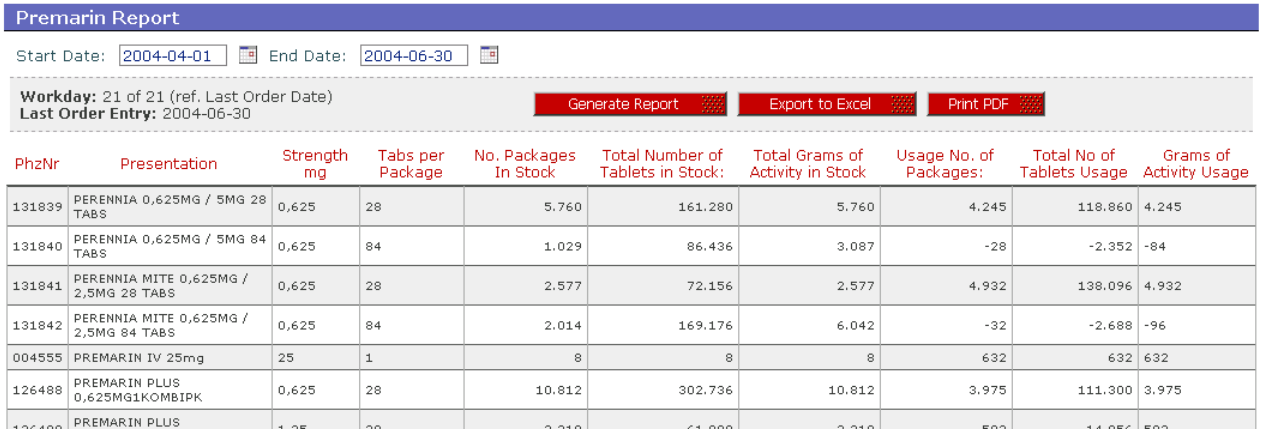
- `StartDate`
- `EndDate`
- `line_id` (Actuals, Samples, …)
- `prod_id` (product ID (if the ID is 0 the statement selects all products))

The procedure selects columns from:

- **`V_STOCK`**
  - ↳ **`T_STOCK`**
  - ↳ **`T_PRODUCT`**
  - ↳ **`T_TRANSMISSION`**
  - ↳ **`T_PRODUCT_GROUP`**
  - ↳ **`T_CODE`**
  - ↳ **`T_LINE`**
  - ↳ **`T_CUST_DISTRIB`**
  - ↳ **`T_CUSTOMER`**

5.4.2 Premarin Report

5.4.2.1 Introduction


Premarin Report

Start Date: 2004-04-01 End Date: 2004-06-30

Workday: 21 of 21 (ref. Last Order Date)
Last Order Entry: 2004-06-30

Generate Report | Export to Excel | Print PDF

PhzNr	Presentation	Strength mg	Tabs per Package	No. Packages In Stock	Total Number of Tablets in Stock:	Total Grams of Activity in Stock	Usage No. of Packages:	Total No of Tablets Usage	Grams of Activity Usage
131839	PERENNIA 0,625MG / 5MG 28 TABS	0,625	28	5.760	161.280	5.760	4.245	118.860	4.245
131840	PERENNIA 0,625MG / 5MG 84 TABS	0,625	84	1.029	86.436	3.087	-28	-2.352	-84
131841	PERENNIA MITE 0,625MG / 2,5MG 28 TABS	0,625	28	2.577	72.156	2.577	4.932	138.096	4.932
131842	PERENNIA MITE 0,625MG / 2,5MG 84 TABS	0,625	84	2.014	169.176	6.042	-32	-2.688	-96
004555	PREMARIN IV 25mg	25	1	8	8	8	632	632	632
126488	PREMARIN PLUS 0,625MG1KOMBIPK	0,625	28	10.812	302.736	10.812	3.975	111.300	3.975
	PREMARIN PLUS								

The Premarin Report displays inventory level (of the last date of the selected time period) and sales information of the products Perennia, Premarin and Sequennia.

5.4.2.2 Structure

Report Layout:

Prod. No.: Product number (Phznr).

Presentation: Product presentation.

Strength mg: Strength of the product in mg (Prod_Strength of T_PRODUCT).

Tabs p. Package: Number of tablets in a product package (Prod_Packsize of T_PRODUCT)

Total Number of Tablets in Stock: The number is the inventory level (number of packages in stock) multiplied with the number of tablets per package. For the inventory level the last inventory level of the product in the selected time period is taken!

Total Grams of Activity in Stock: The number is the inventory level (number of packages in stock) multiplied with the number of tablets per package and the strength per tablet (The database field Prod_Units_Measure holds the multiplied value of packsize and strength).

Usage No. of Packages: Number of packages sold in the selected time period.

Total No of Tablets Usage: Usage No. of Packages multiplied with Tabs per Package

Grams of Activity Usage: Usage No. of Packages multiplied with Tabs per Package multiplied with Strength in mg

5.4.2.3 Construction

The stored procedure **`PKG_STOCK.GetStockWE`** is started with the following parameters:

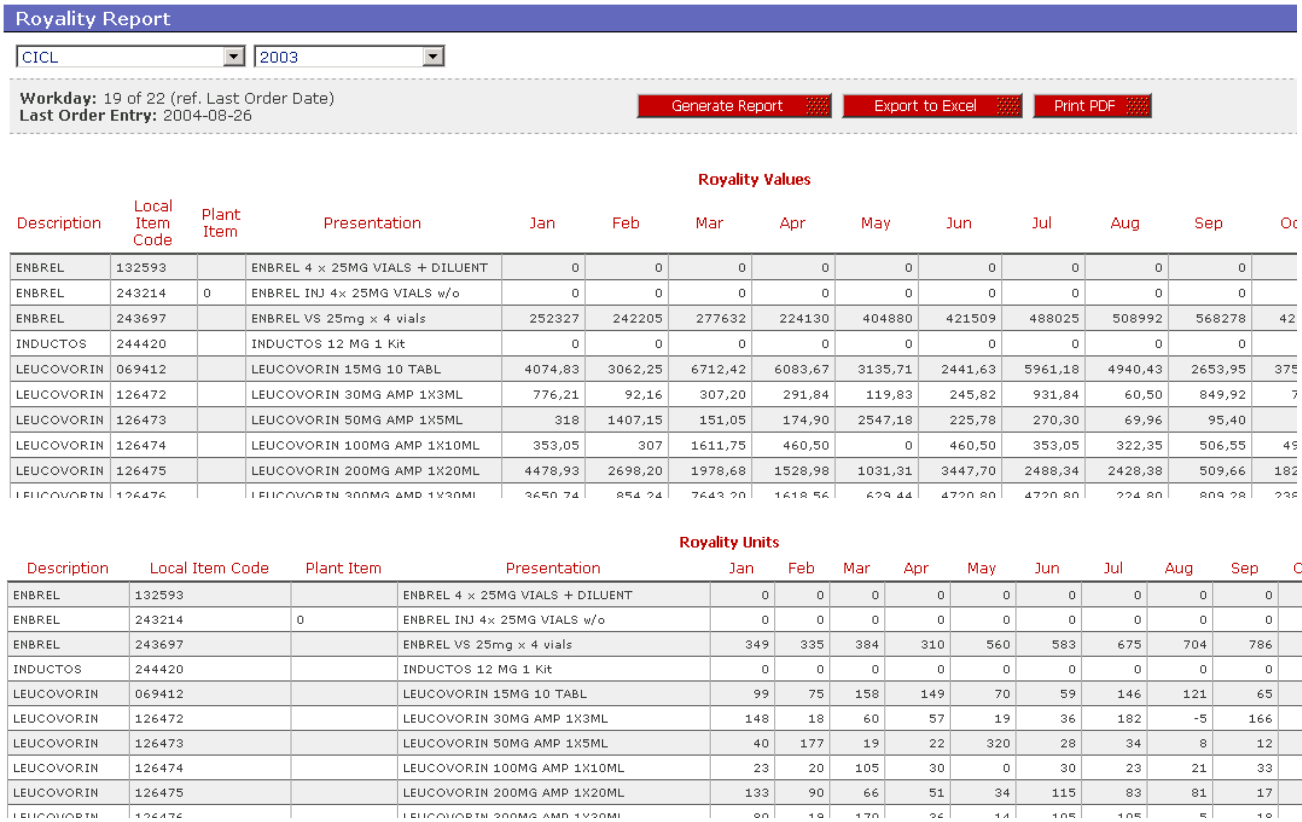
- `StartDate`
- `EndDate`
- `ctry_id` (country ID)

The procedure selects columns from:

- **T_INVENTORY_LEVEL**
- **MV_SALES_PER_DAY2**
  - ↳ **V_SALES_PER_DAY2**
    - ↳ T_LINE
    - ↳ T_PRODUCT_GROUP
    - ↳ T_ORDER_POSITION
    - ↳ T_ORDER_HEAD
    - ↳ **V_TRANSMISSION_PRODUCT**
      - ↳ T_TRANSMISSION
      - ↳ **MV_ACTIVE_PRODUCT**
        - ↳ T_PRODUCT
        - ↳ T_ORDER_POSITION
        - ↳ T_TRANSMISSION
        - ↳ T_ORDER_HEAD
        - ↳ T_INVENTORY_LEVEL
        - ↳ T_STOCK
        - ↳ T_BUDGET
    - ↳ T_CUSTOMER
    - ↳ **V_FXRATE_EUR_USD_ACT**
      - ↳ T_FXRATE
      - ↳ T_CODE
      - ↳ T_CURRENCY
    - ↳ T_COGS_HISTORY

5.4.3 Royalty Report

5.4.3.1 Introduction


Royality Report

CICL 2003

Workday: 19 of 22 (ref. Last Order Date)
Last Order Entry: 2004-08-26

Generate Report Export to Excel Print PDF

Royality Values

Description	Local Item Code	Plant Item	Presentation	Jan	Feb	Mar	Apr	May	Jun	Jul	Aug	Sep	Oc
ENBREL	132593		ENBREL 4 x 25MG VIALS + DILUENT	0	0	0	0	0	0	0	0	0	
ENBREL	243214	0	ENBREL INJ 4x 25MG VIALS w/o	0	0	0	0	0	0	0	0	0	
ENBREL	243697		ENBREL VS 25mg x 4 vials	252327	242205	277632	224130	404880	421509	488025	508992	568278	42
INDUCTOS	244420		INDUCTOS 12 MG 1 Kit	0	0	0	0	0	0	0	0	0	
LEUCOVORIN	069412		LEUCOVORIN 15MG 10 TABL	4074,83	3062,25	6712,42	6083,67	3135,71	2441,63	5961,18	4940,43	2653,95	375
LEUCOVORIN	126472		LEUCOVORIN 30MG AMP 1X3ML	776,21	92,16	307,20	291,84	119,83	245,82	931,84	60,50	849,92	7
LEUCOVORIN	126473		LEUCOVORIN 50MG AMP 1X5ML	318	1407,15	151,05	174,90	2547,18	225,78	270,30	69,96	95,40	
LEUCOVORIN	126474		LEUCOVORIN 100MG AMP 1X10ML	353,05	307	1611,75	460,50	0	460,50	353,05	322,35	506,55	49
LEUCOVORIN	126475		LEUCOVORIN 200MG AMP 1X20ML	4478,93	2698,20	1978,68	1528,98	1031,31	3447,70	2488,34	2428,38	509,66	182
LEUCOVORIN	126476		LEUCOVORIN 300MG AMP 1X30ML	3650,74	854,24	7643,20	1618,56	629,44	4720,80	4720,80	224,80	809,28	235

Royality Units

Description	Local Item Code	Plant Item	Presentation	Jan	Feb	Mar	Apr	May	Jun	Jul	Aug	Sep	Oct
ENBREL	132593		ENBREL 4 x 25MG VIALS + DILUENT	0	0	0	0	0	0	0	0	0	
ENBREL	243214	0	ENBREL INJ 4x 25MG VIALS w/o	0	0	0	0	0	0	0	0	0	
ENBREL	243697		ENBREL VS 25mg x 4 vials	349	335	384	310	560	583	675	704	786	58
INDUCTOS	244420		INDUCTOS 12 MG 1 Kit	0	0	0	0	0	0	0	0	0	
LEUCOVORIN	069412		LEUCOVORIN 15MG 10 TABL	99	75	158	149	70	59	146	121	65	9
LEUCOVORIN	126472		LEUCOVORIN 30MG AMP 1X3ML	148	18	60	57	19	36	182	-5	166	1
LEUCOVORIN	126473		LEUCOVORIN 50MG AMP 1X5ML	40	177	19	22	320	28	34	8	12	
LEUCOVORIN	126474		LEUCOVORIN 100MG AMP 1X10ML	23	20	105	30	0	30	23	21	33	3
LEUCOVORIN	126475		LEUCOVORIN 200MG AMP 1X20ML	133	90	66	51	34	115	83	81	17	6
LEUCOVORIN	126476		LEUCOVORIN 300MG AMP 1X30ML	80	18	170	36	14	105	105	5	18	5

The dropdown box holds all Invoicer Codes of the products, and by selecting a code only the products from the specified invoicer are displayed.

The report shows the sales per packer / invoicer (in the upper sheet the values and in the lower sheet the units) on a yearly basis (with columns for the months).
WHAT IS THE PURPOSE OF THIS REPORT ??? – REMARK BY C. STEHLE

5.4.3.2 Structure

Report Layout:

Product: Product name.

Local Item Code: Product number (Phznr).

Plant Item: Product number of the manufacturer

Presentation: Product presentation.

Months: Values / Units of the sales in the according month.

5.4.3.3 Construction

There are two stored procedures: one delivering the values and one delivering the units.

The stored procedure **`PKG_REPORT.GetRoyaltyReportValue`** is started with the following parameters:

- `StartDate`
- `EndDate`
- `ctry_id` (country ID)
- `invoicer_code`

The stored procedure **`PKG_REPORT.GetRoyaltyUnit`** is started with the following parameters:

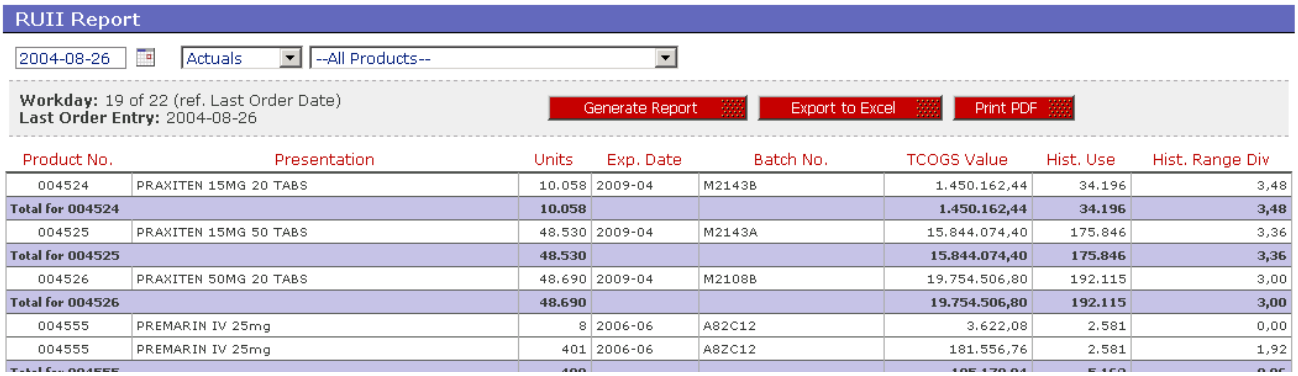
- `StartDate`
- `EndDate`
- `ctry_id` (country ID)
- `invoicer_code`

Both procedures select columns from the MV_ROYALTY:

- **MV_ROYALITY**
  - ↳ **MV_SALES_PER_DAY2**
    - ↳ **V_SALES_PER_DAY2**
      - ↳ **T_LINE**
      - ↳ **T_PRODUCT_GROUP**
      - ↳ **T_ORDER_POSITION**
      - ↳ **T_ORDER_HEAD**
      - ↳ **V_TRANSMISSION_PRODUCT**
      - ↳ **T_CUSTOMER**
      - ↳ **V_FXRATE_EUR_USD_ACT**
        - ↳ **T_FXRATE**
        - ↳ **T_CODE**
      - ↳ **T_COGS_HISTORY**

5.4.4 RUII Report

5.4.4.1 Introduction


RUII Report

2004-08-26 Actuals --All Products--

Workday: 19 of 22 (ref. Last Order Date)
Last Order Entry: 2004-08-26

Generate Report Export to Excel Print PDF

Product No.	Presentation	Units	Exp. Date	Batch No.	TCOGS Value	Hist. Use	Hist. Range Div
004524	PRAXITEN 15MG 20 TABS	10.058	2009-04	M2143B	1.450.162,44	34.196	3,48
Total for 004524		**10.058**			**1.450.162,44**	**34.196**	**3,48**
004525	PRAXITEN 15MG 50 TABS	48.530	2009-04	M2143A	15.844.074,40	175.846	3,36
Total for 004525		**48.530**			**15.844.074,40**	**175.846**	**3,36**
004526	PRAXITEN 50MG 20 TABS	48.690	2009-04	M2108B	19.754.506,80	192.115	3,00
Total for 004526		**48.690**			**19.754.506,80**	**192.115**	**3,00**
004555	PREMARIN IV 25mg	8	2006-06	A82C12	3.622,08	2.581	0,00
004555	PREMARIN IV 25mg	401	2006-06	A8ZC12	181.556,76	2.581	1,92

This report shows inventory levels and the sales of the last year and calculates a Historic Range Dividend. Example: 2003-08-26 to 2004-08-26

It is possible to select a single product.

5.4.4.2 Structure

Report Layout:

Product No.: Product number (Phznr).

Presentation: Product presentation.

Units: Inventory Level

Exp. Date: Expiring date of the batch on stock.

Batch No: Batch number of the batch on stock.

TCOGS Value: Cost of Goods of the products on stock.

Hist. Use: Units of the sales in the last year.

Hist Range Div.: Inventory Units / Sales Units multiplied with 12

5.4.4.3 Construction

The stored procedure **`PKG_REPORT.GetRUII`** is started with the following parameters:

- `StartDate`
- `line_id` (Actuals, Samples, …)
- `prod_id` (product ID (if the ID is 0 the statement selects all products))

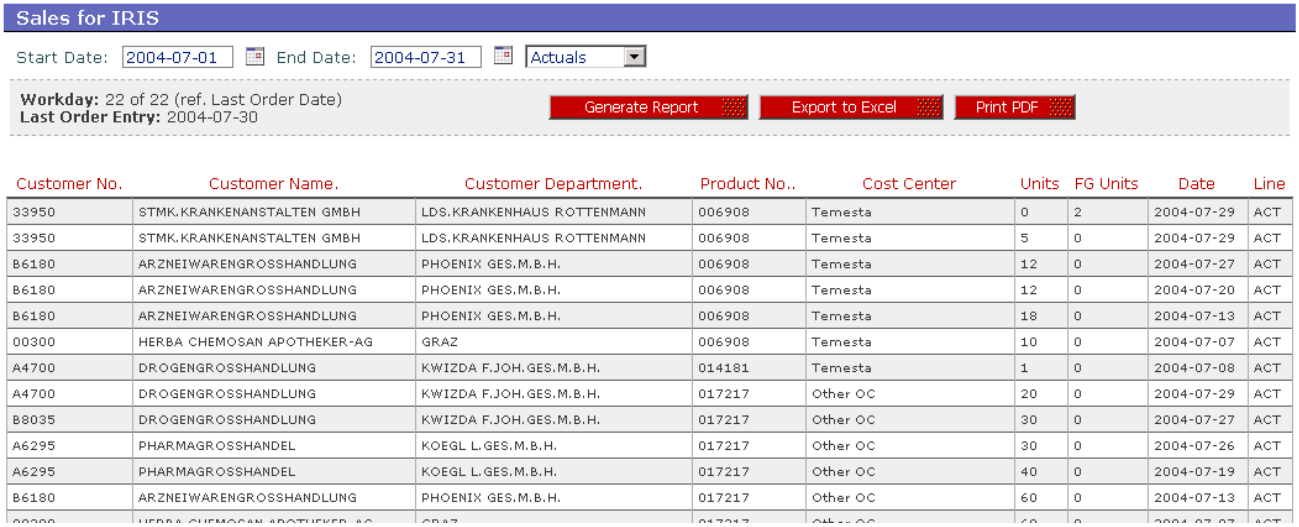
The procedure selects columns from:

- **MV_SALES_PER_DAY2**
  - ↳ **V_SALES_PER_DAY2**
    - ↳ **T_LINE**
    - ↳ **T_PRODUCT_GROUP**
    - ↳ **T_ORDER_POSITION**

- T_ORDER_HEAD
- V_TRANSMISSION_PRODUCT
- T_CUSTOMER
- V_FXRATE_EUR_USD_ACT
  - T_FXRATE
  - T_CODE
- T_COGS_HISTORY

5.4.5 Sales for IRIS

5.4.5.1 Introduction



Customer No.	Customer Name.	Customer Department.	Product No..	Cost Center	Units	FG Units	Date	Line
33950	STMK.KRANKENANSTALTEN GMBH	LDS.KRANKENHAUS ROTTENMANN	006908	Temesta	0	2	2004-07-29	ACT
33950	STMK.KRANKENANSTALTEN GMBH	LDS.KRANKENHAUS ROTTENMANN	006908	Temesta	5	0	2004-07-29	ACT
B6180	ARZNEIWARENGROSSHANDLUNG	PHOENIX GES.M.B.H.	006908	Temesta	12	0	2004-07-27	ACT
B6180	ARZNEIWARENGROSSHANDLUNG	PHOENIX GES.M.B.H.	006908	Temesta	12	0	2004-07-20	ACT
B6180	ARZNEIWARENGROSSHANDLUNG	PHOENIX GES.M.B.H.	006908	Temesta	18	0	2004-07-13	ACT
00300	HERBA CHEMOSAN APOTHEKER-AG	GRAZ	006908	Temesta	10	0	2004-07-07	ACT
A4700	DROGENGROSSHANDLUNG	KWIZDA F.JOH.GES.M.B.H.	014181	Temesta	1	0	2004-07-08	ACT
A4700	DROGENGROSSHANDLUNG	KWIZDA F.JOH.GES.M.B.H.	017217	Other OC	20	0	2004-07-29	ACT
B8035	DROGENGROSSHANDLUNG	KWIZDA F.JOH.GES.M.B.H.	017217	Other OC	30	0	2004-07-27	ACT
A6295	PHARMAGROSSHANDEL	KOEGL L.GES.M.B.H.	017217	Other OC	30	0	2004-07-26	ACT
A6295	PHARMAGROSSHANDEL	KOEGL L.GES.M.B.H.	017217	Other OC	40	0	2004-07-19	ACT
B6180	ARZNEIWARENGROSSHANDLUNG	PHOENIX GES.M.B.H.	017217	Other OC	60	0	2004-07-13	ACT

All sales need to be imported into the IRIS system and the report provides the required format.

5.4.5.2 Structure

Report Layout:

Customer No.: Customer number.

Customer Name: Customer name.

Customer Dept.: Customer department (or name 2).

Product No.: Product number (Phznr).

Cost Center: Cost Center of the product.

Units: Units sold.

Units: Units as free goods granted

Date: Date of the sale.

