



Affordable, Adaptable ERP Software



Release Notes

Version 6.00

Table of Contents

INTRODUCTION	9
Introducing Fitrix ERP Version 6.0	9
New Manufacturing Modules and Programs.....	9
Job Shop	10
Product Configurator	10
Production Scheduling	12
Labor Processing.....	13
Capacity Requirements Planning.....	15
Standard Costing	16
New Customer Relationship Management (CRM) Module	18
Barcode Support.....	20
Other Major Application Features.....	21
NEW APPLICATION FEATURES	22
GENERAL	22
FR 2418/2664/2715 – Ability to send password protected attachments with automated email alerts	22
FR2720 – View subsidiary screens while not in update mode	23
FR2729/2809 – Improve post locking	23
FR2767 – Merge or replace department codes.....	24
FR2813 – Merge tax groups codes	24
FR3079 – User defined role based menus (a.k.a. “favorites menu”)	24
FR3457 – Print ALL option.....	25

ACCOUNTS PAYABLE.....	26
FR2416 – Open PO detail	26
FR2423 – AP Check Print	26
FR2424 – Positive Pay file creation.....	27
FR2530 – New buttons in Vendor Master to access email and website.....	27
FR3181 – Add payment detail to the Vendor Ledger report.....	27
FR3267 – New cash register report	28
FR3484 – Streamlined AP transaction processing	28
FR3636 – Merge Vendors.....	28
ACCOUNTS RECEIVABLE.....	28
FR2143 – Account statements by salesperson.....	28
FR2177 – New fields for YTD and LTD sales in customer master	29
FR2337 – Additional contact fields	29
FR2348/2420 – Auto assign next customer number and ship-to code	30
FR2417 – Account type and account source fields.....	30
FR2522 – Variable commission rates at the Customer Master level.....	30
FR2530 – New buttons in Customer Master to access email and website.....	31
FR2804 – Improvements to Cash Receipts screen program	31
FR3103 – AR Statement can be run retroactively	32
FR3104 – Process multiple cash receipts per invoice	32
FR3180 – Add payment detail to the Customer Ledger report.....	32
FR3484 – Streamlined AR transaction processing	32
BILL OF MATERIAL	33
FR1799 – Increase sequence Number size.....	33
FIXED ASSETS.....	33

FR3650 – Depreciation Activity Report.....	33
FR3654 – Net Book Value Report	33
FR3654 – Net Book Value Report	33
GENERAL LEDGER	33
FR2002 – User definable balance recalculation start date.	33
FR2920 – Suppress print of accounts with zero balance on GL activity reports	33
FR3484 – Streamlined GL transaction processing	33
INVENTORY CONTROL	34
FR2253 – Item lead time analysis	34
FR2333 – Substitute or Alternate items.....	34
FR2343 – Restocking fee can now be a dollar amount or percentage	34
FR2335 – Special handling fees	34
FR2341 – Test pricing screen.....	35
FR2342 – New fields on Purchase/Sales History screen	35
FR2345 – Inventory labels	36
FR2349 – Lost sales notes	36
FR2352 – Sales projections.....	36
FR2378 – Display available quantity	37
FR2408 – Cross sell items	38
FR2410 – Special order items.....	39
FR2414 – Inventory audit report	39
FR2426 – Multiple units of measure.....	39
FR2433 – Print item specific notes	40
FR2566 – Item shortage by warehouse inquiry screen	40
FR2932 – New Serial/Lot Composition screen	41

FR2932 – New serial master screen.....	41
FR3020 – Update list price by item class	42
FR3051 – Display committed quantity in the Inventory Adjustment program.....	42
FR3185 – Changes to the Reorder Advice report	43
FR3273 – Indented BOM report can now be exported to Excel.....	43
FR3343 – Product warranty tracking	43
FR3347 – Quality Control	43
FR3389 – Reference designators for bills of material.....	44
FR3421 – Update of vendor code in item master	45
FR3428 – Display of bin location screen in Component Issue program	45
FR3484 – Streamlined inventory transaction processing	46
FR3447 – Production line inventory	46
FR3541 – Indented serial/lot composition.....	47
FR3542 – Indented serial/lot where used.....	48
MATERIAL PLANNING	48
FR2478 – Allow planned orders from MRP generation to be added to existing purchase orders.....	48
ORDER ENTRY.....	48
FR2242- Order type display.....	48
FR2332/3182 – Streamlined order entry process	48
FR2339 – Payment summary on sales register	49
FR2351 – Export invoice to Excel.....	49
FR2353 – Print lot number on product detail report.....	49
FR2362 – Interface with Federal Express and UPS	49
FR2394 – RMA status code	49
FR2406 – Item count on pick ticket	50

FR2407 – Salesperson contact information	50
FR2409 – Clone sales orders	50
FR2411 – Sales Order Archive	51
FR2412 – Recurring or standing sales orders.....	51
FR2413 – Split payment terms	52
FR2422 - On the fly ship-to addresses	52
FR2421- Print invoices for specific ship-to location.....	53
FR2425 - Federal Express billing import and cost analysis.....	54
FR2483/2570 – Additional information on the purchase/sales history drilldown screen.....	54
FR2494- Salesperson default logic.....	55
FR2514- Search by phone number	56
FR2523- Reduce commission paid on invoices paid late	56
FR2565- Added business name and salesperson code to backorder release screen	57
FR2619 - Added customer business name to the salesperson Sales History reports.....	57
FR2630 – Any freeform notes attached to ship-to will display in sales order entry	57
FR2631 – Set default warehouse at customer and ship-to level.....	57
FR2719 – Access to alternate items.....	59
FR2722 – Flip warehouse	59
FR2750 – Add gross margin percent to all sales order history reports	59
FR2794 – Update end date for special pricing	59
FR2841 – Can now cancel customer invoice after it’s been printed	59
FR2874 – New screen program to add vendors and costs to drop ship PO types	59
FR3021 – Default Customer PO number on RMA	60
FR3033 – Change to Pick Ticket print when quantities or items change.....	60
FR3097 – Variable customer pricing by warehouse location.....	60

FR3098 – Expanded description and contact fields on sales order header	60
FR3100 – Access to production orders from sales order entry	60
FR3190 – Print production work order number on customer invoice.....	61
FR3235 – Blanket sales orders.....	61
FR3270 – New open sales orders report (exported to Excel)	61
FR3300 – Available to promise date (ATP).....	62
FR3323 – Customer credit management	62
PAYROLL.....	63
FR1927 – FDD for direct deposit check stubs.....	63
FR2436 – Payroll date verification.....	63
FR3617 – Additional Medicare Tax.....	64
PRODUCTION ORDER PROCESSING	64
FR2573 – New Production Order Status report	64
FR3125/3374 – Work in process cost report.....	65
FR3159 – Add short components to existing purchase order	65
FR3269 – Production order log report (exported to Excel).....	65
FR3298 – Job cost summary report by salesperson	66
FR3505 – Recording production scrap	66
FR3555 – Assign component serial numbers to Parent items	67
PURCHASING	68
FR2284 – New fields for YTD and LTD purchases in vendor master	68
FR2340 – Email alerts for backordered merchandise	69
FR2415- Vendor rebate processing	69
FR2417- Reverse PO receipts	69
FR2442 – Print your account number with your vendor on the purchase order	70

FR2569 – Change warehouse on all detail line if changed in PO header.....	70
FR2569 – Print labels for PO receipts	70
FR3052 – Display vendor item code on PO screens	70
FR3182/3202 – Streamlined purchasing transaction processing	70
FR3191 – Print production order number on PO form.....	71
FR3206 – Approved vendor list (AVL)	71
FR3217 – Compliance support for regulatory and industry standards.....	71
FR3223 – Blanket purchase orders	72
FR3268 – Purchase order log report (exported to Excel).....	73
FR3501 – Vendor delivery performance report	73
REPLENISHMENT	74
FR2620 – Add vendor business name to recommendations screen	74
FR2621 – Ability to create buy recommendations by vendor.....	74
FR2755 – Add excel export capabilities to the Expedite report	74
FR3034 – Base average usage calculation on date item was added to the database	74
FR3035 – Missing catalog entries report	75
FR3133 – Utility to reset RL Script definitions.....	75
TECHNICAL NOTES.....	76
NEW TABLES.....	77
MODIFIED TABLES	144
SUPPLEMENTAL PATCHES AND ENHANCEMENTS	171
Accessing Fourth Generation’s FTP site.....	171
Installing Supplemental Patches and Enhancements.....	172
Naming Conventions.....	172
Program Installation procedures.....	172

INTRODUCTION

Introducing Fitrix ERP Version 6.0

Fourth Generation Software Solutions is pleased to release version 6.0 of its Fitrix ERP software. This new release is the culmination of thousands of hours of development by our software engineering team since our last **major** release, version 5.4.

Version 6.0 includes significant enhancements across the Fitrix ERP suite: the release of additional manufacturing modules, an integrated CRM module, barcode support, over 100 new application features, and new “tools” features. The remainder of this document discusses these new features and capabilities in detail, but below is a brief overview.

New Manufacturing Modules and Programs

Fourth Generation Software acquired several advanced manufacturing software modules written with our CASE Tools from our longtime reseller Kingwood Systems. We initiated the process of converting these from the older Four J's Genero BDL user interface to the latest Genero technology in addition to making these new modules work with Fourth Generation's latest Accounting and Distribution modules and the Fitrix ERP data model.

With version 5.4 we released Bill of Material, Production Order Processing, Standard Routing, and Material Requirements Planning modules. With version 6.0 we have added job shop capabilities to the Production Order Processing module, added a Product Configurator to the Inventory Control module and these new manufacturing modules: Production Scheduling, Labor Processing, Capacity Requirements Planning and Standard Costing. Below is brief description of each of these.

Job Shop

Jobs are an accumulation of one or more of these cost elements:

- Material, both stock and nonstock
- Labor
- Overhead
- Miscellaneous charges
- Outside process or contract costs

Using Sales Order Entry in conjunction with Production work orders, Routing, and Labor Processing you can create “jobs” for make to order stock or nonstock items (MTO, MTN) and roll up costs that will then calculate a rolled up price unless the sales order is designated as fixed priced. The Job Cost/Price Detail report gives the user an analysis of expected costs to actual costs and actual costs to price charged.

Product Configurator

The Product Configurator defines optional features to be selected and included in the production of an item. Features are defined in the form of questions to be answered at the time an order is placed for a ‘configurable’ item. Questions have one or more answers that in turn:

- Include specific additional component items to be included on the production order
- Include specific additional routing steps to be included on the production order
- Activate/deactivate other related questions
- Activate/deactivate other related answers
- Automatically set answers to other related questions

FEATURES/FUNCTION HIGHLIGHTS

Configuration Definition

- Define one or more questions (feature), with expanded text, for clear readability
- For each question, define one or more answers, with expanded text
- For each answer, define one or more responses, as explained above

Configuration Generation

- Automatically accessed when a configurable item is placed on a sales order
- Automatically accessed when a production order is entered for a configurable item
- Prompts the user to answer questions

- Allows user to select a previous configuration to auto-fill default answers to questions

Configuration Inventory Inquiry

- Displays the configuration for an item produced to stock

Configuration Inquiry

- Displays the configuration details for any previously configured item

Configuration Detail Report

- Prints the configuration definition in expanded format, including questions, answers within questions, and responses within answers

Configuration Simulator

- Displays material availability and resource load for a proposed configuration
- Useful in a 'what-if' scenario, before an order is placed for a configurable item
- Resources analyzed are Work Centers, Departments, Teams, Machines
- User-defined period intervals

The screenshot shows the 'Product Configuration' window. The menu bar includes File, Edit, View, Navigation, Tools, Actions, Options, and Help. The toolbar contains icons for various actions like Find, Prev, Next, Add, Update, Delete, Browse, and Options. The main area displays configuration details for item 'RB-B9' (B9 ROBOT) with options like 'MIAMI' for Whse and 'ROBOT' for Group. A table lists configuration questions and answers:

Question	Answer
1 Do you want a clear or neon head bubble?	Clear bubble
3 Do you want a camera?	Yes
4 What type of lens for the camera?	100MM Lens
5 Do you want clear or neon chest lights?	Clear chest light
7 Do you want a male or female voice?	Male voice
8 What language do you want?	English
9 Do you want a soil sampler?	Yes

On the right, there is a preview image of the robot and fields for Cost, Price, Margin, and Percent (0.00).

Production Scheduling

Fitrix Production Scheduling gives you the tools to maximize efficiency while reducing costs. You can view at a glance which resources are being under or over utilized and can adjust your production accordingly to balance the load. As new orders are placed or existing order due dates change, you can see real time the effect on your production capacity. The application also provides recommendations to changes in order due dates based on the work remaining to be done.

FEATURES/FUNCTION HIGHLIGHTS

Period Intervals

- Define how the future is to be divided into time periods
- Multiple period intervals may be defined
- Used to display and print future production activity by time period

Capacity Maintenance

- Allows for non-standard override of daily capacity, by shift, for Work Center, Department, Teams, Machines

Reschedule Operations

- Recalculates operation due dates when order due dates are changed

Work List Inquiries and Reports

- Display the status of operations for open orders by resource. Types of operations include Active, Waiting, Arriving
- Resource types are Work Center, Department, Teams, Machines

Capacity Inquiries

- Displays capacity and load for a given resource, with overload or under-load percentage
- Resources analyzed are Work Center, Department, Teams, Machines
- User-defined period intervals to view loads in a variety of formats
- Drill-down from period load to details of orders and operations
- Review alternates for an overloaded resource

Reschedule Recommendations

- Report which highlights recommendations for changes in order due dates based on work remaining

Order Status by Warehouse

File Edit View Navigation Tools Actions Help

Find

Prev

Next

Warehouse Code: MIAMI

Start Date: 09/15/2014

Interval Code: A

Order	Rel	Item	Start	Due	Pty	Customer/Stock	Sales Value	Qty Remain	Hrs Rem	Material Status	Past	09/15	09/22	09/29	10/06	10/13
345	000	MISC	05/02/2014	05/02/2014		1262/12	228.00	50.000	0.000	Issued						
345	000	MISC	05/02/2014	05/02/2014		1261/12	222.00	50.000	0.000	Issued						
343	000	REPAIR	05/02/2014	05/02/2014		1258/12	500.00	1.000	0.000	OK to Issue						
344	000	REPAIR	05/02/2014	05/02/2014		1259/12	599.00	1.000	0.000	OK to Issue						
351	000	REPAIR	05/02/2014	05/02/2014		1267/12	350.00	50.000	0.000	OK to Issue						
348	000	REPAIR	05/02/2014	05/02/2014		1264/12	100.00	50.000	0.000	OK to Issue						
350	000	REPAIR	05/02/2014	05/02/2014		1266/12	300.00	50.000	0.000	OK to Issue						
347	000	REPAIR	05/02/2014	05/02/2014		1263/12	500.00	1.000	0.000	Issued						
342	000	12104	05/01/2014	05/05/2014	Stock/		0.00	50.000	0.000	Short-12112-3444						
336	000	C-MAC	04/30/2014	05/05/2014	Stock/		0.00	10.000	10.00	OK to Issue						
352	000	C-MAC	05/02/2014	05/05/2014	Stock/		0.00	10.000	20.00	OK to Issue						
339	000	C-MAC	04/30/2014	05/05/2014	Stock/		0.00	10.000	30.00	OK to Issue						

1 of 1

OK

Cancel

Header

Oper Details

Mat Short

Enter Production Order

Labor Processing

Fitrix Labor Processing collects labor transactions for production orders. Labor is entered for both setup and run hours, and for direct and indirect hours. Order completion statistics are updated real-time. The hours and completion are analyzed for efficiency at multiple levels within the organization. Efficiency inquiries and reports show this performance in summary and detail formats. Labor Processing is an application in the Production management family. It provides a complete picture of the activities that occur and the costs that were incurred in completing production work orders and supports the flow of this information into the general ledger, payroll and costing modules.

FEATURES/FUNCTION HIGHLIGHTS

Employee Maintenance

- Employees can belong to a Department, Team, Group, Job class
- Labor rates applied to orders by Regular time, Overtime, Premium time

Shift Maintenance

- Multiple shifts per physical shift
- Early/late start time window
- Early/late stop time window
- Lunch extraction
- Break extraction

Job Class Maintenance

- Standard rates used as alternate for labor costing

Labor Type Maintenance

- User-defined labor transaction types
- Setup or run hours based
- Direct or indirect hours based

Labor Transaction Entry

- Setup labor
- Run labor
- Indirect labor
- Start/stop time or elapsed time
- Completed and scrapped quantities
- User-defined percent complete
- On-line order update for hours and quantities

Labor Efficiency Inquiries

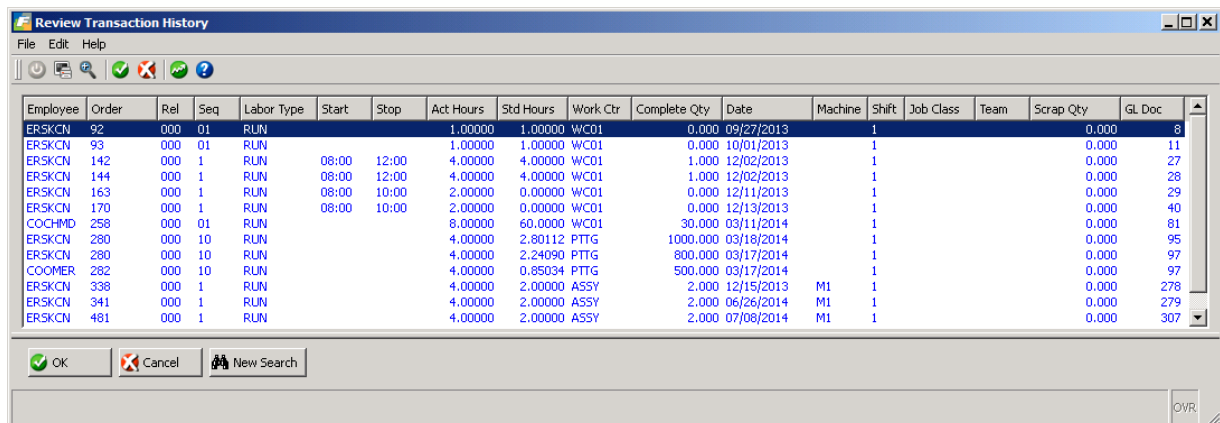
- Efficiency by Team, Employee, Work center, Department, Item, Order
- Transactions analyzed by ranges of Resources, Transaction dates, Shifts

Labor Transaction Reports

- Print by ranges of Department, Work center, Machine, Group, Team, Foreman, Employee, Order

Labor Efficiency Reports

- Efficiency by Department, Employee, Order, Work center, Machine, Team
- History by Department, Employee, Order, Work center, Machine, Team



The screenshot shows a software window titled "Review Transaction History" with a menu bar (File, Edit, Help) and a toolbar. Below the toolbar is a table with the following columns: Employee, Order, Rel, Seq, Labor Type, Start, Stop, Act Hours, Std Hours, Work Ctr, Complete Qty, Date, Machine, Shift, Job Class, Team, Scrap Qty, and GL Doc. The table contains 20 rows of transaction data. At the bottom of the window are buttons for "OK", "Cancel", and "New Search", and a small "OVR" indicator in the bottom right corner.

Employee	Order	Rel	Seq	Labor Type	Start	Stop	Act Hours	Std Hours	Work Ctr	Complete Qty	Date	Machine	Shift	Job Class	Team	Scrap Qty	GL Doc
ERSKCN	92	000	01	RUN			1.00000	1.00000	WC01	0.000	09/27/2013		1			0.000	8
ERSKCN	93	000	01	RUN			1.00000	1.00000	WC01	0.000	10/01/2013		1			0.000	11
ERSKCN	142	000	1	RUN	08:00	12:00	4.00000	4.00000	WC01	1.000	12/02/2013		1			0.000	27
ERSKCN	144	000	1	RUN	08:00	12:00	4.00000	4.00000	WC01	1.000	12/02/2013		1			0.000	28
ERSKCN	163	000	1	RUN	08:00	10:00	2.00000	0.00000	WC01	0.000	12/11/2013		1			0.000	29
ERSKCN	170	000	1	RUN	08:00	10:00	2.00000	0.00000	WC01	0.000	12/13/2013		1			0.000	40
COCHMD	258	000	01	RUN			8.00000	60.00000	WC01	30.000	03/11/2014		1			0.000	81
ERSKCN	280	000	10	RUN			4.00000	2.80112	PTTG	1000.000	03/18/2014		1			0.000	95
ERSKCN	280	000	10	RUN			4.00000	2.24090	PTTG	800.000	03/17/2014		1			0.000	97
COOMER	282	000	10	RUN			4.00000	0.85034	PTTG	500.000	03/17/2014		1			0.000	97
ERSKCN	338	000	1	RUN			4.00000	2.00000	ASSY	2.000	12/15/2013	M1	1			0.000	278
ERSKCN	341	000	1	RUN			4.00000	2.00000	ASSY	2.000	06/26/2014	M1	1			0.000	279
ERSKCN	481	000	1	RUN			4.00000	2.00000	ASSY	2.000	07/08/2014	M1	1			0.000	307

Capacity Requirements Planning

Capacity Requirements Planning analyzes loads placed on production resources. It reviews standard routing information for planned and released production orders, to determine the resources needed to satisfy production requirements. Loads are measured against available capacity, in user-defined time intervals, to support exception analysis.

FEATURES/FUNCTION HIGHLIGHTS

Period Intervals

- Define how the future is to be divided into time periods
- Multiple period intervals may be defined
- Used to display and print future production activity by time period

Capacity Maintenance

- Allows for non-standard override of daily capacity, by shift, for Work Center, Department, Teams, Machines

CRP Generation

- Generates planned order routings
- Typically run in conjunction with MRP generation

Change Routing

- Allows changes to routing for each planned order
- Handy when shifting load to optimize capacity

Capacity Inquiries

- Displays capacity and load for a given resource, with overload or under-load percentage
- Resources analyzed are Work Center, Department, Teams, Machines
- User-defined period intervals to view loads in a variety of formats
- Drill-down from period load to details of orders and operations
- Review alternates for an overloaded resource

Capacity Reports

- Prints capacity and load for a given resource, with overload or under-load percentage
- Loads prints in graphical format for easy exception analysis
- Resources analyzed are Work Centers, Departments, Teams, Machines
- User-defined period intervals

Capacity/Load by Work Center

File Edit View Navigation Tools Actions Help

Warehouse Code: MIAMI Start Date: 09/15/2014 Interval Code: A

Percent Loaded

Work Ctr	Description	09/15	09/22	09/29	10/06	10/13	10/20	10/27	11/03	11/10	11/17	11/24	12/01	12/08	12/15	12/22	12/29
ASSY	GENERAL ASSEMBLY	150	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
MOLD	INJECTION MOLDING	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
QC	QUALITY CONTROL	75	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
WCD1	PRIMARY ASSEMBLY	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

1 of 1

View Detail

OVR

Standard Costing

Fitrix Standard Costing is an application in the Item Management family. It defines the expected or standard costs to be incurred to produce or purchase items. It uses cost elements defined in the Bill of Material and Standard Routing applications to compute standard costs per unit of an item. These costs can be tracked against actual costs to produce or purchase and the variances can then be analyzed.

In addition to the current standard cost, Fitrix supports the definition of an unlimited number of other standard costs for items. This means you can store multiple historical standards, multiple future standards, and unlimited simulation costs. Any standard may be compared to any other standard via inquiries and reports and variances are analyzed at multiple levels of detail.

FEATURES/FUNCTION HIGHLIGHTS

- Modular Integration – direct integration with Bill of Material, Standard Routing, Production Order Processing, and Purchasing
- Current standard cost stored at the item level. When used with Bill of Material and Standard Routing costs can be “rolled up” to compute a current standard cost
- An unlimited number of other costs (historical, future, simulated) can be established for each item for variance analysis
- Flexible cost roll up and user defined control over automatic rollup
- Allow manual entry of cost elements with no roll up
- Costs can be used as templates that can then be transferred to create other standard costs (i.e. – current costs can be used to create a new year frozen standard, simulated costs can be used to create a new current standard)
- Routing steps are costed based on setup hours, run hours per unit or machine hours per unit

- Labor and overhead costs can be optionally be calculated based on standard work center labor and overhead rates
- Actual costs can be compared to either current or other standard costs
- Standard unit cost will be the default used on purchase orders entered manually, released from MRP, and released from production order entry unless the vendor is found in the approved vendor catalog

Inquiries

- Current cost details
- Comparison of multiple cost types: Summary, Component Detail, Routing Detail

Reports

- Current cost details
- Comparison of multiple cost types by: Summary, Component Detail, Routing Detail
- Inventory valuation at standard cost type

Setup Standard Costing

File Edit View Navigation Tools Actions Help

Update

Inventory Cost Codes

Current Costs: STD

Default Cost Elements

Material: MATERIAL

Non-Stock Material: NON-STOCK MATERIAL

Labor: LABOR

Overhead: OVERHEAD

Outside Process: OUTSIDE

Setup: SETUP

Application Setup

Setup Complete: Y

1 of 1

OVR

New Customer Relationship Management (CRM) Module

Functions within a business organization are often referred to as 'front-office' or 'back-office' depending upon whether they are 'customer-facing' like an inside sales group or a customer support call center, or are more internally- or supplier-oriented like accounting/finance, purchasing or manufacturing. The Fitrix Customer Relationship Management or CRM system addresses the former, and includes capabilities that are intended to help 'customer facing' departments in the company like sales, marketing and customer service/support do their day-to-day jobs.

The main focus of a CRM system is to help ensure that customer needs are met in a timely and efficient manner, so that customer satisfaction is maintained at high levels throughout the organization and with every customer interaction, and so that the company's revenue and profitability goals are able to be met or exceeded. By having CRM integrated with an ERP system, it is possible to gain and maintain a '360 degree view' of your company's customers and to support analysis which helps identify your most important and profitable customers, as well as your less profitable and more costly ones.

The Fitrix Customer Relationship Management module is designed to meet the most critical needs of the three main 'front office' constituents:

- Sales,
- Marketing, and
- Service/support personnel

Since companies often vary greatly as to how these operations are organized, all of Fitrix CRM's functions within each of these areas may or may not be utilized by your company, or might be implemented in a 'hybrid' manner to best meet your business objectives. Certain functions like Activity Management might be used by all users in all of these departments, or just those in one or two.

There are sophisticated applications that are designed to be 'best in class' individual or suite-based CRM solutions that include more specialized capabilities than Fitrix CRM has at this time. Fitrix CRM is designed to meet the core CRM needs of the typical discrete manufacturer or wholesale distributor, and to provide close integration between these core CRM capabilities and the other ERP functionality provided within the Fitrix software suite. With this CRM/ERP integration, a '360 degree view' of your customers and their individual impact on your operations and profitability can be accurately measured and optimized over time.

FEATURES/FUNCTION HIGHLIGHTS

- **Modular Integration** – Direct integration with other Fitrix ERP modules
- **Account Management** – Fitrix CRM allows users to enter and maintain various demographic and other information about suspects, prospects and customers, and even other organizations like

business partners, associations and competitors. Companies can be organized into hierarchies to track the 'parent-child' relationships of parent companies and subsidiaries at multiple levels.

- **Contact Management** – An unlimited number of contacts can be maintained for each company/account, with one being designated as 'primary' for each Account. Multiple phone numbers, email addresses and physical addresses can be managed for each contact.
- **Lead & Opportunity Management** – Fitrix CRM's Lead and Opportunity management capabilities allow sales and marketing personnel to track interactions and interest with Accounts from the very first (e.g. a trade show 'lead') to multiple individual sales opportunities for different products or services the Account may be interested in, with forecasting and tracking.
- **Quotes & Proposal Management** – With Fitrix CRM you can quickly and easily see quotes and proposals that you have sent your prospects and customers and turn these into sales orders when they are sold.
- **Team Selling Support** – Fitrix is designed with team selling capabilities built in that will continue to be enhanced over time to support a robust 'role-based' multi-member Account team model that is typical in more sophisticated sales organizations, but can also be used in more simplified settings with single-member or small sales teams.
- **Activity & Project Management** – Fitrix CRM lets users manages all kinds of Activities within Fitrix like calls, appointments, 'to-do's as well as build their own specialized types of Activities and track them. Projects are templates of reusable sets of Activities that can be scheduled and assigned to all or just Primary Contacts using Fitrix CRM's Campaigns features.
- **Campaign Management** – Fitrix CRM's Campaigns capabilities allow sales, marketing and service/support teams to build robust 'multi-media/multi-modal' campaigns that can be scheduled, assigned and managed to various lists of Accounts based on user-defined criteria.
- **List & Query Management** – The CRM module of Fitrix ERP allows users with proper access the ability to build, use and reuse powerful SQL queries via a standard Query By Example query capability. These lists can then drive Campaigns or feed other applications targeting prospects and/or customers with your company's communications or other interactions.
- **Case/Incident Management** – Fitrix CRM includes a powerful Case Management capability that allows customer service/support personnel to track and manage cases/calls/incidents/issues/etc. Help desk or call center users can assign cases to the most competent/available personnel using a rules-based facility.
- **Problem Management** – Tracking problems and identifying trends so that major customer service issues can be addressed proactively is an important feature that Fitrix CRM's Problem management feature helps automate.
- **Scheduling Service/Support Calls** - This set of programs enables the scheduling of work out in the field. They allow for the dispatcher/service coordinator to enter appointments for customers, print or email the service ticket for the appointment, and then review all active appointments via either an inquiry program or a calendar program in four different views; by employee, by service coordinator, by customer or by type of service being performed. Also included is a suite of reports that can be emailed to the employee, service coordinator, or customer that contain details on active appointments for the date range specified.

