

A decorative graphic on the left side of the page, featuring a dense cluster of various blue and grey icons representing retail items like clothing, shoes, and accessories, which then disperses into a sparse trail of similar icons extending towards the right.

*Code Distribution Services (CDS)
Administrator's Guide*

Code Distribution Services (CDS) Administrator's Guide

Date of Issue	Product Identification Number	Part Number	Brief Description
December 2008	45001/114	89000485	Initial Release
January 2014	45001/114	89000797	Version 1.3

©Copyright Retalix, Inc. 1995-2014
All rights reserved

This publication is protected by federal copyright law. This document and its contents are the proprietary, confidential information and property of Retalix, Inc. ("Retalix"). Unauthorized disclosure, reproduction, distribution or use of this document and/or its contents in any form is strictly prohibited. No part of this publication may be reproduced or transmitted into any human or computer language in any form or by any means, stored in a retrieval system, transmitted, redistributed, translated or disclosed to third parties, or de-compiled in any way including, but not limited to, photocopy, photograph, electronic, mechanical, magnetic or manual without the express written permission of Retalix or its licensors, if any. This document, notwithstanding the above, may be distributed in electronic or printed form to personnel who are employed by either (1) a Retalix customer of the subject system of this document or (2) personnel from an authorized Retalix Channel Partner of the subject system of this document. All copies, so distributed and/or so authorized, shall contain a full copy of this copyright notice.

Retalix has prepared this manual for use by users, authorized third parties and personnel of Retalix as a guide to the proper installation, operation, customization and/or maintenance of Retalix equipment and software. The drawings and specifications contained herein are the property of Retalix and/or its licensors.

Retalix endeavors to ensure that the information in this document is correct and fairly stated but does not accept liability for any error or omission. Retalix makes no representation or warranties with respect to the contents hereof, and specifically disclaims any implied warranties of merchantability or fitness for a particular purpose or non-infringement. No commitments by Retalix or its suppliers are made by or from this documentation which is provided for information only.

Development of Retalix products and documentation is continuous: Retalix reserves the right to revise this publication and to make changes from time to time in the contents hereof or in the products herein described or discussed without notice and without any obligation of Retalix to notify any person or organization of such revision or changes. Information published in this document will likely become obsolete over time and it is recommended that users regularly check for updates and newer versions.

The trademarks Retalix, StoreNext, ISS45, ScanMaster, R10, PocketOffice, RBO, Connected Payments, Connected Services, Connected Loyalty and all other Retalix product names are registered trademarks or trademarks of Retalix; all other trademarks or registered trademarks are the property of their respective owners. Third-party products, services, or company names referenced in this document may be trademarked or copyrighted by their respective owners, and are for identification purposes only. Copyrights, trademarks and license agreements shall be governed and construed in accordance with the laws of the State of Texas and the Federal Arbitration Act, and shall benefit Retalix, its successors, and assigns.

Address comments and corrections to:

Retalix, Inc.
Software Program Director
797 Commonwealth Drive
Warrendale, PA 15086

About This Guide

The StoreNext Code Distribution Services (*CDS*) Administrator's Guide is divided into the following chapters:

Chapter 1, Introduction, provides an overview of the *CDS* software application, including a description of its basic functionality, and how to use this document. In addition, this chapter summarizes the steps in the Upgrading process performed by *CDS*.

Chapter 2, CDS Setup, describes the system requirements for *CDS*, and how to set the required parameters related to the upgrade.

Chapter 3, Package Management Utility, describes the main functions of *RegistrPack*, the Package Management utility for *CDS*.

Chapter 4, Package Administration, describes the main functions of *PkgAdmin* utility and the upgrade processes that are performed on the MFS1 server, the MFS2 workstation, the LFS and the POS terminals.

Chapter 5, Running CDS on Demand, describes the processes when *CDS* is run on demand from remote access.

Chapter 6, CDS Support for PFS, describes how *CDS* supports the installation of PFS components, to enable upgrades of Fuel systems.

Chapter 7, Package Structure, describes the structure of *CDS* upgrade packages.

Chapter 8, Alerts, describes the alert messages related to *CDS*.

Chapter 9, Maintenance Structure, describes the structure of a *CDS* small Maintenance.

Table of Contents

About This Guide	i
Table of Contents	ii
Introduction	1
Introduction	1-2
How to Use This Document	1-3
Glossary	1-4
How Does CDS Work?	1-5
For ISS45 v8:.....	1-5
For ISS45 v7:.....	1-6
How are Packages Sent?	1-7
Installing the Connected Services & Commpoint Interface.....	1-8
CommPoint.....	1-8
Connected Services.....	1-8
Package Selection	1-9
Store Selection	1-11
Advanced Selection	1-13
CDS Setup.....	1
System Requirements	2-5
Getting Started	2-5
Defining CDS Parameters.....	2-5
Configuring MFSs in System using <i>PC In System</i>	2-6
Configuring Active POS terminals.....	2-8
Defining Checkout Banks Parameters	2-10
Defining CDS and R-Session Parameters	2-13
Defining Parameters for High-Sensitivity Packages	2-24
Defining Blackout Periods.....	2-25
Defining the Activation Date.....	2-27
Scheduling Activation Date for EOD	2-31
Quick Reference to Scheduling Activation Dates During EOD	2-31
Selecting Trading Date:	2-32
Selecting System Date:	2-33

Selecting Next Working Date:	2-34
Scheduling Activation Outside of EOD.....	2-35
Quick Reference to Scheduling Activation Dates Outside of EOD	2-36
Selecting Trading Date:	2-36
Selecting System Date:	2-37
Selecting Next Working Date:.....	2-38
Setting Up Disk Space Alerts.....	2-40
Setting the System to Auto Log-On	2-41
Package Management Utility	1
Overview.....	3-2
Registration	3-3
Validation of .CTL file.....	3-3
<i>RegistrPack</i> Functionality at EOD.....	3-5
CheckComplete	3-5
Housekeeping.....	3-6
Registering	3-7
<i>RegistrPack</i> Functionality from the Application	3-7
Package Administration.....	1
Overview.....	4-2
Low-Sensitivity & High-Sensitivity Packages.....	4-2
Version Upgrading Process in Low Sensitivity Packages.....	4-4
Running the Package During EOD	4-5
Process Performed by PkgAdmin:	4-5
Running the Package from the Application	4-5
Process Performed by PkgAdmin:	4-5
Opener Utility.....	4-6
Process performed by the Opener:.....	4-6
Viewing Opener Log Files for MFS Upgrades.....	4-7
Viewing Opener Logs for POS Terminal Upgrades.....	4-8
Upgrading MFS1.....	4-9
Upgrading MFS2 & LFSS	4-10
Upgrade Failure in v8 MFSS	4-11
Upgrade Failure in v7 MFSS	4-13
Upgrading the POS Terminals	4-13
Viewing Upgrade Statuses in POS Terminals.....	4-16
Aborting an Upgrade on POS Terminals.....	4-16

Activating and Monitoring Upgrades via Package Maintenance	4-17
Activating an Upgrade from the Application	4-19
Monitoring v8 Upgrades	4-23
Using the ' <i>Checked Complete</i> ' Function from the Application	4-23
General Tab.....	4-27
Files List Tab	4-29
PCs Status Tab	4-31
<i>Refresh</i> and <i>Auto-Refresh</i> Functions	4-35
Force Upgrade.....	4-35
<i>Declare PC End Package</i> Button	4-37
Guidelines for Changing System Status	4-38
Monitoring v7 Upgrades	4-39
Running CDS on Demand.....	1
Overview.....	5-2
Defining CDS Parameters.....	5-2
<i>RegistrPack</i> and <i>PkgAdmin</i>	5-2
Running CDS "On Demand" at the Store	5-2
CDS Support for PFS	1
Overview.....	6-2
Defining CDS Parameters.....	6-2
Using <i>RegistrPack</i>	6-3
Package Administration	6-4
Version Upgrade Process	6-5
Upgrading Fuel POS Terminals	6-5
Upgrade Failure	6-7
Package Structure.....	1
Package Structure	7-2
Control File	7-2
Front Office File	7-4
WinPos File	7-4
External Hook Structure.....	7-4
Hook Rules and Regulations for POS Hooks.....	7-5
Order of Execution for POS Hooks	7-7
Hook Rules and Regulations Front Office Hooks.....	7-7
Order of Execution for Front Office Hooks.....	7-8
Distribution of Hook Files	7-9

Alerts	1
Overview.....	8-2
Setup	8-2
Alerts for Head Office	8-5
During Registration of a Package	8-5
During Running of a Package	8-5
Structure of a General Alert for CDS	8-6
Viewing CDS Alerts in QDX files.....	8-8
Monitoring Errors & Messages in the Upgrade Process.....	8-9
Viewing CDS Alerts in the POSTT Application.....	8-25
Viewing MFSS Statuses in the CDS Process	8-27
Connected Services CDS Alert Tracking/Monitoring	8-28
Maintenance Structure.....	9-1
Small Maintenance Structure	9-2

1

Introduction

This chapter provides an overview of the StoreNext Code Distribution Services (CDS) software application, including a description of its basic functionality, and how to use this document. In addition, this chapter summarizes the steps in the Upgrading process performed by CDS.

In this chapter:

- What is CDS?
- How to Use This Document?
- How Does CDS Work?

Introduction

StoreNext's Code Distribution Services (CDS) is a software application that automatically distributes and performs upgrades to the Front Office and to the POS terminals. *CDS* works behind the scenes with StoreNext ISS45 software to provide automatic version upgrades.

In the past, software version upgrades were performed manually by the system administrator — a labor-intensive and time-consuming process, since every computer had to be upgraded individually.

Today, *CDS* distributes upgrade-packages with minimal intervention on the part of the system administrator. In addition, *CDS* is compatible with remote support software, e.g., *pcAnywhere*, enabling the system administrator to monitor upgrades from anywhere in the world.

This document describes the different processes available for use when performing upgrades to systems, using *CDS*.

In each process used to upgrade a package, there is the option to run *CDS* to *Register* and *Administer* the package. The following options are available when using *CDS*:

- **Run *CDS* during EOD to automatically upgrade packages. (Refer to *Chapter 3, Package Management Utility*, and *Chapter 4, Package Administration*.)**
- **Run *CDS* via the Application to upgrade packages. (Refer to *Chapter 3, Package Management Utility*, and *Chapter 4, Package Administration*.)**
- **Run *CDS* on demand to send upgrade packages from a location not on site, as well as on site. (Refer to *Chapter 5, Running CDS on Demand*.)**

Refer to the sections in this manual that pertain to your needs. Note that the two utilities: *Regstrpack* (Chapter 3) used to *Register* packages, and *PkgAdmin* (Chapter 4) used to *Administer* packages, are the basis for *CDS* functionality in all the processes, and, therefore, there are references to these Chapters throughout the document.

How to Use This Document

The following conventions are used to convey instructions and information:

- Key words, commands, and Configuration parameters are *italicized*.
- Field names are in **Boldface**.
- Paths are in **Boldface**.
- Notes and Warnings use the following conventions:

Note:

1. Notes contain important suggestions or references for the Reader.
2. The implications of many procedures to Upgrading Fuel POS terminals are explained in Notes at the end of each section.

Glossary

The following technical terms are used in this document:

Term	Definition
Control file (Vnnnnnnnn.ctl)	Include parameters for the current upgrade (store number, date to execute, package sensitivity, etc.). Critical, if parameters are invalid upgrade package will not register.
High Sensitivity	Used to describe a package which contains critical system components (like rmsrv.exe) or other crucial components. This type of package is rare and systems need to be fully in sync before running this type of upgrade.
Housekeeping	A process that determines the packages "age" on the system and perform cleaning of old packages (files and records) according to parameters.
Initiator	The MFS PC that initiates the end of day (EOD) process.
InstallShield	Microsoft utility allows us to build an executable to perform update/upgrade on different Windows systems.
LFS	In ISS45, LFS refers to non-MFS workstation PCs. Within these procedures LFS references any PC on system other than a POS or MFS1.
Low Sensitivity	Most packages are of this type. They have different kind of components inside, PosW32.exe, PPM import screens, etc.
Package	Zip file (Vnnnnnnnn.PKG), which includes the executable upgrade for the Front Office (Fnnnnnnnn.exe), WinPos (Wnnnnnnnn.exe) and a Control file (Vnnnnnnnn.CTL) with upgrade detail (where nnnnnnnn is equal to the version number). For example, 8121050 for ISS45 8.1.2.1-050. See Chapter 7 <i>Package</i>

	<i>Structure</i> for more information.
Package Key	Encoded plug number in the control file which allows package to run ONLY on selected store.
Plug	A software key containing all functional flags for ISS45 and CDS to run. Keys are critical and unique to each store.
Registration	A process in which new packages are detected, validated and if valid transferred to the Wait folder (adding a record to the database table).
Rsession (Remote session)	Utility used by MFS/CDS to schedule and manage the upgrade process and communication channel for each lane. Defining Rsession parameters enables CDS software upgrades and prevents overloads in MFS1.

How Does CDS Work?

The Upgrading process performed by CDS includes the following steps:

For ISS45 v8:

- 1 Administrator schedules the date and times each store or group of stores is to receive an upgrade package. The package files contain the version information and all data pertaining to upgrades of the Front Office and POS terminals.
- 2 The upgrade-package is placed by the administrator on MFS1, the designated server for CDS. This can be performed either on store premises or remotely. From this point on, the upgrade process is automatic. The package is placed in **C:\CDPackagesRoot\New**.
- 3 *RegistrPack*, the Package Management utility that validates and registers the new packages, is invoked by the Support Person, the EOD process or by entering a command from remote access.

- 4 *RegistrPack* transfers the unzipped files from **New** to **Wait** folder in **C:\CDPackagesRoot**, and adds a record to the *SQL* tables with the package data. If package is invalid it enters the **Reject** directory.
- 5 After *RegistrPack* completes the registration, the *PkgAdmin* utility is invoked. *PkgAdmin* checks the *SQL* tables to confirm whether an upgrade is to be performed. *PkgAdmin* is either run when the *EOD* process begins, manually via the application or on demand using a command line.
- 6 **EOD process:**

When *PkgAdmin* detects a valid upgrade package it notifies the *EOD* via a flag called **pkssyneod.flg**, to run synchronically; in other words, *PkgAdmin* ensures that the upgrade will be performed only after the *EOD* process has been completed.

Or

Upgrade via the Application:

From the Package Maintenance, selecting the option **Run Package** allows the Support Person to run a package. If the package is valid and ready to run the upgrade, the package is executed immediately.

Or

Upgrade on Demand

Manually executing upgrade from the command prompt.

- 7 The *Opener* utility is responsible for starting the upgrade and is invoked by *PkgAdmin*. Version upgrades are performed in the following order: MFS1 server upgrades; after MFS1 transmits an **End** status in *SQL* tables, the process resumes in the following components (prioritized by package sensitivity settings): MFS2 workstation, any additional LFS PCs (workstations) and POS terminals. (For further details see *Chapter 4, Package Administration*).
- 8 Upon completion of upgrade, the *SQL* tables are updated with the upgrade status of each machine. Real time alerts are issued (indicating successful upgrade or system malfunction) and package status can be viewed through Package Maintenance.

For ISS45 v7:

- 1 Administrator schedules the date and times each store or group of stores is to receive an upgrade package. The package files contain the

version information and all data pertaining to upgrades of the Front Office and POS terminals.

- 2 These packages (in InstallShield format) are placed by the Administrator in **C:\Pos_pkgslwait** folder. From this point on the upgrade process is automatic.
- 3 After the critical phase of EOD is completed on MFS1 and the *plutoday.zip* file is copied, the upgrade process starts on MFS1, the initiating PC. *Pcmpkgnw.exe* is invoked which validates package(s), reads control file(s), verifies software key settings, registers new packages, performs “housekeeping” on old files based on Back Office parameters and initiates the upgrade process.
- 4 Upgrades on MFS1, MFS2 and LFSs can be prioritized based on Sensitivity settings in BackOffice control parameters.
- 5 After upgrade on MFS1 is completed successfully, maintenance is sent to POS terminals and executed based on Back Office Rsession control parameters.
- 6 Upgrade status for all PCs and POS terminals are monitored and logged to the *packages.qdx* file. Real time alerts are issued (indicating successful upgrade or system malfunction) and package status can be viewed through Package Maintenance.

Note:

Even though many upgrade packages may be validated and registered, only one upgrade package is run each EOD.

Note:

ISS45 v7 does not support the “Upgrade via the Application” and “Upgrade on Demand” options that exist in ISS45 v8.

How are Packages Sent?

Packages are sent to the store using two components:

- CommPoint
- Connected Services

Installing the Connected Services & Commpoint Interface

The StoreNext Connected Services Interface Installation for ISS45 bundles both the client Connected Services Interface (both V7 and V8) and CommPoint communication software. Please refer to the *StoreNext CS Install Guide for ISS45* for more information.

CommPoint

CommPoint is the communications utility used to transmit packages to the ISS45 system. Packages containing upgrade files are sent to each store server designated as the *CommPoint* connection server. During registration at EOD, the package is transferred to MFS1 (the designated *Code Distribution Server*). These files contain version information and all data pertaining to upgrades of the BackOffice and POS terminals. This transmission can be initiated by the Administrator in-store, remotely or automatically at End of Day.

Connected Services

The workflow for upgrading the store through StoreNext Connected Services is as follows:

1. Administrator logs on to *StoreNext Connected Services* application, selects Code Distribution Maintenance, selects the version or patch to be distributed and the relevant stores to be included in that distribution.
2. Based on the input and configuration selected by the administrator, the system prepares an upgrade package to be transmitted to the stores.
3. During the next communication process from the store (usually the EOD process run on the POS system), that package will be "downloaded" to the store.
4. Once the package is at store level it will be registered and ready to be triggered at the appropriate time (as defined by the administrator during the distribution phase).
5. At the defined date the in-store system (POS system) will trigger the upgrade process that will automatically upgrade the various ISS45

systems in the store with the particular version/patch distributed by the administrator.

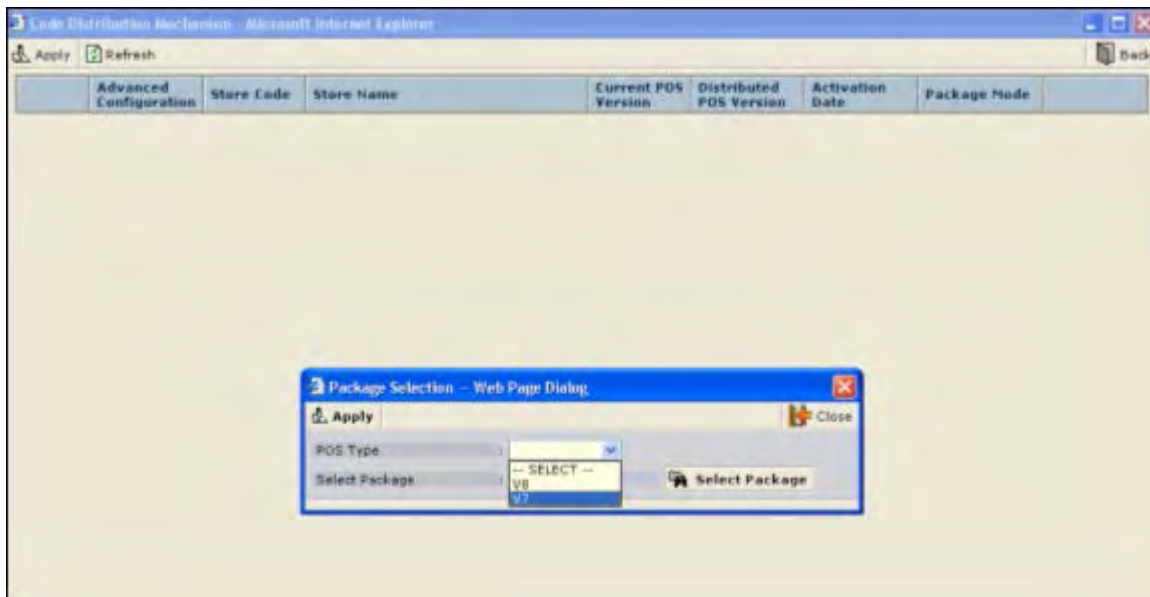
6. Throughout the upgrade process the system will log its activity and that log will be transmitted back to the *Connected Services* servers and provide the administrator with monitoring capabilities.

Package Selection

Use the following steps to select a package for distribution using Connected Services.

➤ **To select a package for distribution:**

- 1 Logon to StoreNext Connected Services website and select the *Code Distribution Maintenance* option from the menu. *The screen that loads has a Package Selection “popup”.*

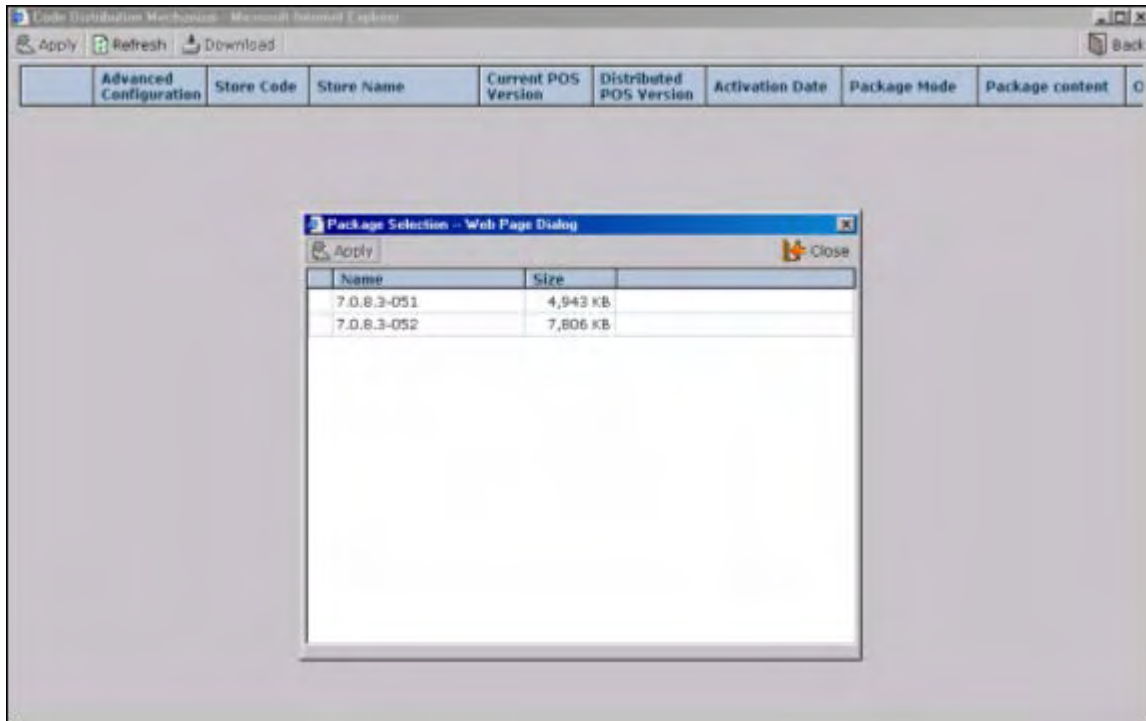


- 2 Click the POS Type field selection arrow and choose whether the package is for an ISS45 V7 or an ISS45 V8 site.

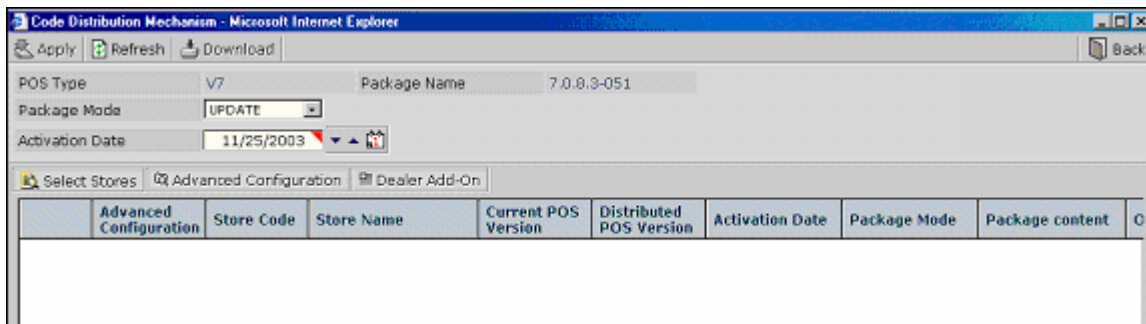
Note:

In the event that only one type of POS system is installed, that selection will automatically be made for the user.

- 3 Click the **Select Package** icon. *A window displays the various software packages that are available for distribution.*



- 4 Highlight the desired package to be distributed (by clicking on the appropriate row) and click **Apply**. *The Code Distribution Mechanism screen displays with the following information:*
- **POS Type:** V7 or V8 (as previously selected).
 - **Package File Name:** The version/patch number to be distributed.
 - **Package Mode:** Update or Delete. (The default is Update).
 - **Activation Date:** The date in which this package will be activated at store level. The default date will be tomorrow's date.



- 5 Select a Package Mode. This option allows the Administrator the ability to UPDATE or DELETE the package from the store. The Package Mode will be UPDATE by default.
- 6 Select the Activation Date. The Activation Date is the date the upgrade of software will occur at the store regardless of the package send date. The default is the next day. The packages selected for distribution will actually be sent to the store in the next communication interval with the store, usually at EOD.

Note:

The packages selected for distribution will be sent to the store in the next communication interval with the store (usually the next time the store performs EOD).

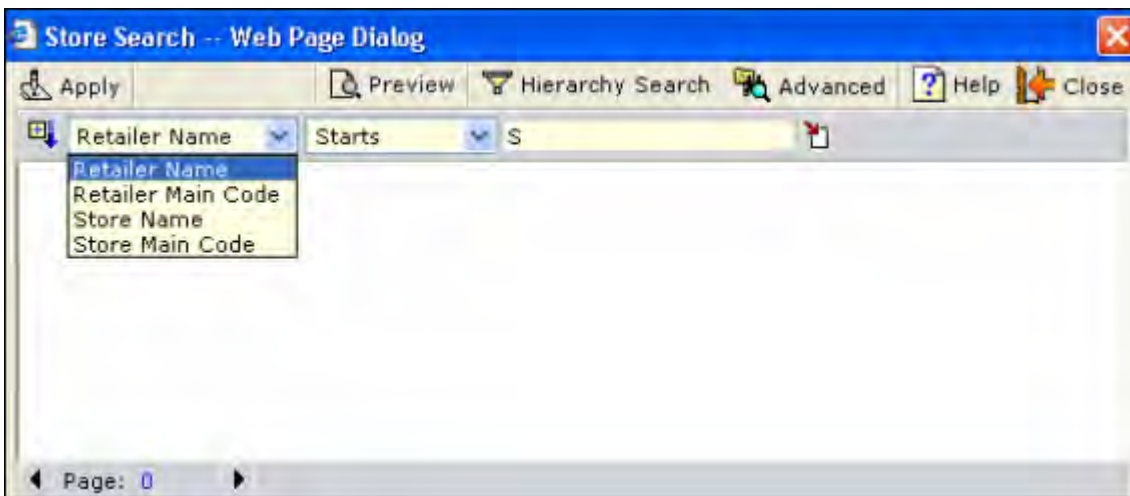
The Activation Date is the date on which the upgrade of software will occur regardless of the package send date.

Store Selection

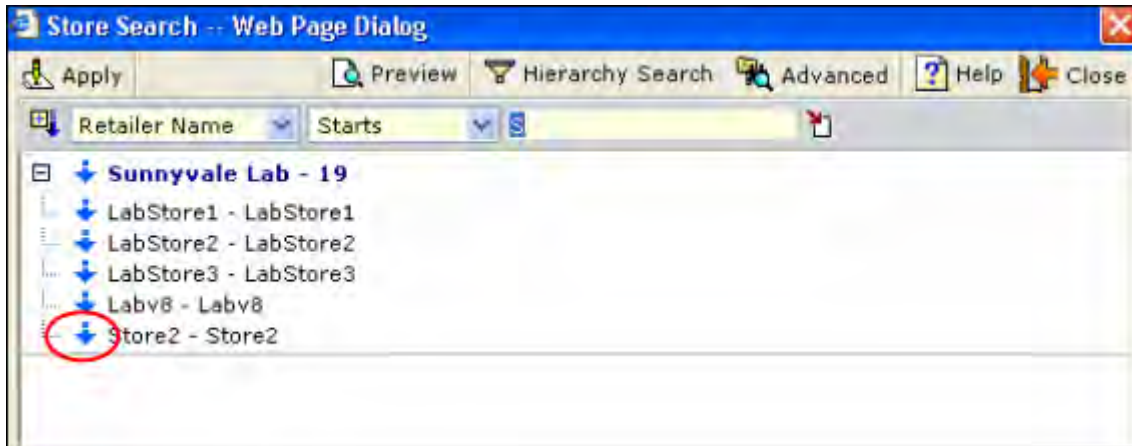
After selecting the package for distribution (above), use the following steps to select the stores where to distribute the package:

➤ **To select the stores for distribution:**

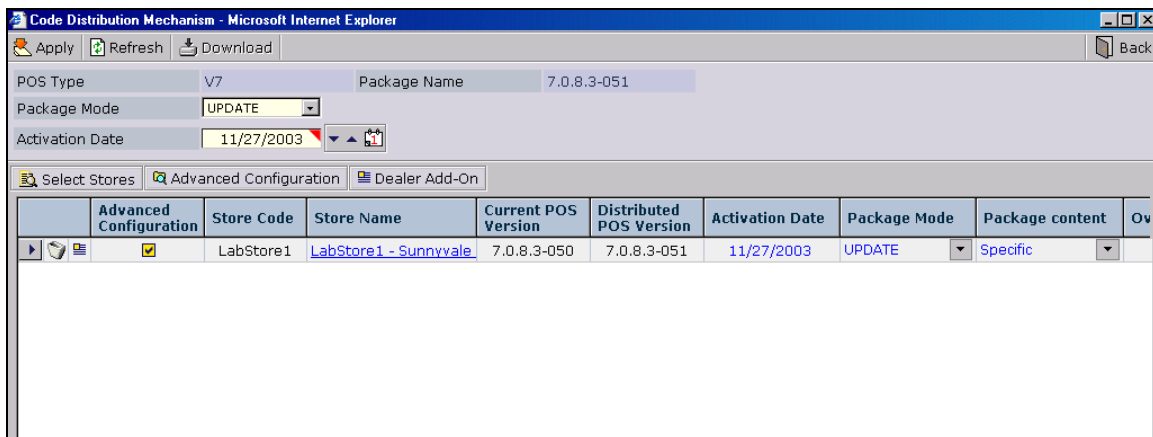
- 1 Click the **Select Stores** button to invoke a store search window. *The Store Search dialog window displays:*



- 2 Search for retailers/stores by name or code, which starts or matches a defined string or click the  button to get a complete listing of the stores within the installed base.



- 3 Use the Blue arrows to select the desired retailer/store. Each selection made will appear in the bottom part of the window. When all of the selections are made, click the *Apply* button to proceed. *The selected stores display.*



Note:

In the event the store(s) selected have a higher version than the package selected for distribution the message "At least one of the stores version is incompatible with the package version" will display. Click **OK** and then delete the store(s) with a later version highlighted in yellow by clicking on the delete  icon.

Note:

Each line contains the following information/functionality:



– Click on this icon to delete this specific line.



– Click on this icon to make a *Dealer Add-on* selection.



– These icons indicate whether an *Advance Configuration* has been set for this store. Click on this icon to make the *Advance Configuration* selection.

Current POS Version – The current POS version at this store.

Distributed POS Version – The version of software that was most recently distributed to this store.

Activation Date – The activation date as defined in previous steps. This enables the dealer to update the activation date for this specific store.

Package Mode – The package mode (Update or Delete) as defined in previous steps. This enables the dealer to update the package mode for this specific store.

Package Content – The package content (Specific or Inclusive) will be set automatically by the system depending on the stores current POS version and the version selected for distribution. The dealer has the option to change the default setting by clicking the drop down arrow.

Specific Package – This package will be selected when the store's current version is only one version behind the version being selected for deployment.

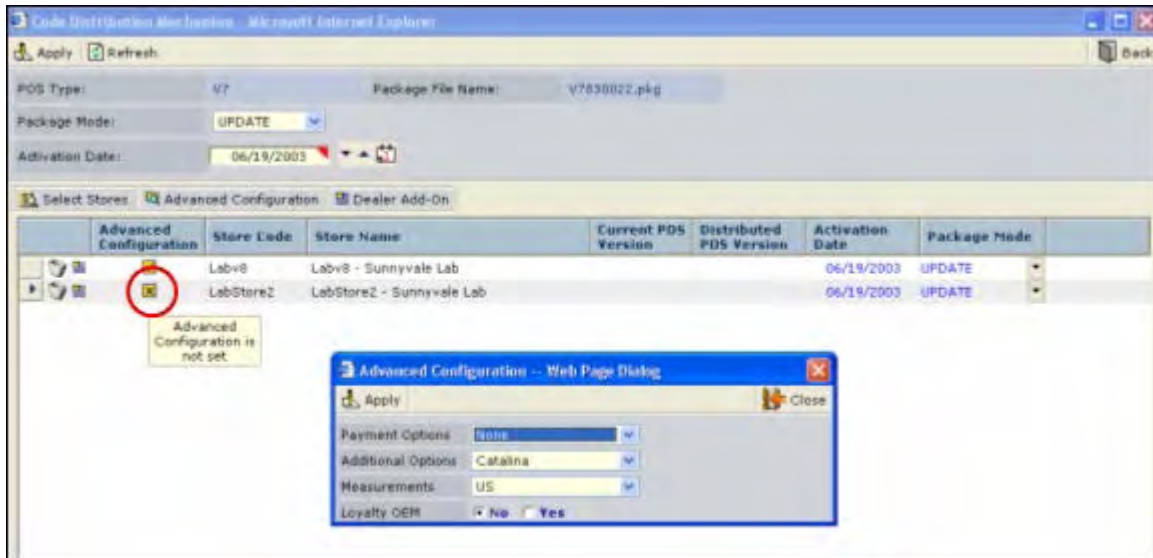
Inclusive Package – This package will be selected when the system identifies that the store version is more than one version behind the one selected for deployment.

Advanced Selection

After selecting the stores for distribution (above), use the Advanced Selection screen to make the required configurations as they pertain to the EFT package that that specific store is using:

- 1 Click on the *Advanced Configuration* icon or highlight the specific store and select the *Advanced Configuration* button, to load the configuration screen.

INTRODUCTION

**Note:**

- Indicates that an Advanced Configuration has been defined for this store.



- Indicates that an Advanced Configuration **has not** been defined for this store.

- 2 Click the **Payment Options** selection arrow and choose the EFT provider for the store.
- 3 Click the **Additional Options** selection arrow and choose a third-party option being used by the store (if any).
- 4 Click the **Measurements** selection arrow and choose either US or Metric.
- 5 Repeat steps 1 through 4 for each store.
- 6 When finished, click *Apply*.

Note:

If one of the stores is missing this setting, the message "Missing Advanced configuration data for all stores marked with 'X' under the Advanced Configuration column" will display.

Note:

Each Advanced Configuration combination points to a specific set of files that reside on the Connected Services servers. The message "Missing Advanced configuration file for one of the stores" displays when a corresponding file or files that fit the configuration settings are missing. Please contact Technical Support if this message appears.

Note:

The **Dealer Add-On** feature allows retailer specific packages (created by the dealer) to be distributed in conjunction with the selected ISS45 package. Click the box to the left of the package name to select the dealer package. Click the Apply button to save changes. For more information on building packages, see Chapter 7 *Package Structure*.

2

CDS Setup

This chapter describes the system requirements for CDS, and how to set the required parameters related to the upgrade, including Setting the Activation Date.

In this chapter:

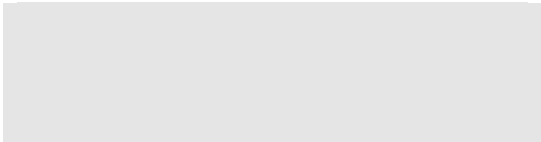
- System Requirements

About This Guide	
Table of Contents.....	
Introduction	
How to Use This Document.....	
Glossary.....	
How Does CDS Work?.....	
How are Packages Sent?.....	
Installing the Connected Services & Commpoint Interface	
CommPoint	
Connected Services	
System Requirements.....	
Getting Started.....	
Defining CDS Parameters	
Configuring MFSS in System using <i>PC In System</i>	
Configuring Active POS terminals	
Defining Checkout Banks Parameters	
Defining CDS and R-Session Parameters	
Defining the Activation Date	
Scheduling Activation Date for EOD	
Scheduling Activation Outside of EOD	
Setting Up Disk Space Alerts	
Setting the System to Auto Log- On.....	
Overview	
Registration.....	

<i>RegistrPack</i> Functionality at EOD
CheckComplete
Housekeeping
Registering
<i>RegistrPack</i> Functionality from the Application
Overview
Low-Sensitivity & High-Sensitivity Packages
Version Upgrading Process in Low Sensitivity Packages
Running the Package During EOD
Running the Package from the Application
Opener Utility
Upgrading MFS1
Upgrading MFS2 & LFSs
Upgrading the POS Terminals
Activating and Monitoring Upgrades via Package Maintenance
Activating an Upgrade from the Application
Monitoring v8 Upgrades
General Tab
Files List Tab
PCs Status Tab
Monitoring v7 Upgrades
Overview
Defining CDS Parameters
<i>RegistrPack</i> and <i>PkgAdmin</i>
Running CDS "On Demand" at the Store
Overview
Defining CDS Parameters
Using <i>RegistrPack</i>



- Package Administration
- Upgrading Fuel POS Terminals
- Package Structure
- Control File.....
- Front Office File
- WinPos File.....
- External Hook Structure
- Overview
- Setup
- Alerts for Head Office.....
- Structure of a General Alert for CDS.....
- Viewing CDS Alerts in QDX files
- Monitoring Errors & Messages in the Upgrade Process
- Viewing CDS Alerts in the POSTT Application
- Viewing MFSs Statuses in the CDS Process.....
- Connected Services CDS Alert Tracking/Monitoring.....
- Small Maintenance Structure



System Requirements

The following are the minimum system requirements that must be met before installing *CDS*:

- **Windows NT Service Pack 5, or Windows 2000 Service Pack 1**
- **MS Internet Explorer 5.5 or higher, with Full Installation (Select *Custom*.)**

Getting Started

The *CDS* application is installed automatically with the initial installation of both ISS45 v8 (v8.1.0.1 and higher) and ISS45 v7 (v7.1.0.1 and higher). If you have an older version installed, you must first upgrade to a supported version to use *CDS*.

On the ISS45 V8 MFS, a directory **C:\CDPackagesRoot** is created containing a **Done**, **New**, **Reject**, and **Wait** folders. On the ISS45 V7 MFS, a directory **C:\POS_PKGS** is created containing a **Done**, **Reject**, and **Wait** folder. On the POS terminals (for both versions) a directory **C:\CDPackagesRoot** is created containing a **Done**, **New**, **Reject**, and **Wait** folder.