Line: Acutals / Samples / Promotionals / …

5.4.5.3 Construction

The stored procedure **`PKG_REPORT.GetSalesForeCastAccuracy`** is started with the following parameters:

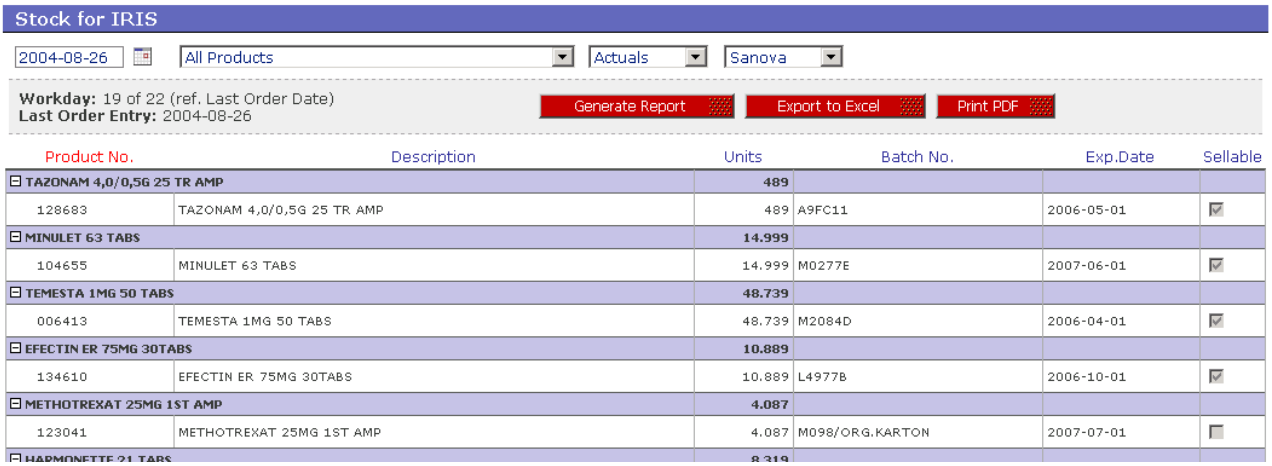
- `StartDate`
- `EndDate`
- `line_id` (Actuals, Samples, …)
- `ctry_id` (Country ID)

The procedure selects columns from:

- **MV_SALES**
  - ↳ **T_CUSTOMER**
  - ↳ **T_CUSTOMER_GROUP**
  - ↳ **T_CUST_DISTRIB**
  - ↳ **T_ORDER_HEAD**
  - ↳ **T_TRANSMISSION**
  - ↳ **T_ORDER_POSITION**
  - ↳ **T_PRODUCT**
  - ↳ **T_PRODUCT_GROUP**
  - ↳ **T_LINE**
  - ↳ **T_COUNTRY**
  - ↳ **T_CURRENCY**
  - ↳ **T_TARGET_PG**
  - ↳ **T_CODE**
  - ↳ **T_COGS_HISTORY**

5.4.6 Stock for IRIS

5.4.6.1 Introduction



Product No.	Description	Units	Batch No.	Exp.Date	Sellable
⊟ TAZONAM 4,0/0,5G 25 TR AMP		489			
128683	TAZONAM 4,0/0,5G 25 TR AMP	489	A9FC11	2006-05-01	☑
⊟ MINULET 63 TABS		14.999			
104655	MINULET 63 TABS	14.999	M0277E	2007-06-01	☑
⊟ TEMESTA 1MG 50 TABS		48.739			
006413	TEMESTA 1MG 50 TABS	48.739	M2084D	2006-04-01	☑
⊟ EFECTIN ER 75MG 30TABS		10.889			
134610	EFECTIN ER 75MG 30TABS	10.889	L4977B	2006-10-01	☑
⊟ METHOTREXAT 25MG 1ST AMP		4.087			
123041	METHOTREXAT 25MG 1ST AMP	4.087	M098/ORG.KARTON	2007-07-01	☐
⊟ HARMONETTE 21 TABS		8.319			

This is the second report for the IRIS system with additional information for product managers (batch number and expiring date).

5.4.6.2 Structure

Report Layout:

Product No.:	Product number (Phznr).
Presentation:	Product presentation.
Units:	Units in the stock.
Batch No:	Batch number of the batch in the stock.
Exp. Date:	Expiring date of the batch in the stock.
Sellable:	Interpretation of the Quarantine Code stock. ('S' and 'R' are not saleable, all others are saleable!)

5.4.6.3 Construction

The stored procedure **`PKG_REPORT.GetStockForIris`** is started with the following parameters:

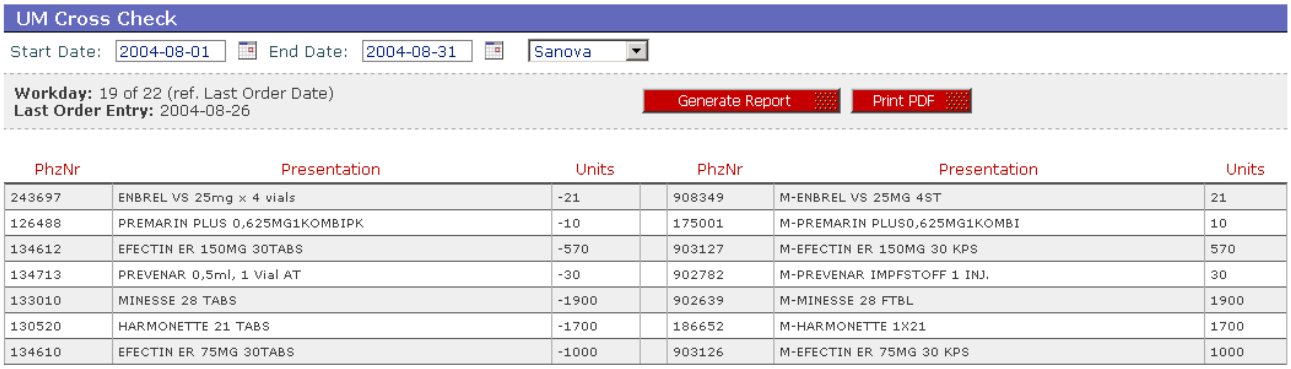
- `StartDate`
- `dist_id` (Distributor ID (Sanova))
- `line_id` (Actuals, Samples, …)
- `prod_id` (product ID (if the ID is 0 the statement selects all products))

The procedure selects columns from:

- **`T_INVENTORY_LEVEL`**
- **`MV_ACTIVE_PRODUCT`**
  - ↪ **`T_PRODUCT`**
  - ↪ **`T_ORDER_POSITION`**
  - ↪ **`T_TRANSMISSION`**
  - ↪ **`T_ORDER_HEAD`**
  - ↪ **`T_INVENTORY_LEVEL`**
  - ↪ **`T_STOCK`**
  - ↪ **`T_BUDGET`**
- **`T_CODE`**

5.4.7 UM Cross Check

5.4.7.1 Introduction



PhzNr	Presentation	Units		PhzNr	Presentation	Units
243697	ENBREL VS 25mg x 4 vials	-21		908349	M-ENBREL VS 25MG 4ST	21
126488	PREMARIN PLUS 0,625MG1KOMBIPK	-10		175001	M-PREMARIN PLUS0,625MG1KOMBI	10
134612	EFECTIN ER 150MG 30TABS	-570		903127	M-EFECTIN ER 150MG 30 KPS	570
134713	PREVENAR 0,5ml, 1 Vial AT	-30		902782	M-PREVENAR IMPFSTOFF 1 INJ.	30
133010	MINESSE 28 TABS	-1900		902639	M-MINESSE 28 FTBL	1900
130520	HARMONETTE 21 TABS	-1700		186652	M-HARMONETTE 1X21	1700
134610	EFECTIN ER 75MG 30TABS	-1000		903126	M-EFECTIN ER 75MG 30 KPS	1000

In the UM Cross Check Report, the transition of sales to samples (transaction code = UM = 'Musterumwandlungen') are reported.
The tracked units moved from the actuals stock into the samples stock are reported.
This transaction is the source of most errors reported by Sanova, resulting from incorrect execution of booking procedures at Sanova.

If the compared units (actual vs. sample) are not the same a warning (see example) is shown.

5.4.7.2 Structure

Report Layout:

Product No.: Product number (Phznr).

Presentation: Product presentation.

Units: Units taken from the actuals stock / Units moved into the samples stock.

5.4.7.3 Construction

The stored procedure **`PKG_STOCK.GetStockCrossCheckUM`** is started with the following parameters:

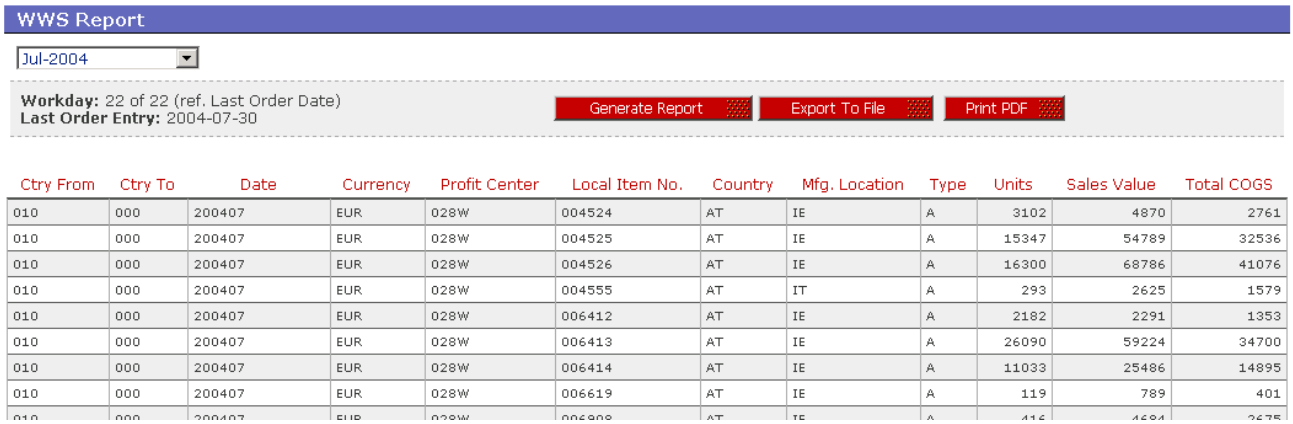
- `StartDate`
- `EndDate`
- `dist_id` (Distributor ID (Sanova))

The procedure selects columns from:

- **V_STOCK_SAM**
  - ↳ **V_STOCK**
    - ↳ **T_STOCK**
    - ↳ **T_PRODUCT**
    - ↳ **T_TRANSMISSION**
    - ↳ **T_PRODUCT_GROUP**
    - ↳ **T_CODE**
    - ↳ **T_LINE**
    - ↳ **T_CUST_DISTRIB**
    - ↳ **T_CUSTOMER**
- **V_STOCK_ACT**
  - ↳ **V_STOCK**
    - ↳ **T_STOCK**
    - ↳ **T_PRODUCT**
    - ↳ **T_TRANSMISSION**
    - ↳ **T_PRODUCT_GROUP**
    - ↳ **T_CODE**
    - ↳ **T_LINE**
    - ↳ **T_CUST_DISTRIB**
    - ↳ **T_CUSTOMER**

5.4.8 WWS Report

5.4.8.1 Introduction



Ctry From	Ctry To	Date	Currency	Profit Center	Local Item No.	Country	Mfg. Location	Type	Units	Sales Value	Total COGS
010	000	200407	EUR	028W	004524	AT	IE	A	3102	4870	2761
010	000	200407	EUR	028W	004525	AT	IE	A	15347	54789	32536
010	000	200407	EUR	028W	004526	AT	IE	A	16300	68786	41076
010	000	200407	EUR	028W	004555	AT	IT	A	293	2625	1579
010	000	200407	EUR	028W	006412	AT	IE	A	2182	2291	1353
010	000	200407	EUR	028W	006413	AT	IE	A	26090	59224	34700
010	000	200407	EUR	028W	006414	AT	IE	A	11033	25486	14895
010	000	200407	EUR	028W	006619	AT	IE	A	119	789	401

The WWS (world wide sales) Report is the format used to report data for the data warehouse of Wyeth and displays sales information following corporate format requirements.

5.4.8.2 Structure

Report Layout:

Ctry From:	Country code of the reporting country (this is always '010')
Ctry To:	Country code of the report receiving country (this is always '000')
Date:	Standardized date of the month
Currency:	Currency (EUR for Austria).
Profit Center:	This is always '028W' for Austria
Local Item No:	Product number of products sold in Austria
Country:	2-letter Country Code (AT = Austria)
Mfg. Location:	Mfg. Location of the product.
Type:	Datatype ('A' for actuals and 'F' for free goods).
Units:	Sales units.
Value:	Sales Value.
Total COGS:	Sales units multiplied with Std COGS.

5.4.8.3 Construction

The stored procedure **`PKG_REPORT.GetWWSReport`** is started with the following parameters:

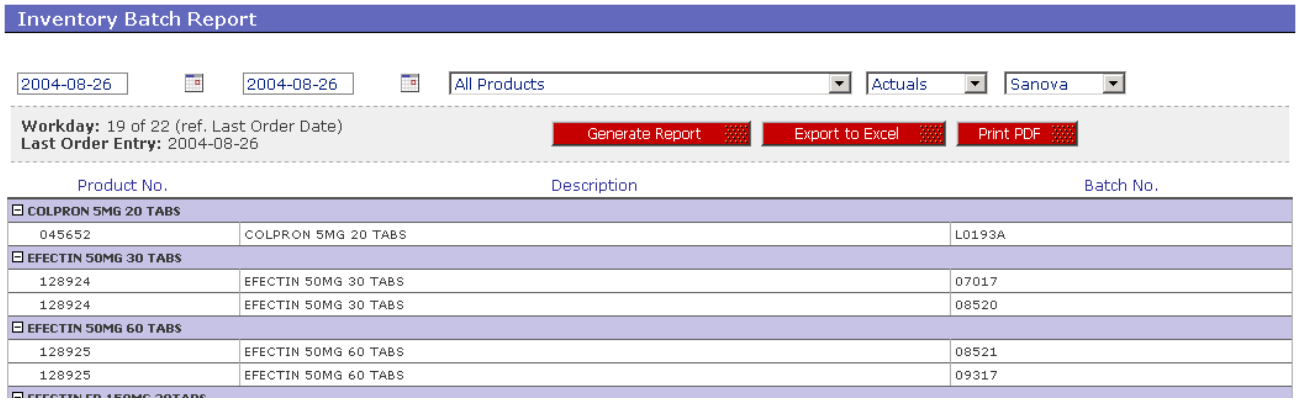
- `StartDate`
- `ctry_id` (country ID)

The procedure selects columns from:

- v_WWS
  - ↳ MV_SALES
    - ↳ T_CUSTOMER
    - ↳ T_CUSTOMER_GROUP
    - ↳ T_CUST_DISTRIB
    - ↳ T_ORDER_HEAD
    - ↳ T_TRANSMISSION
    - ↳ T_ORDER_POSITION
    - ↳ T_PRODUCT
    - ↳ T_PRODUCT_GROUP
    - ↳ T_LINE
    - ↳ T_COUNTRY
    - ↳ T_CURRENCY
    - ↳ T_TARGET_PG
    - ↳ T_CODE
    - ↳ T_COGS_HISTORY

5.4.9 Inventory Batch Report

5.4.9.1 Introduction



This Report displays the batches of products on stock in the selected time period.

5.4.9.2 Structure

Report Layout:

Product No.: Product number (Phznr).

Presentation: Product presentation.

Batch No.: Batch number.

5.4.9.3 Construction

The stored procedure **`PKG_REPORT.GetInvlBatchReport`** is started with the following parameters:

- `StartDate`
- `EndDate`
- `dist_id` (Distributor ID (Sanova))
- `line_id` (Actuals, Samples,…)
- `prod_id` (Product ID)

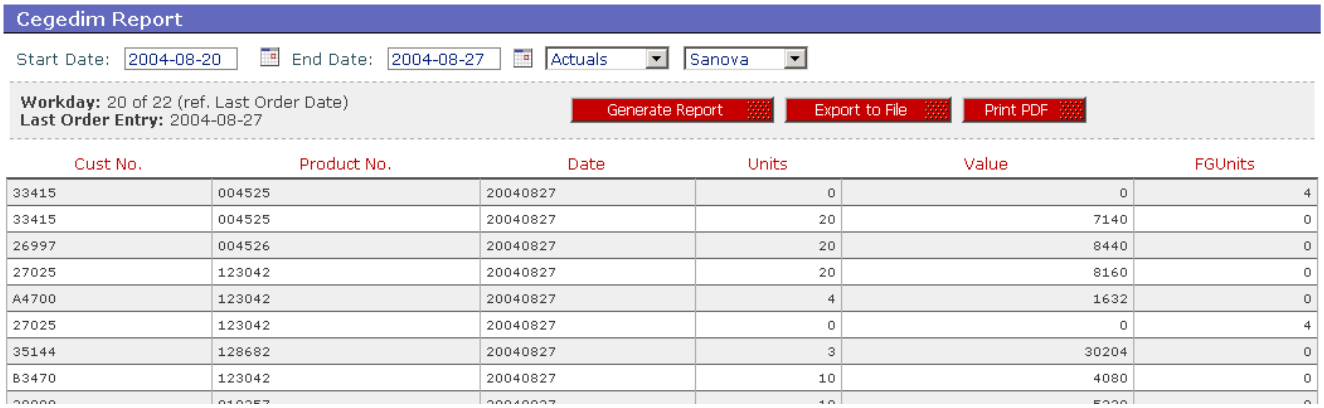
The procedure selects columns from:

- **`v_inventory_batch`**
  - ↳ **`T_INVENTORY_LEVEL`**
  - ↳ **`T_PRODUCT`**

5.5 Other Reports

5.5.1 Cegedim Report

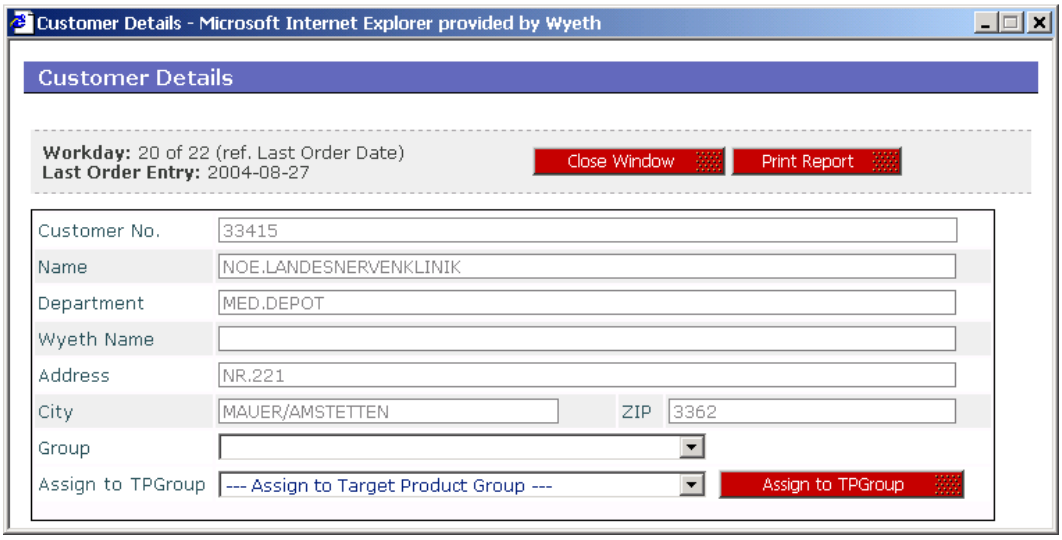
5.5.1.1 Introduction



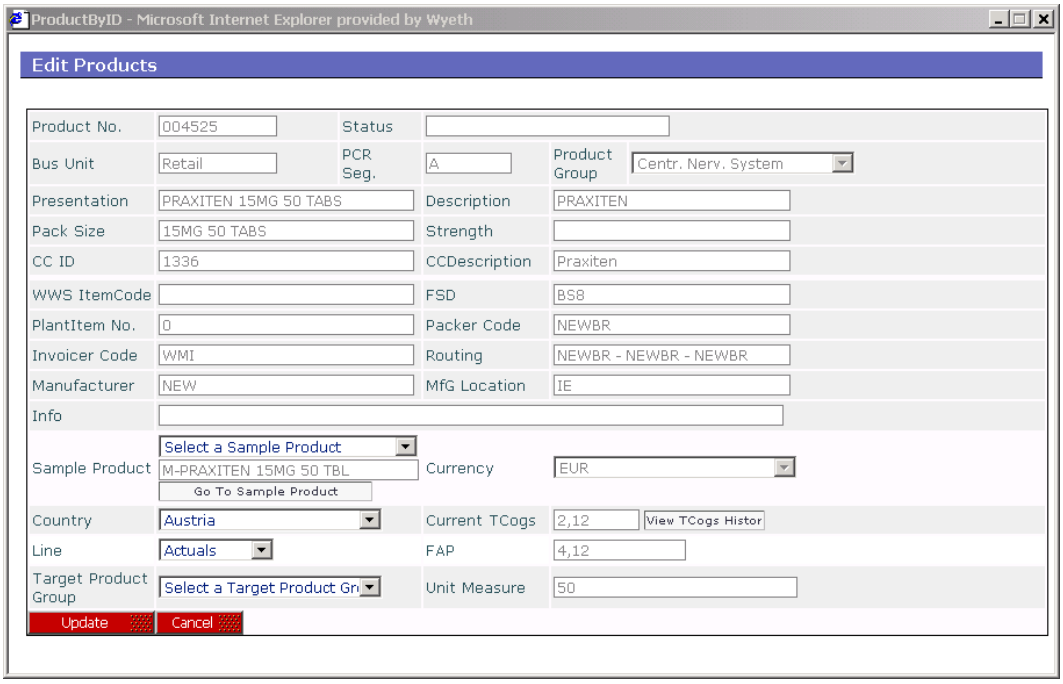
Cust No.	Product No.	Date	Units	Value	FGUnits
33415	004525	20040827	0	0	4
33415	004525	20040827	20	7140	0
26997	004526	20040827	20	8440	0
27025	123042	20040827	20	8160	0
A4700	123042	20040827	4	1632	0
27025	123042	20040827	0	0	4
35144	128682	20040827	3	30204	0
B3470	123042	20040827	10	4080	0

This report displays the sales in the correct format for importing it into the Cegedim “Teams” system. Austria downloads this report into Excel and formats it to Cegedim Austria for reporting on a weekly basis.

By clicking on a customer the following Customer Details window will pop up:



By clicking on a product the following Product Details window will pop up:



5.5.1.2 Structure

Report Layout:

Customer No.: Customer number.

Product No.: Product number (Phznr).

Date: Order day in the standardized Cegedim format

Units: Sales units.

Value: Sales Value.

FG Units: Free goods units.

5.5.1.3 Construction

The stored procedure **`PKG_REPORTS.GetSalesStatCustomer2`** is started with the following parameters:

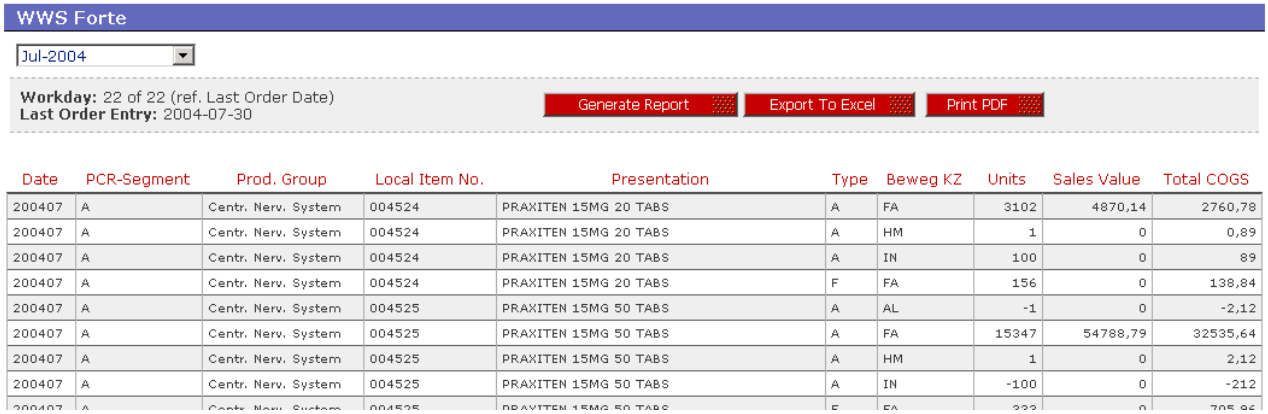
- `dateFrom`
- `dateTo`
- `line_id` (Actuals, Samples,…)

The procedure only selects columns of MV_SALES.

