

---

# *QVS Software*

*mPOS (Mobile Point of Sale)*

*USER MANUAL*

*QVS Software, Inc*

*v.4.0*

*March 31, 2005*

---

## *Revision History*

<b>Document Number</b>	<b>Date</b>	<b>Author</b>	<b>Comments</b>
	3/31/2005	Donn Machak	Updated Documentation per review.
	3/21/2005	Donn Machak	Updated Documentation for 4.0 Release
	9/15/2002	Jon S Akhtar	Updated Documentation for 3.0 Release
	11/26/2001	Jon S Akhtar	Added new configuration utility documentation
	5/01/01	Donn Machak	Added keymap definitions for SA, CSA, and GSA.
	3/23/01	Donn Machak	Added <i>mPOS</i> Application Messages section.
	2/23/01	Donn Machak	Initial version

Copyright © 2002 by QVS Software, Inc. All rights reserved.

No part of this document may be reproduced or used in any form, or by any electrical or mechanical means, without permission in writing from QVS Software. This includes electronic or mechanical means, such as photocopying, recording, or information storage and retrieval systems. The material in this document is subject to change with notice.

QVS Software does not assume any product liability arising out of, or in connection with, the application or use of any product, circuit, or application described herein.

No license is granted, either expressly or by implication, estoppel, or otherwise under any QVS Software, Inc, intellectual property rights. An implied license only exists for equipment, circuits, and subsystems contained in QVS products.

Symbol and Spectrum24 are registered trademarks of Symbol Technologies, Inc. Other product names mentioned in this document may be trademarks or registered trademarks of their respective companies and are hereby acknowledged.

QVS Software, Inc  
5711 Six Forks Rd. Suite 300  
Raleigh, NC 27609

## Table of Contents

<b>Revision History</b> .....	<b>i</b>
<b>Table of Contents</b> .....	<b>2</b>
<b>1 Introduction</b> .....	<b>4</b>
1.1 Overview.....	4
1.2 Intended Audience.....	4
1.3 Retailer Features and Benefits.....	4
1.4 Customer Benefits.....	5
1.5 mPOS Diagrams.....	6
1.5.1 Standard mPOS Configuration.....	6
1.5.2 Printing Options.....	7
1.6 Features and Benefits.....	7
1.6.1 Features.....	7
1.6.2 Limitations.....	9
<b>2 Installation &amp; Configuration</b> .....	<b>10</b>
2.1 Overview.....	10
2.1.1 Anatomy of a screen.....	10
2.2 Configuring the IBM Store Controller.....	15
2.2.1 Assumptions.....	15
2.2.2 LAN Terminal Number Definitions.....	15
2.3 Installing and Configuring the QVS Terminal Concentrator for Windows product.....	16
2.3.1 Assumptions.....	16
2.3.2 Installation.....	16
2.3.3 TC Screen: No Active mPOS Terminals.....	16
2.4 Configuring the Symbol handheld device network parameters.....	17
2.4.1 Configuring the PPT 8846/8146.....	17
2.4.2 Configuring the MK2000.....	17
2.5 Building the Keypmap File ( XML Configuration File).....	17
2.5.1 Purpose of the configuration file.....	17
2.5.2 Configuration File Structure.....	18
2.5.3 Views And Default.....	20
2.5.4 Layouts.....	23
2.5.5 Transitions.....	24
2.6 Loading the mPOS Terminal Client Code.....	27
2.6.1 ZIP file.....	27
2.7 Configuring the mPOS Terminal Client.....	28
2.7.1 mPOS Configuration Application.....	29
2.8 Removing the Pocket PC Welcome Screen.....	33
2.9 Configuring the mPOS Printer.....	33
2.9.1 Zebra QL320 wireless printer.....	33
2.9.2 Comtec 2 inch printer.....	34
<b>3 General Instructions</b> .....	<b>34</b>
3.1 Symbol Handheld Device Information.....	34
3.2 Running the mPOS Application.....	34