When upgrade packages arrive in the store, they should be placed in **New** (for v8) or **Wait** (for v7).

Defining *CDS* Parameters

The following specific parameters must be defined in order to automatically perform software upgrades:

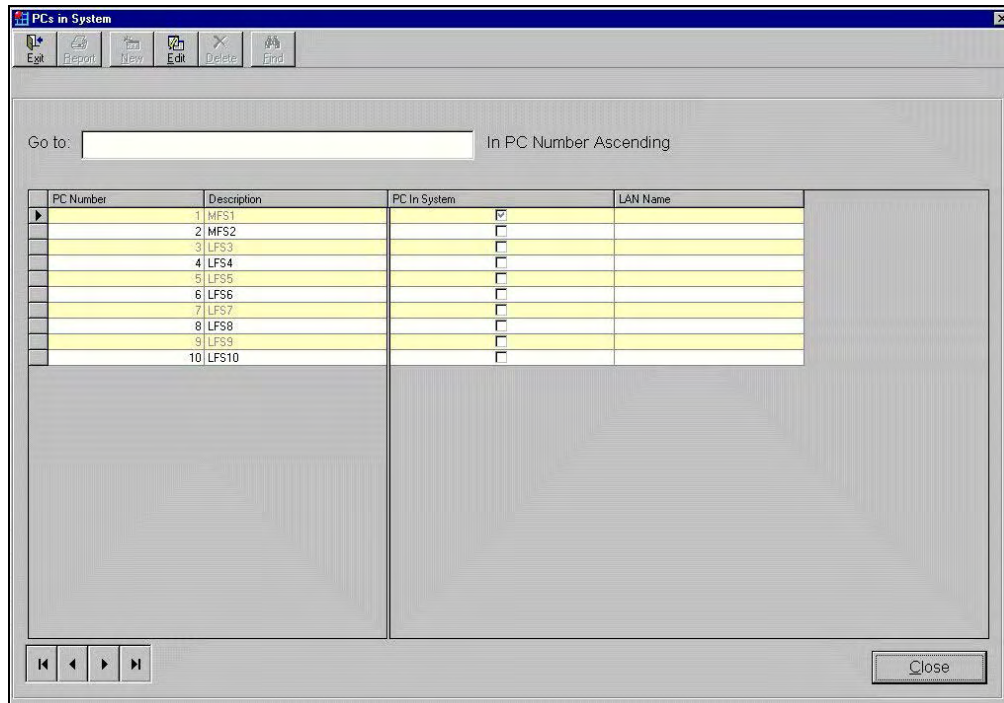
- **Configuring MFSs in System using PC in System**
- **Configuring Active POS terminals**
- **Configuring *CDS* Checkout bank parameters**
- **Configuring *CDS* and R-Session**
- **Setting Up Disk Space Alerts**
- **Setting the System to Auto Log-On**

Configuring MFSs in System using *PC In System*

In order for all MFSs to receive an upgrade, you must configure the MFSs in the system.

➤ **To Configure MFSs in an ISS45 v8 System:**

- 1 In the *Main Menu* tab, browse to the following folder: System Administration\Front Office Parameters\PCs in System.
- 2 Double-click the **PCs in System** icon. The *PCs in System* dialog is displayed.



- 3 Double-click the line of the required MFS or LFS to be configured. The following form is displayed:

- 4 Enter the information in the fields as described in the table below:

Field	Description
PC Number	Click  to select the required PC number. Once selected the MFS number connected to this PC is displayed in the Description field.
Description	The name of the server connected to the PC selected, e.g., MFS1. Note: This field is grayed out; as the MFS1 is configured previously, see <i>Configuring MFSs in System</i> , above.
LAN Name	Enter the name of the Network connection.
PC in System	Check the checkbox to indicate the PC is active in the system.

- 5 Click *Apply* to return to the *PCs in System* dialog.

Follow this procedure to configure all the required MFSs in the system.

➤ **To Configure v7 PCs in System:**

- 1 At Main Menu go to *PCs in System* (6-1-1-2). The *PCs in System* screen displays:

Ver 7.1.1.2005 TIM'S MARKETS PC# 01 MFS1
PCs in System Page 01

1.	PC number	1	<MFS1> in system		[]
2.	PC number	2	<MFS2> in system		[]
3.	PC number	3	<Workstation> in system	...[]	Boot at EOD... []
4.	PC number	4	<Workstation> in system	...[]	Boot at EOD... []
5.	PC number	5	<Workstation> in system	...[]	Boot at EOD... []
6.	PC number	6	<Workstation> in system	...[]	Boot at EOD... []
7.	PC number	7	<Workstation> in system	...[]	Boot at EOD... []
8.	PC number	8	<Workstation> in system	...[]	Boot at EOD... []
9.	PC number	9	<Workstation> in system	...[]	Boot at EOD... []
10.	PC number	10	<Workstation> in system	...[]	Boot at EOD... []
11.					
12.					
13.					
14.					
15.					
16.					
17.					
18.					

[F2] Update

10/29/2007 17:19:00 Copyright StoreNext

- 2 Use the spacebar to check the desired fields (use the Tab key to move between entries and use the F2 key to update the parameters when finished).

Note:

All PCs (MFS & LFS) must be selected to receive upgrades.

Configuring Active POS terminals

POS terminals receive upgrades directly from the MFS1 server, according to store requirements, and must be configured to enable them to receive version upgrades. In most cases, the POS terminals are already configured to receive upgrades, a procedure performed in *POS Configuration*. This defines the upgrade source.

➤ **To Configure Active v8 POS Terminals:**

- 1 In the *Main Menu* tab, browse to the following folder:
System Administration\POS Setup\POS Configuration\POS Configuration.
The *POST Configuration* dialog is displayed:

Go to: In POS No. Ascending

POS No.	POS Type	Active Till	Scale Type	Display	Printer Type	Monitor Type	Touch Screen Type
1	TeamPOS & TeamCOM	☐	None	No	None	VGA B/W	None
2	TeamPOS & TeamCOM	☐	None	No	None	VGA B/W	None
3	TeamPOS & TeamCOM	☐	None	No	None	VGA B/W	None
4	TeamPOS & TeamCOM	☐	None	No	None	VGA B/W	None
5	TeamPOS & TeamCOM	☐	None	No	None	VGA B/W	None
6	TeamPOS & TeamCOM	☐	None	No	None	VGA B/W	None
7	TeamPOS & TeamCOM	☐	None	No	None	VGA B/W	None
8	TeamPOS & TeamCOM	☐	None	No	None	VGA B/W	None
9	TeamPOS & TeamCOM	☐	None	No	None	VGA B/W	None
10	TeamPOS & TeamCOM	☐	None	No	None	VGA B/W	None
11	TeamPOS & TeamCOM	☐	None	No	None	VGA B/W	None
12	TeamPOS & TeamCOM	☐	None	No	None	VGA B/W	None
13	TeamPOS & TeamCOM	☐	None	No	None	VGA B/W	None
14	TeamPOS & TeamCOM	☐	None	No	None	VGA B/W	None
15	TeamPOS & TeamCOM	☐	None	No	None	VGA B/W	None
16	TeamPOS & TeamCOM	☐	None	No	None	VGA B/W	None
17	TeamPOS & TeamCOM	☐	None	No	None	VGA B/W	None
18	TeamPOS & TeamCOM	☐	None	No	None	VGA B/W	None
19	TeamPOS & TeamCOM	☐	None	No	None	VGA B/W	None
20	TeamPOS & TeamCOM	☐	None	No	None	VGA B/W	None
21	TeamPOS & TeamCOM	☐	None	No	None	VGA B/W	None
22	TeamPOS & TeamCOM	☐	None	No	None	VGA B/W	None
23	TeamPOS & TeamCOM	☐	None	No	None	VGA B/W	None
24	TeamPOS & TeamCOM	☐	None	No	None	VGA B/W	None
25	TeamPOS & TeamCOM	☐	None	No	None	VGA B/W	None

Copy Close

- Double-click the relevant POS Type to receive upgrade. The *Post Configuration* dialog is displayed:

POS Number: POS Type: ☒ Active till

General | WinPOS H/W | Non WinPOS 1 | Non WinPOS 2 | Auth. - EFT - I/F | E-Ticket - OPOS

Receipt options

Header/ Footer:

Graphic Logo:

☐ Print VAT receipt

Cashier Queue Length

Prompt cashier every minutes

Alert when count is:

Amounts in Cash Drawer

Minimum: Control Check:

Maximum: Control Check:

General options

Checkout Bank:

Keyboard type:

Debug mode:

☒ Print text to Transaction File

☐ Prompt for Bagger ID

Menusys Assignment:

Till options

Till Profile:

Till Type:

Self Checkout

Self Checkout Type:

Self Checkout Cashier:

Self Checkout POS No.:

- In the top right-hand corner of the screen, check the *Active Till* checkbox.
- Click *Apply* to activate.
- Repeat steps 2 through 4 to configure all required POS terminals in the store.

Note:

If a POS terminal is not marked active, it will still receive the upgrade but there is no guarantee of its success. In addition, it will not appear in the Status of the upgrade process in the system. For further details on the Status of an upgrade process, refer to the section *Upgrading MFS1*, in Chapter 4.

➤ **To Configure Active v7 POS Terminals:**

- 1 At Main Menu go to *Post Configuration* (6-1-4-1).
- 2 Highlight the first Post to update using the arrow (↑↓) keys and press <Enter>. The *Post Configuration* screen displays.

Ver 7.1.1.2005 TIM'S MARKETS PCN 01 MFS1
POST Configuration 1

POS number.....▶ 1	CDS Active [✓]	Scanner:	
POS type.....▶ 5000	[↓]	COM.....▶ 5	
General:		IRQ.....▶ 5	
2X20 kbd. display.....▶ []		Baud Rate.....▶ 9600	[↓]
Generate sound from.....▶ PC SOUND	[↓]	Data Bits.....▶ 8	
Battery backup.....▶ [✓]		Parity.....▶ Even	[↓]
Monitor type.....▶ UGA Color	[↓]	Handshake.....▶ [✓]	
Touch screen type.....▶ None	[↓]	Scale:	
Weight on TeamPOS.....▶ [✓]		Type.....▶ 9000	[↓]
Port for print commands▶ None	[↓]	COM.....▶ Com 4	[↓]
Drawer in use.....▶ [✓]		IRQ.....▶ 5	
connected to printer.▶ []		Baud Rate.....▶ 2400	[↓]
Dual drawer enabled...▶ []			
Printer:		Customer Display:	
Type.....▶ Axiohm 758	[↓]	COM.....▶ 0	
Port.....▶ None	[↓]	IRQ.....▶ 0	
MICR reader in use.....▶ [✓]		Type.....▶ None	[↓]
Enable Check Flip.....▶ [✓]		Baud rate.....▶ None	[↓]

[F2] Update [Auth/API] [F6] Next Page [F9] Prev [F10] Next

10/29/2007 17:17:45 Copyright StoreNext

- 3 In *Post Configuration* press the <TAB> key to move to the **CDS Active** field.
- 4 Press the spacebar to check the field.
- 5 Press <F2>, <F10> to update and advance to next Post.
- 6 Repeat steps 2 through 5 for each POS terminal in the store.
- 7 When finished, press <ESC>.

Defining Checkout Banks Parameters

Checkout Banks represent a department or a group of POS terminals within a store. You can define the percentage of POS terminals in each checkout bank that can be upgraded simultaneously. For example, you may decide to upgrade 50% of the POS terminals in one checkout bank, and 70% of the POS terminals in another checkout bank, ensuring minimum percentage of POS terminals in each checkout bank available for sales.

➤ **To Define v8 Checkout Banks Parameters:**

- 1 In the *Main Menu* tab, browse to the following folder:
System Administration\POS Setup \POS Configuration\Checkout Banks.
- 2 Double-click the **Checkout Banks** icon. The *Checkout Banks* dialog is displayed.

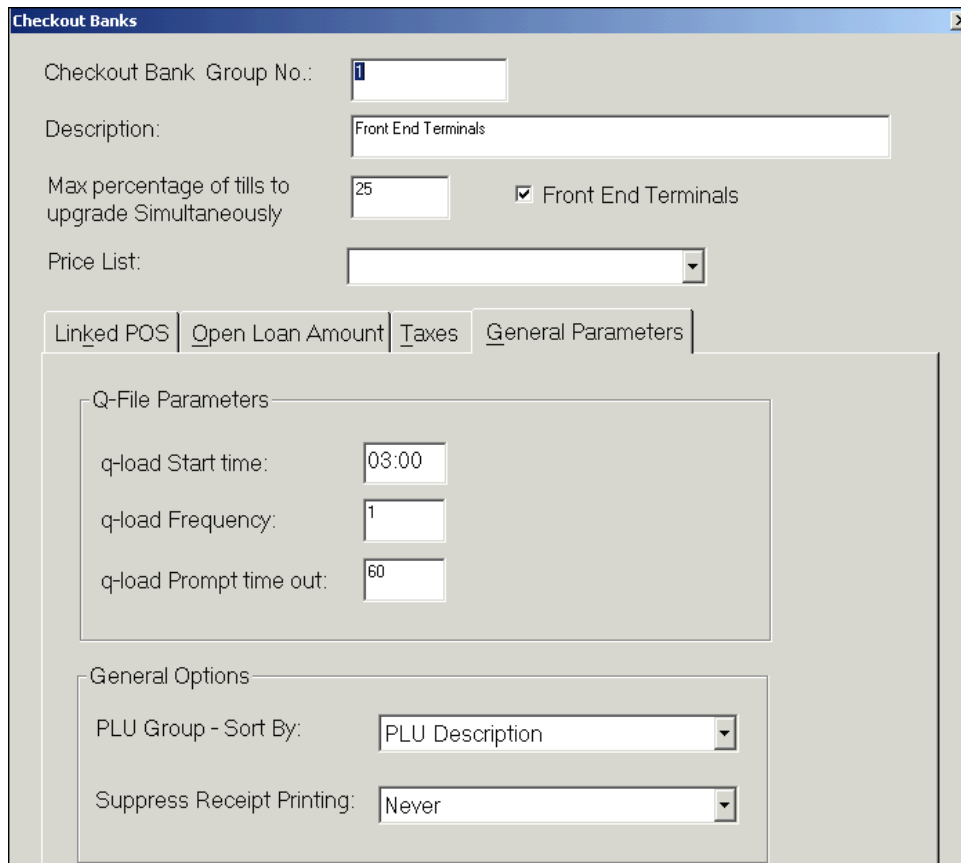
Group Number	Description	Front End POS
1	Front End Terminals	☒
2	Service Desk	☐

The table below describes the following fields:

Field	Description
Group Number	Indicates the number representing the group of POS terminals within the Checkout Bank. A typical store may consist of a main row of checkouts called Front End Terminals, as well as various terminals located in different areas of the store.
Description	The name describing the Group of POS terminals in the Checkout Bank. Each Group represents a different Checkout Bank.
Front End POS	A check in this field indicates the Checkout Bank contains the Front End POS Terminals.

- From the list of available checkout banks, select the Checkout Bank for which you want to specify the maximum percentage of POS Terminals to perform file maintenance simultaneously. Then, click *Edit*, or double-click on the required Checkout Bank in the **Group Number** list.

The *Checkout Banks* dialog is displayed for the selected Checkout Bank:



Checkout Bank Group No.: 1

Description: Front End Terminals

Max percentage of tills to upgrade Simultaneously: 25 ☒ Front End Terminals

Price List: [Dropdown]

Linked POS | Open Loan Amount | Taxes | General Parameters

Q-File Parameters

q-load Start time: 03:00

q-load Frequency: 1

q-load Prompt time out: 60

General Options

PLU Group - Sort By: PLU Description

Suppress Receipt Printing: Never

- 4 In the **Max Percentage of Tills to Perform File Maintenance Simultaneously** field, enter the maximum percentage of POS terminals to be upgraded for this checkout bank at one time.

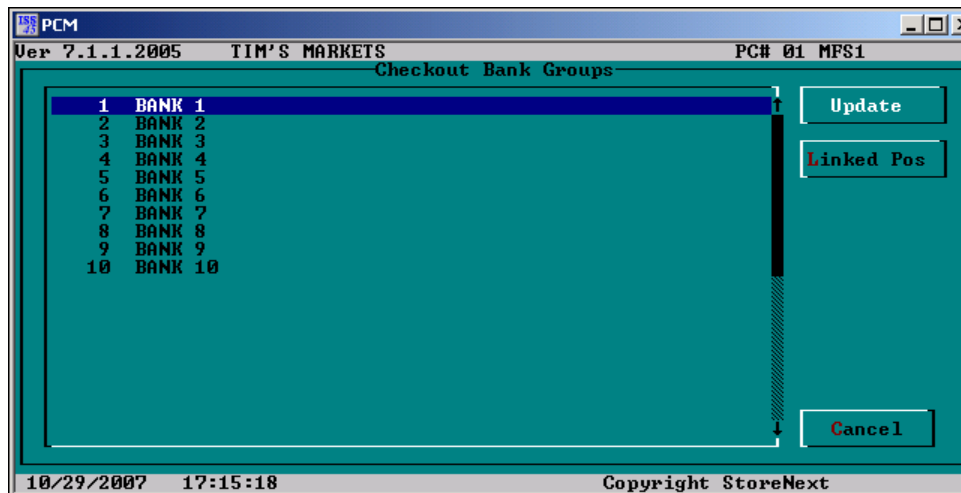
Note:

If the upgrade is for a Fuel system, it is recommended to set this parameter to 100%, allowing all the PFS POS Terminals to start the upgrade together. The reason is that *PumpSrv* and *PosPump*, the PFS components used when Upgrading Fuel systems, are inter-dependent. (For further information refer to *Chapter 6, CDS Support for PFS*)

- 5 Click OK to save changes and close the dialog.

➤ **To Define v7 Checkout Banks Parameters:**

- 1 At Main Menu go to *Checkout Bank Groups* (6-1-4-2). The *Checkout Bank Groups* screen displays:



- 2 Highlight the first Bank to update using the arrow (↑↓) keys and press <Enter>.



- 3 Press the <TAB> key until the "Max percentage of tills to upgrade simultaneously" field is highlighted.
- 4 Enter xxx (where xxx is the percent of bank to upgrade).
- 5 Press <F2>, <F10> to update and advance to next Bank.
- 6 Repeat steps 3 through 5 for each Bank to update.
- 7 When finished, press <ESC>.

Note:

The biggest impact of this setting is in a 24-hour store environment. System impact is dependant on terminals and server processing power and is unique to each site. See Rsession parameters below).

Defining CDS and R-Session Parameters

An R-Session is a communication channel that controls network traffic between POS terminals and Servers. Defining R-Session parameters enables CDS software upgrades, and prevents overloads in MFS1.

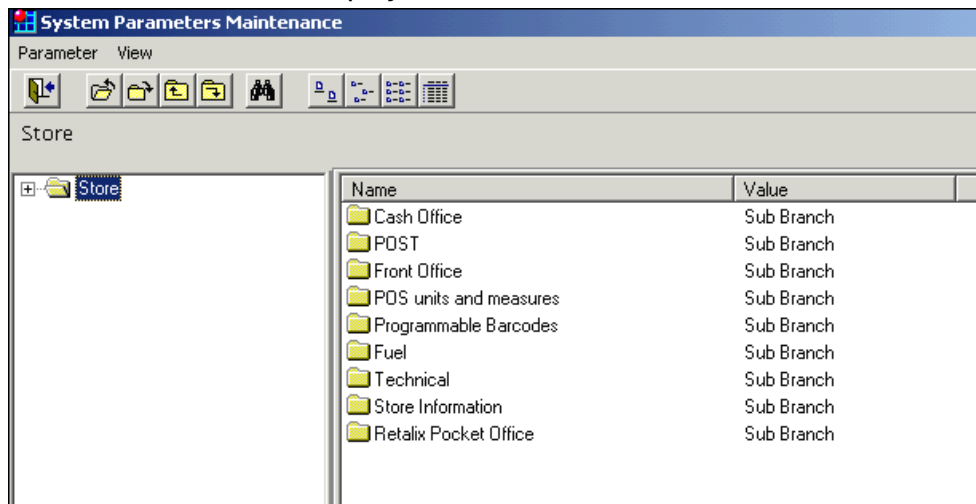
Once a POS terminal has completed the upgrade process, it immediately frees up the R-Session for MFS1.

➤ **To Define the v8 CDS and R-Session Parameters:**

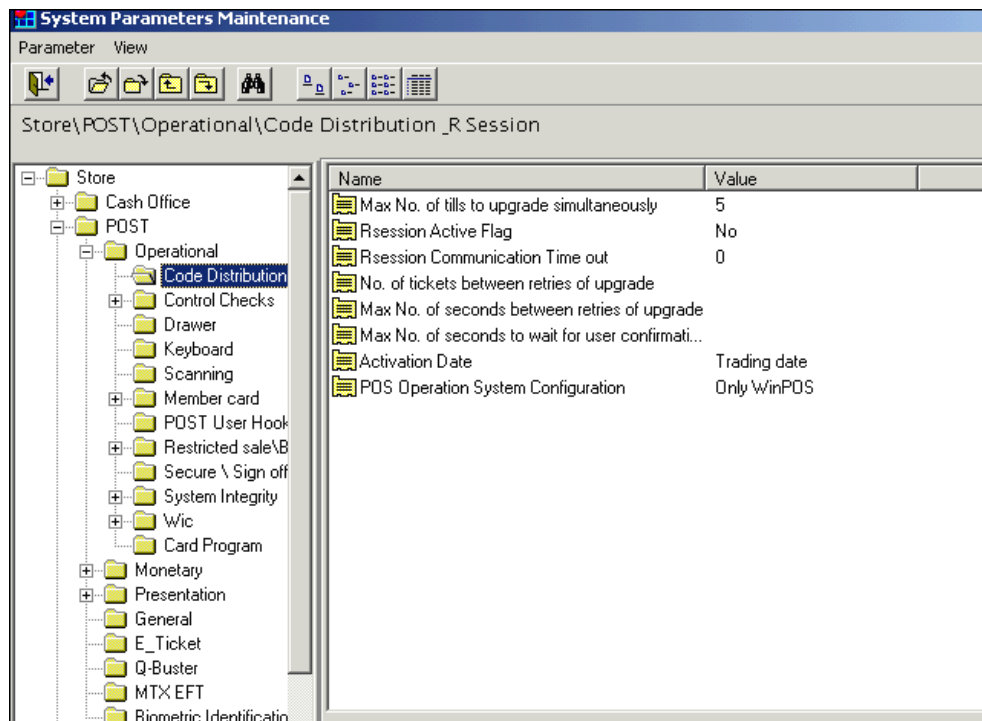
Note:

The Rsession Parameters can also be established through the following ISS45 v8 folder option: System Administration\Technical Support\Rsession Parameters.

- 1 In the *Main Menu* tab, browse to the following folder:
System Administration\General System Parameters\General System Parameters.
- 2 Double-click the **General System Parameters** icon. The *System Parameters Maintenance* menu is displayed, as follows:



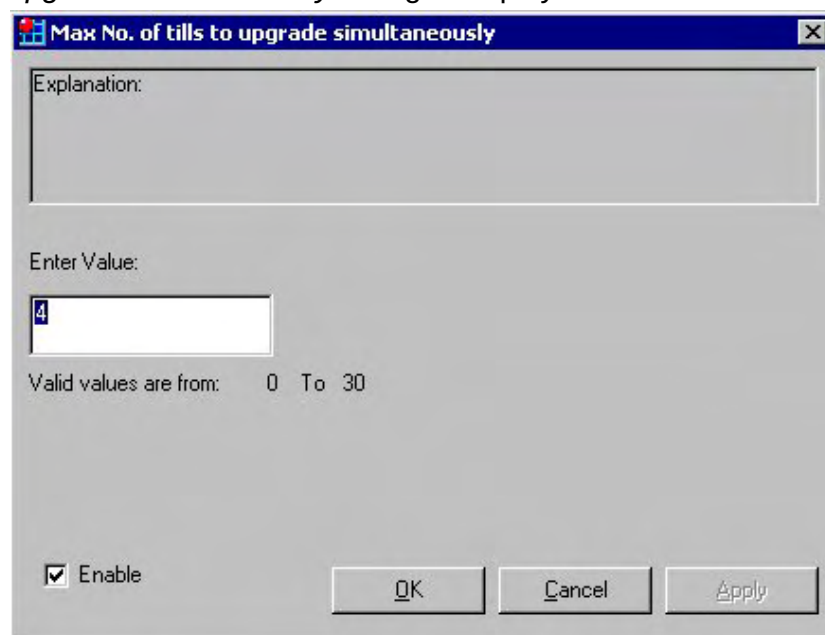
- 3 From the *Store Menu*, double-click the **POST** icon and browse to the following folder: Store\POST\Operational.
- 4 From the *Store\POST\Operational Menu*, double-click the **Code Distribution & R-Session** icon. The list of dialogs in the Code Distribution and R-Session folder are displayed as follows:



You have accessed the *CDS* and *R-Session* parameters. Continue to define each one to complete the *CDS* Setup.

➤ **To define v8 Max No. of Tills to upgrade Simultaneously:**

- 1 From the *Code Distribution* and *R-Session* folder, double-click the **Max No. of tills to upgrade simultaneously** icon. The *Max No. of POS Terminals to upgrade simultaneously* dialog is displayed:



- 2 In the **Enter Value** field, enter the maximum number of POS terminals that will upgrade simultaneously. For example, if your store has 40 POS terminals, you may decide to upgrade only 20 POS terminals simultaneously.

Notes:

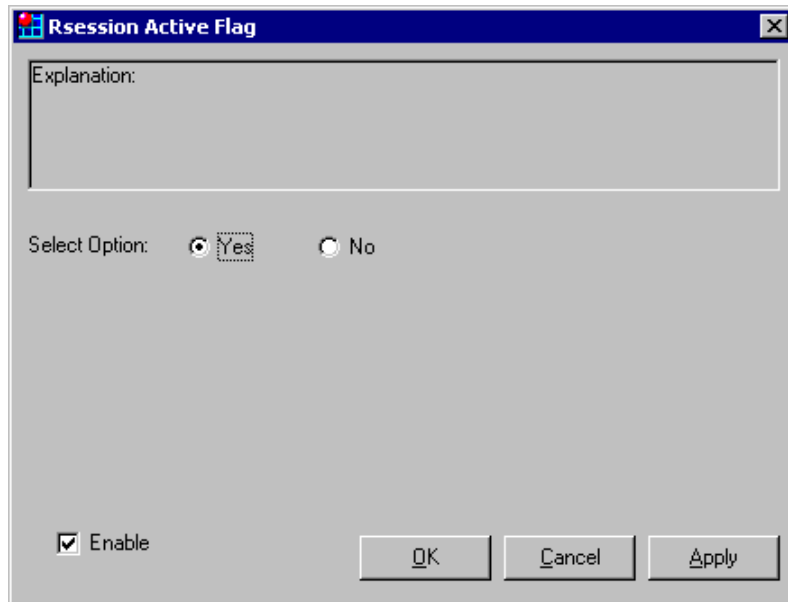
1. The maximum number of POS terminals is affected by the CPU power of your server, and by your network's capacity. This number should be determined after running tests in your specific environment.
2. A POS terminal that failed the upgrade process will not automatically free its Rsession. This will prevent all other POS terminals in the store from failing as well, as a result of a corrupted upgrade package.

- 3 Select the **Enable** checkbox to enable the viewing of this parameter.
- 4 Click *Apply* to save changes.
- 5 Click *OK* to close the dialog.

➤ **To Define v8 Rsession Active Flag:**

The *Rsession Active Flag* indicates whether an Rsession is being used on the system, controlling staggered upgrading of POS Terminals.

- 1 From the *Code Distribution and R-Session* folder, double-click the **Rsession Active Flag** icon. The *Rsession Active Flag* dialog is displayed:



- 2 Select the appropriate **Select Option** field, by selecting either the Yes radio button to activate R-Session, or the No radio button to de-activate R-Session.
- 3 Select the **Enable** checkbox to enable the viewing of this parameter.
- 4 Click *Apply* to save changes.

Note:

Please stop and restart the POS_Server_Manager service after changing the Rsession Active flag.

Note:

The **Rsession Communication Time out** parameter should always be set to '0'.

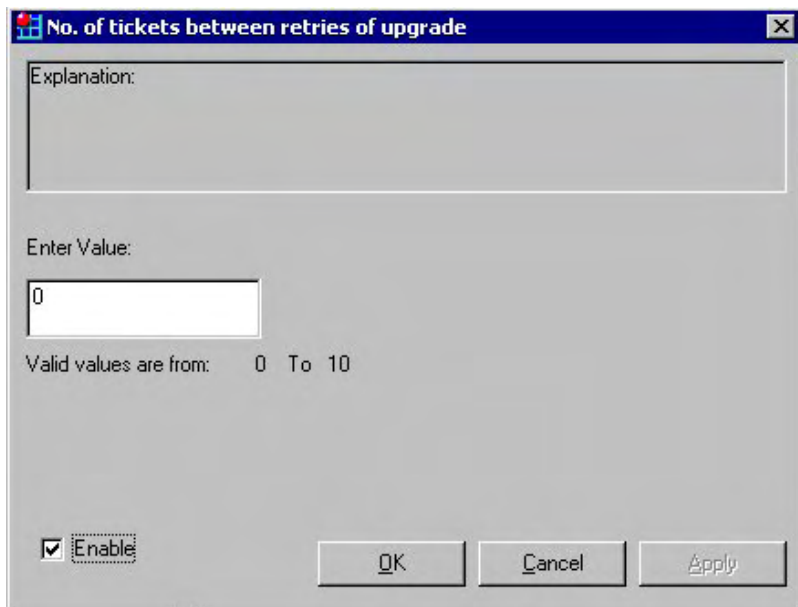
➤ **To Define the v8 No. of Tickets Between Retries of upgrade:**

CDS enables POS terminals to upgrade automatically when in *Sign-Off* mode. In *Sign-On* mode, CDS offers several options to enable seamless software upgrades without interfering with normal sales transactions.

Note:

The following three parameters can also be established through the following ISS45 v8 folder option: System Administration\Technical Support\Code distribution Management Parameters (POS Tab).

- 1 From the *Code Distribution and R-Session* folder, double-click the **No. of tickets between retries of upgrade** icon. The *No. of Tickets Between Retries of upgrade* dialog is displayed:



- 2 In the **Enter Value** field, enter the number of sales that can be completed before CDS tries to upgrade the POS terminal and the message "Proceed with software upgrade?" reappears on the POS terminal (in *Sign-On* mode).

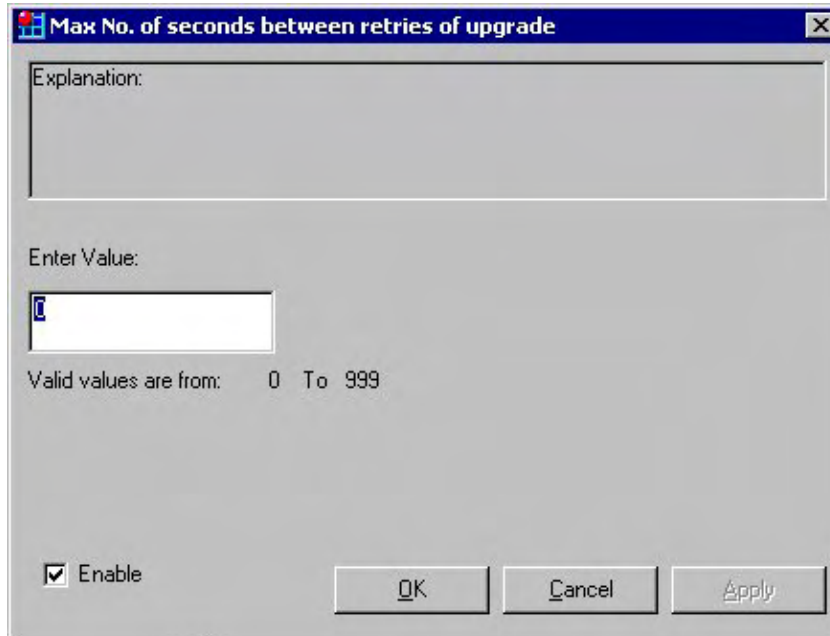
Note:

If the value is '0', the message will not display until you sign-off. The upgrade will then start without any warning.

- 3 Select the **Enable** checkbox to enable the viewing of this parameter.
- 4 Click *Apply* to save.
- 5 Click *OK* to close the dialog.

➤ **To Define the v8 Max No. of Seconds Between Retries of Upgrade:**

- 1 From the *Code Distribution and R-Session* folder, double-click the **Max No. of Seconds Between Retries of Upgrade** icon. The *Max No. of Seconds Between Retries of Upgrade* dialog is displayed:



Max No. of seconds between retries of upgrade

Explanation:

Enter Value:

Valid values are from: 0 To 999

☒ Enable

OK Cancel Apply

- 2 In the **Enter Value** field, enter the interval (in seconds) after which a message will reappear prompting "Proceed with the software upgrade?" and CDS will retry the software upgrade.

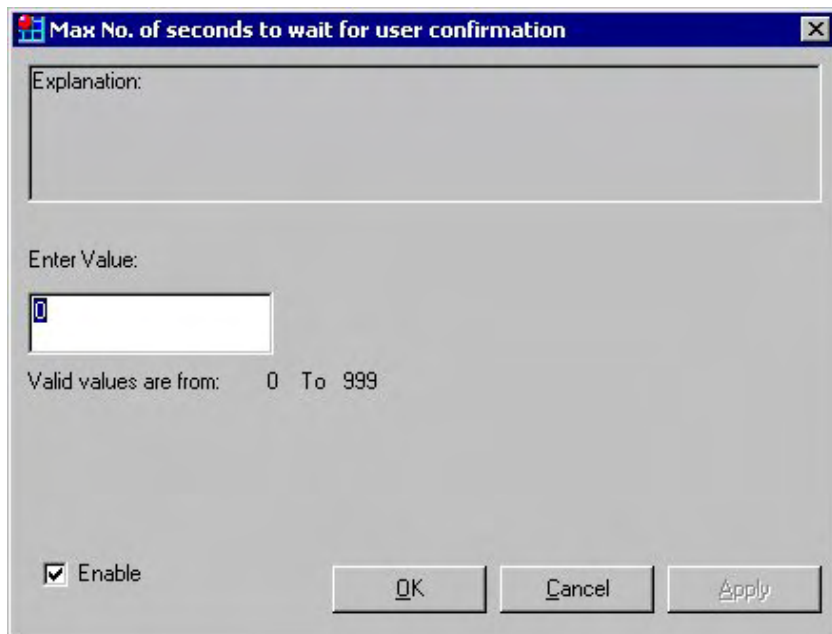
Note:

If the value is '0', the message will not be displayed until you sign-off. The upgrade will start without any warning on signing-off.

- 3 Select the **Enable** checkbox to enable the viewing of this parameter.
- 4 Click *Apply* to save.
- 5 Click *OK* to close the dialog.

➤ **To Define the v8 Max No. of Seconds to Wait for User Confirmation:**

- 1 From the *Code Distribution and R-Session* folder, double-click the **Max No. of seconds to wait for user confirmation** icon. The *Max No. of Seconds to Wait for User Confirmation* dialog is displayed:



- 2 In the **Enter Value** field, enter the interval (in seconds) that CDS will wait for cashier confirmation before it automatically starts the upgrade.

Note:

If the value is zero, the on-screen message prompting the cashier confirmation to start the upgrade will not display until you sign-off. The upgrade will then start without any warning on signing-off.

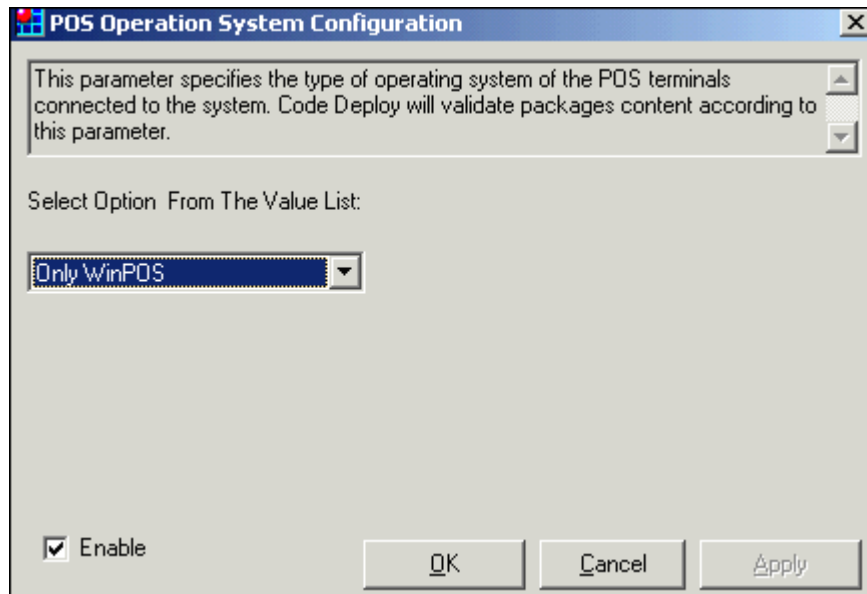
- 3 Select the **Enable** checkbox to enable the viewing of this parameter.
- 4 Click *Apply* to save.
- 5 Click *OK* to close the dialog.

Note:

For further details on the above parameters refer to section on *Upgrading the POS terminals* in *Chapter 4.*)

➤ **To Define the v8 POS Operation System Configuration:**

- 1 From the *Code Distribution and R-Session* folder, double-click the **POS Operation System Configuration** icon. The *POS Operation System Configuration* dialog is displayed:



- 2 In the **Select Option From the Value List** field, click the field selection arrow and choose the type of Operating System of the POS terminals within the Store Configuration.

Note:

CDS will validate packages content according to this parameter. Options include **Only WinPOS**, **Only LinPOS** or **WinPOS & LinPOS**.

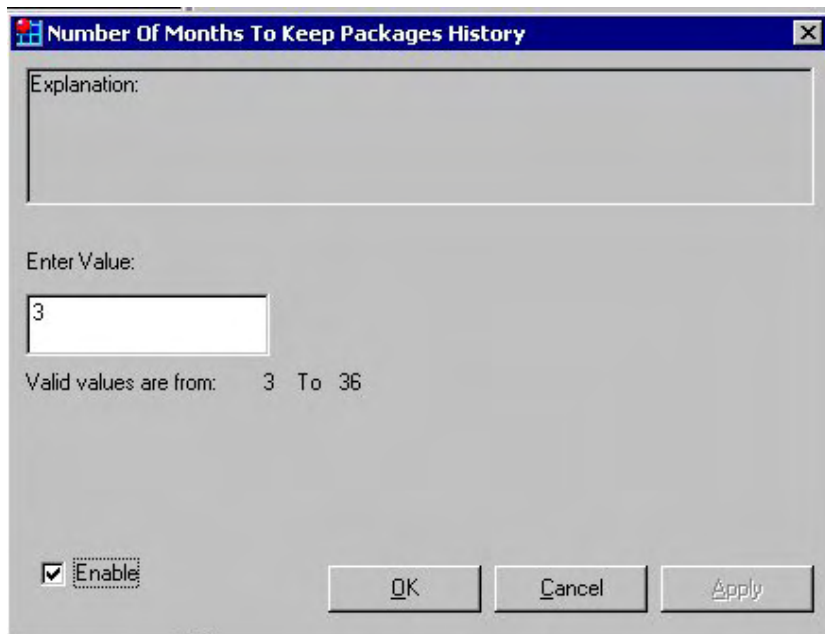
- 3 Select the **Enable** checkbox to enable the viewing of this parameter.
- 4 Click *Apply* to save.
- 5 Click *OK* to close the dialog.