- **MV_SALES**
  - ↪ **T_CUSTOMER**
  - ↪ **T_CUSTOMER_GROUP**
  - ↪ **T_CUST_DISTRIB**
  - ↪ **T_ORDER_HEAD**
  - ↪ **T_TRANSMISSION**
  - ↪ **T_ORDER_POSITION**
  - ↪ **T_PRODUCT**
  - ↪ **T_PRODUCT_GROUP**
  - ↪ **T_LINE**
  - ↪ **T_COUNTRY**
  - ↪ **T_CURRENCY**
  - ↪ **T_TARGET_PG**
  - ↪ **T_CODE**
  - ↪ **T_COGS_HISTORY**

5.5.2 WWS FORTE

5.5.2.1 Introduction



Date	PCR-Segment	Prod. Group	Local Item No.	Presentation	Type	Beweg KZ	Units	Sales Value	Total COGS
200407	A	Centr. Nerv. System	004524	PRAXITEN 15MG 20 TABS	A	FA	3102	4870,14	2760,78
200407	A	Centr. Nerv. System	004524	PRAXITEN 15MG 20 TABS	A	HM	1	0	0,89
200407	A	Centr. Nerv. System	004524	PRAXITEN 15MG 20 TABS	A	IN	100	0	89
200407	A	Centr. Nerv. System	004524	PRAXITEN 15MG 20 TABS	F	FA	156	0	138,84
200407	A	Centr. Nerv. System	004525	PRAXITEN 15MG 50 TABS	A	AL	-1	0	-2,12
200407	A	Centr. Nerv. System	004525	PRAXITEN 15MG 50 TABS	A	FA	15347	54788,79	32535,64
200407	A	Centr. Nerv. System	004525	PRAXITEN 15MG 50 TABS	A	HM	1	0	2,12
200407	A	Centr. Nerv. System	004525	PRAXITEN 15MG 50 TABS	A	IN	-100	0	-212

This report is an alternative WWS view of the sales with additional data for import into the FORTE system.

5.5.2.2 Structure

Report Layout:

Date:	Order month in the standardized FORTE format
PCR Segment:	PCR Segment (Prod_Segment).
Prod. Group:	Product group
Local Item No:	Product number.
Presentation:	Product presentation.
Type:	Datatype ('A' for actuals and 'F' for free goods).
BewegKZ:	Code of the stock movement. The code tells the kind of movement in case of special transactions.
Units:	Sales units.
Sales Value:	Sales Value.
Total COGS:	Sales units multiplied with Std COGS.

5.5.2.3 Construction

The stored procedure **`PKG_REPORT.GetWWSFORTEReport`** is started with the following parameters:

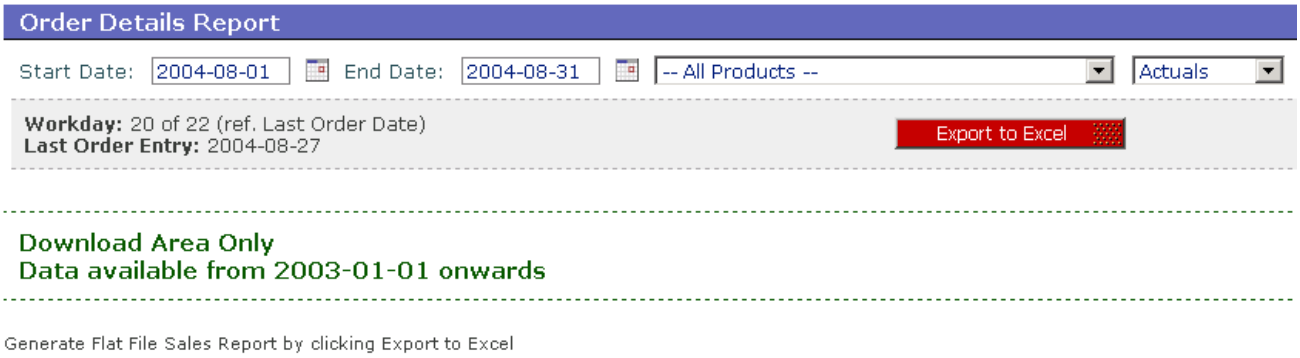
- `StartDate`

The procedure selects columns from:

- **`V_WWS_FORTE`**
  - ↳ **`MV_SALES`**
    - ↳ **`T_CUSTOMER`**
    - ↳ **`T_CUSTOMER_GROUP`**
    - ↳ **`T_CUST_DISTRIB`**
    - ↳ **`T_ORDER_HEAD`**
    - ↳ **`T_TRANSMISSION`**
    - ↳ **`T_ORDER_POSITION`**
    - ↳ **`T_PRODUCT`**
    - ↳ **`T_PRODUCT_GROUP`**
    - ↳ **`T_LINE`**
    - ↳ **`T_COUNTRY`**
    - ↳ **`T_CURRENCY`**
    - ↳ **`T_TARGET_PG`**
    - ↳ **`T_CODE`**
    - ↳ **`T_COGS_HISTORY`**
- **`T_PRODUCT`**

5.5.3 Order Details Report

5.5.3.1 Introduction



With the Order Details Report it is possible to export detailed order information to Excel for certain or all products for a certain period of time.

5.5.3.2 Structure

The export information holds the following information.

Columns:

- Order Date
- Segment
- Product Group
- Description
- Product No.
- Presentation
- Customer Group
- Customer Name
- ZIP
- City
- Customer No.
- Order No.
- Units
- Value
- FGUnits
- FGValue
- STD Cogs
- Total Cogs

5.5.3.3 Construction

The stored procedure **`PKG_REPORTS.GetSalesOrders`** is started with the following parameters:

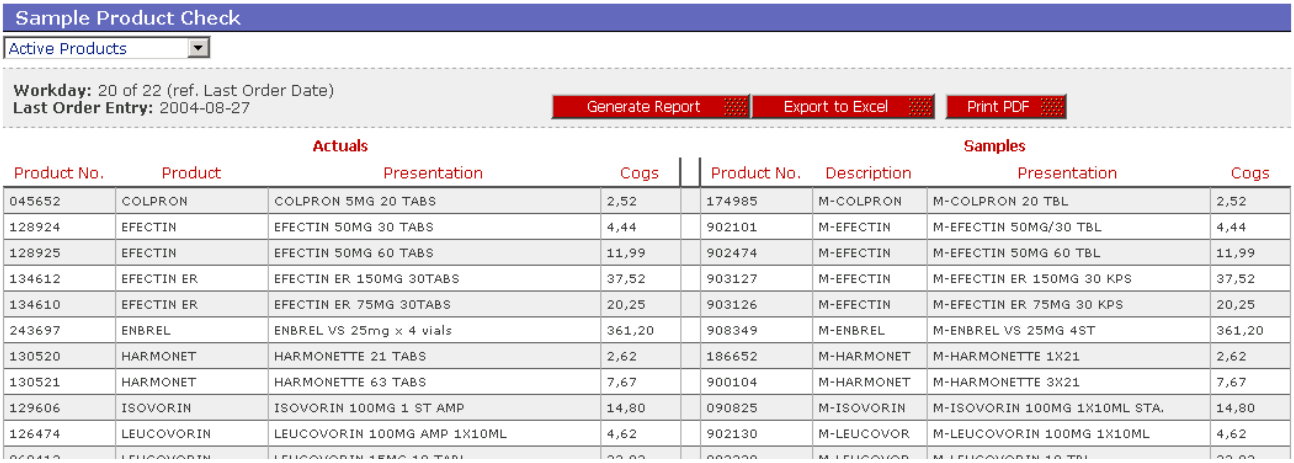
- `startDate`
- `endDate`
- `line_id` (Actuals, Samples,…)
- `ctry_id` (country ID)
- `product_desc` (Product description)

The procedure only selects columns of MV_SALES.

- **MV_SALES**
  - ↳ **T_CUSTOMER**
  - ↳ **T_CUSTOMER_GROUP**
  - ↳ **T_CUST_DISTRIB**
  - ↳ **T_ORDER_HEAD**
  - ↳ **T_TRANSMISSION**
  - ↳ **T_ORDER_POSITION**
  - ↳ **T_PRODUCT**
  - ↳ **T_PRODUCT_GROUP**
  - ↳ **T_LINE**
  - ↳ **T_COUNTRY**
  - ↳ **T_CURRENCY**
  - ↳ **T_TARGET_PG**
  - ↳ **T_CODE**
  - ↳ **T_COGS_HISTORY**

5.5.4 Sample Product Check

5.5.4.1 Introduction



Actuals				Samples			
Product No.	Product	Presentation	Cogs	Product No.	Description	Presentation	Cogs
045652	COLPRON	COLPRON 5MG 20 TABS	2,52	174985	M-COLPRON	M-COLPRON 20 TBL	2,52
128924	EFECTIN	EFECTIN 50MG 30 TABS	4,44	902101	M-EFECTIN	M-EFECTIN 50MG/30 TBL	4,44
128925	EFECTIN	EFECTIN 50MG 60 TABS	11,99	902474	M-EFECTIN	M-EFECTIN 50MG 60 TBL	11,99
134612	EFECTIN ER	EFECTIN ER 150MG 30TABS	37,52	903127	M-EFECTIN	M-EFECTIN ER 150MG 30 KPS	37,52
134610	EFECTIN ER	EFECTIN ER 75MG 30TABS	20,25	903126	M-EFECTIN	M-EFECTIN ER 75MG 30 KPS	20,25
243697	ENBREL	ENBREL VS 25mg x 4 vials	361,20	908349	M-ENBREL	M-ENBREL VS 25MG 4ST	361,20
130520	HARMONET	HARMONETTE 21 TABS	2,62	186652	M-HARMONET	M-HARMONETTE 1X21	2,62
130521	HARMONET	HARMONETTE 63 TABS	7,67	900104	M-HARMONET	M-HARMONETTE 3X21	7,67
129606	ISOVORIN	ISOVORIN 100MG 1 ST AMP	14,80	090825	M-ISOVORIN	M-ISOVORIN 100MG 1X10ML STA.	14,80
126474	LEUCOVORIN	LEUCOVORIN 100MG AMP 1X10ML	4,62	902130	M-LEUCOVOR	M-LEUCOVORIN 100MG 1X10ML	4,62

This report compares the TCOGS of actual products to their sample products and due to this shows which sample product is assigned to an actual product.
It is possible to check active and obsolete products via “drop down” box selection).

5.5.4.2 Structure

Report Layout:

Product No: Product number (Phznr).

Product: Product name.

Presentation: Product presentation.

COGS: Cost of Goods.

5.5.4.3 Construction

The stored procedure **`PKG_REPORTS.GetActandSamProducts`** is started with the following parameters:

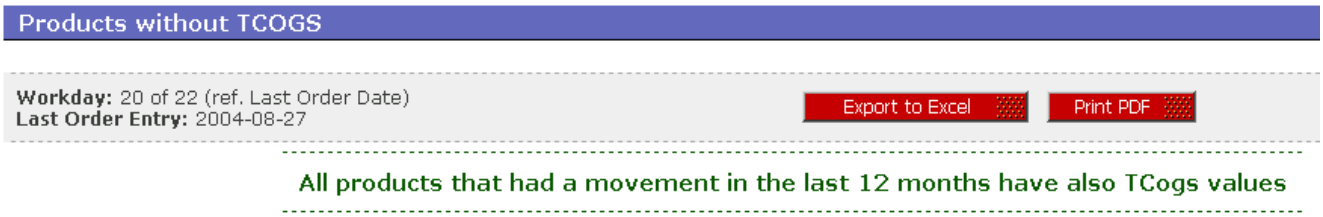
- `ctry_id` (country ID)
- `obs_code` (if 0 then active products otherwise obsolete products)

The procedure only selects columns of V_PRODUCT.

- **V_PRODUCT**
  - ↳ **T_PRODUCT**
  - ↳ **T_COGS_HISTORY**
  - ↳ **T_LINE**

5.5.5 TCOGS Check

5.5.5.1 Introduction



This Check Report identifies products without COGs.

If all active products (products which had a movement in the last 12 months) have TCOGS the message above appears.

5.5.5.2 Construction

The stored procedure **`Pkg_Application.GetProductsWithNoTCOGS`** is started with the following parameters:

- `ctry_id` (country ID)

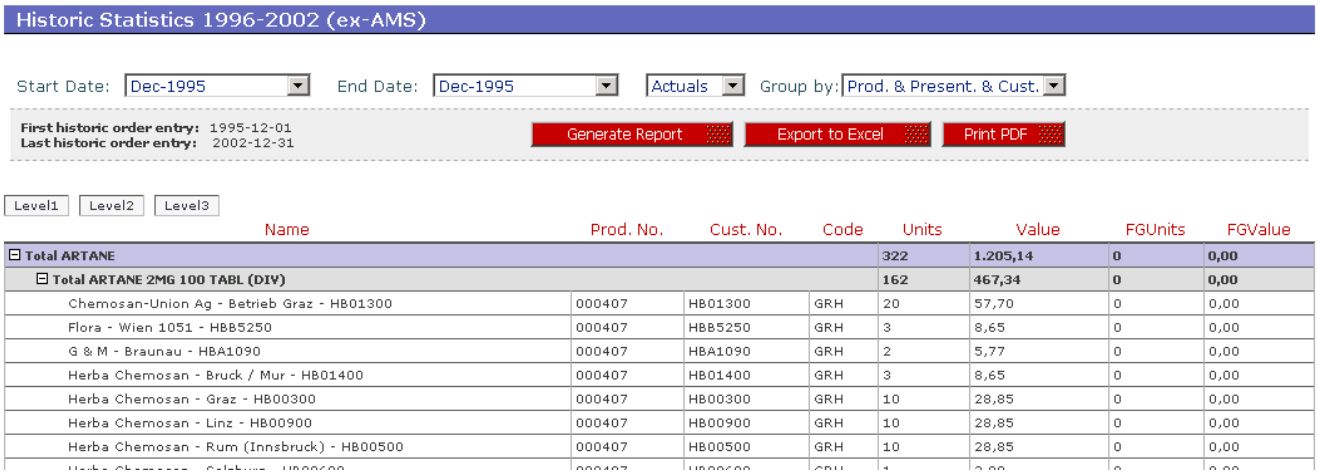
The procedure selects columns from:

- **V_PRODUCT**
  - ↳ **T_PRODUCT**
  - ↳ **T_COGS_HISTORY**
  - ↳ **T_LINE**

- **T_ORDER_POSITION**
- **T_TRANSMISSION**
- **T_ORDER_HEAD**
- **T_INVENTORY_LEVEL**
- **T_STOCK**

5.5.6 Historic Statistics 1996-2002 (ex-AMS)

5.5.6.1 Introduction



Name	Prod. No.	Cust. No.	Code	Units	Value	FGUnits	FGValue
Total ARTANE				**322**	**1.205,14**	**0**	**0,00**
Total ARTANE 2MG 100 TABL (DIV)				**162**	**467,34**	**0**	**0,00**
Chemosan-Union Ag - Betrieb Graz - HB01300	000407	HB01300	GRH	20	57,70	0	0,00
Flora - Wien 1051 - HBB5250	000407	HBB5250	GRH	3	8,65	0	0,00
G & M - Braunau - HBA1090	000407	HBA1090	GRH	2	5,77	0	0,00
Herba Chemosan - Bruck / Mur - HB01400	000407	HB01400	GRH	3	8,65	0	0,00
Herba Chemosan - Graz - HB00300	000407	HB00300	GRH	10	28,85	0	0,00
Herba Chemosan - Linz - HB00900	000407	HB00900	GRH	10	28,85	0	0,00
Herba Chemosan - Rum (Innsbruck) - HB00500	000407	HB00500	GRH	10	28,85	0	0,00

This report shows the 'old' sales information of the former AMS system, which was the old reporting system for sales from 1996 to 2002 (as the ALF-System holds all sales starting from Jan 2004).

5.5.6.2 Structure

Report Layout:

Name: Customer name.

Prod. No: Product number (Phznr).

Cust. No: Customer number.

Product: Product name.

Presentation: Product presentation.

COGS: Cost of Goods.

5.5.6.3 Construction

The stored procedure **`Pkg_Reports.GetAMSSalesStat`** is started with the following parameters:

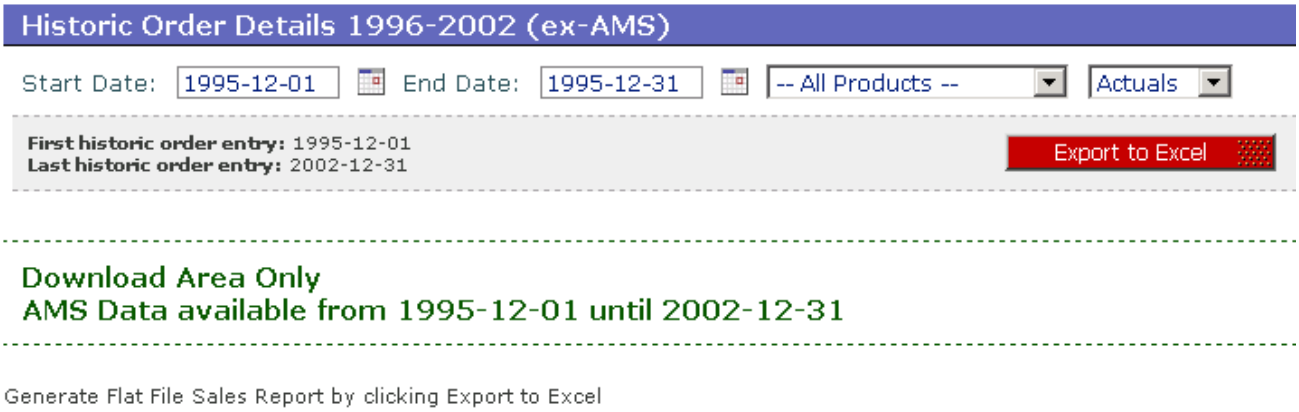
- `StartDate`
- `endDate`
- `line_type` (WYL (for actuals) or MWL (for samples) (old declaration))

The procedure selects columns from:

- **MV_AMS_STATISTICS_WYL**
  - ↳ **AMS_T_STATARTKUND_WYL**
  - ↳ **AMS_T_PRODUKT_WYL**
  - ↳ **AMS_T_KUNDE**
- **MV_AMS_STATISTICS_MWL**
  - ↳ **AMS_T_STATARTKUND_MWL**
  - ↳ **AMS_T_PRODUKT_MWL**
  - ↳ **AMS_T_KUNDE**

5.5.7 Historic Order Details 1996-2002 (ex-AMS)

5.5.7.1 Introduction



This report provides means to export the historic order details to Excel.

5.5.7.2 Structure

Columns:

- Order Date
- Product
- Presentation
- Product Group
- Phznr
- Auftrags Nr
- KdNr.
- KonzNr
- Name
- Name
- Units
- Value
- FGUnits

5.5.7.3 Construction

The stored procedure **`Pkg_Reports.GetAMSOrderDetails`** is started with the following parameters:

- `StartDate`
- `endDate`
- `line_type:` WYL - for actuals or MWL - for samples (old declaration)
- `product` (Product name)

The procedure selects columns from:

- **MV_AMS_ORDER_DETAILS_WYL**
  - ↳ **AMS_T_FAKTHIST_WYL**
  - ↳ **AMS_T_FAKTHISTPOS_WYL**
  - ↳ **AMS_T_FIRMEN_WYL**
  - ↳ **AMS_T_KUNDE**
  - ↳ **AMS_T_PRODUKT_WYL**
  - ↳ **AMS_T_SANBEZ**
- **MV_AMS_ORDER_DETAILS_MWL**
  - ↳ **AMS_T_FAKTHIST_MWL**
  - ↳ **AMS_T_FAKTHISTPOS_MWL**
  - ↳ **AMS_T_FIRMEN_MWL**
  - ↳ **AMS_T_KUNDE**
  - ↳ **AMS_T_PRODUKT_MWL**
  - ↳ **AMS_T_SANBEZ**

6 Master Data Interfaces

6.1 Introduction

The Master Data Interfaces provide ways of administering important data of the ALF-System explicitly designed for the needs of master users.

Master Data only handles data explicitly designed for master users. Data which is important for the system (like the menu or application settings) are administered in the Admin menu.

In this chapter interfaces are not described in all detail since they all work in the same way.

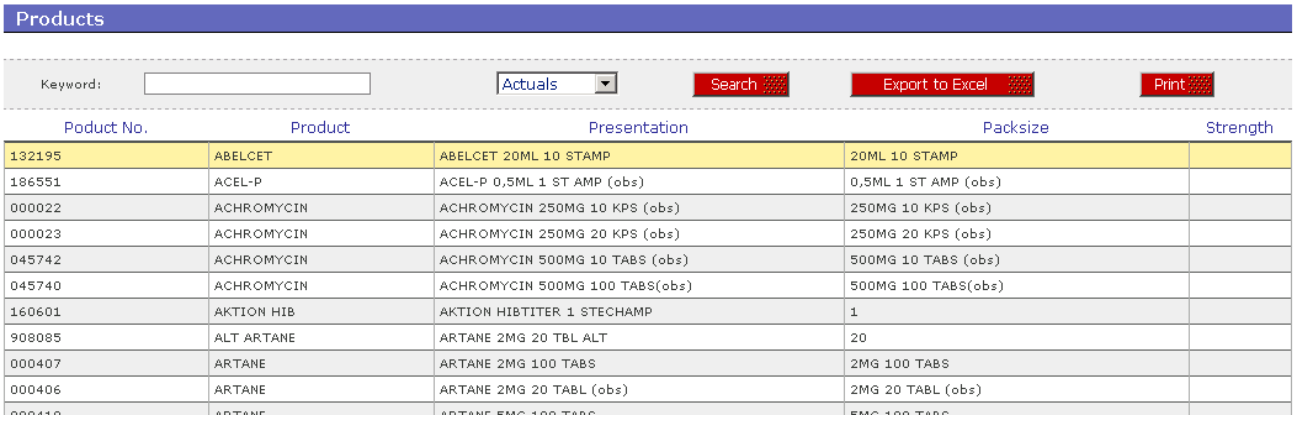
6.2 The Interfaces in General

6.2.1 Introduction

The interfaces are designed to be used in two-fold:

- The master interface displays information in overview.
- After choosing a record, the detail interface will be displayed (with all administrable fields of the dataset)

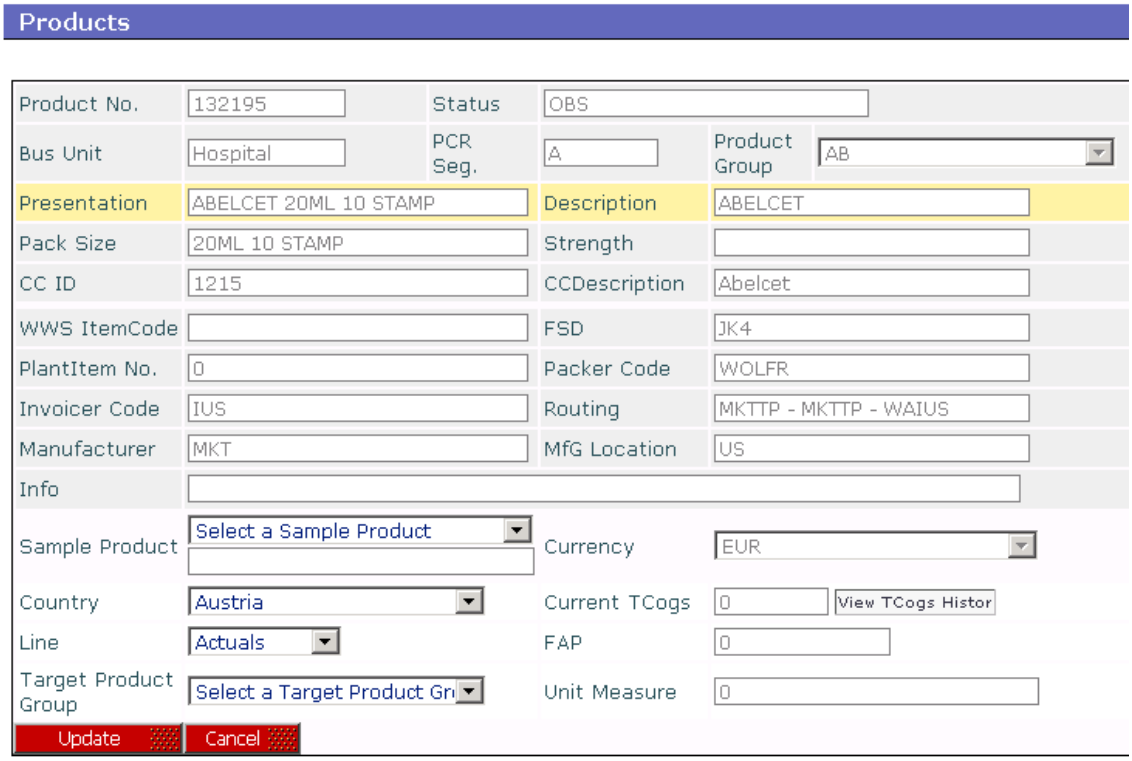
6.2.2 The Master Interface



The master interfaces display the main information in rows and optionally provide the possibility to filter the record set with keywords or dropdown boxes, and to sort the data (if the header is red, by clicking on the specific header).