Accounts

File Edit View Navigation Tools Actions Help

Find Prev Next Add Update Delete Browse

Company

Account Name: ACTION ELECTRONICS Account ID: 1

Parent Name: Parent ID:

Web Site: WWW.ACTIONAUTO.COM Revenue: 5,000,000

Industry Type: WHOLESALE Employees: 85

Standard: NAICS Code: 335110 Ticker:

ELECTRIC LAMP BULB AND PART MANUFACTURING Created By: fitrix 05/30/2015

Info

Account Type: DISTRIBUTOR

Status: ACTIVE

Source: TRADE SHOW

Customer Code: 1

Account Rep: 1 - MARY BROWN

Team: 1 - SALES

Credit Limit: 200,000

Balance: 65

Yr-to Date: 36,234

Lifetime: 142,682

Address

Type: OFFICE City: ATLANTA

Address1: 14307 1ST STREET State: GA County: FULTON

Address2: Post: 30399 Country: 1

Primary Contact

Name: MR. BILL WALKER

Title: PRESIDENT Type: DECISION MAKER

Department: EXECUTIVE Email: OFFICE BILLW@ACTIONELECTRONICS.COM

Phones

MOBILE (404) 345-2019 x Int'l

x Int'l

x Int'l

Contacts (2) | Addresses (2) | Activities (2) | Activity History | Campaigns (1) | Campaign History | Quotes (3) | Orders (105) | Leads | Opportunities (2) | Cases (1)

Primary	Name	Title	Phone Type	Number	Ext.	Email Address
<input checked="" type="checkbox"/>	BILL WALKER	PRESIDENT	OFFICE	(404) 567-0498	1200	BILLW@ACTIONELECTRONICS.COM
<input type="checkbox"/>	JEAN A DONOVAN	PURCHASING MAN...	OFFICE	(404) 567-0498	1206	

Barcode Support

Using the barcode functionality Fitrix ERP provides is a great way to save time and cut costs. It captures data faster so efficiency is higher and costs are lower, mistakes are minimized and managing inventory in general becomes much easier.

Barcode scanning can be used in the following processes in Fitrix ERP

- Purchase Receipts and Putaway
- Component Issue
- Production Receipt
- Sales Order Picking and Shipping
- Bin location moves
- Inventory Transfers between Distribution Centers
- Inventory Adjustments
- Inventory Cycle Counting
- Forms and Labels to support these operations

The barcode scanning devices are connected real-time to the Fitrix modules eliminating the overhead involved in synchronizing data between separate systems. Users can choose to work with a graphical interface (GUI) very similar to the desktop-based windows in the rest of Fitrix or a text based interface (TUI).

Fitrix supports the use of both handheld devices that are connected to the keyboard via a keyboard wedge and wireless handheld devices that connect to Fitrix via a Wi-Fi network or the internet. Though wireless devices give you a lot more mobility you can definitely keep your costs down by using wedges devices.



Other Major Application Features

Details on these exciting new features are listed under their respective module.

- Customer Credit Management
- Role Based Menus
- Quality Control Management
- Streamlined Processes throughout the Modules
- Real-time Interface to Federal Express and UPS via web services
- Email Alerts can now include Password Protected Attachments

NEW APPLICATION FEATURES

The following section lists all of the new application features added since the official release of version 5.40. Your company may already have some of these features depending upon when you last upgraded. When an upgrade takes place, Fourth Generation always installs the latest version and then also includes any features completed to date.

GENERAL

FR 2418/2664/2715 – Ability to send password protected attachments with automated email alerts

An example of this is the order acknowledgement (program name = o_order) being attached to the email alert sent to both the credit manager and sales rep when an order is placed on hold.

The screenshot displays two overlapping windows from the Fourth Generation Software interface. The top window, titled 'Update E-Mail Templates', contains fields for 'Event Code' (HOLD_SALES_ORDER), 'Description' (SALES ORDER ON CREDIT HOLD), 'Active' (Y), 'Type' (EMAIL), and 'Priority' (1). The 'Message Subject' field is populated with 'Sales Order &order_no held with Hold Code &hold_code'. Below these fields is a 'Text' area containing two lines of text: '1 Sales Order &order_no dated &order_date for &total_amount is being held&n' and '2 because &bus_name has exceeded their credit limit.' The bottom window, titled 'Define Reports', shows a table with columns: Sequence, Module, Program, Ver, Parameters, PW, Password, and File Name. The first row contains the following data: Sequence 1, Module oe, Program o_order, Ver 4gs, Parameters -reprint &reprint -docno &doc_no_list, PW checked, Password empty, and File Name empty. At the bottom of the 'Define Reports' window are 'OK' and 'Cancel' buttons.

Sequence	Module	Program	Ver	Parameters	PW	Password	File Name
1	oe	o_order	4gs	-reprint &reprint -docno &doc_no_list	<input checked="" type="checkbox"/>		

FR2720 – View subsidiary screens while not in update mode

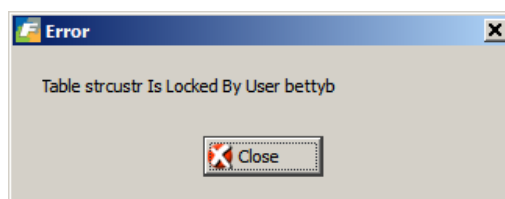
This feature applies to many programs in the various modules. An example would be being able to view the customer activity screen in the Customer Master without first having to go into update mode. This was done to cut down on record locking errors users were experiencing. Also see FR2729 for the new record locking notifications that are in place in version 6.00

FR2729/2809 – Improve post locking

Print message on the posting report that hash user name and table locked instead of having the program error out in the following posting programs:

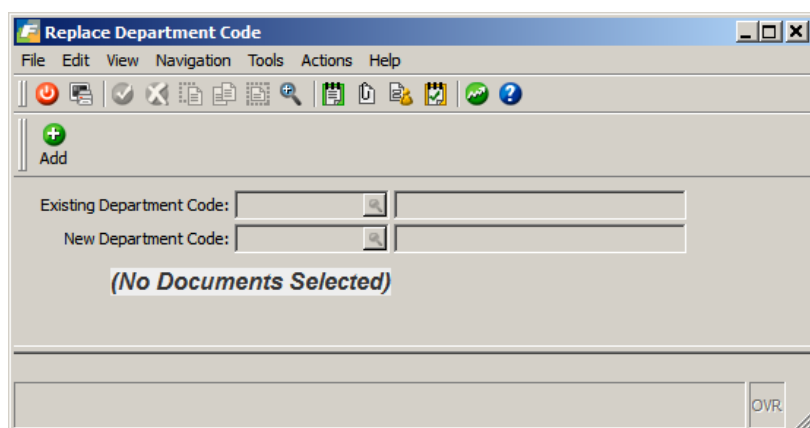
AR:	post cash receipts
	post AR invoices
IC :	cycle counts
	inv received
	inv adjustments
	inv shipped
	inv transfers
OE:	post oe invoices and memos
PU:	post AP invoices
	post PO receipts

If the new Use Posting Error Notification check box on the Update Company Information screen is checked, the user will see a message like this display on their screen.



FR2767 – Merge or replace department codes

New program that allows user to merge or replace accounting department codes

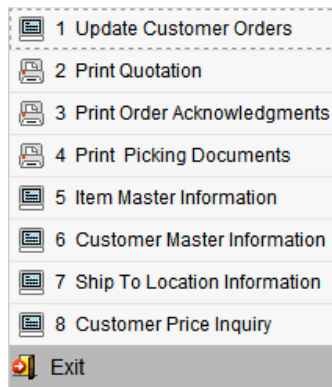
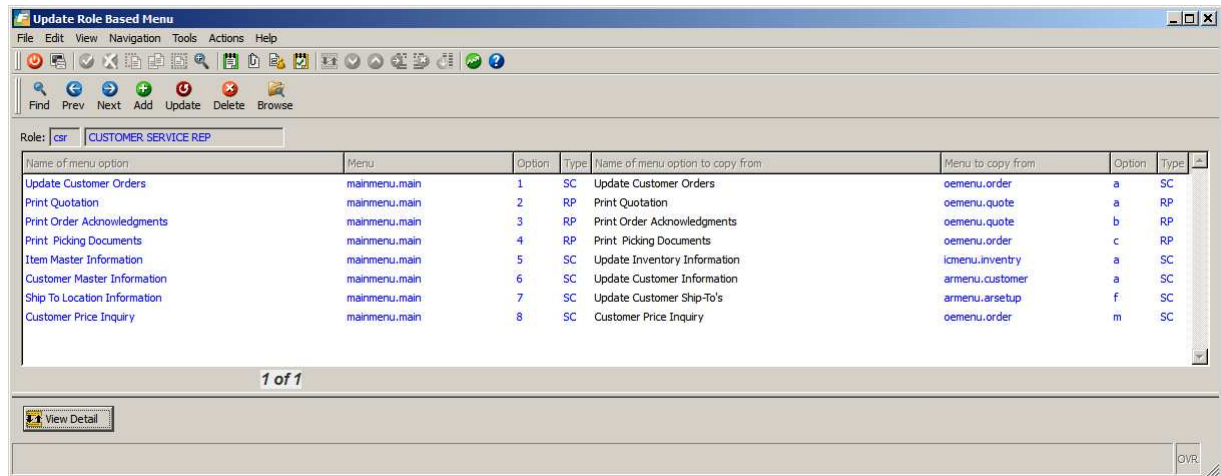


FR2813 – Merge tax groups codes

Use this program in the event that more than one tax group was set up for the same tax jurisdiction in error. The program will transfer all historical information from the Existing Tax code to the New Tax code (the real one you want to retain) and delete the code entered in error.

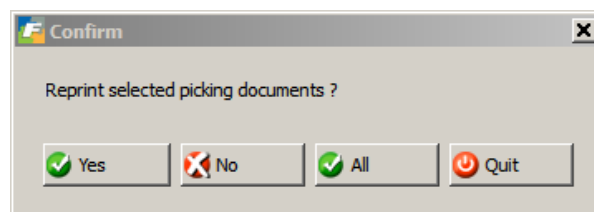
FR3079 – User defined role based menus (a.k.a. “favorites menu”)

These utility programs allow you to specify what menu options employees should have access to based on their job responsibilities thereby reducing the amount of menu navigation they need to do on a daily basis. Once the roles have been set up you can assign users or groups of users to one or more roles. When they log in to Fitrix the menu displays the programs associated with the role marked as their default. Fitrix is delivered with eight predefined roles that can be edited or added to. Below is an example of the role-based menu defined for a customer service representative (csr).



FR3457 – Print ALL option

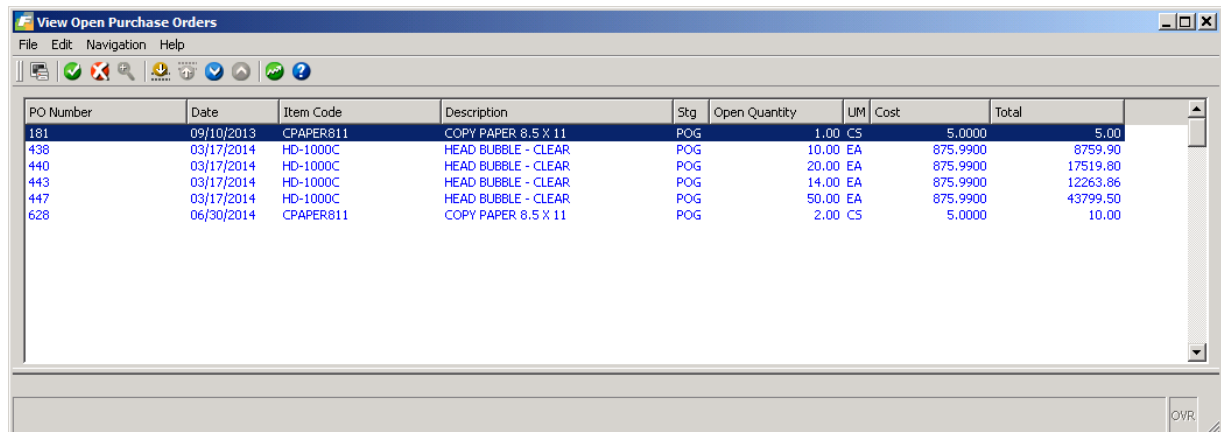
All print programs that had the option to Reprint Y/N option have been changed to include a Print All option. This is handy if you don't know if the document has already been printed or not (i.e. - pick tickets).



ACCOUNTS PAYABLE

FR2416 – Open PO detail

New open PO detail screen accessible from the Vendor Master screen.



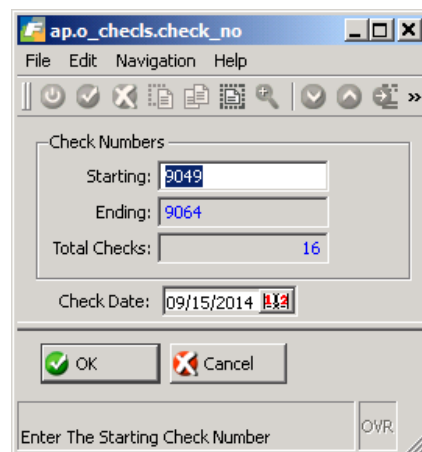
The screenshot shows a window titled "View Open Purchase Orders" with a menu bar (File, Edit, Navigation, Help) and a toolbar. Below is a table with the following data:

PO Number	Date	Item Code	Description	Stg	Open Quantity	UM	Cost	Total
181	09/10/2013	CPAPER811	COPY PAPER 8.5 X 11	POG	1.00	CS	5.0000	5.00
438	03/17/2014	HD-1000C	HEAD BUBBLE - CLEAR	POG	10.00	EA	875.9900	8759.90
440	03/17/2014	HD-1000C	HEAD BUBBLE - CLEAR	POG	20.00	EA	875.9900	17519.80
443	03/17/2014	HD-1000C	HEAD BUBBLE - CLEAR	POG	14.00	EA	875.9900	12263.86
447	03/17/2014	HD-1000C	HEAD BUBBLE - CLEAR	POG	50.00	EA	875.9900	43799.50
628	06/30/2014	CPAPER811	COPY PAPER 8.5 X 11	POG	2.00	CS	5.0000	10.00

An "OVR" button is located at the bottom right of the window.

FR2423 – AP Check Print

Now display the number of checks that will print, populate starting check number, and calculate last check number.



The screenshot shows a dialog box titled "ap.o_checks.check_no" with a menu bar (File, Edit, Navigation, Help) and a toolbar. It contains the following fields and buttons:

- Check Numbers** section:
 - Starting: 9049
 - Ending: 9064
 - Total Checks: 16
- Check Date: 09/15/2014
- OK and Cancel buttons.
- Footer text: "Enter The Starting Check Number" and an "OVR" button.

FR2424 – Positive Pay file creation

Positive Pay is an automated fraud detection tool offered by the Cash Management Department of most banks. When a check is presented to your bank and they do not have a “match” in the file created by Fitrix that you sent to them, you must first authorize payment before the bank will honor the check.

For more information go to <http://www.positivepay.net/>.

FR2530 – New buttons in Vendor Master to access email and website

Email:	<input type="text" value="jpelin@championinc.com"/>	 E-Mail
Web:	<input type="text" value="www.championinc.com"/>	 Web

FR3181 – Add payment detail to the Vendor Ledger report

Date: 09/17/2014		Vendor Ledger by Document Date						Page: 1		
Time: 17:59:56		ABC MANUFACTURING								
		Between 08/01/2014 and 09/17/2014								
		A/P Activity for 123457 - CHAMPION INC								
Date	QJ	DocNo	TType	Tran #	Description	Amount	Balance	DocNo	InvNo	Inv. Balance
Beginning Balance							.00			
08/19/2014	AP	192	DM	REFUND-080	FREIGHT OVERCHARGE	-125.00	-125.00			-125.00
09/03/2014	PU	193	IN	4234242342		5,500.00	5,375.00			5,500.00
09/03/2014	PU	194	IN	3123132		12.00	5,387.00			12.00
09/17/2014	CD	64	CD	50055		- 5.50	5,381.50	167	167	.00

FR3267 – New cash register report

This new report prints a running total for activity in the cash account for a date range the user specifies. It can replace any check and deposit register kept by hand.

Date: 09/18/2014 Time: 10:10:10		Cash Account Register ABC MANUFACTURING For the Period From 09/01/2014 to 09/18/2014 100000000 - 000 CASH ACCOUNT				Page: 1	
Date	Reference	Type	Payee/Paid By	Memo	Payment Amt	Receipt Amt	Balance
=====							
			BEGIN BALANCE				366,141.83
09/02/14	3123131	CASH RECEIPTS	2			3,354.60	369,496.43
09/03/14	4242342	CASH RECEIPTS	1			2,224.96	371,721.39
09/17/14	50055	CASH DISBURSEMENT	123457		5.50		371,715.89
			END BALANCE				371,715.89

FR3484 – Streamlined AP transaction processing

New control flags are in place that let the user determine if programs should be called automatically from another program instead of from the menu (i.e.- post the AP invoices as soon as the payable list is run instead of having to run the post program from the menu).

Streamline AP Setup				
Print Payable List:	Post AP Invoices:	<input checked="" type="checkbox"/>	Print Post Report:	<input type="checkbox"/>
Print AP Checks:	Post AP Checks:	<input checked="" type="checkbox"/>	Print Post Report:	<input type="checkbox"/>
Void Posted Checks:	Post Void Checks:	<input checked="" type="checkbox"/>	Print Edit/Post Report:	<input type="checkbox"/>

FR3636 – Merge Vendors

A new program was created that allows the user to create a new vendor record and then merge the historical activity and open items from an existing vendor and then delete that vendor. This is helpful when there a name change, merger or acquisition.

ACCOUNTS RECEIVABLE

FR2143 – Account statements by salesperson

Customer statements can now be run for specific sales people.

FR2177 – New fields for YTD and LTD sales in customer master

Two new fields for year-to-date and lifetime-to-date sales have been added to the customer master screen. The invoice posting program was modified to keep these totals up to date. The begin new year program clears out YTD sales at the beginning of each fiscal year.

FR2337 – Additional contact fields

A new screen has been added for additional contact information that is accessible from the Customer Master.

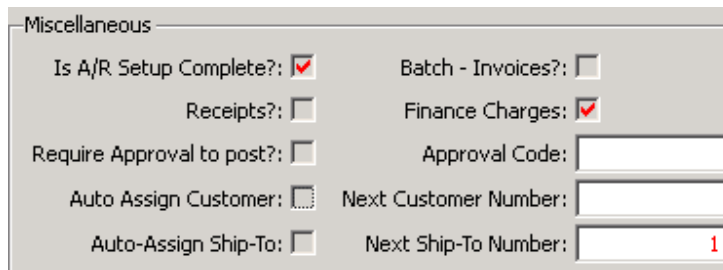
Contact	Telephone Number
JOHN M	404-567-4039
BETSY M	404-567-4040
JILL N	404-567-4045

OK Cancel

Enter label one OVR.

FR2348/2420 – Auto assign next customer number and ship-to code

User now has the option of letting the system assign the next sequential customer number or ship-to code.

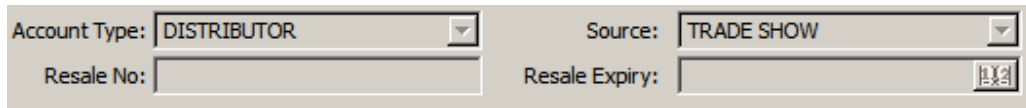


Miscellaneous

Is A/R Setup Complete?: <input checked="" type="checkbox"/>	Batch - Invoices?: <input type="checkbox"/>
Receipts?: <input type="checkbox"/>	Finance Charges: <input checked="" type="checkbox"/>
Require Approval to post?: <input type="checkbox"/>	Approval Code: <input type="text"/>
Auto Assign Customer: <input type="checkbox"/>	Next Customer Number: <input type="text"/>
Auto-Assign Ship-To: <input type="checkbox"/>	Next Ship-To Number: <input type="text" value="1"/>

FR2417 – Account type and account source fields


Two new fields added to the customer master to display account type and account source



Account Type: <input type="text" value="DISTRIBUTOR"/>	Source: <input type="text" value="TRADE SHOW"/>
Resale No: <input type="text"/>	Resale Expiry: <input type="text"/>

FR2522 – Variable commission rates at the Customer Master level



Clicking on the  icon displays this screen. Use this screen to enter variable commission rates that your salesperson will be paid that are based on the customer code and the item's price rather than the global commission rates stored with the item master record.

Update Customer Commission Rates

File Edit Navigation Help

Item Code Begin Price End Price Commission

12104	3.0000	3.5000	5.00
12104	3.5001	3.9900	5.50
12104	3.9901	4.5000	6.00

Quit

OVR

FR2530 – New buttons in Customer Master to access email and website

Email:  E-Mail

Web:  Web

FR2804 – Improvements to Cash Receipts screen program

Added new columns to display invoice date and original invoice balance.

Update A/R Cash Receipts

File Edit View Navigation Tools Actions Options Help

Batch Options

Find Prev Next Add Update Delete Browse

Customer

Date: 09/17/2014 Currency Code: STD

Customer: 1 Name: ACTION PARTS

Check Number: 4545666 Description:

Receipt Amount

Receipt Amount: 950.25

On Account: 0.00

Deposits: 0.00

Automatic Pay: N

Account

Posted: ☐

Cash Account: 100000000 - 000

Non-Attached: ☐

Cmd?	Invoice No.	Inv. Date	Orig. Amount	Due Date	Dsc. Date	Dsc. Bal	Balance Due	Discount	Applied
156	10/01/2013	950.25	10/31/2013	10/01/2013	0.00	950.25	0.00	0.00	
MEMO	12/30/2013	100.00	12/30/2013	12/30/2013	0.00	100.00	0.00	0.00	
275	12/30/2013	350.00	01/29/2014	12/30/2013	0.00	350.00	0.00	0.00	
525	06/02/2014	100.00	07/02/2014	06/02/2014	0.00	100.00	0.00	0.00	
526	06/02/2014	200.00	07/02/2014	06/02/2014	0.00	200.00	0.00	0.00	
547	06/12/2014	100.00	07/12/2014	06/12/2014	0.00	98.00	0.00	0.00	
551	06/12/2014	100.00	07/12/2014	06/12/2014	0.00	100.00	0.00	0.00	

Totals

Available: 950.25 Total Discount Allowed: 0.00

Total Applied: 0.00 Total Charge Backs/Write-offs: 0.00

FR3103 – AR Statement can be run retroactively

The Customer Statement of Account program can now be printed as of a past date, similar to how the AR aging report program works.

FR3104 – Process multiple cash receipts per invoice

The Update AR Cash Receipts program has been modified so that the user can apply multiple cash receipts to an invoice without having to post the transactions between receipts.

FR3180 – Add payment detail to the Customer Ledger report

Date: 09/17/2014 Time: 17:49:42		Customer Ledger ABC MANUFACTURING Between 01/01/2014 and 09/17/2014				Page: 1				
A/R Activity for BRAYSOM - BRAYSOM MPG										
Date	OJ	DocNo	TType	Tran #	Description	Amount	Balance	-----	Applied Against	-----
								DocNo	InvNo	Inv. Balance
Beginning Balance							.00			
01/15/2014	OE	279	IN	279		600.00	600.00			.00
01/16/2014	CR	45	CR	4422		-600.00	.00	279	279	.00
01/16/2014	OE	280	IN	280		1,995.00	1,995.00			1,995.00
01/16/2014	OE	281	CM	281		-50.00	1,945.00			-50.00

FR3484 – Streamlined AR transaction processing

New control flags are in place that let the user determine if programs should be called automatically from another program instead of from the menu (i.e. - print and post the AR invoices as soon as the receivable list is printed instead of having to run those programs from the menu).

Streamline AR Setup			
Print Receivable List:	Print Invoices: <input checked="" type="checkbox"/>	Post Invoices: <input checked="" type="checkbox"/>	Print Post Report: <input type="checkbox"/>
Print Cash Receipts:	Post Cash Receipts: <input checked="" type="checkbox"/>	Print Post Report: <input checked="" type="checkbox"/>	
Update Bank Deposits:	Print Bank Deposits: <input checked="" type="checkbox"/>	Post Bank Deposits: <input checked="" type="checkbox"/>	Print Post Report: <input checked="" type="checkbox"/>

BILL OF MATERIAL

FR1799 – Increase sequence Number size

FIXED ASSETS

FR3650 – Depreciation Activity Report

This new report can be run by Book, Asset Class Code or Department for a range of Accounting periods.

FR3654 – Net Book Value Report

New report that lists the net book value of each asset.

FR3654 – Net Book Value Report

Added the ability to depreciate an asset by a dollar amount each period versus a percentage of its value.

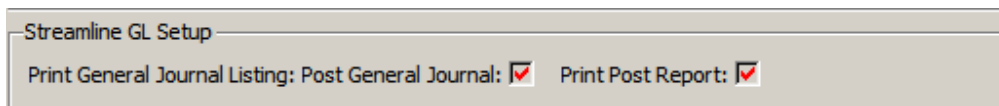
GENERAL LEDGER

FR2002 – User definable balance recalculation start date.

FR2920 – Suppress print of accounts with zero balance on GL activity reports

FR3484 – Streamlined GL transaction processing

New control flags are in place that let the user determine if programs should be called automatically from another program instead of from the menu (i.e. - post the journal entries as soon as the edit listing is printed instead of having to run the program from the menu).



Streamline GL Setup

Print General Journal Listing: Post General Journal: ☒ Print Post Report: ☒

INVENTORY CONTROL

FR2253 – Item lead time analysis

New report that prints average, last and previous lead times for items/warehouses selected.

FR2333 – Substitute or Alternate items

Store substitute or alternate inventory items that will then display in sales order entry should the item ordered not be available.

FR2343 – Restocking fee can now be a dollar amount or percentage

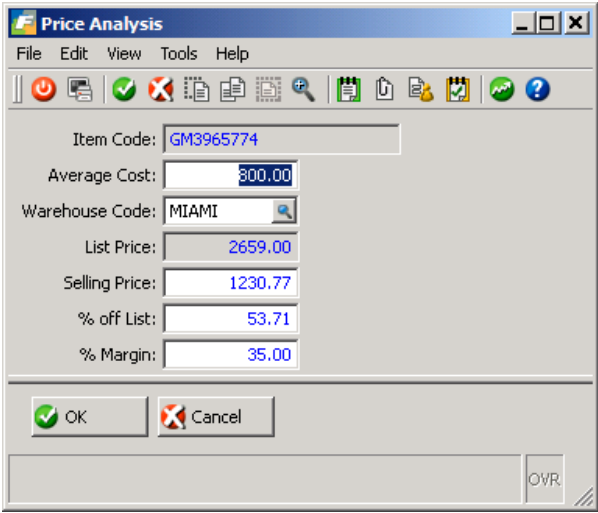
Restocking fee can now be a fixed dollar amount or a percentage of the value of merchandise being returned.

FR2335 – Special handling fees

New field added to the item master to store special handling fees. These fees are then automatically added to the customer invoice.

FR2341 – Test pricing screen

A new screen program has been added to the item master where user can enter a sales price to then see % off list and gross margin, % off list to see price to charge and gross margin, or gross margin to see price to charge and percent off list.



The image shows a software window titled "Price Analysis". It has a menu bar with "File", "Edit", "View", "Tools", and "Help". Below the menu is a toolbar with various icons. The main area contains several input fields with their corresponding values:

Field	Value
Item Code:	GM3965774
Average Cost:	800.00
Warehouse Code:	MIAMI
List Price:	2659.00
Selling Price:	1230.77
% off List:	53.71
% Margin:	35.00

At the bottom, there are "OK" and "Cancel" buttons. In the bottom right corner, there is a small button labeled "OVR".

FR2342 – New fields on Purchase/Sales History screen

Added fields to display total sold last twelve months, average sold per month and last sold date

View Purchase/Sales History

File Edit View Tools Help

Transaction History

Item: 12104 SCM A SERIES MULSTRIKE

Warehouse: MIAMI MIAMI WHSE

Total sold last 12 months: 786.000

Average sold per month: 65.500

Last sold date: 05/12/2014

Month	Year	\$ Cost	\$ Sales	Cost Qty	Sales Qty
9	2014	5511.000	0.000	1002.000	0.000
8	2014	11.000	0.000	2.000	0.000
7	2014	807.500	94.200	117.000	12.000
6	2014	1806.000	400.350	324.000	51.000
5	2014	50.000	158.500	10.000	18.000
4	2014	559.000	0.000	235.000	0.000
3	2014	2169.500	23.550	518.000	3.000
2	2014	323.040	7.850	117.000	1.000
1	2014	804.000	2558.500	117.000	83.000
12	2013	618.000	1000.000	102.000	1.000
11	2013	19104.500	4835.600	3161.000	616.000
10	2013	5950.300	7.850	980.000	1.000

(Ctrl-z) to View Transaction Detail

OK Cancel

OVR

FR2345 – Inventory labels

New inventory label that can be attached to items.

FR2349 – Lost sales notes

New note type added to record lost sales notes.

FR2352 – Sales projections

New screen program to set up and track sales projections and compare to actual.

Sales Projections Analysis

File Edit View Navigation Tools Actions Options Help

Monthly Detail

Find Prev Next Update

Period: 03 Year: 2014 Business Day: 21 Total Day: 21

Item Class	MTD Sales	Margin	% Margin	Projected Sales	Variance	MTD Purchases
ALARM	19.87	11.57	58.23	0.00	19.87	7.20
AUDIO	1875.00	1007.00	53.71	2000.00	-125.00	6193.00
COMP	2599.90	1789.90	68.84	10000.00	-7400.10	11742.00
PARTS	23.55	8.66	36.77	0.00	23.55	2165.00
PHOTO	15000.00	10500.00	70.00	5000.00	10000.00	22500.00
RBT	27999.50	18999.50	67.86	15000.00	12999.50	18697.00
TOTALS:	47517.82	32316.63	68.01	32000.00	15517.82	61304.20

16 of 25

View Detail

OVR

FR2378 – Display available quantity

Calculate and display the available quantity on the item status screen accessed from the Item Master and Sales Order Entry screen programs.

View Item Availability

File Edit Help

Item Code: 12104

Unit: EA

Factor: 1.000000

Warehouse	On Hand	Committed	On BKO	Available	On Req	On PO/Transfer
ATLANTA	817.000	419.000	0.000	398.00	0.000	5270.000
BALTIMORE	0.000	0.000	0.000	0.00	0.000	0.000
CHICAGO	0.000	0.000	0.000	0.00	0.000	0.000
EDM	6.000	0.000	0.000	6.00	0.000	0.000
MIAMI	12777.000	7329.000	0.000	5452.00	0.000	9222.000
NEBRASKA	0.000	0.000	0.000	5.00	0.000	5.000
SEATTLE	208.000	278.000	0.000	-49.00	100.000	3758.000

OK Cancel Detail

OVR

FR2408 – Cross sell items

Cross sell items are items that can go with another item being ordered. An example of a cross sell is that you may sell a computer monitor and there are other items that can be sold along with it. During order entry a list of these items will display so you can ask your customer if they also want to purchase these items.