➤ **To Define the v8 Number of Months to Keep Packages History:**

Note:

The following parameter is located in the Store \ Technical folder.

- 1 From the *Technical* folder, double-click the **Number of Months to Keep Packages History** icon. The *Number of Months to Keep Packages History* dialog is displayed:



Number Of Months To Keep Packages History

Explanation:

Enter Value:

3

Valid values are from: 3 To 36

☒ Enable

OK Cancel Apply

- 2 In the **Enter Value** field, enter number of months, after which packages will be deleted from **Done** and **Reject** folders.

Note:

Package is moved to **Done** after upgrade is completed.) Default is 3 months.

- 3 Select the **Enable** checkbox to enable the viewing of this parameter.
- 4 Click *Apply* to save.
- 5 Click *OK* to close dialog.

➤ **To Define the v7 R-Session Parameters:**

Note:

Rsession parameters control how many POSTs can be upgraded simultaneously and how the network between MFS1, MFS2, LFSs and each POS is impacted.

- 1 At Main Menu go to *Rsession Parameters* (6-1-1-11-2). *The Rsession Parameters Screen displays:*

Ver 7.1.1.2005 TIM'S MARKETS PC# 01 MFS1

RSession Parameters

1. Rsession active..... [✓]
 2. Max No. of tills to upgrade simultaneously from one server..... 1
 3. Rsession communication time out..... 0

[F2] Update

10/29/2007 17:20:50 Copyright StoreNext

- 2 Key enter the following fields (use the Tab key to move between entries and use the F2 key to update the parameters when finished).

Parameter	Comments
Rsession active	This parameter enables the Rsession feature and associated parameters. It is recommended to always check the Rsession Active checkbox. Selecting this checkbox ensures that the system will not overload.
Max No. of Tills to upgrade simultaneously from one server	This parameter specifies the maximum number of POS terminals to upgrade simultaneously. For example, if the store has 20 POS terminals, they may elect to upgrade only 10 POS terminals simultaneously. (Dictated by server power and network capacity).
Rsession communication timeout	Always set to 0. It is the responsibility of the POS to get and release Rsessions from MFS.

Note:

Please stop and restart the POS_Server_Manager service after changing the Rsession Active flag.

➤ **To Define CDS in a v7 System:**

- 1 From the *Main Menu*, browse to *Code Management Parameters* (6-1-1-11-1). The *Code Distribution Parameters* displays:

```

Ver 7.1.1.2005    TIM'S MARKETS    PC# 01 MFS1
Code Distribution Parameters
1. No. of tickets between retries of upgrade..... 2
2. Max No. of seconds between retries of upgrade..... 120
3. Max No. of seconds to wait for user confirmation..... 60
4. Max No. of minutes to wait for 2nd PC synchronization..... 2
5. Number of months to keep records in PACKAGES.QDX..... 3

[F21 Update]

10/29/2007 17:22:55    Copyright StoreNext
  
```

Note:

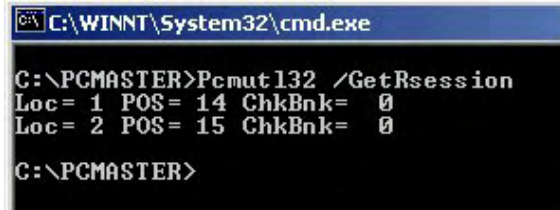
Code Distribution enables terminals to automatically upgrade in *Sign-Off* mode, based on Rsession and Checkout Bank Parameters. Parameters 1, 2 & 3 are for POSTS in *Sign-On* mode. Parameters 4 & 5 are for Back Office.

- 2 Key enter the following fields (use the Tab key to move between entries and use the <F2> key to update the parameters when finished).

Parameter	Comments
No. of tickets between retries of upgrade	If the upgrade is declined by cashier, this parameter specifies the number of sale tickets that can be processed after which a cashier prompt message will reappear and CDS will retry the POS terminal software upgrade in Sign-on mode.
Max No. of seconds between retries of upgrade	If upgrade declined by cashier, enter the number of seconds after which the message will reappear and upgrade is retried (when in "No-Sale" mode).
Max No. of seconds to wait for user confirmation	This parameter specifies the interval (in seconds) that CDS will wait for cashier confirmation before it automatically starts the upgrade.
Max. No. of minutes to wait for 2 nd PC synchronization	Enter the maximum number of seconds that Code Distribution will wait to synchronize the 2 nd PC during the upgrade. (Used for <i>High Sensitivity</i> mode – see below).
Number of Months to keep records in PACKAGES.QDX	Enter the months of software upgrade information in the PACKAGES.QDX file.

➤ **To View Which POS Terminals Have Obtained an R-Session:**

- 1 From Windows **Main Menu** select **Start**, and then **Run**.
- 2 In **Run Dialog**, enter **cmd** and press **OK**.
- 3 In the **Command Window**, enter **CD:\pcmaster**
- 4 Type in **Pcmutl32 /GetRsession** and press **Enter**. The results display:



```
C:\WINNT\System32\cmd.exe

C:\PCMASTER>Pcmutl32 /GetRsession
Loc= 1 POS= 14 ChkBnk= 0
Loc= 2 POS= 15 ChkBnk= 0

C:\PCMASTER>
```

The above screenshot shows that POS terminals #14 and #15 have obtained Rsessions.

- 5 Type in **Exit** and press **Enter**. The command window is closed.

Notes:

1. Using **Pcmutl32** you can simulate the obtaining and releasing of Rsessions by a POS terminal.
2. To simulate the obtaining of an Rsession of a POS terminal, from the Command Window enter: **C:\pcmaster\Pcmutl32 /SetSentry /sPosNo=Xnn**
3. To release simulation of POS terminal obtaining Rsession, from the Command Window enter: **C:\pcmaster\Pcmutl32 /ResetSentry /sPosNo=Xnn**

Warning:

This simulation procedures described above should only be used by technical personnel to prevent a system malfunction.

Defining Parameters for High-Sensitivity Packages

The three v8 procedures described below, are intended only for *High-Sensitivity* packages. (For further information on *High-Sensitivity & Low Sensitivity* packages see *Overview* in *Chapter 4*). These parameters ensure that if one of the MFSs completes a task before the others, it will wait for the others so that they will all remain synchronized:

- To Define Max No of Minutes to Wait for MFS1/MFS2 Synchronization Before Starting Upgrade,
- To Define Max No of Minutes to Wait for MFS1/MFS2 Synchronization Before Starting Boot,
- To Define Max No of Minutes to Wait for MFS1/MFS2 Synchronization Before Ending Upgrade.

➤ **To Define the v8 Max No. of Minutes to Wait for MFS 1 and MFS 2 Synchronization Before Starting Upgrade:**

- 1 In the *Main Menu* tab, browse to the following folder:
System Administration\Technical Support.
- 2 Double-click the **Code distribution Management Parameters** icon. The *Code distribution Management Parameters* window is displayed (with the General Tab selected):

The screenshot shows a dialog box titled "Code distribution Management Parameters". It has three tabs: "General", "POS", and "Restricted Times". The "General" tab is active. Inside the dialog, there are four labeled input fields on the left and three empty input boxes on the right. The first field is "Number of months to keep history packages:" with a value of "3" in the box. The second field is "Max No. of minutes to wait for Mfs's synchronization before starting upgrade:". The third field is "Max No. of minutes to wait for Mfs's synchronization before starting boot:". The fourth field is "Max No. of minutes to wait for Mfs's synchronization before ending upgrade:". At the bottom of the dialog are three buttons: "OK", "Cancel", and "Apply".

- 3 (Optional) By default the **Number of months to keep history packages** field is set to "3" (to keep packages from the previous three months on hand prior to deleting). If desired, change this housekeeping parameter.
- 4 In **Max No. of Minutes to Wait for MFS1/MFS2 Synchronization Before Starting Upgrade** field, enter number of minutes to wait for MFS1/MFS2 synchronization before CDS starts upgrading. (Default is 60).
- 5 In the **Max No. of Minutes to Wait for MFS1/MFS2 Synchronization Before Starting Boot** field, enter number of minutes to wait for MFS1/MFS2 synchronization before starting boot. (Default is 90).
- 6 In the **Max No. of Minutes to Wait for MFS1/MFS2 Synchronization Before Ending Upgrade** field, enter number of minutes to wait for MFS1/MFS2 synchronization before ending the upgrade. (Default is 60).
- 7 Click *Apply* to save.
- 8 Click *OK* to close the dialog.

Defining Blackout Periods

ISS45 can be configured to prevent Code Distribution operations from occurring during designated times on designated days.

Note:

If a Code Distribution begins on MFS1 before a Restricted Time, the store will continue the upgrade process. If the package cannot run because of a Restricted Time, it will be retried after the next EOD and Alert 2975 (Blackout Period) will be issued.

➤ **To Define v8 Blackout Periods:**

- 1 In the *Main Menu* tab, browse to the following folder:
System Administration\Technical Support.
- 2 Double-click the **Code distribution Management Parameters** icon. The *Code distribution Management Parameters* window is displayed (with the General Tab selected).
- 3 Click the **POS** tab. The Restricted Times Tab displays:

The screenshot shows the 'Code distribution Management Parameters' window with the 'POS' tab selected. The 'Restricted Times' section is active, displaying a table for selecting restricted times. The table has columns for 'Start' and 'End' times, and a 'Select Days' section with checkboxes for each day of the week. The 'Restricted Times' tab is selected.

Select Restricted Times	Start	End	Select Days
Monday through Thursday:	00:00	00:00	☐ Sunday
Friday:	00:00	00:00	☐ Monday
Saturday:	00:00	00:00	☐ Tuesday
Sunday:	00:00	00:00	☐ Wednesday
			☐ Thursday
			☐ Friday
			☐ Saturday

Buttons: OK, Cancel, Apply

- 4 Set the options and parameters as described below:

Parameter	Comments
Start	Enter the beginning time (i.e. 23:00 = 11:00 PM) of the range to exclude CDS operations (for the designated day or day range).
End	Enter the ending time of the range to exclude CDS operations (for the designated day or day range). After this time Code Distribution can run.
Select Days	Place a check mark next to the days where the CDS restrictions should apply.

- 5 Click *Apply* to save.
- 6 Click *OK* to close the dialog.

➤ **To Define v7 Blackout Periods:**

- 1 From the *Main Menu*, browse to *Blackout Periods (6-1-1-11-4)*. The *Blackout Period Setup* screen displays:

Parameter	Value
1. No. of tickets between retries of upgrade.....	2
2. Max No. of seconds between retries of upgrade.....	120
3. Max No. of seconds to wait for user confirmation.....	60
4. Max No. of minutes to wait for 2nd PC synchronization.....	2
5. Number of months to keep records in PACKAGES.QDX.....	3

[F2] Update

10/29/2007 17:22:55 Copyright StoreNext

- 2 Key enter the following fields (use the Tab key to move between entries and use the <F2> key to update the parameters when finished):

Parameter	Comments
Start	Enter the beginning time (i.e. 23:00 = 11:00 PM) of the range to exclude CDS operations (for the designated day or day range).
End	Enter the ending time of the range to exclude CDS operations (for the designated day or day range). After this time Code Distribution can run.
Select Days	Place a check mark next to the days where the CDS restrictions should apply.

Defining the Activation Date

Defining the *Activation Date* provides the option of scheduling the dates on which a package will be *Registered* and *Administered* in the Store. Each option can be applied to any type of store, whether it operates on a 24hr a day schedule or not.

Defining the *Activation Date* prevents version mismatching in the following way: Each upgrade package arrives at the Store with a pre-assigned **Package Date**.

Package Date refers to the date defined in the .CTL file. This is the execution date scheduled to *Register* and *Administer* the package.

The **Package Date** is compared to the **Trading Date**, **System Date** or **Next Working Date** before it can be *Registered*, ensuring that the upgrade version sent in the package will not upgrade to a lower version than that in the Store.

When selecting the *Activation Date*, you have the following options:

- **Trading Date:** Trading Date is a 24-hour period that begins after the last EOD process is completed and ends when the next EOD process is completed.
- **System Date:** System Date is a 24-hour period that begins at 12 midnight and ends at 12 midnight. It is the calendar date that is displayed on your PC system in the Control Panel of the operating system.
- **Next Working Date:** The Next Working Date is the date of the nearest EOD that has not begun yet. In *CDS* the Next Working Date is used to define the *Activation Date* in the following scenario: If the Store runs EOD every day, but will be closed for a period of time, for example, a public holiday, the date on which a Store reopens is the Next Working Date.

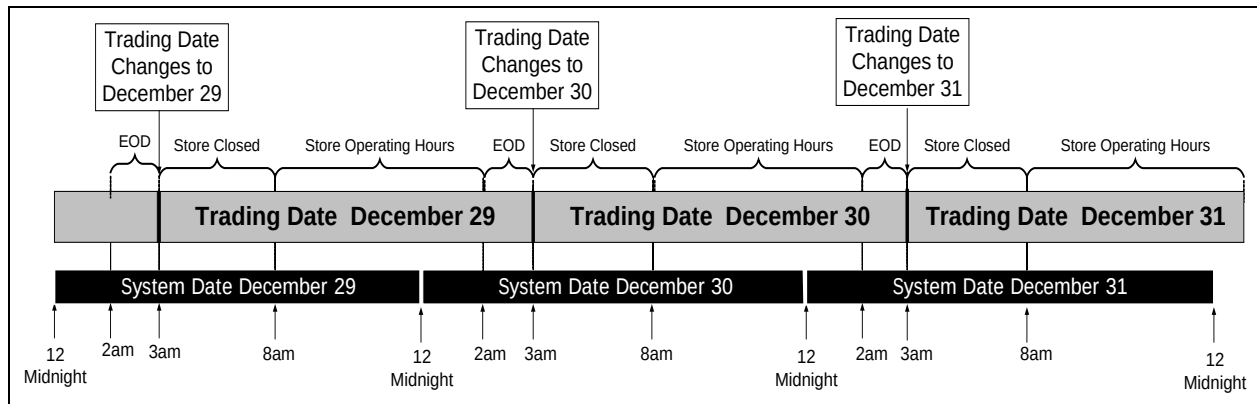
Later in this section, the effects of the *Activation Date* on *Registering* and *Administering* a package are explained:

- During EOD
- Not During EOD (Via the Application)

Below are 6 different scenarios, illustrating how **Trading Date** (2 scenarios), **System Date** (2 scenarios), and **Next Working Date** (2 scenarios), are applied in various stores:

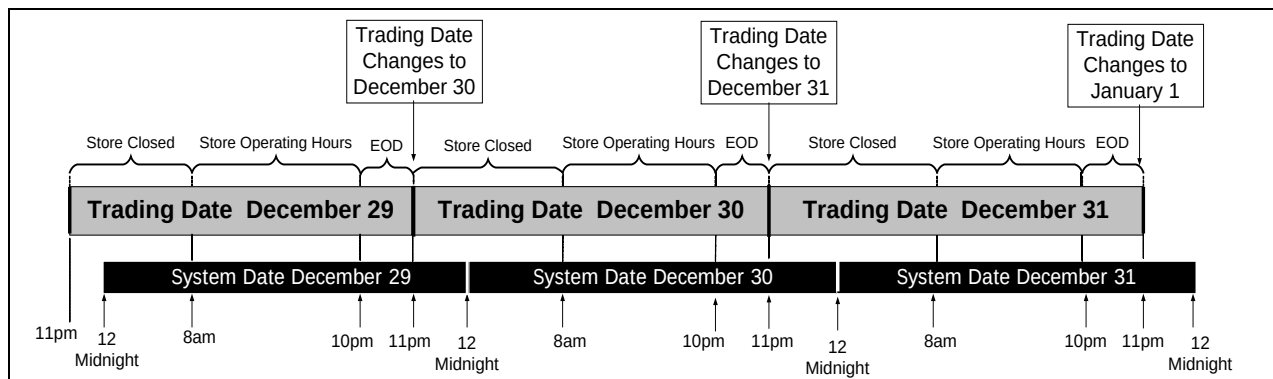
The following illustration refers to a Store that operates from 8am to 2am:

- The **Trading Date** changes at the end of EOD - in this case after midnight.
- The **System Date** changes at 12 midnight.



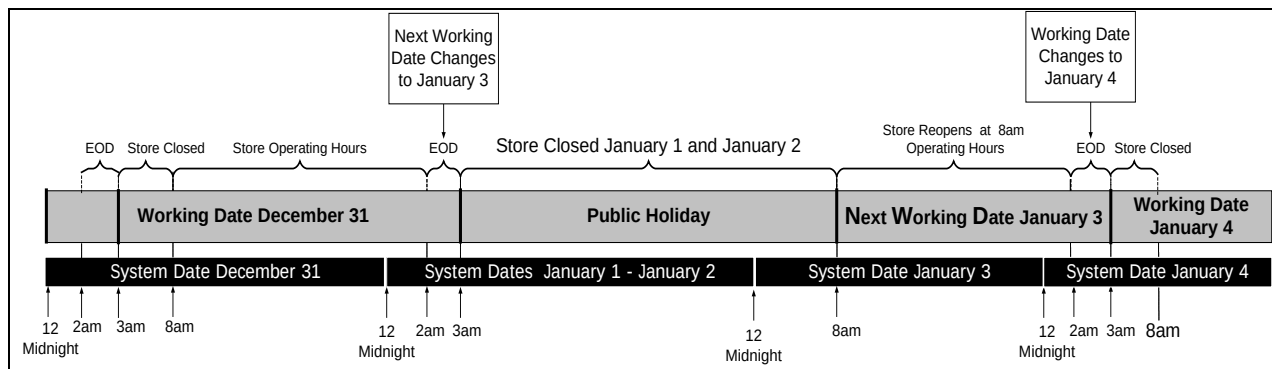
The following illustration refers to a Store that operates from 8am to 10pm:

- The **Trading Date** changes at the end of EOD – in this case before midnight.
- The **System Date** changes at 12 midnight.



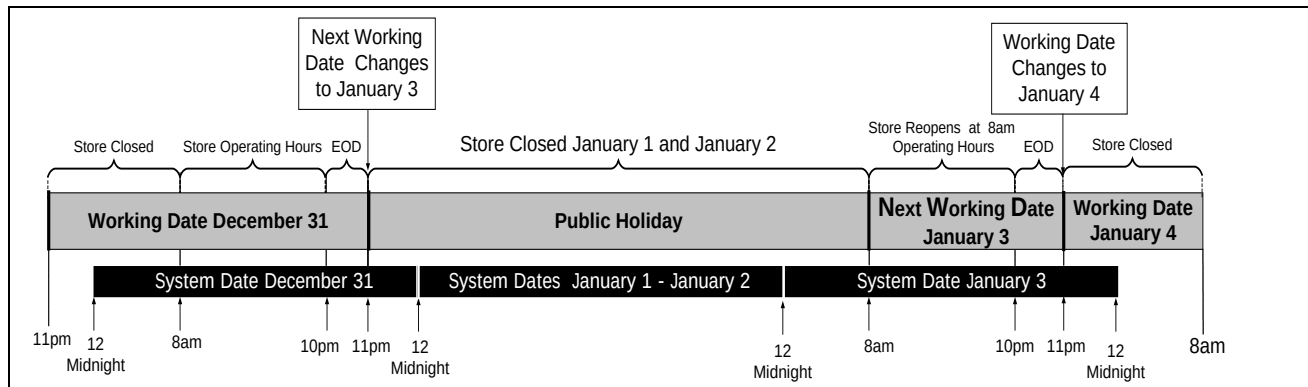
The following illustration refers to a Store that operates from 8am to 2am, and closes for a 2-day holiday:

- The **Next Working Date** is set for January 3, and actually begins after EOD on December 31. It is effective on the day and time that the Store reopens on January 3, 8am.
- January 1 and January 2 are the days on which the Store is closed due to a public holiday and are not regarded as Working Dates. January 3 is the Next Working Date.
- The **System Date** changes at 12 midnight every day.



The following illustration refers to a Store that operates from 8am to 10pm, and closes for a 2-day holiday:

- The **Next Working Date** is set for January 3, and actually begins after EOD on December 31. It is effective on the day and time that the Store reopens on January 3, 8am.
- January 1 and January 2 are the days on which the Store is closed due to a public holiday, and are not regarded as Working Dates. January 3 is the Next Working Date.
- The **System Date** changes at 12 midnight every day.



Scheduling Activation Date for EOD

The dates scheduled to *Register* and *Administer* a package must be in accordance with the specifications of the parameters described below; otherwise the package will be rejected, and the system will send an alert indicating a problem.

Note:

The date in the .CTL file is entered as YYYYMMDD, e.g., 20080101. For further details on the .CTL file, refer to *Chapter 7, Package Structure*.

Quick Reference to Scheduling Activation Dates During EOD

The following Quick Reference table specifies the formula used when **CDS** *Registers* or *Administers* a package:

Field	Effect on Package during Registration	Effect on Package during Administration
Trading Date	If Package Date is Greater than or Equal to the Trading Date it will be <i>Registered</i> .	If Package Date is Less than or Equal to the Trading Date it will be <i>Administered</i> .
System Date	If Package Date is Greater than or Equal to the System Date it will be <i>Registered</i> .	If Package Date is Less than or Equal to the System Date it will be <i>Administered</i> .
Next Working Date	If Package Date is Greater than or Equal to the current Working Date it will be <i>Registered</i> .	If Package Date is Less than the Next Working Date it will be <i>Administered</i> .

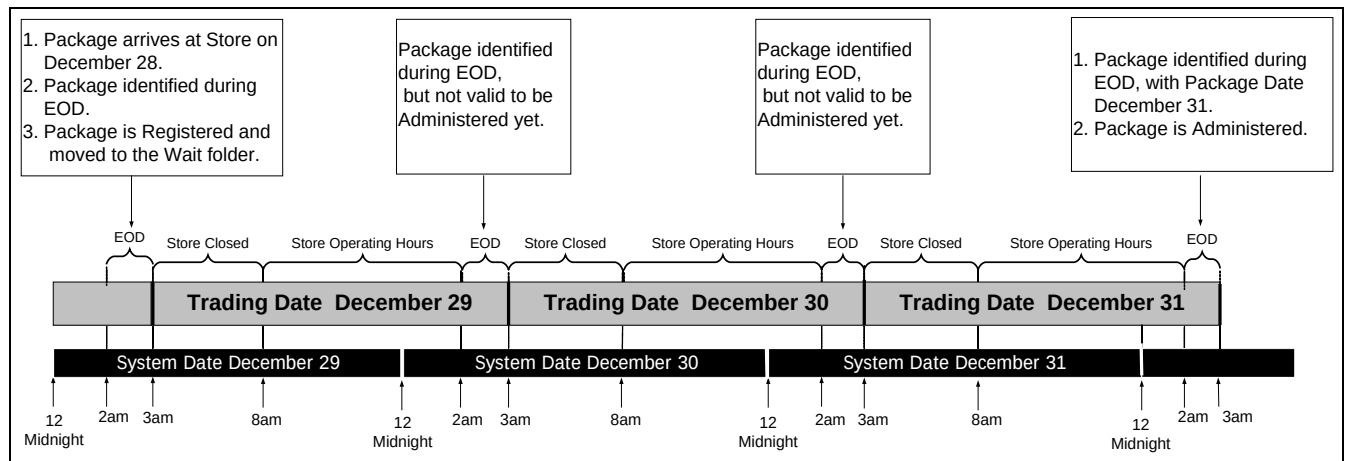
The following 6 illustrations reflect various scenarios when defining the *Activation Date* of a package, specifying when the package would be *Registered* and *Administered*:

- Selecting Trading Date in a Store that operates 8am-2am.
- Selecting Trading Date in a Store that operates 8am-10pm.
- Selecting System Date in a Store that operates 8am-2am.
- Selecting System Date in a Store that operates 8am-10pm.
- Selecting Next Working Date in a Store that operates 8am-2am.
- Selecting Next Working Date in a Store that operates 8am-10pm.

Selecting Trading Date:

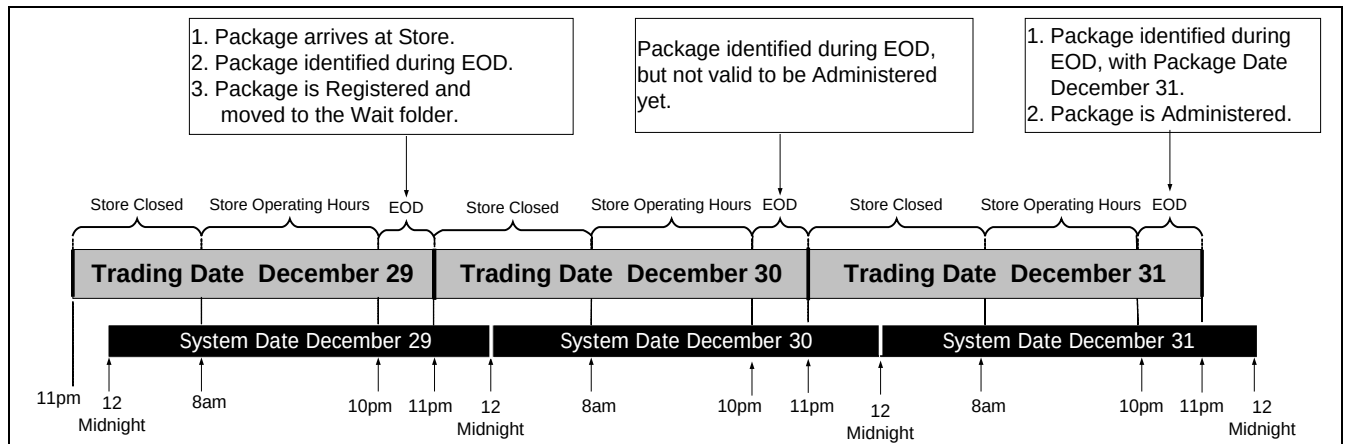
The following illustration refers to a Store that operates from 8am to 2am and receives a package upgrade with Package Date of December 31:

- Package is *Registered* during EOD if the Package Date is **Greater Than** or **Equal To** Trading Date.
- Package is *Administered* during EOD if package Date is **Less Than** or **Equal To** Trading Date.



The following illustration refers to a Store that operates from 8am to 10pm and receives a package upgrade with Package Date of December 31:

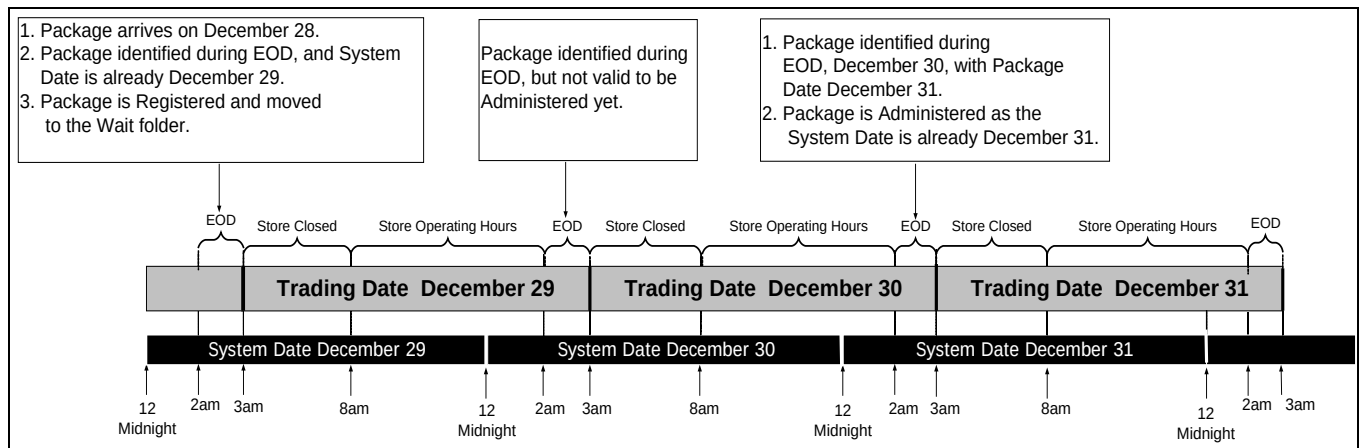
- Package is *Registered* during EOD if the Package Date is **Greater Than** or **Equal To** Trading Date.
- Package is *Administered* during EOD if the package Date is **Less Than** or **Equal To** Trading Date.



Selecting System Date:

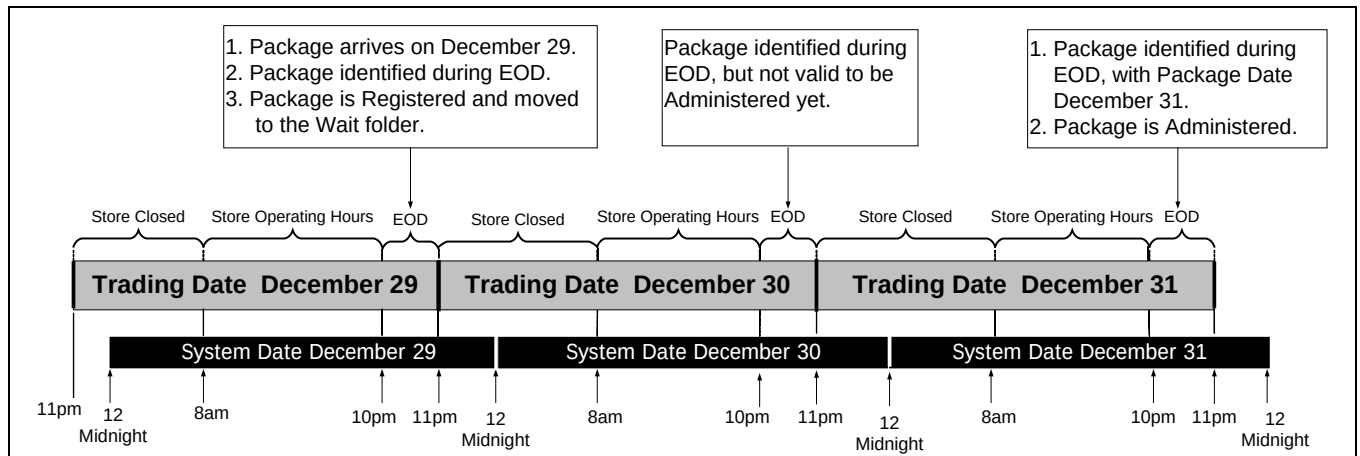
The following illustration refers to a Store that operates from 8am to 2am and receives a package upgrade with Package Date of December 31:

- Package is *Registered* during EOD if the Package Date is **Greater Than or Equal To** System Date.
- Package is *Administered* during EOD if the package Date is **Less Than or Equal To** System Date.



The following illustration refers to a Store that operates from 8am to 10pm and receives a package upgrade with Package Date of December 31:

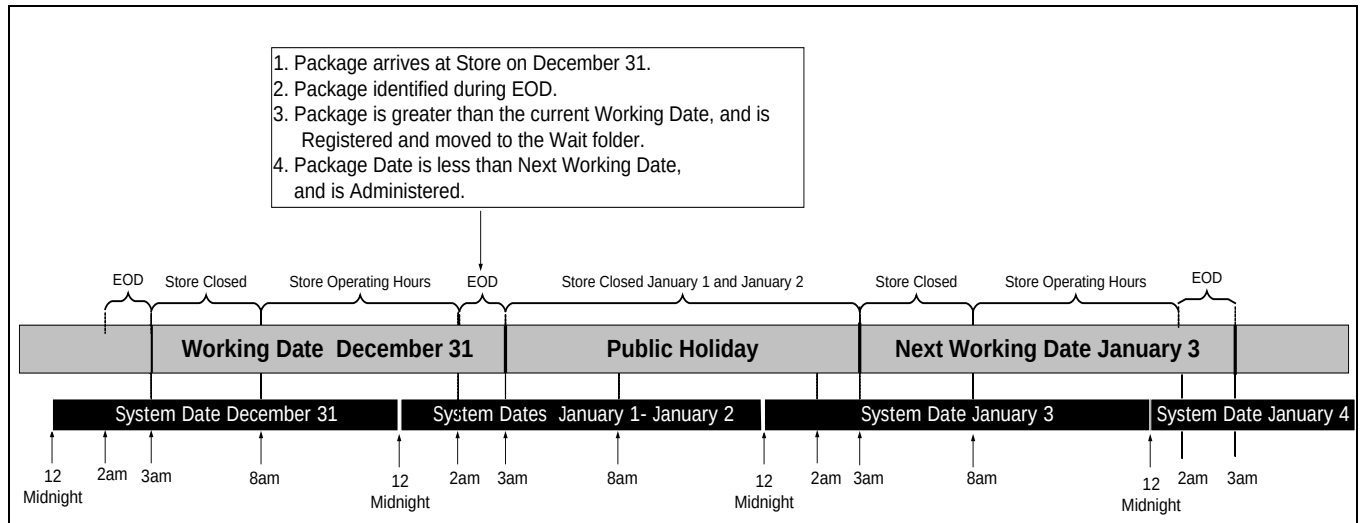
- Package is *Registered* during EOD if the Package Date is **Greater Than or Equal To** System Date.
- Package is *Administered* during EOD if the package Date is **Less Than or Equal To** System Date.



Selecting Next Working Date:

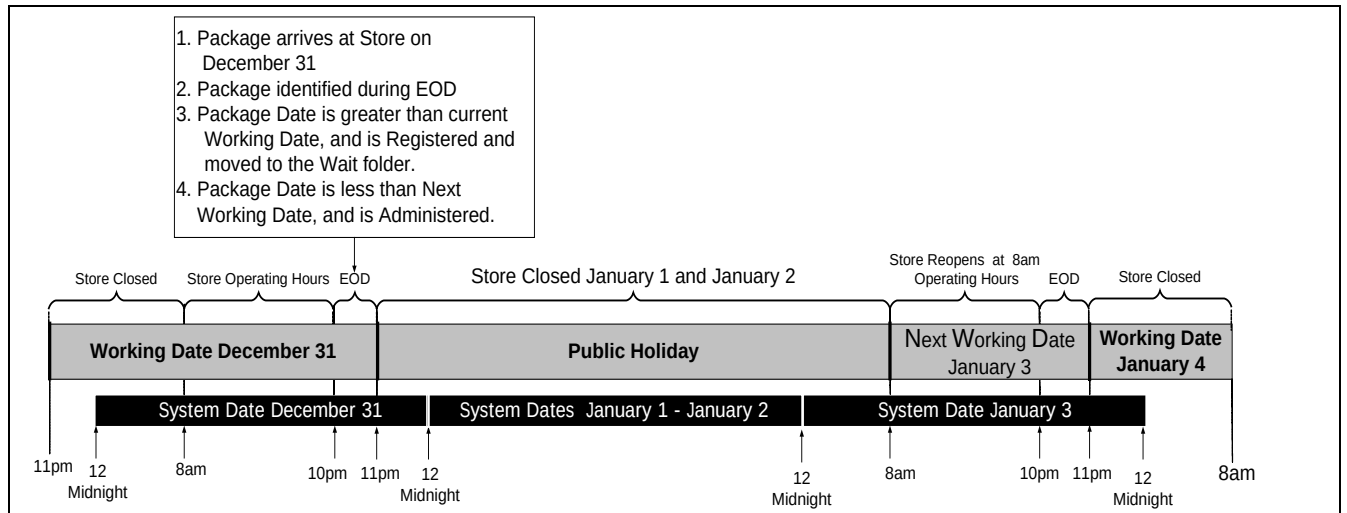
The following illustration refers to a Store that operates from 8am to 2am, and closes for a 2-day holiday. A package upgrade arrives with Package Date of January 1, and will be run before the 2-day holiday starts:

- Package is *Registered* during EOD if the Package Date is **Greater Than** or **Equal To** the current Working Date.
- Package is *Administered* during EOD if the Package Date is **Less Than** Next Working Date.



The following illustration refers to a Store that operates from 8am to 10pm and only runs EOD once a week. A package upgrade arrives with Package Date of January 1, and will be run before the 2-day holiday starts:

- Package is *Registered* during EOD if the Package Date is **Greater Than** or **Equal To** the current Working Date.
- Package is *Administered* during EOD if the Package Date is **Less Than** Next Working Date.



For further details on running a package during EOD, refer to *Registering a Package, Chapter 3*, and *Package Administration, Chapter 4*.

Scheduling Activation Outside of EOD

The following section refers to rules governing activating packages outside of EOD through either through the Package Maintenance application or upon demand. (See *Package Management* in Chapter 3 for more information).

The dates scheduled to *Register* and *Administer* a package must be in accordance with the specifications of the parameters described below; otherwise the package will be rejected, and the system will send an alert indicating a problem.

Note:

The date in the .CTL file is entered as YYYYMMDD, e.g., 20080101. For further details on the .CTL file, refer to *Chapter 7, Package Structure*.

Quick Reference to Scheduling Activation Dates Outside of EOD

The following Quick Reference table specifies the formula used when **CDS** *Registers* or *Administers* a package:

Field	Effect on Package during Registration	Effect on Package during Administration
Trading Date	If Package Date is Greater than or Equal to the Trading Date it will be <i>Registered</i> .	No Validation on Package Date Required.
System Date	If Package Date is Greater than or Equal to the System Date it will be <i>Registered</i> .	No Validation on Package Date Required.
Next Working Date	If Package Date is Greater than or Equal to the current Working Date it will be <i>Registered</i> .	No Validation on Package Date Required.