By clicking on a record or line, the details of the record will be displayed, as shown in the next chapter.

6.2.3 The Details Interface



The details interface shows all relevant data of the record.

The data is displayed in two colors:

- If the letters are **grey**, the field is disabled and the information cannot be changed. For example: if the information comes from Sanova or FORTE, the data is maintained in their systems and ALF only displays this foreign data.
- If the letters are **blue**, the field is enabled and it is possible for the user to add and change data. Such data is normally information, which could not be taken from other systems, and it must be administered in the ALF-System. For example: the additional information field of the product for Wyeth users, or the connection to the referencing sample product.

The detail interface allows to save changes with the update button or to discard them with cancel.

6.2.4 Important Master Data Activities

6.2.4.1 New Customers

It is important for correct calculation of area sales that customers are assigned to a Target Product Group (TPG) and to sales reps in that TPG because if customers are not assigned, the sales of these customers will not be added to the target statistics of the sales reps who serves them.

The New Customer interface displays all customers which are not assigned to a Target Product Group and need to be assigned.

6.2.4.2 Customers

On the customer detail interface it is only possible to change the following fields:

- Short Name: The Short Name is required for the (semi-)automatic generation of SMS messages sent to the sales reps. It should be as brief as possible (<20 characters) to fit well into the SMS.
- Group: The Group is the customer group which identifies the customer's type of enterprise. To-date there are no reports considering the group. The customer master interface gives the possibility to display all customers of a group.

6.2.4.3 Countries

The main part of the country information is taken from the FORTE system; due to this the only fields which need to be administered are the currency, and the Balance Sheet (BS) and the Profit and Loss (PL) fields, which are needed for the accounting records (see below).

6.2.4.4 FX-Rates

All Currency Exchange Rates (FX-Rates) are taken from the FORTE system; it is not possible to change any information in the ALF-System.
The interface provides the possibility to take a look the currencies and their rates taken for the currency calculation.

6.2.4.5 Holidays

The holidays are automatically created by starting **`Pkg_Calendar.add_holidays_of_a_year`** with a desired year. This stored procedure can only be executed directly in the database.

The routine considers the following holidays:

- Good Friday
- Easter Day
- Easter Monday
- Ascension of Christ
- Whitsunday
- Whitmonday
- Corpus Christi Day
- New Years Day
- Epiphany (Dreikönigstag)
- Staatsfeiertag (Austria)
- Nationalfeiertag (Austria)
- Christmas
- Christ Day (Christtag)
- St. Stephens Day (Stefanitag)
- New Years Eve

With Add New Record the master data user is able to add holidays or other work free days to the system.

6.2.4.6 Products

With the product details interface it is possible to change the following fields:

- Sample Prod: The Sample product field creates a connection between actual and sample products. To create this connection the users chooses an actual product and assigns it to the respective sample product.
- Country: The country is currently information not yet used, since ALF is prepared to service other countries but the only country currently used is Austria, which is also set as default country.
- Line: If the product was mistakenly reported (by Sanova) with an incorrect line code, it is possible to correct this error here.
- Target Prod G: This is a shortcut to assign the current product to a Target Product Group. The product will then appear in the target reports.

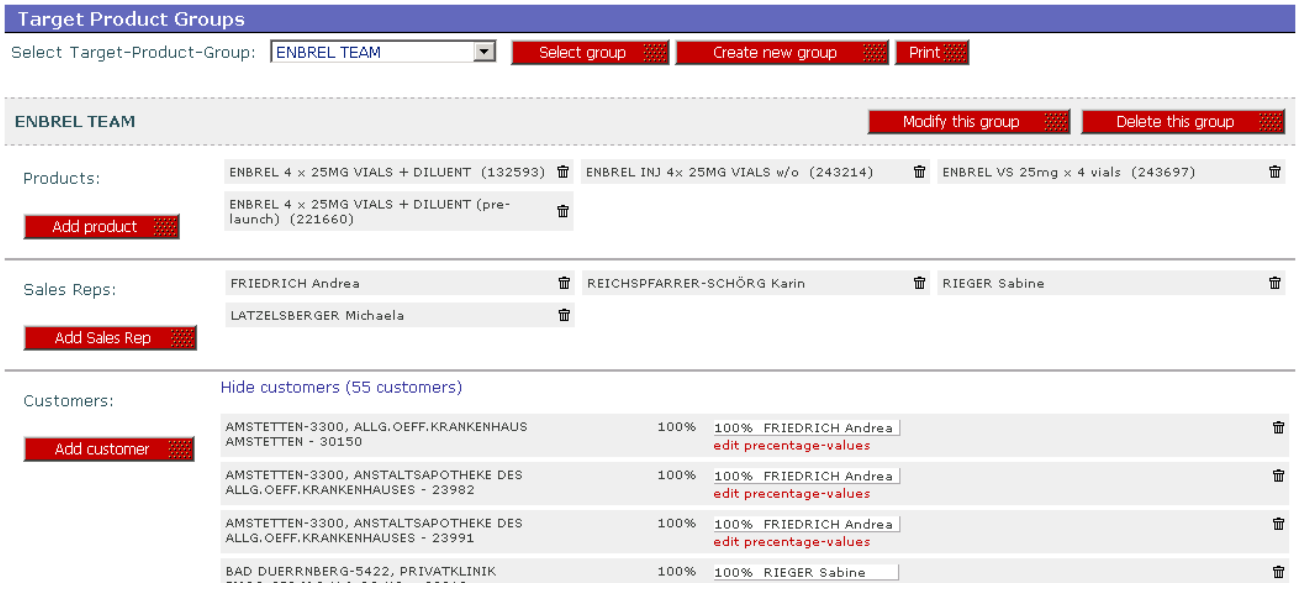
6.2.4.7 Sales Reps

Since the sales reps are entities who need to be defined in the ALF system, the administration can be done here.

It is important to assign the sales rep entry to the corresponding intranet user. This connection is neede for sending the TPG Quick Statistics per SMS to the mobile phone of the sales rep (the mobile phone number is stored in the intranet users table).

6.2.4.8 Target Product Groups

The Target Product Groups administration is quite complicated and will therefore be described in detail:



Master data users do not only administer the TPG, but also their products, their assigned sales reps and furthermore the customers of the sales reps.

Clicking the button 'Create new group' opens the pop-up to create new TPGs. A TPG is identified only by its name. After saving it, the process of editing/maintaining can be started.
First, the user chooses a TPG and clicks the button 'Select group', then the main part (concerning specific TPG) appears.

By clicking 'Modify this group' users can change the name of the TPG.

In the Products part the user can assign (currently unassigned) products to the TPG or can delete already assigned products by clicking the trash bin icon.

In the Sales Reps section unassigned sales reps can be assigned, and already assigned ones can be removed from the TPG.

In the Customers section customers are administered. After selecting a customer in the customers pop up the customers need to be assigned to the sales reps. Every customer must be assigned with 100% to the sales reps. It is not possible to save an assignment if a percentage is below or above 100%.

7 Admin interfaces

7.1 Introduction

The Admin Interfaces are to be used for system relevant parameters and functions and require in-depth knowledge of system processes.

Only staff with ALF admin rights has access to these interfaces.

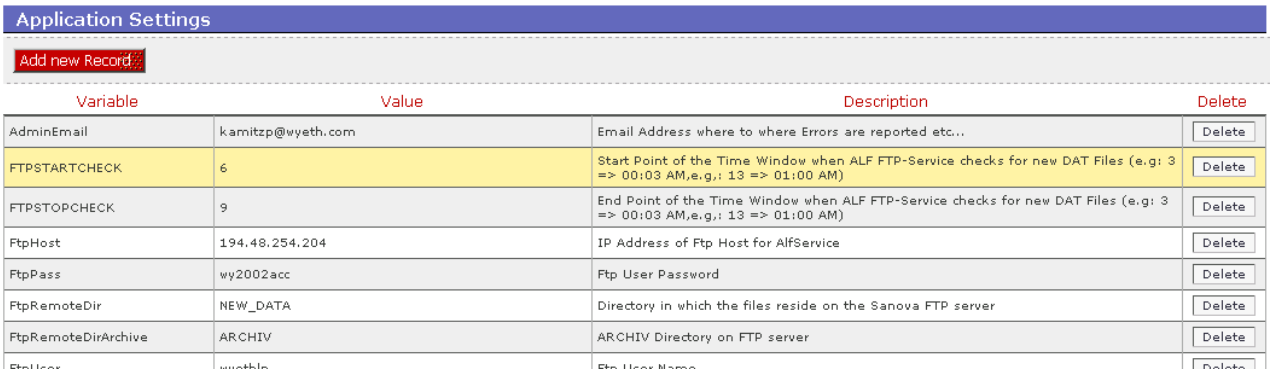
The data administering interfaces work the same way as the Master Data administering interfaces, with master and detail views.

7.2 Administering Data

7.2.1 Introduction

This chapter describes the interfaces which administer data as well as the effects if specific data is changed.

7.2.2 Application Settings



The Application Setting Parameters are used for the ALF – Service (see the chapter ALF-Service) to control the data imports.

Before importing, the service fetches the parameters from the database (selected by the variable name – **so do not change the variable names**!!!) to use them for the importing process.

The ranges and the meanings of the values are described in the Description field.

New records will only make sense if the ALF application code or the service is changed.

7.2.2.1 ALF Config File

The ALF Config File [ALF.config] is in XML format and is located in the application root folder on the ALF application server. It holds database and administrative settings. File locations see below.

!!The file needs to be changed if server names, connection users or passwords or URLs change!!

Content:

```
<?xml version="1.0" encoding="utf-8" standalone="yes" ?>
<applicationSettings>
  <!--Database settings-->
<DB_CONNECTION_STRING_LIVE>User Id=alf2;Password=alf2;Data Source=dintra;Pooling=False; Connection
Lifetime=20; Connection Timeout=10; Max Pool Size=3; Min Pool Size=2; Incr Pool Size=2; Decr Pool
Size=2</DB_CONNECTION_STRING_LIVE>
<DB_CONNECTION_STRING_DEV>User Id=alf2;Password=alf2;Data Source=dintra;Pooling=False; Connection
Lifetime=20; Connection Timeout=10; Max Pool Size=3; Min Pool Size=2; Incr Pool Size=2; Decr Pool
Size=2</DB_CONNECTION_STRING_DEV>

  <!--Administrative settings-->
  <APPLICATION_URL_LIVE> http://aece.insidewyeth.com/application_alf/ </APPLICATION_URL_LIVE>
  <APPLICATION_URL_DEV> http://dev-aece.insidewyeth.com/application_alf/ </APPLICATION_URL_DEV>
  <APPLICATION_URL_LOCALHOST>http://localhost/alf/</APPLICATION_URL_LOCALHOST>

  <MACHINE_NAME_LIVE>vi01w34</MACHINE_NAME_LIVE>
  <MACHINE_NAME_DEV>vi01w32</MACHINE_NAME_DEV>

  <COMPANY_NAME>Wyeth</COMPANY_NAME>

  <EMAIL_SERVER_ADDRESS>10.248.57.234</EMAIL_SERVER_ADDRESS>

  <!--Directories -->
  <LOG_FILE_PATH>c:\inetpub\wwwroot\logfiles\</LOG_FILE_PATH>
</applicationSettings>
```

File locations:

Development server: \\vi01w32\d$\oratest\application_alf\ALF.config

Live server \\vi01w34\d$\intranet\application_alf\ALF.config

NOCH KORREKT ??? REMARK V. CHRIS STEHLE

7.2.3 Codes

The Codes Admin Interface shows all entries of the T_CODE table in a direct way and gives access to change, insert and delete code, category and description, as long as the database constraints are not violated.

The different codes are used to mark database entries with different flags. In order to reduce numbers of tables, all codes (except T_Line & T_Statistic_Type) are consolidated in the T_Code table; codes are identified by category.

7.2.4 Menu

The menu which is configured in the Menu Admin Interface is the menu of the ALF System Web Interface (the Home, Reports, Master Data, Admin, … etc. entries).

As it is important to the admin to know what the information means - all administrable columns are described here:

Columns:

Name:	Internal name of the menu.
Label:	Label of the menu entry (the name which is shown in the menu)
Link:	To which .aspx file the menu links.
Target Frame:	In which frame the .aspx file will be opened.
Display No.:	Manages the order in which the menus are displayed.
Display Menu:	If this flag is ticked the menu will appear in the system, if not it is hidden.
Category:	Additional Category of the menu. The category of the entry and its parent need to be equal.
Module Section:	Connection to the access rights management of the intranet. (Intranet user ACCESS table)
Parent ID:	Connection to the parent menu entry.

The Menu Interface entries are ordered by its parent menu ID (due to this all sub menu entries of a parent menu are grouped together).

7.2.5 Distributors

Distributors are companies selling products of Wyeth and whose sales details are gathered and reported using the ALF System.

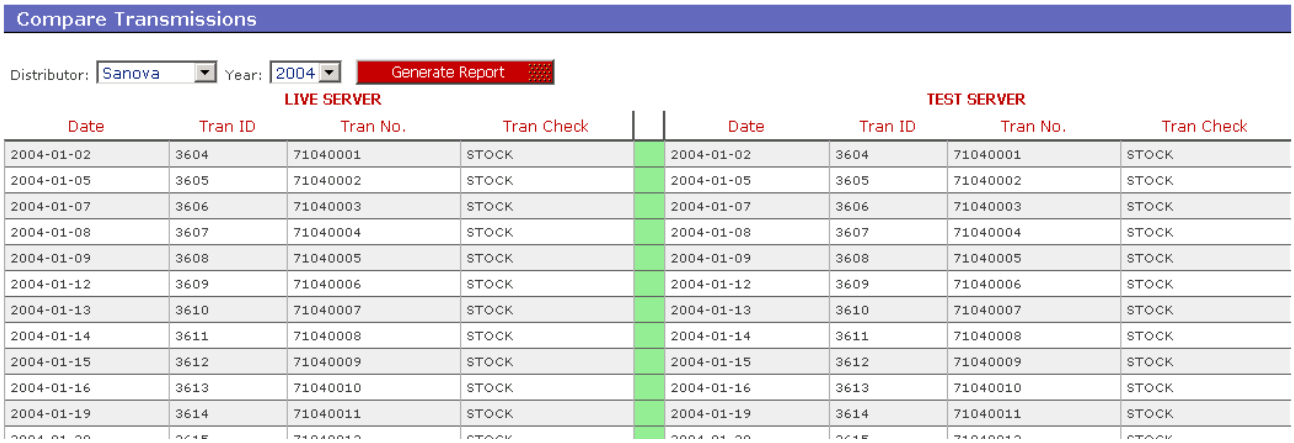
Currently the system holds information of only one distributor (Sanova / Herba-Chemosan). Therefore all routines, which manage the data import, are written for the format and the data environment of Sanova. This means in case there will be other distributors with (their own) different data representation, it would be necessary to write new/additional importing routines.

This administration menu only exists in a 'pro forma' way, meaning that if there will be a new distributor the database will need to be changed. The only information which could be changed is the distributor number and the description. **Do not change the distributor name of the Sanova entry!!!!** The importing and reporting routines search the record by the name 'Sanova', so any changes would corrupt importing and reporting routines, i.e, rendering them invalid.

Columns:

Number:	Number of the distributor.
Name:	Name of the distributor. The Sanova importing routines seek the distributor by its name. Therefore do not change it!!!
Description:	Description to give additional information, like the full name of the company.

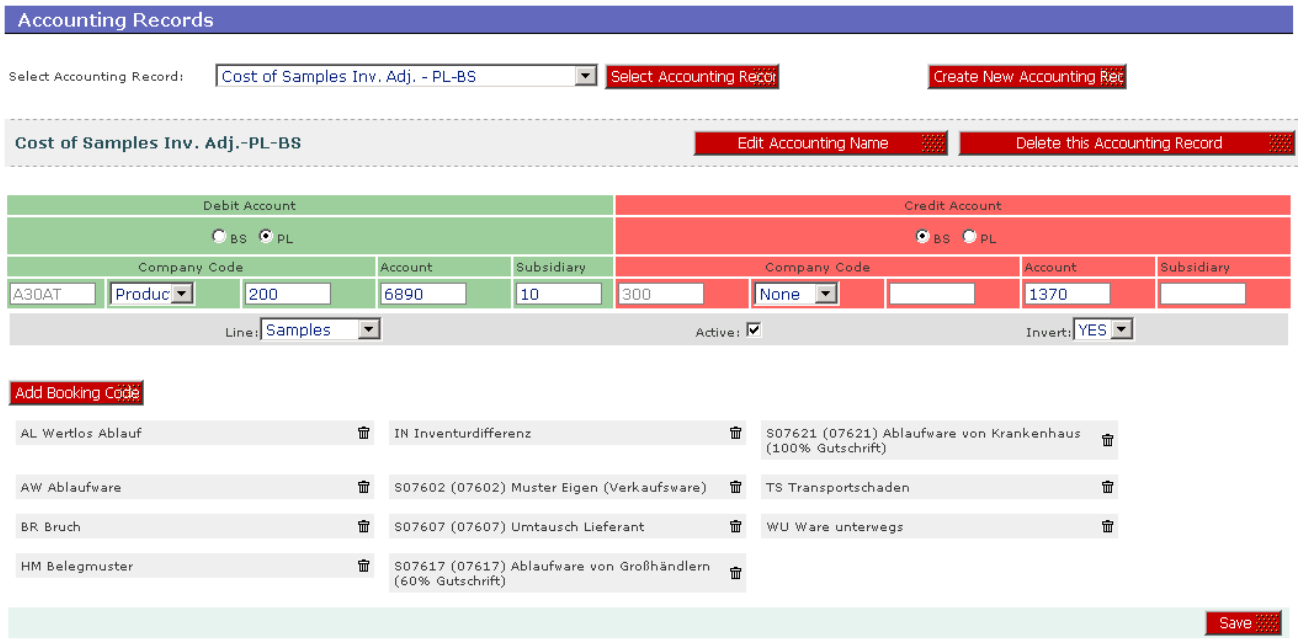
7.2.6 Compare Transmissions



The compare transmissions report compares all transmission of the live server with those on the test server.

7.2.7 Accounting Records

7.2.7.1 Introduction



The ALF Accounting Records are representations of accounting records used for the Finance Month End Report (former Stock & Cogs Report), generating booking lines which will be transferred into the JDE system.

The accounting records admin interface is the construction system to create accounting records.

7.2.7.2 Administration

First it is required to select an accounting record (or create a new by giving it a name and a description) to see the details of the record.

The details of the accounting record will ultimately be used in the JDE export lines.

7.2.7.3 Debit Account and Credit Account

There are two sides of every accounting record: the Debit Account side and the Credit Account side (i.e., plus and minus)

It is very important for the correct function of a record to define it as a Balance Sheet (BS) to Profit and Loss (PL) accounting record; or as a Balance Sheet to Balance Sheet or as a Profit and Loss to Profit and Loss record. Therefore the type needs to be selected on the debit and credit side (selecting BS or PL) on this admin form.

7.2.7.4 Fields

Every acocunting record side has five edit fields, which will tell the JDE system how to interpret the data.

Fields:

Company Code: The company code consists of the following three fields:

- **Country/Company:** The country/company field is disabled and filled by the BS- and PL-Country information of the urser's country who created the accounting record.
- **Cost Center:** The cost center field can be filled with three types of data: **General**, which means '0000'; **Product**, which means the field will be filled with the cost center information of the product; **None**, which leaves the field blank in the export.
- **Department:** Free 5-letter code.

Account: Free 5-letter code.

Subsidiary: Free 4-letter code.

These fields will identify every JDE export line of the accounting record.

7.2.7.5 Parameters

The next parameters give control of the accounting record.

Parameters:

Line: Controls the line of products for which the accounting record is created:

Active: If the record is active it will be included in the (month end) JDE Export.

Invert: Since the JDE Export uses the calculations of the Finance Month End Report, the numbers from that report could have a wrong sign. If the invert field is 'yes', the number will be multiplied with –1.

7.2.7.6 Booking Codes

The Booking Section Code defines which codes of the Finance Month End Report will be added to the accounting record, e.g. if the GITO ("Goods in Transit Open") code is added to the accounting record the values of the GITO code from the Finance Month End Report will be used for the JDE Export lines of the currently edited accounting record.

As shown in the screenshot at the beginning of this chapter, it is possible to add more than one booking code to an accounting record. To delete a booking code from the accounting record, click the trash bin icon.

7.3 Maintaining Functions

7.3.1 Introduction

The Maintenance Functions in the ALF-System are used to control and manage the importing routines: FTP download, transactional data import, master data from the FORTE system and refreshing of the materialized views.

7.3.2 FORTE Import

7.3.2.1 Introduction

The FORTE Import interface is used for importing of the master data from the FORTE System (see Master Data from FORTE).

The tables or the specific columns of some tables which are imported from the FORTE system are not administered in the ALF-System (the fields are disabled), since a new import from FORTE will discard all changes done by the user.

7.3.2.2 FORTE Products

By clicking the button 'FORTE Products' the stored procedure **`PKG_IMPORT_FORTE.F_Products`** is started.

The stored procedure connects to the FORTE system and checks the v_alf_products (view in the FORTE system exclusively for use by ALF) for changes or new entries.

The system looks at every entry of the FORTE system and checks if the entry is already available in the ALF-System, e.g. if there is a product with the same product number.

If the data is already available in the ALF database, the routine updates all fields in the ALF database with the FORTE data, as long as the FORTE data is not null

If there is a new product in the FORTE system it will be inserted into the ALF database together with all FORTE information.

7.3.2.3 FORTE Countries

By clicking the button 'FORTE Countries' the stored procedure **`PKG_IMPORT_FORTE.F_country`** is started.

The stored procedure connects to the FORTE system and checks the v_alf_country for changes and new entries.

The system looks at every entry of the FORTE system and checks if the entry is already available in the ALF-System e.g., if there is a country with the same ID, since the ID is taken from FORTE and entries are not sequenced in the ALF-System.

If the country already exists in ALF the following fields are checked if they are equal; if they are not equal they will be updated with the information from FORTE.

- Ctry_Code (country code)
- Ctry_Description (country description)

If there is a new country in the FORTE system the country will be added to the ALF-System.