Update Cross Sell Items

File Edit View Navigation Tools Actions Help

Find Prev Next Add Update Delete Browse

Item Code: C-MAC

Description: MAC LAPTOP NEW VERSION

Item Code	Description
C-MOUSE	MOUSE
C-USB	USB FLASH DRIVE

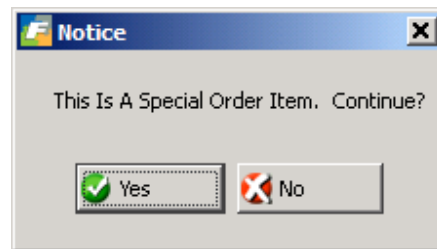
1 of 1

View Detail

OVR

FR2410 – Special order items

A new check box has been added to the item master to identify any items that are special order items not typically stocked. When a special order items is entered on the sales order entry screen the order taker will be notified.



FR2414 – Inventory audit report

New report that details the inventory movement (ins and outs) for warehouse, item and date range specified.

Date: 09/16/2014 Time: 08:49:05		ABC MANUFACTURING Item Audit Report Between 08/01/2014 and 08/31/2014			Page: 1	
Item Code	Description	Warehouse	Date	SJ	Quantity	Balance
=====						
C-MEMORY	2GB SO-DIMM 1047MHZ	MIAMI				
Beginning Balance						52.000
	Component Issue: MFG-513		08/06/2014	CI	-4.000	48.000
	Purchase Order: 752		08/06/2014	PU	50.000	98.000
	Production Receipt: MFG-515		08/06/2014	PR	-4.000	94.000
	Undefined IC-SH: 128		08/19/2014	SH	-94.000	0.000
	Purchase Order: 777		08/25/2014	PU	18.000	18.000
	Production Receipt: MFG-547		08/25/2014	PR	-2.000	16.000
	IC Receipt: TEST		08/26/2014	PU	500.000	516.000
	Production Receipt: MFG-555		08/26/2014	PR	-6.000	510.000
Ending Balance						510.000

FR2426 – Multiple units of measure

Assigning a unit of measure list to an item allows you to purchase and sell an item in multiple units of measure. For example, you may purchase an item in both boxes and cases.

Update UOM Lists

File Edit View Navigation Tools Actions Help

Find Prev Next Add Update Delete Browse

UOM List Code: COMPONENTS Description: COMPUTER PARTS

Stock Unit: EA EACHES

Quantity of Decimal Places: 2

UOM	RefUnit	Quantity	Type
EA	EACHES	1.000	None
BX	EACHES	12.000	Sell
CS	BOX	20.000	Both

1 of 1

View Detail

OVR

FR2433 – Print item specific notes

In the item master you now enter item specific notes and designate which documents they should print on.

Choose a Note Topic

File Edit Help

Picking Notes
Print Notes
Item Notes
Lost Sale Notes

OK Cancel

OVR

Select Document(s)

File Edit Navigation Help

Document Name	Print?
Sales Quotations	<input type="checkbox"/>
Order Acknowledgements	<input type="checkbox"/>
Picking Documents	<input checked="" type="checkbox"/>
Packing Slips	<input checked="" type="checkbox"/>
Invoices And Memos	<input type="checkbox"/>
Purchase Orders	<input type="checkbox"/>

OK Cancel

Check If The Notes Should Print On This Document

OVR

FR2566 – Item shortage by warehouse inquiry screen

Item Shortage by Warehouse							
File Edit View Navigation Tools Actions Options Help							
<div> <div> <div>+</div> <div>+</div> </div> <div>Committed On Order</div> </div> <div> <div>Find</div> <div>Prev</div> <div>Next</div> <div>Browse</div> </div>							
Warehouse: <input type="text" value="MIAMI"/> <input type="text" value="MIAMI WHSE"/>							
Item Code	Description	U/M	On Hand	Commit/Bko	Net Available	On Order	
12104-3433	SCM A SERIES MULSTRIKE	EA	0.00	2.00	-2.00	0.00	
12104-BLO	SCM A SERIES MULSTRIKE	EA	0.00	1.00	-1.00	0.00	
12112-1001	SCM A SERIES CVR-UP TAPE	EA	100.00	110.00	-10.00	0.00	
12112-3444	SCM A SERIES CVR-UP TAPE	EA	86.00	90.00	-4.00	81.00	
12112-BLO	SCM A SERIES CVR-UP TAPE	EA	0.00	100.00	-100.00	0.00	
12138	SCM A SERIES CORR (PK/2)	BX	0.00	1.00	-1.00	3.00	
12195	SCM A SERIES NYLON	EA	110.00	151.00	-41.00	330.00	
5 of 7							
<div>View Detail</div> <div>OVR</div>							

FR2932 – New Serial/Lot Composition screen

Allows user to see the serial/lot #s of the component that went into the end item produced.

Serial/Lot Composition Inquiry							
File Edit View Navigation Tools Actions Help							
<div> <div>Find</div> <div>Prev</div> <div>Next</div> <div>Browse</div> </div>							
Item Code: <input type="text" value="C-BOARD-ASSY"/>		Production Order: <input type="text" value="515"/>		Release: <input type="text" value="000"/>			
<input type="text" value="COMPLETED MOTHER BOARD - C-MAC"/>		Receipt Date: <input type="text" value="08/06/2014"/>					
Warehouse: <input type="text" value="MIAMI"/>		Sales Order: <input type="text"/>					
Serial No: <input type="text" value="ABC2851"/>		Line: <input type="text"/>					
Lot No: <input type="text"/>		Customer: <input type="text"/>					
On Hand: <input type="text" value="0.000"/>		Date Sold: <input type="text"/>					
Sequence	Component	Description	Quantity Issued	Date Issued	Warehouse	Serial Number	Lot Number
1	C-BOARD	MOTHER BOARD - C-MAC-RJ	1.000		MIAMI	ABC2849	
11	C-MEMORY		4.000		MIAMI		201408A
2	C-PROCESSOR	INTEL CORE I5 PROCESSOR	1.000		MIAMI	ABC2850	
189 of 259							
<div>View Detail</div> <div>OVR</div>							

FR2932 – New serial master screen

This screen program displays information for serialized or lot controlled items that are purchased or produced for historical tracking.

Update Serial Master	
File Edit View Navigation Tools Actions Help	
Find Prev Next Update Browse	
General Item Code: RB-B9 Warehouse: MIAMI Serial No: ABC2148 Lot No: Bin Location: A1	Warranty Warranty Expiration: 06/04/2015 Warranty Material Cost: Warranty Labor Cost: Warranty Sales Amount:
Purchase Order PO Number: Receipt Date: Vendor: MFG-414 Manf Serial No: RB302910 Unit Cost: 7198.5000 Received Qty: 1.000	Sales Order Sales Order: Line: Customer: Shipped Date: Shipped Qty:
Production Order Production Order: 414 Release: 000 Produced Date: Eng Revision Level: A	

FR3020 – Update list price by item class

The Class field on the Update Inventory Pricing screen is now a query field so all items in a specific class can have their prices globally updated.

FR3051 – Display committed quantity in the Inventory Adjustment program

If you happen to enter a quantity adjustment that will now prevent you from fulfilling all of the outstanding commitments you have on sales orders this screen will display so you know which sales orders are outstanding. Once the adjustment is posted you will need to adjust the quantity on the sales order accordingly

Customer	Sales Order No	Stage	Quantity
12	1617	ORD	5.0000
12	1687	ORD	1.0000

QUANTITY ORDERED EXCEEDS AVAILABLE QUANTITY IN WAREHOUSE. ADJUST SALES ORDERS ACCORDINGLY

FR3185 – Changes to the Reorder Advice report

When this report is printed the user is now prompted as follows so that they do not have to also see items where there is already enough on PO to meet the reorder point. The report now also prints the quantity on the PO.

Skip Items Where Reorder Point is < Quantity Already Ordered?

Yes No

FR3273 – Indented BOM report can now be exported to Excel

FR3343 – Product warranty tracking

Manufactured or purchased items can now have the number of warranty days defined at the item level. When sold the warranty expiration date is calculated and stored with the shipment history information. Any warranty work done in house (for either exchange or repair) is managed by warranty work orders that are tied back to the original sales order.

FR3347 – Quality Control

Product quality can be recorded at one or more of these checkpoints:

- At the time of purchase receipt
- When material is consumed by production

- When labor transactions are reported
- When a produced item is received into inventory
- As a completely separate transaction independent of all these functions

When the quality transaction is recorded and saved, an email alert is sent to the buyer for purchased items or to the production manager for manufactured items notifying them of the corrective action needed. When the corrective action has been completed and the QC transaction closed, email alerts are again sent out notifying these managers that the quality issue has been resolved.

The quality analysis report will show you for any time period what items had quality issues, what the issues were and the percent pass versus fail. By analyzing this data you can determine if there is a particular vendor that is selling you defective product or maybe there is a problem with a resource in your production cycle.

The screenshot displays the 'Quality Transactions' software window. It features a menu bar (File, Edit, View, Navigation, Tools, Actions, Help) and a toolbar with icons for Find, Prev, Next, Add, Update, Delete, and Browse. The main interface is divided into several sections:

- Item and Measurement:** Includes fields for Item (C-MAC), Warehouse (MIAMI), Measurement (SECTOR TEST), Collection Type (Production Component), and Collection Stage (Component Issue).
- Production/Purchase Order:** Includes fields for Order (528), Release (000), Line, Routing Sequence, Component Sequence (4), and Component Item (C-DISK).
- General Information:** Includes Date (08/13/2014), Quantity Tested (3.000), Quantity Passed (2.000), Quantity Failed (1.000), Reason (DISK), and Quantity Warning (0.000). A comment field contains 'BAD SECTORS > ALLOWANCE OF 3'.
- Optional References:** Includes a Measured field (5) and Low/High Tolerance fields.
- Order Hold Information:** Includes a checkbox for 'Hold Order?' and a 'Hold Code' field (HLD).
- Corrective Action:** Includes fields for CAR Code (REFMT), Number (77), Entered By (bettyb), Completed, and Comment.
- User-Defined Data:** Includes three empty fields for User Field 1, User Field 2, and User Field 3.
- Serial/Lot Details:** A table at the bottom showing transaction data.

Serial No	Lot No	Pass Qty	Fail Qty	Reason	Warning Qty	Reason	Measured	Comment	User 1	User 2
AC23554645645		2.000	1.000	DISK						

FR3389 – Reference designators for bills of material

Reference designators provide Electronics manufacturers a clear indication of component placement on an electrical schematic or printed circuit board. They are especially useful when multiple usages of a component are called for.

Sequence: 1

Component: C-BOARD

Quantity: 1

Designator:

- F18
- R12

OK Cancel

Enter the reference designator OVR

FR3421 – Update of vendor code in item master

If the primary vendor in the catalog is changed this will change the vendor code found in the item master.

FR3428 – Display of bin location screen in Component Issue program

This screen no longer displays for non lot/serial items if there is enough of the component in its primary bin location.

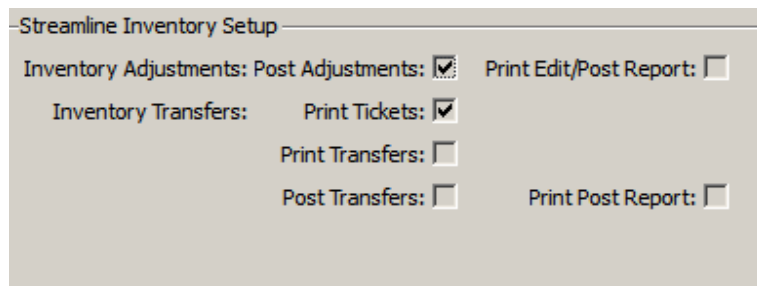
Location	Lot	Serial	Expire Date	On Hand	Issued	This Issue	Parent Serial	Parent Lot
A1		ABC2231		1.000		1.000		
A1		ABC2232		1.000		.000		
A1		ABC2233		1.000		.000		
A1		ABC2234		1.000		.000		
A1		ABC2235		1.000		.000		
A1		ABC2236		1.000		.000		
A1		ABC2237		1.000		.000		
A1		ABC2238		1.000		.000		
A1		ABC2239		1.000		.000		

OK Cancel

Enter the parent item's serial number being produced OVR

FR3484 – Streamlined inventory transaction processing

New control flags are in place that let the user determine if programs should be called automatically from another program instead of from the menu (i.e. – post the adjustment as soon as the transaction is saved instead of having to run the edit and post programs from the menu).



The screenshot shows a dialog box titled "Streamline Inventory Setup". It contains several checkboxes for configuring inventory transactions. The "Post Adjustments" checkbox is checked, while "Print Edit/Post Report" is unchecked. Under the "Inventory Transfers" section, "Print Tickets" is checked, "Print Transfers" is unchecked, and "Post Transfers" is unchecked. The "Print Post Report" checkbox is also unchecked.

Option	Status
Inventory Adjustments: Post Adjustments:	<input checked="" type="checkbox"/>
Print Edit/Post Report:	<input type="checkbox"/>
Inventory Transfers: Print Tickets:	<input checked="" type="checkbox"/>
Print Transfers:	<input type="checkbox"/>
Post Transfers:	<input type="checkbox"/>
Print Post Report:	<input type="checkbox"/>

FR3447 – Production line inventory

This feature adds the ability to:

- Define a new production resource called a 'Production Line'
- Define a new issue function called 'Component Issue by Routing' – issues only components used at a selected routing step (operation)
- Associate a manufactured item and its Bill of Material with one or more Production Lines
- For each Item/Production Line combination, identify the bin locations from which to issue material during a 'component issue by routing'
- Create a new report 'Production Line Inventory' – lists on hand balances for bin locations associated with a Production Line

Item/Production Line

File Edit View Navigation Tools Actions Options Help

Quit Print OK Cancel Cut Copy Paste Zoom Notes Attachments U Fields To Do View Detail Next Page Previous Page Insert Row Delete Row

Copy Bill

Find Prev Next Add Update Delete Browse

Item: C-MAC Description: MAC LAPTOP NEW VERSION

Warehouse: MIAMI

Production Line: TEST Description: TEST PRD LINE

Putaway Location: A1

Routing Seq	Comp Seq	Component Item	Component Description	Line Location	Min Quantity	Max Quantity
2	1	C-MON	24" MONITOR	D1	10.000	500.000
2	2	C-KEY	KEYBOARD	D1	10.000	500.000
2	3	C-MOUSE	MOUSE	D1	10.000	500.000
1	4	C-DISK	HARD DRIVE	D1	10.000	500.000

8 of 12

View Detail

QVR

FR3541 – Indented serial/lot composition

This screen shows indented serial and lot number traceability for a produced/sold product. Enter an item code and a serial or lot number and an indented list of serial and/or lot numbers that were purchased or produced to make the item is displayed.

Indented Serial Composition

File Edit View Navigation Tools Actions Help

Find Prev Next Browse

Item Code: C-MAC-RJ MAC LAPTOP - WITH BOARD DETAIL Serialized: Serialized

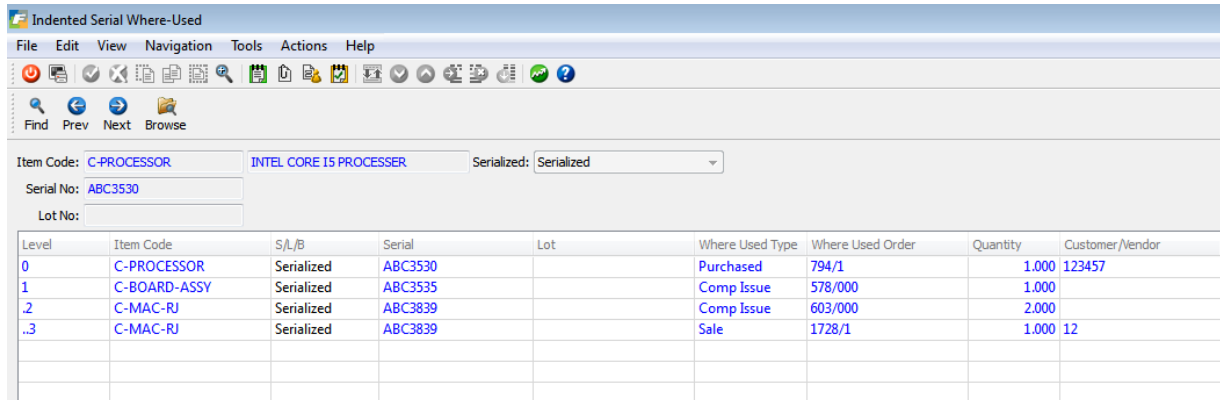
Serial No: ABC3839

Lot No:

Level	Item Code	S/L/B	Serial	Lot	Source Type	Source Order	Quantity	Vendor
1	C-MAC-RJ	Serialized	ABC3839		Production	603/000	1.000	
2	C-MON	Serialized	ABC2233		Purchase	606/1	1.000	123458
2	C-BOARD-ASSY	Serialized	ABC3535		Production	578/000	1.000	
3	C-PROCESSOR	Serialized	ABC3530		Purchase	794/1	1.000	123457

FR3542 – Indented serial/lot where used

This screen shows indented serial and lot number where used traceability for a produced/sold product. Enter an item code and a serial or lot number and an indented list of serial and/or lot numbers that consumed the entered serial/lot is displayed.



Level	Item Code	S/L/B	Serial	Lot	Where Used Type	Where Used Order	Quantity	Customer/Vendor
0	C-PROCESSOR	Serialized	ABC3530		Purchased	794/1	1.000	123457
1	C-BOARD-ASSY	Serialized	ABC3535		Comp Issue	578/000	1.000	
2	C-MAC-RJ	Serialized	ABC3839		Comp Issue	603/000	2.000	
..3	C-MAC-RJ	Serialized	ABC3839		Sale	1728/1	1.000	12

MATERIAL PLANNING

FR2478 – Allow planned orders from MRP generation to be added to existing purchase orders

ORDER ENTRY

FR2242- Order type display

A new field has been added to the Update Invoices/Memos program to display order type.

FR2332/3182 – Streamlined order entry process

New control flags are in place that let the user determine if programs should be called automatically from another program instead of from the menu (i.e.- print the pick ticket as soon as the sales order is saved)

Streamline Order Entry Setup			
Update Customer Orders:	Print Pick Ticket: <input checked="" type="checkbox"/>	Create Invoice: <input type="checkbox"/>	
Update Picked Quantities:	Print Packing List: <input type="checkbox"/>	Create Invoice: <input type="checkbox"/>	
Update Shipped Quantities:	Print Packing List: <input type="checkbox"/>	Create Invoice: <input type="checkbox"/>	
Create Auto Invoices:	Print Invoice: <input checked="" type="checkbox"/>	Post Invoice: <input checked="" type="checkbox"/>	Print Post Report: <input type="checkbox"/>
Update Invoices/Memos:	Print Invoice: <input checked="" type="checkbox"/>	Post Invoice: <input checked="" type="checkbox"/>	Print Post Report: <input type="checkbox"/>
Update Backorder Released:	Print Pick Ticket: <input checked="" type="checkbox"/>		

FR2339 – Payment summary on sales register

A summary of customer payment methods offered on invoices (cash, on account, credit card) prints at the end of the Daily Sales Register report

FR2351 – Export invoice to Excel

The ability to export a customer invoice to Excel is now an option in the Print Invoices and Memos print routine.

FR2353 – Print lot number on product detail report

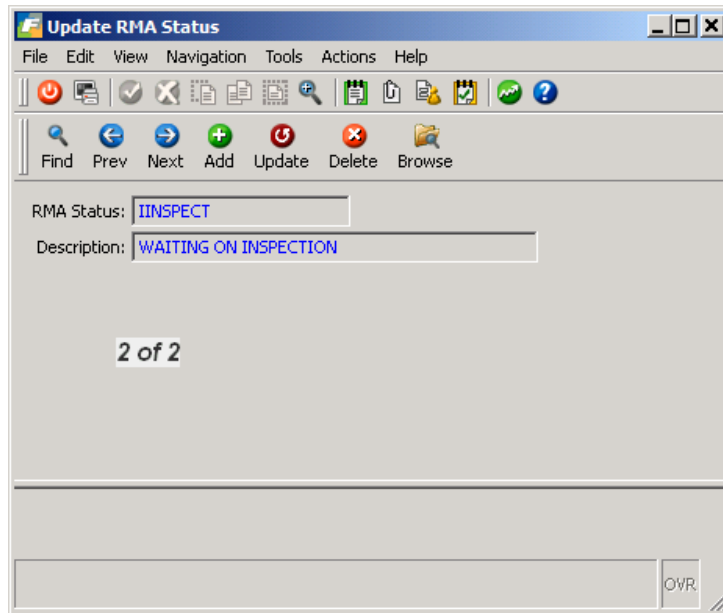
Print lot number information on the Product Detail Sales History report.

FR2362 – Interface with Federal Express and UPS

Real-time interface to Federal Express Ship Manager and UPS Worldship databases via web services. You can see real time what the estimated freight charges will be, automatically send tracking numbers to your customers, track the shipments progress from the order entry screen, and get actual freight charges automatically added to the customer invoice at time of invoice.

FR2394 – RMA status code

New RMA status code so that user can search for open RMAs based on status (i.e. - Waiting on approval, etc).



FR2406 – Item count on pick ticket

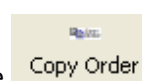
Print the quantity count on the pick ticket and also display a real-time calculation of this on the Update Picked Quantities screen so user can confirm there has been no changes made to the order after the pick ticket was printed.

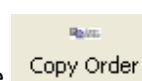
FR2407 – Salesperson contact information

Salesperson's name and telephone number now print on customer invoices if the "Print contact information on invoice" check box on the Update Salesperson Definition screen program is checked.

FR2409 – Clone sales orders

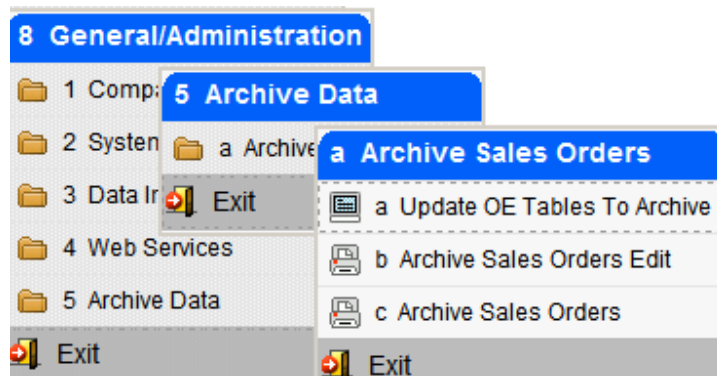
You use the clone sales order feature to copy existing sales orders. When you clone a sales order a new sales order is created with an order type of QUO for quote. You can make changes to the new quote or accept it as is. When you are satisfied with the quote, you update the quote and change the order type to REG for regular sales order so that the order can be processed. You use the Update Customer Orders program to clone sales orders. You can also copy quotes using the Sales Quotations/Recurring Orders program.



Select the order you want to copy and then click on the  icon on the toolbar.

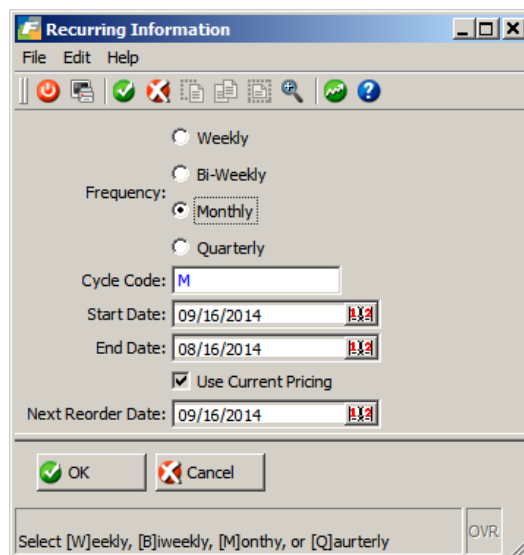
FR2411 – Sales Order Archive

With sales order archiving you can store older orders in archive tables. This strategy can speed up sales order processing for operations with a large number of orders. Archived orders are searchable and available using the Order History option from within the Update Customer Orders program.



FR2412 – Recurring or standing sales orders

Recurring or standing orders are orders that the customer will place with you on an ongoing basis and specific time intervals. In Fitrix you can create an order template for this type of order and then use the copy recurring order function to create orders from it at time intervals you select.



FR2413 – Split payment terms

Split payment terms allow you to split payment of invoices into multiple payments. The split payments are created when you run Post Order Entry Documents. When an order entry invoice is split into multiple payments the original AR open item is created, the open item is then credited by an attached credit memo and new AR open items are created for the split payments. A separate AR open item is created for each split payment with the customer term you defined for the split payments.

Terms Code	Description	Due Days	Disc Days	Disc Pct	Split Percentage
N30	NET 30 DAYS	30	0	0.00	33.00
N60	NET 60 DAYS	60	0	0.00	33.00
N90	NET 90 DAYS	90	0	0.00	34.00

FR2422 - On the fly ship-to addresses

Fields have been added to the order entry screen so that ship-to addresses can be entered and/or edited, stored with the sales order and then printed on the customer invoice.

Customer	
Phone:	219 993 5436
Type:	REG
Customer:	12
Name:	CLASSIC PARTS UNLIMITED
Customer PO:	312312
Sales Person:	BJ BILL JOHNSON
Ship To	
Ship To:	SHIPTO
Ship/Return Date:	03/24/2014
Ship Type:	OTHER
Required Date:	
Ship Via:	YELLOW
Name:	CLASSIC PARTS UNLIMITED
Address:	22501 72ND UNIT 1A
	CLEARING POINT BUSINESS CENTER
City:	SOUTH BED
State:	TX
Zip:	77060
Country:	US UNITED STATES

FR2421- Print invoices for specific ship-to location

Added the ability to reprint invoices for a specific ship-to location.

Enter Selection Criteria

File

Invoice Selection Criteria

Customer:

Ship-To:

Order Number:

Invoice Date:

Invoice Number:

Invoice Doc. No.:

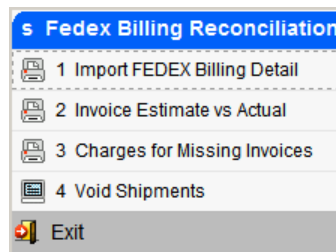
OK Cancel

Enter customer code. OVR

FR2425 - Federal Express billing import and cost analysis

Cost analysis report shows comparisons between estimated Fedex shipping charges and actual Fedex invoiced charges for shipments. The estimated and actual charges are cross-referenced by Fitrix sales Invoice number, and FEDEX Tracking number. This enhancement includes the following new functions:

- Import Fedex Invoice billing details into a Fitrix table
- Print report comparing Shipment estimated costs to actual costs from Fedex Billing
 - Report is requested for a range of invoice dates
- Print report listing Fedex Billing details for Invoices which do not exist in the Fitrix Order Entry Invoicing tables.



FR2483/2570 – Additional information on the purchase/sales history drilldown screen

Display purchase order number and customer invoice number on the purchase/sales history screen instead of system assigned document numbers. The vendor and customer business name has also been included.

View Purchase/Sales History

File Edit Navigation Help

Document	Date	OJ	Type	Who	Business Name	Qty	Stats	Unit Cost	Unit
655	07/01/2014	PR	PU	123457	CHAMPION INC	111.00	Y	5.5000	
729	07/16/2014	PR	PU	123457	CHAMPION INC	1.00	Y	5.5000	
735	07/24/2014	PR	PU	123457	CHAMPION INC	1.00	Y	5.5000	
741	07/30/2014	PR	PU	123457	CHAMPION INC	1.00	Y	5.5000	
742	07/30/2014	PR	PU	123457	CHAMPION INC	1.00	Y	5.5000	
609	07/10/2014	OE	SH	12	CLASSIC PARTS UNLIMITED	-10.00	Y	4.9436	
610	07/10/2014	OE	SH	12	CLASSIC PARTS UNLIMITED	5.00	N	5.5000	
610	07/10/2014	OE	SH	12	CLASSIC PARTS UNLIMITED	5.00	N	4.9436	
610	07/10/2014	OE	SH	12	CLASSIC PARTS UNLIMITED	3.00	N	5.5000	
610	07/10/2014	OE	SH	12	CLASSIC PARTS UNLIMITED	3.00	N	4.9436	
608	07/10/2014	OE	SH	12	CLASSIC PARTS UNLIMITED	-10.00	Y	4.9436	
610	07/10/2014	OE	SH	12	CLASSIC PARTS UNLIMITED	2.00	N	5.5000	

OK Cancel

OVR

FR2494- Salesperson default logic

If you enter the salesperson's Linux login here the salesperson assigned to the sales order will be based on the login of the person entering the order instead of entering a salesperson at the customer or ship-to level (if it is entered there this login id/salesperson combination will not be used so make sure they are null).

Update Salesperson Definitions

File Edit View Navigation Tools Actions Options Help

CommRed

Find Prev Next Add Update Delete Browse

Salesperson Code:

Salesperson Name:

Print contact information on invoice: ☐

Phone:

Cell:

Fax:

Login:

Email:

4 of 5

OVR

FR2514- Search by phone number

A field for phone number has been added to the sales order entry screen so that you can now find the customer by phone number.

The screenshot shows the 'Customer' section of a software interface. It includes a toolbar with icons for Find, Prev, Next, Add, Update, Delete, Browse, and Options. Below the toolbar, there are several input fields: 'Phone:' with the value '404', 'Type:' with the value 'REG', 'Customer:', 'Name:', 'Customer PO:', and 'Sales Person:'. Each field has a search icon to its right.

FR2523- Reduce commission paid on invoices paid late

This program allows you to set up commission reduction levels if you pay your salespeople once your customer pays you. Using the example below, Tom Mack will have the commission percentage he's paid reduced by 2% for any invoices that are paid 30-59 past terms and 4% for any invoices that are paid 60 days or more past terms.

The screenshot shows the 'Commission Reduction' window. It has a menu bar (File, Edit, View, Navigation, Tools, Actions, Help) and a toolbar with icons for Find, Prev, Next, Add, Update, Delete, and Browse. Below the toolbar, there is a 'Salesperson:' field with the value 'TM' and a dropdown menu showing 'TOM MACK'. Below this is a table with two columns: 'Days Past Due' and 'Reduction %'. The table contains two rows of data: 30 days past due with a 2.00% reduction, and 60 days past due with a 4.00% reduction. Below the table, there is a status bar showing '1 of 1'. At the bottom, there are buttons for 'OK', 'Cancel', and 'Header'. A text field at the bottom left says 'Enter days past due.' and a button labeled 'OVR' is at the bottom right.