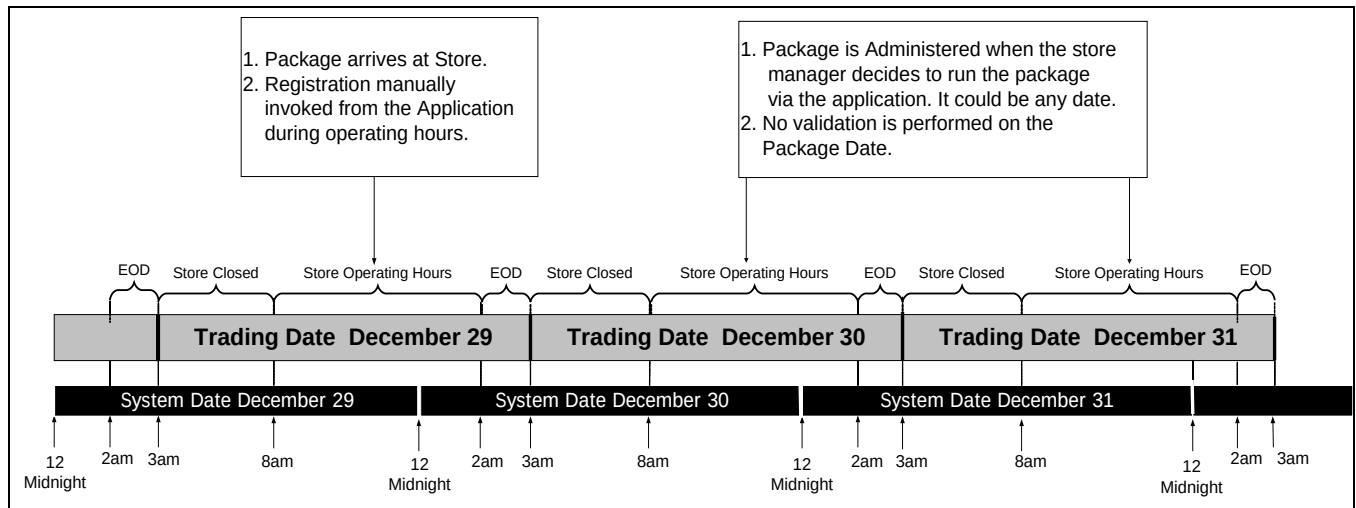
The following 3 illustrations reflect various scenarios when defining the *Activation Date* of a package, specifying when the package would be *Registered* and *Administered*:

- Selecting Trading Date in a Store that operates 8am-2am.
- Selecting System Date in a Store that operates 8am-2am.
- Selecting Next Working Date in a Store that operates 8am-10pm.

Selecting Trading Date:

The following illustration refers to a Store that operates from 8am to 2am and receives a package upgrade with Package Date of December 31:

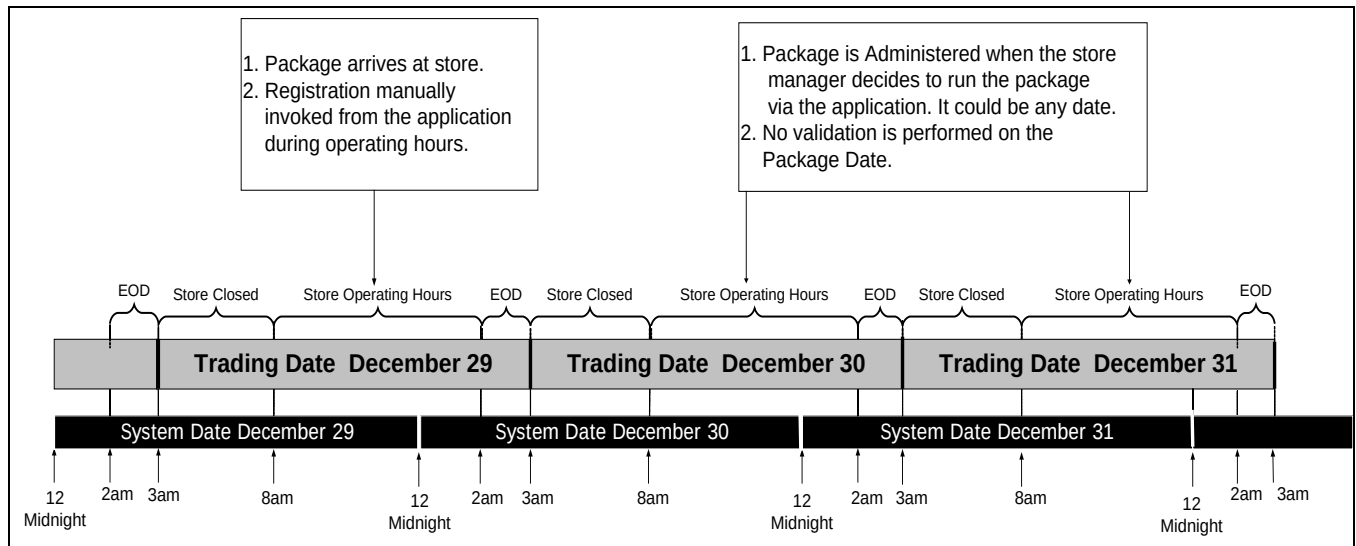
- Package is *Registered* via the application during operating hours if the Package Date is **Greater Than** or **Equal To** Working Date.
- Package *Administered* via the application during operating hours as soon as it is invoked, with no validation of the Package Date, on the day required.
- Since this option is used when CDS is run manually via the application or on demand, there is no difference between a Store that operates from 8am-10pm and a Store that operates from 8am-2am. The package would be *Registered* or *Administered* during the operating hours.



Selecting System Date:

The following illustration refers to a Store that operates from 8am to 2am and receives a package upgrade with Package Date of December 31:

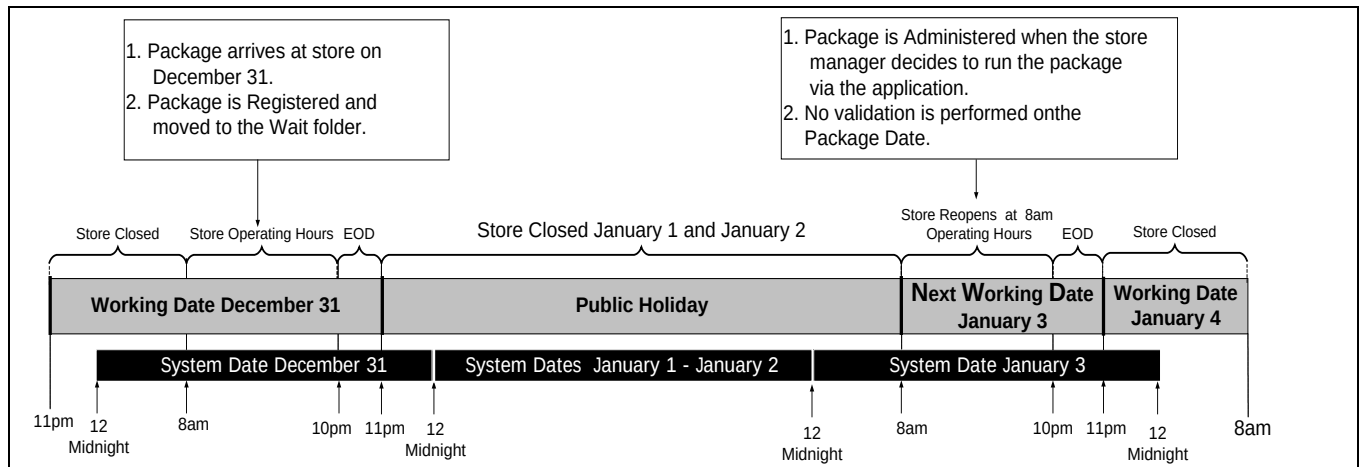
- Package is *Registered* during operating hours if the Package Date is **Greater Than** or **Equal To** System Date.
- Package *Administered* via the application during operating hours as soon as it is invoked, with no validation of the Package Date, on the day required.
- Since this option is used when CDS is run manually via the application or on demand, there is no difference between a Store that operates from 8am-10pm and a Store that operates from 8am-2am. The package would be *Registered* or *Administered* during the operating hours.



Selecting Next Working Date:

The following illustration refers to a Store that operates from 8am to 10pm and receives a package upgrade with Package Date of December 31:

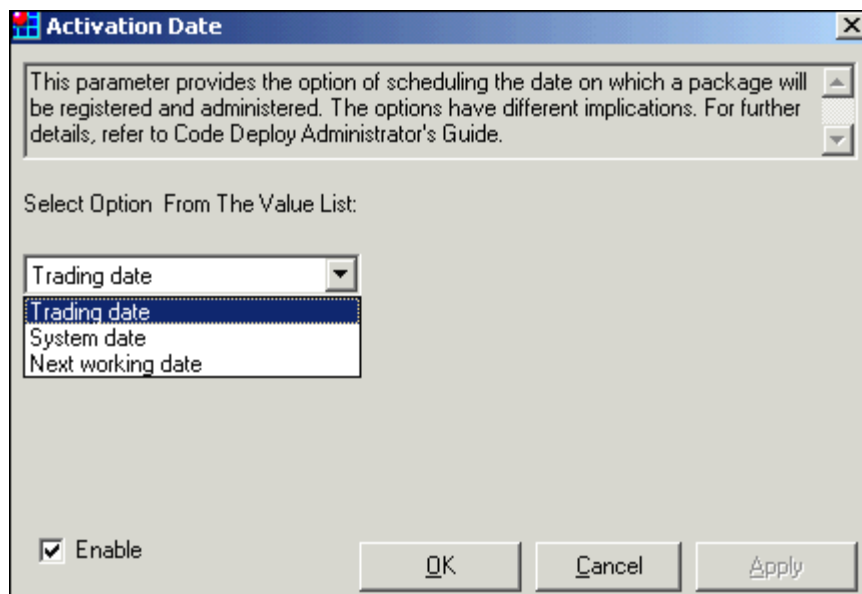
- This option is not relevant when running a package via the application, but rather is intended for packages that are run only during EOD.
- Package is *Registered* via the application during operating hours if the Package Date is **Greater Than** or **Equal To** the current Working Date.
- Package is *Administered* via the application during operating hours on or before the Next Working Date, as decided by the Store Manager. No validation of the Package Date is performed.
- Since this option is used when CDS is run manually via the application or on demand, there is no difference between a Store that operates from 8am-10pm and a Store that operates from 8am-2am. The package would be *Registered* or *Administered* during the operating hours.



For further details on running a package directly from the application refer to *Package Management Utility, Chapter 3*, and *Package Administration, Chapter 4*.

➤ **To Define the v8 Activation Date:**

- 1 In the *Main Menu* tab, browse to the following folder:
System Administration\General System Parameters\General System Parameters.
- 2 Double-click the **General System Parameters** icon. The *System Parameters Maintenance* menu is displayed.
- 3 From the *Store\POST\Operational Menu*, double-click the **Code Distribution & R-Session** icon. The list of dialogs in the Code Distribution and R-Session folder are displayed.
- 4 From the *CDS and R-Session* menu, double-click the **Activation Date** icon. The *Activation Date* dialog is displayed:



- 5 From the *Select Option From The Value List* drop-down box, select the appropriate *Activation Date*. The options include: Trading date, System date, or Next working date.

Notes:

1. The default is Trading Date.
2. For further details, refer to the beginning of this chapter.

- 6 Select the **Enable** checkbox to enable the viewing of this parameter.
- 7 Click *Apply* to save.
- 8 Click *OK* to close the dialog.

Setting Up Disk Space Alerts

In order to perform an upgrade it is necessary to verify that MFS1, MFS2, and the POS terminals have sufficient disk space.

In v8, the utility that performs this check is *PkgAdmin*, which performs an additional check to the v8 Task, which is a mechanism on the MFS that checks the disk space every 10 minutes, (for further details on *PkgAdmin* functionality refer to *Chapter 4*).

To perform the check in v8, it is recommended to set the following two parameters:

1. The parameter by which *PkgAdmin* verifies that MFS1 has sufficient disk space, located in the *General System Parameter Tree* in the following path: Store\Front Office\Minimum Disk Space Drive.
2. The POST parameter is located in the following path: Store\Post\General\Megabytes Remaining on POS Disk to Initiate a Warning Message.

To perform the check in v7, it is recommended to set the following:

1. The General Back Office Parameter “Megabytes Remaining on Disk to Initiate a Warning Message” (6-1-1-1, Line 9). The default is 50 MB.

The disk space required on the v8 MFS1 terminals depends on the size of the SQL Database on the system. For example, a large store that contains a vast amount of items will have a larger database than a small store. It is recommended that if the SQL Database is located on the MFS1 in the C drive, the disk space should be 1000MB. MFS2 and each LFS should have 500MB, and each POS terminal 50MB of disk space. However, these numbers should be determined after running tasks in the specific environment to take into account a Store with a particularly large database.

If insufficient disk space is detected, there are two types of general system alerts that are displayed:

1. When insufficient space is detected in the v8 Front Office, an alert mechanism V8Task (not triggered by *CDS*), #601 is sent, displaying “Disk Space Below Limit”.
2. When insufficient disk space is detected on the POS terminal application, alert #115 will display “POS Disk Space Below Limit”. The POS terminals also perform a check for insufficient disk space independently to the MFSs.

CDS periodically verifies disk space, and if insufficient, the upgrade will not begin.

Setting the System to Auto Log-On

Auto Logon should be set up in the system. Once set, in the event of a power failure, and the power resumes, the system will automatically load the *Service Monitoring* utility (for further details on this utility refer to section on *Upgrading MFS* in *Chapter 4.*), thus enabling the upgrade process to resume. If the Auto Logon is not set, the user will have to logon manually to enable upgrade to resume. Auto Logon is normally setup during Windows Operating System Installation and requires editing the Registry. Please contact your System Administrator for assistance.

3

Package Management Utility

This chapter describes the main functions of RegistrPack, the Package Management utility for CDS.

In this chapter:

- Overview
- Registration
- Validation of .CTL File
- *RegistrPack* Functionality at EOD
- CheckComplete
- HouseKeeping
- Registering
- *RegistrPack* Functionality from the Application

Overview

RegistrPack, CDS's package management utility, functions behind the scenes to automatically provide seamless upgrades for ISS45 v8 software.

Note:

RegistrPack is unique to ISS45 v8 installations only.

Note:

In ISS45 v7, *Pcmpkgnw.exe* is used to validate package(s), read control file(s), verify software key settings, register new packages, and perform housekeeping during EOD using the following commands (found in the Eod-of-day.bat file):

PCMPKGNW /seq2lfs – Copies NextWorkingDate file to LFS to perform date check of package file

PCMPKGNW /registration – Registers the package

PCMPKGNW /e /del – Initiates housekeeping function

PCMPKGNW /initiator – Updates QDX record with all the package/upgrade details

After an upgrade package arrives at a store, the system administrator transfers the package to **New**. *RegistrPack* identifies and handles the new package in one of the following ways:

- **At EOD**

If the upgrades are defined to run at EOD, *RegistrPack* is invoked at EOD. (Refer to *RegistrPack Functionality at EOD*, later in this chapter.)

- **From the application (no dependency on EOD Process)**

If the support person is running the upgrade from the application, the package can be *Registered* immediately, through the Package Maintenance found in the System Administration\Technical Support folder.

Selecting the option *Registration* will start the *RegistrPack* function and the package is *Registered* directly from the application. (Refer to *RegistrPack Functionality from the Application*, further on in this chapter.) This is useful in the following scenario: If the *Registration* of a package failed during EOD, the support person has the option to run the package from the Application, and *Register* the package immediately on site.

- **On demand using a command line:**

If the upgrade is performed on demand using a command line, the command will trigger the *RegistrPack* function to start. (Refer to *Chapter 5, Running CDS on Demand Using a Command Line*.)

The Store Manager determines which way upgrades in the store will be run.

➤ **To Enable v8 RegistrPack.exe:**

- 1 In Windows Explorer, copy and paste the .PKG file to the following folder: **C:\CDPackagesRoot\New**

Notes:

1. **CD Packages Root** folder includes 4 directories: **Done, New, Reject** and **Wait**.
2. The log file **C:\pcmaster\Log\RegPack.Log** records all the activities performed by *RegistrPack*, and can be used for problem analysis. In addition, alerts are recorded at key stages in the process of the *RegistrPack* functionality.

Registration

The *Registration* switch performs the process of checking for new packages in **New** folder, validating new packages, and, if package is valid, transferring the package into **Wait** folder, adding a record into the *SQL* tables with the package data.

Validation of .CTL file

The .CTL file contains the data about the package. The validation process includes the following:

- Verifies that package sensitivity level is indicated (i.e. High or Low).
- Verifies that a scheduled date is indicated in .CTL file, and that it is later than, or equal to the system parameter date. If the date 00000000, it indicates that the **Ignore Date Validation** field is active, and the Registration ignores date validation, enabling the package to run immediately. The date is entered in the following format CCYYMMDD.
- Verifies the Stores number. When 00000 is entered in the Store entry, this specifies that the package is intended for all Stores.
- Identifies one of the three package modes: **New, Edit** or **Delete**. This validation is performed by cross-referencing data with VER_DATA table.

New mode (also called First Arrival) is of a new package that does not exist in the database.

Edit mode is of a package that was previously registered, and, therefore, exists in the database. There are two options for **Edit** mode packages: **Full Edit** in which the F*.EXE and the W*.EXE files have been changed; or – **Partial Edit** in which only the V*.CTL file has been changed, e.g., a changed execution date. If the package was assigned Ignore Date

Validation, and it is then edited, the Ignore Date Validation field is updated according to the .CTL file. If the .CTL file is other than the date 00000000 for the specific store(s), the field Ignore Date Validation flag will be set to off.

Notes:

1. When a **New** or **Full Edit** package is sent, it must contain all three files, i.e., V*.CTL, W*.EXE & F*.EXE; otherwise, the package will be rejected.
2. Fuel packages must contain the following additional files: S*.EXE, P*.EXE. (For further details on Fuel packages and the PFS components, refer to *CDS Support for PFS, Chapter 6.*)
3. The Xs represent the numeric characters in the package name, e.g., 8121050.

Delete mode is for a package which is to be deleted. In this case, a zip file is sent containing a V*.CTL file, in which the date is displayed for example as "20070820" (for August 20th, 2007). Inserting these values will prevent the package from running, and delete all existing records for this package.

Note:

For further details on the contents of each file refer to *Package Structure* in *Chapter 7*.

- ♦ If Registration was successful:

RegistrPack deletes .PKG folder from **New**, copies the unzipped folder to **Wait**.

- ♦ If Registration failed:

RegistrPack copies entire .PKG file to **Reject**, and deletes the unzipped folder.

Registration of a package will fail if any one of these conditions exist:

- If the date in the CTL file is earlier than the last date defined in system parameters.
- If the date section in the CTL file is empty.
- If the number of files in the package correspond with the package.
- If the version number in the CTL file is empty.
- If the version number in the CTL file is mismatched to the file number within the file name.
- If the sensitivity value in the CTL file is empty.
- If the sensitivity value is different from "Low" or from "High" (spelling and capitalization must match).
- If the release version value in the CTL file is empty.

- If the release version value in the CTL file contains anything other than 3 dots.
- If the release version value in the CTL file contains non-numeric characters (exception is three dots).
- If a section between the two dots is longer than five digits (in the release version value).
- If the Store Number section in the CTL file is not equal to the System Parameter STR_NBR (Store Number).
- If the version range value in the CTL file is empty.

After *RegistrPack* has completed the procedures described above, the system is ready to be upgraded. Upgrades are performed by the *PkgAdmin* utility at *EOD*. *PkgAdmin* is responsible for coordinating the synchronization of the upgrade process, and is described in detail in *Chapter 4, Package Administration*.

RegistrPack* Functionality at *EOD

RegistrPack is invoked at *EOD*, and includes three main functions triggered by three switches that run in the following order:

- **CheckComplete**
- **Housekeeping**
- **Registration**

CheckComplete

The **CheckComplete** switch informs the system by flagging it, when upgrade status is **End**. This occurs only when all the PCs in the store have reached **End** status in the upgrade process.

CheckComplete identifies the exact number of active POS terminals, MFSSs and LFSSs in each store, via *PCs In System* function. (For further details refer to section on *Defining CDS Parameters* in *Chapter 2*). If one or more PCs have reached **Error** status, the entire system will report **Error** status.

The list below describes the **CheckComplete** functions:

- Identifies package with status other than **End**.
- Ensures that the upgrade in all PCs (including POSs and MFSSs) in each package version, as indicated in VER_DATA table, has succeeded.
- If upgrade was successful in all PCs, sets the Field System Status as **End**. If failed, sets as **Error**.
- If upgrade was successful in all PCs, checks Sensitivity level in VER_DATA. This is performed by selecting Sensitivity from VER_DATA table where PKG_VER='VXXXXXXX' . If package is *High- Sensitivity* package, deletes flag C:\pcmaster\Flag\VXXXXXXX.END, if exists.

Note:

For further information on the flag VXXXXXXX.END refer to section on *Version Upgrading Process in Low-Sensitivity Packages*, in *Chapter 4*.

Housekeeping

The **Housekeeping** switch in *RegisterPack* transfers all packages that have completed the upgrade process in the store, from **Wait** folder to **Done** folder.

After a predefined period has elapsed, **Housekeeping** then deletes the package from **Done** and **Reject** folders. (For further information refer to *Defining CDS Parameters* in *Chapter 2*).

The list below describes the Housekeeping functions:

- Deletes package history using the parameter *Number of Months to Keep History Packages*.

Note:

The *Number of months to keep history package parameter* is setting is established in **Code distribution Management Parameters** (under the System Administration folder's Technical Support folder).

- **Done** folder:
 1. Verifies the date on which the upgrade was performed, by comparing the Execution date (UPD_DT) parameter in VER_DATA to the System date.
 2. If there is no Execution date (UPD_DT) parameter in the VER_DATA, it will compare the file date to the system date.

- **Reject** folder: Compares the file date (of the last update) to the system date.
- Moves required packages in zip file from **Wait** to **Done**, if System Status (SYS_STATUS) in VER_DATA is marked as **Complete (=End)**.

Notes:

1. Deletion can be performed from *CDS* folders, only in MFS1, and only from **Done** and **Reject** folders.
2. There is an additional **Housekeeping** function performed in MFS2 and in LFS, not by *RegistrPack*, but by *PkgAdmin*. This function moves the latest upgrade package that was installed, from **Wait** to **Done**, or from **Wait** to **Reject**.

Registering

The following is a description of the *Registration* procedures:

- ♦ Validates Package Contents:
 - In **Wait** folder creates new folder for each new package, with the name of the package.
 - Extracts the three package files to **Wait** folder, in which the .PKG file is located.
 - Verifies that there are 3 files in the package: The FXXXXXXX.EXE, the WXXXXXXX.EXE file, and the VXXXXXXX.CTL file.

- ♦ Validates .CTL File.

See section on *Validation of .CTL File* in the *Registration* section, above.

RegistrPack Functionality from the Application

If you are running *CDS* from the Application, *RegistrPack* is invoked manually, and is not dependent on *EOD*. The Store Manager determines when it is necessary to run the package from the Application.

Note:

You have the option to start *RegistrPack* manually using a command line from a remote location. For further details on running *CDS* on demand using a command line, refer to *Chapter 5, Running CDS On Demand*.

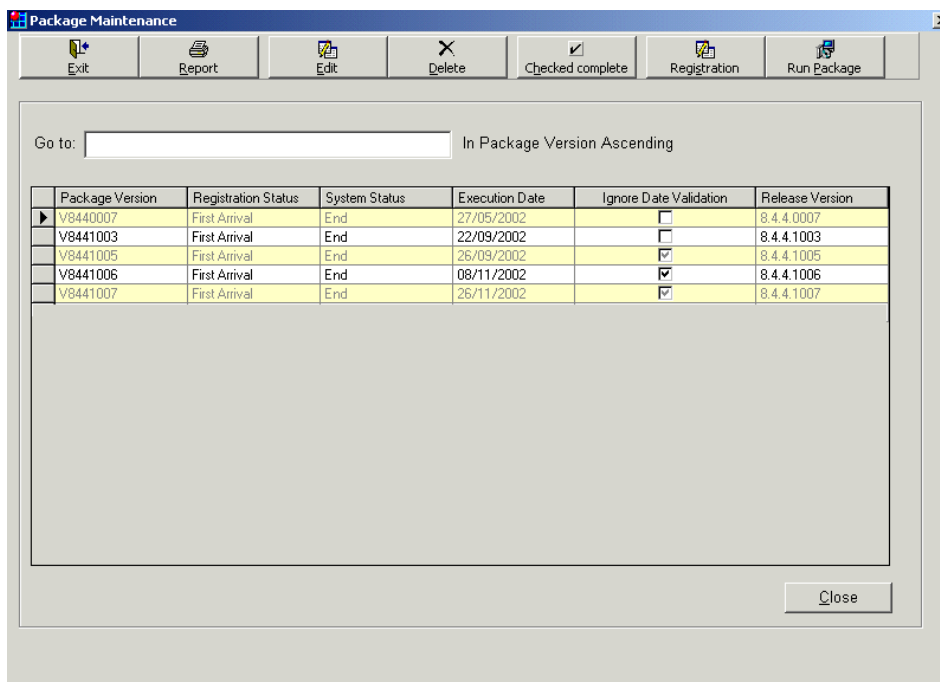
➤ **To Register a v8 Package from the Application:**

- 1 In the **Main Menu** tab, browse to the following folder: **System Administration\Technical Support\Package Maintenance**. The following dialog is displayed:



- 2 Enter Administrator's password, and click **OK**.

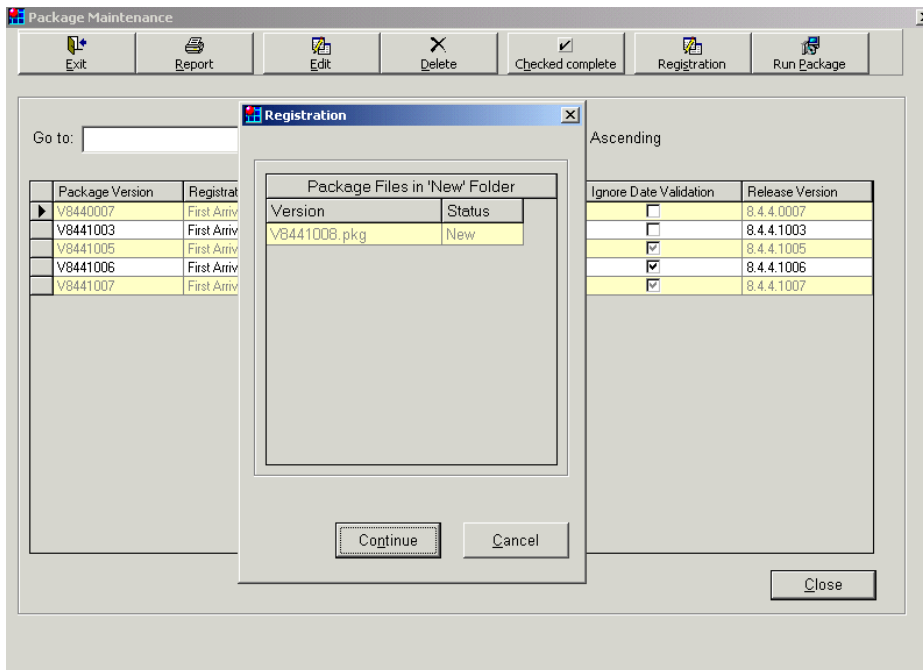
The main *Package Maintenance* dialog is displayed:



Note:

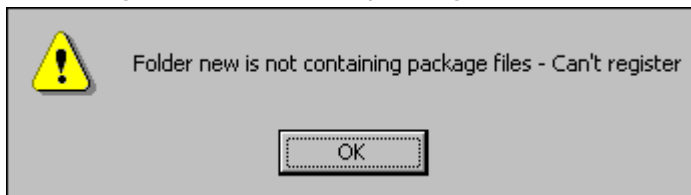
One of the files in the **New** package is the CTL file. The package is registered according to the date in the .CTL. If a value in the CTL file is "00000000", the *Registration* ignores the date validation. The **Ignore Date Validation** field in the *Package Maintenance* dialog will be automatically checked. (For further information on the CTL files see the section *Registration - Validation of CTL file*, below.)

- 3 From the *Package Maintenance* dialog, click *Registration* to invoke *RegistrPack*. If there is a package in the **New** folder, the *Registration* dialog is displayed as follows:



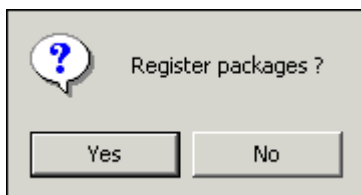
Or

If there was no package in the **New** folder, the following message is displayed indicating there is no **New** package:

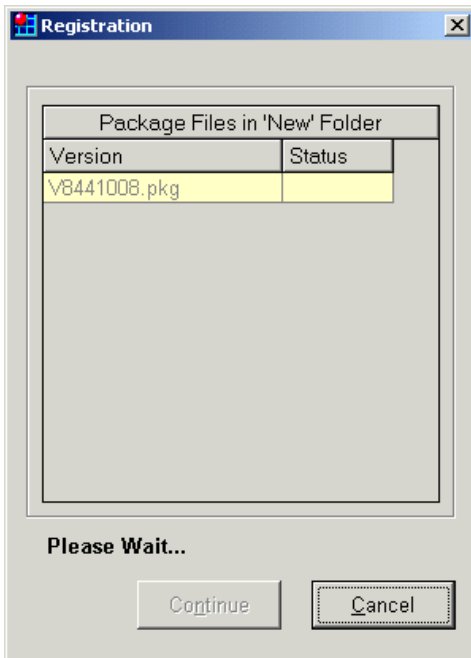


Click *OK* to close the dialog.

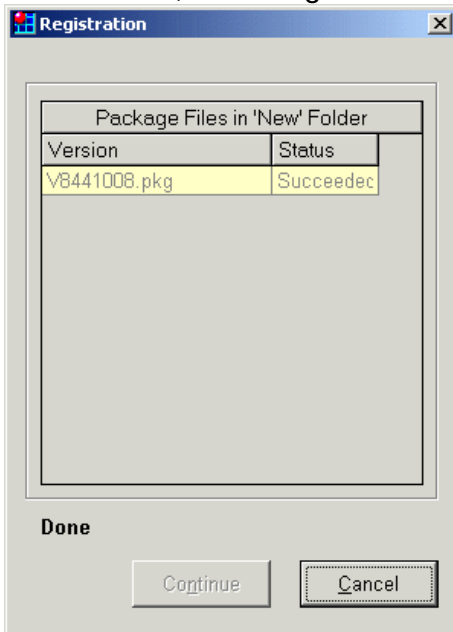
- 4 From the *Registration* dialog, click *Continue* to register the new package. The following message is displayed:



- 5 Click Yes to continue. The following dialog is displayed:



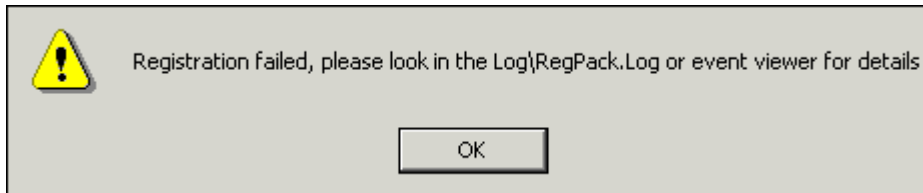
- 6 If the *Registration* succeeded, the following dialog is displayed within a few seconds, indicating a status of **Succeeded**:



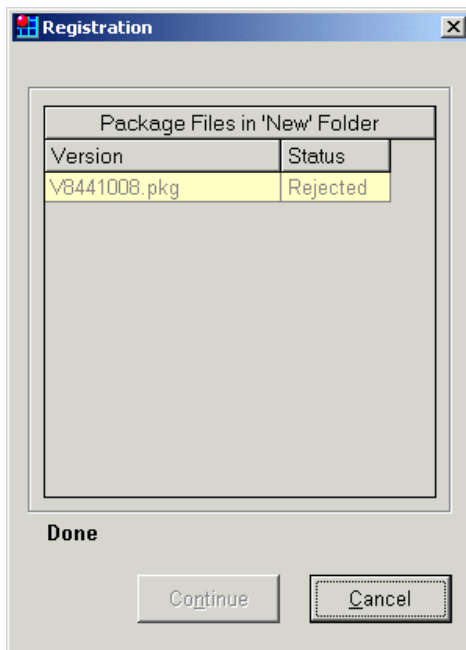
Proceed to Step #9.

Or:

If the *Registration* fails, the following message is displayed:|



- 7 Click *OK*. The package is rejected and the *Registration* dialog is displayed as follows:

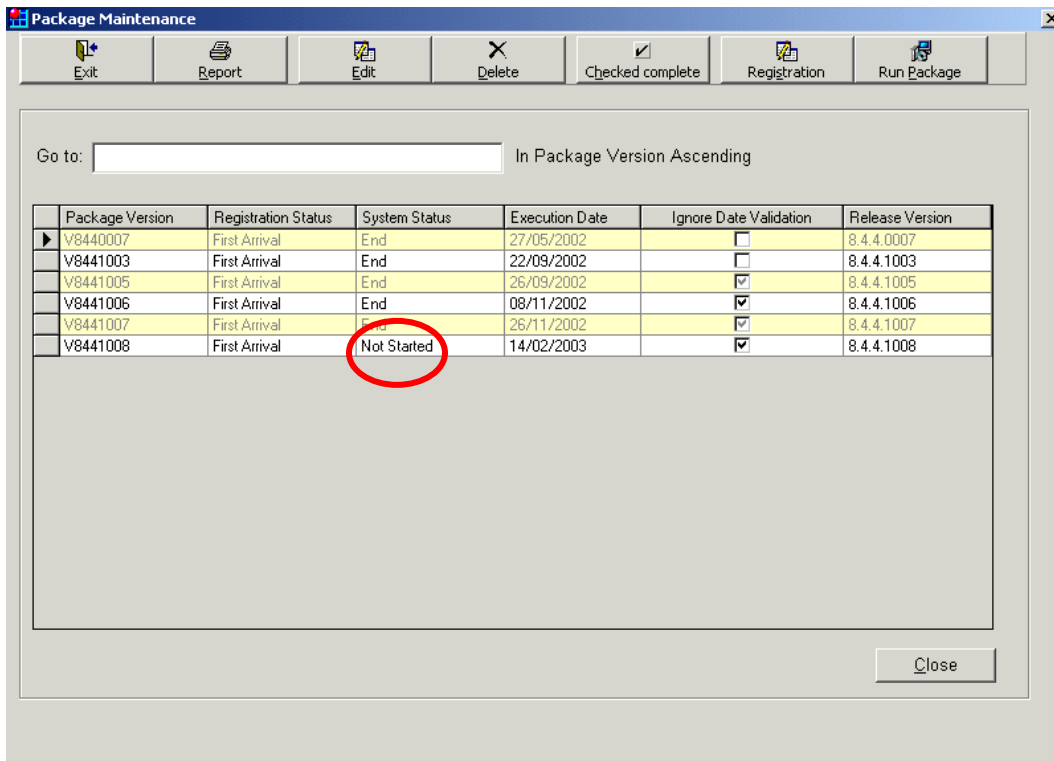


- 8 Refer to the Log to identify the problem.

Note:

There may be several different reasons for a package to fail *Registration*. For further details refer to section on *Registration/Validation of .CTL file* at the beginning of this chapter.

- 9 Once the package has been *Registered*, it is moved to the **Wait** folder, and is displayed in the *Package Maintenance* dialog with the status of **Not Started**, as shown below:

**Note:**

The **Ignore Date Validation** checkbox is automatically checked, as the .CTL file was defined as 00000000, meaning no date validation is necessary for this package.

The package is *Registered* and ready to be run. (For further details, refer to *Package Administration, Activating an Upgrade from the Application*, Chapter 4.)

4

Package Administration

This chapter describes the main functions of PkgAdmin utility and the upgrade processes that are performed on the MFS1 server, the MFS2 workstation, the LFS and the POS terminals.

In this chapter:

- Overview
- Low-Sensitivity and High Sensitivity Packages
- Version Upgrading Process in Low Sensitivity Packages
- Running the Package During EOD
- Running the Package from the Application
- Opener Utility
- Upgrading MFS1
- Upgrading MFS2 & LFSs
- Upgrade Failure in MFSs
- Upgrading the POS Terminals
- Activating and Monitoring Upgrades via Package Maintenance

Overview

PkgAdmin, a utility that is an integral part of the *CDS* process, serves two important functions – to detect new packages that have been *Registered* by *RegistrPack* and to initiate upgrades.

Like *RegistrPack*, *PkgAdmin* can also run during EOD, via the application or by entering a command line. (Refer to the *Overview*, in *Chapter 3, Package Management Utility*.)

Note:

PkgAdmin is unique to ISS45 v8 installations only.

Low-Sensitivity & High-Sensitivity Packages

There are 2 types of upgrade packages: *Low-Sensitivity* and *High-Sensitivity*.

Low-Sensitivity Packages: This is the more common type of package. *Low-Sensitivity* packages may include changes in DLLs, EXEs and other changes in database, excluding changes in *QDX* or *RsmSrv*.

The Low Sensitivity Upgrade Process:

- 1 MFS1 downloads the package first.
- 2 MFS1 sends an **End** status alert.
- 3 The MFS2, LFS & POS terminals run the package.

High-Sensitivity Packages: This type of package includes changes such as *QDX* entry length, or changes in *RsmSrv* functionality. In *High-Sensitivity* packages, all MFSs run the package upgrade simultaneously, which means that they must begin, boot and end the upgrade process at the same time. There are several parameters relating to the beginning, booting and ending of the upgrade process of MFSs in *High-Sensitivity* packages, (for further details on the following parameters refer to *Chapter 2, CDS Setup*, section on *Defining CDS and R-Session Parameters*.)

VERSION UPGRADING PROCESS

- Max No. of Minutes to Wait for MFSSs Synchronization Before Starting Upgrade.
- Max No. of Minutes to Wait for MFSSs Synchronization Before Starting Boot.
- Max No. of Minutes to Wait for MFSSs Synchronization Before Ending Upgrade.

The High Sensitivity Upgrade Process:

- 1 When a POS terminal receives an alert regarding the arrival of a *High-Sensitivity* package, it immediately goes into *Logic Offline* mode (which means that it is not actually disconnected from the network).
- 2 The POS terminal checks, at frequent intervals, if the following flag was raised: **C:\pcmaster\Flag\VXXXXXXXX.END**. This flag is raised in MFS1 indicating a successful upgrade in MFS1.
- 3 If the flag is detected the POS terminal begins the upgrade process and then reverts to *Online* mode.
- 4 The above flag is deleted once the *CheckComplete* process is done, and once all POS terminals have completed the upgrade.
- 5 In the event that an upgrade fails in any one of the MFSSs, in order to bring them back to *Online* mode, the support personnel must first set three flags in the POS terminal:

C:\program files\winpos\abort_cd.flg

C:\program files\winpos\reset_cd.flg

C:\program files\winpos\online.flg

- 6 Once the POS terminals go into *Online* mode, the flags **reset_cd.flg** and **online.flg** are deleted automatically; the **abort_cd.flg** flag should then be deleted manually by support personnel.

Warning:

1. Check the **reset_cd.flg** and **online.flg** were automatically deleted once the POS terminals go into *Online* mode after an upgrade failure in one of the MFSSs, otherwise it should be deleted manually by support personnel.
 2. Remember to delete the **abort_cd.flg**, otherwise CDS will not perform the upgrade, once the POS terminals go back into *Online* mode.
- 7 If upgrade is unsuccessful, the package is rejected, and none of the POSs will upgrade.

Notes:

1. In *High-Sensitivity* packages POS terminals remain in *Offline* mode while server is upgrading, and will only revert back to *Online* mode when upgrade is complete and successful.
2. Failure of *High-Sensitivity* packages must be attended to urgently, as POS terminals will remain in *Offline* mode until the error is recovered.