7.3.2.4 FORTE Budget

By clicking the button 'FORTE Budget' the stored procedure **`PKG_IMPORT_FORTE.F_budget`** is started.

The stored procedure connects to the FORTE system and checks v_alf_budget for changes or new entries.

The system looks at every entry of the FORTE system and checks if the entry is already available in the ALF-System; e.g., if there is a budget with the same budget date (budg_date), product id (prod_id), budget code (code_id_budget_type) and line id.

If the budget is already imported into ALF, the following fields are checked if they are equal.

If they are not equal the ALF-System will be updated with the information from FORTE.

- Budg_units (budget units)
- Budg_value (budget value)

If there is a new budget in the FORTE system it will be added to the ALF-System.

7.3.2.5 FORTE TCogs

The TCogs (Transfer Cost of Goods) shows product costs for each product presentation Wyeth sells. The TCogs are taken from RPIS, which is connected to the FORTE. In the v_rpis_currentprices a view of the current RPIS proces are stored on an actual basis. ALF builds its own history of the prices: If there is a new price for a product, that price si transferred into the ALF database.
The ALF-System will also store the old price in the following way: the old price will get a date to identify that the price was current until 'yesterday' (the day before the new effective date) and from 'today' on there is a new price. As the ALF-System checks daily for updates a workable history will be built.

By clicking the button 'FORTE TCogs' the stored procedure **`PKG_IMPORT_FORTE.F_cogs`** is started.

The stored procedure connects to the FORTE system and checks the v_rpis_currentprices for changes or new entries.

The system looks at every entry of the FORTE system and checks if the entry is already imported into ALF (if there is a price for the same product and the same time period).

If there is already a price in the ALF-System it is checked if the value is the same.
If the value is not the same but the time period is equal, the system will perform a straight update to store the new price.

The ALF-System also checks if there was a **historic price change**; e.g. the old price was xx€ from the 1st of September and the new price is xy€ from the 1st of August, and updates the affected prices.

If there is a new price in the FORTE system it will be added to ALF (and the old price is invalidated by the day before).

7.3.2.6 FORTE FX Rate

Currency exchange rates are stored for every month for different cross-rates, i.e. from currency and to currency.

By clicking the button 'FORTE FX Rate' the stored procedure **`PKG_IMPORT_FORTE.F_fx_rate`** is started.

The stored procedure connects to the FORTE system and checks the v_alf_fx for changes or new entries.

The system looks at every entry of the FORTE system and checks if the entry is already imported into ALF; i.e., if there is an exchange rate with the same month (fxrt_date_from) and the same from- and to-currencies (curr_id_from and curr_id_to).

If the rate is already in the ALF-System, the value is checked if it is equal. If it is not equal, the ALF-System will be updated with the information from FORTE.

If there is a new FX-rate in the FORTE system it will be added to ALF.

7.3.2.7 FORTE Currencies

By clicking the button 'FORTE Currencies' the stored procedure **`PKG_IMPORT_FORTE.F_currency`** is started.

The stored procedure connects to the FORTE system and checks the v_alf_currency for changes or new entries.

The system looks at every entry of the FORTE system and checks if the entry is already available in the ALF-System; i.e., if there is a currency with the same ID (as the ID is taken from FORTE).

If the currency is already in ALF, the following fields are checked if they are equal. If there are not equal, ALF will be updated with information from FORTE.

- Curr_Code (country code)
- Curr_Description (country description)

If there is a new currency FORTE system it will be added to ALF.

7.3.3 MV Refresh

With this interface it is possible to start the recalculation of the materialized views manually. For example if a transmission was deleted or new data were manually imported, all sales values need to be recalculated.

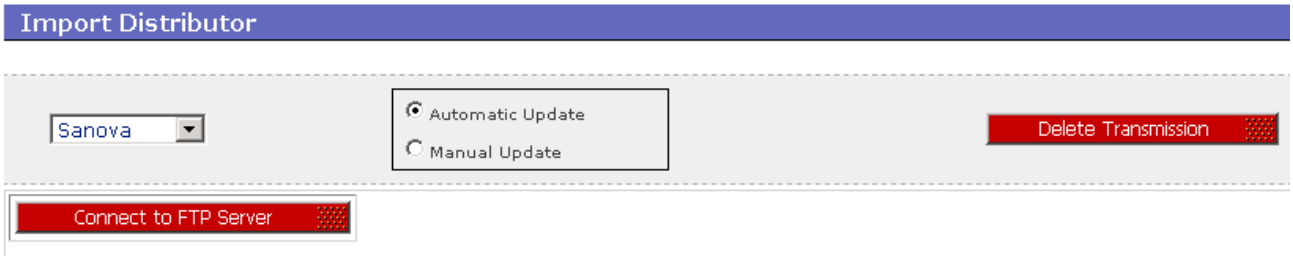
During the refreshing of the materialized views reports, which are based on MV data, will not be available and a status bar will show the progress of the MV refreshing. Approximate time for refreshing is between **4min - 6min**, depending on system load.

7.3.4 Import Distributors

The Import Distributors interface manages the manual import of transactional data, which is normally handled by the ALF Service (see ALF – Service) automatically.

However, if the situation occurs that the ALF Service did not import the transactional data from the FTP server of Sanova, this interface gives the chance to download and import the transactional data files from the FTP server.

7.3.4.1 Automatic Import



When working with the automatic update mode the administrator needs to click the 'Connect to the FTP Server' button.

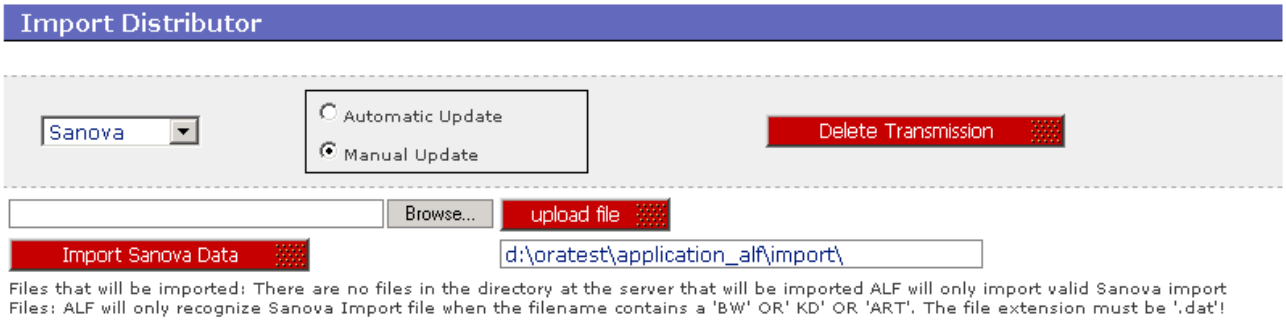
After clicking the ALF-System will connect to the FTP server of the selected distributor (currently only Sanova), a list box will appear which shows all files which are on the server.

Next step, the administrator needs to select the files to import (use CTRL-key to mark more entries).

Finally, the 'Import Selected Files' button needs to be clicked and the files will be imported with the following steps: downloaded, imported (via SQL-Loader), archived (moved to a specific folder on the server to gather all imported files with chronological information) and the importing routines will be started, which will also start the refresh of the materialized views.
A complete and detailed action report will be shown on the screen during and after the automatic update, showing the detailed actions and progress of the update process.

7.3.4.2 Manual Import

If, however, the files are not on the FTP server the manual import gives the possibility to import files which were received on an alternative route, e.g. e-mail file attachments.



(Note: this screen is from the oratest server – the live server will import into this folder: d:\intranet\application_alf\import\)

First the administrator needs to identify the file to upload. The uploading process works only for single files. To import all 3 files (the KD file for the customer information, the ART file for the product information and the BW file for the transactional data) the upload needs to be used three times.

After uploading the files the system restates which files are in the import folder with the following text: '*Files that will be imported*:' then the files will be listed.

If the administrator mistakenly imported wrong files, these files need to be deleted manually (with the Windows Explorer) from one of these folders:
(**test server**) \\vi01w32\oratest\application_alf\import\sanova_file_archive\manually_imported
(**live server**) \\vi01w34\intranet\application_alf\import\sanova_file_archive\manually_imported

It is also possible to copy the files via the Windows Explorer to the directory then there is no need to upload the files.

When the correct files are at their place the 'Import Selected Files' button needs to be clicked and the files will be imported as described in the previous chapter "Automatic Import".

7.4 Logs

An additional menu for the administrators is the Logs interface, which gives the possibility to look through the logging information of the database.

The following categories of logged information exist:

FtpLogEntry: These log entries provide information on the ALF Service running on the server. When the service was stopped or started and when it imported downloaded files from the Sanova FTP server via SQL-Loader into the database. If further gives additional summaries of the import process.

Import: These logs show when and which import routine was started, and how it ended (how many entries of each type it inserted).

ImportErr: These logs show errors that occurred during the import.

Application Exception: Application exceptions are errors of the ALF-System itself, e.g. if a user wants to see some report, but there is an internal error and the call fails, the internal error will be written into the logs.

Check: The check logs show when (and which) transmissions were checked and if the stock check found some inconsistency.

Materialized View Maintenance: This tells when the MV refresh was started and when it was finished.

New Entities imported: Tells when and which new entities (customers and products) were imported.

When a specific view is shown (Import or ImportErr, etc.) the button on the right-hand side gives the possibility to delete this type of entries from the log.

8 Processing

8.1 Introduction

ALF is designed to support the Logistics and Finance departments with month-end closing processes.

The export for JDE is done once a month (at the beginning of the month for the preceding month) – this procedure is called month end processing ('Monatsabschluß').

Therefore the system needs to freeze all historic data which must not change after the month-end processing and JDE export has been accomplished for each month.

8.2 Processing Concept

In order to provide consistent data, ALF works with a two-phase concept.
First, the Logistics department prepares all data for month-end processing and finally closes the month.
Second, the Finance department approves the Logistics closing and in return prepares the financial month-end closing. All closes can be rolled back, except the JDE export.

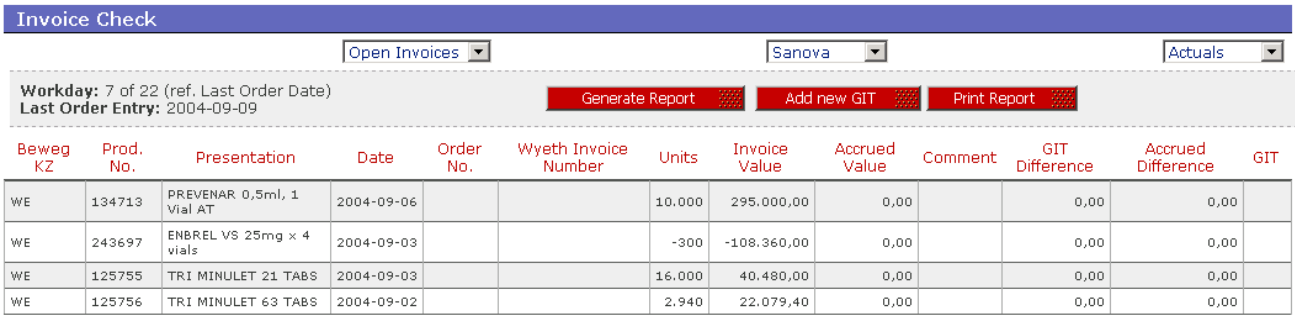
8.2.1 Logistics Closing

1. The Logistics closing will freeze the data of the Invoice Check. If needed, this closing could be rolled back.
2. A finance manager looks through the data and if correct approves the values. After this approval the Logistics closing could not be rolled back, but the approval itself can be retracted and then the logistics closing could again be retracted and data changed. As soon as Logistics closed the month, Finance can work on the financial part of ALF and prepare their month closing.

8.2.2 Finance Closing and JDE Export

3. After approval from Finance, JDE export can be done; i.e., prepare and check the JDE accounting records for export.
4. If the export was correct and imported into the JDE system the JDE export final approval will lock the processing month and no rollback will be possible any more!

8.3 Invoice Check

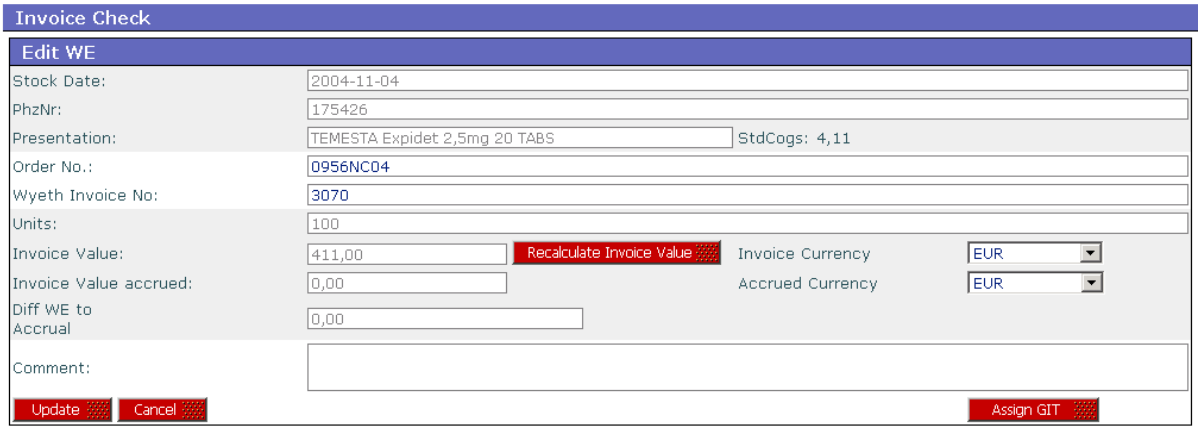


Beweg KZ	Prod. No.	Presentation	Date	Order No.	Wyeth Invoice Number	Units	Invoice Value	Accrued Value	Comment	GIT Difference	Accrued Difference	GIT
WE	134713	PREVENAR 0,5ml, 1 Vial AT	2004-09-06			10.000	295.000,00	0,00		0,00	0,00	
WE	243697	ENBREL VS 25mg x 4 vials	2004-09-03			-300	-108.360,00	0,00		0,00	0,00	
WE	125755	TRI MINULET 21 TABS	2004-09-03			16.000	40.480,00	0,00		0,00	0,00	
WE	125756	TRI MINULET 63 TABS	2004-09-02			2.940	22.079,40	0,00		0,00	0,00	

The aim of the invoice check is to map the invoice of a manufacturer for a specific delivery of products to the WE ('Wareneingang' – receipt of goods) stock information received from the distributor.

In order to map the information, all WE records of the stock which have not yet been processed are shown on the first page. For every record shown there should be an invoice of a manufacturer.

When the Wyeth materials management officer gets the invoice he/she should open the invoice check and click on the respective WE record to insert the details of the invoice, such as:



Wyeth Invoice No.: Wyeth number of the invoice.

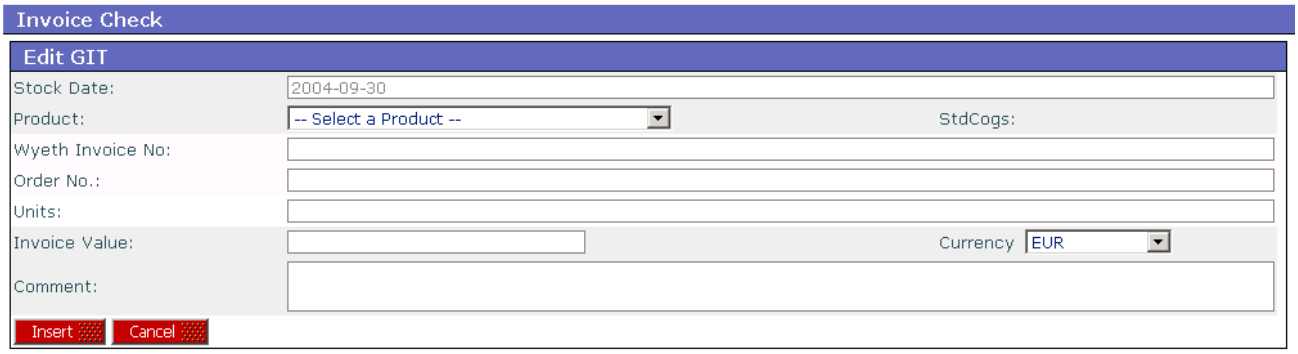
Order No.: Number of the product order (there could also be more than one order number for an invoice).

Invoice Currency: Currency of the invoice.

Accrued Currency: Currency of accruement (if some accruement will be made).

Comment: Field for comments on the invoice, e.g. wrong price on invoice of manufacturer; or counter booking of Sanova, etc.

8.3.1 GIT – Goods in Transit



Since it is possible that an invoice arrives before the goods have arrived, a GIT ("Goods in Transit") record can be made to insert the details of the prematurely arrived invoice. If then the goods arrive in the warehouse, the GIT record can simply be assigned to the WE record with the button 'Assign GIT' on the WE record detail.

8.4 Logistics Month End

8.4.1 Introduction

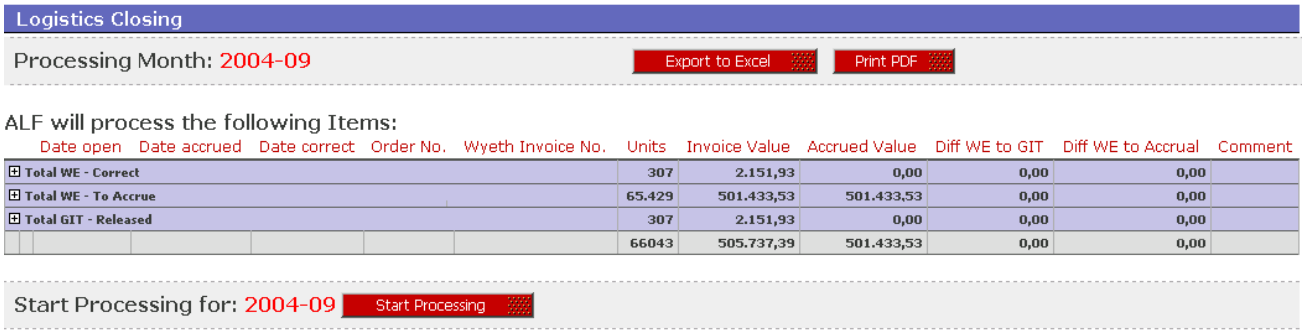
The Logistics Month End Processing is based on the data from the Invoice Check. For the Logistics department it is important to report the invoices (and their progress) to the accounting department in order for them to book the accounts properly.
Therefore, product delivery and its invoicing is the main topic for Logistics.

When a month is being prepared for closing, the different states of the product deliveries and their invoices will lead to different accounting records.

The following states exist:

WE - Correct:	Product deliveries of the processing month whose invoices arrived in time for the processing month.
WE - To Accrue:	Product deliveries of the processing month whose invoices **did not** arrive in time. The value of the invoice needs to be accrued.
WE - Still Accrued:	Old accruals which could be released now.
WE - Released:	Old accruements, which have to be released (Product deliveries of a previous month, which invoices **did** arrive).
GIT - Released:	Goods in Transit which could correctly be assigned to an invoice.
GIT - Open:	Goods in Transit which where created **in the processing month** but not released.
GIT - Still Open:	Goods in Transit which where created **in a previous month** but not released in the processing month.

8.4.2 Logistics Closing



The Logistics Closing shows the states of the product deliveries for the Logistics as well as the Finance departments. It highlights which accounting records will be required in order to be booked for the stock.

These accounting records are not exported; the Accounting department books them manually.

After clicking on the 'Start Processing' button the data will be locked and no more change in the invoice check for the processed month is possible.

8.4.3 Logistics Rollback

With the Logistics Rollback menu the user has the possibility to recreate the invoice data as it was before the Logistics closing took place.

For all "WE – Correct" the correct flag will be removed, all accruals will be released etc., for the processing month only.

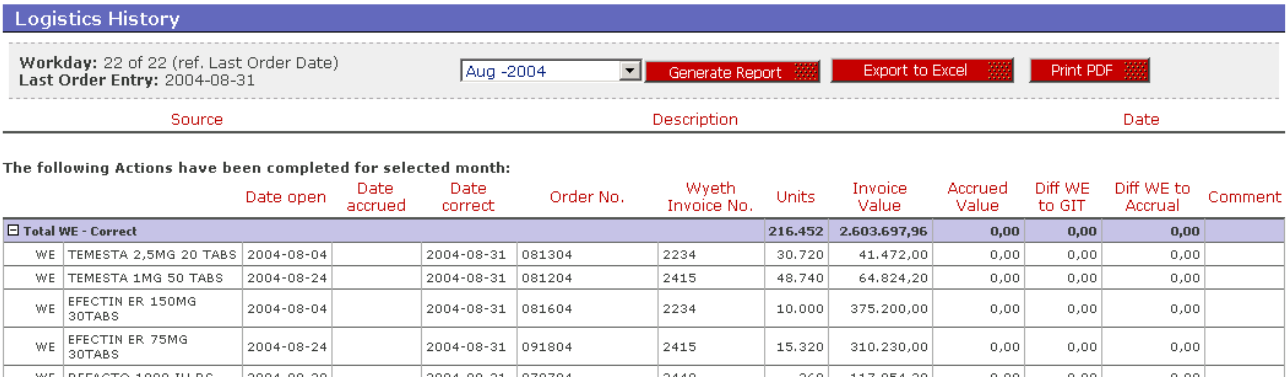
Also the mapping of all assigned GITs to their product deliveries will be removed.

A Logistics rollback will only be possible as long as there is no finance approval or JDE export accomplished.

The following statements indicate if the rollback is possible or not:

- ALF will roll back the following items for: 2004-08
- A rollback for the month: 2004-08 is not possible because it has already been locked by Finance.

8.4.4 Logistics History



		Date open	Date accrued	Date correct	Order No.	Wyeth Invoice No.	Units	Invoice Value	Accrued Value	Diff WE to GIT	Diff WE to Accrual	Comment
Total WE - Correct							**216.452**	**2.603.697,96**	**0,00**	**0,00**	**0,00**	
WE	TEMESTA 2,5MG 20 TABS	2004-08-04		2004-08-31	081304	2234	30.720	41.472,00	0,00	0,00	0,00	
WE	TEMESTA 1MG 50 TABS	2004-08-24		2004-08-31	081204	2415	48.740	64.824,20	0,00	0,00	0,00	
WE	EFECTIN ER 150MG 30TABS	2004-08-04		2004-08-31	081604	2234	10.000	375.200,00	0,00	0,00	0,00	
WE	EFECTIN ER 75MG 30TABS	2004-08-24		2004-08-31	091804	2415	15.320	310.230,00	0,00	0,00	0,00	

The Logistics History is a report to trace the previous, historic Logistics closings.

Selecting a month and generating the report, will show the previous states of the product deliveries.

8.4.5 Finance Approval for Logistics

After the Logistics closing the Finance Approval for Logistics gives the opportunity to lock the data of the Logistics closing, so that it is impossible to roll back the Logistics closing.

If there are wrong values detected in the data, the finance approver can roll back the finance approval, giving the possibility of rolling back the Logistics closing, too.

After the JDE final approval the rollback of the finance approval is not possible either.

The following statements indicate if the processing or rollback is possible or not:

- Final Month End Closing (month will be locked): 2004-09 [Confirm]
- The Month: 2004-08 is already approved by Finance. [Rollback Finance approval]
- A Rollback is not possible because the final JDE approval is already set!