Days Past Due	Reduction %
30	2.00
60	4.00

FR2565- Added business name and salesperson code to backorder release screen

The screenshot shows the 'Update Backorder Received' window. It has a menu bar (File, Edit, View, Navigation, Tools, Actions, Help) and a toolbar with various icons. Below the toolbar is an 'Add' button. The main area contains several input fields: 'Warehouse' (MIAMI), 'Item Code' (C-BOARD), 'Qty. To Release' (10.000), 'Available' (20.000), and 'Units' (EA). There is also an 'Auto Release?' checkbox. Below these fields is a table with the following data:

Rel YNP	Order No	Customer Business Name	Salesperson	B/O Date	Qty To Release	Qty Backordered
	1699	ACTION PARTS	BJ	09/16/2014		10.000

Below the table are 'Totals' fields showing 0.000 and 10.000. At the bottom are 'OK', 'Cancel', and 'Detail' buttons. A status bar at the very bottom says 'Enter [Y] to automatically release the amount received.' and has an 'OVR' button.

FR2619 - Added customer business name to the salesperson Sales History reports

FR2630 – Any freeform notes attached to ship-to will display in sales order entry

FR2631 – Set default warehouse at customer and ship-to level

User can now define the warehouse sales order should default to at the customer and/or ship-to level

Customer Master OE Infor screen

Enter Order Entry Information

File Edit View Tools Help

Order Entry Information

Discount/Customer Class:

Salesperson: BJ BILL JOHNSON

Pay.Method: AR

Ship Terms: PREPAID FREIGHT PREPAID- DO NOT INVOICE BUYER

Split Payment Terms:

UPS Account:

Route Code:

Residential: N

Ship Complete: N

Discount Level:

Warehouse Code: MIAMI MIAMI WHSE

OK Cancel

Enter the discount code for this customer. OVR

Ship-to Information screen

Update Ship To Information

File Edit View Navigation Tools Actions Help

Find Prev Next Add Update Delete Browse

Customer Code: 1 ACTION PARTS

Ship-To: CALIF

Ship-To Name: ACTION AUTOPARTS WEST

Residential:

Route Code:

Freight Tax Code:

Tax Code: NOTAX

Misc Tax Code:

Contact: STAN WAVERLY

Phone: 805 885 5487

Address 1: 4389 APPLE WAY

Address 2:

City, State, Zip: VENTURA CA 93003

Country: US UNITED STATES

Salesperson: BJ BILL JOHNSON

Discount: CASH CASH DISCOUNT

Ship Terms:

Cell:

Fax:

Email:

Web Address:

Warehouse Code: SEATTLE SEATTLE DISTRIBUTION CENTER

1 of 3

OVR

FR2719 – Access to alternate items

See FR2333. A button to access alternate items has been added to the toolbar in the Update Customer Orders program. It is accessible from any line items with stage = BKO that have alternate items set up.

FR2722 – Flip warehouse

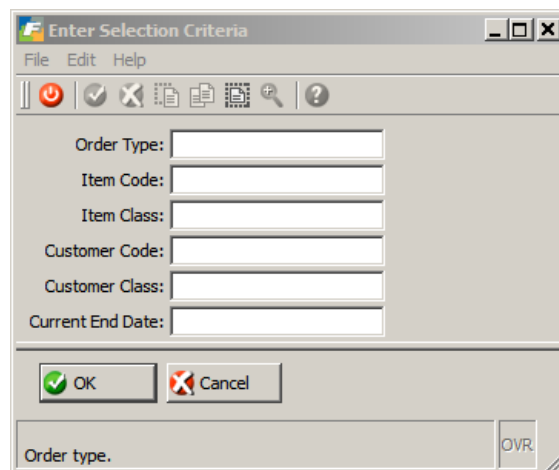
Changing the warehouse code in the sales order header screen will now update the warehouse code for all line items and recalculate product availability.

FR2750 – Add gross margin percent to all sales order history reports

FR2794 – Update end date for special pricing

New program to extend the end date for customer pricing each year

FR2841 – Can now cancel customer invoice after it's been printed



FR2874 – New screen program to add vendors and costs to drop ship PO types

If any items on the sales order are not in the vendor catalog a new screen will display so that user can designate the vendor and enter the cost prior to the drop ship PO being created.

FR3021 – Default Customer PO number on RMA

If the RMA is attached to the original sales order, the customer PO number on the RMA will be set to the customer PO # on the original sales order.


FR3033 – Change to Pick Ticket print when quantities or items change

If you add items to the order, release a backorder, or revise quantities and then reprint the picking ticket the word REPRINT will print on the ticket and the sales order number that prints will be appended with a -2 (or with whatever the last reprint # was incremented by 1). All items will print on the ticket again (not just the changes as in previous versions of Fitrix), and an * will print to the right of the quantity of the item added/changed so that it will be easy for the order pickers to know what changed since the last print.

FR3097 – Variable customer pricing by warehouse location

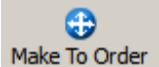
Add warehouse location to the Update Special Price Defaults screen program so that product pricing can vary by warehouse.

FR3098 – Expanded description and contact fields on sales order header



The screenshot shows a software window titled "Order". Inside, there are two input fields: "Contact Name:" with the value "JOHN MACY" and "Contact Phone:" with the value "404-567-0398". Below these is a label "Extended Description" followed by a text area containing the text "SHIPMENT FOR HOLIDAY SALE". The text area has a vertical scrollbar on the right side.

FR3100 – Access to production orders from sales order entry

Add a  button to the sales order entry item so that user can easily access the production work order for make to order items.

FR3190 – Print production work order number on customer invoice

FR3235 – Blanket sales orders

A blanket sales order is used when a customer has committed to buying large quantities at a set price over a given period of time and usually has an expiration date associated with it. When a release order is created from the blanket order the sales order number assigned to it is the blanket order number followed by the release number (i.e. -1 for the first release, -2 for the second release, etc.). This allows you to easily see all shipments generated from the blanket order. The Blanket Order Release Status report is used to identify upcoming releases that need to be processed. The Blanket Order Expiration report is used to identify active blanket orders that are about to expire.

Release Quantity	Balance Remaining	Ship Date	Required Date
250.000	0.000	12/26/2013	01/08/2014
250.000	250.000	03/01/2014	03/16/2014
250.000	250.000	06/01/2014	06/15/2014
250.000	250.000	09/01/2014	09/15/2014

FR3270 – New open sales orders report (exported to Excel)

Prints the detail for sales orders not yet closed (open and closed) entered for the year

A	B	C	D	E	F	G	H	I	J	K	L	M	N	O	P
Sales Order	Order Date	Due Date	Deliver Date	Invoiced Date	Customer	Customer Name	WO No	Sales Rep	Description	Type	Low Stage	Material Issued	Labor Reported	Production Complete	Invoice Status
1036	3/17/2014	3/23/2014			1	ACTION PARTS		TM		REG	ORD	No	No	No	No
1037	3/17/2014	3/23/2014			1	ACTION PARTS		TM		REG	ORD	No	No	No	No
1379	6/5/2014	6/5/2014			1	ACTION PARTS		TM		REG	ORD	No	No	No	No
1380	6/5/2014	6/5/2014			1	ACTION PARTS		TM		REG	ORD	No	No	No	No

FR3300 – Available to promise date (ATP)

When entering sales orders for any product that is not available, the date it will be available will display on the backorder screen so that the customer can be informed.

The image shows two screenshots from a software application. The top screenshot is a 'Message' dialog box with a blue title bar. It contains a text area with the message: 'Quantity can be filled by an open production order(s) due 06/05/2014' and 'You should contact Production to confirm this due date'. Below the text area is a 'Close' button with a red 'X' icon. The bottom screenshot is the 'Back Order Information' window. It has a menu bar with 'File', 'Edit', and 'Help'. Below the menu bar is a toolbar with icons for back, forward, and other functions. The main area is titled 'Backorder Information' and contains several fields: 'Line Number' (1), 'Item Code' (FR3000MFG), 'Warehouse' (MIAMI), 'Allow Backorder?' (Y), 'Qty. Ordered' (3.0000), 'Qty. Available' (0.0000), 'Avg. Lead Time' (0 days), 'Last Lead Time' (0 days), 'Prior Lead Time' (0 days), 'Qty. To Order' (0.0000), 'Qty. To Back Order' (3.0000), and 'Backorder Available Date' (06/05/2014). At the bottom of the window are buttons for 'OK', 'Cancel', 'Alternate Items', 'New BKO Date', and 'Availability'. There is also an 'OVR' button in the bottom right corner.

FR3323 – Customer credit management

The customer master screen now shows the information you need to see about the customer's credit status all in one place. It shows payment terms, credit limit, aged balance due amounts and total open orders, available credit, average days to pay, % over limit allowed and who to contact if there are issues with collections.

When a sales order is entered and the customer is designated as one that should go through the credit check process, a real time calculation is made that compares the customer's credit limit to their current balance due plus any outstanding orders not yet shipped. If the credit limit is exceeded the order is placed on hold. When a sales order goes on hold, a copy of the order acknowledgement is emailed to the both credit manager and the sales representative assigned to the account notifying them that the order is on hold.

When the order is released from hold an email is sent to both the person that entered the order and the customer's sales representative so the customer can be contacted and given an estimated delivery date. If a production work was created from the sales order (Make to order) and it was also placed on hold, the person responsible for processing the work order receives an email so they know that they can proceed with production.

Just because an order has been placed on credit hold doesn't mean that everything should stop, especially if you're a make to order manufacturer. If you have a lengthy production processing time you may still want to produce the item so you have it available to ship when the order is taken off hold. In Fitrix this is controlled via Hold Codes you define and assign to the customers that determine which processes should be allowed to continue while the orders is on hold.

Additional Information		Account Balances	
Active: <input checked="" type="checkbox"/>	Date Established: 01/30/2014	Inactive Date:	
Terms: A	NET 30	Credit Limit:	2500.00
Pay Method: AR	ACCOUNTS RECEIVABLE	Balance:	1049.99
Avg Days to Pay: 43		Open Orders:	2048.37
Last Pmt Amount: 850.01	Last Pmt Date: 01/31/2014	Avail. Credit:	-598.36
		YTD Sales:	8,045
		Lifetime Sales:	8,045
Credit Information			
Credit Check: <input checked="" type="checkbox"/>	Send Letters: <input checked="" type="checkbox"/>	Credit Hold: <input type="checkbox"/>	Credit Hold Date:
Credit Manager: maryj	Credit Limit: 2500.00	Limit Approved: bettyb	Approved Date: 02/12/2014
% Over Limit Allowed: 0.00	Hold Code: HOLD		
Collection Contact: SAM WHITE	Contact Telephone: 404-567-9287	DUNS No:	092876128

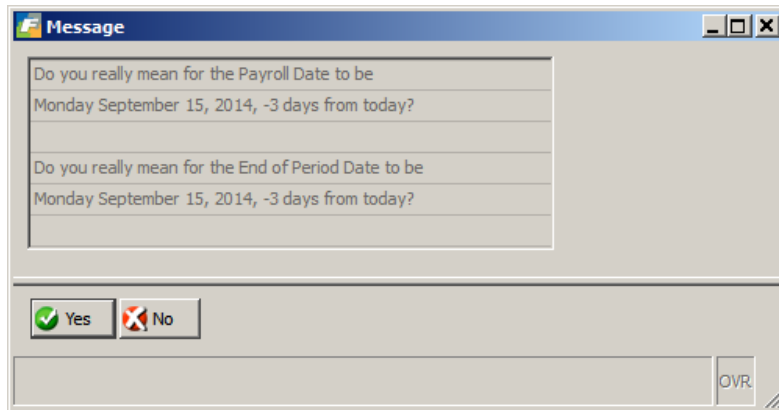
PAYROLL

FR1927 – FDD for direct deposit check stubs

Added flexible document delivery for direct deposit pay stubs so that the employee can receive it via email.

FR2436 – Payroll date verification

The Create Automatic Payroll program will display this prompt after the user has entered the payroll data and end of period date.



FR3617 – Additional Medicare Tax

Added logic to handle the additional .9 % medicare tax that is withheld from employees that are paid in excess of \$200,000 in a calendar year. This additional tax went into affect 01/01/2013

PRODUCTION ORDER PROCESSING

FR2573 – New Production Order Status report

09/16/2014 15:59:50

ABC MANUFACTURING

Page: 2

User: bettyb

Production Order Status Report

Pgm: sc422

=====

Order	Item	Sales Order	Line	Customer	Type	Due Date	Prty Oper	Responsible	Sts
509	C-MAC MAC LAPTOP				ST	08/05/2014			A
----- UN ISSUED COMPONENTS -----									
Sequence	Item	Description	Total Required		Issued				
1	C-MON	24" MONITOR	1.000		0.000				
2	C-KEY	KEYBOARD	1.000		0.000				
3	C-MOUSE	MOUSE	1.000		0.000				
4	C-DISK	HARD DRIVE	1.000		0.000				

Order	Item	Sales Order	Line	Customer	Type	Due Date	Prty Oper	Responsible	Sts
525	C-MAC MAC LAPTOP NEW VERSION	1630	1	12	CLASSIC PARTS UNL MTO	08/12/2014	1		A
----- UN ISSUED COMPONENTS -----									
Sequence	Item	Description	Total Required		Issued				
1	C-MON	24" MONITOR	1.000		0.000				
2	C-KEY	KEYBOARD	1.000		0.000				
3	C-MOUSE	MOUSE	1.000		0.000				
4	C-DISK	HARD DRIVE	1.000		0.000				

FR3125/3374 – Work in process cost report

This report provides the detail as to what makes up the work in process balance in the General Ledger. It can be run for any date in the past similar to how the AR and AP aging reports work.

09/18/2014 12:15:30
User: bettyb

ABC MANUFACTURING
Work in Process Cost Status Report
Detail by Item/Order

Page: 1
Pgm: cm410

Aging Date: 09/18/2014

Item C-MAC MAC LAPTOP Warehouse MIAMI

Order	Rel	Type	Sts	Quantity	Due Date	Material	Setup	Labor	Overhead	Costs	Misc	Charge	Total In	Recvd	Scrap	Balance
341	000	ST	A	10.000	05/06/14											
Components																
Seq	Item	Element														
1	C-MON	MATERIAL				675.00							675.00			
3	C-MOUSE	MATERIAL				72.00							72.00			
4	C-DISK	MATERIAL				589.99							589.99			
Order Totals						1336.99	0.00	0.00	0.00	0.00	0.00		1336.99	677.46	0.00	659.53
Item Totals						1336.99	0.00	0.00	0.00	0.00	0.00		1336.99	677.46	0.00	659.53
Report Totals						1336.99	0.00	0.00	0.00	0.00	0.00		1336.99	677.46	0.00	659.53

FR3159 – Add short components to existing purchase order

When entering a production work order you can now add short components orders to an existing purchase instead of creating a new one. On this screen simply enter the existing PO number in the Order field and check the confirm box.

Additional Information

File Edit Navigation Help

Confirm Order Type Sequence Item Warehouse Due Date Required Qty On Hand Available Short Qty On POs Qty to Order Vendor Buyer Unit Cost Status

<input type="checkbox"/>		REG-Regular To Stock	01	RB-HEAD	MIAMI	09/18/2014	100.000	46.000	-76.000	76.000	0.000	76.000				
<input type="checkbox"/>		REG-Regular To Stock	0102	HD-2000	MIAMI	09/18/2014	100.000	50.000	-98.000	98.000	0.000	98.000	123462		500.000	
<input type="checkbox"/>		REG-Regular To Stock	0103	HD-3000	MIAMI	09/18/2014	100.000	26.000	-98.000	98.000	0.000	98.000	123462		300.000	
<input type="checkbox"/>		REG-Regular To Stock	0104	HD-4000	MIAMI	09/18/2014	100.000	26.000	-98.000	98.000	0.000	98.000	123462		2000.000	
<input type="checkbox"/>		REG-Regular To Stock	0105	HD-5000	MIAMI	09/18/2014	200.000	50.000	-196.000	196.000	0.000	196.000	123462		100.000	
<input type="checkbox"/>		REG-Regular To Stock	0107	HD-7000	MIAMI	09/18/2014	200.000	52.000	-196.000	196.000	0.000	196.000	123462		100.000	
<input type="checkbox"/>		REG-Regular To Stock	0108	HD-8000	MIAMI	09/18/2014	100.000	28.000	-98.000	98.000	0.000	98.000	123462		100.000	
<input type="checkbox"/>		REG-Regular To Stock	02	RB-TORSO	MIAMI	09/18/2014	100.000	46.000	-84.000	84.000	0.000	84.000				
<input type="checkbox"/>		REG-Regular To Stock	0201	TR-1000	MIAMI	09/18/2014	200.000	52.000	-196.000	196.000	0.000	196.000	123462		300.000	
<input type="checkbox"/>		REG-Regular To Stock	0202	TR-1500	MIAMI	09/18/2014	200.000	52.000	-196.000	196.000	0.000	196.000	123462		75.000	

OK Cancel Create Orders

Enter Y to confirm

OVR

FR3269 – Production order log report (exported to Excel)

Prints the detail for all work orders (open and closed) entered for the year

Order No	Sales Order	Order Description	Extended Description	End Item	Warehouse	Qty	Start Date	Current Due Date	Customer PO	Customer Name
1		WIDGET 12999		12999-A	MIAMI	1	4/23/2014	4/23/2014		
1100		WIDGET 12999		12999-A	MIAMI	1	4/23/2014	4/23/2014		
173		SCM A SERIES MULSTRIKE		12104	MIAMI	1	1/17/2014	1/20/2014		
212		SCM A SERIES MULSTRIKE		12104	MIAMI	1	1/27/2014	1/28/2014		

FR3298 – Job cost summary report by salesperson

This report prints all closed production work orders with an associated sales order that has been invoiced for the date range and salesperson selected.

```

09/18/2014 11:44:46
User:
Job Price by SalesPerson/Customer
=====
Account      Prod Order   Sales Order  Line  Invoiced   Material    Labor      Hours    Rate      Description
=====
BJ          BILL JOHNSON
=====
12          CLASSIC PARTS UNLIMITED
=====
521         1623         000        1998.00     405.97     1592.03     0.00     0.00
529         1633         000        999.00     405.97     593.03     0.00     0.00
534         1663         000        4995.00    2279.85    2715.15     0.00     0.00
544         1671         000        2754.85    2529.85     225.00     0.00     0.00
=====
Account Totals      10746.85     5621.64     5125.21     0.00     0.00
=====
Sales Person Totals 10746.85     5621.64     5125.21     0.00     0.00
=====
Report Totals      10746.85     5621.64     5125.21     0.00     0.00
=====

```

FR3505 – Recording production scrap

New program to record production scrap and remove the material from inventory.

Production Scrap

File Edit View Navigation Tools Actions Help

Find Prev Next Add Browse

Order 426 Release 000 Scrap Number 1 Production Scrap Entry
 Item C-MAC Warehouse MIAMI Ordered 2.000 Scrap at Sequence 1 This Scrap 1.000 Update Now? ☒
 Total Received .000 Scrap Date 09/03/2014 11:11 Reason DEF
 Total Scrapped 1.000 GL Document 389 Unit Cost 0.00

Seq	Warehouse	Item	On Hand	Available	Required	Total Issued	Serial/Lot	This Issue	Sts	Detail	Comment
2	MIAMI	C-KEY	2766.000	2343.000				.000		<input checked="" type="radio"/> Needed <input checked="" type="radio"/> Supplied	
3	MIAMI	C-MOUSE	639.000	259.000				.000		<input checked="" type="radio"/> Needed <input checked="" type="radio"/> Supplied	
4	MIAMI	C-DISK	3894.000	2112.000				1.000	Close	<input checked="" type="radio"/> Needed <input checked="" type="radio"/> Supplied	

1 of 1

View Detail

OVR

FR3555 – Assign component serial numbers to Parent items

In prior versions of Fitrix full serial number traceability required that production orders be entered for a single unit at a time. This was the only way to ensure that individual component serial numbers could be traced back to individual end item serial numbers. In version 6.00 the production order processing programs have been modified so that:

- Serial numbers are assigned to parent items when the production work is created rather than when the end item is received into inventory
- Component serial numbers can be assigned to parent items during component issue, production receipt, or with a standalone program accessed from the transaction processing menu
- Two new inquiry programs have been added, one that shows indented serial composition for parent items and the other indented where used for components

Indented Serial Composition

File Edit View Navigation Tools Actions Help

Find Prev Next Browse

Item Code: C-MAC-RJ MAC LAPTOP - WITH BOARD DETAIL Serialized: Serialized

Serial No: ABC3838

Lot No:

Level	Item Code	S/L/B	Serial	Lot	Source Type	Source Order	Quantity	Vendor
1	C-MAC-RJ	Serialized	ABC3838		Production	603/000	1.000	
.2	C-MON	Serialized	ABC2232		Purchase	606/1	1.000	123458
.2	C-BOARD-ASSY	Serialized	ABC3536		Production	578/000	1.000	
.3	C-PROCESSOR	Serialized	ABC3531		Purchase	794/1	1.000	123457

4 of 9

Exit Detail

OVR

Indented Serial Where-Used

File Edit View Navigation Tools Actions Help

Find Prev Next Browse

Item Code: C-BOARD-ASSY COMPLETED MOTHER BOARD - C-MAC Serialized: Serialized

Serial No: ABC3536

Lot No:

Level	Item Code	S/L/B	Serial	Lot	Where Used Type	Where Used Order	Quantity	Customer/Vendor
0	C-BOARD-ASSY	Serialized	ABC3536		Produced	578/000	1.000	
1	C-MAC-RJ	Serialized	ABC3838		Comp Issue	603/000	2.000	

4 of 265

Exit Detail

OVR

PURCHASING

FR2284 – New fields for YTD and LTD purchases in vendor master

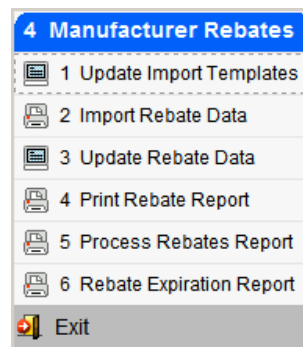
Two new fields for year-to-date and lifetime-to-date sales have been added to the vendor master screen. The AP invoice posting program was modified to keep these totals up to date. The begin new year program clears out YTD purchases at the beginning of each fiscal year.

FR2340 – Email alerts for backordered merchandise

Two new email alerts, one when a purchase order is created for a backordered item and another when merchandise is received that fills the backorder.

FR2415- Vendor rebate processing

- Option to Import rebates from an external source (i.e. spreadsheet)
- Modify OE post to store rebate cost with sales order
- End of month refresh program to set any null rebate costs due to contract not being in place when sales orders were processed
- On demand and end of month rebate report.
- Automatic GL transactions generated at end of month to accrue rebates due from suppliers
- Rebate expiration report, run on demand, to provide visibility into rebate contracts which are nearing expiration



FR2417- Reverse PO receipts

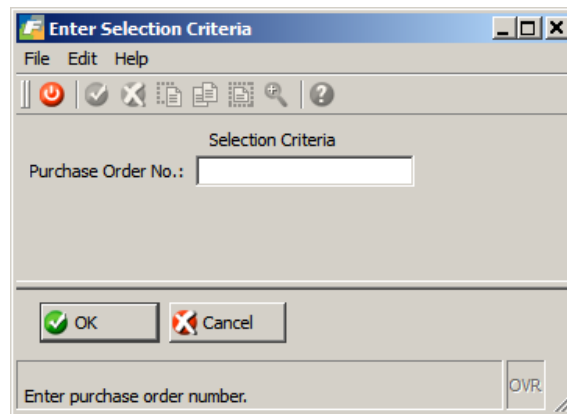
In the event that you receive the wrong PO or the quantities or cost were wrong, you can use this function to reverse the receipt. The posted PO can be deleted and all tables affected by the original receipt will be updated. You can then fix the quantities and costs and receive again.

FR2442 – Print your account number with your vendor on the purchase order

FR2569 – Change warehouse on all detail line if changed in PO header

FR2569 – Print labels for PO receipts

Print labels for all items received on a PO receipt

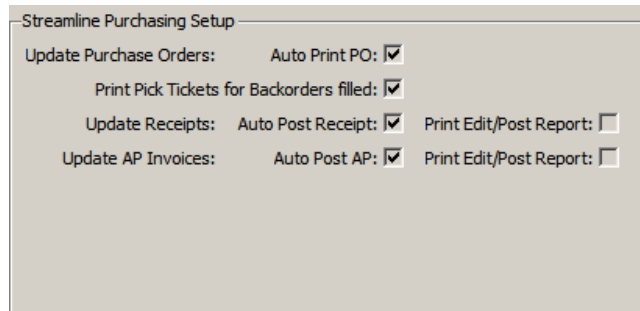


FR3052 – Display vendor item code on PO screens

In addition to displaying the internal item code fields have been added to display the vendor's item code in both the Update Purchase Orders screen and Update AP Invoices from purchasing screen.

FR3182/3202 – Streamlined purchasing transaction processing

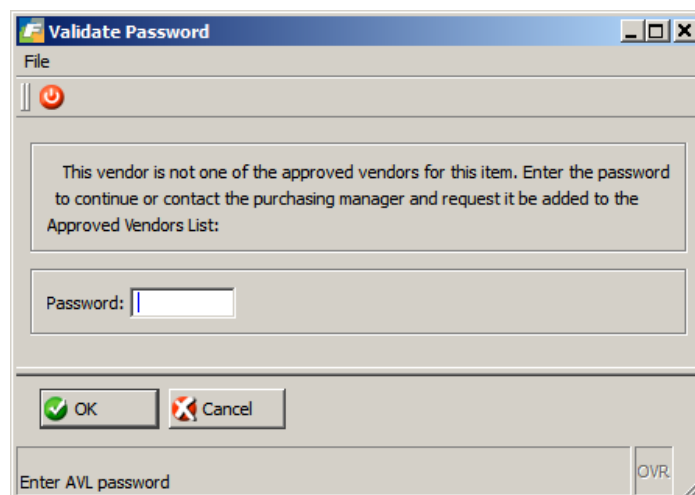
New control flags are in place that let the user determine if programs should be called automatically from another program instead of from the menu (i.e.- print the purchase order as soon as the purchase order is saved)



FR3191 – Print production order number on PO form

FR3206 – Approved vendor list (AVL)

If the AVL checkbox in the Purchasing Defaults program is checked and user tries to order an item from a vendor that is not set up as an approved vendor in the item catalog this message displays:



FR3217 – Compliance support for regulatory and industry standards

The item master contains two values that relate to compliance. The "FMD required" checkbox should be checked if a full materials declaration is required from your supplier.

General

Item Code: C-DISK

Description: HARD DRIVE

Item Class: COMP COMPUTER EQUIPMENT

Serial/Lot: Market Price: N

Mfg Name:

Mfg Item:

FMD Required: ☐ RoHS Compliant: ☒

The "RoHS Compliant" checkbox should be checked when the FMD is received and the item is deemed compliant. For any item that has "FMD required" set to Yes, an order line note will print underneath the item on the purchase order sent to the supplier that reads "This item requires that vendor provides a full material declaration document with shipment". Once the FMD has been received, you mark it received in the vendor item catalog and the note will no longer print on purchase orders sent to that supplier for that item.

The Non-Compliant Item List by Vendor report provides you with a list of items that require you have an FMD on file but you have yet to receive it.

FR3223 – Blanket purchase orders

Using blanket purchase orders allows you to submit orders to your vendors for large quantities and then release individual purchase orders against it with varying shipment dates. This is advantageous in that it prevents you from holding excess stock in your warehouses, cuts down on administrative costs and the vendor may offer volume discounts.

Blanket Release

File Edit View Navigation Tools Actions Help

PO Number: 314 Enter Date Vendor Should Ship: 10/01/2014

Release Date: 09/18/2014 Enter Date You Require Receipt of Product: 10/10/2014

Vendor Code: 123457

Vendor Name: CHAMPION INC

PO Date: 11/15/2013

Last Release No: 2

Item Code	Description	Qty Ordered	Qty Released	Qty Remaining	Qty to Release
C-DISK	HARD DISK	100.000	78.000	22.000	0.000
C-KEY	KEYBOARD	500.000	310.000	190.000	0.000
C-MON	24" MONITOR	400.000	162.000	238.000	0.000
C-TRANS	TRANSMITTER	800.000	465.000	335.000	0.000

OK Cancel Header

Enter qty to release. OVR

FR3268 – Purchase order log report (exported to Excel)

Prints the detail for all purchase orders (open and closed) entered for the year

A	B	C	D	E	F	G	H	I	J	K	L	M	N	O
PO#	Stage	Buyer	Receipt Date	Invoice Date	PO Date	Date Promised	Production Order	Sales Order	Item	Description	Qty	Vendor	Name	Cost
314-2	POG	CATHY			2/3/2014	2/15/2014			C-DISK	HARD DISK	8	123457	CHAMPION INC	40.00
314-2	POG	CATHY			2/3/2014	2/15/2014			C-KEY	KEYBOARD	10	123457	CHAMPION INC	35.00

FR3501 – Vendor delivery performance report

This report provides information about the delivery performance of your vendors for the date range specified. If the detail version is run it prints the purchase order numbers each item was received on.