Version Upgrading Process in Low Sensitivity Packages

Upgrades are performed in the following order:

- 1 MFS1 server upgrades.
- 2 MFS1 changes status in VER_STATUS table to **End** status, and sends upgrade maintenances to POS terminals, instructing them to begin the upgrade.
- 3 MFS2 workstation, additional LFS PCs, and all POS terminals may now begin the upgrade together, but they do not have to upgrade at the exact same time.

Notes:

The timing and number of POS terminals to be upgraded is determined by the CDS parameters (Refer to *Chapter 2*, section on *Defining CDS Parameters*.)

Running the Package During EOD

The *EOD* process runs once a day to summarize transactions for the store. *PkgAdmin* is an extension of the *EOD* process and runs within it synchronically.

Process Performed by PkgAdmin:

- 1 *PkgAdmin* checks the SQL tables for new packages.
- 2 When a new package is detected, *PkgAdmin* raises a flag notifying *EOD* to run *PkgAdmin* again when the *EOD* process is complete. This flag can be viewed in the following folder:
C:\pcmaster\Flag\Pksyneod.FLG
- 3 Then, *EOD* continues summarizing store transactions.
- 4 After *EOD* is completed, it prompts *PkgAdmin*, which then verifies that the MFS1 terminal has sufficient disk space. (For further details refer to *Chapter 2*, section on *Setting Up Disk Space Alerts*.)
- 5 *PkgAdmin*, in turn, initiates the *Opener* utility (see the section on the *Opener Utility*, below), which is responsible for the actual upgrade. This is done via the flag created by *PkgAdmin*, called
C:\Pcmaster\Flag\Pkgstart.FLG.

The *Opener* opens the package and performs the upgrade.

Running the Package from the Application

When *CDS* is run through the Application, *PkgAdmin* is triggered manually, independent of the *EOD* process. (For further details on the different *CDS* options, refer to the *Introduction*, in *Chapter 1*.)

Process Performed by PkgAdmin:

- 1 A package is first registered as described in *Chapter 3*, *Package Management Utility*.
- 2 Once the *Registration* process is completed, by selecting the option **Run Package** from the *Package Maintenance* screen in ISS45 Office, in the following folder: *System Administration\Technical Support\Package*

Maintenance, the support personnel is able to manually invoke *PkgAdmin* and run the package immediately.

- 3 *PkgAdmin* checks the SQL tables for new packages, executing the package immediately, without performing any validation of the package date.
- 4 *PkgAdmin* then verifies that the MFS1 terminal has sufficient disk space. (For further details refer to Chapter 2, section on *Setting Up Disk Space Alerts*.)
- 5 *PkgAdmin*, in turn, initiates the *Opener* utility (see the section on the *Opener Utility*, below). This is done via a flag created by *PkgAdmin*, called **C:\Pcmaster\Flag\Pkgstart.FLG**. The *Opener* opens the package and performs the upgrade.

(For further information on running a package manually, refer to *Activating and Monitoring Upgrades via Package Maintenance*, in this chapter.)

Opener Utility

The *Opener* utility opens new packages and performs upgrades when invoked by *PkgAdmin*.

Process performed by the Opener:

- 1 When the *Opener* begins the upgrade on the SQL database, it raises a flag in: **C:\Pcmaster\Flag\PkgUPSQL.FLG**.
- 2 The *Opener* then reports whether the upgrade was successful or not, via a status message.
- 3 The status message is recorded in the VER_STATUS SQL table in the FO database, which can be viewed in the *Package Maintenance* form. (For further information refer to *Activating and Monitoring Upgrades via Package Maintenance*, in this chapter).
- 4 The *Opener* can report one of the following statuses and *PkgAdmin* acts accordingly:
 - **1:** The package upgrade was successful, and there is no need to reboot. (MFS Default Boot)
 - **2:** The package upgrade was successful and the computer must be rebooted.
 - **3:** An error was encountered when opening the package. *PkgAdmin* will treat it as error No. 260. (For further details, refer to section *Upgrade Failure in MFSSs*, later in this chapter.)

- 4: The *Opener* did not transmit a response.
- 5 When the *Opener* completes the upgrade, one minute before booting the system, *PKGAdmin* creates a flag:
C:\Pcmaster\Flag\Pkg_Boot.FLG

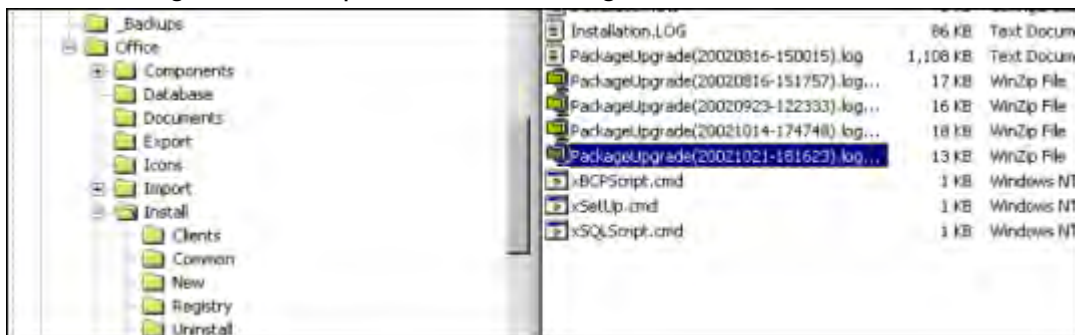
Viewing Opener Log Files for MFS Upgrades

Once the *Opener* has completed performing the upgrade, it creates either a Success log file or an Error log file, depending on the status of the upgrade.

If upgrade succeeded, a log file is created in the following path:

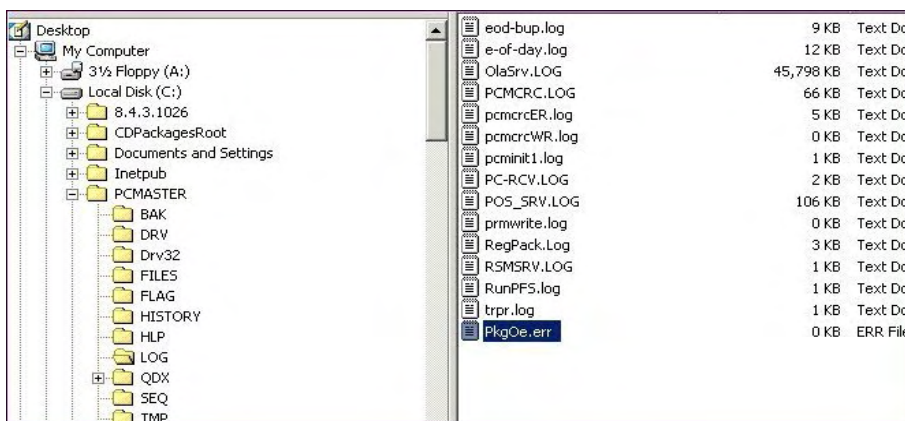
C:\Program Files\POSWare\Office\Install

The following is an example of a Success log file:

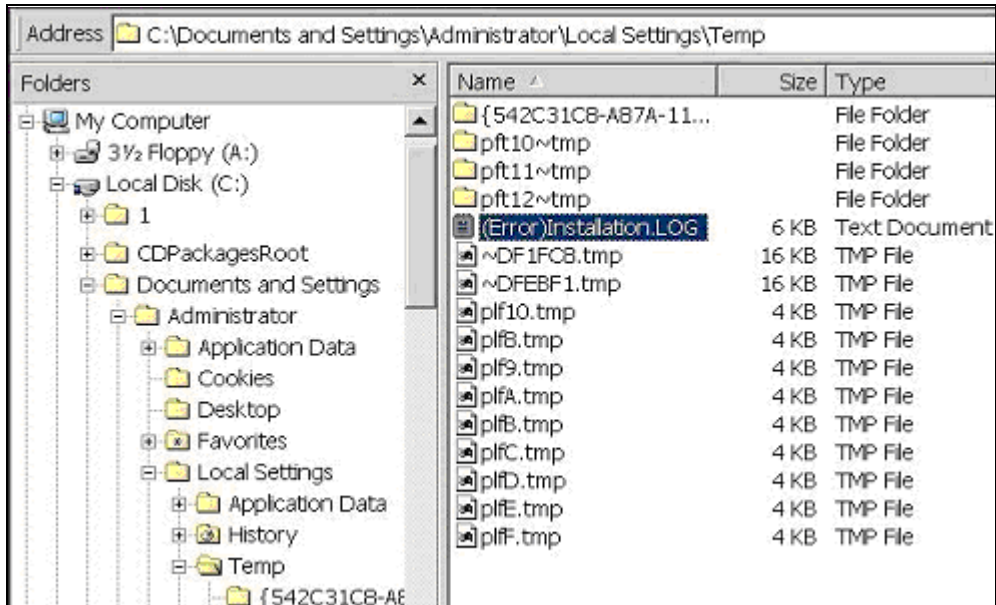


The log file includes the date and time (year, month, day, hour, minutes and seconds) of the upgrade.

If upgrade failed, a file is created in the following path: **C:\Pcmaster\Log**



The file PkgOe.err is a copy of the following Error log file:



The Error log file created in the Temp directory can be located according to the settings defined in the *Environment Variables Setup* in *System Properties*. The example above displays the Temp file created in the following path: **C:\Documents and Settings\Administrator\Local Settings\Temp**.

Viewing Opener Logs for POS Terminal Upgrades

If upgrade succeeded on the POS terminal, a log file is created in the following path:

C:\Program Files\POSware\WinPOS\Install

To view a similar example of a log file, refer to the section *Opener Log Files for MFS Upgrades*, above.

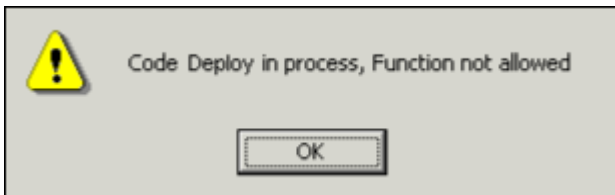
If an upgrade fails on a POS terminal, the following Error log file is created: C:\Program Files\POSware\WinPOS\Log\PkgOe.err.

PkgOe.err is a copy of the POS Error log found in the Temp directory, as explained in the section above, *Opener Log Files for MFS Upgrades*.

Upgrading MFS1

Upgrading MFS1 occurs as follows: *PkgAdmin* checks for new packages either once a day, every *EOD*; or when selecting **Run Package** from the *Package Maintenance* screen from *ISS45 Office Main Menu\System Administration\Technical Support\Package Maintenance*. The *Opener* opens the package and performs the upgrade automatically on the MFS1 server. After a successful upgrade, the *VER_STATUS* SQL table is updated.

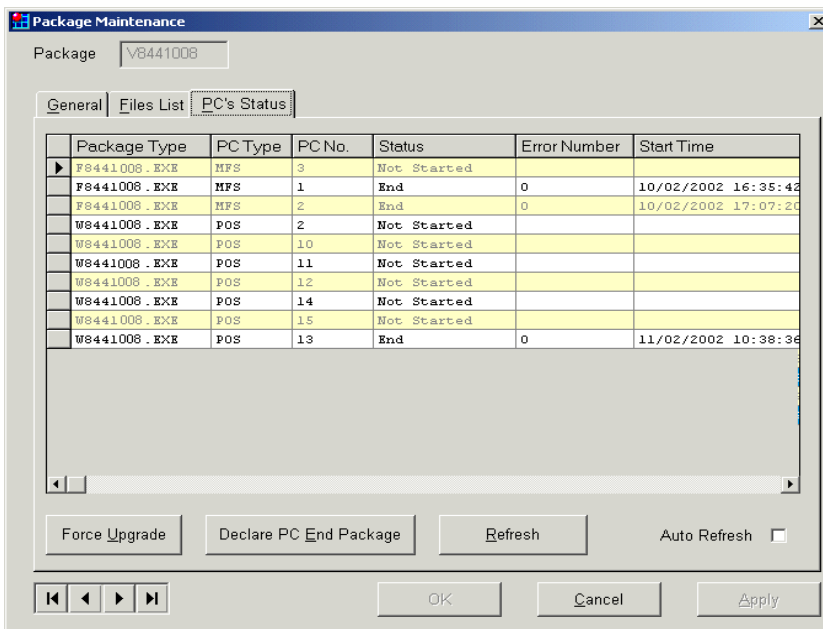
Once the Upgrading process begins, you cannot access the *ISS45 Office* application menus. If you try to access the menus, the following *ISS45 Office* dialog will be displayed:



Note:

In *Low-Sensitivity* packages the MFS1 server always upgrades before the MFS2 workstation. For further details on Upgrading of *Low-Sensitivity* packages, refer to *Version Upgrading Process in Low Sensitivity Packages*, earlier in this chapter.

Additionally, the following v8 dialog is displayed automatically, after MFS1 reboots:



The table below describes the columns in this dialog:

Columns	Description
Package Type	Version number of package
PC Type	Indicates whether Package is for MFS or POS terminal.
PC No.	Indicates PC number
Status	Status of upgrade process in the system, e.g.: End: Indicates the specified PC or MFS in the store has been upgraded. (End status will be transmitted by CheckComplete only at the next <i>EOD</i>, or via the system by an Alert which updates the tables VER_STATUS in the FO database. For further details refer to section on <i>RegisterPack</i> in <i>Chapter 3</i>.) Start: Indicates that the PC or MFS is still Upgrading. Not Started: Indicates that the PC or MFS has not started the upgrade.
Error Number	Code of Error. An error has occurred on one of the PCs; this will display Error status for the entire system.
Start Time	Start Time and Date of upgrade.
End Time	End Time and Date of upgrade.
Prev. Release	Number of previous version released.
Action Time	This field is for the developers and Internal Use only.
Stage	This field is for the developers and Internal Use only.
Syn Status	This field is for the developers and Internal Use only.

Upgrading MFS2 & LFSs

The upgrade process in MFS2 workstation is different than MFS1. MFS2 contains an application called the **POSSVMON.exe**, or *Service Monitoring*.

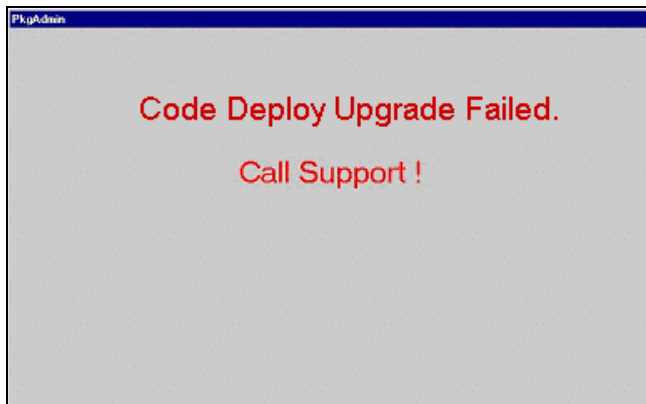
This application is symbolized by an icon of an eye , displayed in the status bar. Every five minutes *Service Monitoring* checks the VER_STATUS table in the MFS1 database for a new package, as well as for a message that MFS1 was upgraded successfully. When an **End/Success** message appears in the VER_STATUS table for MFS1, *Service Monitoring* will copy the F*EXE file from the MFS1 **Wait** folder to MFS2 **Wait** folder, and then *PkgAdmin* invokes it to begin upgrading MFS2, (as is done in MFS1). When upgrade is completed, transfers F*.EXE file from **Wait** to **Done**.

When MFS2 has upgraded successfully, the VER_STATUS table for MFS2 is updated.

The same process occurs in LFS.

Upgrade Failure in v8 MFSs

If the *Opener* fails to upgrade in one of the MFSs, *PkgAdmin* treats the failure as error No. 260 and the following v8 message is displayed:



Warning:

To Clear the CDS Upgrade Failed Message, or the CDS in Process, Function not Allowed Message- contact technical support personnel!

➤ **To Clear the CDS Upgrade Failed Message:**

- 1 From the Windows **Main Menu** select **Start**, then **Run**, and enter **CMD**.
- 2 From the Command Window, enter CD\PCMASTER, and press *Enter*.
- 3 Enter the command: **PCMUTL32 /EXITPKGFAIL**
- 4 Type in **Exit** and press **Enter**. *The command window is closed.*

Or

- 1 Create a manual flag:

C:\PCMASTER\Flag\ExitPkgF.Flq

If you tried to access the *ISS45 Office* application menus once the CDS process started, the *ISS45 Office* dialog "CDS in Process, Function Not Allowed" is displayed. This dialog may need to be cleared after an upgrade failure to be able to access the *ISS45 Office* application menus. (For further details refer to the section above, *Upgrading MFS1*.)

➤ **To Clear the v8 CDS in Process, Function Not Allowed:**

- 1 From the Windows **Main Menu** select **Start**, then **Run**, and enter **CMD**.
- 2 From the Command Window, enter CD\PCMASTER, and press *Enter*.
- 3 Enter the command: **C:\PcMaster\PkgAdmin /SignCdsOff**.
- 4 Type in **Exit** and press **Enter**. *The command window is closed.*

You can then access the *ISS45 Office* application menus.

- An error that occurred in one of the MFSs, can be identified by browsing to the following *PkgAdmin* log file:

C:\pcmaster\Log\Pos_Srv.Log

- If the error occurred in the *Opener*, browse to the following log file:

C:\pcmaster\Log\PkgOe.Err (For further details on PkgOe.Err, refer to *Opener Log files for MFS Upgrades*.)

Upgrade Failure in v7 MFSs

Perform the following steps to perform CDS MFS Recovery for v7:

- 1 Empty QDX number 83 ("q-empty 83") on MFS1 and MFS2 from a command prompt.
- 2 Delete any of these files (if exist):

```
c:\PCMASTER\FLAG\Pkg_strt.flg  
c:\PCMASTER\FLAG\Pkgs_Run.flg  
c:\PCMASTER\FLAG\Pkgs_Due.flg  
c:\PCMASTER\FLAG\MIRROR.BAD  
c:\PCMASTER\FLAG\Pkgs_Due.BAD  
c:\PCMASTER\FLAG\PKG_BOOT.FLG  
c:\PCMASTER\FLAG\INITATOR.FLG  
c:\PCMASTER\FLAG\OFFLINE.FLG  
c:\PCMASTER\FLAG\BEFORE1.BOT  
c:\PCMASTER\FLAG\BEFORE2.BOT  
c:\PCMASTER\FLAG\AFTER1.BOT  
c:\PCMASTER\FLAG\AFTER2.BOT  
c:\PCMASTER\FLAG\NO_BATCH.FLG  
c:\PCMASTER\FLAG\ABORT.LFS  
c:\PCMASTER\FLAG\REGISTER.FLG
```

Upgrading the POS Terminals

In each store, in each checkout bank, the upgrade occurs according to the parameters specified in *Chapter 2, CDS Setup*. This means that POS terminals receive Maintenance messages from the MFS1 server, enabling up to 30 POS terminals to be upgraded simultaneously.

After the MFS1 server has been upgraded, the POS terminals will be upgraded. Each POS terminal obtains an Rsession from MFS according to the settings defined in *Chapter 2*. If an Rsession was successfully obtained, the POS terminal will begin the upgrade process as described below, otherwise it will continue to retry obtaining an Rsession at pre-defined intervals.

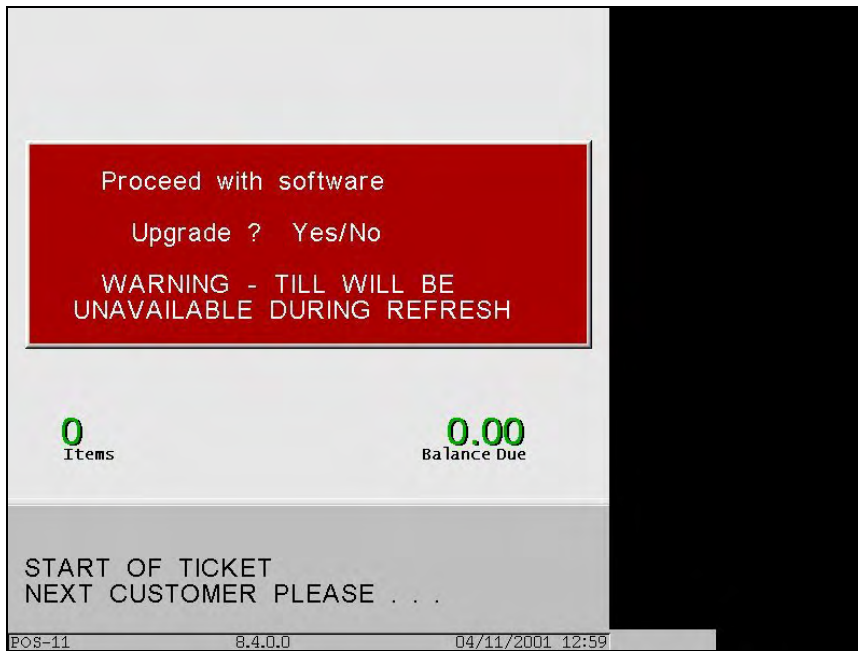
Upgrades are performed automatically when POS terminals are in *Sign-Off* mode.

➤ **To upgrade a POS Terminal when in Sign-On Mode:**

Note:

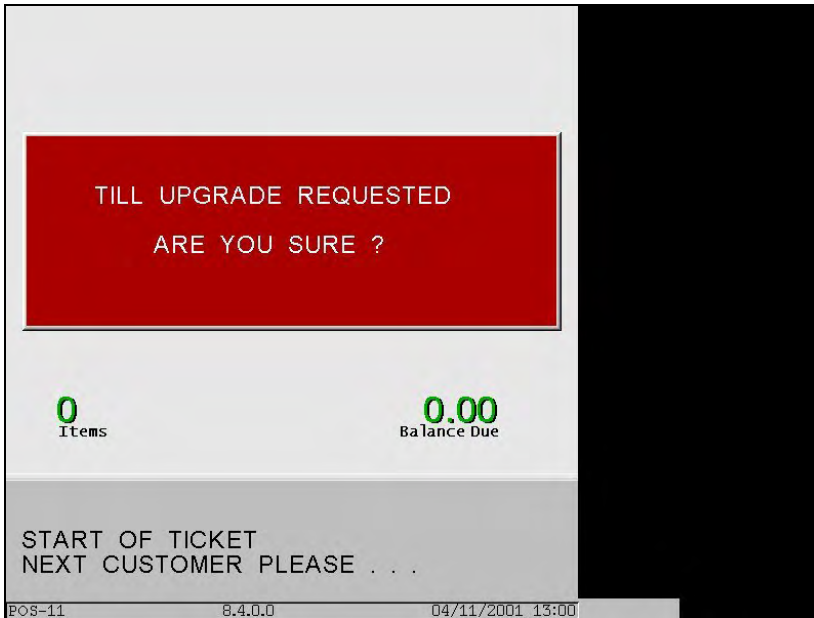
The following screens will also appear for ISS45 v7 POS upgrades.

- 1 In *Sign-On* mode, the following on-screen message will be displayed:



This message prompts the cashier to proceed with the upgrade, or postpone it. If a POS terminal is in the middle of a transaction, this message will be displayed only after the transaction is completed.

- 2 The cashier must respond to the on-screen message within a specified amount of time, otherwise the upgrade will begin. When *Yes* is selected, a second message is displayed, as follows:



- 3 When selecting *Yes* the POS terminal will go into *Sign-Off* mode, and the upgrade will begin.

Or

If cashier does not respond within the number of seconds defined in the relevant parameter (Refer to *Defining CDS Parameters* section in *Chapter 2*), system will *Sign-off*, and upgrade will begin.

When a v8 upgrade begins, the POS terminal performs *Rcopy* of the WXXXXXXX.EXE file to **C:\CDPackagesRoot\Wait**

Or

When selecting *No* the Pos terminal stays in *Sign-On* mode. The system retries to start *CDS* according to the parameters defined in *Defining CDS Parameters* section, in *Chapter 2*, and in the end the POS terminal will free its *Rsession*, enabling other POS terminals to upgrade.

- 4 Upon successful completion of the upgrade, a status alert is recorded. MFS1 then adjusts the status in VER_STATUS table in the FO database. (For further details on this process refer to *Chapter 8, Alerts*).

RSession is freed after the POS application successfully starts following the upgrade. If it is not successful, *Rsession* is not freed.

Or

In the event of upgrade failure, refer to section below, *Aborting an Upgrade on POS Terminals*.

Notes:

1. If cashier selected *No* in both messages shown above, the POS terminal will free its Rsession, enabling other POS terminals to upgrade.
2. In upgrades for Fuel systems the PFS package is located in the same folder of the WinPOS package. The names of the PFS package are:
PumpSrv – SXXXXXXX.EXE.
POSPump – PXXXXXXX.EXE.
3. For more details on Fuel packages and the PFS components during Upgrading of the PFS POS Terminals see *CDS Support for PFS, Chapter 6*, in this manual.

Viewing Upgrade Statuses in POS Terminals

The POS terminals will usually transmit alerts to the MFSs indicating upgrade status. However, in the event of malfunction in a POS terminal, the upgrade status can be viewed by browsing to the following log file:

Programs Files\Posware\WinPos\Log\WinPos.Log

Aborting an Upgrade on POS Terminals

This option describes how to abort upgrades on the POS terminals. In a Store, Rsession is defined to enable the POS terminals to complete the upgrade process without overloading MFS1. For example, a store that contains 50 POS terminals can be defined to upgrade 6 terminals at a time, by setting the *Max No. of Tills to Upgrade Simultaneously* parameter to 6. (Refer to *Chapter 2* for more details on setting CDS parameters.) If all 6 POS terminals fail to upgrade, it indicates that Rsession was not released, meaning that the POS terminals failed to upgrade and stopping the next 6 from starting an upgrade. If the services were stopped and then restarted, this would release the session, but the next 6 POS terminals are likely to fail as well.

To release an Rsession, the technician can use the function *Abort Upgrade*. Using this function a message can be sent to abort an upgrade on all POS terminals or a specific POS terminal.

➤ **To Abort a v8 Upgrade on All POS Terminals:**

- 1 From the Windows **Main Menu** select **Start**, then **Run**, and enter **CMD**.
- 2 From the Command Window, enter CD\PCMASTER, and press *Enter*.

- 3 Enter the command: `PkgAdmin /SendCdsMaint /AbortUpg /VerName=Vxxxxxxx /PkgName=Wxxxxxxx.EXE /RelVer=x.x.x.xxxx`

Or

```
PkgAdmin /SendCdsMaint /AbortUpg /VerName=Vxxxxxxx  
/PkgName=Wxxxxxxx.EXE /RelVer=x.x.x.xxxx /PosNo=0
```

The POS terminals, which have started the upgrade process, will then Abort the upgrade, and an alert, is sent to notify 'POS abort CD'.

Note:

The Xs represent the numeric characters in the package name, e.g., W8121050.

- 4 Type in **Exit** and press **Enter**. *The command window is closed.*

➤ **To Abort a v8 Upgrade on a Specific POS Terminal:**

- 1 From the Windows **Main Menu** select **Start**, then **Run**, and enter **CMD**.
- 2 From the Command Window, enter **CD\PCMASTER**, and press **Enter**.
- 3 Enter the command: `PkgAdmin /SendCdsMaint /AbortUpg /VerName=Vxxxxxxx /PkgName=Wxxxxxxx.EXE /RelVer=x.x.x.xxxx /PosNox`
- 4 Type in **Exit** and press **Enter**. *The command window is closed.*

The POS terminal, which will have started the upgrade process, will then Abort the upgrade, and an alert, is sent to notify 'POS abort CD'.

Activating and Monitoring Upgrades via Package Maintenance

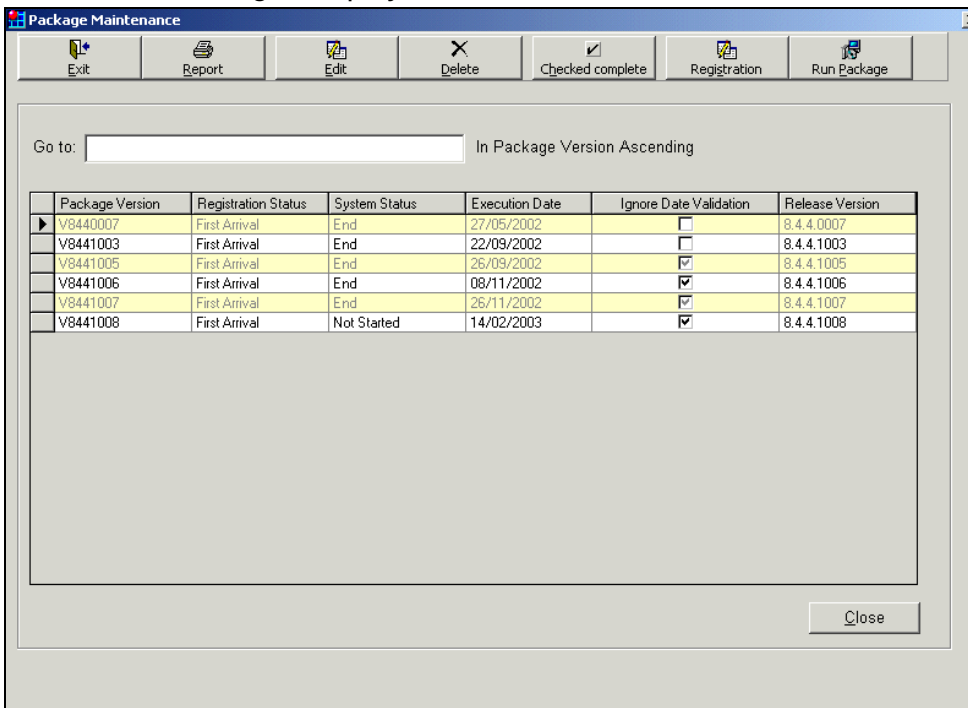
Package Maintenance is a utility that allows support personal to activate an upgrade manually. In addition, it is a monitoring utility that informs the user as to the status of each upgrade package that exists in the system, at any given moment. Using this utility eliminates the need to refer to SQL tables to obtain the status of each MFS and each PC. Relevant data for each upgrade package is obtained from the SQL tables. Possible statuses are: **First Arrival**, **Edit** and **Start**. Other information that can be viewed includes: Execution date and version number.

➤ **To Access v8 Package Maintenance:**

- 1 In the Main Menu tab, browse to the following folder:
System Administration\Technical Support\Package Maintenance. A dialog is displayed as follows:



- 2 Enter Administrator's password, then click **OK**. The main *Package Maintenance* dialog is displayed.



- 3 You have accessed the *Package Maintenance* dialog successfully. You can continue to activate a package manually from the *Package Maintenance* dialog, or monitor your upgrades from the *Package Maintenance* dialog.

Activating an Upgrade from the Application

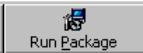
Once a package is *Registered* successfully, it is ready to run and perform the upgrade.

The **System Status** field is in a status of “**Not Started**”. If the status is anything else, the following message is prompted:

“This package can’t run. Its status is different from “**Not Started**.”

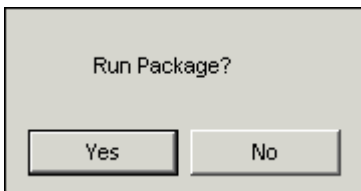
By activating an upgrade to run the package, the system triggers *PkgAdmin* to *Administer* the package. Once *PkgAdmin* begins, the process is automatic, as described in the procedure below.

➤ **To Run the v8 Package from the Application:**

- 1 From the *Package Maintenance* dialog, click the required Upgrade Package Version Number. The  sign will point to the line of the package you are running.
- 2 Click  to run the package. The process is synchronous and a message is displayed:

“Please Wait.”

The following message is displayed:



- 3 Click Yes to run the package. The package runs as an asynchronous process.

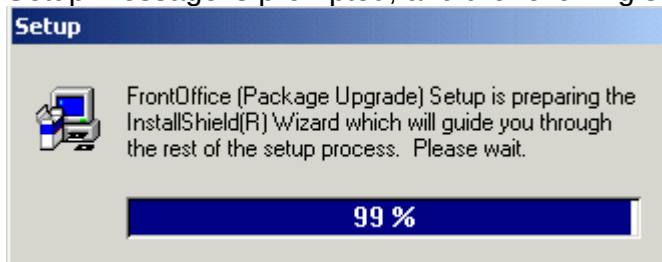
If the package is ready, the *Setup* dialog is displayed as shown in Step #4, indicating that the Package Upgrade is getting ready to start the process. Proceed to Step #4.

If the package is not ready to run, the following dialog is displayed:

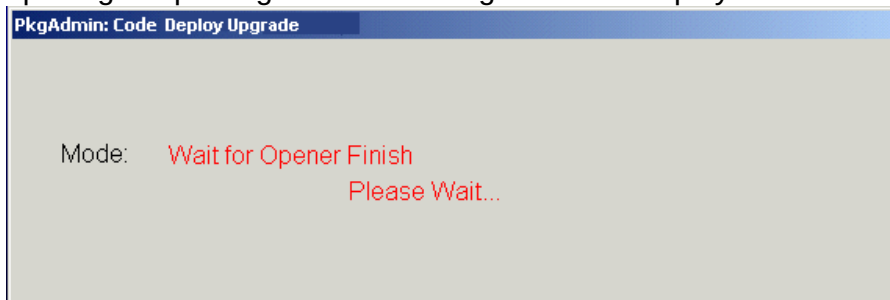


Click *OK*, and refer to the Pos_Srv.Log file located in C:\PcMaster\Log\POS_Srv.Log to locate the specific error, and to obtain additional information on the error, its location, package number etc.

- 4 Once the package starts to run, there will be no user interaction. The Setup message is prompted, and the following screen is displayed:



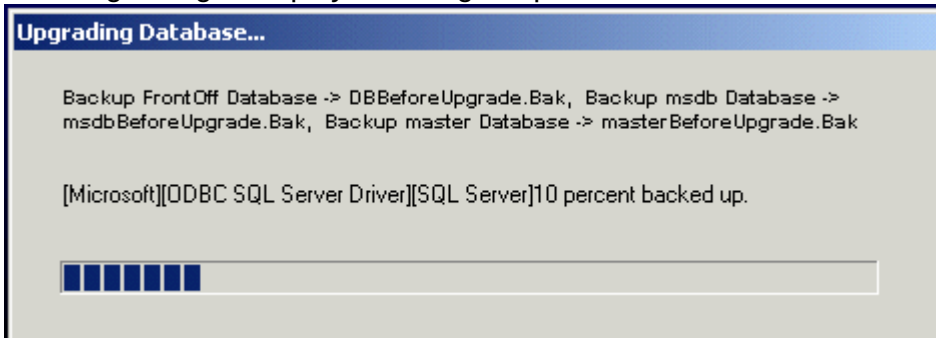
- 5 *PkgAdmin* is then invoked and the *Opener* utility begins the process of opening the package. The following screen is displayed:



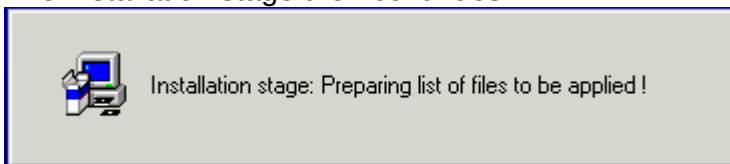
Once the Opener is finished opening the package, the files in the package are ready to be installed on MFS1. The process is automatic.

VERSION UPGRADING PROCESS

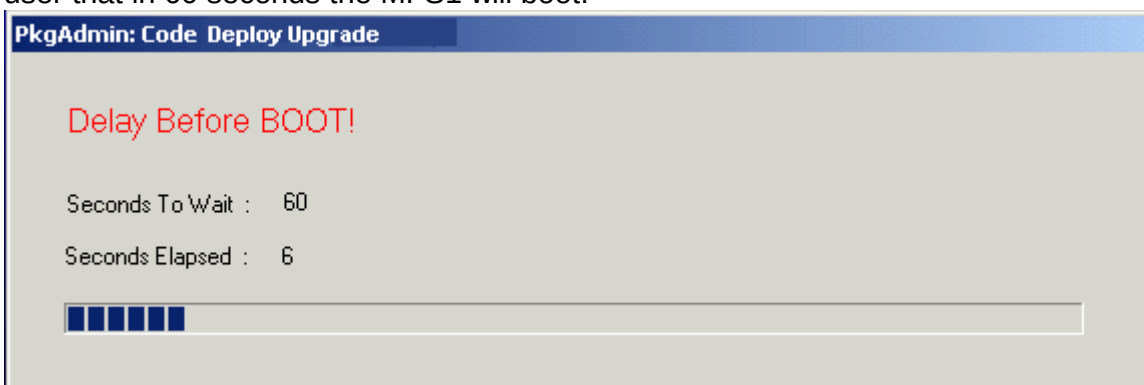
- 6 The next stage, which is referred to as the Installation Stage, automatically shuts down the POS Service Monitoring.
- 7 The upgrading of the database follows. The upgrade screen is will display the To and From versions.
- 8 The system saves the database before and after the upgrade. The following dialog is displayed during the process:



- 9 The Installation stage then continues....

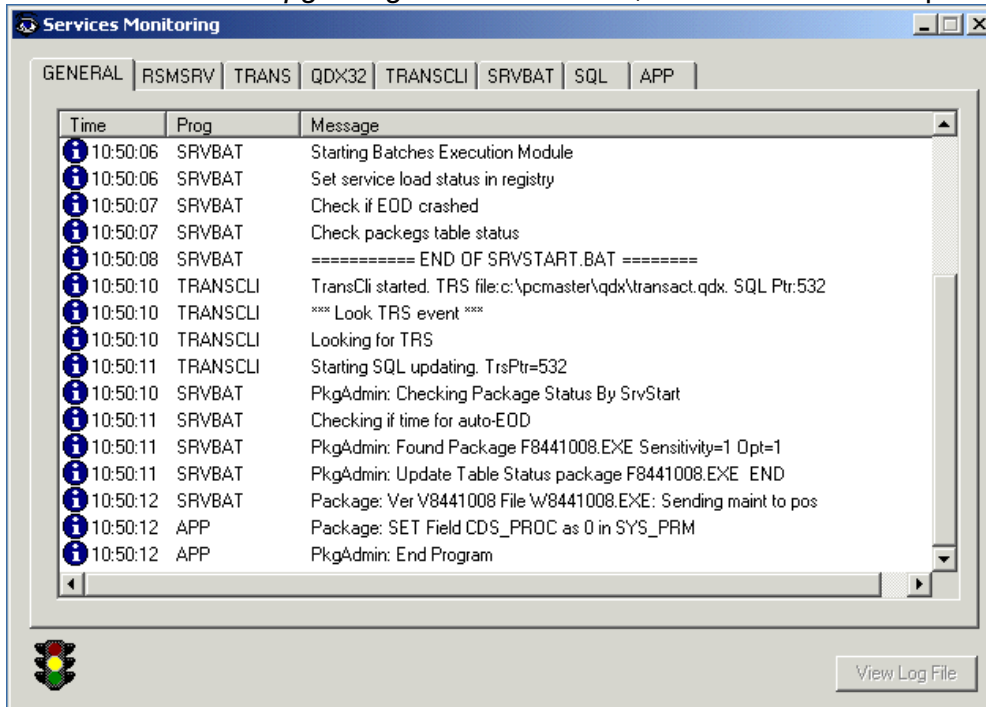


- 10 The files are opened and installed. The MFS1 is upgraded with the new package.
- 11 The MFS1 reboots, and the following screen is displayed, warning the user that in 60 seconds the MFS1 will boot:



- 12 The System boots and restarts, and the *Services Monitoring* screen is displayed, as follows:

The *Services Monitoring* screen displays general information on the package that was run, and indicates that maintenance will be sent to the POS terminals. (For further details on the upgrade of the MFS2 and LFSs see the section on *Upgrading MFS2 and LFSs*, earlier on in this chapter.)