8.5 Finance Month End

8.5.1 Introduction

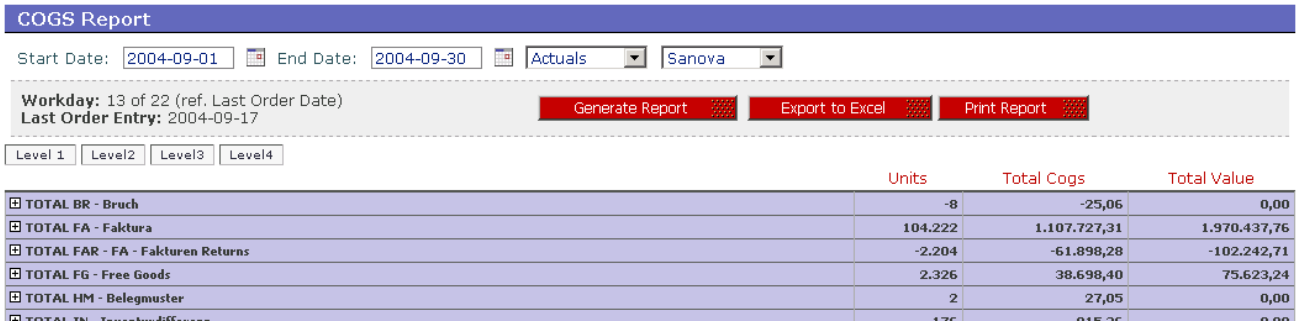
The Finance Month End Processing consists of several checking and information reports, the JDE exporting and JDE approving.

The checking reports will show that all data relevant for the month end processing is consistent and correct.

The information reports will display sales and stock data in a different view, which is relevant for the finance management.

The JDE export gives the possibility to export the accounting records created in the Admin section (see "Admin Accounting Records"). After having been correctly exported and processed by JDE, data export from JDE can be finally approved.

8.5.2 COGS Report



The COGS Report displays all product movements such as sales and stock movements by its transactional code for the selected time period, which is normally the current processing month.

The report was created to easily track the goods in their different movements and to provide means to calculate the resulting Cost of Goods.

8.5.2.1 Structure

Report Layout:

BewegKZ:	The transactional code of the movement (e.g. Faktura = invoice).
Product Group:	Name of the Product Group.
CC - Prod:	Cost Center Number and Product Name.
Product No.:	Product number (Phznr = Pharmazentralnummer).
Presentation:	Product presentation.
Units:	Units of goods.
Total Cogs:	Total Cogs of movement (units multiplied with std cogs per unit).
Total Value:	Value of the movement (only sales movements have a value).

8.5.2.2 Construction

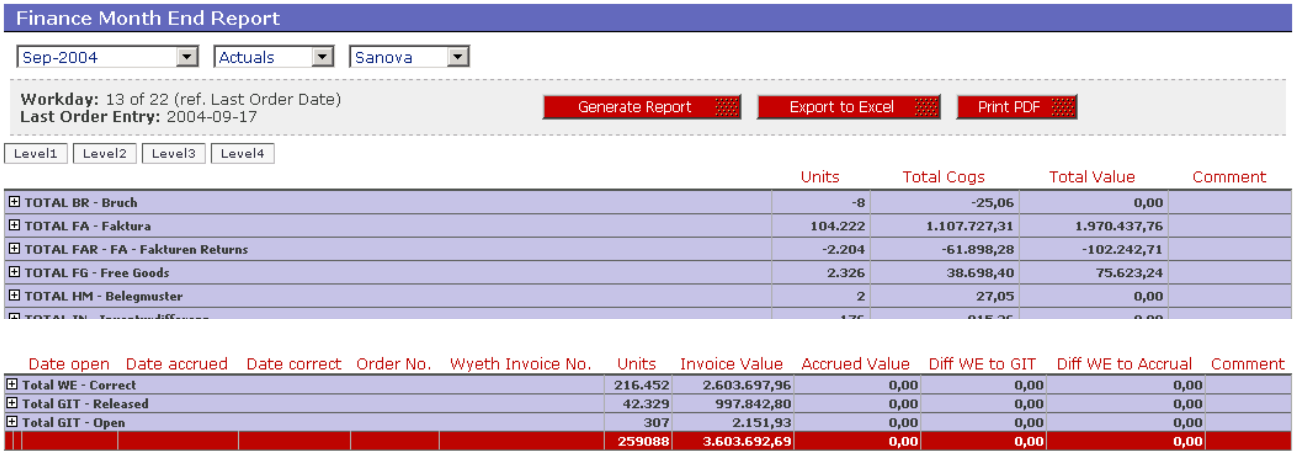
The stored procedure **`PKG_Stock.GetStockCOGS`** is started with the following parameters:

- `StartDate`
- `EndDate`
- `line_id` (actuals, samples, …)
- `dist_id` (Sanova)

The procedure selects columns from:

- MV_STOCK_COGS
  - ↳ T_CODE
  - ↳ MV_SALES
    - ↳ T_CUSTOMER
    - ↳ T_CUSTOMER_GROUP
    - ↳ T_CUST_DISTRIB
    - ↳ T_ORDER_HEAD
    - ↳ T_TRANSMISSION
    - ↳ T_ORDER_POSITION
    - ↳ T_PRODUCT
    - ↳ T_PRODUCT_GROUP
    - ↳ T_LINE
    - ↳ T_COUNTRY
    - ↳ T_CURRENCY
    - ↳ T_TARGET_PG
    - ↳ T_CODE
    - ↳ T_COGS_HISTORY
  - ↳ V_STOCK
    - ↳ T_STOCK
    - ↳ T_PRODUCT
    - ↳ T_TRANSMISSION
    - ↳ T_PRODUCT_GROUP
    - ↳ T_CODE
    - ↳ T_LINE
    - ↳ T_CUST_DISTRIB
    - ↳ T_CUSTOMER

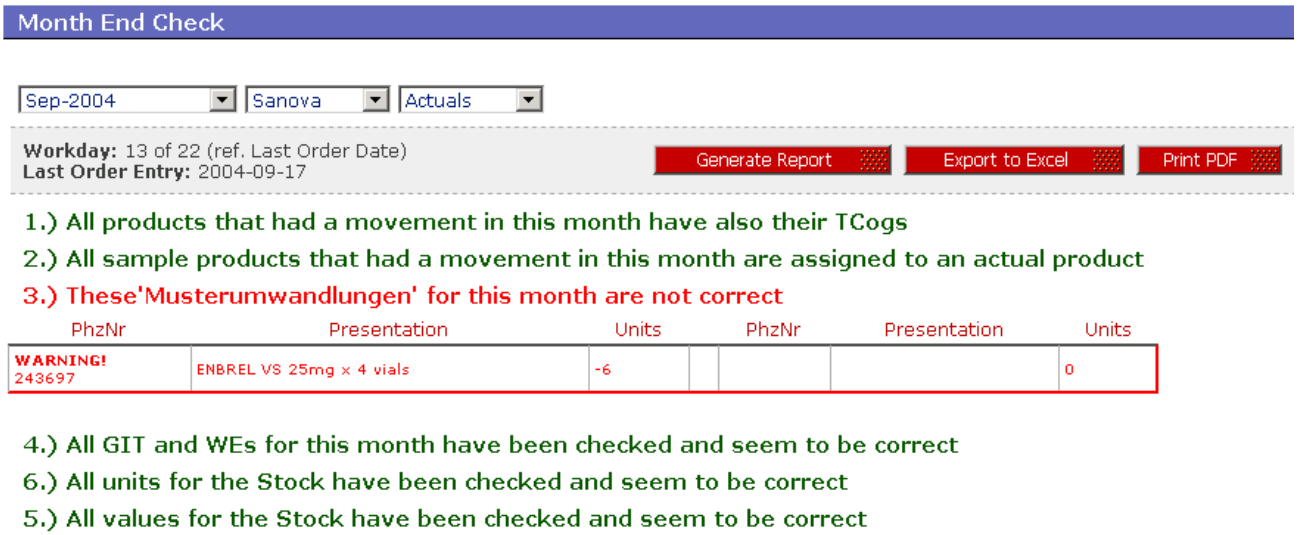
8.5.3 Finance Month End Report



The Finance Month End Report is the COGS Report in combination with the Logistics Closing bookings for a selected month.

This report gives an easy-to-use overview of the accounting values of an already processed or currently processed month.

8.5.4 Month End Check



The Month End Check is a combination of all relevant checking reports for the processing month in order to provide the finance management with a comfortable possibility to check all finance month-end relevant data at one glance.

If a check is correct, only a green conclusion line will be displayed; if a check is not correct, a red warning line will be shown.

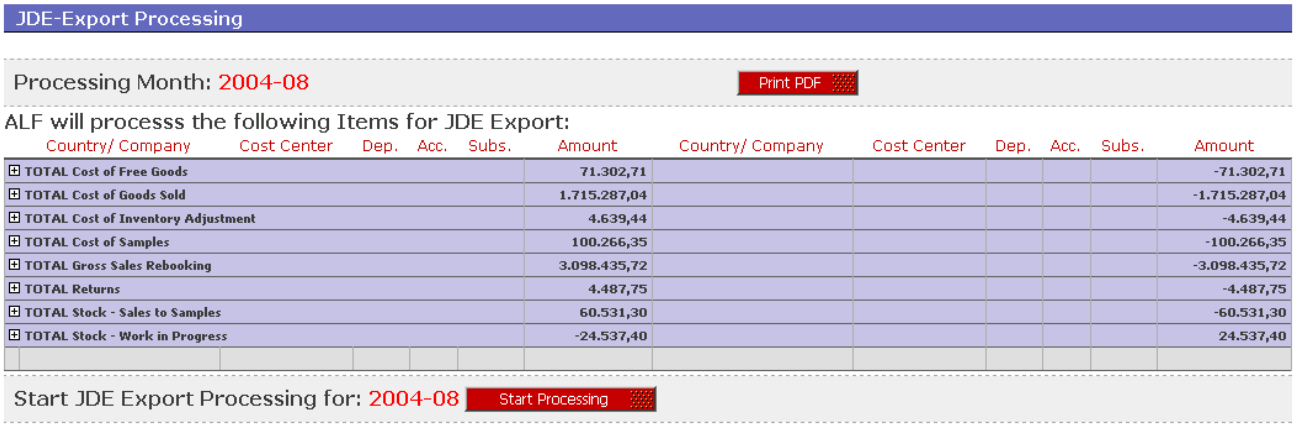
The following reports are displayed in compressed format. Only if there is wrong data, details will be shown:

- TCOGS Check
- Sample Product Check
- UM Cross Check
- TCOGS Check for Processing
- Stock Statistics for units and values

The following stored procedures are involved to gather the information

- **PKG_Stock.GetCrossCheckUM**
- **PKG_Application.GetProductsWithNoTCOGSforMonth**
- **PKG_Stock.GetStockStatValues**
- **PKG_Stock.GetStockStat**
- **PKG_Application.GetSamWithNoActual**
- **PKG_Month_End.CheckTCOGSforProcessing**

8.5.5 JDE-Export Processing

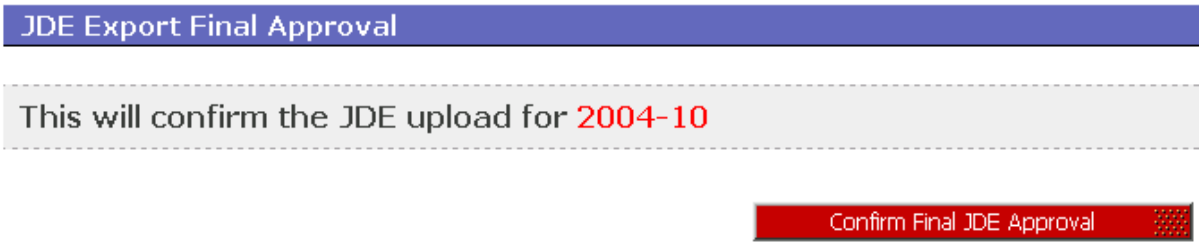


The JDE Processing displays the accounting records (rom Admin --> Accounting Records) with the data from the finance month-end report, matched with the assigned codes of the accounting records for the current, already finance approved processing month.

By clicking 'Start Processing' a pop-up window will appear asking if the user is sure to process the displayed month. After clicking [OK] the JDE file for processing will be downloaded to the user and the displayed month will be marked as JDE processed.

The next step is to import the data into the JDE system, and if everything is correct the JDE Export Final Approval has to be done.

8.5.6 JDE Export Final Approval

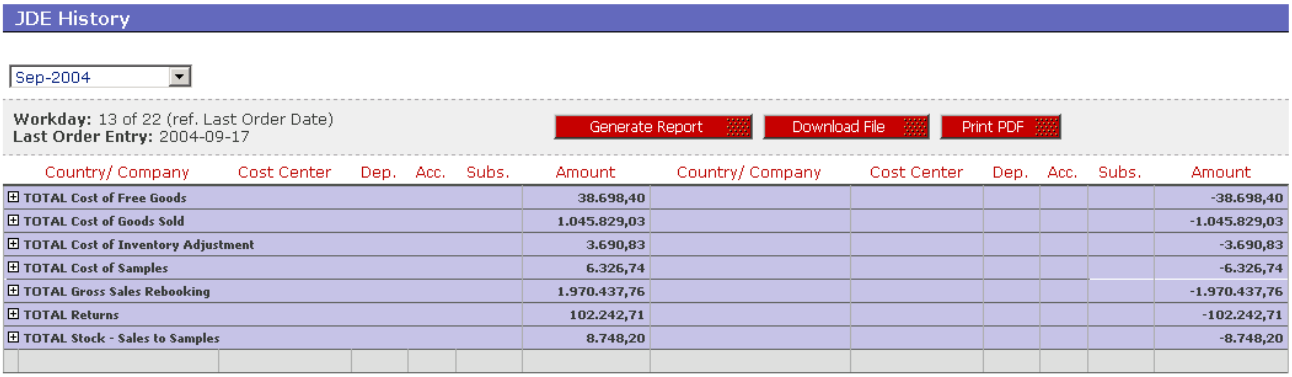


The final step in the month end processing is to approve that the JDE exported data was correctly uploaded into the JDE system and also correctly processed.

After the final approval has been done no rollback will possible any more, the data is ultimately locked and no changes can take place.

If it appears that the JDE Export file is needed again, the file is available at any time in the JDE History report (see next section).

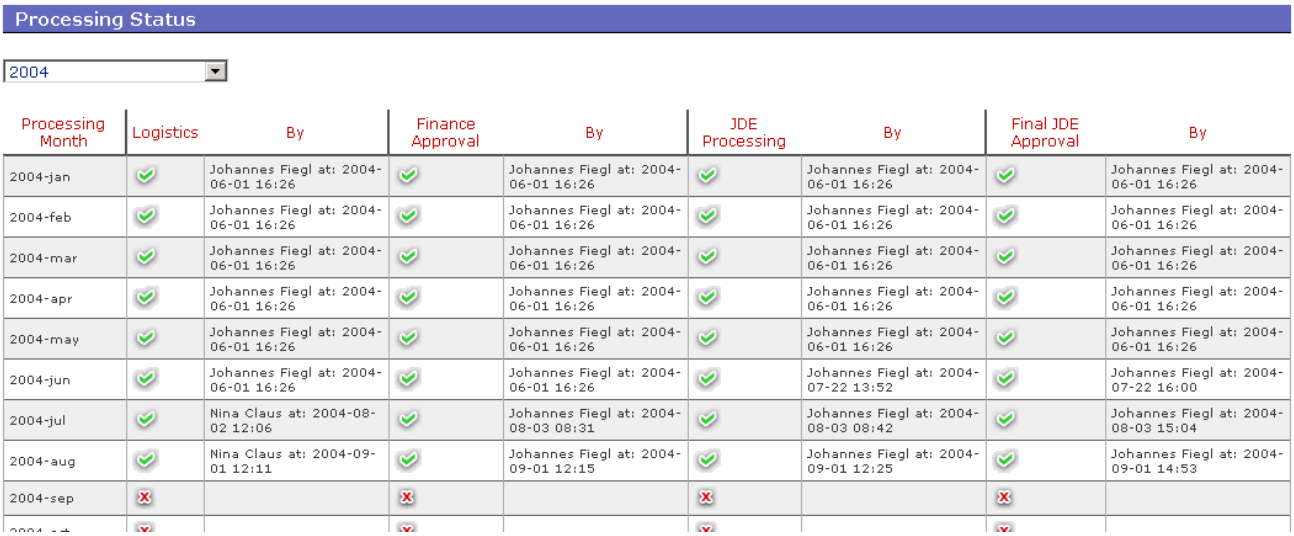
8.5.7 JDE History



The JDE History is a report to trace the history of the JDE exports.

Selecting a month and generating the report will display which information was exported to JDE in which month.

8.6 Processing Status



Processing Status

2004

Processing Month	Logistics	By	Finance Approval	By	JDE Processing	By	Final JDE Approval	By
2004-jan	✓	Johannes Fiegl at: 2004-06-01 16:26	✓	Johannes Fiegl at: 2004-06-01 16:26	✓	Johannes Fiegl at: 2004-06-01 16:26	✓	Johannes Fiegl at: 2004-06-01 16:26
2004-feb	✓	Johannes Fiegl at: 2004-06-01 16:26	✓	Johannes Fiegl at: 2004-06-01 16:26	✓	Johannes Fiegl at: 2004-06-01 16:26	✓	Johannes Fiegl at: 2004-06-01 16:26
2004-mar	✓	Johannes Fiegl at: 2004-06-01 16:26	✓	Johannes Fiegl at: 2004-06-01 16:26	✓	Johannes Fiegl at: 2004-06-01 16:26	✓	Johannes Fiegl at: 2004-06-01 16:26
2004-apr	✓	Johannes Fiegl at: 2004-06-01 16:26	✓	Johannes Fiegl at: 2004-06-01 16:26	✓	Johannes Fiegl at: 2004-06-01 16:26	✓	Johannes Fiegl at: 2004-06-01 16:26
2004-may	✓	Johannes Fiegl at: 2004-06-01 16:26	✓	Johannes Fiegl at: 2004-06-01 16:26	✓	Johannes Fiegl at: 2004-06-01 16:26	✓	Johannes Fiegl at: 2004-06-01 16:26
2004-jun	✓	Johannes Fiegl at: 2004-06-01 16:26	✓	Johannes Fiegl at: 2004-06-01 16:26	✓	Johannes Fiegl at: 2004-07-22 13:52	✓	Johannes Fiegl at: 2004-07-22 16:00
2004-jul	✓	Nina Claus at: 2004-08-02 12:06	✓	Johannes Fiegl at: 2004-08-03 08:31	✓	Johannes Fiegl at: 2004-08-03 08:42	✓	Johannes Fiegl at: 2004-08-03 15:04
2004-aug	✓	Nina Claus at: 2004-09-01 12:11	✓	Johannes Fiegl at: 2004-09-01 12:15	✓	Johannes Fiegl at: 2004-09-01 12:25	✓	Johannes Fiegl at: 2004-09-01 14:53
2004-sep	✗		✗		✗		✗	

The Processing Status report gives a comprehensive overview of the processing and approval statuses of all months, including the current processing month.

It provides the finance manager with a clear overview of the processing progress with information of the executing user and time.

The green tick indicates that a correct closing/approving/processing took place.

The red tick indicates that no action has taken place yet.

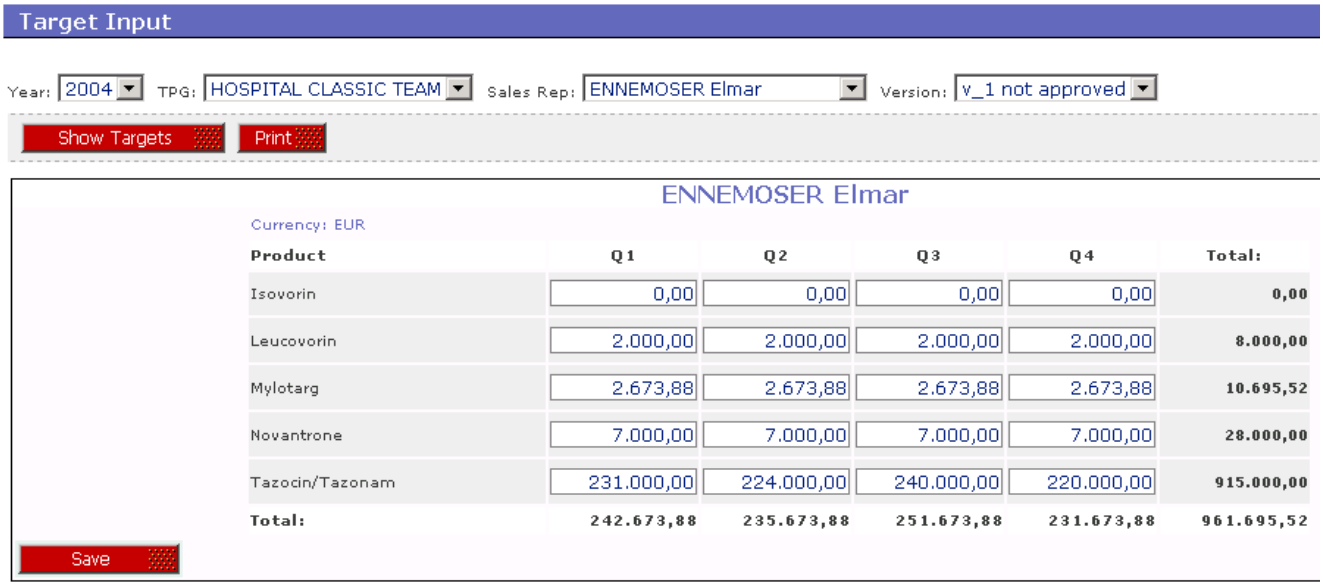
9 Targeting

9.1 Introduction

The Targeting is a mechanism to set specified sales targets for sales reps. Sales targets are defined as a set of four quarterly sales values per identified target product per year and per sales rep. A supervisor must approve each set of sales targets. Reports will display the percentage of target achievement. According to these achievement values (if sales reps reached their goals or not) the sales reps will get their bonuses. Bonus calculation is not done in ALF but externally in the Sales department.

Through the targeting interfaces the targets will be set and approved by a user with the "target approved"). The Target Statistics report the achievement values; i.e. compare budget vs. actuals.

9.2 Target Input



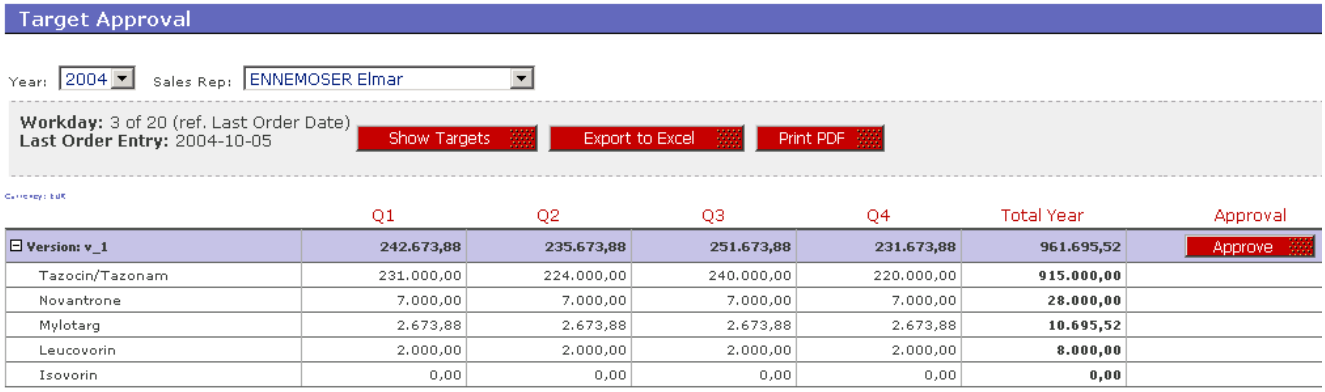
The Target Input is structured in three steps:

1. The user needs to choose the Target Product Group (TPG)
2. A sales rep needs to be selected via drop-downt box.
3. An unapproved version needs to be selected or a new target version needs to be created.

The button 'New Target Version' appears between the 'Show Targets' and the 'Print' button if all target versions are approved.

Approved versions cannot be changed any more; the button save will not be shown on the screen.