Date: 09/18/2014		Vendor Delivery Performance							
Time: 16:39:46		ABC MANUFACTURING						Page: 1	
Warehouse	Vendor	Item	Number of Receipts			Percentage			
			Early	On-time	Late	Early	On-time	Late	
=====									
MIAMI	123458								
		C-MEMORY	0	0	1	0.00%	0.00%	100.00%	
		C-MON	0	2	0	0.00%	100.00%	0.00%	
		C-PROCESSOR	0	2	0	0.00%	100.00%	0.00%	
		FR3000MFG-1	0	0	1	0.00%	0.00%	100.00%	
		FR3000MFG-1A	0	1	0	0.00%	100.00%	0.00%	
		FR3000MFG-2	0	0	1	0.00%	0.00%	100.00%	
		FR3000MFG-2A	0	1	0	0.00%	100.00%	0.00%	
		FR4000WO-1	1	0	0	100.00%	0.00%	0.00%	
		FR4000WO-2	1	0	0	100.00%	0.00%	0.00%	
		GPA-2	0	4	0	0.00%	100.00%	0.00%	
	Subtotal		2	10	3	13.33%	66.67%	20.00%	
Total MIAMI			2	10	3	13.33%	66.67%	20.00%	
Grand Total			2	10	3	13.33%	66.67%	20.00%	

REPLENISHMENT

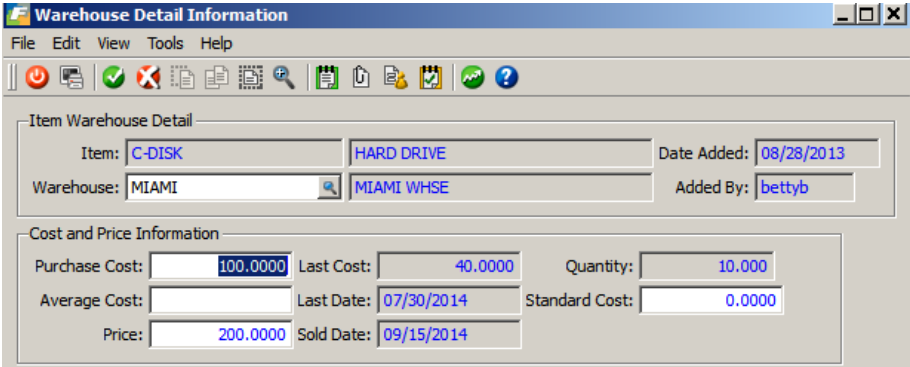
FR2620 – Add vendor business name to recommendations screen

FR2621 – Ability to create buy recommendations by vendor

FR2755 – Add excel export capabilities to the Expedite report

FR3034 – Base average usage calculation on date item was added to the database

The average usage calculation will be based on the date the item was added if a date added is found in the warehouse detail table, as shown here:



The screenshot shows a software window titled "Warehouse Detail Information". It contains two main sections: "Item Warehouse Detail" and "Cost and Price Information".

Item Warehouse Detail:

Item:	C-DISK	HARD DRIVE	Date Added:	08/28/2013
Warehouse:	MIAMI	MIAMI WHSE	Added By:	bettyb

Cost and Price Information:

Purchase Cost:	100.0000	Last Cost:	40.0000	Quantity:	10.000
Average Cost:		Last Date:	07/30/2014	Standard Cost:	0.0000
Price:	200.0000	Sold Date:	09/15/2014		

For example if the user is tracking 12 months of usage and the item was added 2 months ago average usage will be usage/2 not usage/12. The number of periods lookup definition has been changed to use this added date when determining the number of periods to use in the average usage calculation.

Update Recommendations

File Edit View Navigation Tools Actions Help

Find Prev Next Add Update Delete Browse

Minimum: 1000.00 Stock: 22676.32 Whse: SEATTLE Script: SEATTL
 Target: 1000.00 Nonstk: 0.00 Line: SCM Period: MONTH
 Current: 22676.32 Volume: 0.000 Buyr: Cycle: 1.00
 Roll: Weight: 294.128 Nxt Rvw: 06/04/2014
 Vend: SCM SMITH-CORONA CORP. PO No: 536

Item Code	Avail	On PO	Rec Qty	Price	Extended
12112	-8118	11371	9488	2.3900	22676.32

Volume: Ext:
 Weight: Ext:
 Item Type: Purchase UM: Lead Time: ABC Rank: Discounts:

1 of 2

View Detail

OVR

FR3035 – Missing catalog entries report

This report that lists items not set up in the item catalog or in the catalog but missing the replenishment script code and therefore not picked up in the replenishment process.

FR3133 – Utility to reset RL Script definitions

If you ever need to zero out usage, calculations, and recommendations and reset all dates use this utility. It will delete all the data for the script code entered and reset all the dates to 01/01/2000 so you can run the replenishment process from beginning to end again for your current cycle. This utility is useful if you decide to change a vendor from a weekly replenishment cycle to a monthly replenishment cycle. It basically resets all data to 0 so you can start from day one.

TECHNICAL NOTES

Per James:

I think this is going to have to be brad when we get to this and we don't need much for version 6.0, the rest can come later

Ones I can think of:

New version of Informix, 4Js and Linux supported

2338 auto timestamp notes

No # print database images on reports

No # up and down arrows no work on menus

2405 FDD/GUI print enhancements

PDF print expanded to 132

Change to code generator that Brad dealt with at SLE

Tom also emailed me a pdf of the CASE tool info file that Brad should review when he's ready to write this section of the release notes

NEW TABLES

--- FR3079 - Utility to generate role based menus

```
create table "informix".cgsmnrol
  ckey char(3),
  user_id char(20),
  user_default char(1)
```

--- FR2664 - Add ability to run report and attach file

--- Mapping table for email addresses

```
create table "informix".stxemaph (
  user_id char(20),
  email_address char(60) );
```

--- FR2664 - Add ability to run report and attach file

--- Add table for BCC addresses

```
create table "informix".stxevtmb
  event_name char(20),
  bcc_recipient char(60),
  conditional_exp char(80)
```

--- FR2664 - Add ability to run report and attach file

--- Add table for Cc addresses

```
create table "informix".stxevtmc
  event_name char(20),
  cc_recipient char(60),
  conditional_exp char(80)
```

-- FR2664 - Add ability to run report and attach file

--- Add table for Reply-to addresses

```
create table "informix".stxevtmy
  event_name char(20),
  reply_recipient char(60),
```

conditional_exp char(80)

--- FR2664 - Add ability to run report and attach file

--- EVenT RePort

```
create table "informix".stxevtrp (  
  event_name char(20),  
  sequence smallint,  
  report_module char(8),  
  report_program char(8),  
  report_progver char(8),  
  report_params char(200),  
  report_filename char(60),  
  attach_encrypt char(1),  
  attach_passwd char(20)) ;
```

--- FR2836 - Utility to fix indexes missing in IDS

```
create table "informix".stxindex  
  tabname varchar(128,0),  
  idxname varchar(128,0),  
  idxtype char(1),  
  part1 smallint,  
  part2 smallint,  
  part3 smallint,  
  part4 smallint,  
  part5 smallint,  
  part6 smallint,  
  part7 smallint,  
  part8 smallint
```

-- FR2415 - Create rebate import mapping tables

```
create table "informix".stxmaptd  
  (tablename char(20),  
  template char(20),  
  sheet_column char(3),  
  column_no smallint,  
  db_column char(20));
```

-- FR2415 - Create rebate import mapping tables

```
create table "informix".stxmapte
```

```
(tablename char(20),  
template char(20));
```

--- FR3079 - Utility to generate role based menus

```
create table "informix".stxmngrd  
  group_id char(20),  
  user_id char(20)  
alter table stxmngrd lock mode(row);
```

--- FR3079 - Utility to generate role based menus

```
create table "informix".stxmngrh  
  group_id char(20)  
alter table stxmngrh lock mode(row);  
create unique index "informix".i_stxmngrh1 on "informix".stxmngrh (group_id);
```

--- FR3079 - Utility to generate role based menus

```
create table "informix".stxmnith  
  c_key char(3),  
  description char(30)  
  
alter table stxmnith lock mode(row);
```

--- FR3079 - Utility to generate role based menus

```
create table "informix".stxmnitm  
  c_key char(3),  
  line_no smallint,  
  c_txt char(64),  
  c_mname char(32),  
  c_opt char(1),  
  c_mtype char(2),  
  txt char(64),  
  mname char(32),  
  
  opt char(1),  
  
  mtype char(2)
```

--- FR2332 - Streamline order entry

```
create table "informix".stxprntd  
  
  default_printer char(40)
```


--- FR2332 - Streamline order entry

```
create table "informix".stxprntr
(
  prog_name      char(18),
  prog_desc      char(30),
  default_printer char(40)
```

--- FR3120 - Utility to move sample data dates forward

```
create table "informix".stxredate
  idxtype char(1),
  tabname varchar(128,0),
  colname varchar(128,0)
```

--- FR3079 - Utility to generate role based menus

```
create table "informix".stxrmnid
  c_key char(3),
  user_id char(20),
  user_default char(1)
```

--- FR3079 - Utility to generate role based menus

```
create table "informix".stxrmnih
  user_id char(20)
```

--- FR2715 add email address to each user

```
alter table stxsecur add (
  email char(60))
```

--- FR2522 - Add new table for COMMISSION CALCULATION BY CUSTOMER

```
create table "informix".strcommd (
  cust_code char(20),
  item_code char(20),
  begin_price decimal(14),
  end_price decimal(14),
  comm_pct decimal(14)
grant all on "informix".strcommd to "public" as "informix";
alter table "informix".strcommd lock mode (row);
```

-- FR3323 - Customer Credit Management

```
create table "informix".strcrhdd
  cust_code char(20),
  ag_current decimal(12),
  ag_over_30 decimal(12),
  ag_over_60 decimal(12),
  ag_over_90 decimal(12)
```

-- FR3323 - Customer Credit Management

```
create table "informix".strcrhde
  hold_code char(6),
  description char(50),
  hold_wo char(1),
  hold_po char(1),
  print_opicker char(1),
  update_pqtys char(1),
  update_sqtys char(1),
  print_oshipr char(1),
  create_invoice char(1),
  print_oinvce char(1)
```

--- FR2127 - Add Product Configurator module to Fitrix

```
drop table cfgselans;
create table "informix".cfgselans
  select_number_gv decimal(9,0),
  group_gv char(15),
  question_gv char(10),
  sequence_gv smallint,
  answer_gv char(30),
  default_gv char(1),
  curr_dsp_mode_gv char(1),
  orig_dsp_mode_gv char(1),
  set_by_qst_gv char(10),
  set_by_ans_gv char(15),
  add_date_gv date,
  change_date_gv date
```

--- FR2127 - Add Product Configurator module to Fitrix

```
drop table cfgselopt;
create table "informix".cfgselopt
  select_number_gu decimal(9,0),
  group_gu char(15),
```

```
question_gu char(10),
answer_gu char(15),
option_gu char(10),
unit_price_gu money(13,5),
unit_cost_gu money(11,4),
add_date_gu date,
change_date_gu date
```

--- FR2127 - Add Product Configurator module to Fitrix

```
drop table cfgselqst;
create table "informix".cfgselqst
  select_number_gw decimal(9,0),
  group_gw char(15),
  sequence_gw smallint,
  question_gw char(10),
  answer_gw char(15),
  default_answer_gw char(15),
  ans_required_gw char(1),
  curr_dsp_mode_gw char(1),
  orig_dsp_mode_gw char(1),
  set_by_qst_gw char(10),
  set_by_ans_gw char(15),
  add_date_gw date,
  change_date_gw date
```

--- FR2127 - Add Product Configurator module to Fitrix

```
create table "informix".cfgselser
  select_number_ge decimal(9,0),
  serial_number_ge char(10),
  order_ge char(7),
  order_lot_ge char(3)
```

--- FR2127 - Add Product Configurator module to Fitrix

```
drop table cfgselsumm;
create table "informix".cfgselsumm
  select_number_gs decimal(9,0),
  group_gs char(15),
  type_gs char(1),
  catalogue_gs char(30),
  desc_gs char(80),
  order_gs char(7),
```

```

release_gs char(3),
line_gs decimal(4,0),
item_gs char(20),
whse_gs char(10),
serial_number_gs char(20),
unit_price_gs money(13,5),
unit_cost_gs money(11,4),
add_date_gs date,
change_date_gs date
) lock mode row ;

```

--- **FR2127 - Add Product Configurator module to Fitrix**

```

drop table cfgtplans;
create table "informix".cfgtplans
  group_ga char(15),
  question_ga char(10),
  sequence_ga smallint,
  answer_ga char(30),
  default_ga char(1),
  auto_display_ga char(1),
  text1_ga char(80),
  text2_ga char(80),
  start_date_ga date,
  stop_date_ga date,
  add_date_ga date,
  change_date_ga date

```

--- **FR2127 - Add Product Configurator module to Fitrix**

```

drop table cfgtplcomp;
create table "informix".cfgtplcomp
  group_gc char(15),
  option_gc char(10),
  comp_sequence_gc char(4),
  component_item_gc char(20),
  desc1_gc char(30),
  desc2_gc char(30),
  quantity_type_gc char(1),
  quantity_gc decimal(14,7),
  qty_formula_id_gc char(20),
  start_date_gc date,
  stop_date_gc date,

```

```

op_where_used_gc char(4),
ofst_from_strt_gc smallint,
sply_to_vendor_gc char(1),
issue_method_gc char(1),
shrinkage_gc decimal(5,2),
packet_print_gc char(1),
price_gc decimal(12,4),
add_date_gc date,
change_date_gc date

```

--- **FR2127 - Add Product Configurator module to Fitrix**

```

drop table cfgtplfrmla;
create table "informix".cfgtplfrmla
  group_gf char(15),
  formula_id_gf char(20),
  formula_text_gf char(256),
  add_date_gf date,
  change_date_gf date

```

--- **FR2127 - Add Product Configurator module to Fitrix**

```

drop table cfgtplmisc;
create table "informix".cfgtplmisc
  group_gm char(15),
  option_gm char(10),
  charge_code_gm char(15),
  description_gm char(40),
  cost_gm money(9,2),
  price_gm money(9,2),
  start_date_gm date,
  stop_date_gm date,
  add_date_gm date,
  change_date_gm date

```

--- **MR2562 - Add columns to cfgtplqst, cfgtplans for numeric answers**

```

create table "informix".cfgtplmult
  group_gu char(15),
  option_gu char(10),
  description_gu char(40),
  sequence_gu smallint,
  question_gu char(10),
  answer_type_gu char(1),

```

```
answer_gu char(30),
from_answer_gu char(30),
to_answer_gu char(30),
add_date_gu date,
change_date_gu date
```

--- FR2127 - Add Product Configurator module to Fitrix

```
drop table cfgtplopt;
create table "informix".cfgtplopt
  group_go char(15),
  option_go char(10),
  description_go char(40),
  plan_mix_go decimal(5,2),
  start_date_go date,
  stop_date_go date,
  add_date_go date,
  change_date_go date
```

--- FR2127 - Add Product Configurator module to Fitrix

```
drop table cfgtplqst;
create table "informix".cfgtplqst
  group_gq char(15),
  sequence_gq smallint,
  question_gq char(10),
  text1_gq char(80),
  text2_gq char(80),
  answer_type_gq char(1),
  answer_count_gq smallint,
  ans_required_gq char(1),
  auto_display_gq char(1),
  start_date_gq date,
  stop_date_gq date,
  add_date_gq date,
  change_date_gq date
```

--- FR2127 - Add Product Configurator module to Fitrix

```
drop table cfgtplres;
create table "informix".cfgtplres
  group_gr char(15),
  question_gr char(10),
```

```
answer_gr char(30),
sequence_gr smallint,
response_gr char(10),
type_gr char(2),
option_gr char(10),
on_off_quest_gr char(10),
on_off_ans_gr char(15),
ans_set_val_gr char(15),
start_date_gr date,
stop_date_gr date,
add_date_gr date,
change_date_gr date
```

--- FR2127 - Add Product Configurator module to Fitrix

```
drop table cfgtplrout;
create table "informix".cfgtplrout
```

```
group_gt char(15),
option_gt char(10),
routing_group_gt char(10),
routing_seq_gt char(4),
operation_gt char(4),
desc_gt char(80),
workctr_gt char(4),
machine_gt char(4),
dept_gt char(3),
team_gt char(5),
tool_item_gt char(15),
lbr_or_mach_hr_gt char(1),
setup_hrs_gt decimal(9,5),
run_hrs_gt decimal(9,5),
mach_hrs_gt decimal(9,5),
inside_outside_gt char(1),
outs_unit_cost_gt money(11,4),
outside_item_gt char(15),
packet_print_gt char(1),
lbr_tran_type_gt char(1),
price_gt decimal(12,4),
start_date_gt date,
stop_date_gt date,
add_date_gt date,
```

change_date_gt date

--- FR2127 - Add Product Configurator module to Fitrix

```
mtl_elmt_stk_pc = "MATERIAL",  
mtl_elmt_non_pc = "NON-STOCK MATERIAL",  
lbr_elmt_pc = "LABOR",  
ovh_elmt_pc = "OVERHEAD",  
outs_elmt_pc = "OUTSIDE PROCESS"  
where 1=1;
```

---FR3389 - New table billcomprd - reference designators 04/04/2014

```
create table "informix".billcomprd  
parent_item_br char(20),  
bill_br char(5),  
comp_item_br char(20),  
comp_sequence_br char(10),  
ref_designator_br char(10)
```

---FR3447.sql - new tabels and rows for Production Line Inventory

```
create table "informix".compline  
whse_lc char(10),  
item_lc char(20),  
prod_line_lc char(10),  
routing_seq_lc char(4),  
comp_sequence_lc char(10),  
component_item_lc char(20),  
location_lc char(20),  
minimum_qty_lc decimal(10,3),  
maximum_qty_lc decimal(10,3)
```

---FR1493 - Create tables, alter tables, add rows to support PS module

```
create table initps  
init_key_iq char(1),  
shift1_cap_hr_iq decimal(8,4),  
shift2_cap_hr_iq decimal(8,4),  
shift3_cap_hr_iq decimal(8,4),  
cap_start_date_iq date,  
cap_end_date_iq date,  
dflt_prior_iq char(1),
```



```
dflt_secn_prior_iq char(1),
recmd_win_iq smallint,
schedule_type_iq char(1),
sched_horizon_iq smallint,
dflt_cap_intv_iq char(3),
cap_green_lvl_iq smallint,
cap_yellow_lvl_iq smallint
```

--- FR3290 - OE Sales history detail view

```
---drop view "fitrix".uv_oe_sales_history_detail;
create view "fitrix".uv_oe_sales_history_detail (
  order_type,
  sell_to_code,
  ship_to_code,
  doc_no,
  inv_doc_no,
  item_code,
  ship_qty,
  price,
  item_cost,
  net_amount,
  gross_margin,
  desc1,
  desc2,
  ext_desc,
  mfg_name,
  mfg_item,
  item_class,
  inv_date,
  bus_name,
  city,
  state,
  zip,
  country
as
select
  stoordre.like_type,
  stoshipd.sell_to_code,
  stoshipd.ship_to_code,
  stoshipd.doc_no,
  stoshipd.inv_doc_no,
```

```

stoshipd.item_code,
case
  when stoordre.like_type = "CRM"
    then stoshipd.ship_qty * (-1)
  else stoshipd.ship_qty
end ship_qty,
stoshipd.price,
stoshipd.item_cost,
case
  when stoordre.like_type = "CRM"
    then stoshipd.net_amount * (-1)
  else stoshipd.net_amount
end net_amount,
case
  when stoordre.like_type = "CRM"
    then stoshipd.gross_margin * (-1)
  else stoshipd.gross_margin
end gross_margin,
stiinvtr.desc1,
stiinvtr.desc2,
stiinvtr.ext_desc,
stiinvtr.mfg_name,
stiinvtr.mfg_item,
stiinvtr.item_class,
stoinvce.inv_date,
strcustr.bus_name,
strcustr.city,
strcustr.state,
strcustr.zip,
strcustr.country
from stoordre, stoshipd, stiinvtr, stoinvce, strcustr
where
  stoordre.doc_no = stoshipd.doc_no and
  stoshipd.sell_to_code = strcustr.cust_code and
  stoshipd.inv_doc_no = stoinvce.inv_doc_no and
  stoshipd.item_code = stiinvtr.item_code and
  stoshipd.stage = "PST";

```

--- FR3504 - Purchase History Detail

---drop view "fitrix".uv_pu_purchase_history_detail ;

```

create view "fitrix".uv_pu_purchase_history_detail (
  transaction_date,
  vendor_code,
  vendor_name,
  city,
  state,
  zip,
  country,
  warehouse_code,
  item_class,
  item_code,
  item_description1,
  item_description2,
  qty_purchased,
  item_cost
as
select
  stxtranr.doc_date,
  stxtranr.ref_code,
  stpvendr.bus_name,
  stpvendr.city,
  stpvendr.state,
  stpvendr.zip,
  stpvendr.country,
  stiactvd.warehouse_code,
  stiactvd.item_class,
  stiactvd.item_code,
  stiinvtr.desc1,
  stiinvtr.desc2,
  stiactvd.qty,
  stiactvd.cost
from stxtranr, stpvendr, stiactvd, stiinvtr
where
  stpvendr.vend_code = stxtranr.ref_code and
  stxtranr.doc_no = stiactvd.doc_no and
  stxtranr.orig_journal = stiactvd.orig_journal and
  stiactvd.item_code = stiinvtr.item_code and
  stiactvd.orig_journal = "PR";

```

--- FR3516 - Obsolete Inventory Items

```

---drop view "fitrix".uv_ic_obsolete_items_report ;

```

```

create view "fitrix".uv_ic_obsolete_items_report (
  item_code,
  item_description1,
  item_description2,
  last_date_sold,
  unit_of_measure,
  qty_on_hand,
  average_cost,
  warehouse_code
as
select
  stiinvtr.item_code,
  stiinvtr.desc1,
  stiinvtr.desc2,
  stilocar.sold_date,
  stiinvtr.stock_unit,
  stilocar.qty_on_hand,
  stilocar.avg_unit_cost,
  stilocar.warehouse_code
from stiinvtr, stilocar
where
  stiinvtr.item_code = stilocar.item_code and
  stilocar.obsolete = "Y";

```

--- FR3517 - Customer Shipment Performance

```

---drop view "fitrix".uv_oe_cust_ship_performance_report;
create view "fitrix".uv_oe_cust_ship_performance_report (
  customer_code,
  customer_name,
  order_no,
  ship_date,
  required_date
as
select
  stoinvce.sell_to_code,
  strcustr.bus_name,
  stoinvce.doc_no,
  stoinvce.inv_date,
  stoordre.required_date
from stoordre, stoinvce, strcustr
where

```

```
stoordre.order_no = stoinvce.order_no and
stoinvce.sell_to_code = strcustr.cust_code;
grant select on "fitrix".uv_oe_cust_ship_performance_report to "public" as "fitrix";
```

--- FR3518 - Customer Backordered Reports

```
---drop view "fitrix".uv_oe_backorder_log ;
create view "fitrix".uv_oe_backorder_log (
    item_code,
    item_description1,
    item_description2,
    customer_name,
    date_backordered,
    promised_date,
    order_no,
    line_no,
    qty_backordered,
    amount_backordred
as
select
    stoshipd.item_code,
    stiinvtr.desc1,
    stiinvtr.desc2,
    strcustr.bus_name,
    stoshipd.bko_date,
    stoshipd.atp_date,
    stoordre.order_no,
    stoshipd.line_no,
    stoshipd.ship_qty,
    stoshipd.net_amount
from stoordre, stoshipd, stiinvtr, strcustr
where
    stoordre.doc_no = stoshipd.doc_no and
    stoshipd.sell_to_code = strcustr.cust_code and
    stoshipd.item_code = stiinvtr.item_code and
    stoshipd.stage = "BKO";
```

--- FR3522 - RMA's issued

```
---drop view "fitrix".uv_oe_rma_report ;
create view "fitrix".uv_oe_rma_report (
    item_code,
    item_description1,
```

```

item_description2,
order_type,
order_status,
customer_code,
order_date,
rma_number,
original_order_no,
returned_qty,
returned_amount,
reason_returned
as
select
  stoordrd.item_code,
  stoordrd.desc1,
  stoordrd.desc2,
  stoordre.order_type,
  stoordre.order_status,
  stoordre.cust_code,
  stoordre.order_date,
  stoordre.order_no,
  stoordre.base_doc_no,
  stoordrd.ordr_qty,
  stoordrd.net_amount,
  stoordre.rma_reason
from stoordre, stoordrd
where
  stoordre.doc_no = stoordrd.doc_no and
  stoordre.order_type in ("RMA","RMU");

```

--- FR3523 - Serial/Lot Composition Report

```

---drop view "fitrix".uv_ic_serial_lot_comp_report ;
create view "fitrix".uv_ic_serial_lot_comp_report (
  item_code,
  item_description,
  warehouse_code,
  serial_no,
  lot_no,
  qty_on_hand,
  receipt_date,
  component_item ,

```

```

    comp_description,
    component_qty,
    comp_serial_no,
    comp_lot_no
as
select
    wrecloc.item_wf,
    wosumm.desc_ws,
    wrecloc.whse_wf,
    wrecloc.serial_number_wf,
    wrecloc.batch_lot_wf,
    wrecloc.onhand_balance_wf,
    wrecloc.date_rec_wf,
    woissloc.component_item_wk,
    wocomp.desc1_wm,
    woissloc.iss_qty_wk,
    woissloc.serial_number_wk,
    woissloc.batch_lot_wk
from wrecloc, wosumm, wocomp, woissloc
where
    wocomp.order_wm = wrecloc.order_wf and
    wocomp.order_lot_wm = wrecloc.order_lot_wf and
    wocomp.comp_sequence_wm = woissloc.comp_sequence_wk and
    wocomp.component_item_wm = woissloc.component_item_wk and
    woissloc.order_wk = wrecloc.order_wf and
    woissloc.order_lot_wk = wrecloc.order_lot_wf and
    wosumm.order_ws = wrecloc.order_wf and
    wosumm.order_lot_ws = wrecloc.order_lot_wf;

```

--- FR3525 - QC Results for PO Receipts

```

---drop view "fitrix".uv_pu_vendor_qc_report ;
create view "fitrix".uv_pu_vendor_qc_report (
    warehouse_code,
    vendor_code,
    vendor_name,
    item_code,
    po_number,
    date_inspected,
    quantity_tested,
    quantity_passed,

```

```

quantity_failed,
fail_code,
code_description,
comment
as
select
    stiqcene.warehouse_code,
    stuordre.vend_code,
    stpvendr.bus_name,
    stiqcene.item_code,
    stuordre.po_no,
    stiqcene.inspect_date,
    stiqcene.msmt_qty,
    stiqcene.pass_qty,
    stiqcene.fail_qty,
    stiqcene.fail_reason,
    stxinfor.src_desc,
    stiqcene.comment
from stiqcene, stuordre, stpvendr, stxinfor
where
    stiqcene.order_no = stuordre.po_no and
    stpvendr.vend_code = stuordre.vend_code and
    stxinfor.src_key = stiqcene.fail_reason and
    stxinfor.src_type = "Q" and
    stiqcene.collect_type = "P";

```

--- FR3526 - Purchases Analysis by vendor

```

---drop view "fitrix".uv_pu_purch_analysis_by_vendor_dtl ;
---drop view "fitrix".uv_pu_purch_analysis_by_vendor ;
---drop view "fitrix".uv_pu_purch_last_year_by_vendor ;
---drop view "fitrix".uv_pu_purch_last_year_dates ;
create view "fitrix".uv_pu_purch_last_year_dates (
    start_date,
    end_date
as
select
    min(stxperdr.start_date),
    max(stxperdr.end_date)
from stxperdr, stgcntrc
where stxperdr.period_year = stgcntrc.curr_year - 1;
grant select on "fitrix".uv_pu_purch_last_year_dates to "public" as "fitrix";

```



```

create view "fitrix".uv_pu_purch_last_year_by_vendor (
    vend_code,
    lytd_purchases
as
select
    stpvendr.vend_code,
    sum(stpopend.orig_amount)
from stpopend, stpvendr, uv_pu_purch_last_year_dates
where stpopend.vend_code = stpvendr.vend_code and
    stpopend.inv_date between uv_pu_purch_last_year_dates.start_date and
uv_pu_purch_last_year_dates.end_date
group by stpvendr.vend_code;
grant select on "fitrix".uv_pu_purch_last_year_by_vendor to "public" as "fitrix";
create view "fitrix".uv_pu_purch_analysis_by_vendor (
    vendor_code,
    vendor_name,
    ytd_purchases,
    last_ytd_purchases,
    increase_decrease,
    lifetime_purchases
as
select
    stpvendr.vend_code,
    stpvendr.bus_name,
    stpvendr.ytd_purchases,
    uv_pu_purch_last_year_by_vendor.lytd_purchases,
    stpvendr.ytd_purchases - uv_pu_purch_last_year_by_vendor.lytd_purchases,
    stpvendr.lifetime_purchases
from stpvendr, uv_pu_purch_last_year_by_vendor
where
    stpvendr.vend_code = uv_pu_purch_last_year_by_vendor.vend_code;
grant select on "fitrix".uv_pu_purch_analysis_by_vendor to "public" as "fitrix";
create view "fitrix".uv_pu_purch_analysis_by_vendor_dtl (
    vendor_code,
    vendor_name,
    inv_doc_no,
    inv_date,
    po_no,
    item_code,
    item_description1,
    item_description2,

```

```

order_qty,
item_cost,
net_amount
as
select
  stpvendr.vend_code,
  stpvendr.bus_name,
  stpopend.doc_no,
  stpopend.inv_date,
  stuordre.po_no,
  stuordrd.item_code,
  stuordrd.desc1,
  stuordrd.desc2,
  stuinvcld.cost_qty,
  stuinvcld.cost,
  stuinvcld.net_price
from stpvendr, stpopend, stuinvcld, stuordrd, stuordre
where
  stpvendr.vend_code = stpopend.vend_code and
  stpopend.doc_no = stuinvcld.inv_doc_no and
  stuordre.doc_no = stuordrd.doc_no and
  stuinvcld.po_doc_no = stuordrd.doc_no and
  stuinvcld.po_line_no = stuordrd.line_no
union
select
  stpvendr.vend_code,
  stpvendr.bus_name,
  stpopend.doc_no,
  stpopend.inv_date,
  "",
  "",
  "",
  "",
  "",
  1,
  stpopend.orig_amount,
  stpopend.orig_amount
from stpvendr, stpopend
where
  stpvendr.vend_code = stpopend.vend_code and
  stpopend.doc_no not in (select stuinvcld.inv_doc_no from stuinvcld)

```

-- FR2047 - CRM Module

create table "informix".stcaccth

acct_id serial not null ,
bus_name char(30),
customer_yn char(1),
cust_code char(20),
parent_company integer,
web_site char(80),
revenue decimal(20),
num_employees smallint,
stock_ticker char(10),
industry_type char(25),
industry_standard char(5),
sic_naics_code char(6),
account_type char(25),
account_status char(20),
account_source char(25),
contact_id integer,
name_courtesy char(12),
first_name char(20),
middle char(20),
last_name char(20),
name_suffix char(12),
title char(20),
email_id integer,
email_address char(50),
addr_id integer,
address_type char(10),
address1 char(30),
address2 char(30),
city char(30),
county char(30),
state_prov char(3),
post_code char(10),
country_code char(3),
phone_id integer,
international char(1),
phone_type char(10),
phone char(18),
phone_ext char(10),
rep_id integer,

team_id integer,
ytd_sales decimal(20),
ltd_sales decimal(20),
create_date date,
create_by char(8),
dup_code char(14)