- 13 If at any stage, there is a problem, the upgrade will fail, and a screen will prompt: "*CDS Upgrade Failed. Call Support!*" (For further details on upgrade failure, see the section on *Upgrade Failure in MFSs*, earlier in this chapter.)

Monitoring v8 Upgrades

Using the ‘*Checked Complete*’ Function from the Application

As described in the section on *RegistrPack* in *Chapter 3*, *CheckComplete* performs the necessary checks, and updates the system status to **End**, either after the next *EOD*, or via an alert, which updates the tables *VER_STATUS* in the *FO* database.

If you wish to *Administer* a package during *EOD*, but do not wish to wait for the next *EOD*, you have the option to click the *CheckComplete* button to change the system status to **End** preparing the PC to be able to start an upgrade. This function will activate the *CheckComplete* process, enabling you to quickly locate errors that may have occurred in the upgrade process. After the errors are fixed, you can activate *CheckComplete*, and if the process succeeded, the system status will be changed to **End**. If the process failed, a message prompts you to view the *Package Maintenance Report* as displayed below.

If you wish to *Administer* a package from the Application, as described in *Activating a Package from the Application*, you can activate the *CheckComplete* function in the same way as described above.

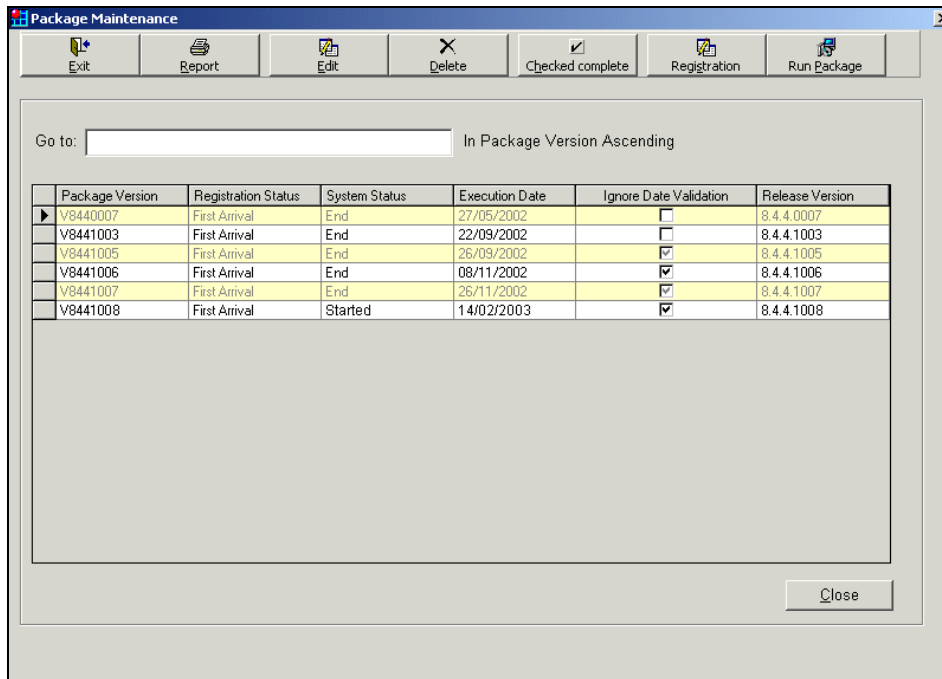
Below is a description of this process:

Warning:

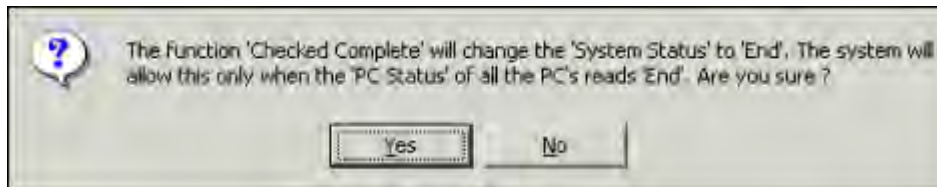
This procedure should be performed only by technical support personnel!

➤ **Using Checked Complete to Change v8 System Status:**

- 1 From the main *Package Maintenance* dialog displayed below, click the *Checked Complete* button:



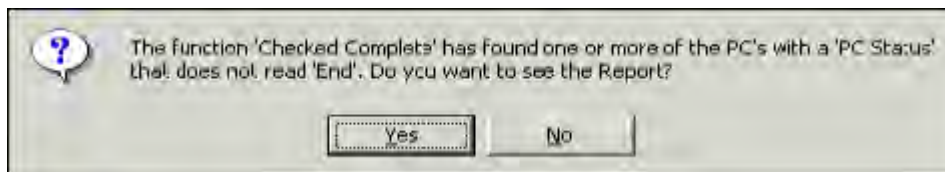
The following message is displayed:



- Click Yes to activate *CheckComplete* and to return to return to the *Package Maintenance* dialog. If no errors occurred, then the system status is changed to **End**.

Or

If any errors occurred, a message will be displayed as follows:



- Click Yes to display the report. The *Package Maintenance Report Preview* dialog is displayed:

VERSION UPGRADING PROCESS

Report Preview

100% 1 of 1

Package Maintenance Report
Food & Drug Store
Branch 801

Package Version	Release Version	Arrival Date	Registration Status	Execution Date
V8441008	8.4.4.1008	2/21/2003 12:00:00AM	First Arrived	2/21/2003 12:00:00AM

PCs Status

PC Type	PC No.	Status	Opener status	Error No.	Last Step	Last Step Date
BO	1	End	OK Boot	0	200	2/21/2003 10:26:13AM
PDS	1	End	OK Boot	0	0	2/21/2003 10:34:21AM
BO	2	Error	No Answer	260	100	2/21/2003 11:02:51AM
PDS	2	Not Start	0	0	0	
PDS	16	Not Start	0	0	0	
PDS	88	Error	0	0	0	2/21/2003 10:26:25AM

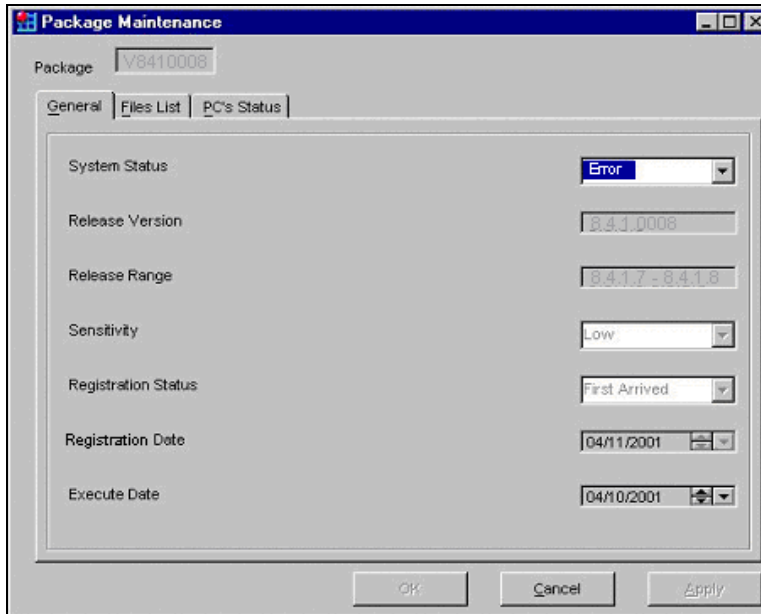
Files List

File Type
F8441008.EXE
V8441008.CTL
W8441008.EXE

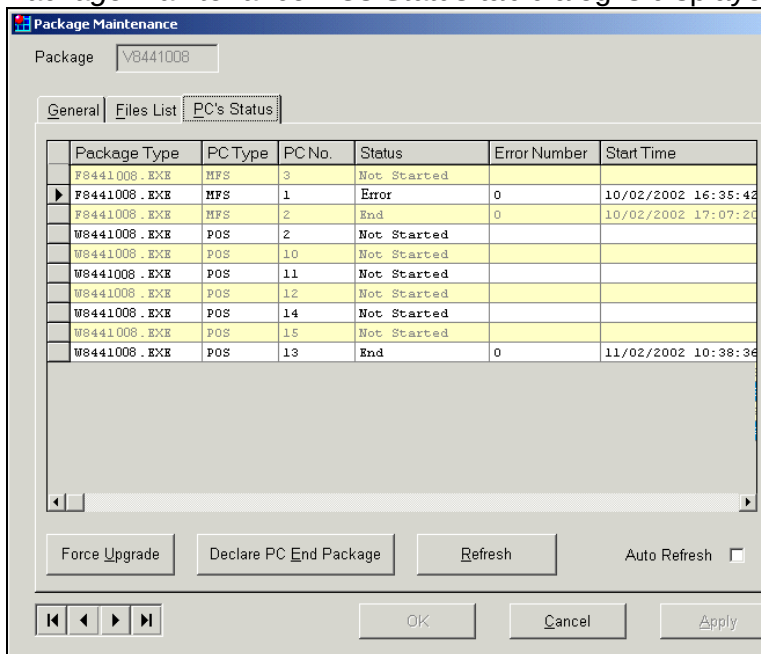
- 4 From the top right-hand corner of the *Package Maintenance Report Preview* dialog, click  to close the dialog and to return to the *Package Maintenance* dialog.
- 5 From the main *Package Maintenance* dialog, double-click the line of the Package Version where you require an **End** status,

Or

Click to highlight the line of the Package Version, and then click . The *Package Maintenance* dialog, displaying the *General*, *Files List*, and *PCs Status* tabs, is displayed:



- 6 Select the *PCs Status* tab to locate the PCs on which errors occurred during the upgrade process and to fix errors on the relevant PCs. The *Package Maintenance PCs Status* tab dialog is displayed:



- 7 From the *PCs Status* tab dialog, click the line of the PC where the error was displayed, and then click *Force Upgrade* to begin the upgrade on the required PC. An alert will be recorded indicating a successful upgrade. The system status will be changed to **End**.

Warning:

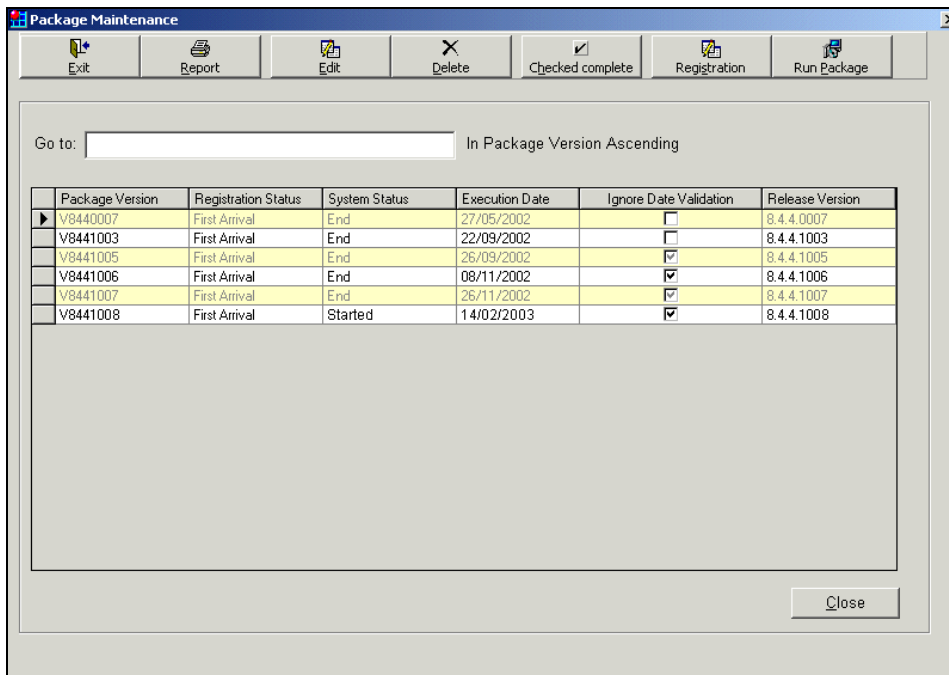
Refer to section below on *Force Upgrade*. This procedure should be performed only by technical support personnel!

General Tab

The *General* tab of the *Package Maintenance* dialog represents the data in VER_DATA table. Viewing data in this tab enables you to locate problems and errors that have occurred during the upgrade.

➤ **To Access the v8 General tab:**

- 1 From the main *Package Maintenance* dialog, as displayed below, click the required upgrade version number:

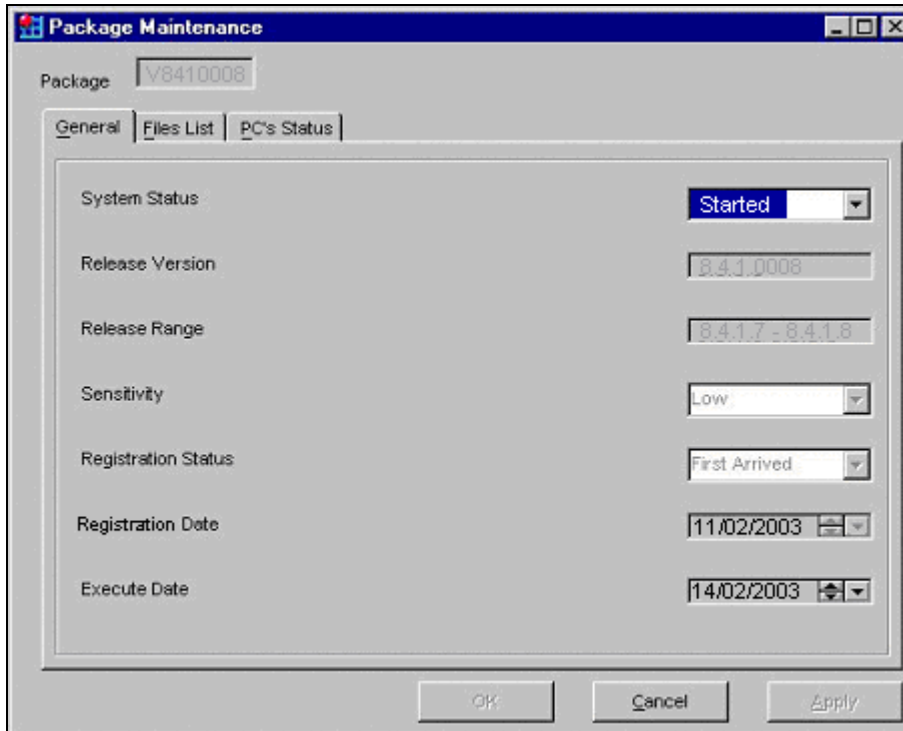


- 2 Then click



Or

Double click the relevant Package version line. The *Package Maintenance* dialog is displayed:



The table below describes the data in the fields of the *General Tab*:

Fields	Description
System Status	Optional statuses include: Started ; Not Started ; Error ; End (for further information on the implications of changing System Status refer to section on <i>Guidelines for Changing System Status</i> , below.)
Release Version	Number of release version.
Release range	Version range covered by upgrade package.
Sensitivity	Indicates package sensitivity, e.g., Low or High.
Registration Status	Registration status of upgrade package. Statuses include: First Arrival : Indicates a package that has first arrived. Edit : Indicates a package in which only the V*.CTL file has been changed. Edit All : Indicates a package in which there were changes in all 3 files.
Registration Date	Date on which Registration was performed.

Fields	Description
Execute Date	Scheduled date for upgrade as appears in the .CTL file.

Note:

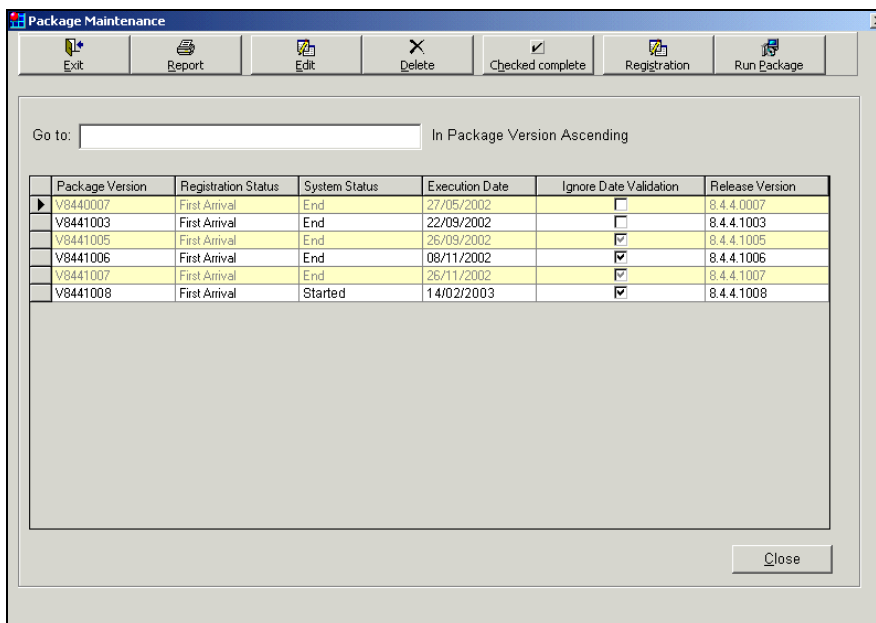
If the **Ignore Date Validation** field is checked, the .CTL file contains a value of 00000000 and the date of execution is ignored. This field is located on the *Package Maintenance* dialog.

Files List Tab

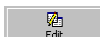
The *Files List* tab of the *Package Maintenance* dialog, represents the data in VER_FILES table, including .EXE and .CTL files. Viewing data in this tab informs you of the location of the various files of the upgrade package, in the database, in **Wait**, in **Done** or in **Reject** folders.

➤ **To Access the v8 Files List tab:**

- 1 From the main *Package Maintenance* dialog, as displayed below, click the required upgrade version number:



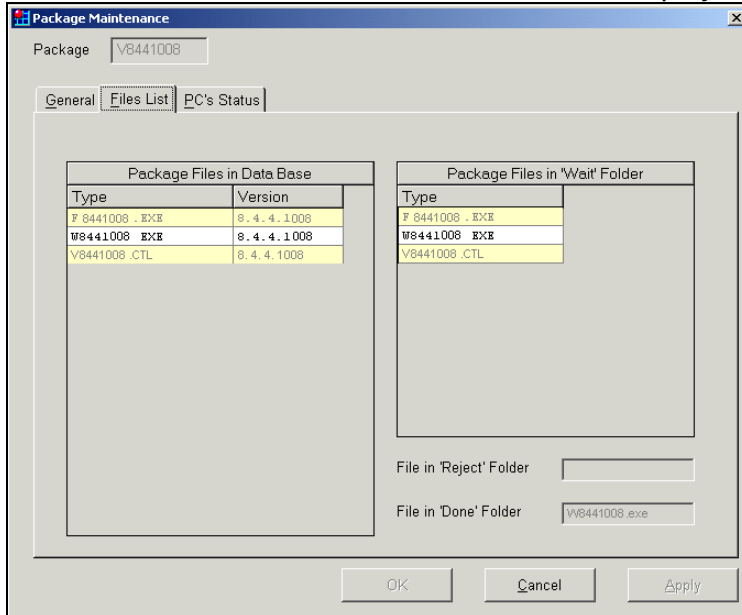
- 2 Then click



Or

Double click the relevant Package version line. The *Package Maintenance* dialog is displayed.

- 3 Select the *Files List* tab. The *Files List* tab is displayed, as follows:



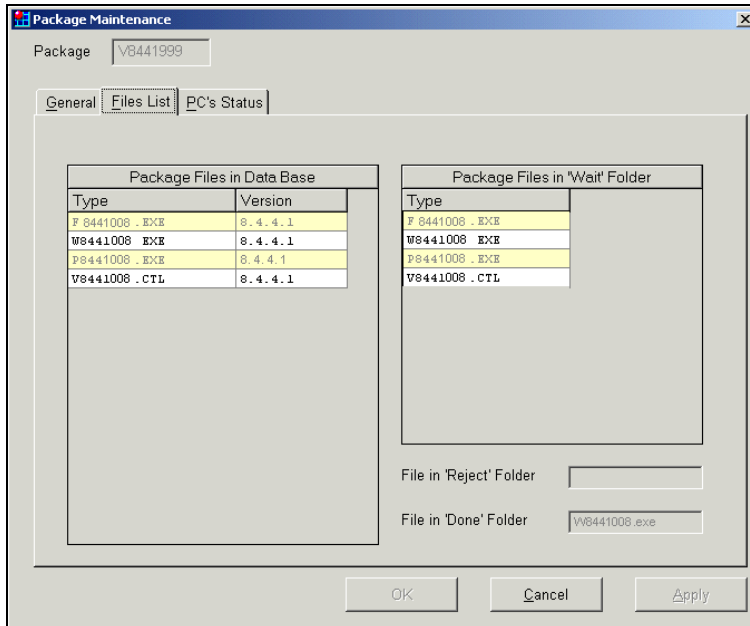
Note:

In upgrades for Fuel Stations the PFS package is located in the same folder of the WinPOS package. The names of the PFS packages are:

PumpSrv – SXXXXXXX.EXE.

POSPump – PXXXXXXX.EXE.

VERSION UPGRADING PROCESS

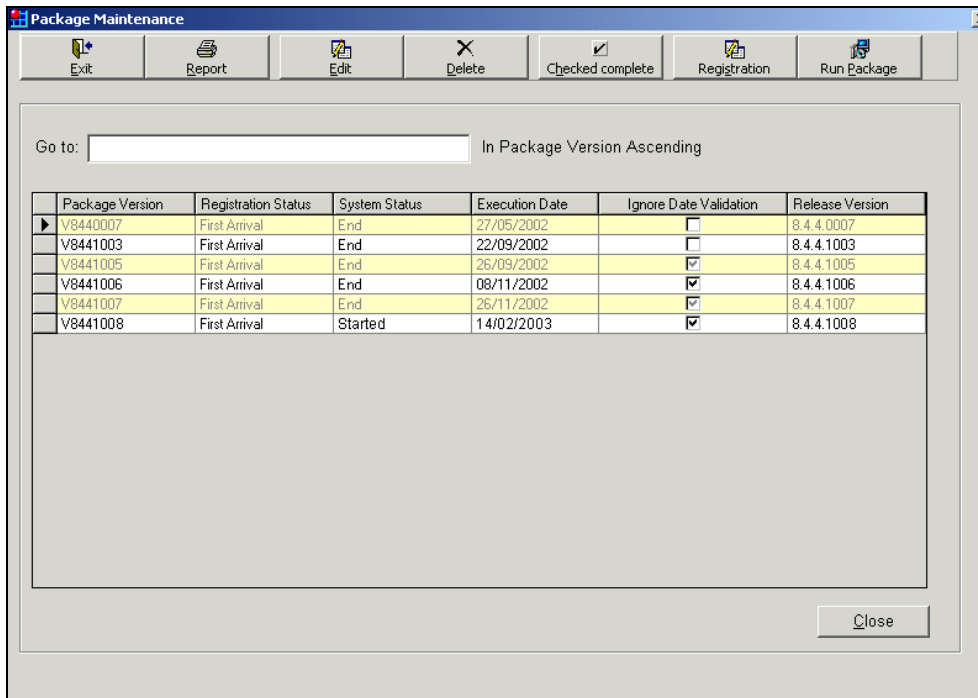


PCs Status Tab

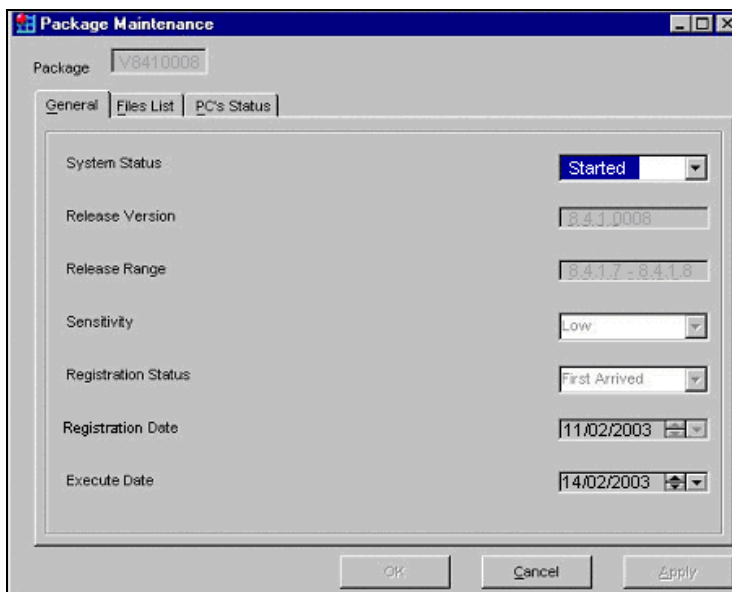
The *PCs Status* tab of the *Package Maintenance* dialog represents the data in *VER_STATUS*. This tab enables you to view the upgrade status of each PC in the system, as well as to locate problems and errors that have occurred in the upgrade process.

➤ **To View v8 Status of Upgrade in PC in the PCs Status Tab:**

- 1 From the main *Package Maintenance* dialog, as displayed below, click the required upgrade version number. The  points to the package you select, then click .

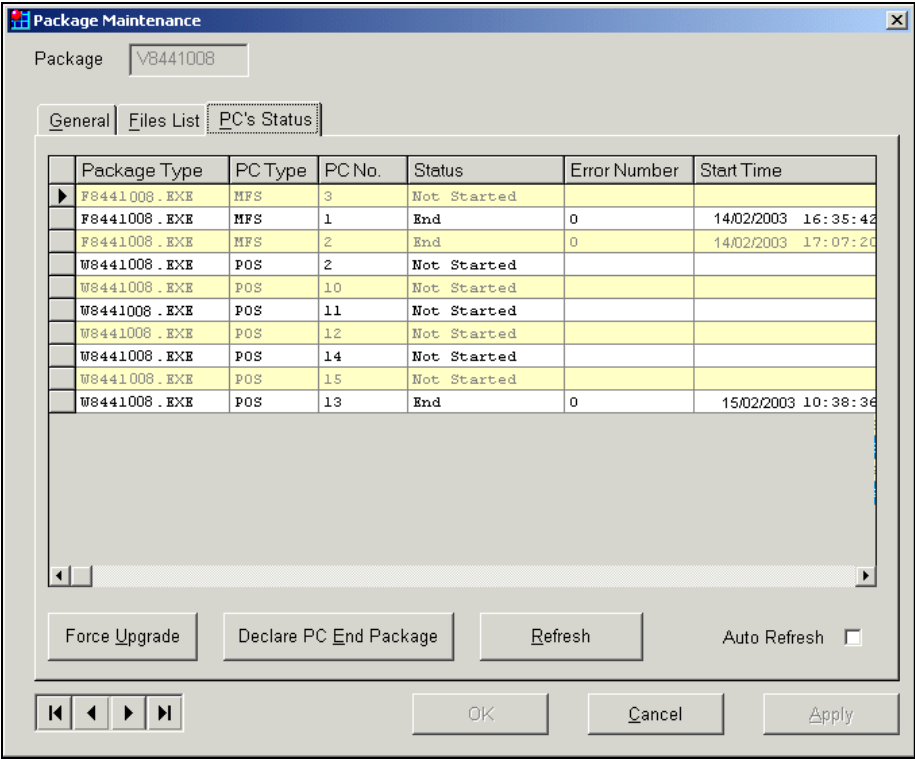


The *Package Maintenance* dialog is displayed:



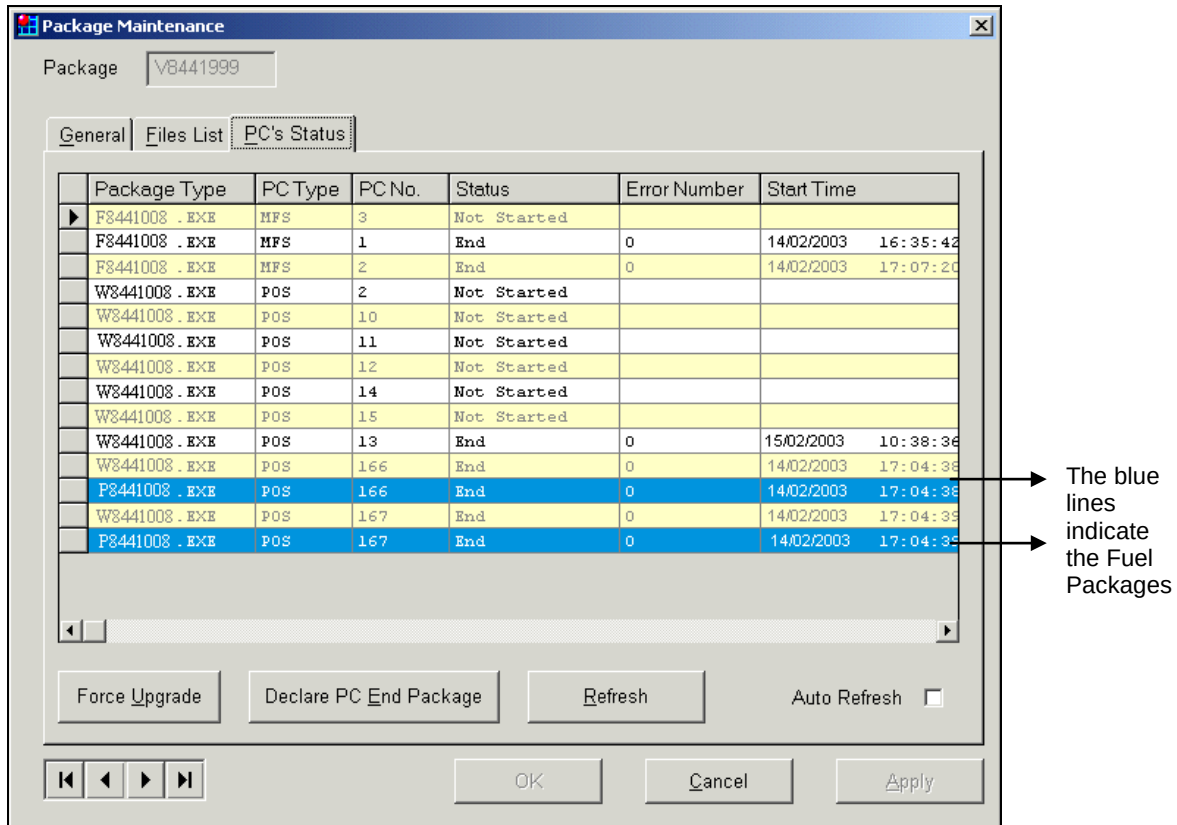
- 2 Select the *PC's Status* tab. The *PC's Status* tab dialog is displayed:

VERSION UPGRADING PROCESS



The *PC Status* tab displays the status of the package upgrade on each PC, including Error numbers if any, Start and End time.

The following dialog displays the PC Status of a package including Fuel Packages:

**Note:**

For further details on PFS components, refer to *Chapter 6, CDS Support for PFS*.

The table below describes the data in the columns:

Columns	Description
Package type	Indicates the type of package. Screens with yellow and white striped lines indicate a package designated for POS Terminals in a Store. Screens with blue and white striped lines indicate a package designated for POS Terminals in a Fuel Station.
PC Type	Indicates the type of machine, e.g., MFS or POS terminal.
PC Number	Code Number of PC or MFS in system.
Status	Indicates type of status, e.g., End , Started , etc.
Error Number	Code of error type.
Start Time	Indicates time when upgrade process began.
End Time	Indicates time when upgrade process ended.
Previous Release	Indicates previous version installed in system.

Refresh and Auto-Refresh Functions

You can click the *Refresh* button to refresh status data displayed in the dialog.

Alternatively, you can select the *Auto Refresh* checkbox to have the data refreshed automatically every 5 minutes.

Force Upgrade

If a POS terminal, MFS2 or LFS did not begin upgrading, or did not upgrade properly, whether due to a power failure or for any other reason, you have the option of performing *Force Upgrade*.

Note:

You cannot perform *Force Upgrade* on a specific package in a Fuel package. *Force Upgrade* is performed on the WinPOS package, which may include the PFS components. For further details on Fuel packages, refer to *Chapter 6, CDS Support for PFS*.

- In *Low-Sensitivity* packages only, *Force Upgrade* can be performed for MFS2, LFSs and for POS terminals.
- In *High-Sensitivity* packages, *Force Upgrade* cannot be performed for MFSs at all, but only for POS terminals.
- *Force Upgrade* cannot be performed on MFS1. If you attempt to do so, a message will be displayed indicating that you must first run the package from the beginning.
- You can perform *Force Upgrade* on the POS terminals from MFS1, MFS2 or LFS.

Warning:

This procedure should be performed only by technical support personnel!

➤ To perform v8 Force Upgrade in High and Low-Sensitivity Packages on MFSs LFSs or POSTs:

- 1 In *Package Maintenance* in the *PCs Status* tab highlight the required PC to be upgraded.
- 2 Click *Force Upgrade*. A message is displayed confirming that a *Force Upgrade* will be performed on the specified PC, and prompting you to confirm.
- 3 Click *Yes* to begin upgrade. When upgrade process is completed, *End* status will appear in table.

➤ **To Perform v8 Force Upgrade in Low-Sensitivity Packages to a Specific POS Terminal from the Command Line:**

- 1 From the Windows **Main Menu** select **Start**, then **Run**, and enter **CMD**.
- 2 From the Command Window, enter **CD\PCMASTER**, and press *Enter*.
- 3 Then enter the command: **PkgAdmin /SendCDsMaint/Force /VerName=Vxxxxxxx /PkgName =Wxxxxxxx.EXE /RelVer =x.x.x.x /PosNo=till no.**
- 4 Type in **Exit** and press **Enter**. *The command window is closed.*

➤ **To Perform v8 Force Upgrade in Low-Sensitivity Packages to all POS Terminals Using the Command Line:**

- 1 From the Windows **Main Menu** select **Start**, then **Run**, and enter **CMD**.
- 2 From the Command Window, enter **CD\PCMASTER**, and press *Enter*.
- 3 Then enter the command: **PkgAdmin /SendCDsMaint /Low /VerName=Vxxxxxxx /PkgName =Wxxxxxxx.EXE /RelVer =x.x.x.x**
- 4 Type in **Exit** and press **Enter**. *The command window is closed.*

➤ **To Perform v8 Force Upgrade in High-Sensitivity packages on POS Terminals Using the Command Line:**

- 1 Create a flag file in MFS1 in the following path: **C:\pcmaster\FLAG**
The flag should be named **Vxxxxxxx.END**
- 2 From the Windows **Main Menu** select **Start**, then **Run**, and enter **CMD**.
- 3 From the Command Window, enter **CD\PCMASTER**, and press *Enter*.
- 4 Then enter: **PkgAdmin /SendCDsMaint /High /VerName =Vxxxxxxx /PkgName =Wxxxxxxx.EXE /RelVer =x.x.x.x**
- 5 When finished, type in **Exit** and **Enter**.

Declare PC End Package Button

If, at the end of the upgrade process, the status of a particular PC or MFS is other than **End**, (i.e. **Start**, **Not Started** or **Error**) and you wish to change the status of the PC to **End**, this can be performed using the *Declare PC End Package* function. However, this function will not perform an upgrade (as opposed to the *Force Upgrade* function), but rather will only change the status of a PC or MFS in *Package Maintenance* to display **End**.

Warning:

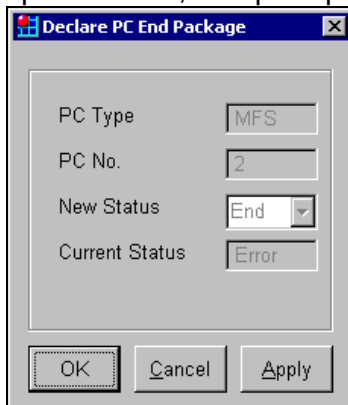
This procedure should be performed only by technical support personnel!

Note:

You cannot use *Declare PC End Package* on a Fuel package.

➤ To perform v8 Declare PC End Package on a PC or MFS:

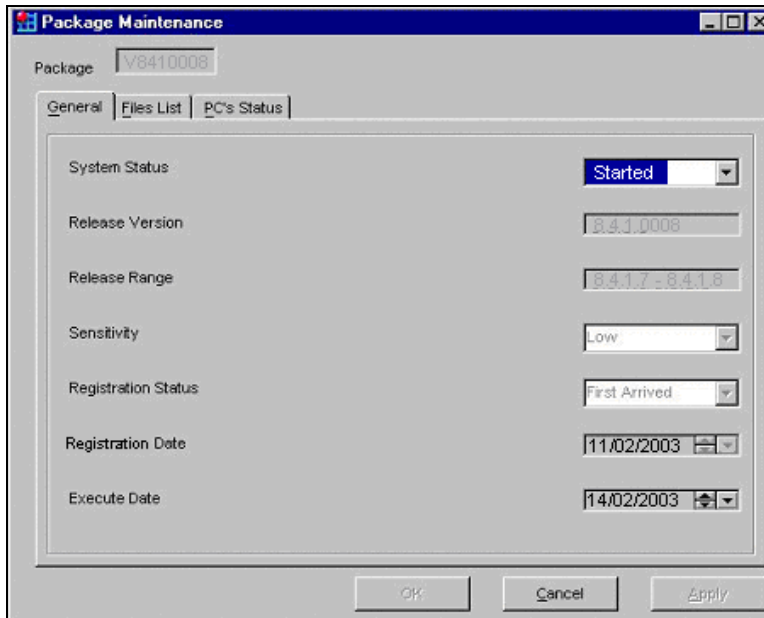
- 1 In *Package Maintenance* in the *PCs Status* tab highlight the relevant PC or MFS.
- 2 Click *Declare PC End Package*. The *Declare PC End* dialog is displayed, confirming that *Declare PC End Package* will be performed on the specified PC, and prompting you to confirm:



- 3 Click *Ok* to change status to **End**.

Guidelines for Changing System Status

There are several possible statuses of an upgrade in the system. The current status of an upgrade can be viewed in the **System Status** field of the following screen:



The screenshot shows a Windows-style dialog box titled "Package Maintenance". It has three tabs: "General", "Files List", and "PC's Status". The "General" tab is selected. At the top, there is a "Package" field containing the text "V8410008". Below this, there are several fields with labels on the left and controls on the right:

- System Status:** A dropdown menu currently showing "Started".
- Release Version:** A text field containing "8.4.1.0008".
- Release Range:** A text field containing "8.4.1.7 - 8.4.1.8".
- Sensitivity:** A dropdown menu currently showing "LOW".
- Registration Status:** A dropdown menu currently showing "First Arrived".
- Registration Date:** A date field showing "11/02/2003".
- Execute Date:** A date field showing "14/02/2003".