9.3 Target Approval

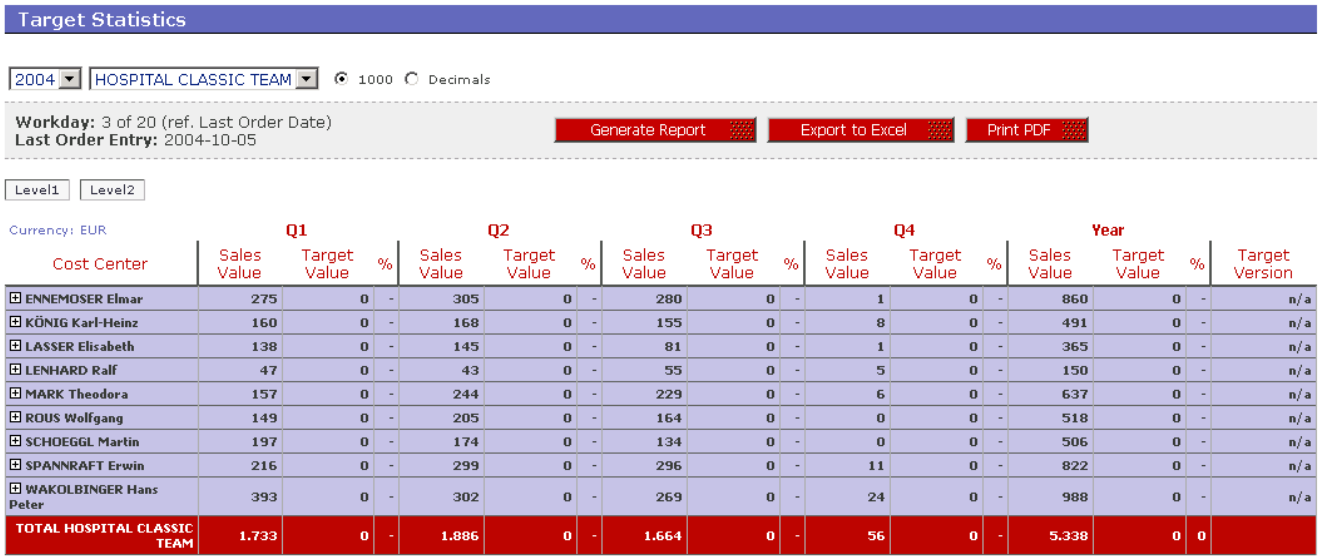


	Q1	Q2	Q3	Q4	Total Year	Approval
Version: v_1	242.673,88	235.673,88	251.673,88	231.673,88	961.695,52	Approve
Tazocin/Tazonam	231.000,00	224.000,00	240.000,00	220.000,00	**915.000,00**	
Novantrone	7.000,00	7.000,00	7.000,00	7.000,00	**28.000,00**	
Mylotarg	2.673,88	2.673,88	2.673,88	2.673,88	**10.695,52**	
Leucovorin	2.000,00	2.000,00	2.000,00	2.000,00	**8.000,00**	
Isovorin	0,00	0,00	0,00	0,00	**0,00**	

In the Target Approval interface all already approved targets of a sales rep will be shown and the details of the approval (who and when).

If a target is not approved, the button 'Approve' will appear in the right-most column. After clicking a pop-up will confirm the approval of the target version will be maked and displayed in the reports.

9.4 Target Statistics



Cost Center	Q1 Sales Value	Q1 Target Value	Q1 %	Q2 Sales Value	Q2 Target Value	Q2 %	Q3 Sales Value	Q3 Target Value	Q3 %	Q4 Sales Value	Q4 Target Value	Q4 %	Year Sales Value	Year Target Value	Year %	Target Version
ENNEMOSER Elmar	275	0	-	305	0	-	280	0	-	1	0	-	860	0	-	n/a
KÖNIG Karl-Heinz	160	0	-	168	0	-	155	0	-	8	0	-	491	0	-	n/a
LASSER Elisabeth	138	0	-	145	0	-	81	0	-	1	0	-	365	0	-	n/a
LENHARD Ralf	47	0	-	43	0	-	55	0	-	5	0	-	150	0	-	n/a
MARK Theodora	157	0	-	244	0	-	229	0	-	6	0	-	637	0	-	n/a
ROUS Wolfgang	149	0	-	205	0	-	164	0	-	0	0	-	518	0	-	n/a
SCHOEGGL Martin	197	0	-	174	0	-	134	0	-	0	0	-	506	0	-	n/a
SPANNRAFT Erwin	216	0	-	299	0	-	296	0	-	11	0	-	822	0	-	n/a
WAKOLBINGER Hans Peter	393	0	-	302	0	-	269	0	-	24	0	-	988	0	-	n/a
TOTAL HOSPITAL CLASSIC TEAM	1.733	0	-	1.886	0	-	1.664	0	-	56	0	-	5.338	0	0	

The Target Statistics displays the proportional sales of the sales reps and calculates the percentage of target achievement according to the target values. A percentage of 100% indicates that the sales exactly reached target.

As this report shows the sales of the sales rep for the **whole year** it takes longer to be calculated as other reports.

10 ALF - Service

10.1 Introduction

The ALF Service is a Windows 2000 service running on the LIVE (intranet) server, which manages the data download from the Sanova FTP server, the SQL-Loader upload into the database and the starting of the importing stored procedures.

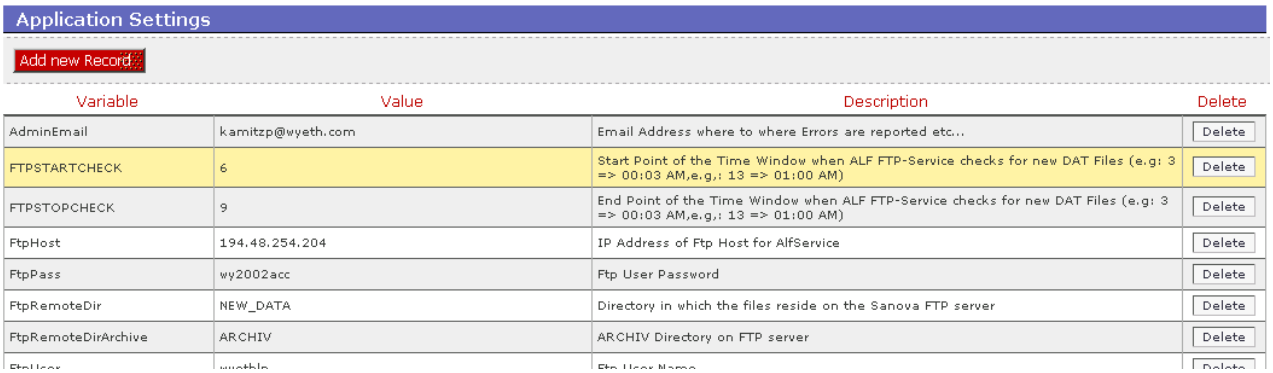
10.2 Stored Procedure

The stored procedure which is called up by the ALF Service is: **`PKG_IMPORT_SANOVA.P_Import`** and will execute the following steps in the following order:

1. Customer import
2. Product import
3. Transaction data (BW) import
4. Refreshing of the materialized views
5. Performing the Stock Check all unchecked transmissions

10.3 Parameterization

The parameters of the service are stored in the database and could be managed in the ALF-System with the following interface (Admin --> Application Settings):



Please be sure to understand any settings before changing them. Uncontrolled changes will render the ALF System and/or its data working inconsistently or invalid!

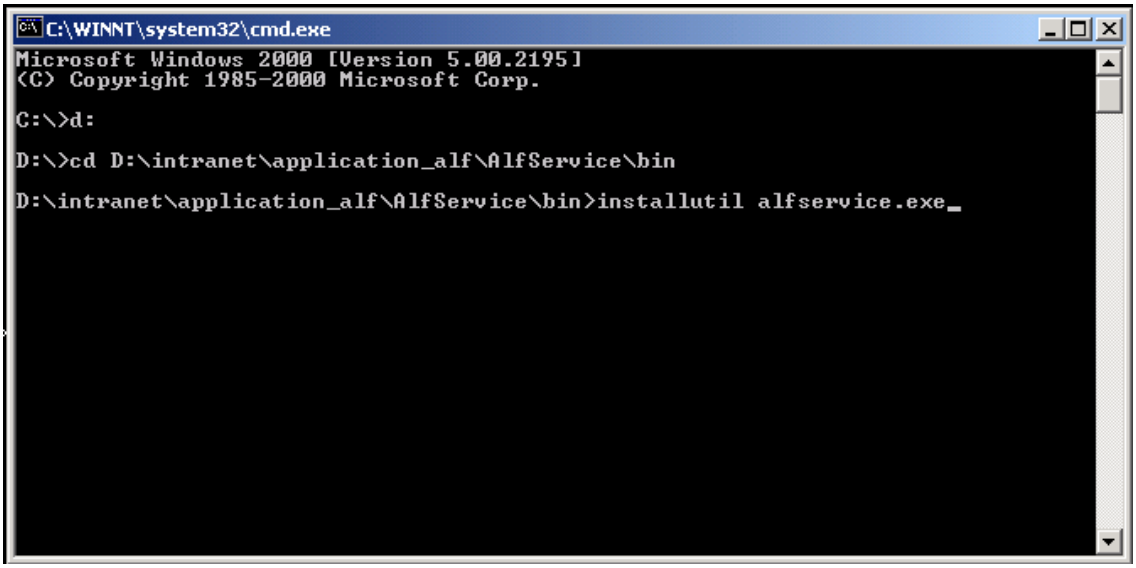
10.4 Running on which Server?

The behavior of the service is different when running on the live server or on the development server! If the service is running on the live server it imports the new data to the live AND development server and starts the import routines!
If the service is running on the development server it imports the new data **only to the development** server! The cause for this behaviour is clear: when running live, the development server should have congruently matching data; if running on development there is (allegedly) no live server available or it is in testing mode.

10.5 Install / Uninstall

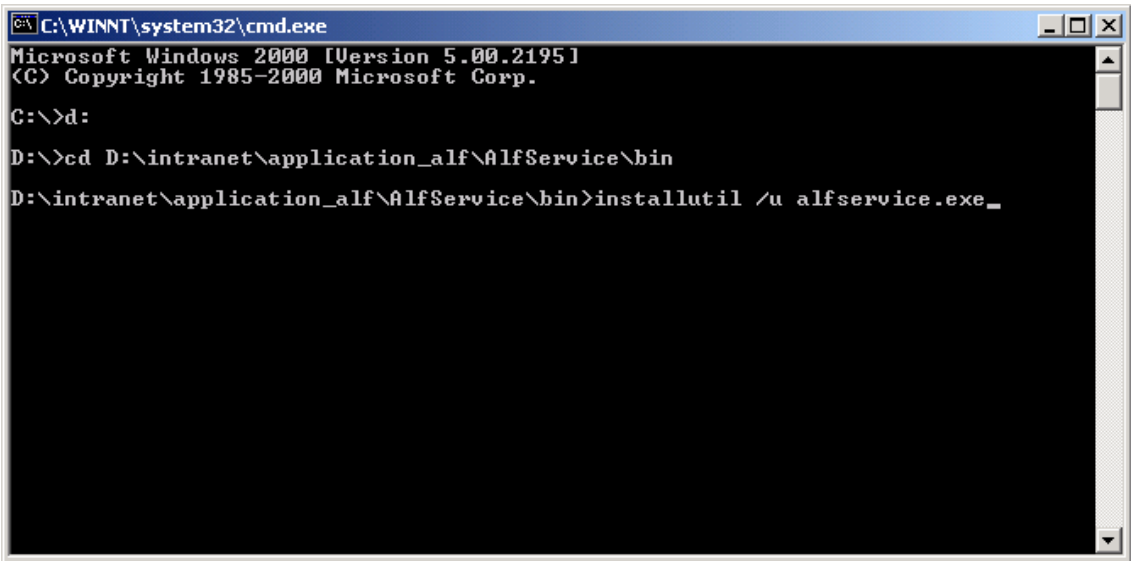
10.5.1 Install:

- Open a Command Window (on the workstation or server where the service should be installed)
- Change to the \bin Folder of the Alfservice Project
- Install Alfservice by entering: ***[ALF-Bin]:\ installutil alfservice.exe***



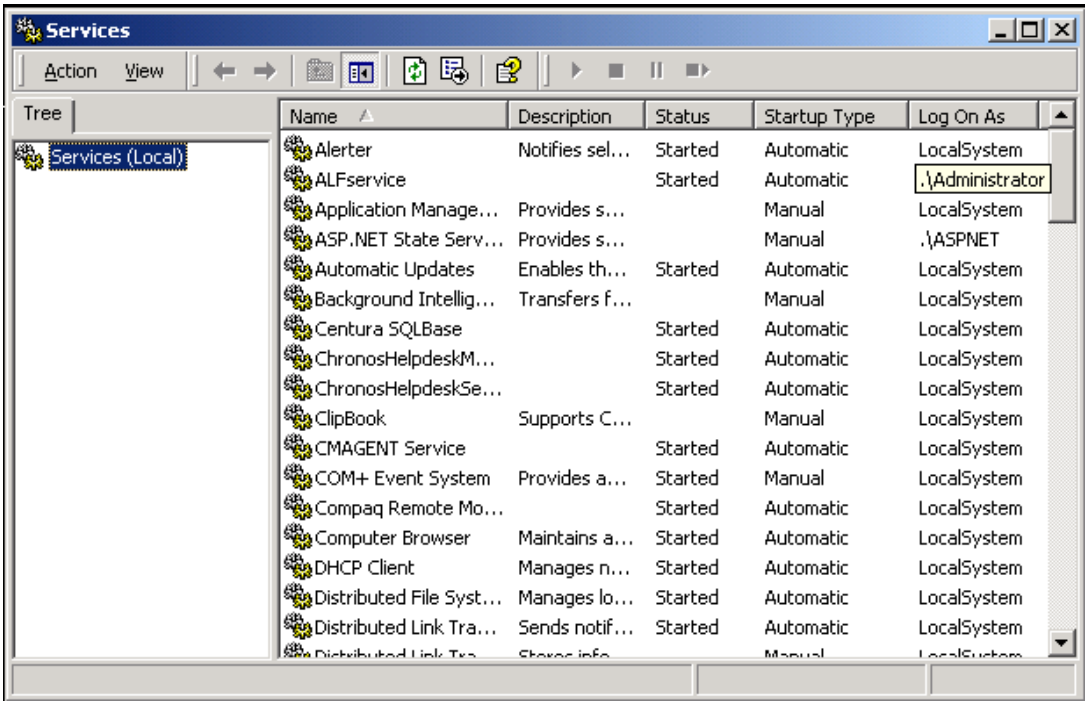
10.5.2 Uninstall:

- Open a Command Window
- Change to the bin Folder of the Alf Service Project
- Install Alf Service by entering: ***[ALF-Bin]:\ installutil /u alfservice.exe***

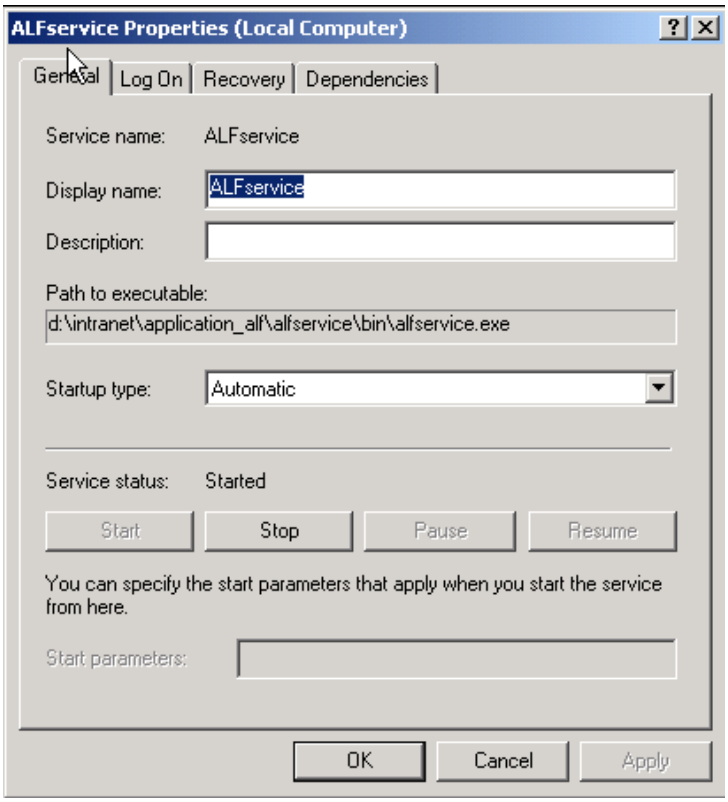


10.6 Managing the Service

The service starts per default with the server system start up. This start up automatic can be changed in: **ALFservice in Control Panel -> Administrative Tools -> Services**

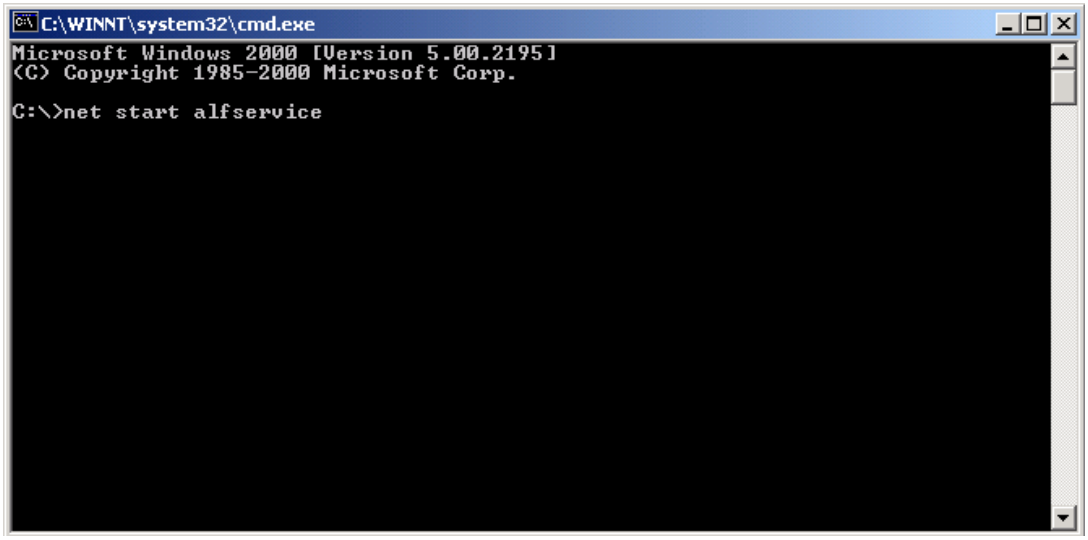


Other details can be selected there:



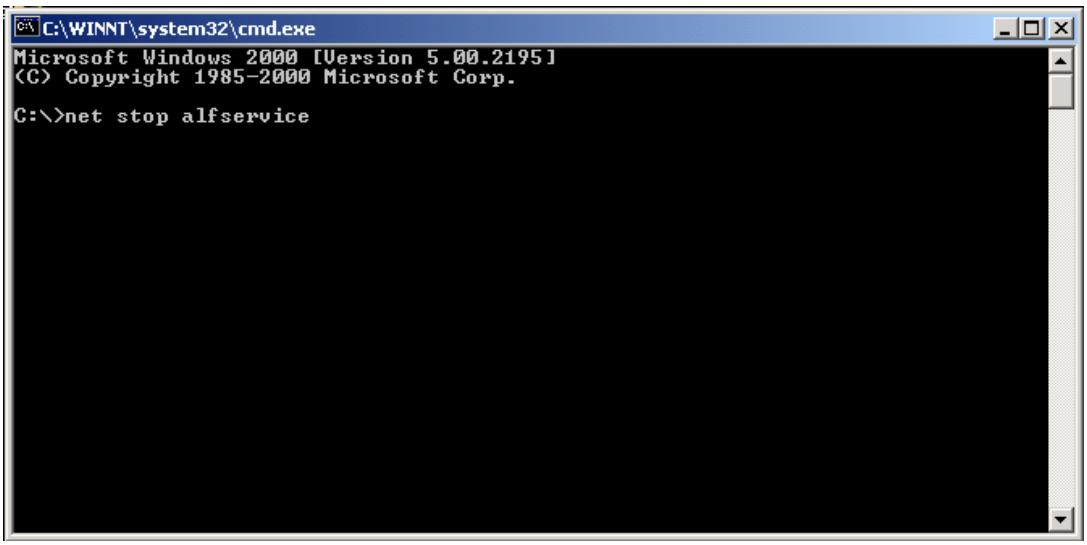
10.6.1 Alternative Service Start

The service can alternatively also be started in the command box by entering ***c:\ net start alfservice***.



10.6.2 Alternative Service Stop

The service can also be stopped in the command box by entering ***c:\ net stop alfservice***.



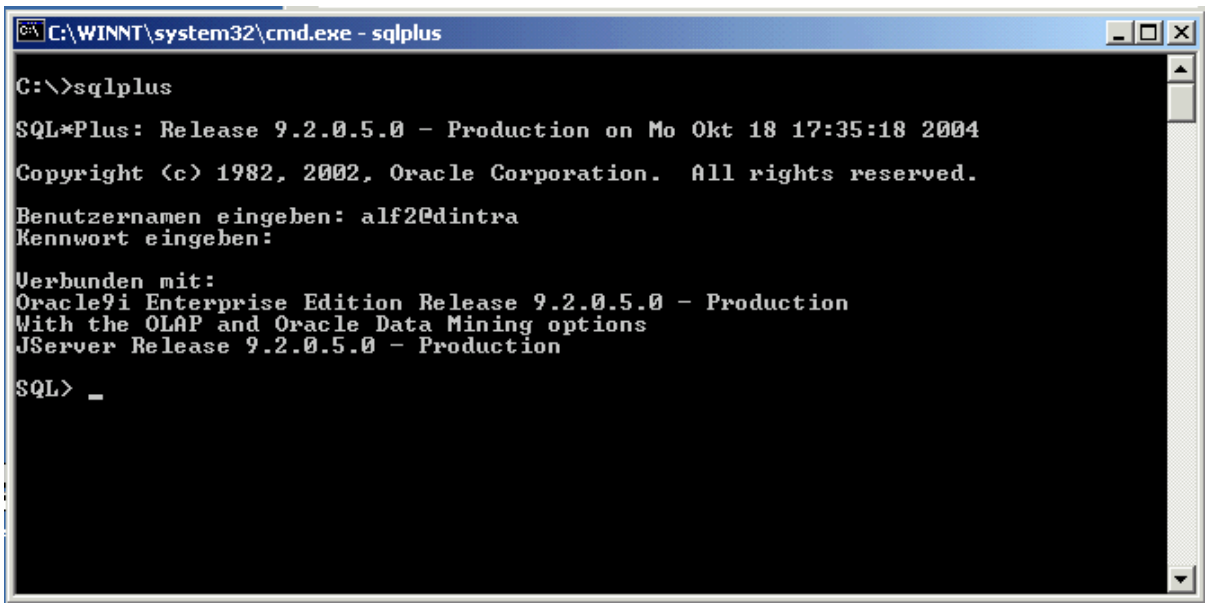
10.7 After a Server Update

It is very important to know that **after a server update the ALF Service needs to be uninstalled and reinstalled again to run correctly!!**

11 Database Internals

11.1 Connection to the Database

11.1.1 Oracle Version



Both the Live and the Development Server are running Oracle 9.2.0.5.0.

11.1.2 Oracle Client

In order to be able to connect from a local workstation to the database, an Oracle client needs to be installed.

As the server version is 9.2.0 the Oracle Client should also be a 9.2 version.

11.1.3 TNS-Names Entry

If an Oracle Client is installed on the workstation the following TNS-Names entries need do be added to the tnsnames.ora file, which is located in the \network\ADMIN folder of the oracle client directory. (e.g. C:\orawin924\network\ADMIN (when having installed the 9.2.4 Oracle Client)).