-- FR2047 - CRM Module

```
create table "informix".stcactvd
  activity_id serial not null ,
  acct_id integer,
  contact_id integer,
  phone_id integer,
  email_id integer,
  lead_id integer,
  list_id integer,
  campaign_id integer,
  parent_activity_id integer,
  activity_type char(25),
  description char(60),
  create_date date,
  create_by char(8),
  activity_status char(20),
  scheduled_date date,
  scheduled_chartime char(12),
  scheduled_time datetime hour to second,
  duration_days smallint,
  duration_hours smallint,
  duration_minutes smallint,
  recurring_yn char(1),
  freq_id integer,
  rep_id integer,
  team_id integer,
  reminder_yn char(1),
  reminder_lead interval day to minute,
  reminder_id integer,
  complete_yn char(1),
  complete_date date,
  complete_time datetime hour to second,
  complete_by char(8),
  letter_code char(8)
```

-- FR2047 - CRM Module

create table "informix".stcaddrd

addr_id serial not null ,
acct_id integer,
contact_id integer,
address_type char(10),
nickname char(15),
primary_yn char(1),
address1 char(30),
address2 char(30),
city char(30),
county char(30),
state_prov char(3),
post_code char(10),
country_code char(3)

-- FR2047 - CRM Module

create table "informix".stcalerh

alert_id serial not null ,
user_id char(8),
alert_date date,
alert_time datetime hour to minute,
alert_datetime datetime year to minute,
remind_lead_days interval day to day,
remind_lead_hours interval hour to hour,
remind_lead_mins interval minute to minute,
remind_datetime datetime year to minute,
alert_status char(10),
subject char(80),
alert_importance char(25),
time_zone_adj interval hour to hour,
recurring_yn char(1),
freq_id integer,
activity_id integer

-- FR2047 - CRM Module

TABLE "informix".stccampd row size = 206 number of columns = 26 index size = 61

create table "informix".stccampd

campaign_id serial not null ,
acct_id integer,

```

contact_id integer,
lead_id integer,
list_id integer,
campaign_type char(25),
description char(60),
prime_or_all char(1),
project_id integer,
create_date date,
create_by char(8),
campaign_status char(20),
start_date date,
end_date date,
recurring_yn char(1),
freq_id integer,
budget_amount decimal(16,2),
estimated_cost decimal(16,2),
campaign_cost decimal(16,2),
estimated_revenue decimal(16,2),
manager_id integer,
team_id integer,
complete_yn char(1),
select_method char(1),
query_id integer,
executed char(1)
revoke all on "informix".stccampd from "public" as "informix";

```

-- FR2047 - CRM Module

```

create table "informix".stccased
  case_id serial not null ,
  acct_id integer,
  addr_id integer,
  contact_id integer,
  phone_id integer,
  email_id integer,
  create_date date,
  create_time datetime hour to second,
  create_by char(8),
  brief_description char(60),
  type_major char(25),
  type_minor char(25),

```

```

detail_description varchar(254),
case_status char(25),
priority char(25),
severity char(25),
team_id integer,
rep_type char(25),
assigned_to_id integer,
assigned_date date,
assigned_time datetime hour to second,
resolved_yn char(1),
resolved_date date,
resolved_time datetime hour to second,
resolve_solution varchar(254)

```

-- FR2047 - CRM Module

```

create table "informix".stccntd
  campaign_id integer,
  contact_id integer

```

-- FR2047 - CRM Module

```

create table "informix".stccitmd
  case_id integer,
  item_code char(20),
  replaced_item_code char(20),
  quantity integer,
  comment char(60)

```

-- FR2047 - CRM Module

```

create table "informix".stccmpdd
  campaign_id integer,
  activity_id integer,
  complete_yn char(1)

```

-- FR2047 - CRM Module

```

create table "informix".stccnold
  lead_id integer,
  contact_id integer

```

-- FR2047 - CRM Module

```

create table "informix".stccntd
  contact_id serial not null ,

```

```

acct_id integer,
name_courtesy char(12),
first_name char(20),
middle char(20),
last_name char(20),
name_suffix char(12),
full_name char(63),
title char(20),
primary_yn char(1),
contact_type char(20),
department char(20),
salutation char(50),
birth_day datetime month to day,
assistant char(30),
addr_id integer

```

-- FR2047 - CRM Module

```

create table "informix".stcemald
  email_id serial not null ,
  acct_id integer,
  contact_id integer,
  primary_yn char(1),
  email_type char(10),
  email_address char(50)

```

-- FR2047 - CRM Module

```

create table "informix".stc
indsr
  industry_standard char(5),
  sic_naics_code char(6),
  description char(60)

```

-- FR2047 - CRM Module

```

create table "informix".stcleadd
  lead_id serial not null ,
  acct_id integer,
  contact_id integer,
  phone_id integer,
  email_id integer,

```



```

create_date date,
create_by char(8),
lead_type char(25),
description char(80),
rep_id integer,
team_id integer,
lead_status char(25),
lead_source char(25),
prod_serv_category char(25),
opportunity_yn char(1),
forecast_yn char(1),
forecast_amt decimal(16,0),
expect_close_date date,
probability decimal(6,2),
quote_id integer,
win_yn char(1),
close_date date,
order_no char(20),
last_modified date,
last_modified_by char(8)
using btree ;

```

-- FR2047 - CRM Module

```

create table "informix".stcletrd
  letter_code char(8) not null ,
  line_no smallint not null ,
  text char(120)

```

-- FR2047 - CRM Module

```

create table "informix".stcletre
  letter_code char(8),
  description char(30)

```

-- FR2047 - CRM Module

```

create table "informix".stcletsd
  letter_code char(8) not null ,
  line_no smallint,
  special_chars char(20),
  data_source char(255)

```

-- FR2047 - CRM Module

```

create table "informix".stclstd

```

```
list_id integer,  
acct_id integer,  
cust_code char(20),  
contact_id integer
```

-- FR2047 - CRM Module

```
create table "informix".stclsth  
list_id serial not null ,  
user_id char(8),  
description char(60),  
create_date date,  
create_time datetime hour to second
```

-- FR2047 - CRM Module

```
create table "informix".stclitmd  
lead_id integer not null ,  
item_code char(20)
```

-- FR2047 - CRM Module

```
create table "informix".stcphond  
phone_id serial not null ,  
acct_id integer,  
contact_id integer,  
international char(1),  
primary_yn char(1),  
phone_type char(10),  
phone char(18),  
phone_ext char(10),  
phone_digits char(15)
```

-- FR2047 - CRM Module

```
create table "informix".stcprojd  
project_id integer,  
seq_no smallint,  
days_from_start smallint,  
activity_type char(25),  
description char(60),  
duration_days smallint,  
duration_hours smallint,  
duration_minutes smallint,  
activity_role char(25),
```

letter_code char(8)

-- FR2047 - CRM Module

```
create table "informix".stcprpsd
  proposal_id integer,
  line_no smallint,
  item_code char(20),
  quantity decimal(14),
  list_price decimal(14),
  discount_price decimal(14),
  line_net_amount decimal(14)
```

-- FR2047 - CRM Module

```
create table "informix".stcprpse
  proposal_id serial not null ,
  proposal_no char(20),
  create_by char(10),
  proposal_desc char(80),
  acct_id integer,
  addr_id integer,
  contact_id integer,
  warehouse_code char(10),
  proposal_date date,
  discount_pct decimal(10,3),
  proposal_status char(25),
  total_amount decimal(16,2),
  total_discount decimal(16,2),
  net_amount decimal(16,2),
  create_date date
```

-- FR2047 - CRM Module

```
create table "informix".stcqrybe
  query_id serial not null ,
  user_id char(8),
  query_type char(10),
  query_name char(20),
  description char(100),
  sql_statement char(2048),
  list_id integer
```

-- FR2047 - CRM Module

```
create table "informix".stcrclsd
  rep_id integer,
  item_class char(6)
```

-- FR2047 - CRM Module

```
create table "informix".stcrepr
  rep_id serial not null ,
  rep_type char(25),
  level smallint,
  territory_id integer,
  name_courtesy char(12),
  first_name char(20),
  middle_name char(20),
  last_name char(20),
  name_suffix char(12),
  full_name char(63),
  title char(25),
  phone_no char(18),
  cell_phone char(18),
  email_address char(50),
  department char(25),
  role char(25),
  team_id integer,
  supervisor_id integer,
  supervisor_yn char(1),
  quota_type char(25),
  quota decimal(16,2),
```

--- FR3327 - Scheduling Program

```
alter table stcrepr
  add (coordinator char(1));
```

-- FR2047 - CRM Module

```
create table "informix".stcritmd
  rep_id integer,
  item_code char(20)
```

--- FR3327 - Scheduling Program

```
create table "informix".stcrsvcd
  rep_id integer,
```

svc_type char(50)

-- FR2047 - CRM Module

```
create table "informix".stcsnozr
  snooze_id serial not null ,
  description char(25),
  snooze_days interval day to day,
  snooze_hours interval hour to hour,
  snooze_minutes interval minute to minute
```

-- FR2047 - CRM Module

```
create table "informix".stctarcd
  territory_id integer,
  rep_id integer,
  from_area_code smallint,
  to_area_code smallint
  using btree ;
```

-- FR2047 - CRM Module

```
create table "informix".stctcloud
  territory_id integer,
  rep_id integer,
  country_code char(2)
```

-- FR2047 - CRM Module

```
create table "informix".stcteamd
  team_id integer,
  rep_id integer
```

-- FR2047 - CRM Module

```
create table "informix".stcteamr
  team_id serial not null ,
  team_name char(30),
  department char(25),
  level smallint,
  supervisor_id integer,
  team_role char(25),
  territory_id integer
```

-- FR2047 - CRM Module

```
create table "informix".stctstpd
  territory_id integer,
  rep_id integer,
  state_province char(3)
```

-- FR2047 - CRM Module

```
create table "informix".stcttorh
  territory_id serial not null ,
  territory_type char(25),
  territory_name char(30)
```

-- FR2047 - CRM Module

```
create table "informix".stctzipd
  territory_id integer,
  rep_id integer,
  from_post_code char(7),
  to_post_code char(7)
```

--- FR3327 - Scheduling Program

```
create table "informix".stcworke
  doc_no serial not null,
  bill_to      char(20),
  contact char(50),
  contact_phone char(18),
  svc_type char(30),
  po_no char(10),
  project_no char(10),
  service_id char(20),
  svc_address1 char(30),
  svc_address2 char(30),
  svc_city char(30),
  svc_state char(2),
  svc_zip char(10),
  requestor char(50),
  requestor_phone char(18),
  customer_email char(50),
  sched_date date,
  appt_date date,
  appt_time char(10),
```

```
appt_length decimal(4,2),
work_desc char(255),
equipment char(255),
equip_location char(50),
employee_id integer,
facility char(20),
vehicle_id char(20),
install_agrmnt_no char(20),
salesman char(20),
salesman_phone char(18),
install_coord_id char(20),
status char(20),
status_chg_date date,
date_added date,
added_by char(20),
priority char(20)
```

-- FR2047 - CRM Module

```
create table "informix".stcfreqr
  freq_id serial not null ,
  description char(25),
  freq_days integer,
  freq_weeks smallint,
  freq_months smallint,
  freq_years smallint
revoke all on "informix".stcfreqr from "public" as "informix";
```

-- FR2047 - CRM Module

```
create table "informix".stxlkupd
  lookup_id integer,
  sequence_no integer,
  item_value char(50)
```

-- FR2047 - CRM Module

```
create table "informix".stxlkuph
  lookup_id serial not null ,
  module char(3),
  lookup_name char(25),
  lookup_desc varchar(254)
```

-- FR2047 - CRM Module

```
create table "informix".stcprojh
  project_id serial not null ,
  project_type char(25),
  project_name char(25),
  manager_id integer,
  start_date date,
  project_status char(20),
  complete_yn char(1),
  list_id integer,
  created_by char(10),
  created_date date
```

-- FR2047 - CRM Module

```
create table "informix".stxstatr
  country_code char(3),
  state_prov char(2),
  description char(30)
```

-- FR2047 - CRM Module

```
create table "informix".stxzipr
  post_code char(10),
  city char(35),
  county_district char(35),
  state_province char(2),
  county_code char(4),
  state_code char(4),
  country_code char(3)
```

--- FR3327 - Scheduling Program

```
create table "informix".tmpfacir
  facility char(20)
```

--- FR3327 - Scheduling Program

```
create table "informix".tmpsvctr
  svc_type char(50)
```

-- FR2408 - Cross Sells

```
create table "informix".sticrsle -- Inventory Cross Sells hEader table
  item_code char(20)
```


--- MR2481 - Fix DC issues

```
create table "informix".dciserla
  inv_no char(10),
  order_no char(20),
  line_no smallint,
  ship_no smallint,
  lot_no char(20),
  serial_no char(20),
  lot_qty decimal(10),
  cost decimal(12)
```

--- FR2352 - Sales projection screen

```
create table "informix".stiprsld
  item_class char(6),
  period char(2),
  period_year char(4),
  sales_projection decimal(14)
```

--- FR2352 - Sales projection screen

```
create table "informix".stiprsle
  item_class char(6)
```

--- FR2352 - Sales projection screen

```
create table "informix".stiprslr
  period char(2),
  period_year char(4)
```

---FR3347 - New table to capture quality information

```
create table "informix".stiqcend
  (item_code char(20),          -- Item Code
  trans_seq integer,           -- Transaction sequence number
  serial_no char(20),          -- Serial Number
  lot_no char(20),             -- Lot Number
  msmt_qty decimal(10,3),      -- Unit count related to measure taken (ie sample size)
  pass_qty decimal(10,3),      -- Passed Unit count related to measure taken
  fail_qty decimal(10,3),      -- No Pass Unit count related to measure taken
  warn_qty decimal(10,3),      -- Pass with warning count related to measure taken
  msmt_data_type char(1),      -- Type of data collected: N=numeric, C=Character/numeric
  msmt_val_num decimal(10,3),   -- Quality measurement taken, when msmt_data_type_w1=N
  msmt_val_char char(20),      -- Quality measurement taken, when msmt_data_type_w1=C
  comment char(255),           -- Free form comment
```

```

pass_type char(1),      -- Y=pass, N=fail, W=pass with warning
fail_reason char(10),   -- Validate against table 'reason'
warn_reason char(10),   -- Validate against table 'reason'
hold_order char(1),     -- Y=order to be held when pass_type_w1=N, N=no hold
hold_code char(3),      -- Production order hold code to be used when hold_order_w1=Y
user_num1 decimal(14,7), -- User-defined field - numeric 1
user_num2 decimal(14,7), -- User-defined field - numeric 2
user_num3 decimal(14,7), -- User-defined field - numeric 3
user_char1 char(20),    -- User-defined field - character 1
user_char2 char(20),    -- User-defined field - character 2
user_char3 char(20));   -- User-defined field - character 3

```

---FR3347 - New table to capture quality information

create table "informix".stiqcene

```

(item_code char(20),
warehouse_code char(10),
order_no char(20),      -- Production or Purchase Order
order_rel char(3),      -- Production Order release
line_no decimal(4,0),   -- Purchase order line
trans_seq integer,      -- Transaction sequence number
collect_type char(1),   -- S=production order summary, O=routing step/operation, C=component,
P=purchase receipt
collect_stage char(1),  -- R=Prod Rect, L=Labor reporting, C=component issue, Q=Quality

```

transaction Entry, P=Purch rcpt

```

routing_seq char(10),   -- Production Order Routing Sequence
comp_sequence char(10), -- Production Order Component Sequence
component_item char(20), -- Production Order Component Item
inspect_date date,      -- Date of collection
msmt_desc char(30),     -- Measurement description
msmt_qty decimal(10,3), -- Unit count related to measure taken (ie sample size)
pass_qty decimal(10,3), -- Passed Unit count related to measure taken
fail_qty decimal(10,3), -- No Pass Unit count related to measure taken
warn_qty decimal(10,3), -- Pass with warning count related to measure taken
msmt_data_type char(1), -- Type of data collected: N=numeric, C=Character/numeric
msmt_val_num decimal(10,3), -- Quality measurement taken, when msmt_data_type_w1=N
msmt_val_char char(20),  -- Quality measurement taken, when msmt_data_type_w1=C
comment char(255),       -- Free form comment
fail_reason char(10),    -- Validate against table 'reason'
warn_reason char(10),    -- Validate against table 'reason'
hold_order char(1),      -- Y=order to be held when pass_type_w1=N, N=no hold
hold_code char(3),       -- Production order hold code to be used when hold_order_w1=Y

```

```

car_code char(10),          -- Corrective Action Required code
car_compl date,            -- Corrective Action Completion date
car_comment char(255),     -- Corrective Action comment
user_num1 decimal(14,7),   -- User-defined field - numeric 1
user_num2 decimal(14,7),   -- User-defined field - numeric 2
user_num3 decimal(14,7),   -- User-defined field - numeric 3
user_char1 char(20),       -- User-defined field - character 1
user_char2 char(20),       -- User-defined field - character 2
user_char3 char(20),       -- User-defined field - character 3
car_no integer,           -- Next sequential # for CAR alert emailed
entered_by char(8),        -- Display only user id
completed_by char(8),      -- Display only user id
date_added datetime year to second, -- Date/Time added to table
added_by char(20));        -- Added to table by user

```

---FR3347 - New table to capture quality information

```

create table "informix".stiqcmsr
(
  item_code char(20),
  collect_type char(10)
  collect_stage char(1),
  routing_seq char(10),      -- For collect_type O, this is required
  comp_sequence char(10),    -- For collect_type C, this is required
  component_item char(20),    -- For collect_type C, this is required
  msmt_desc char(30),        -- Measurement description
  msmt_data_type char(1),    -- Type of data collected: N=numeric, C=Character/numeric
  low_tol_num decimal(14,7),  -- Quality low tolerance, when msmt_data_type_w1=N
  hi_tol_num decimal(14,7),   -- Quality high tolerance, when msmt_data_type_w1=N
  low_tol_char char(20),     -- Quality low tolerance, when msmt_data_type_w1=C
  hi_tol_char char(20),      -- Quality high tolerance, when msmt_data_type_w1=C
  detail_reqd char(1),       -- Y=Yes(entries needed in woquald also), N=No
  hold_order char(1),        -- Y=order to be held when pass_type_w1=N, N=no hold
  hold_code char(3),         -- Production order hold code to be used when hold_order_w1=Y
  car_code char(10),         -- Corrective Action Required code
  user_hdg_num1 char(30),     -- User-defined field title - numeric 1
  user_hdg_num2 char(30),     -- User-defined field title - numeric 2
  user_hdg_num3 char(30),     -- User-defined field title - numeric 3
  user_hdg_char1 char(30),    -- User-defined field title - character 1
  user_hdg_char2 char(30),    -- User-defined field title - character 2
  user_hdg_char3 char(30));   -- User-defined field title - character 3

```

--- FR2932 - Create new table for Serial/Lot Composition Inquiry

```
create table "informix".stiserlb
```

```
( item_code char(20),  
  warehouse_code char(10),  
  lot_no char(20),  
  serial_no char(20),  
  lot_qty decimal(10),  
  cost decimal(12),  
  vend_code char(20),  
  expiry_date date,  
  po_no char(20),  
  rec_doc_no integer,  
  receipt_date date,  
  recv_qty decimal(10),  
  recv_cost decimal(12),  
  bin_location char(15),  
  so_doc_no integer,  
  so_line_no decimal(4,0),  
  cust_code char(20),  
  shipped_date date,  
  shipped_qty decimal(10,3),  
  mfr_serial char(30),  
  prod_order char(7),  
  prod_order_lot char(3),  
  warr_expire date,  
  warr_matl_cost decimal(10,2),  
  warr_labor_cost decimal(10,2),  
  warr_sales_amt decimal(10,2)
```

--- FR2333 - Alternate Items (aka substitutes)

```
create table "informix".stitemsd
```

```
  item_code    char(20),  
  sub_item_code char(20)
```

--- FR2333 - Alternate Items (aka substitutes)

```
create table "informix".stitemse
```

```
  item_code char(20)
```

--- FR2426 - Add UOM List feature for multiple units of measure

```
create table "informix".stiuomcr (
```

```
  uom_code char(2),  
  uom_desc char(30)) lock mode row;
```

--- FR2426 - Add UOM List feature for multiple units of measure

```
create table "informix".stiuomld (  
    uom_list_code char(10),  
    line_no integer,  
    uom_code char(2),  
    ref_unit_code char(2),  
    quantity decimal(14),  
    uom_type char(1) ) lock mode row;
```

--- FR2426 - Add UOM List feature for multiple units of measure

```
create table "informix".stiuomlr (  
    uom_list_code char(10),  
    uom_list_desc char(30),  
    base_unit_code char(2),  
    qty_dec_places integer) lock mode row;
```

-- MR2974 - Install/Integrate Fitrix Labor Processing 02/22/2013 RJ-FGSS

```
drop table empgroup;  
create table "informix".empgroup  
    group_eg char(5) not null ,  
    desc_eg char(30),  
    add_date_eg date,  
    change_date_eg date  
alter table "informix".empgroup lock mode (row);
```

-- MR2974 - Install/Integrate Fitrix Labor Processing 02/22/2013 RJ-FGSS

```
drop table empteam;  
create table "informix".empteam  
    team_et char(5) not null ,  
    desc_et char(30) not null ,  
    shift1_cap_hr_et decimal(8,4),  
    shift2_cap_hr_et decimal(8,4),  
    shift3_cap_hr_et decimal(8,4),  
    add_date_et date,  
    change_date_et date,  
    rough_rsc_id_et char(10),  
    rough_conv_et decimal(8,4)
```

-- MR2974 - Install/Integrate Fitrix Labor Processing 02/22/2013 RJ-FGSS

```

drop table initlp;
create table "informix".initlp
  init_key_lp char(1) not null ,
  trn_hst_flag_lp char(1) not null ,
  labor_rate_type_lp char(1) not null ,
  auto_lunch_sup_lp char(1) not null ,
  act_cost_type_lp char(1),
  wip_labor_lp char(13),
  wip_ovh_lp char(13),
  wip_outs_lp char(13),
  wip_labor_ctl_lp char(13),
  wip_ovh_ctl_lp char(13),
  wip_outs_ctl_lp char(13),
  py_interface_lp char(1),
  gl_interface_lp char(1),
  setup_done_lp char(1),
  eff_good_pct_lp decimal(5,2),
  eff_warn_pct_lp decimal(5,2),
  eff_poor_pct_lp decimal(5,2)

```

-- MR2974 - Install/Integrate Fitrix Labor Processing 02/22/2013 RJ-FGSS

```

drop table jobclass;
create table "informix".jobclass
  jobclass_jc char(3) not null ,
  desc_jc char(40),
  status_jc char(1),
  department_jc char(3) not null ,
  std_rate_jc smallfloat not null ,
  min_rate_jc smallfloat not null ,
  max_rate_jc smallfloat not null ,
  month_req_jc smallint,
  add_date_jc date,
  change_date_jc date

```

-- MR2974 - Install/Integrate Fitrix Labor Processing 02/22/2013 RJ-FGSS

```

drop table labrtype;
create table "informix".labrtype
  labor_type_lt char(3) not null ,
  desc_lt char(25),
  status_lt char(1),
  direct_indirect_lt char(1),

```

```
setup_run_lt char(1),
run_rework_lt char(1),
acctcd_lt char(13),
add_date_lt date,
change_date_lt date,
py_inc_code_lt char(6)
```

-- MR2974 - Install/Integrate Fitrix Labor Processing 02/22/2013 RJ-FGSS

```
drop table lbrgltran;
create table "informix".lbrgltran
  tran_type_lg char(2),
  whse_lg char(3),
  session_lg integer,
  user_lg char(10),
  trn_hst_no_lg integer,
  reference_lg char(10),
  tran_date_lg date,
  entry_date_lg date,
  entry_time_lg datetime hour to second,
  post_date_lg date,
  post_time_lg datetime hour to second,
  compl_qty_lg decimal(10,3),
  setup_hr_lg decimal(8,4),
  labor_hr_lg decimal(8,4),
  mach_hr_lg decimal(8,4),
  setup_amount_lg money(9,2),
  labor_amount_lg money(9,2),
  mach_amount_lg money(9,2),
  labor_rate_lg decimal(11,4),
  ovh_rate_lg decimal(11,4),
  order_lg char(7),
  order_lot_lg char(3),
  routing_seq_lg char(4),
  workctr_lg char(4),
  department_lg char(3),
  doc_no_lg integer,
  su_debit_acct_lg integer,
  su_debit_alias_lg char(30),
  su_debit_dept_lg char(3),
  su_credit_acct_lg integer,
  su_credit_alias_lg char(30),
```

```

su_credit_dept_lg char(3),
lb_debit_acct_lg integer,
lb_debit_alias_lg char(30),
lb_debit_dept_lg char(3),
lb_credit_acct_lg integer,
lb_credit_alias_lg char(30),
lb_credit_dept_lg char(3),
mc_debit_acct_lg integer,
mc_debit_alias_lg char(30),
mc_debit_dept_lg char(3),
mc_credit_acct_lg integer,
mc_credit_alias_lg char(30),
mc_credit_dept_lg char(3),
ok_to_post_lg char(1),
gl_doc_no_lg integer,
post_flag_lg char(1),
order_type_lg char(2)

```

-- MR2974 - Install/Integrate Fitrix Labor Processing 02/22/2013 RJ-FGSS

```

drop table lbrtrndt;
create table "informix".lbrtrndt
  session_id integer,
  employee_id char(6) not null,
  update_date_id date not null,
  update_time_id datetime hour to second,
  trn_date_id date not null,
  start_time_id char(5),
  stop_time_id char(5),
  trn_type_id char(1),
  order_id char(7),
  order_lot_id char(3),
  routing_seq_id char(4),
  workctr_id char(4),
  compl_qty_id smallfloat,
  expense_acct_id char(12),
  act_lbr_hrs_id smallfloat not null,
  act_mach_hrs_id smallfloat not null,
  act_lbr_cost_id money(11,2),
  act_ovh_cost_id money(11,2),
  std_hours_id smallfloat,
  std_lbr_cost_id money(11,2) not null,

```



```

std_ovh_cost_ld money(11,2) not null,
home_department_ld char(3),
work_department_ld char(3),
machine_ld char(4),
shift_ld char(3),
wage_account_ld char(12),
status_ld char(1),
lbr_or_mach_hr_ld char(1),
scrap_qty_ld smallfloat,
completion_ld char(1),
emp_team_ld char(5),
job_class_ld char(3),
mult_machine_ld smallint,
mult_person_ld smallint,
pct_compl_ld smallfloat,
lbr_type_ld char(3),
interval_ld interval hour to minute,
posting_date_ld date,
posting_time_ld datetime hour to second,
trans_seq_ld smallint,
whse_ld char(10),
emp_group_ld char(5)

```

-- MR2974 - Install/Integrate Fitrix Labor Processing 02/22/2013 RJ-FGSS

```

drop table lbrtrnhd;
create table "informix".lbrtrnhd
    session_lh integer,
    employee_lh char(6),
    shift_lh char(3),
    user_lh char(10),
    trn_date_lh date,
    status_lh char(1),
    start_time_lh char(5),
    stop_time_lh char(5),
    interval_lh interval hour to minute,
    elapsed_hours_lh decimal(7),
    entry_time_lh datetime hour to second,
    entry_date_lh date,
    update_time_lh datetime hour to second,
    update_date_lh date,
    post_time_lh datetime hour to second,

```

post_date_lh date

-- MR2974 - Install/Integrate Fitrix Labor Processing 02/22/2013 RJ-FGSS

drop table lbrtrnprod;

create table "informix".lbrtrnprod

session_to integer,
employee_to char(6),
update_date_to date,
update_time_to datetime hour to second,
trn_date_to date,
start_time_to char(5),
stop_time_to char(5),
trn_type_to char(1),
order_to char(7),
order_lot_to char(3),
routing_seq_to char(4),
workctr_to char(4),
compl_qty_to decimal(10,3),
expense_acct_to char(12),
act_lbr_hrs_to decimal(10,5),
act_mach_hrs_to decimal(10,5),
act_lbr_cost_to decimal(11,2),
act_ovh_cost_to decimal(11,2),
std_hours_to decimal(10,5),
std_lbr_cost_to decimal(10,5),
std_ovh_cost_to decimal(10,5),
home_department_to char(3),
work_department_to char(3),
machine_to char(4),
shift_to char(3),
status_to char(1),
lbr_or_mach_hr_to char(1),
scrap_qty_to decimal(10,3),
completion_to char(1),
emp_team_to char(5),
job_class_to char(3),
mult_machine_to smallint,
mult_person_to smallint,
pct_compl_to decimal(5,2),
lbr_type_to char(3),
interval_to interval hour to minute,

```
posting_date_to date,  
posting_time_to datetime hour to second,  
gl_doc_no_to integer,  
whse_to char(10),  
emp_group_to char(5)
```

-- MR2974 - Install/Integrate Fitrix Labor Processing 02/22/2013 RJ-FGSS

```
drop table resource;  
create table "informix".resource  
  whse_md char(3),  
  resource_md char(10),  
  desc_md char(40),  
  unit_of_meas_md char(2),  
  capacity_md decimal(10,5),  
  cost_per_unit_md money(11,4),  
  add_date_md date,  
  change_date_md date
```