3.2.1	<i>mPOS</i> Screen Layout.....	34
3.2.2	<i>mPOS</i> Application Messages .....	36
3.2.3	Starting the <i>mPOS</i> Application .....	37
3.2.4	Running the <i>mPOS</i> Application .....	38
3.2.5	Buttons on the PPT 8846/8146 Terminal.....	39
3.2.6	MK2000 Layout.....	40
3.2.7	Turn the Manager Key On and Off.....	40
3.2.8	Exit the <i>mPOS</i> Application .....	40
3.2.9	Clearing the NVRAM for the <i>mPOS</i> terminal .....	41
<b>4</b>	<b><i>Problem Determination</i></b> .....	<b>42</b>
4.1	<i>mPOS</i> Application starts but the “Connection Fails” .....	42
4.2	The Symbol device(s) Powers Off Too Frequently.....	43
4.3	The Comtec printer prints ‘not’ characters.....	43
	<b><i>Appendix A. Sample Barcodes</i></b> .....	<b>44</b>
	<b><i>Appendix B. mPOS XML Configuration Schema</i></b> .....	<b>45</b>
	<b><i>Appendix C. Acronyms</i></b> .....	<b>77</b>
	<b><i>Appendix D. Related Publications</i></b> .....	<b>78</b>

## **1 Introduction**

### **1.1 Overview**

Symbol's *mPOS* (Mobile Point Of Sale) solution offers retailers a mobile client that can function as a point-of-sale register. Using a Symbol PPT 8846, 8146, or MK 2000, a mobile cashier can scan items, take credit card payment, and issue a customer receipt, anytime and anywhere in the store.

The *mPOS* solution is a product developed for Symbol Technologies by QVS Software, Inc. It allows retailers to run their existing IBM 4690 Point of Sale (POS) applications on the Symbol family of products, incorporating integrated bar code scanning and integrated wireless LAN connectivity on the Microsoft Pocket PC platform. Architecturally the *mPOS* product is integrated into an existing IBM 4690 OS platform using the QVS 4690 Terminal Concentrator for Windows product.

The *mPOS* system presents the end user with a friendly user interface which incorporates touch or pen input while interfacing to the standard list of POS peripherals needed in the retail industry. The product was designed to resolve the need for a fully functional mobile POS solution required in retail stores where the necessity to satisfy and increase customer convenience is of the utmost importance while allowing a retailer to easily support and run the POS applications without modifications or recompilation. The *mPOS* Terminal is perceived by the IBM 4690 controller as an IBM 4690 terminal.

### **1.2 Intended Audience**

This document is to serve as a guide for all users of *mPOS*. It is assumed that the reader is experienced in using IBM 4690/OS as well as Microsoft Windows.

### **1.3 Retailer Features and Benefits**

The *mPOS* solution allows seamless integration to the retailers existing Operating System and IBM Retail Applications, without modifications