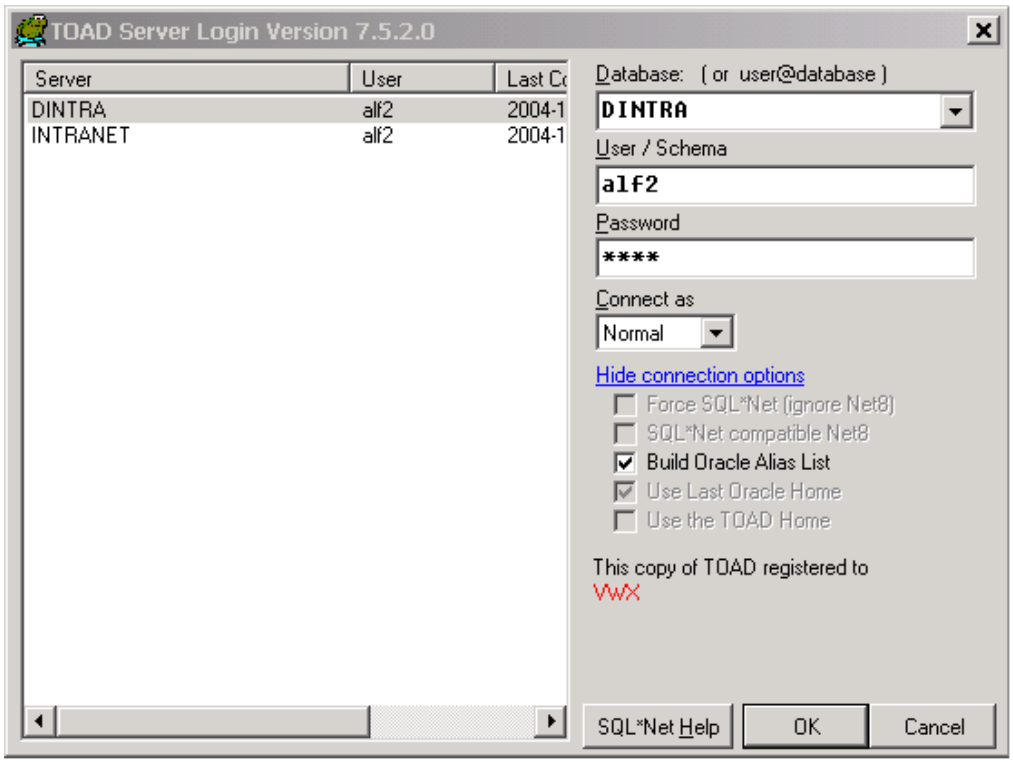
Entries to add:

```
INTRANET =
  (DESCRIPTION =
    (ADDRESS_LIST =
      (SOURCE_ROUTE = OFF)
      (LOAD_BALANCE = OFF)
      (FAILOVER = ON)
      (ADDRESS = (PROTOCOL = TCP)(HOST = 10.248.57.222)(PORT = 1521))
      (ADDRESS = (PROTOCOL = TCP)(HOST = 10.248.57.223)(PORT = 1521))
    )
    (CONNECT_DATA =
      (SERVICE NAME = INTRANET.VIENNA)
    )
  )

DINTRA =
  (DESCRIPTION =
   (ADDRESS_LIST =
     (ADDRESS = (PROTOCOL = TCP)(HOST = 10.248.57.232)(PORT = 1521))
   )
   (CONNECT_DATA =
     (SERVICE_NAME = DINTRA.VIENNA)
   )
 )
```

As shown, the live database server is listed with two addresses, one is the live server IP address and the second is the IP address of the stand-by server which is running synchronous to the live server.

11.1.4 TOAD



The ALF-System was developed with the TOAD tools, which is a comfortable development environment for PL/SQL, the programming language of the stored procedures in the Oracle database.

To login the following parameters need to be set:

Database:	DINTRA or INTRANET (according to the tnsnames.ora)
User / Schema:	alf2
Password:	alf2

The TOAD tool provides the user with a comfortable view of all database objects, such as:

- Tables
- Views
- Stored procedures
- Triggers
- Snapshots (materialized views)
- Indexes
- Constraints
- Sequences
- DBlinks
- Jobs

There are even more database object categories selectable, but those are not used in the ALF-System.

11.2 Database Objects

11.2.1 Introduction

This chapter will explain the database objects of the ALF-System more thoroughly. It will describe the objects and their aims in detail.

11.2.2 Tables

All tables are described in detail in the data model chapter.

11.2.3 Views and Materialized Views

Due to the constraints of data selecting statements and in order to present the data of the tables in different aspects, many views were built.

Delivering the desired information in acceptable time was one of the main criteria of ALF, therefore materialized views (views which are stored as tables and due to this are instantly available) are used extensively to gain performance.
The materialized views represent the highest form of aggregated data in the system and therefore is build on complex statements using many views which gather their data from other views and view/queries, so ALF uses multi-tier cascaded statements.

11.2.4 Stored Procedures

Using the ER-Model default stored procedures for inserting, updating and deleting datasets in each table, three stored procedures are mainly used: one for inserting, one for updating and one for deleting a single set in the table.

The naming convention for these procedures adhere to the following rules:

Prefix

P_	To identify the object as a stored procedure.

Table

TABLENAME	The name of the table to know which table the procedure is affecting.

Postfix

_INS	For the inserting procedure.
_UPD	For the updating procedure.
_DEL	For the deleting procedure.

11.2.5 Packages

Packages are bundles of stored procedures and functions which are designed for specific tasks. For example: all procedures which are relevant for calculating the non-fixed holidays of a year, are packed up in the PKG_Calendar. The Prefix PKG_ identifies the object as a package.

The following describes all packages in the database in more detail, ordered alphabetically.

11.2.5.1 PKG_APPLICATION

The application package holds all procedures and functions which provide the application with relevant data such as procedures to gather information for the start page of the ALF-System, or functions which calculate important values for different reports.

A list of all procedures and functions in the package:

- `GetNextValidTranDate`
- `GetMaxTranID`
- `GetMaxTransmission`
- `GetLastOrderDate`
- `GetWorkdaysYear`
- `GetWorkdaysYearToDate`
- `GetWorkdaysForMonthToday`
- `GetWorkdaysForMonth`
- `GetMViewsStatus`
- `GetCodeID`
- `GetTCogs`
- `GetLastOfProcessMonth`
- `GetCurrentProcessMonth`
- `GetLastMonthApproved`
- `GetPageTitle`
- `GetMenuModuleID`
- `GetUserName`
- `GetLastHoliDay`
- `IsHoliDay`
- `GetLastProcessedMonth`
- `GetLastFinalJDEMonth`
- `GetFxRate`
- `GetUserForAcessRight`
- `GetApplicationMessages`
- `GetAllApplicationMessages`
- `GetIntranetUsers`
- `GetMenuModuleIds`
- `GetCuSr`
- `GetSalesRepIDs`
- `GetTargetSalesRep`
- `GetNotImportedART`
- `GetNotImportedKD`
- `GetNotImportedBW`
- `GetInvoicer`
- `GetMenuLeftNavi`

- `GetMenuIDParent`
- `GetMenu`
- `GetAppVars`
- `GetLogs`
- `GetCodesByCat`
- `GetCodes`
- `DeleteLogs`
- `GetHoliDays`
- `GetHoliDaysformonth`
- `GetFXRates`
- `GetCurrencies`
- `GetDistributors`
- `GetCountries`
- `GetApplicationSettings`
- `GetSalesReps`
- `GetSalesRepsTAPG`
- `GetTransmissions`
- `GetProductsWithNoTCOGS`
- `GetProductsWithNoTCOGSForMonth`
- `GetSamProductsWithNoActual`
- `GetTCOGSHistoryByProduct`
- `FillLeftNaviMenu`

11.2.5.2 PKG_CALENDAR

In the calendar package are all procedures stored which used to fill the holidays table with correct holiday dates.

Starting Pkg_Calendar.add_holidays_of_a_year with a desired year will fill the table. This stored procedure can only be executed directly in the database!

The routine considers the following holidays:

- Good Friday
- Easter Day
- Easter Monday
- Ascension of Christ
- Whitsunday
- Whitmonday
- Corpus Christi Day
- New Years Day
- Epiphany (Dreikönigstag)
- Staatsfeiertag (Austria)
- Nationalfeiertag (Austria)
- Christmas
- Christ Day (Christtag)
- St. Stephens Day (Stefanitag)
- New Years Eve

To fill the table with weekends the **`Pkg_Calendar.add_weekend_into_t_holiday`** needs to be started. As parameter the year for which the holidays should be added needs to be set.

11.2.5.3 PKG_CUSTOMER

Here all procedures to get customer information are bundled together.

The following procedures are implemented and used in the different master data interfaces:

- `GetCustomerList`
- `GetCustomerGroups`
- `GetCustomerStatType`
- `GetCustomerDist`
- `GetCustomerPLZ`
- `GetCustomerCodes`
- `GetCustomer`
- `GetCustomerByID`
- `GetCustomerByCudiNr`

11.2.5.4 PKG_IMPORT_ERRORS

This package is relevant for error and new entity information extracted from the t_logs table. As there is no specific table for errors, the table t_logs is searched for error entries.

The new entity information is new customers or new products which will be displayed in the start page of the system as they need user maintenance.

The following procedures are implemented:

- `GetNumberOfImportNewCustomers`
- `GetNumberOfImportNewProducts`
- `GetNumberOfImportError`
- `GetNumberOfStockError`
- `GetStockErrorDetails`
- `GetNumberTempTableRecords`
- `GetNumberTempTableSalesRecords`
- `GetNumberTempTableRecordsKD`
- `GetNumberTempTableRecordsART`
- `GetLatestImportNewProducts`
- `GetLatestImportNewCustomers`
- `GetLatestImportError`
- `GetLatestImport`
- `GetLatestStockError`

11.2.5.5 PKG_IMPORT_FORTE

The FORTE import package manages the import of data from the FORTE system which is used in the ALF-System. For each table to which data is imported, there is a function performing the import routine:

- `F_Country`
- `F_Currency`
- `F_FX_Rate`
- `F_Products`
- `F_Budget`
- `F_COGS`
- `F_Product_Group`
- `F_GM_Layout`

For details see chapter Maintenance Functions → FORTE Import.

11.2.5.6 PKG_IMPORT_SANOVA

The Sanova import package manages the import of the data from the Austrian distributor Sanova. As described in the Data Input chapter, there are three types of data from Sanova: Customer, products and transactional data.

For each type of data import, for maintaining the transmission check, the MV refresh and the import there is a function:

- `F_Customers` (for importing and updating customer information)
- `F_Products` (for importing and updating product information)
- `F_BW` (for importing the transactional (sales, inventory & stock) data)
- `F_add_broken_BW` (system function to import data of the transactional data, which could not be imported (e.g. if the customer or the product was missing.))
- `F_Refresh_MVs` (to refresh all MVs in the correct order)
- `P_Check_Transmissions` (performing the stock check for all unchecked transmissions)
- `P_Import` (running all importing function in the correct order and then refreshing the MVs)

11.2.5.7 PKG_JDE

The JDE package is for maintaining the accounting records and exporting the JDE data (accounting data).

The following procedures are therefore used:

- `GetBewegKZsForAcre`
- `GetAllBewegKZs`
- `AddBewegKZsForAcre`
- `DelBewegKZsForAcre`
- `GetAccountingRecord`
- `GetBeweKZList`
- `GetAccountNamesList`
- `DeleteAccountingRecord`

- ModifyAccountingRecord
- UpdateAccountingRecord
- GetAccountingData

11.2.5.8 PKG_MONTH_END

The Month End package holds all procedures, which are needed for the month end processing (see chapter Processing).

The following procedures are used to process or rollback the different levels of approval for the processing month and different checks which need to be done before processing can be started:

- P_Invoice_Processing
- P_Invoice_Rollback
- P_GIT_Rollback
- P_Get_ProcessedMonth
- P_Get_MonthEndHistoryLogs
- P_Set_FinanceApproval
- P_Rollback_FinanceApproval
- P_Set_JDE_Processed
- P_Set_JDE_FinalApproval
- P_Rollback_JDE_FinalApproval
- P_GetV_PM
- P_GetV_PMForYear
- P_SetLogisticsClosingUserID
- F_Check_For_FinanceApproval
- F_Check_For_Jde_finalApproval
- F_Check_For_JDEProcessing
- CheckTGCOGSForProcessing

11.2.5.9 PKG_PRODUCT

In this package all procedures which are needed for maintenance processes and attributes are combined.

A list of all procedures and functions in the package:

- GetProducts
- GetProductByID
- GetProductByPhznr
- GetProductList
- GetProductGroups
- DeleteProduct
- GetLines
- GetCountries
- GetCurrencies
- GetTargetPrGr
- GetNotAssignedSampleProducts
- GetProductDescriptions

11.2.5.10 PKG_REPORTS

This is the most important package as it includes all functions and procedures which are needed for querying and displaying.

For every report or view of a report there is procedure:

- `GetRUII`
- `GetCegedim`
- `GetNetSales`
- `GetDailySales`
- `GetDailySalesTotal`
- `GetRoyalityReportValue`
- `GetRoyalityReportUnit`
- `GetInventoryExctract`
- `GetPremarinReport`
- `GetStockForIris`
- `GetSalesForeCastAccuracy`
- `GetWWSReport`
- `GetWWSFORTEReport`
- `GetSales`
- `GetSalesOrders`
- `GetGMReport`
- `GetGMTotal`
- `GetSalesStatistic`
- `GetCustomerStatistik`
- `GetSalesStatCustomer2`
- `GetSalesAreaStatTAPG`
- `GetSalesAreaStatSMS`
- `GetSalesAreaStatDetail`
- `GetSalesCustomerStatDetail`
- `GetMonthEndProcessedData`
- `GetActandSamProducts`
- `GetSamandActProducts`
- `GetNewEntitiesReportCustomers`
- `GetAMSSalesStat`
- `GetAMSOrderDetails`
- `GetAMSProdKurzBez`
- `GetInvlBatchReport`

11.2.5.11 PKG_STOCK

The stock package manages the stock relevant checks, user maintenances and reports, and holds the following procedures and functions:

- `GetBalance`
- `GetOUT`
- `GetValidTranDate`
- `GetNextValidTranDate`
- `InsertDummyTransmission`

- GetStockStat
- GetStockStatValues
- CheckStockStat
- GetStockWE
- GetStockOUT
- GetStockOUT_FG
- GetStockKORR
- GetStockKORRByCodeIDBewegKZ
- GetAllStock
- GetStockStatProduct
- GetStockCogs
- GetStockCogsProduct
- GetStockProductDetail
- GetFGProductDetail
- GetFAProductDetail
- GetStockCrossCheckUM
- GetInvoiceCheckData
- GetInvoiceGIT
- CalculateAccruedInvoiceValue
- RemoveGITAssignment
- UpdateGIT
- UpdateWE
- SetGITStockDateCorrect
- GetGitViewData

11.2.5.12 PKG_TARGETING

This package includes all procedures which are involved in the creation, correction and approval of the targets for the sales reps (see chapter "Targeting").

Implemented procedures and functions:

- GetLatestVersionForSaReYear
- CheckForApproval
- GetTargetVersions
- GetTargets
- GetTargetsForReport
- GetTargetsForApproval
- DeleteTargetVersion
- ApproveTargetVersion
- InsertTargetVersion
- CreateNewVersionFromOld

11.2.5.13 PKG_TARGET_PRODUCTGROUP

As the TargetProductGroup (TGP) is a very important central entity. All required procedures are packed together:

- `AddTPG`
- `GetTPGForSalesRep`
- `GetAllSalesReps`
- `GetAllUnasSalesRepsforTPG`
- `GetAllSalesRepByID`
- `GetTargetProductGroup`
- `GetProductsForTPG`
- `GetSalesRepsForTPG`
- `GetCustomersForTPG`
- `GetCustomersForSalesRep`
- `GetCustomersandSalesRepsForTPG`
- `GetAllCustomersandSalesRepsTPG`
- `GetAllCustomersandSalesReps`
- `GetSalesRepsForCustomerAndTPG`
- `GetAllUnasCustomers`
- `GetunassignedProducts`
- `GetUnassignedProductsForLine`
- `AddProducttoTPG`
- `DeleteProductofTPG`
- `AddSalesReptoTPG`
- `DeleteSalesRepofTPG`
- `RenameTPG`
- `DelTPG`
- `SetPercentforCustSaRe`
- `DelCustomer`
- `GetAllUnasCustomersForTPG`
- `GetTPGForCustomer`
- `GetCustomerPercent`

11.2.5.14 PKG_UTILITIES

The utility package was created to store additional utilities to be used in the different areas of programming.

The package only holds two functions which perform the two mathematical operations of division and multiplication.
As a division through zero would lead to an error and therefore to an unfilled materialized view or a broken report, the two functions perform the operation and in case of zero or invalid values they return zero back, without crashing the report or view.

The two functions are:

- `DIVISION`
- `PRODUCT`

12 Naming Convention

12.1 Introduction

In the implementation of the ALF-System the following naming convention were used:

12.2 Naming Convention Document

12.2.1 Code Layout

12.2.1.1 Indentation/Tabs/Space Policy

Indent using 2, 3, 4, or 8 spaces for each level.
Do not use tabs, use spaces. Most editors can substitute spaces for tabs.
Tabs should be fixed at 8 spaces. Please do not set tabs to a different spacing, use spaces instead.
Indent as much as needed, but no more. There are no arbitrary rules as to the maximum indenting level. If the indenting level is more than 4 or 5 levels you may think about factoring out code.

When people using different tab settings the code is impossible to read or print, which is why spaces are preferable to tabs.
Nobody can ever agree on the correct number of spaces, **just be consistent**. In general people have found 2, 3 or 4 spaces per indentation level workable.
As much as people would like to limit the maximum indentation levels it never seems to work in general. We'll trust that programmers will choose wisely how deep to nest code.

```
function f_func()
{
  if (something bad)
  {
    if (another thing bad)
    {
       while (more input)
       {
       }
    }
  }
}
```

12.2.1.2 Using white space

Always include a space between every identifier and separator in a statement. Instead of this:

```
WHILE(total_sales<maximum_sales AND company_type='NEW')LOOP
```

write this:

```
WHILE (total_sales < maximum_sales AND company_type = 'NEW')
LOOP
```

Use spaces to make module calls and their parameter lists more understandable. Instead of this:

```
calc_totals(company_id,LAST_DAY(end_of_year_date),total_type);
```

write this:

```
calc_totals (company_id, LAST_DAY (end_of_year_date),
total_type);
```

12.2.1.3 Blank Lines

Use **two blank lines** between logic sections of the source code.

Use **one blank line** between functions/procedures;
between declarations and main body;
between logic sections inside a code block for improved readability

12.2.1.4 UPPER-lower Style

All reserved words should be written in UPPERCASE and all application names should be kept in lowercase:

```
IF TO_NUMBER (the_value) > 22 AND
   num1 BETWEEN lval AND hval
THEN
   newval := 100;
ELSIF TO_NUMBER (the_value) < 1
THEN
   calc_tots (TO_DATE ('12-jan-95'));
ELSE
   clear_vals;
END IF;
```

12.2.1.5 Formatting Single Statements

The rule is: One line holds maximum one statement:

Wrong:

```
new_id := 15; calc_total (new_id); max_dollars := 105 * sales_adj;
```

Right:

```
new_id := 15;
calc_total (new_id);
max_dollars := 105 * sales_adj;
```

12.2.1.6 Formatting Multi Statements

Avoid lines longer 80 characters.
If longer use indentions for better readability and break lines according to these general principles:

- break after a comma
- break before an operator
- prefer higher-level breaks to lower-level breaks

```
generate_company_statistics (company_id, last_year_date,
   rollup_type, total, average, variance, budgeted,
   next_year_plan);
```

12.2.2 Naming Database Objects:

12.2.2.1 Prefix for Database Objects:

F_ Function
P_ Procedure
T_ Table
U_ User
V_ View
FK_ ForeignKey (only for Constrain)
MV_ Materialized View
PK_ PrimaryKey (only for Constrain)
CON_ Constrain
DBL_ DatabaseLink
DIM_ Dimension
DIR_ Directory
IDX_ Index
JOB_ Database Job
LIB_ Library
LOG_ Logs
PKG_ Package
POL_ Policy
PRO_ Profiles
QRY_ Query
QUE_ Queue
QUT_ QueueTable
RBS_ Rollback Segment
ROL_ Roles
SEQ_ Sequence
SYN_ Synonym
TBS_ TableSpace
TRG_ Trigger
TYP_ Types
UNI_ Unique Constrain (only for Constrain)
VAR_ Variables

12.2.2.2 Suffix for Database Objects:

_CD Code
_ID ID

12.2.2.3 ColumnNames

For easier working with columns of different tables each column name should hold a 4-letter identification of the table name:

T_Products
Prod_ID
Prod_Description
Prod_Price
Prod_…
…

12.2.2.4 ForeignKey ColumnNames

Due to the upper convention foreign keys will be stored like this:

T_ProductGroup.PrGr_ID

T_Products.PrGr_ID

T_Order.Prod_ID
T_Stock.Prod_ID

Through the different prefix you will immediately see foreign keys.

If there are several foreign keys of a specific table in a table then the different kinds of function should be added.

T_SampleOfProduct.Prod_ID_Sale_Product
T_SampleOfProduct.Prod_ID_Sample_Product
T_SampleOfProduct.SamP_Description

12.2.2.5 Singular / Plural

All Table names should be written in singular:

T_Product
T_Order
T_User
T_Customer

All column names should be written in singular:

T_Table.Description
T_Table.Price
T_Table.Info
T_Table.User_ID

12.2.3 Inline Documentation

12.2.3.1 Headers

Every Function, Procedure and Package should contain the following inline doc header:

```
/*****************************************************
* Author: Full Name – Abbreviation (here: DS)
* Date:   2003-08-13
*
* Overview: (Functionality of code)
*
* Major Modifications (when, who, what)
* (2003-08-13 DS Add Exception Handling)
*****************************************************/
```

12.2.3.2 Commenting out

If any modifications (after a release – not within an implementation phase) were done, the original code should be commented out (not deleted!) adding the date of out commendation and the author. The original code should insist!

```
/* 2003-08-12 DS because of the assumption below useless
var_sum1 = var_a + var_b
var_sum2 = var_c
*/
...
-- 2003-08-12 DS   var_sum3 = var_sum1 + var_sum2


var_sum3 = var_a + var_b + var_c
```

12.2.3.3 Descriptions

Every logically separating part of the code should contain a description of what it is going to do!

Note:

It is more important to explain the WHY of your program than the HOW!!

```
/*****************************************************
* Author: Full Name – Abbreviation
* Date:   2003-08-13
*
* Overview: This procedure fills the group number cells
*
*****************************************************/
PROCEDURE make_array (num_rows_in IN INTEGER)
IS
  col_id GROUPCOLUMN;
  rg_id RECORDGROUP;
BEGIN
  rg_id := CREATE_GROUP ('array');
  col_id := ADD_GROUP_COLUMN ('col');

  /* Use a loop to create the specified number of rows
and
  set the value in each cell.  */
  FOR row_index IN 1 .. num_rows_in
  LOOP
    ADD_GROUP_ROW (return_value, END_OF_GROUP);

    FOR col_index IN 1 .. num_columns_in
    LOOP
      /* Set the initial value in the cell */
      SET_GROUP_NUMBER_CELL (col_id, row_index, 0);

    END LOOP; //ending col_index loop

  END LOOP; //ending row_index loop
END;
```

If LOOPS or IFS or any other beginning and ending statement gets too encapsulated (too nested) it should be defined in the ending statement which beginning statement has been closed.

```
    END LOOP; //ending col_index loop
  END LOOP; //ending row_index loop
```