-- MR2974 - Install/Integrate Fitrix Labor Processing 02/22/2013 RJ-FGSS

```
drop table shift;  
create table "informix".shift  
  shift_sf char(3) not null ,  
  desc_sf char(30),  
  status_sf char(1),  
  hold_code_sf char(3),  
  start_time_sf char(5) not null ,  
  stop_time_sf char(5) not null ,  
  clock_on_early_sf char(5) not null ,  
  clock_on_late_sf char(5) not null ,  
  clock_off_early_sf char(5) not null ,  
  clock_off_late_sf char(5) not null ,  
  break_1_min_sf interval minute to minute,  
  break_2_min_sf interval minute to minute,  
  break_3_min_sf interval minute to minute,  
  break_4_min_sf interval minute to minute,  
  break_5_min_sf interval minute to minute,  
  lunch_min_sf interval minute to minute,  
  lunch_early_str_sf char(5),  
  lunch_late_str_sf char(5),  
  shift_premium_sf money(3,3),  
  add_date_sf date,
```

change_date_sf date

-- MR2974 - Install/Integrate Fitrix Labor Processing 02/22/2013 RJ-FGSS

drop table wolbrout;

create table "informix".wolbrout

order_wb char(7),
order_lot_wb char(3),
employee_wb char(6),
trans_number_wb decimal(7,0),
routing_seq_wb char(4),
workctr_wb char(4),
department_wb char(4),
machine_wb char(4),
team_wb char(5),
compl_qty_wb decimal(10,3),
compl_qty_tmp_wb decimal(10,3),
setup_hr_wb decimal(8,4),
setup_hr_tmp_wb decimal(8,4),
labor_hr_wb decimal(8,4),
labor_hr_tmp_wb decimal(8,4),
mach_hr_wb decimal(8,4),
mach_hr_tmp_wb decimal(8,4),
setup_amt_wb money(13,4),
labor_amt_wb money(13,4),
mach_amt_wb money(13,4),
outs_amt_wb money(13,4),
status_wb char(1),
comments_wb char(30),
date_trans_wb date,
time_trans_wb datetime hour to second,
gl_doc_number_wb integer

-- MR2974 - Install/Integrate Fitrix Labor Processing 02/22/2013 RJ-FGSS

drop table wolbrsumm;

create table "informix".wolbrsumm

order_wa char(7),
order_lot_wa char(3),
employee_wa char(6),
trans_number_wa decimal(7,0),
tran_type_wa char(1),
comments_wa char(30),

trans_date_wa date,
add_date_wa date,
change_date_wa date,
session_wa integer

--- FR2411 - archive order entry tables into history tables

```
create table "informix".arofedce
  doc_no integer not null ,
  fed_cert char(1),
  contact char(30),
  fed_agency char(1),
  fed_contract_no char(30),
  fed_eu_support char(1),
  fed_comrcl_tc char(1)
```

--- FR2411 - archive order entry tables into history tables

```
create table "informix".aroinvce
  doc_no integer not null ,
  order_no char(20),
  bill_to_code char(20),
  sell_to_code char(20),
  ship_to_code char(6),
  inv_doc_no integer,
  stage char(3),
  inv_no char(10),
  inv_date date,
  inv_printed char(1),
  ok_to_post char(1) not null ,
  terms_code char(6),
  terms_approval char(6),
  pay_method char(6),
  payment char(1),
  card_no char(20),
  exp_date char(5),
  check_no char(8),
  fob_point char(15),
  ship_via char(15),
  ship_weight decimal(10),
  freight_doc char(15),
  st_tx_code char(6),
  co_tx_code char(6),
```

ci_tx_code char(6),
st_tx_rate decimal(6),
co_tx_rate decimal(6),
ci_tx_rate decimal(6),
tax_rate decimal(6),
trd_ds_rate decimal(6),
item_amount decimal(12),
discountable decimal(12),
trd_ds_amount decimal(12),
taxable decimal(12),
st_tx_amount decimal(12),
co_tx_amount decimal(12),
ci_tx_amount decimal(12),
frght_amount decimal(12),
total_amount decimal(12),
td_ds_acct integer,
st_tx_acct integer,
co_tx_acct integer,
ci_tx_acct integer,
freight_acct integer,
asset_acct integer,
td_ds_dept char(3),
st_tx_dept char(3),
co_tx_dept char(3),
ci_tx_dept char(3),
freight_dept char(3),
asset_dept char(3),
mtaxg_code char(6),
tax_amount decimal(12),
currency_code char(3),
curr_rate_type char(6),
currency_rate decimal(16),
batch_id integer,
ship_terms char(15),
mtax_freight char(6),
settle_decl_code char(8),
decline_message char(60),
settled char(1),
cc_batch_id char(12),
deposit_applied decimal(12,2),
restock_fee decimal(8,2),

handling_fee decimal(8,4)

--- FR2411 - archive order entry tables into history tables

create table "informix".aroordrd

doc_no integer not null ,
line_no smallint,
kit_group smallint,
kit_line_no smallint,
last_ship_line smallint,
line_type char(3),
like_type char(3),
hi_stage char(3),
lo_stage char(3),
cm_dm_reason char(3),
our_po_no char(20),
sls_psn_code char(6),
warehouse_code char(10),
item_code char(20),
desc1 char(30),
desc2 char(30),
alias_code char(20),
vend_code char(20),
interchanged char(1),
serialized char(1),
td_disc_allowed char(1),
tax char(1),
ordr_qty decimal(14),
back_qty decimal(14),
commit_qty decimal(14),
sell_unit char(2),
unit_factor decimal(6),
price decimal(12),
price_code integer,
tax_amount decimal(12),
net_amount decimal(14),
ship_weight decimal(14),
inv_acct integer,
inv_dept char(3),
sls_acct integer,
sls_dept char(3),
cog_acct integer,

cog_dept char(3),
intl_lic_no char(30),
price_lock char(1),
release_qty decimal(14),
resale_price decimal(18),
mtaxg_code char(6)

--- **FR2411 - archive order entry tables into history tables**

create table "informix".aroordre

doc_no integer not null,
orig_doc_no integer,
order_no char(20),
inv_doc_no integer,
inv_no char(10),
po_no char(24),
pic_ticket_no smallint,
next_kit_group smallint,
ack_printed char(1),
order_type char(3) not null,
like_type char(3) not null,
order_status char(3) not null,
hi_stage char(3) not null,
lo_stage char(3) not null,
bo_allowed char(1),
recur_unit char(1),
recur_every smallint,
recur_times smallint,
recur_through date,
prev_recur date,
next_recur date,
num_releases smallint,
release_type char(3),
order_date date,
to_ship_date date,
alloc_date date,
ship_date date,
complete_date date,
warehouse_code char(10),
department char(3),
sls_psn_code char(6),
cust_code char(20),

ship_to_code char(6),
bill_to_code char(20),
bus_name char(30),
contact char(20),
address1 char(30),
address2 char(30),
city char(20),
state char(2),
zip char(10),
country char(2),
terms_code char(6),
terms_approval char(6),
pay_method char(6),
payment char(1),
card_no char(20),
exp_date char(5),
card_holder char(20),
check_no char(8),
trd_ds_code char(6),
trd_ds_type char(1),
trd_ds_rate decimal(6),
multi_shipto char(1),
tax_rate decimal(6),
staging_area char(6),
fob_point char(15),
ship_via char(30),
ship_weight decimal(14),
item_amount decimal(14),
discountable decimal(12),
trd_ds_amount decimal(14),
taxable decimal(12),
tax_amount decimal(14),
frght_amount decimal(14),
total_amount decimal(14),
create_date date,
create_time char(8),
create_id char(8),
l_mod_date date,
l_mod_time char(8),
l_mod_id char(8),
system_order char(1),

spr_no char(7),
cust_ord_date date,
cust_po_date date,
fact_ack_date date,
fact_rec_date date,
moto_rec_date date,
sent_to_wwop char(1),
mtaxg_code char(6),
intl_order char(1),
intl_lic_no char(30),
currency_code char(3),
curr_rate_type char(6),
currency_rate decimal(16),
edi_sent char(1),
blo_exp_date date,
dpas_rating char(2),
resale_cust char(6),
resale_po char(30),
actual_frght_amt decimal(12),
orig_frght_amt decimal(12),
ship_terms char(15),
residential_cust char(1),
email char(50),
ups_account char(10),
mtax_freight char(6),
auth_amt decimal(10,2),
auth_code char(8),
auth_date date,
decline_code char(8),
decline_message char(60),
ship_complete char(1),
contract_no char(20),
multiple_orders char(1),
deposit decimal(10,2),
docs_sent date,
destination char(30),
consignee_name char(20),
consignee_addr1 char(30),
consignee_addr2 char(30),
consignee_city char(20),
consignee_state char(2),

consignee_zip char(10),
 consignee_country char(20),
 notify_name char(20),
 notify_info char(240),
 truck_bol char(20),
 route_code char(10),
 resale_no char(15),
 resale_expiry date,
 rma_reason char(6),
 base_doc_no integer,
 rma_doc_no integer,
 order_description char(80),
 ready_to_invoice char(1),
 required_date date,
 default_rma_type char(3),
 restock_fee decimal(8,2),
 ship_to_name char(30),
 one_time_cust char(1),
 ship_type char(6),
 restock_percent decimal(6,2),
 restock_amount decimal(8,2),
 rma_status_code char(20),
 handling_fee decimal(8,4),

--- FR2411 - archive order entry tables into history tables

create table "informix".aroordrh

doc_no integer,
 ship_type char(6),
 service_type char(50),
 package_type char(30),
 weight decimal(9,2),
 weight_unit char(5),
 cost_amount decimal(9,2),
 charge_amount decimal(9,2),
 estimate_date date,
 commit_days smallint

--- FR2411 - archive order entry tables into history tables

create table "informix".aroordsd

doc_no integer,
 line_no smallint,

dock_receipt date,
to_repair_ctr date,
repair_complete date,
to_customer date,
turn_time smallint,
sched_ship date,
rep_rep_prt date,
symptom char(40),
serial_no char(20),
last_chg_user char(8),
last_chg_date date

--- **FR2411 - archive order entry tables into history tables**

create table "informix".aroorwtd

doc_no integer,
line_no smallint,
ship_no smallint,
po_doc_no integer,
po_line_no smallint,
seq_no smallint,
unit_no smallint,
uom char(6),
gross decimal(12),
tare decimal(12),
net decimal(12)

--- **FR2411 - archive order entry tables into history tables**

create table "informix".aroshpdd

doc_no integer not null ,
line_no smallint not null ,
ship_no smallint,
kit_group smallint,
stage char(3),
ship_qty decimal(14),
sell_unit char(2),
ship_weight decimal(14),
commit_qty decimal(14),
pic_ticket_no smallint,
mfs_printed char(1),
inv_doc_no integer,

```

sell_to_code char(20),
ship_to_code char(6),
bill_to_code char(20),
item_code char(20),
warehouse_code char(10),
stock_location char(12),
price_group char(6),
price decimal(12),
orig_price decimal(12),
retail_price decimal(12),
price_approval char(6),
tax_rate decimal(6),
tax_amount decimal(12),
net_amount decimal(14),
item_cost decimal(12),
gross_margin decimal(10),
new_date date,
bko_date date,
ord_date date,
pic_date date,
shp_date date,
inv_date date,
pst_date date,
can_date date,
proj_ship_date date,
request_date date,
reject_date date,
actual_ship_date date,
fact_sched_date date,
ship_via_cd char(3),
mtaxg_code char(6),
po_doc_no integer,
ship_lookup char(30)

```

--- FR2411 - archive order entry tables into history tables

```

create table "informix".aroshpfr
  pack_lookup_key char(40),
  pack_tracking_no char(40),
  pack_freight decimal(10,2),
  pack_shp_date char(10),
  pack_shp_boxes char(4),

```

```

pack_void_flag char(1),
pack_del_date date,
pack_del_city char(20),
pack_del_state char(2),
pack_del_type char(20),
pack_timestamp datetime year to second
    default current year to second,
order_no char(20),
shp_weight decimal(9,1),
shp_cost decimal(10,2),
ship_type char(6),
package_type char(30),
stage char(5),
inv_doc_no integer

```

--- **FR2411 - archive order entry tables into history tables**

```

create table "informix".aroshtxd
    doc_no integer not null ,
    line_no smallint not null ,
    ship_no smallint,
    tax_code char(6),
    tax_rate decimal(6),
    net_amt decimal(12),
    tax_amt decimal(12)

```

--- **FR2411 - archive order entry tables into history tables**

```

create table "informix".arotrckd
    contract_no char(20),
    doc_no integer,
    line_no smallint,
    po_doc_no integer,
    fwdr_invoice char(20),
    sales_basis char(5),
    loadg_location char(30),
    port_of_exit char(30),
    gross_weight decimal(10),
    net_weight decimal(10),
    tare_weight decimal(10),
    container_no char(20),

```

bill_lading char(20),
ocean_bol char(20),
booking_no char(20),
vessel char(25),
voyage char(25),
cutoff_date date,
loadg_date date,
release_date date,
est_depart date,
est_arrive date,
pymt_due_date date,
est_demurrage decimal(10,2),
insurance decimal(10,2),
relse_rqstd date,
relse_rcvd date,
draft_obl_rcvd date,
consignee_name char(20),
consignee_addr1 char(30),
consignee_addr2 char(30),
consignee_city char(20),
consignee_state char(2),
consignee_zip char(10),
consignee_country char(20),
notify_name char(20),
notify_info char(240),
gross_mt decimal(8,2),
net_mt decimal(8,2),
tare_mt decimal(8,2),
forwarder_name char(20),
carrier char(20),
send_docs char(20),
container_size char(20),
transshipment1 char(20),
vessel1 char(20),
voyage1 char(10),
eta1 date,
etd1 date,
transshipment2 char(20),
vessel2 char(20),
voyage2 char(10),
eta2 date,

```

etd2 date,
transshipment3 char(20),
vessel3 char(20),
voyage3 char(10),
eta3 date,
etd3 date,
transshipment4 char(20),
vessel4 char(20),
voyage4 char(10),
eta4 date,
etd4 date,
transshipment5 char(20),
vessel5 char(20),
voyage5 char(10),
eta5 date,
etd5 date,
port1 char(10),
port2 char(10),
port3 char(10),
port4 char(10),
port5 char(10),
seal_no char(10)

```

--- **FR2411 - archive order entry tables into history tables**

```

create table "informix".stoarctr
  sys_table_name char(20),
  arch_table_name char(20),
  join_column char(20)

```

--- **FR2523 - display on report the commission reductions**

```

create table "informix".stocompd (
  sls_psn_code char(6),
  past_due_days smallint,
  comm_reduce_pct decimal(5,2)

```

--- **FR2425 - Fedex Cost Reconciliation**

```

create table "informix".stofdxdt
( bill_to_acct char(20),
  invoice_date date,
  invoice_no char(20),
  store_id char(20),

```


orig_amt_due decimal(10,2),
curr_balance decimal(10,2),
payor char(15),
grnd_trk_pfx char(10),
grnd_trk_id char(20),
trans_chg_amt decimal(10,2),
net_chg_amt decimal(10,2),
service_type char(30),
ground_service char(20),
shipment_date date,
pod_delvry_date date,
pod_delvry_time char(5),
pod_svc_area char(2),
pod_sign_desc char(20),
actual_weight decimal(10,2),
actual_weight_unit char(3),
rated_weight decimal(10,2),
rated_weight_unit char(3),
number_pieces integer,
bundle_number integer,
meter_number char(10),
recpnt_name char(50),
recpnt_company char(50),
recpnt_addr1 char(30),
recpnt_addr2 char(30),
recpnt_city char(30),
recpnt_state char(2),
recpnt_zip char(12),
recpnt_country char(30),
shippr_company char(50),
shippr_name char(50),
shippr_addr1 char(30),
shippr_addr2 char(30),
shippr_city char(30),
shippr_state char(2),
shippr_zip char(12),
shippr_country char(30),
cust_ref_1 char(30),
cust_ref_2 char(30),
cust_ref_3 char(30),
dept_ref char(30),

cust_ref_1u char(30),
cust_ref_2u char(30),
cust_ref_3u char(30),
dept_refu char(30),
rma_number char(20),
orig_rcpnt_addr1 char(30),
orig_rcpnt_addr2 char(30),
orig_rcpnt_city char(30),
orig_rcpnt_state char(2),
orig_rcpnt_zip char(12),
orig_rcpnt_country char(30),
zone_code char(5),
entry_date date,
entry_number integer,
customs_value decimal(10,2),
customs_val_cur char(10),
declared_value decimal(10,2),
declared_val_cur char(10),
commod_desc1 char(30),
commod_cntry1 char(10),
commod_desc2 char(30),
commod_cntry2 char(10),
commod_desc3 char(30),
commod_cntry3 char(10),
commod_desc4 char(30),
commod_cntry4 char(10),
cur_conv_date date,
cur_conv_rate decimal(12),
multi_no integer,
multi_units integer,
multi_wght decimal(10),
multi_ship_chg decimal(10,2),
multi_ship_wght decimal(10),
grnd_trk_ac_disc decimal(10,2),
grnd_trk_ac_gros decimal(10,2),
chg_desc_01 char(30),
chg_amt_01 decimal(10,2),
chg_desc_02 char(30),
chg_amt_02 decimal(10,2),
chg_desc_03 char(30),
chg_amt_03 decimal(10,2),

chg_desc_04 char(30),
chg_amt_04 decimal(10,2),
chg_desc_05 char(30),
chg_amt_05 decimal(10,2),
chg_desc_06 char(30),
chg_amt_06 decimal(10,2),
chg_desc_07 char(30),
chg_amt_07 decimal(10,2),
chg_desc_08 char(30),
chg_amt_08 decimal(10,2),
chg_desc_09 char(30),
chg_amt_09 decimal(10,2),
chg_desc_10 char(30),
chg_amt_10 decimal(10,2),
chg_desc_11 char(30),
chg_amt_11 decimal(10,2),
chg_desc_12 char(30),
chg_amt_12 decimal(10,2),
chg_desc_13 char(30),
chg_amt_13 decimal(10,2),
chg_desc_14 char(30),
chg_amt_14 decimal(10,2),
chg_desc_15 char(30),
chg_amt_15 decimal(10,2),
chg_desc_16 char(30),
chg_amt_16 decimal(10,2),
chg_desc_17 char(30),
chg_amt_17 decimal(10,2),
chg_desc_18 char(30),
chg_amt_18 decimal(10,2),
chg_desc_19 char(30),
chg_amt_19 decimal(10,2),
chg_desc_20 char(30),
chg_amt_20 decimal(10,2),
chg_desc_21 char(30),
chg_amt_21 decimal(10,2),
chg_desc_22 char(30),
chg_amt_22 decimal(10,2),
chg_desc_23 char(30),
chg_amt_23 decimal(10,2),
chg_desc_24 char(30),

```
chg_amt_24 decimal(10,2),
chg_desc_25 char(30),
chg_amt_25 decimal(10,2)
```

--- **FR2433 - Item Notes with Selective Print**

```
create table "informix".stoitmpd
  prog_no smallint,
  item_code char(20),
  prt_notes char(1)
```

--- **FR2433 - Item Notes with Selective Print**

```
create table "informix".stoitmpr
  prog_no smallint,
  prt_module char(8),
  prt_program char(8),
  prt_description char(30),
  primary key (prog_no) constraint "informix".pk_oitmpr
revoke all on "informix".stoitmpr from "public" as "informix";
```

--- **MR2400 - Multiple Orders Streamline process**

```
create table "informix".stomlsoe
  doc_no      integer,
  init_doc_no integer,
  next_doc_no integer,
  inv_doc_no  integer
```

--- **MR2879 - implement table stooedoc to control oe_doc_no calls.**

```
create table "informix".stooedoc(
  next_doc_no serial not null
) lock mode row;
```

--- **FR3099 - Add contact name and number**

```
alter table stoordre add
  contact_name char(20),
  contact_phone char(20)
```

--- **FR2362 - Interface to Fedex and UPS web services**

```
create table "informix".stoordrh
  doc_no integer,
  ship_type char(6),
```

```

service_type char(50),
package_type char(30),
weight decimal(9,2),
weight_unit char(5),
cost_amount decimal(9,2),
charge_amount decimal(9,2),
estimate_date date,
commit_days smallint

```

-- FR3235 - Blanket SOs

```

create table "informix".stoorrid
  doc_no integer,
  line_no smallint,
  ship_no smallint,
  rlse_qty decimal(10,4),
  balance decimal(10,4),
  ship_date date,
  required_date date

```

--- FR2332 - Streamline order entry

```

create table "informix".stopicke
  doc_no integer,
  line_no smallint,
  ship_no smallint,
  orig_ship_qty decimal(14)

```

--- FR2332 - Streamline order entry

```

create table "informix".stoprompt
  prompt_type char(10),
  doc_no_list char(500)
create unique index "informix".i1stoprompt on stoprompt (prompt_type);

```

-- FR2412 - Recurring Orders.

```

create table "informix".storecod (
  reorder_doc_no serial,
  order_doc_no integer,
  cycle_frequency char(1),
  cycle_code char(1),
  start_date date,
  end_date date,

```

```
current_price char(1),  
next_reorder_date date
```

-- FR3323 - Customer Credit Management

```
create table "informix".storlccd  
  cust_code char(20),  
  rel char(1),  
  doc_no integer,  
  order_no char(20),  
  order_date date,  
  total_amount decimal(14),  
  hold_code char(6)
```

-- FR3323 - Customer Credit Management

```
create table "informix".storlcce  
  cust_code char(20)
```

-- FR3235 - Blanket SOs

```
create table "informix".storlsed  
  doc_no integer,  
  line_no smallint,  
  item_code char(20),  
  ord_r_qty decimal(14),  
  rlse_qty decimal(14),  
  remain_qty decimal(14),  
  qty_to_rlse decimal(14)
```

-- FR3235 - Blanket SOs

```
create table "informix".storlsee  
  doc_no serial,  
  so_doc_no integer,  
  so_no char(20),  
  rlse_date date,  
  ship_date date,  
  required_date date
```

--- FR2394 - RMA Status

```
create table "informix".stormasr  
  status_code char(20),  
  description char(40)
```

--- FR2362 - Interface to Fedex and UPS web services

```
alter table stoshipd add  
    ship_lookup char(30)
```

-- FR2415 - Create rebate import mapping tables

```
create table "informix".stoshbrd  
    doc_no integer,  
    line_no integer,  
    ship_no integer,  
    item_code char(20),  
    rebate_cost decimal(11,4),  
    vend_code char(20),  
    posted char(1)
```

-- FR2413 - Split Payment Terms

```
create table "informix".stospltd (  
    order_doc_no integer,  
    orig_inv_doc_no integer,  
    split_inv_doc_no integer
```

-- FR2413 - Split Payment Terms

```
create table "informix".stotermd (  
    split_terms_code char(6), -- join to stotermh  
    line_no smallint, -- for ordering of entries  
    terms_code char(6), -- terms code from strtermr  
    terms_percent decimal(6) -- percent of order at this term
```

-- FR2413 - Split Payment Terms

```
create table "informix".stotermh (  
    split_terms_code char(6),  
    split_terms_desc char(30),  
    cm_memo_acct integer, -- for debit account for credit memo attached to the original invoice, the  
credit account for the split out invoices if not left blank  
    split_ar_acct integer -- Debit account for split out invoices if not left blank.
```

--- FR2362 - Interface to Fedex and UPS web services

```
create table "informix".stoupsws  
    ship_type char(6),  
    service_name char(30),  
    version_id_svc char(10),
```

```

    version_id_major char(10),
    version_id_int char(10),
    version_id_minor char(10)
alter table stoupsws lock mode (row);

```

-- FR2415 - Create rebate import mapping tables

```

create table "informix".stumfrbe
    vend_code char(20),
    contract char(20),
    description char(60),
    start_date date,
    end_date date
alter table stumfrbe lock mode (row);
create unique index "informix".i1stumfrbe on "informix".stumfrbe (vend_code, contract) using btree ;

```

-- FR3223 - Blanket POs

```

create table "informix".stuorrlid
(
    doc_no integer,
    line_no smallint,
    ship_no smallint,
    rlse_qty decimal(10,4),
    balance decimal(10,4),
    ship_date date,
    required_date date

```

-- FR3223 - Blanket POs

```

create table "informix".sturlsed
    doc_no integer,
    line_no smallint,
    item_code char(20),
    ord_r_qty decimal(14),
    rlse_qty decimal(14),
    remain_qty decimal(14),
    qty_to_rlse decimal(14)
);

```

-- FR3223 - Blanket POs

```

create table "informix".sturlsee
    doc_no serial,

```



```
po_doc_no integer,  
po_no char(20),  
rlse_date date,  
ship_date date,  
required_date date
```

--- FR3447.sql - new tabels and rows for Production Line Inventory

```
drop table "informix".itemline;  
create table "informix".itemline  
  whse_il char(10),  
  item_il char(20),  
  prod_line_il char(10),  
  location_il char(20),  
  cycle_time_il decimal(12,6),  
  setup_time_il decimal(12,6),  
  teardown_time_il decimal(12,6),  
  add_date_il date,  
  change_date_il date
```

--- FR3447.sql - new tabels and rows for Production Line Inventory

```
drop table "informix".prodline;  
create table "informix".prodline  
  whse_pd char(10),  
  prod_line_pd char(10),  
  description_pd char(30),  
  type_pd char(1),  
  std_efficiency_pd decimal(8,4),  
  avg_efficiency_pd decimal(8,4),  
  add_date_pd date,  
  change_date_pd date
```

MODIFIED TABLES

---FR1493 - Create tables, alter tables, add rows to support PS module

```
alter table "informix".deptcap add override_dc char(1);
```

--- FR1493 - Create tables, alter tables, add rows to support PS module RJ-FGSS }

```
alter table "informix".workctrkap add override_wp char(1);
```

--- FR1493 - Create tables, alter tables, add rows to support PS module RJ-FGSS }

```
alter table "informix".machcap add override_mp char(1);
```

--- **FR3376 - expand mrprecom column to char(20) }**

alter table mrprecom modify order_rc char(20);

--- **FR1493 - Create tables, alter tables, add rows to support PS module RJ-FGSS }**

alter table "informix".teamcap add override_ep char(1);

--- **FR2423 - Check Numbering**

alter table stxckrgr add next_check_no char(10);

--- **FR2729 - Add post_err_notify to set posting error pop-up display**

alter table stxcntrc add(post_err_notify char(1));

--- **FR3342 - Add gwc_server and help text for the gwc_server field;**

alter table stxcntrc add (gwc_server char(60));

--- **FR2362 - Interface to Fedex and UPS web services**

alter table stxcntrc add

--- **FR2340 - Email alert for special orders**

alter table stxevtms modify (msg_header char(80));

--- **MR2697 - Extend subject line**

alter table stxevtms modify(msg_header char(200));

--- **FR2664 - Add ability to run report and attach file**

alter table stxevtms add x_priority smallint;

--- **FR2494 - Salesperson login logic**

alter table stxinfor add

--- **FR2424 - Positive Pay file**

alter table stpcashe add

pospay_sent char(1),

send_pospay char(1)

--- **FR3484 - Real Time Processing**

alter table stpcntrc add (auto_invce_post char(1));

alter table stpcntrc add (auto_check_post char(1));

alter table stpcntrc add (auto_vdcheck_post char(1));

alter table stpcntrc add (auto_invce_post_d char(1));

```
alter table stpcntrc add (auto_check_post_d char(1));
alter table stpcntrc add (auto_vdchk_post_d char(1));
```

-- FR3031 Add support for Job Shop

```
alter table stpinvcd
  drop (prod_order);
alter table stpinvcd
  drop (prod_order_lot);
alter table stpinvcd
  drop (element);
alter table stpinvcd
  add (prod_order char(7),
      prod_order_lot char(3),
      element char(20));
```

-- MR3154 - Missing columns in DC

```
alter table stppytor add (cc_ap_acct_no integer);
alter table stppytor add (cc_ap_department char(3));
```

--- FR2284 - Display of purchases totals in vendor record

```
alter table stpvendr add (
  ytd_purchases decimal(14),
  lifetime_purchases decimal(14))
```

--- FR2416 - Total Open Purchase Orders

```
alter table stpvendr add(open_po_amt decimal(14));
```

-- FR2415 - Create rebate import mapping tables

```
alter table stpvendr add (reb_recv_acct_no integer);
```

-- MR3154 - Missing columns in DC

```
alter table stpvendr add (cc_template char(20));
```

--- FR2348 - Auto assign customer code

alter table strcntrc add

```
  cust_assign char(1),
  cust_next integer
```

--- FR2420 - Auto Sequence ship_to_code

```
alter table strcntrc
```

add (ship_to_assign char(1), ship_to_next integer);

-- FR3323 - Customer Credit Management

```
alter table strcntrc add (  
    credit_manager char(8),  
    credit_check char(1),  
    over_credit_pct decimal(4,2),  
    hold_code char(6)
```

--- FR3484 - Real Time Processing

```
alter table strcntrc add (auto_invce_print char(1));  
alter table strcntrc add (auto_invce_post char(1));  
alter table strcntrc add (auto_rcpt_post char(1));  
alter table strcntrc add (auto_deposit_print char(1));  
alter table strcntrc add (auto_deposit_post char(1));  
alter table strcntrc add (auto_invce_post_d char(1));  
alter table strcntrc add (auto_rcpt_post_d char(1));  
alter table strcntrc add (auto_dep_post_d char(1));
```

--- FR2177 - Display of sales totals in customer record

```
alter table strcustr add (  
    ytd_sales decimal(14),  
    lifetime_sales decimal(14)  
    (select sum(stropend.orig_amount)  
    from stropend, stxperdr, stgcntrc  
    where stropend.inv_date >= stxperdr.start_date and  
          stropend.inv_date <= (stxperdr.start_date + 364) and  
          stxperdr.period_year = stgcntrc.curr_year and  
          stxperdr.period = "01" and  
          stropend.cust_code = strcustr.cust_code)  
where strcustr.ytd_sales is null;  
    (select sum(stropend.orig_amount)  
    from stropend  
    where stropend.cust_code = strcustr.cust_code)  
where strcustr.lifetime_sales is null;
```

where strcustr.lifetime_sales is null;

where strcustr.ytd_sales is null;

--- FR2337 - Multiple telephone numbers in customer record

```

alter table strcustr add (
    label1 char(10),
    phone1 char(20),
    label2 char(10),
    phone2 char(20),
    label3 char(10),
    phone3 char(20),
    label4 char(10),
    phone4 char(20),
    label5 char(10),
    phone5 char(20),
    label6 char(10),
    phone6 char(20),
    label7 char(10),
    phone7 char(20),
    label8 char(10),
    phone8 char(20),
    label9 char(10),
    phone9 char(20),
    label10 char(10),
    phone10 char(20)

```

-- FR2413 - Split Payment Terms

```

alter table "informix".strcustr add split_terms_code char(6);

```

-- FR2568 - Account Type and Source

```

alter table strcustr add (
    account_type char(25),
    account_source char(25) );

```

-- FR2631 - Add strcustr.warehouse_code to use as default in oe:i_order

```

alter table strcustr add (warehouse_code char(10));

```

-- FR3323 - Customer Credit Management

```

alter table strcustr add (
    duns_no char(9),
    active char(1),
    acct_established date,
    collection_contact char(30),
    collection_phone char(20),
    credit_check char(1),