At the bottom of the dialog box, there are three buttons: "OK", "Cancel", and "Apply".

The possible statuses are:

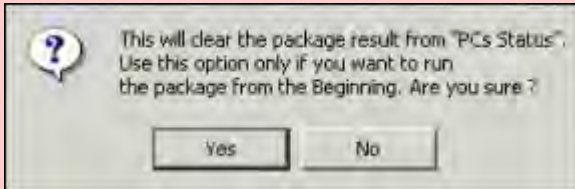
Not Started - automatically displayed once a package is registered by *RegistrPack* to the database, but has not yet begun the upgrade; once selected, the field becomes disabled.

End - indicates that the upgrade process has ended. This field becomes disabled in the event that the next scheduled package for upgrade is other than **Not Started**; in this case all buttons in the PC's Status tab are disabled, and become enabled only when the next scheduled package for upgrade is in **Not Started** status.

Warning:

There are severe implications if the *System Status* field displays a status other than **Not Started** (i.e., **Start**, **End** or **Error**) and administrator changed it to **Not Started**:

1. The following warning message is displayed:



2. Selecting Yes will impact all MFSs and PCs in the store, and should be applied only if you wish to run the entire upgrade package from the beginning.
3. The **Not Start** field will become automatically disabled.

Note:

If **Start** or **Error** are displayed in the field and then changed to **End**, the system first verifies within the PCs status tab that all PCs have completed the upgrade (i.e. with a status **End**). If at least one PC is not in **End**, the message "System not completed. Are you sure?" displays. Selecting Yes will change the status to **End**.

Note:

If the system has found that all PCs and MFSs have reached **End**, once the status has been changed to **End** the message "Are you sure?" displays.

Note:

If **Start** is displayed in the field and then changed to **Error**, the message "This will stop the MFSs from running the package. Are you sure?" displays.

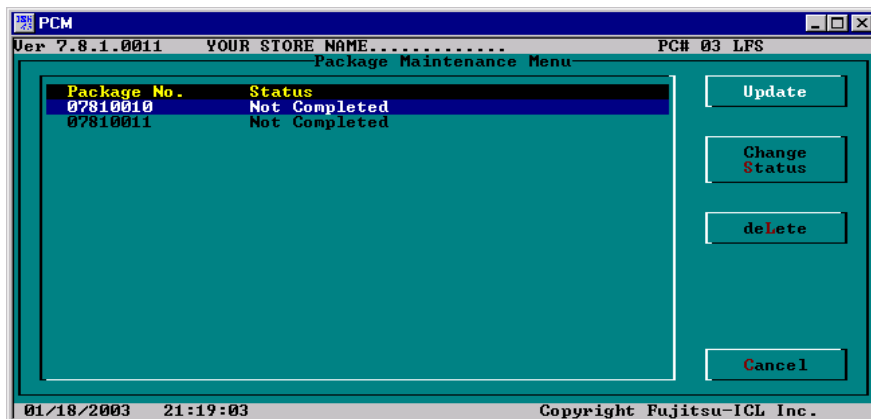
Note:

If **Error** or **End** are displayed in the field and then changed to **Error**, the message "Are you sure?" displays.

Monitoring v7 Upgrades

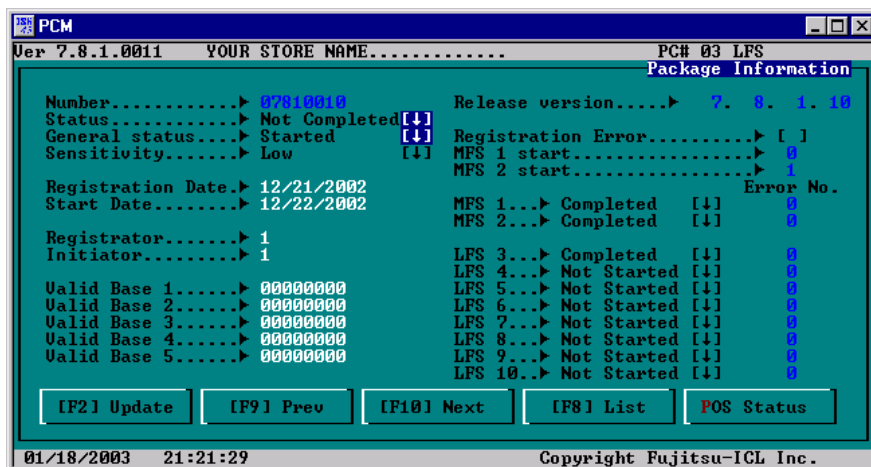
➤ **To check Package Status on ISS45 v7:**

- 1 From the *Main Menu*, browse to *Package Status Maintenance* (6-1-1-11-3). The *Blackout Period Setup* screen displays each package status:

**Note:**

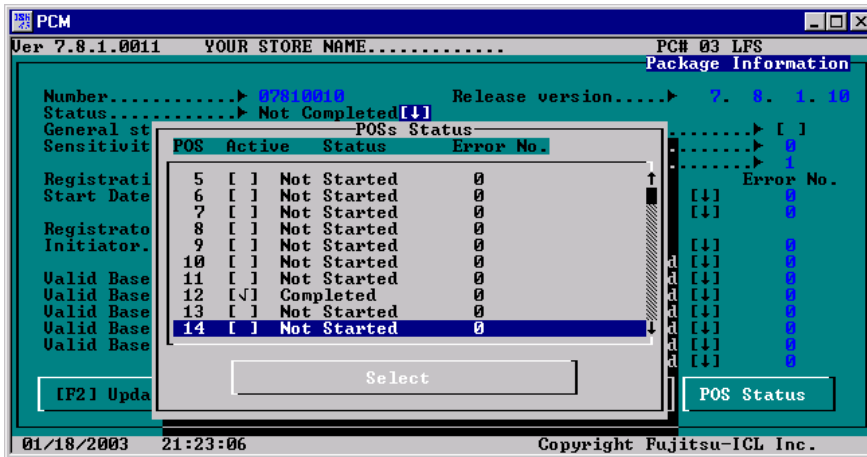
You may change the status of a highlighted package by pressing the <Alt><U>.

- To view details, highlight the desired package and press <Alt><U>. The Package Information screen displays:

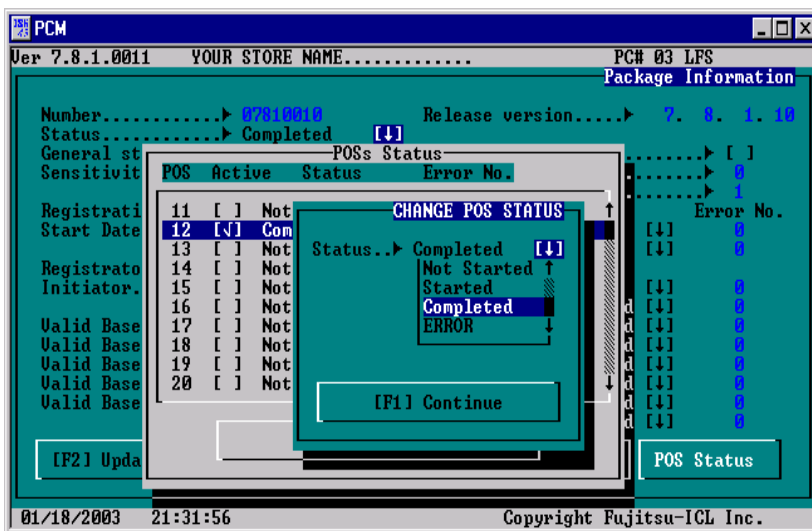
**Note:**

The status screen displays the status of each terminal and any associated error number. You may optionally change Package Information from this screen as well.

- To view the POS package status, press <Alt><P>. The POS Status screen displays:



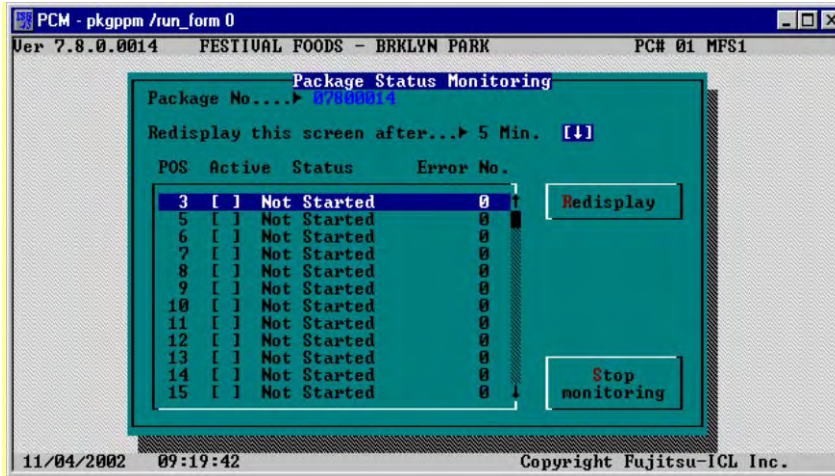
- 4 (Optional) To change the package status for a POS terminal, highlight the POS and press Enter. *The Change POS Status screen displays:*



- 5** Choose the desired status and press **<F1>** to continue.

Note:

The C:\PCMASTER\PKGPPM.EXE application, when called from within *PosSvMon.exe* (based on a flag set during *pcmpkgnw.exe* at EOD) using the switches */run_form* and *[Record Number]* will launch the following monitoring screen:



For example: **C:\pcmaster\pkgppm.exe /run_form 0**

The record number is generated when flag is set.

5

Running CDS on Demand

This chapter describes the processes when CDS is run on demand.

In this chapter:

- Overview
- Defining CDS Parameters
- *RegistrPack* and *PkgAdmin*
- Running CDS on Demand
- Running CDS on Demand from on Site

Overview

This chapter describes how to Run *CDS* on Demand to *Register* and *Administer* a package upgrade in stores manually. Running *CDS* on demand may only be performed on MFS1.

Defining *CDS* Parameters

The *CDS* parameters must be defined to be able to send package upgrades. For further details on defining *CDS* parameters, refer to *Chapter 2, CDS Setup*.

RegistrPack and *PkgAdmin*

There are no differences between the functionality of *RegistrPack* and *PkgAdmin* when sending upgrades to a store to be performed “on demand”. For further details on *RegistrPack*, refer to *Chapter 3, Package Management Utility* and on *PkgAdmin* refer to *Chapter 4, Package Administration*.

Note:

Running *CDS* “on Demand” using *RegistrPack* and *PkgAdmin* is not available in ISS45 v7.

Running *CDS* “On Demand” at the Store

The following procedure describes how to run *CDS* on demand from MFS1 on site in the store.

This procedure could be used in the following scenario. If the Store Manager needs to send an immediate package upgrade, but does not have access to the *ISS45 Office* application, using remote access the user can run *CDS* by entering a command from MFS1 to *Register* and *Administer* a package.

The process for *CDS* on demand includes the following command lines, which are used in the following order:

- 1 RegistrPack /Manual /Registration.
- 2 PkgAdmin /Manual /RunPkg /VerName=Vxxxxxxx.

Notes:

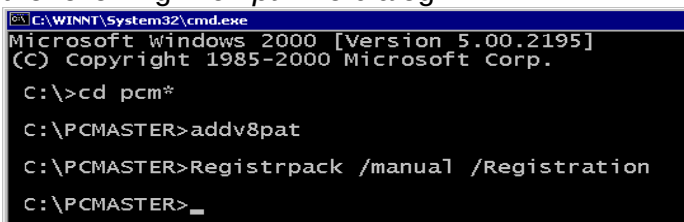
The Xs represent the numeric characters in the package name, e.g., 8441008.

The following commands should not run if the EOD process is scheduled to run. Check that the flag EODxxxxx.flg does not exist before entering and running the command. The flag is located in the PCMASTER \ FLAG folder.

➤ **To Run CDS on Demand on a v8 Site:**

The command lines trigger the exact same processes described in *Chapter 3, RegistrPack Functionality from the Application* and *Chapter 4, Activating the Package from the Application*.

- 1 From the Windows **Main Menu** select **Start**, then **Run**. The *Run* dialog is displayed.
- 2 Enter **cmd**. The *Prompt Line* dialog is displayed. The *Prompt Line* is located in the directory set up by your system administrator, e.g., C:\WINNT\System32\cmd.exe.
- 3 From the C:\WINNT\System32\cmd.exe window, enter the command: **C:\PCMASTER>advv8pat**, to enable the path from which you are running the application.
- 4 Press *Enter*, and then enter the command: **C:\PCMASTER>Registrpack /manual /Registration** to invoke *Registrpack*, as shown in the following *Prompt Line* dialog:



```
C:\WINNT\System32\cmd.exe
Microsoft Windows 2000 [Version 5.00.2195]
(C) Copyright 1985-2000 Microsoft Corp.

C:\>cd pcm*

C:\PCMASTER>advv8pat

C:\PCMASTER>Registrpack /manual /Registration

C:\PCMASTER>
```

The process is run as a synchronized process.

- 5 Verify for success or failure of the *Registration* in the Log files located in **C:\PcMaster\Log**.

Notes:

1. The log file **C:\PCMASTER\Log\RegPack.Log** records all the activities performed by *RegistrPack*, and can be used for problem analysis. In addition, alerts are recorded at key stages in the process of the *RegistrPack* functionality.
2. There are several different reasons for a package to fail registration. For further details refer to the section on *Registration/If Registration fails* in *Package Management Utility, Chapter 3*.
- 6 Verify that the package is ready to be *Administered* and run by checking the **Wait** or **Reject** folders located in **C:\CDPackagesRoot**. If the package is in the **Wait** folder, it indicates that it is ready to be run.
- 7 If the package is ready to run, enter the following command:
C:\PCMASTER>PkgAdmin /Manual /RunPkg /Vername=Vxxxxxxx.
- 8 The process is run as an unsynchronized process.
- 9 If the package did not run, refer to the Pos_Srv.log file located in **C:\PCMaster\Log\Pos_srv.Log**.

Note:

Once the package starts to run, there will be no user interaction. The process is the same as in the section that describes *Activating a Package from the Application* described in *Chapter 4*. To view the process, and the setup and installation screens, refer to *Chapter 4, Package Administration, Activating the Package from the Application/To Run the Package from the Application*.

6

CDS Support for PFS

This chapter describes how CDS supports the installation of PFS (Primary Fuel Server) components, to enable upgrades of Fuel systems.

In this chapter:

- Overview
- Defining CDS Parameters
- Using RegistrPack
- Package Administration
- Version Upgrade Process
- Upgrading Fuel POS Terminals
- Upgrade Failure

Overview

PFS (Primary Fuel Server) components are installed as part of the WinPOS file, which contains the upgrade for the POS terminals, providing flexibility and consistency in applying *CDS* upgrades to Fuel systems.

Note:

CDS Support for Primary Fuel Server components is not available in ISS45 v7.

The *PFS* components include:

- ***PumpSrv*** – ISS45's main fuel server providing an interface to the POS Terminals, the Convert Layer, and Back Office.
- ***PosPump*** – Transfers transactions between *PumpSrv* and the POS Terminals.

CDS may be run during *EOD*, or via the Application as explained in *Chapters 3 and 4*, earlier on in this manual.

This chapter describes how *CDS* performs upgrades on a Fuel system that includes *PFS* components.

Defining CDS Parameters

The following procedures include the defining of specific parameters and must be performed to enable software upgrades:

- Configuring MFSSs in System using PC in System.
- Configuring Active POS terminals.
- *CDS* Checkout bank parameters.
- *CDS* and R-Session.
- Setting Up Disk Space Alerts.
- Setting the System to Auto Log-On.

For further details on defining *CDS Setup* Parameters, refer to *Chapter 2, CDS Setup*.

Using *RegistrPack*

Once you have defined the *CDS* parameters listed above in *Defining CDS Parameters*, the Store is ready to receive an upgrade package.

After an upgrade package arrives at a Fuel Station, the System Administrator transfers the package to **New**. *RegistrPack* identifies and handles the new package in 2 different ways:

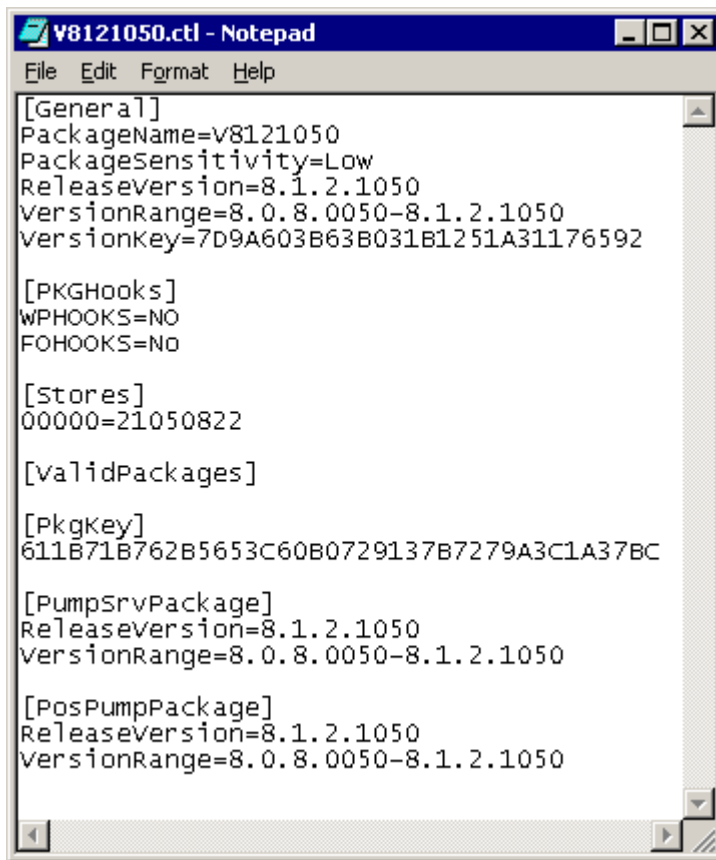
- **At *EOD***
- **From the Application (no dependency on *EOD* Process)**

For further details on the functionality of *RegistrPack* during *EOD*, or when run manually from the Application, refer to *Chapter 3, Package Management Utility*, in this manual.

Note:

There are no differences between the functionality of *RegistrPack* when sending upgrades to a Fuel Station or to a Store, other than the addition of the *PFS* components in the .CTL file.

The following displays the format of a .CTL file, which includes the *PumpSrv* and *PosPump* packages required for upgrades on a Fuel system:



```
[General]
PackageName=v8121050
PackageSensitivity=Low
ReleaseVersion=8.1.2.1050
VersionRange=8.0.8.0050-8.1.2.1050
VersionKey=7D9A603B63B031B1251A31176592

[PKGhooks]
WPHOOKS=NO
FOHOOKS=NO

[Stores]
00000=21050822

[ValidPackages]

[PkgKey]
611B71B762B5653C60B0729137B7279A3C1A37BC

[PumpSrvPackage]
ReleaseVersion=8.1.2.1050
VersionRange=8.0.8.0050-8.1.2.1050

[PosPumpPackage]
ReleaseVersion=8.1.2.1050
VersionRange=8.0.8.0050-8.1.2.1050
```

Package Administration

After *RegistrPack* has completed the registration procedures, the system is ready to be upgraded. Upgrades are performed by the *PkgAdmin* utility either at *EOD*, or when *CDS* is activated from the Application. *PkgAdmin* is responsible for coordinating the synchronization of the upgrade process, and is described in detail in *Chapter 4, Package Administration*.

Note:

There are no differences between the functionality of *PkgAdmin* when sending upgrades to a Fuel Station or to a Store.

Version Upgrade Process

Before the package reaches the POS terminals to continue with the upgrade process, the package is validated and registered.

Packages may be designated for the MFS or for a POS terminal.

- *PkgAdmin* invokes the *Opener*; the utility, which opens new packages and performs the upgrades.
- The *Opener* performs upgrades on the MFS1 server, and after a successful upgrade, the VER_STATUS SQL table is updated.
- Upon completion of the MFS1 upgrade, the MFS2 is upgraded, and the VER_STATUS table for MFS2 is updated.

The same process occurs in the LFS.

For further details on Upgrading MFS1 and MFS2, *Activating and Monitoring Upgrades via Package Maintenance*, refer to *Chapter 4, Package Administration*, in this manual.

Upgrading Fuel POS Terminals

The Upgrading process performed by CDS in Fuel systems includes the same steps as in CDS when sending upgrades to POS Terminals in a store.

The difference is that a package for a Fuel system includes the fuel components. The PFS components are located in the same folder as the WinPOS package, which includes the upgrade for the POS terminals.

For Fuel POS terminals the following PFS components are included:

- *PumpSrv* – Sxxxxxxx.EXE
- *PosPump* – Pxxxxxxx.EXE

Note:

The Xs represent the numeric characters in the package name, e.g., 8441008

The packages are InstallShield executables like the package of WinPOS Wxxxxxxx.EXE.

➤ **To Upgrade v8 Fuel Systems POS Terminals:**

Note:

Once the following procedure begins, there will be no user interaction.

- 1 Each POS Terminal identifies which package it must copy. Each package contains the WinPOS package.

- 2 Each POS terminal performs RCOPY32 of Wxxxxxxx.EXE to copy the files to C:\CDPackagesRoot\Wait.
- 3 Each POS Terminal performs RCOPY32 to copy the relevant packages that it supports, according to the validation performed in the Registry path:

HKEY_LOCAL_MACHINE\Software\Pointofsale\PumpSrv\Install.

If the POS Terminal supports the install of *PumpSrv*, it will copy the Sxxxxxxx.EXE file.

Or

HKEY_LOCAL_MACHINE\Software\Pointofsale\PosPump\Install.

If the POS Terminal supports the install of *PosPump*, it will copy the Pxxxxxxx.EXE file.

- 4 The WinPosCd.EXE executes each package.

The upgrades are performed in the following order:

- a. WinPOS – Wxxxxxxx.EXE
- b. PumpSrv – Sxxxxxxx.EXE
- c. PosPump – Pxxxxxxx.EXE

- 5 If the POS Terminal identifies that *PumpSrv* is in-active, the upgrade will begin:

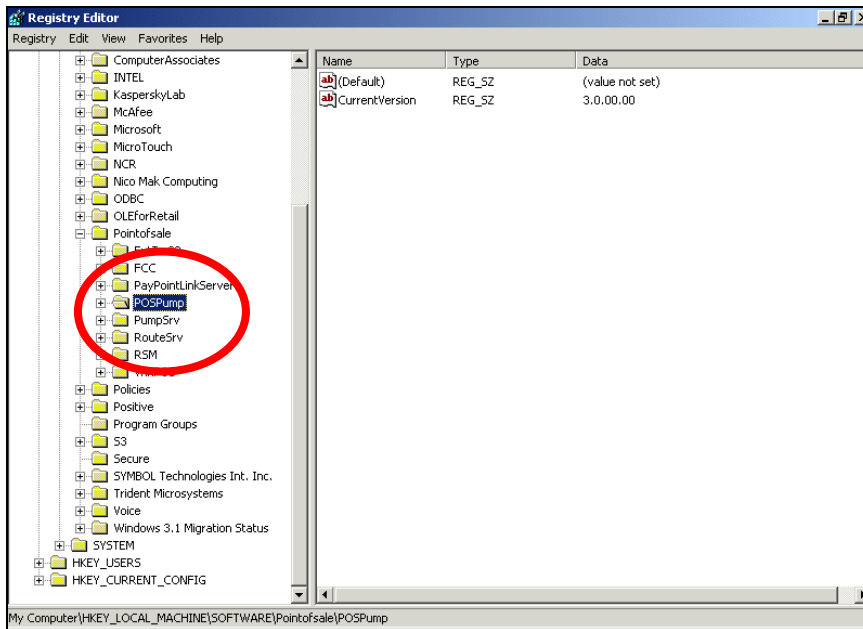
Or

If the POS Terminal identifies that *PumpSrv* is active, the POS terminal will verify that there are no open fuel transactions, and stop the *Site Controller* from opening any new fuel transactions.

Note:

The POS Terminal can verify that there are no open fuel transactions by using the Maint Mode function, which is a *Site Controller* Mode. Not all *Site Controllers* used by the Fuel Stations support the Maint Mode function. During this mode the Fuel Forecourt activity on the POS Terminals is inhibited by the system, and prevents them from opening any new fuel transactions.

The following screen displays the Registry Editor on a Fuel POS Terminal. Note that the *POSPump* and *PumpSrv* folders would not appear on the Registry Editor on a Store POS Terminal:



Upgrade Failure

The upgrade procedure of Fuel POS terminals may fail due to one of the following reasons:

The POS terminal performs RCOPY32 of Wxxxxxxx.EXE to copy the files to `C:\CDPackagesRoot\Wait`. If the RCOPY32 fails to copy one of the relevant packages, one of the following alerts is displayed on the POS Terminal:

- WinPOS packager Error No. 3001 "RCOPY failed: package type W".
- PumpSrv package Error No. 3009 "RCOPY failed: package type S".

The Opener utility opens new packages and performs upgrades when invoked by *PkgAdmin*. If the Opener does not complete the upgrade process, an alert is sent, and the process is stopped. For example, alerts may include:

- WinPOS package Error No.3005 “Opener returned NOK: package type W”,
- PumpSrv package Error No. 3011 “Opener returned NOK: package type S”.

To locate Opener errors, go to the WinPOS.log and open the file: PkgOe.Err. This is a copy of the error log file created in the Temp directory. For more details on PkgOe.err log file, see the section, *Opener Utility, Opener Log Files for MFS Upgrades* in *Chapter 4*.

The POS Terminal reports on the status of Fuel upgrades by transmitting alerts to the MFSs indicating the upgrade status.

If a POS Terminal only supports one of the fuel installations, *PosPump* or *PumpSrv*, it reports by transmitting an alert that the second fuel installation is not needed. Within the alert the function 253 means “Not Supported”.

Note:

If reboot is required, the POS Terminal will wait until all packages are installed. If there is no need to reboot, the WinPosCd.exe will first load:

1. *PumpSrv* components, then
2. WinPOS Services (if needed), and lastly,
3. Run the POS Terminal application.

7

Package Structure

This chapter describes the structure of CDS upgrade packages.

In this chapter:

- Package Structure
- Control File
- Front Office File
- WinPos File
- External Hook Structure

Package Structure

Upgrade files are essentially the *StoreNext ISS45* Installation/Upgrade tool that contains some or all of the Installation files. This upgrade method is based on the principle that the upgrade files inherit the same code as the application's Install tool. In this fashion, the F*.EXE file is based on the Front Office Install file, and the W*.EXE file is based on the WinPos Install file.

CDS Upgrade packages contain the following files:

Control file, Front Office file and WinPos file.

Control File

The Control file comprises 3 sections:

- **General**
- **Stores**
- **Valid Packages**

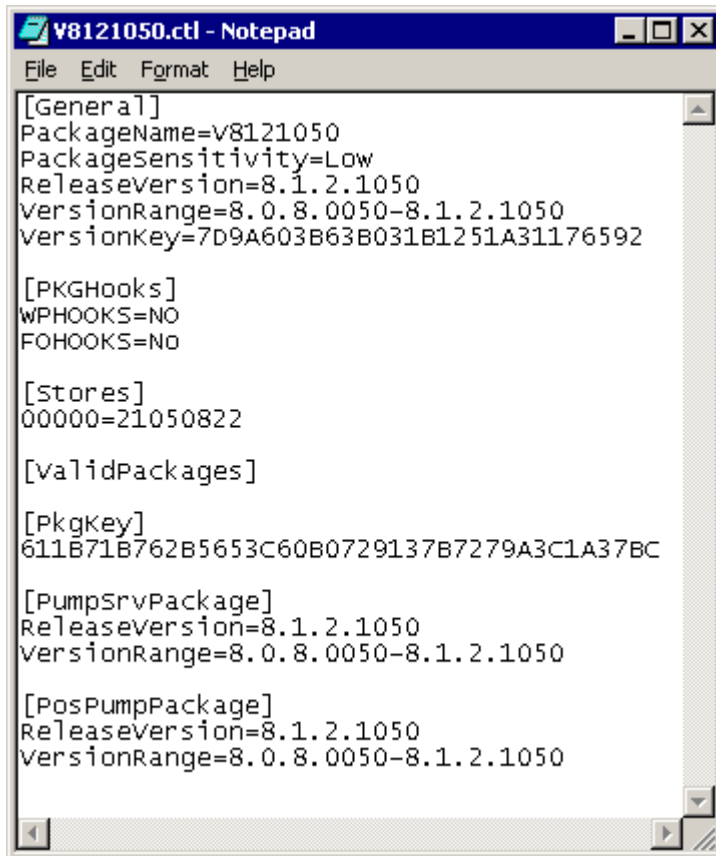
The following displays the format of an exemplary Control file designated for a POS Terminal:

```
[General]
PackageName=V8121050
PackageSensitivity=Low
ReleaseVersion=8.1.2.1050
VersionRange=8.1.2.1000-8.1.2.1050

[Stores]
04321=20030214
00015=20030228

[ValidPackages]
ValidPackages1=v8121050
```

The following displays the format of an exemplary Control file designated for a Fuel system:



```
[General]
PackageName=V8121050
PackageSensitivity=Low
ReleaseVersion=8.1.2.1050
VersionRange=8.0.8.0050-8.1.2.1050
VersionKey=7D9A603B63B031B1251A31176592

[PKGHooks]
WPHOOKS=NO
FOHOOKS=NO

[Stores]
00000=21050822

[ValidPackages]

[PkgKey]
611B71B762B5653C60B0729137B7279A3C1A37BC

[PumpSrvPackage]
ReleaseVersion=8.1.2.1050
VersionRange=8.0.8.0050-8.1.2.1050

[PosPumpPackage]
ReleaseVersion=8.1.2.1050
VersionRange=8.0.8.0050-8.1.2.1050
```

The **General** entry includes package name, package sensitivity level (Low/High), release version and version range. The control file name always has 8 characters, beginning with V, followed by 7 numeric characters and ending with .CTL. The Version Range indicates the range of versions in which the upgrade is possible. If out of the specified range, the package will be rejected.

The **Stores** entry includes the store numbers and indicates the Execution date for each of the stores. When 00000 is entered in the **Stores** entry, this specifies that the package is intended for all stores. The date is entered in either the YYYYMMDD or CCYYMMDD formats. If 00000000 is entered, this specifies that the **Store** will register the package without making validation on the date, and the flag "Ignore Date Validation" is set to on. If 99999999 is entered this specifies that the package will be deleted.

The **ValidPackages** entry indicates the previously installed version. It is not mandatory for this entry to be filled for store packages. In Fuel packages, the Valid Packages entry includes the PFS components, including: package name, release version, and version range

Warning:

In the Control file the only information that may be changed are scheduled dates for the upgrade, including defining the scheduled dates for specific stores. All other information should not be changed, otherwise possibly causing malfunction in the upgrade process.

Front Office File

The **Front Office file** contains the upgrade for the Front Office (V8), and is located on the MFS1 server. The Front Office file name always has 8 characters, beginning with F, followed by 7 numeric characters, and ending with .EXE.

WinPos File

The **WinPOS file** contains the upgrade for the POS terminals. The WinPOS file name always has 8 characters, beginning with W, followed by 7 numeric characters, and ending with .EXE.

In upgrades for Fuel Stations the PFS package is located in the same folder of the WinPOS package. The names of the PFS packages are:

PumpSrv – SXXXXXXXX.EXE.

POSPump – PXXXXXXXX.EXE.

These packages are InstallShield executables like the package of WinPOS – WXXXXXXXX.EXE.

External Hook Structure

This section is designed to provide the rules and regulations for adding external hooks and processes to the CDS distribution mechanism. This is to allow the dealers to update customized files and in house development using the CDS engine.

The composer of the external package should have an understanding of the StoreNext process of CDS and has full responsibility for the operation of the external package.

The standard package structure is as follows (where xxxxxxxx refers to the package version):

- Vxxxxxxx.PKG
- Hxxxxxxx.FO
- Hxxxxxxx.WP
- Vxxxxxxx.CTL (optional if not in the PKG file)

For example: Version 8.1.2.1050 would be V8121050.PKG

Each Hook package will contain the following structure:

FrontOff:

Hxxxxxxx.FO

Hxxxxxxx.zzz.FB0 thru Hxxxxxxx.zzz.FB9 (Dealer add-on hooks before boot – Services Stopped)

Hxxxxxxx.zzz.F00 thru Hxxxxxxx.zzz.F99 (Dealer add-on hooks after boot – Services Running)

WinPOS:

Hxxxxxxx.WP

Hxxxxxxx.zzz.WB0 (Hook file reserved for StoreNext use)

Hxxxxxxx.zzz.WB1 thru Hxxxxxxx.zzz.WB9 (Dealer add-on hooks before WinPOS boot)

Hxxxxxxx.zzz.W00 thru Hxxxxxxx.zzz.W99 (Dealer add-on hooks after WinPOS boot)

Note:

In order for the hooks to be called after the upgrade is complete, there needs to be a root hook used with a .W00 extension.

The **zzz** allows for a sub-set of the **xxxxxxx** hook file. Each of the sub-packages will contain a standard execute batch file and any additional files that will be deployed during the process.

Hook Rules and Regulations for POS Hooks

- Use file numbers. (WB1, WB2, WB3 ... / W01, W02 ...) It is not necessary to use sequential numbers, but the patches will be applied in numerical order.

Note:

WB0 is reserved.

- Do NOT perform any boot during the hook process.
- Do NOT use the "pause" command and any other interaction with a customer, since the customer won't be there and the process will hang.
- Do NOT try to load the services. Services will always be down or closed before executing any of the WBn hooks.
- Do NOT try to start WinPOS on your own. The WinPOS will be rebooted after all the WBn patches have been applied.
- Do NOT launch any asynchronous process from the hook batch file or from sub-processes.
- You should create a back up for any components you are changing in the back up directory which is: C:\Program Files\POSware_Backups\Hxxxxxxx_BAK.WBx. The actual installation path may vary, so you should obtain it from the registry. See the hook example (below). Make sure to add your own "symbol" (WBn) to the back up file so you won't delete other customer's backup file.
- Before creating a back up, delete the previous one.
- Document every step of the hook by writing a log to the file: WinPOS\log\WinPos.log.
- Exit code rule for the hooks. An error level of zero means everything is OK. Any other error level means something failed and the installation process will abort. In this case, THE WINPOS WILL NOT START. Therefore, if the last command is copy or any other system command (like cd, etc), make sure you return "0".
 - The *.WP will be unzipped to:
 \%TEMP%\Hxxxxxxx\
 And the Hxxxxxxx.xyz.wbx will be unzipped to:
 \%TEMP%\Hxxxxxxx\Hxxxxxxx.zzz.wbx_DIR\

Note:

When your install batch is called, you will be located within the directory containing your files.

Order of Execution for POS Hooks

The following is the order in which packages are applied:

- 1 The StoreNext package is first applied. This brings the POS up to the package level.
- 2 This is followed by the WB0 thru WB9 hook files, if they exist.
- 3 Following the reboot of the POS, the W01 thru W99 hook files are applied if they exist.

The format of the batch file name is: Hxxxxxxx.Wnn.bat. For example, H8120052.CCD.WB0 would include:

```
04/11/2007 03:29p      75 H8120052.ccd.ini
10/03/2007 04:54p   6,491 H8120052.wb0.bat
10/15/2007 02:36p  18,332 drv32.zip
10/15/2007 01:25p 158,818 qdx.zip
```

Hook Rules and Regulations Front Office Hooks

- Use file numbers. (FB0, FB1, FB2, FB3 ... / F01, F02 ...) It is not necessary to use sequential numbers, but the patches will be applied in numerical order.
- Do NOT perform any boot during the hook process.
- Do NOT use "pause" command and any other interaction with a customer, since the customer won't be there and the process will be hang.
- Do NOT try to load the services. Services will always be down or closed before executing any of the FBn hooks.
- Do NOT launch any asynchronous process from the hook batch file or from sub processes. You should create a back up for any components you are changing in the back up directory of your choice.

- Document every step of the hook by writing a log to the c:\pcmaster\log-e-of-day.log.
- The *.FO will be unzipped to c:\pcmaster\tmp\Hxxxxxxx and the Hxxxxxxx.xyz.fxx will be unzipped to c:\pcmaster\tmp\Hxxxxxxx\Hxxxxxxx.Fxx_DIR
When your install batch is called, you will be located within the directory containing your files.

Order of Execution for Front Office Hooks

- The StoreNext package is first applied. This brings the MFS\LFS up to date.
- On V8 only, Fuel is then run.
- This is followed by the FB0 thru FB9 hook files, if they exist.
- MFS\LFS is rebooted.
- Services started.
- Run F00 thru F99 hook files, if they exist.

Format of the Front Office batch file name is: Hxxxxxxx.Fnn.bat

In this example, the components are all included in a single file, with the pathnames stored in the zip file.

Example of H8121050.ABC.FB0

10/03/2007	04:54p	6,491	H8121050.FB0.bat
10/15/2007	02:36p	18,332	patch.zip

Distribution of Hook Files

Each dealer will be assigned an FTP site where dealer add-ons will be loaded. The Connected Services applications would be looking at this directory to allow them to select these add-ons for deployment to the store along with the respective software upgrade/patch.

WinPOS Hook Example:

```
@echo off

rem set the following to the update version
set ToRev=H8121050
rem WB1 is used in the following example. WB0 is restricted.

rem get home directory from registry

puGetReg SOFTWARE\PointOfSale\WinPOS\Paths\Short HomeDir "set AppPath=$"
"%temp%\~PreBatch.bat"
if errorlevel 1 goto BAD_REG_ERR
call "%temp%\~PreBatch.bat"
del "%temp%\~PreBatch.bat"

rem set location of the winpos.log

set MYLOG=%AppPath%\log\winpos.log

puEcoNow WB1: ***** Start of %ToRev% Installation >> %MYLOG%

rem when you get control you will be in %temp%\Hxxxxxxx\Hxxxxxxx.WBx_dir

rem back up any files that will be modified to posware\_backups\Hxxxxxxx
puEcoNow WB1: Backing up files that will be over written >> %MYLOG%

rem Delete any previous saved files
if exist %AppPath%..\_backups\H*.WB1.zip del %AppPath%..\_backups\H*.WB1.zip

dzip -q %AppPath%..\_backups%\ToRev%_BAK.WB1 %AppPath%\qdx\posmenu.qdx > nul
dzip -q %AppPath%..\_backups%\ToRev%_BAK.WB1 %AppPath%\qdx\posmenu_gal.qdx > nul
dzip -q %AppPath%..\_backups%\ToRev%_BAK.WB1 %AppPath%\drv32\wpeft.dll > nul
dzip -q %AppPath%..\_backups%\ToRev%_BAK.WB1 %AppPath%\drv32\wphook.dll > nul
```

PACKAGE STRUCTURE

```
puEcoNow WB1: Unzipping drv32 files >> %MYLOG%
dunzip -o -q drv32.zip %AppPath%\drv32 >> nul
if ErrorLevel 1 goto :eUnZip
puEcoNow WB1: drv32 files successfully unzipped >> %MYLOG%
```

```
puEcoNow WB1: Unzipping qdx files >> %MYLOG%
dunzip -o -q qdx.zip %AppPath%\qdx >> nul
if errorlevel 1 goto :eUnZip
puEcoNow WB1: qdx files successfully unzipped >> %MYLOG%
goto GOOD_EXIT
```

```
:BAD_REG_ERR
puEcoNow WB1: Cannot obtain home address from registry >> %MYLOG%
goto BAD_EXIT
```

```
:eUnZip
puEcoNow WB1: Error trying to unzip files >> %MYLOG%
goto BAD_EXIT
```

```
:BAD_EXIT
puEcoNow WB1: ***** Installation aborted!!! >> %MYLOG%
rem force error level to one
gmcdsutl /lvl 1
goto CLEAROUT
```

```
:GOOD_EXIT
puEcoNow WB1: ***** Successful completion >> %MYLOG%
rem force error level to zero
rem NOTE: gmcdsut is a simple utility used for multiple things and is here for example
only.
rem Use whatever means you find handy to accomplish the level setting.
gmcdsutl /lvl 0
```

```
:CLEAROUT
set ToRev=
set AppPath=
set MYLOG=
```

Front Office Hook Example:

```
@echo off
rem in this example, the pathnames are stored in the zip file.
```

PACKAGE STRUCTURE

```
rem NOTE: dunzip automatically restores/creates directory structure stored in the
zip file.
rem This would be equivalent to: PKUNZIP -d

rem This version is for V7. For V8, you should use pcmlog32 instead of printed to
log progress.
rem Refer to the e-of-day.bat file for logging examples in either V7 or V8.

rem set the following to the update version
set ToRev=H8121050

rem set location of the winpos.log
set MYLOG=c:\pcmaster\log\winpos.log

printeod FB0: ***** Start of %ToRev% Installation >> %MYLOG%

rem when you get control you will be in c:\pcmaster\tmp\Hxxxxxxx\Hxxxxxxx.WBx_dir

rem back up any files that will be modified to posware\_backups\Hxxxxxxx
printeod Backing up files that will be over written >> %MYLOG%

rem Delete any previous saved files
rem dzip xxxx.zip file.exe

printeod FB0: Unzipping the patches >> %MYLOG%
copy patch.zip c:\
puchgdir c:\ /SetTag MYDIR
dunzip -o -q patch.zip >> nul
if ErrorLevel 1 goto :eUnZip
del patch.zip
printeod FB0: patch files successfully unzipped >> %MYLOG%
printeod FB0: ***** Successful completion >> %MYLOG%
goto CLEAROUT

:eUnZip
del patch.zip
printeod FB0: Error trying to unzip files >> %MYLOG%
printeod FB0: ***** Installation aborted!!! >> %MYLOG%

:CLEAROUT
    puchgdir /GoTag MYDIR
    set ToRev=
    set MYLOG=
```


8

In this chapter:

Alerts

This chapter describes the alert messages related to Code Distribution.

- Overview
- Setup
- Alerts for Head Office
- Structure of a General Alert for CDS
- Viewing CDS Alerts in QDX files
- Viewing CDS Alerts in the POSTT Application
- Viewing MFSS Statuses in the CDS Process
- Connected Services CDS Alert Tracking/Monitoring

Overview

An alert is an informative message transmitted when an action or process were performed in the system.

The POS terminals, *RegistrPack* and *PkgAdmin* transmit alerts for various actions that occur in the upgrade process, in each of these components. These alerts are transmitted to the Transact.QDX files.

There is a system function called *TranRule*, which identifies all transactions, including alerts, received from the entire system. *TranRule* then sorts the alerts, placing them in their respective tables. *Code Distribution* alerts, represented by op-code 63 Hex, are placed in **AlertTran** table.

V8AlertSrv.EXE, another system function, located in the Status bar, and represented by this icon:  Its task is monitoring alerts in the *SQL* tables.

When an alert is recorded in *AlertTran*, *V8AlertSrv.EXE* identifies it, and responds by performing various actions as described in this chapter.

Each alert, whether it reports success or error, is represented by a numeric code. Codes within pre-assigned code ranges represent alerts for each system component (refer to section *Viewing Code Distribution Alerts in the QDX files* in this chapter).

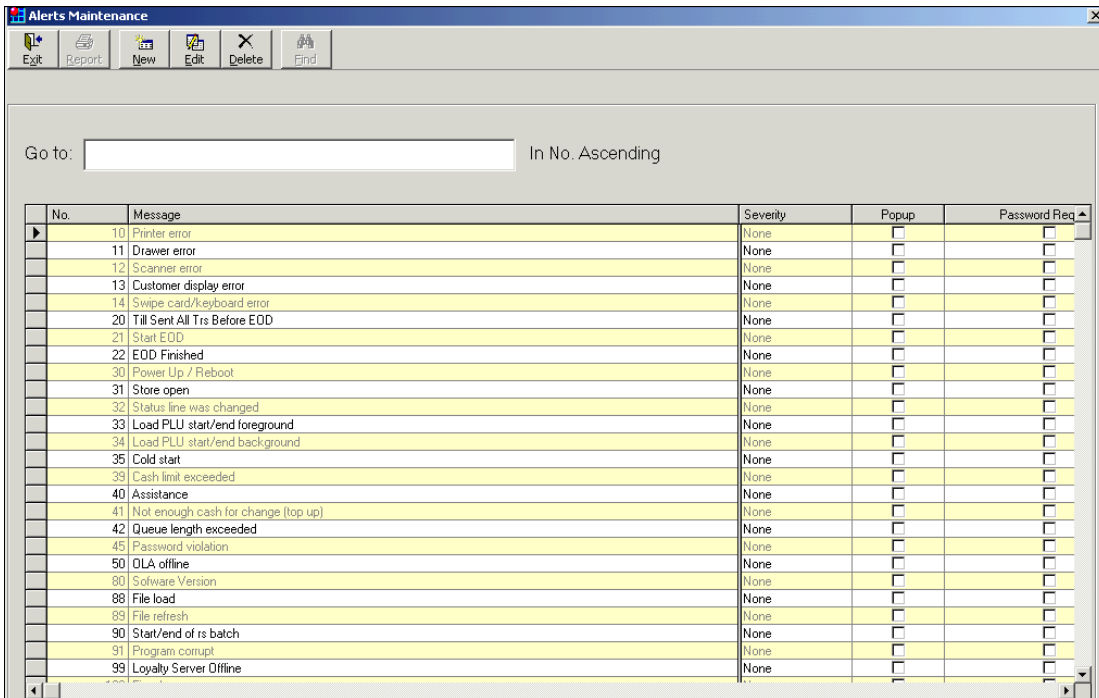
Setup

In order to view alert messages, they must first be configured in your system, in the *Alerts Maintenance* dialog.

The *Alerts Maintenance* dialog enables you to define and modify the text of an alert message, define its severity level, and specify where the alert will be displayed. You can specify that the alert will be displayed in the form of a Pop-up message; or appear in the *NT Event Viewer*, or as otherwise configured.

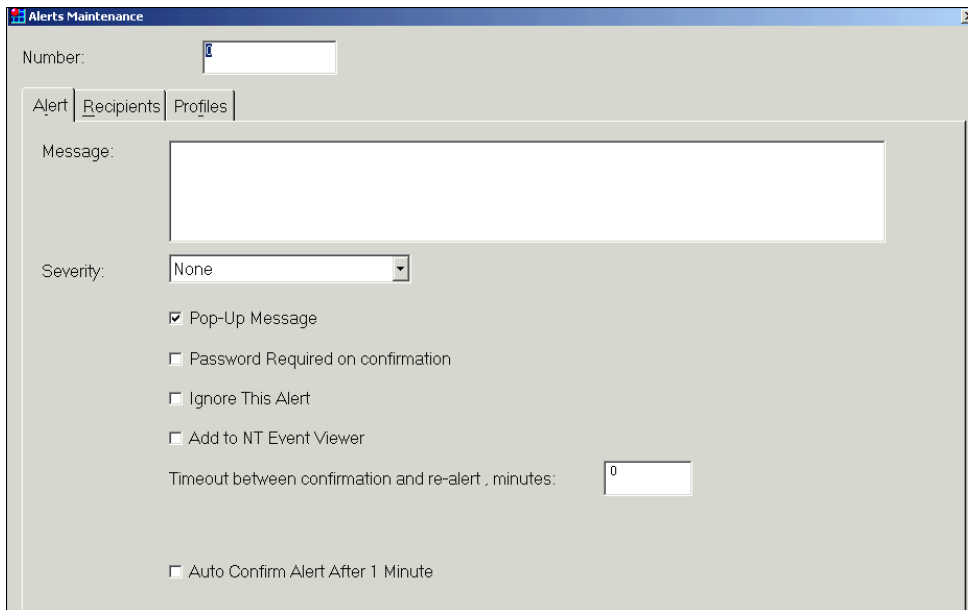
ALERTS➤ **To Define v8 Alerts:**

- 1 In the *Main Menu* tab, browse to the following directory:
Utilities\System Alerts\Alert Maintenance. The *Alert Maintenance* dialog is displayed:



- 2 From the *Alerts Maintenance* dialog, click **New**. The *Alert* tab is displayed:

ALERTS

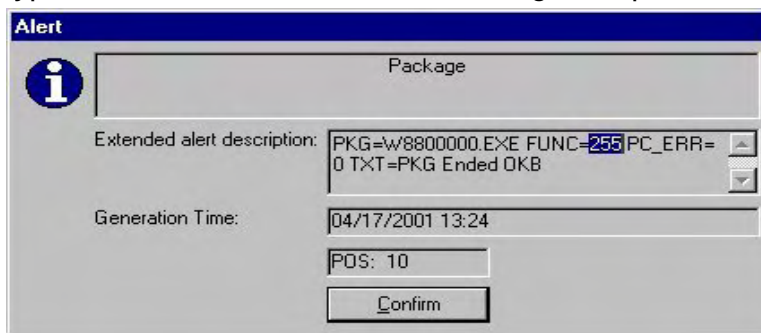


The Alerts Maintenance dialog box is shown with the following fields and options:

- Number: []
- Alert | Recipients | Profiles (tabs)
- Message: []
- Severity: [None]
- ☒ Pop-Up Message
- ☐ Password Required on confirmation
- ☐ Ignore This Alert
- ☐ Add to NT Event Viewer
- Timeout between confirmation and re-alert, minutes: [0]
- ☐ Auto Confirm Alert After 1 Minute

- 3 In the Message field enter the relevant alert type; then check the relevant boxes defining where the alert will be displayed.

For example, if you select the **Pop-Up Message** checkbox, an alert message will be displayed on the MFS1 server each time an alert of this type is issued, as shown in the following example:



The Alert dialog box is shown with the following fields and options:

- Package: []
- Extended alert description: PKG=w8800000.EXE FUNC=255 PC_ERR=0 TXT=PKG Ended OKB
- Generation Time: [04/17/2001 13:24]
- POS: 10
- Confirm

Or

If you select the **Add to NT Event Viewer** checkbox, all alerts of this type are logged in the system, and can be viewed in the *Event Viewer*.

Alerts for Head Office

In order to differentiate specific problems with function components of the *CDS* process, a separate alert ID for each function is defined. In this way, the originating computer name is stored in the Event Log User Name field, for each alert that is sent to the Head Office. This provides the option of monitoring the *CDS* mechanism from a remote location, alerting the system during *CDS* when a PC fails to register or run a package.

During Registration of a Package

If a PC raises an alert number 222 (general alert for *CDS*), it will also raise one of the alerts shown in the table below, e.g., alert number 226 indicating 'PC Registration failed'. Each alert includes the package name and details.

During Running of a Package

If a PC raises an alert number 222 (general alert for *CDS*), it will also raise one of the alerts shown in the table below, e.g., alert number 223 indicating 'Package upgrade failed'. Each alert includes the package name and details.

The table below displays the various alert types for Head Office:

Alert Number	Meaning
223	Package upgrade failed.
224	Package upgrade ended.
225	Package registration success.
226	Package registration failure.

Structure of a General Alert for CDS

The following is the structure of the General Alert 0x63:

struct trs_alert_pkg_

```
{
    unsigned char opcode; // 0x63 //0
    unsigned short subopcode;// 0xDE – E2
//1
    unsigned char function; //3
    unsigned char disp_message :1; //4
    unsigned char drvfile_popup :1;
    unsigned char ignore_alert :1;
    unsigned char use_pos_no :1;
    unsigned char get_text_msg :1;
    unsigned char manual_pkg :1;
```

Subopcode (Alert Number)	
Hex	Decimal
DE	222
DF	223
E0	224
E1	225
E2	226

ALERTS

```
unsigned char create_by_app :1;
unsigned char nu07 :1;
unsigned char pkg_name[12]; //5
unsigned char data[14]; //17
unsigned char opener_result; //31
unsigned short prv_rel_a; //32
unsigned short prv_rel_b; //34
unsigned short prv_rel_c; //36
unsigned short prv_rel_d; //38
unsigned short error; //40
unsigned char pos_no; //42
unsigned char pc_no; //43
struct trans_tail_tail; //44
};
```

Viewing CDS Alerts in QDX files

All alerts transmitted within the system appear in the QDX as op-code 63 (Hex).

This is followed by a sub op-code 222 decimal, (DE Hex), which indicates a CDS alert.

This is followed by an additional code that represents the type of alert that was generated. The code range for this field is 1-255.

The table below displays exemplary sub op-codes of the various alert types for CDS (up to sub op-code 255):

Sub-Code Range	Source	Meaning
0		Not Started
1	POS terminals & PkgAdmin	Started
2	PkgAdmin	Boot
3	PkgAdmin	Srv Started
30	RegistrPack	The ' Delete ' package has successfully completed version registry.
31	RegistrPack	The ' New ' package has successfully completed version registry.
32	RegistrPack	The ' Edit ' (.CTL file) package has successfully completed version registry.
33	RegistrPack	The ' Edit ' (All Files) package has successfully completed version registry.
34	RegistrPack	The ' Edit ' package has successfully completed version registry.
238	POS terminals, PkgAdmin & RegistrPack	Error
250		Package Synchronize
251		System Start
252		Abort CDS

ALERTS

253		Not Support
254	RegistrPack	System End
255	POS terminals & PkgAdmin	End

The remainder of the codes within this range represent miscellaneous Success alerts.

The next field is relevant only in the event of an error, i.e., sub code=238. In this case the Error number specifies the Error type, according to pre-defined ranges for each *Code Distribution* component.

Monitoring Errors & Messages in the Upgrade Process

At the end of the upgrade process the administrator locates possible errors that occurred in the MFSs or the POS terminals. This is done via *Package Maintenance* (for further details refer to *Chapter 4*). Once an error is identified, its numeric code can be located in the table below.

The table below describes the possible codes indicating specific types of errors (Sub-code 238) or messages that have occurred in the upgrade process:

PkgAdmin		
Error Number	Meaning	Type
5	Connection Failed between ALRT_TRAN & TRAN_ID.	
6	Connection Accepted between ALRT_TRAN table & TRAN_ID.	
7	Connection Failed in ALRT_TRAN.	
8	Connection Accepted in ALRT_TRAN.	
10	No Records in VER_DATA.	

ALERTS

PkgAdmin		
Error Number	Meaning	Type
11	Connection Failed in VER_DATA.	
12	Connection Accepted in VER_DATA.	
13	Package System Status – Error.	
14	Package System Status – Not Completed.	
15	Package Wrong Date Format.	
20	VER_FILES No Record.	
21	VER_FILES Connection Failed.	
22	VER_FILES Accepted.	
30	VER_STATUS No Record.	
31	VER_STATUS Connection Failed.	
32	VER_STATUS Accepted.	
33	VER_STATUS Records Exist.	
41	Package Path Does Not Exist.	
42	Package File Does Not Exist.	
43	Path Length Too Long.	

ALERTS

PkgAdmin		
Error Number	Meaning	Type
44	Cannot Create Dos Path File.	
45	Cannot Process; Command Failed.	
49	Rcopy Result=Wrong Value.	
50	Rcopy Result=Communication failed; Driver Not Loaded.	
51	Rcopy Result=File Not Found.	
52	Rcopy Result = 3	
53	Rcopy Result=Files had the same date/time /checksum as SDT/NDT/QCHKxx switches set.	
54	Rcopy Result=Timeout End; Rcopy not completed.	
55	Rcopy Result= Rcopy not completed.	
61	Cannot Create Dos Batch File.	
62	Create Process Command failed.	
65	Package Registration Root Open Failed.	
66	Package Registration Root Query.	

ALERTS

PkgAdmin		
Error Number	Meaning	Type
67	Package Registration Root Field Empty.	
70	Create Opener Base Failed.	
71	Registry 'In' key Opener Base already Empty.	
72	Registry 'Out' key Opener Base already Empty.	
75	Cannot LoadCLN32DLL	
76	Cannot Get Address PTRANS.	
77	PTRANS Failed.	
81	Path Does Not Exist.	
82	Path Length too Long.	
83	Version Length Name too Long.	
84	Pmaint Command Failed.	
85	Path Length Too Long.	
86	Package Name Length Too Long.	
87	POS package – OS type Windows not found	
88	POS Package – OS type Linux not found	
90	File Not Found.	
95	Path Not Found.	

ALERTS

PkgAdmin		
Error Number	Meaning	Type
99	Database OK; No need to run package.	
100	Path Length Too Long.	
101	Path Not Found.	
105	Create Directory Failed.	
106	Remove Directory Failed.	
107	Copy API File Failed.	
110	Cannot Create Status Dos File.	
120	SYS_PRM Connection Failed.	
121	SYS_PRM Accepted.	
122	EOD Field in SYS_PRM is Empty.	
123	Field in SYS_PRM is Empty.	
130	Date Format Accepted.	
131	Date Format Field Empty.	
140	RelVer (Release Version) Field Empty.	
141	RelVer Field Overflow (Also Check Current Registry Version).	
142	RelVer Field Illegal Format a.b.c.d. Numof Dots more Than 3 ((Also Check Current Registry Version).	

ALERTS

PkgAdmin		
Error Number	Meaning	Type
143	RelVer Field Illegal Format a.b.c.d. Numof Dots Less Than 3 ((Also Check Current Registry Version).	
150	SQL RelVer (SQL Records)/ RegRelVer(In Registry) Field Empty.	
151	SQLRelVer (SQL records) is equal or bigger than RegRelVer (in registry)	
152	SQLRelVer (SQL records) is less than RegRelVer (in registry)	
160	Registry Current Version Open Fail.	
161	Registry Current Version Query.	
162	Registry Current Version Field Empty.	
170	Reg (Registry) GlobalStore45 HomeDir Open Failed.	
171	Reg GlobalStore45 HomeDir Query.	
172	Reg GlobalStore45 HomeDir Field Empty.	
180	Create 'In' Key Opener Base Failed.	

ALERTS

PkgAdmin		
Error Number	Meaning	Type
181	RegSetValueEx Field SkipReboot Failed.	
182	RegSetValueEx Field JustChecked Failed.	
183	RegSetValueEx Field StartWork Failed.	
190	Opener Did Not Sign Started.	
191	Opener Did Not Sign Ended.	
210	Directory Not Found.	
215	Table Connection Failed.	
216	Table Accepted.	
220	RSM Not Running.	
221	GetDiskFreeSpace Failed.	
222	Disk Size Limit.	
225	Cannot Read .CTL Dos File.	
226	.CTL File Valid Packages Section Wrong.	
227	.CTL File Execute SQL Command Failed. (Checking Package in VER_DATA)	
250	Bad Synchronize Mode Before Running Upgrade Failed.	

ALERTS

PkgAdmin		
Error Number	Meaning	Type
251	Bad Synchronize Mode Before Ending Upgrade Failed.	
252	Bad Synchronize Mode Before Performing Boot Failed.	
260	Opener Failed.	
261	Failed to get blackout times from DB	

RegistrPack & Package Maintenance		
Error Number	Meaning	
1987	V7 unknown error	
1988	V7 Winpos hook file missing	
1989	V7 Back office hook file missing	
1990	V7 Winpos File Missing	
1991	V7 Back Office File Missing	
1992	V7 Qdx error	
1993	V7 Internal error	
1994	V7 Command failed error	
1995	V7 No valid bases error	
1996	V7 Package in Progress error	

ALERTS

RegistrPack & Package Maintenance		
Error Number	Meaning	
1998	V7 Extract package error	
2000	V7 Package Status error, can't find POS package	
2001	PoS Terminal XX did not upgrade.	
2002	MFS XX did not upgrade.	
2003	Zip Error	
2004	Package Rejected: Not Enough Space on Hard Drive.	
2005	Package Rejected: Unzipping Package File Failed.	
2006	.CTL Version Value is Not Equal to File Value.	
2007	Package Sensitivity Value in .CTL File is Not Equal to 'Low' or 'High'.	
2008	New Package & Edit All Files: If both W & F files exist, rejects and deletes and deletes from Tables.	
2009	.CTL File Missing.	
2010	RegEdit Key Missing: SOFTWARE\Pointofsale\CodeDistribution\Paths\Long\PackagesRoot. Manually Move V*.PKG from 'New' to 'Reject'.	

RegistrPack & Package Maintenance		
Error Number	Meaning	
2011	RegEdit Key Missing: SOFTWARE\POINTOFSAL E\GLOBAL45\ODBC\Con nect	
2012	Since CDPackagesRoot was just Created now, there is no point to proceed.	
2013	Since New folder was just created now, there is no point to proceed.	
2014	There are no PKG files in New folder.	
2015	Reject Package: Key/Value in /CTL is Not Valid.	
2016	Reject Package: Release Version Value in the .CTL is Empty.	
2017	Reject Package: Release Version Value in .CTL file contains more than 3 dots.	
2018	Reject Package: Release Version Value in .CTL file is not valid.	
2019	Package is dependent on other package which is missing from VER_DATA table.	
2020	Error: Reject Package: The Execution date in the .CTL file is not valid.	

ALERTS

RegistrPack & Package Maintenance		
Error Number	Meaning	
2021	Error: Reject Package: The Execution date in the .CTL file is smaller than last working date.	
2022	In VER_DATA table there is a record with PKG_VER which is bigger than the New Package.	
2023	The next package in SYS_STATUS is other than Not Started.	
2024	Error: Reject Package: The package is not valid.	
2025	After 10 minutes the Stage value is not ended as it should be.	
2026	Registry value of SOFTWARE\Pointofsale\GlobalStore45\Install\PackageUpgrade\Out\Ended when the package failed.	
2027	Failed to delete the PackageUpgrade key from the registry.	
2028	Failed to Insert value 'Yes' to the value name 'StartWork'.	
2029	Failed to insert value 'Yes' to the Value name 'JustCheck'.	

ALERTS

RegistrPack & Package Maintenance		
Error Number	Meaning	
2030	Failed to run the Opener.	
2031	Failed to Delete the In key from the Registry.	
2032	The Value name 'Stage' was not updated with the Started value during two minutes.	
2033	If 'Wait' folder is empty, no need to copy .CTL file into it.	
2034	If both W*.EXE & F*.EXE do not exist, reject & delete from tables.	
2035	Front Office Hook File Missing	
2036	WinPOS hook file missing	
2037	Failed to find valid package key	
2974	Front Office Hook after boot failed	
2975	Front Office Hook before boot failed	
2976	CDS did not run today because of blackout times	
2977	Version Key in Control file is not legal	
2978	No Release Version in control file	

ALERTS

RegistrPack & Package Maintenance		
Error Number	Meaning	
2979	No Version Key in control file	
2980	V7 Back Office Hook is damaged	
2981	V7 POS Hook is damaged	
2982	V7 Previous package not ended	
2983	V7 Unable to read HASP plug ID	
2984	V7 Unable to read HASP plug data	
2985	V7 HASP plug is missing	
2986	V7 Package Key doesn't match	
2987	V7 Package Key section missing	
2988	V7 Bad File	
2989	V7 No Package File	
2991	V7 Date Amended	
2992	V7 Control File missing line	
2993	V7 Syntax error in Control file	
2994	V7 No Store Number	
2995	V7 Duplicate Control File	
2996	V7 High Not Start	

RegistrPack & Package Maintenance		
Error Number	Meaning	
2997	V7 Can't mirror	
2998	V7 Not deleted	
2999	V7 Already installed	
3000	V7 Registered Bad	

ALERTS

POS Terminals		
Error Number	Meaning	
3001	Rcopy Failed Package Type W	
3002	Error Reading Registry.	
3003	First Version Check Failed. (Compare CDS Small Maintenance Release Version against Registry Current Version)	
3004	Second Version Check Failed. (Opener with mode justcheck: Compare package release version against Registry Current Version)	
3005	Opener Returned Not OK: Package Type W.	
3006	Rcopy Process Stopped.	
3007	End Flag Not Found (Should be Created by WinPosCD.EXE	
3008	Disk Space Lower Than Limit.	
3009	Rcopy failed Package Type S.	
3010	Rcopy failed package Type P.	
3011	Opener returned NOK: Package Type S.	

ALERTS

POS Terminals		
Error Number	Meaning	
3012	Opener returned NOK: Package Type P.	
3020	Can't Find Package Hook File	
3021	Zip file is damaged	
3022	Process Failed on WaitForSingleObject	
3023	Process execution failed	
3024	WinPosCD created a BAD flag	
3025	Upgrade failed without a specific reason	

ALERTS

An additional field displays a brief text describing the type of alert, followed by the POS terminal number or MFS number from where the alert was sent.

Viewing this table enables you to determine in which system component the error occurred. If the error occurred in an MFS, browse to the following log file:

C:\pcmaster\Log\Pos_Srv.Log to locate the specific error, and to obtain additional information on the error, its location, package number etc.

If an error occurred on a POS terminal, it can be viewed in the following log file:

Programs\Files\Posware\WinPos\Log\WinPos.log. (For further details see section on *Upgrading the POS terminals* in *Chapter 4*).

Viewing CDS Alerts in the POSTT Application

POSTT is an application that reflects all the transactions, as they appear in Transact.QDX. This application enables you to view all alerts (op-code 63 Hex, 99 Decimal), CDS alerts included (DE Hex, 222 Decimal).

Note:

The *TT application* is available in both Windows-based and Console versions. Both versions support the same functions, and both are installed with the entire Installation package.

➤ **To View v8 Alerts Via the TT Application:**

- 1 From the Windows *Start* menu, select *Run*.
- 2 In *Run* Dialog, enter **cmd** and press **OK**.
- 3 In the *Command Window*, enter **CD:\Program Files\POSware\Office\Components\EXE** and press **Enter**. The *Search Criteria* dialog is displayed as follows:

ALERTS

Search Criteria

Transaction file: C:\PCMaster\QDX\Transact.qdx

Read from 0 to 999999999 **Load** **Reset** **Save**

☒ All ☐ Offset search ☒ Advanced search ☐ Field search

Offset search

Offset: ☐ Bit: ☐ And ☐ Or ☐ XAnd ☐ XOr

Length: String: + -

Advanced Search

☐ Enable ☐ Enable ☐ Enable

Opcode **Cashier** **Pos**

+ - + - + -

☒ In ☐ Ex ☒ In ☐ Ex ☒ In ☐ Ex

☐ Enable ☐ Enable ☐ Enable ☐ Enable

Void Ticket **Bad Record** **Training Mode** **Pos Offline**

☒ In ☐ Ex ☒ In ☐ Ex ☒ In ☐ Ex ☒ In ☐ Ex

☐ And ☐ Or ☐ And ☐ Or ☐ And ☐ Or ☐ And ☐ Or

Field Search

Condition: ☒ And ☐ Or

Opcode:

Offset:

Type: BYTE(1)

Operator: >

Value:

+ -

Ok

- 4 Click the Advanced Search radio button.
- 5 Click the **Enable** checkbox for Opcode.
- 6 Enter **63de** in the first Opcode field.
- 7 Click **OK**. The result records are displayed.
- 8 Click the **Translated** button. The translated data displays:

PosTT - [C:\PCMaster\QDX\Transact.qdx]

File View Search Window

Records: 2 **Refresh** ☐ Hex ☒ **Translated** ☐ Fix columns <- Ticket > Line

P.O T.O

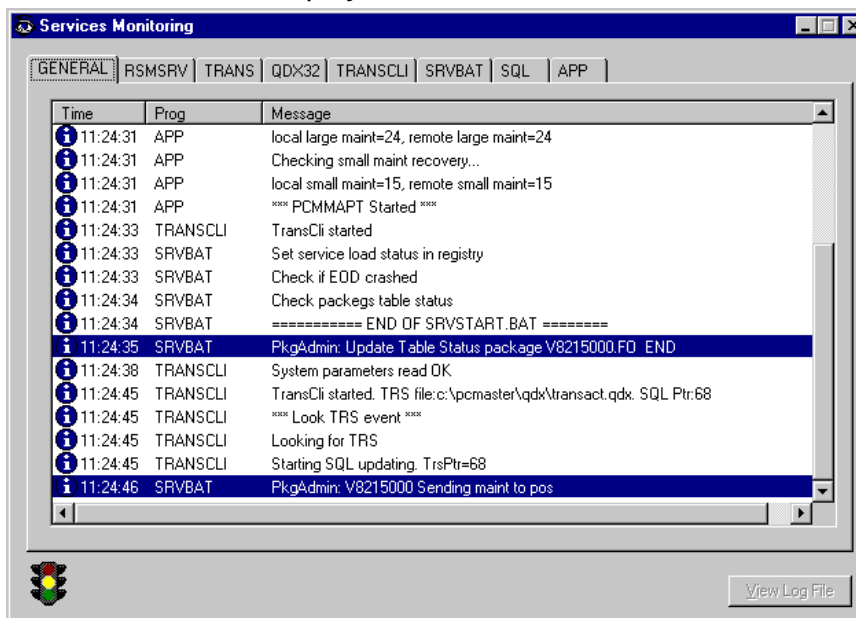
Rec	Pos	Tick	Opc.	TrsData
0	241	0	63	ALERT: PACKAGE ERROR ErrNo:2009 FILE TYPE: V8121035 PC:01 TXT:Reg: Reject Pa
1	241	0	63	ALERT:

Viewing MFSs Statuses in the CDS Process

You can view various statuses in the MFSs using the *Monitoring Utility*.

➤ **To View v8 Statuses of MFSs:**

- 1 Double click the *Services Monitoring* Utility located in the Status Bar. The *General* tab is displayed as follows:



The *Services Monitoring* dialog lists all the messages that are found in the system. The selected messages above apply specifically to *Code Distribution*.

Note:

All historical records of this utility for CDS are saved in a log file in:
C:\pcmaster\Log\Pos_Srv.log.

Connected Services CDS Alert Tracking/Monitoring

The Connected Services CDS Alert Tracking screens are designed to reflect the various *Code Distribution Maintenance* activities and provide the administrator with the necessary insight and statuses regarding the upgrades. Both server and store software log their activities constantly and provide this “feedback” to the Alerts system, for the administrator to review.

➤ **To track and monitor package distribution:**

- 1 Logon to StoreNext Connected Services website and select the *Alert Tracking* option from the menu. *The Alerts Mechanism screen displays outstanding ERRORS identified by the logging process within the previous week for the stores managed by the Administrator:*

Store ID	Store Name	Alert Type	Alert Description	Alert Status	Alert Date and Time
LabStore1	LabStore1 - Sunnyvale	CDS Distribution	Package Distribution Ended	Ended	06/18/2003 15:30:04
LabStore1	LabStore1 - Sunnyvale	CDS Distribution	Package Distribution Started	Started	06/18/2003 15:30:00
LabStore1	LabStore1 - Sunnyvale	CDS Distribution	Package Distribution Ended	Ended	06/18/2003 14:00:05
LabStore1	LabStore1 - Sunnyvale	CDS Distribution	Package Distribution Started	Started	06/18/2003 14:00:02
LabStore1	LabStore1 - Sunnyvale Lab	CDS Distribution	Package Distribution Error	Error	06/17/2003 17:50:01
LabStore1	LabStore1 - Sunnyvale Lab	CDS Distribution	Package Distribution Started	Started	06/17/2003 17:50:01
LabStore1	LabStore1 - Sunnyvale	CDS Distribution	Package Distribution Ended	Ended	06/17/2003 17:00:06
LabStore1	LabStore1 - Sunnyvale	CDS Distribution	Package Distribution Started	Started	06/17/2003 17:00:01
LabStore1	LabStore1 - Sunnyvale	CDS Distribution	Package Distribution Ended	Ended	06/17/2003 13:10:04
LabStore1	LabStore1 - Sunnyvale	CDS Distribution	Package Distribution Started	Started	06/17/2003 13:10:00
LabStore1	LabStore1 - Sunnyvale	CDS Distribution	Package Distribution Ended	Ended	06/13/2003 16:40:00
LabStore1	LabStore1 - Sunnyvale	CDS Distribution	Package Distribution Started	Started	06/13/2003 16:40:00
LabStore1	LabStore1 - Sunnyvale	CDS Distribution	Package Distribution Ended	Ended	06/11/2003 09:42:09
LabStore1	LabStore1 - Sunnyvale	CDS Distribution	Package Distribution Started	Started	06/11/2003 09:42:05

- 2 (Optional) Click the Filter button to reduce the alert list by the following:
 - **Date/Time Range:** Start and End date & time.
 - **Store Selection:** Default is all stores.
 - **Component Selection:** Currently only supports *Code Distribution*. This Alert mechanism will support additional components in future versions.
 - **Alert Selection:** Select the specific Alert you wish to view. Currently supports:

ALERTS

- **Package Distribution** – The action of distributing a package to the stores. Options: *Started, Ended, Error*.
 - **Package Activation** – The actual feedback from the store regarding the upgrade itself. Options: *Started, Ended, Warning, Critical Error*.
 - **Alert Severity:** Each Alert type has a severity level. The current options are: *Informative, Warning, Error* and *Critical Error*.
 - **Alert Status:** Each Alert has a status: The options are *Started, Updated, Error* and *Ended*.
- 3 Select a specific record and view the details for that record. This can be done by double-clicking the desired record or highlighting it and selecting the *View Details* button in the top tool bar. The Alerts Detail dialog window displays:

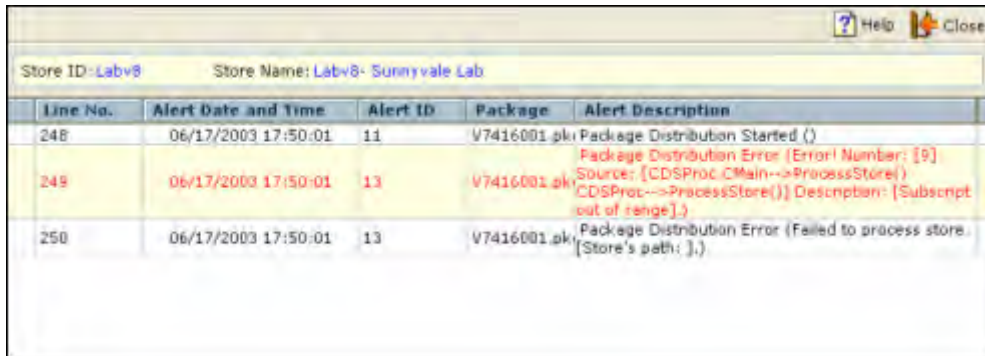
Store ID	Labv8	Store Name	Labv8
Component Name	CDS	Alert Type	CDS Distribution
Alert Date and Time	06/17/2003 17:50:01	Log Line Number	249.00
Alert Status	Error		
Alert Description Package Distribution Error			

Note:

While in the View Details option, you can use the *Arrow* buttons to scroll up/down between the various alerts that are displayed in the grid.

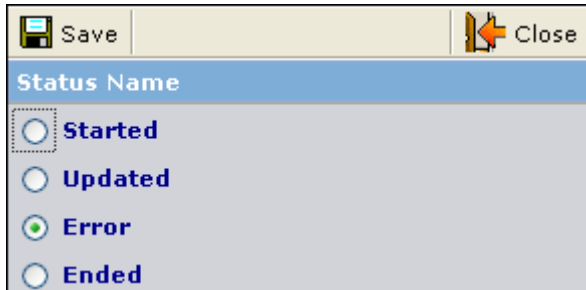
- 4 (Optional) A more detailed view of a particular Alert with the complete step-by-step processes that occurred during that session is facilitated through the *View Log* option. Highlight the specific Alert in the grid and click the *View Log* button. The Log Details dialog window displays:

ALERTS



Line No.	Alert Date and Time	Alert ID	Package	Alert Description
248	06/17/2003 17:50:01	11	V7416001.pk	Package Distribution Started (.)
249	06/17/2003 17:50:01	13	V7416001.pk	Package Distribution Error (Error! Number: [9] Source: {CDSProc.CMain-->ProcessStore} CDSProc-->ProcessStore() Description: {Subscript out of range}.)
250	06/17/2003 17:50:01	13	V7416001.pk	Package Distribution Error (Failed to process store. {Store's path: },.)

- 5 (Optional) Click the **Change Status** button to view the update an error record once the error has been addressed thus eliminating it from re-appearing again when the screen is loaded. Change the status on the Change Status dialog window and click **Save**.



Save Close

Status Name

☐ Started

☐ Updated

☒ Error

☐ Ended

9

Maintenance Structure

This chapter describes the structure of a CDS small Maintenance.

In this chapter:

- Small Maintenance Structure

Small Maintenance Structure

The following is the structure of a small Maintenance; this maintenance is sent by MFS1 to the POSTs, prompting them to upgrade.

Small Maint for CDS 0x98

```
struct mnt_pkg_           //OFS
{
    unsigned char opcode;   //0 0x98
    unsigned char pos_no;   //1 0 - all
        unsigned char func ;   //2 Func 1 - Low sensitivity
                        //      2 - High sensitivity
                        //      3 - Abort CDS
                        //      4 - Force

    unsigned short rel_a;    //3
    unsigned short rel_b;    //5
    unsigned short rel_c;    //7
    unsigned short rel_d;    //9

    unsigned char pkg_loc[80]; //11

    unsigned char maint_time[3]; //91
    unsigned char maint_date[3]; //94
```

MAINTENANCE STRUCTURE

```
struct          //97
{
    unsigned char CmdLine :1;
    unsigned char Manual :1;
    unsigned char App :1;
    unsigned char Eod :1;
    unsigned char nu :4;

    }flag1;

    unsigned char data[MAINT_LEN-97]; //98 MAINT_LEN=160
    struct mnt_tail_ tail;          //161
};                                  //182
```



© Retalix, Inc. 2013

Retalix, Inc. endeavors to ensure that the information in this document is correct and fairly stated but does not accept liability for any error or omission.

The development of Retalix products and services is continuous and published information may not be up to date. It is important to check the current position with Retalix. This document is not part of a contract or license, save insofar as may be expressly agreed.