```

```

over_credit_pct decimal(4,2),
credit_approve_dt date,
credit_approve_by char(8),
avg_pay_days smallint,
last_payment decimal(12,2),
open_orders decimal(12,2),
hold_code char(6)
alter table strcustr modify (
  label1 char(30),
  label2 char(30),
  label3 char(30),
  label4 char(30),
  label5 char(30),
  label6 char(30),
  label7 char(30),
  label8 char(30),
  label9 char(30),
  label10 char(30)

```

--- FR2249 - Expand customer PO to 20 characters

```
alter table strinvce modify (po_no char(20));
```

--- FR3008 - Expand customer PO to 20 characters

```
alter table stropend modify (po_no char(20));
```

-- FR2631 - Add strcustr.warehouse_code to use as default in oe:i_order

```
alter table strshipr add (warehouse_code char(10));
```

-- MR2990 - Integrate Fitrix Standard Costing 02/25/13 RJ-FGSS

```
alter table "informix".billcost
  modify (comp_sequence_cn char(10));
```

{ MR3486 - Add columns to cfgselser 05/14/2014 RJ-FGSS }

```
alter table cfgselser
  add (item_ge char(20));
```

--- MR2562 - Add columns to cfgtplqst, cfgtplans for numeric answers

```
alter table cfgtplans
  add (answer_num_ga decimal(11,4));
```

--- MR2562 - Add columns to cfmtplqst, cfmtplans for numeric answers

```
alter table cfmtplopt
  add (option_type_go char(1));
```

--- MR2562 - Add columns to cfmtplqst, cfmtplans for numeric answers

```
alter table cfmtplqst
  add (ans_data_type_gq char(1),
      ans_precision_gq smallint);
```

-- MR2990 - Integrate Fitrix Standard Costing 02/25/13 RJ-FGSS

```
alter table initpc
  add (setup_elmt_pc char(20));
```

-- FR3031 Add support for Job Shop

```
alter table ordertypesc
  add (po_type_ts char(3));
```

-- MR3222 - add Invoice Price calculation method to ordertypesc

```
alter table ordertypesc
  add (price_calc_type_ts char(1));
```

-- FR3031 Add support for Job Shop

```
alter table routoper drop (ext_desc_ro);
alter table routoper
  add (ext_desc_ro varchar(255));
```

{ MR3119 - Add columns to wosummhst, wocomphst, worouthst previously added to wosumm, wocomp, worout 10/21/2013 RJ-FGSS }

```
alter table wocomphst
  add (desc1_wm char(30) before op_where_used_wm,
      desc2_wm char(30) before op_where_used_wm);
alter table wocomphst drop shrinkage_wm;
alter table wocomphst
  add (shrinkage_wm decimal(5,2) before purge_date_wm,
      ext_desc_wm varchar(255) before purge_date_wm,
      unit_price_wm decimal(11,3) before purge_date_wm,
      include_price_wm char(1) before purge_date_wm,
      nonstk_cost_wm decimal(11,4) before purge_date_wm,
      ack_print_wm char(1) before purge_date_wm,
      quo_print_wm char(1) before purge_date_wm,
```

```
    inv_print_wm char(1) before purge_date_wm,  
    po_no_wm integer before purge_date_wm,  
    po_line_wm smallint before purge_date_wm,  
    taxable_wm char(1) before purge_date_wm);
```

-- FR3031 Add support for Job Shop

```
alter table wocomp  
    drop (ext_desc_wm);  
alter table wocomp  
    drop (unit_price_wm);  
alter table wocomp  
    drop (include_price_wm);  
alter table wocomp  
    drop (nonstk_cost_wm);  
alter table wocomp  
    drop (ack_print_wm);  
alter table wocomp  
    drop (quo_print_wm);  
alter table wocomp  
    drop (inv_print_wm);  
alter table wocomp  
    drop (po_no_wm);  
alter table wocomp  
    drop (po_line_wm);
```

```
alter table wocomp  
    add (ext_desc_wm varchar(255),  
        unit_price_wm decimal(11,3),  
        include_price_wm char(1),  
        nonstk_cost_wm decimal(11,4),  
        ack_print_wm char(1),  
        quo_print_wm char(1),  
        inv_print_wm char(1),  
        po_no_wm integer,  
        po_line_wm smallint);
```

-- FR3153 Add taxable support for Job Shop Orders

```
alter table wocomp  
    add (taxable_wm char(1));  
--- FR2280 - Allow duplicate lot number per item/warehouse  
alter table woissloc add
```



```
(  
    serId_seq_wk integer
```

--- FR2127 - Add Product Configurator module to Fitrix

```
alter table womisc  
    add (option_wb char(10));
```

-- FR3031 Add support for Job Shop

```
alter table womisc  
    drop (ext_desc_wb);  
alter table womisc  
    drop (unit_price_wb);  
alter table womisc  
    drop (include_price_wb);  
alter table womisc  
    drop (ack_print_wb);  
alter table womisc  
    drop (quo_print_wb);  
alter table womisc  
    drop (inv_print_wb);  
alter table womisc  
    add (ext_desc_wb varchar(255),  
        unit_price_wb decimal(11,3),  
        include_price_wb char(1),  
        packet_print_wb char(1),  
        ack_print_wb char(1),  
        quo_print_wb char(1),  
        inv_print_wb char(1));
```

-- FR3153 Add taxable support for Job Shop Orders

```
alter table womisc  
    add (taxable_wb char(1));
```

--- FR2280 - Allow duplicate lot number per item/warehouse

```
alter table worecloc add  
    serId_seq_wf integer
```

-- FR3031 Add support for Job Shop

```
alter table workctr drop (unit_price_wc);  
alter table workctr  
    add (unit_price_wc decimal(11,3));
```

-- FR3031 Add support for Job Shop

```
alter table workctrwhse drop (unit_price_ww);
alter table workctrwhse
  add (unit_price_ww decimal(11,3));
```

{ MR3119 - Add columns to wosummhst, wocomphst, worouthst previously added to wosumm,

wocomp, worout

```
alter table worouthst
  modify (desc_wr char(80));
alter table worouthst
  add (ext_desc_wr varchar(255) before purge_date_wr,
      unit_price_wr decimal(11,3) before purge_date_wr,
      include_price_wr char(1) before purge_date_wr,
      ack_print_wr char(1) before purge_date_wr,
      quo_print_wr char(1) before purge_date_wr,
      inv_print_wr char(1) before purge_date_wr,
      taxable_wr char(1) before purge_date_wr);
```

-- FR3031 Add support for Job Shop

```
alter table worout
  drop (ext_desc_wr);
alter table worout
  drop (unit_price_wr);
alter table worout
  drop (include_price_wr);
alter table worout
  drop (ack_print_wr);
alter table worout
  drop (quo_print_wr);
alter table worout
  drop (inv_print_wr);
alter table worout
  add (ext_desc_wr varchar(255),
      unit_price_wr decimal(11,3),
      include_price_wr char(1),
      ack_print_wr char(1),
      quo_print_wr char(1),
      inv_print_wr char(1));
```

-- FR3153 Add taxable support for Job Shop Orders

```
alter table worout
  add (taxable_wr char(1));
```

{ MR3119 - Add columns to wosummhst, wocomphst, worouthst previously added to wosumm, wocomp, worout 10/21/2013 RJ-FGSS }

```
alter table wosummhst
  add (responsible_ws char(10) before purge_date_ws,
      total_price_ws decimal(11,3) before purge_date_ws,
      margin_pct_ws decimal(5,2) before purge_date_ws,
      ext_desc_ws varchar(255) before purge_date_ws,
      contact_name char(20) before purge_date_ws,
      contact_phone char(20) before purge_date_ws);
```

-- FR3031 Add support for Job Shop

```
alter table wosumm
  drop (total_price_ws);
alter table wosumm
  drop (margin_pct_ws);
alter table wosumm
  drop (ext_desc_ws);
alter table wosumm
  add (total_price_ws decimal(11,3),
      margin_pct_ws decimal(5,2),
      ext_desc_ws varchar(255));
```

--- FR3099 - Add contact name and number

```
alter table wosumm add
  contact_name char(20),
  contact_phone char(20)
```

--- MR3230.sql - new index on wosumm

```
create index wosummi2 on wosumm (sales_order_ws, sales_line_ws);
```

--- FR3362.sql Add index on wosumm.close_date_ws for better performance

```
create index "informix".wosummi3 on "informix".wosumm (close_date_ws);
```

--- FR2410 - Add special_order field

```
alter table dciinvtr add (
  handling_fee decimal(8,4),
```

special_order char(1)

-- FR3031 Add support for Job Shop

```
alter table dciinvtr drop (ext_desc);
alter table dciinvtr
  add (ext_desc varchar(255));
```

--- FR3217 - RoHS Compliance

```
alter table dciinvtr add (fmd_required char(1), mfg_name char(25), mfg_item char(25));
alter table dciinvtr add (rohs_compliant char(1));
```

--- FR3343 - Add warranty days

```
alter table dciinvtr add (warr_days smallint);
```

--- FR2645 - Expand stock location

alter table dcilocar modify

```
  stock_location char(14),
  loc_row char(4),
  loc_bin char(4)
```

--- FR3034 - Base RL usage calculation on item start date

```
alter table dcilocar add
  date_added date,
  added_by char(8)
```

--- MR3453 - Expand average lead time

```
alter table dcilocar modify
  avg_ld_tm smallint
```

-- FR2413 - Split Payment Terms

```
alter table "informix".dcoinvc add split_terms_code char(6);
```

--- MR2481 - Fix DC issues

```
alter table dcoinvc add
  restock_fee decimal(8,2),
  handling_fee decimal(8,4)
alter table dcoinvc drop split_terms_code;
alter table dcoinvc add
  split_terms_code char(6)
```

--- MR2644 - Add column to dcoinvc

```
alter table dcoinvc  
  add (balance decimal(12));
```

--- MR2793 - Ship Via must be 30 characters

```
alter table dcoinvc modify  
  ship_via char(30)
```

--- MR2694 - Add cost to sales order import

```
alter table dcoordrd add  
  item_cost decimal(12)
```

-- FR3235 - Blanket SOs

```
alter table dcoordrd add (  
  blanket_doc_no integer);
```

--- FR2394 - RMA Status

```
alter table dcoordre add  
  rma_status_code char(20)
```

-- FR2524 - Add Phone number to stoordre

```
alter table "informix".dcoordre  
  add (phone char(12));
```

--- MR2481 - Fix DC issues

```
alter table dcoordre drop  
  restock_fee,  
  default_rma_type,  
  rma_status_code,  
  split_terms_code,  
  phone  
alter table dcoordre add  
  default_rma_type char(3),  
  restock_fee decimal(8,2),  
  ship_to_name char(30),  
  one_time_cust char(1),  
  ship_type char(6),  
  restock_percent decimal(6,2),  
  restock_amount decimal(8,2),  
  rma_status_code char(20),  
  handling_fee decimal(8,4),  
  split_terms_code char(6),
```

```
    phone char(12)
alter table dcoordre modify
    ship_via char(30),
    country char(2)
```

--- FR3098 - Extended description in SO

```
alter table dcoordre modify ( order_description varchar(255));
```

--- FR3099 - Add contact name and number

```
alter table dcoordre add
    contact_name char(20),
    contact_phone char(20)
```

-- FR3101 - Add fixed_price to stoordre, dcoordre

```
alter table dcoordre
    add (fixed_price char(1));
```

--- MR3197 - Expand phone to 20 characters

```
alter table dcoordre modify ( phone char(20));
```

-- FR3235 - Blanket SOs

```
alter table dcoordre add (
    rlse_no integer);
```

-- FR3323 - Customer Credit Management

```
alter table dcoordre add (
    hold_code char(6),
    credit_approved char(1),
    approved_by char(8),
    date_approved date
```

--- FR2280 - Allow duplicate lot number per item/warehouse

```
alter table dcoserle add
    cost decimal(14,4),
    seq_no integer
```

-- FR3031 Add support for Job Shop

```
alter table dcpinvcd
    drop (prod_order);
alter table dcpinvcd
```

```
drop (prod_order_lot);
alter table dcpinvcd
drop (element);
alter table dcpinvcd
add (prod_order char(7),
prod_order_lot char(3),
element char(20));
```

-- MR3154 - Missing columns in DC

```
alter table dcp.pytor add (cc_ap_acct_no integer);
alter table dcp.pytor add (cc_ap_department char(3));
```

--- FR2284 - Display of purchases totals in vendor record

```
alter table dcp.vendr add (
ytd_purchases decimal(14),
lifetime_purchases decimal(14))
```

--- FR2416 - Total Open Purchase Orders

```
alter table dcp.vendr add(open_po_amt decimal(14));
```

-- FR2413 - Split Payment Terms

```
alter table "informix".dcoordre add split_terms_code char(6);
```

-- MR3154 - Missing columns in DC

```
alter table dcp.vendr add (reb_rcv_acct_no integer);
alter table dcp.vendr add (cc_template char(20));
```

--- FR2177 - Display of sales totals in customer record

```
alter table dcr.custr add (
ytd_sales decimal(14),
lifetime_sales decimal(14))
```

--- FR2337 - Multiple telephone numbers in customer record

```
alter table dcr.custr add (
label1 char(10),
phone1 char(20),
label2 char(10),
phone2 char(20),
label3 char(10),
phone3 char(20),
```

```
label4 char(10),
phone4 char(20),
label5 char(10),
phone5 char(20),
label6 char(10),
phone6 char(20),
label7 char(10),
phone7 char(20),
label8 char(10),
phone8 char(20),
label9 char(10),
phone9 char(20),
label10 char(10),
phone10 char(20)
```

-- FR2413 - Split Payment Terms

```
alter table "informix".drcustr add split_terms_code char(6);
```

-- FR2568 - Account Type and Source

```
alter table drcustr add (
  account_type char(25),
  account_source char(25) );
```

--- MR2765 Add warehouse_code to drcustr

```
alter table drcustr add
  warehouse_code char(10)
```

-- FR3323 - Customer Credit Management

```
alter table drcustr add (
  duns_no char(9),
  active char(1),
  acct_established date,
  collection_contact char(30),
  collection_phone char(20),
  credit_check char(1),
  over_credit_pct decimal(4,2),
  credit_approve_dt date,
  credit_approve_by char(8),
  avg_pay_days smallint,
  last_payment decimal(12,2),
  open_orders decimal(12,2),
```



```

    hold_code char(6)
alter table dcrcustr modify (
    label1 char(30),
    label2 char(30),
    label3 char(30),
    label4 char(30),
    label5 char(30),
    label6 char(30),
    label7 char(30),
    label8 char(30),
    label9 char(30),
    label10 char(30)

```

--- MR2765 Add warehouse_code to dcrcustr

```

alter table dcrshipr add
    warehouse_code char(10)

```

--- FR3484 - Real Time Processing

```

alter table stgcntrc add (auto_genjrn_post char(1));
alter table stgcntrc add (auto_genjrn_post_d char(1));

```

-- MR3238 - Store qty on hand on adjustment table

```

alter table stiadjmd add (qty_on_hand decimal(10));
(select stilocar.qty_on_hand from stilocar
 where stilocar.item_code = stiadjmd.item_code and
       stilocar.warehouse_code = stiadjmd.warehouse_code)
where stiadjmd.qty_on_hand is null;

```

--- FR2410 - Add special_order field

```

alter table stiarinv add (
    handling_fee decimal(8,4),
    special_order char(1)

```

-- FR3031 Add support for Job Shop

```

alter table stiarinv drop (ext_desc);
alter table stiarinv
    add (ext_desc varchar(255));

```

--- FR3217 - RoHS Compliance

```

alter table stiarinv add (fmd_required char(1), mfg_name char(25), mfg_item char(25));

```

```
alter table stiarinv add (rohs_compliant char(1));
```

--- FR3343 - Add warranty days

```
alter table stiarinv add (warr_days smallint);
```

--- MR3344 - Sync up with Item Master Screen

```
alter table "informix".stiarinv add uom_list_code char(10);  
alter table "informix".stiarinv add td_disc_allowed char(1);  
alter table "informix".stiarinv add tax char(1);  
alter table "informix".stiarinv drop last_purchased;  
alter table "informix".stiarinv drop last_sold;  
alter table "informix".stiarinv drop last_activity;
```

--- MR2472 - add fields to stiarloc

```
alter table stiarloc add (primary_bin char(15), secondary_bin char(15));
```

--- FR2645 - Expand stock location

```
alter table stiarloc modify  
    stock_location char(14),  
    loc_row char(4),  
    loc_bin char(4)
```

--- FR3034 - Base RL usage calculation on item start date

```
alter table stiarloc add  
    date_added date,  
    added_by char(8)
```

--- FR2335 - Special Handling charges

```
alter table sticntrc  
    add (handling_acct_no integer);
```

--- FR3224 - Add barcode-number prefix

```
alter table sticntrc add (  
    pref_bcno char(2)
```

--- FR3484 - Real Time Processing

```
alter table sticntrc add (auto_post_adj char(1));  
alter table sticntrc add (ticket_print char(1));  
alter table sticntrc add (transfer_print char(1));  
alter table sticntrc add (auto_post_tsf char(1));  
alter table sticntrc add (auto_post_adj_d char(1));
```

```
alter table sticntrc add (auto_post_tsf_d char(1));
```

--- FR2188 - Variable commissions rate

```
create table "informix".sticommd
  item_code char(20),
  begin_price decimal(14),
  end_price decimal(14),
  comm_pct decimal(14)
create index "informix".i1icommd on sticommd(item_code);
alter table sticommd lock mode(row);
```

-- FR2408 - Cross Sells

```
create table "informix".sticrsld
  item_code char(20),
  line_no smallint,
  cross_item_code char(20)
```

--- FR2335 - Special Handling charges

```
alter table stiinvtr
  add (handling_fee decimal(8,4));
```

--- FR2410 - Add special_order field

```
alter table stiinvtr add (special_order char(1));
```

--- FR2426 - Add UOM List feature for multiple units of measure

```
alter table "informix".stiinvtr add uom_list_code char(10);
```

-- FR3031 Add support for Job Shop

```
alter table stiinvtr drop (ext_desc);
alter table stiinvtr
  add (ext_desc varchar(255));
```

--- FR3217 - RoHS Compliance

```
alter table stiinvtr add (fmd_required char(1), mfg_name char(25), mfg_item char(25));
alter table stiinvtr add (rohs_compliant char(1));
```

--- FR3343 - Add warranty days

```
alter table stiinvtr add (warr_days smallint);
```

--- MR3258 Add new action to access Mfg Data windows

alter table stiinvtr modify (buyer_it char(6));

--- FR2645 - Expand stock location

alter table stilocar modify
stock_location char(14),
loc_row char(4),
loc_bin char(4)

--- FR3034 - Base RL usage calculation on item start date

alter table stilocar add
date_added date,
added_by char(8)

--- MR3258 Add new action to access Mfg Data windows

alter table stilocar modify (buyer_iw char(6));

--- MR3453 - Expand average lead time

alter table stilocar modify
avg_ld_tm smallint

--- FR3365 - Add default receiving location to stihwser. Used for separate receiving and putaway processes.

alter table stihwser add
(dflt_rec_loc char(15));

-- FR2413 - Split Payment Terms

-- Archive tables
alter table "informix".aroinvce add split_terms_code char(6);

--- MR2793 - Ship Via must be 30 characters

alter table aroinvce modify
ship_via char(30)

-- For FR2514 - phone

alter table aroordre add phone char(12);

-- FR2413 - Split Payment Terms

-- Archive tables
alter table "informix".aroordre add split_terms_code char(6);

--- MR3197 - Expand phone to 20 characters

```
alter table aroordre modify ( phone char(20));
```

--- **FR2645 - Expand stock location**

```
alter table aroshipd modify  
    stock_location char(14)
```

--- **FR2285 - Require password if sold below required profit**

```
alter table stocntrc add  
    profit_approval char(1),  
    profit_override char(10)
```

--- **FR2343 - Restock Fee**

```
alter table stocntrc add (restock_percent decimal(6,2));
```

--- **FR2332 - Streamline order entry**

```
alter table stocntrc add  
    ord_r_printpt char(1),  
    ord_createinv char(1),  
    pick_createinv char(1),  
    ship_createinv char(1),  
    invce_print char(1),  
    invce_post char(1),  
    bko_printpt char(1)  
    ord_r_printpt = "N",  
    ord_createinv = "N",  
    pick_createinv = "N",  
    ship_createinv = "N",  
    invce_print = "N",  
    invce_post = "N",  
    bko_printpt = "N"  
where 1=1;
```

--- **MR2440 - Updates to Fedex and UPS web services interface**

```
alter table stocntrc add  
    ship_type char(6)  
alter table stocntrc modify  
    ship_via char(30)
```

--- **FR3182 - Additional Real Time Processing**

```
alter table stocntrc add
```

```
pick_createpl char(1),
ship_createpl char(1),
auto_invce_print char(1),
auto_invce_post char(1)
```

--- FR3206 - Approved Vendor List

```
alter table stocntrc add
  create_bko_po char(1)
```

--- FR3484 - Real Time Processing

```
alter table stocntrc add (auto_invce_post_d char(1));
alter table stocntrc add (invce_post_d char(1));
```

--- FR2335 - Special Handling charges

```
alter table stoinvce
  add (handling_fee decimal(8,4));
```

-- FR2413 - Split Payment Terms

```
-- Modify related tables
alter table "informix".stoinvce add split_terms_code char(6);
```

--- MR2793 - Ship Via must be 30 characters

```
alter table stoinvce modify
  ship_via char(30)
```

-- FR3235 - Blanket SOs

```
alter table stoordrd add (
  blanket_doc_no integer);
```

--- FR2350 - Display/Change Ship to addresses

```
alter table stoordre add
  ship_to_name char(30),
  one_time_cust char(1)
alter table stoordre modify
  country char(2)
```

--- FR2343 - Restock Fee

```
alter table stoordre add (restock_percent decimal(6,2));
alter table stoordre add (restock_amount decimal(8,2));
```

--- FR2394 - RMA Status

```
alter table stoordre add  
    rma_status_code char(20)
```

--- FR2362 - Interface to Fedex and UPS web services

```
alter table stoordre add  
    ship_type char(6)  
alter table stoordre modify  
    ship_via char(30)
```

--- FR2335 - Special Handling charges

```
alter table stoordre  
    add (handling_fee decimal(8,4));
```

-- FR2524 - Add Phone number to stoordre

```
alter table "informix".stoordre  
    add (phone char(12));
```

-- FR2413 - Split Payment Terms

```
alter table "informix".stoordre add split_terms_code char(6);
```

--- FR3098 - Extended description in SO

```
alter table stoordre modify ( order_description varchar(255));
```

-- FR3101 - Add fixed_price to stoordre, dcoordre

```
alter table stoordre  
    add (fixed_price char(1));
```

--- MR3197 - Expand phone to 20 characters

```
alter table stoordre modify ( phone char(20));
```

-- FR3235 - Blanket SOs

```
alter table stoordre add (  
    rlse_no integer);
```

-- FR3323 - Customer Credit Management

```
alter table stoordre add (  
    hold_code char(6),
```

```
credit_approved char(1),
approved_by char(8),
date_approved date
```

--- MR2969 - PO Number must be character 20

```
alter table stoordre modify ( po_no char(20));
```

--- MR3195 - Add stootypr.fixed_price

```
alter table stootypr add (fixed_price char(1));
```

--- MR3334 - Change to salesperson logic

```
alter table stootypr add (salesperson_req char(1));
```

-- FR3323 - Customer Credit Management

```
alter table stootypr add (
    credit_check char(1)
```

--- FR3097 - Add warehouse code to special pricing

```
alter table stopricr add
    warehouse_code char(10)
```

--- FR2645 - Expand stock location

```
alter table stoshipd modify
    stock_location char(14)
```

--- MR2796 - Create indexes to improve item_avail() performance }

```
create index "informix".i5oshipd on stoshipd (item_code, warehouse_code, stage, doc_no);
```

--- FR3521 - Store ATP Date in stoshipd

```
alter table stoshipd add
    atp_date date
```

--- FR2362 - Interface to Fedex and UPS web services

```
alter table stoshpar add
    ship_type char(6),
    package_type char(30)
```

--- MR3157 - Expand sotshpar ship contact

```
alter table stoshpar modify
```


ship_contact char(30)

--- FR2362 - Interface to Fedex and UPS web services

```
alter table stoshpfr add
  ship_type char(6),
  package_type char(30),
  stage char(5),
  inv_doc_no integer
```

--- MR2987 - Tax posted to GL does not match tax on invoice using multilvel tax

```
alter table stoshtxd modify (tax_amt decimal(12,2));
```

--- FR2362 - Interface to Fedex and UPS web services

```
alter table stoupsrc add
  ship_type char(6),
  meter_no char(15),
  version_id_svc char(10),
  version_id_major char(10),
  version_id_int char(10),
  version_id_minor char(10),
  authent_key char(30),
  authent_pwd char(30),
  svc_available char(1)
```

--- FR2362 - Interface to Fedex and UPS web services

```
alter table stoupstr modify
  ship_code char(30)
```

--- FR2332 - Streamline order entry

```
alter table stucntrc add
  bko_printpt char(1)
  bko_printpt = "N"
where 1=1;
```

-- FR2415 - Create rebate import mapping tables

```
alter table stucntrc add
  reb_cog_acct_no integer,
  reb_rcv_acct_no integer
```

-- MR2990 - Integrate Fitrix Standard Costing 02/25/13 RJ-FGSS

```
alter table "informix".stucntrc
  add (var_acct_no integer);
```

--- FR3182 - Additional Real Time Processing

```
alter table stucntrc add
    ord_r_printpo char(1)
```

--- FR3206 - Approved Vendor List

```
alter table stucntrc add
    avl_password char(8),
    avl_required char(1)
```

--- FR3342 - Add gwc_server and help text for the gwc_server field;

```
alter table stucntrc add (auto_post_rec char(1));
```

--- FR3484 - Real Time Processing

```
alter table stucntrc add (auto_post_ap char(1));
alter table stucntrc add (auto_post_ap_dsp char(1));
alter table stucntrc add (auto_post_rec_dsp char(1));
```

--- FR2426 - Add UOM List feature for multiple units of measure

```
alter table stuctlgd add
    purch_unit char(2),
    purch_factor decimal(6)
```

--- FR3217 - RoHS Compliance

```
alter table stuctlgd add (fmd_received char(1));
```

-- FR2415 - Create rebate import mapping tables

```
create table "informix".stumfrbd
    vend_code char(20),
    contract char(20),
    item_code char(20),
    contract_cost decimal(11,4)
alter table stumfrbd lock mode (row);
create unique index "informix".i1stumfrbd on "informix".stumfrbd (vend_code, contract, item_code)
using btree ;
```

-- FR3223 - Blanket POs

```
alter table stuordrd add (
    ship_date date,
```

```
blanket_doc_no integer);
```

-- MR3265 Link component to supplying PO's when short

```
alter table stuordre  
  add (prod_order_lot char(7));
```

-- FR3223 - Blanket POs

```
alter table stuordre add (  
  rlse_no integer,  
  ship_date date);
```

--- FR2417 - Reverse Receipts

```
alter table sturecte add(reverse char(1), rev_doc_no integer);
```

--- FR3206 - Approved Vendor List

```
alter table sturqstd add  
  cost decimal(14,4)
```

-- MR2974 - Install/Integrate Fitrix Labor Processing 02/22/2013 RJ-FGSS

```
alter table "informix".styemplr  
  add (shift_em char(3),  
    emp_team_em char(5),  
    emp_group_em char(5),  
    department_em char(3),  
    jobclass_em char(3),  
    machine_ovr_em char(1),  
    cost_ctr_ovr_em char(1),  
    department_ovr_em char(1),  
    team_ovr_em char(1),  
    shift_ovr_em char(1),  
    job_class_ovr_em char(1),  
    std_hour_ovr_em char(1),  
    date_ovr_em char(1));
```

-- MR3130 Add'l values added to routoper

```
alter table routoper  
  add (priority_ro char(1));
```

-- FR3031 Add support for Job Shop

```
alter table initsc
```

```
add (dflt_acctcd_is char(13));
```

--- **MR3012 modify column reason.acctcd_rn**

```
alter table reason
```

```
  modify (acctcd_rn char(13));
```

-- **FR3323 - Customer Credit Management**

```
alter table responsb add (
```

```
  email_address char(50)
```

SUPPLEMENTAL PATCHES AND ENHANCEMENTS

Accessing Fourth Generation's FTP site

All patches and enhancements that apply to the Fourth Generation products are posted to our FTP site on the internet as soon as they are made available.

These supplements are supplied as complete program directories with the patches or enhancements fully applied to the program and tested. An 'INFO' file in each program directory will explain the changes applied.

To access our FTP site on the internet:

1. Go to <ftp.fourthgeneration.com>
2. Login: fg_cust
3. Password: fourth9
4. Click on the directory named patches.
5. Click on the README_patches file for general information.
6. Click on the directory of the version of the product you are running to see what patches/enhancements are available.

We highly recommend that you visit our FTP site once every month.

Installing Supplemental Patches and Enhancements

Naming Conventions

Note: In the instructions below the version number is 6.00. If you are running version 5.40 or earlier change the instructions to use your version number

The following naming conventions are used for program and menu directories delivered on the 6.00 media:

.4gs - latest released version. If a directory named .540.x exists , the files in the .4gs directory will be identical.

.600 - version 6.00 as originally released (menu directories have no extension). This directory will only exist if patches exist for the program.

.600.a - 1st patched version of a program or menu directory

.600.b - 2nd patched version of a program or menu directory (includes any patches in .600.a version).

Program Installation procedures

□ Follow the standard installation procedures to extract **but not compile** the accounting modules (this step assumes you are installing the entire new version, if you are only installing supplements downloaded from the ftp site, proceed with the next step:)

□ find all program directories with a name matching: '*.600.*'

□ for each program directory found above:

□ if there is no '*.600' version of the program, rename the .4gs program directory to *.600 (if there is already a .600 version of the program, this step should be omitted)

(this is to preserve the original copy just in case it is needed later, and will serve as notice that the supplement has been applied)

□ create a new .4gs program directory by making a copy of the supplement directory. If more than one supplement directory exists, use the one that has the highest letter.

(i.e. if there is a *i_cashe.600.c*, copy the entire directory to *i_cashe.4gs*)

(the supplements are cumulative, so the highest letter has all the supplements for that program, only the highest supplement letters are distributed on the media)

□ repeat the standard installation procedures, but this time
*compile **but do not extract*** the accounting modules.