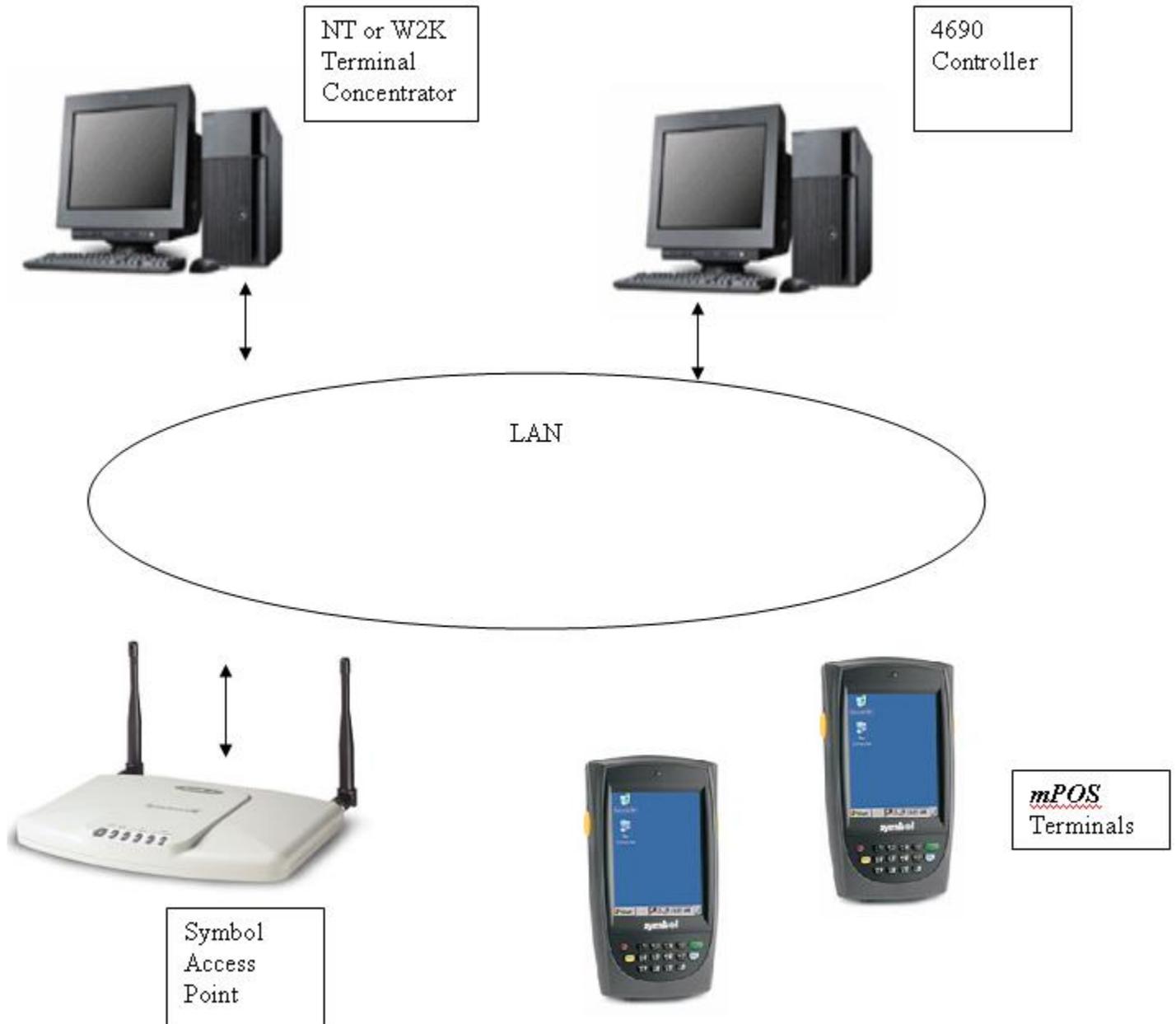
- All register functions are supported on the mobile device with the exception of functions that require specific external devices such as cash drawers, check reader and customer displays.
- *mPOS* executes most register functions on the Pocket PC device. Training is minimal as the keying sequences are identical to the other POS registers in the store. The Store Controller views the mobile device as another register in the store and all back office accounting reports reflect activity consistent with the other registers.
- *mPOS* offers the flexibility to check out customers anywhere in the store. With wireless mobility, store associates can offer a higher level of customer service and eliminate long lines at registers.

## *1.4 Customer Benefits*

- Store Selling Space - *mPOS* does not require any fixed area in the store. Maximize throughput without the requirement for excessive floor space. Freed up space can be utilized to expand merchandise presentation.
- Queue Busting – By utilizing *mPOS*, customers won't have to wait in long lines at the Point of Sale. Check your customers out anytime – anywhere in the store!
- Peak Trading Periods - The *mPOS* system is great for holidays, special promotions and special area sales, such as sidewalk and garden sales.
- “The Customer Comes First!” - The *mPOS* solution allows you to escort your customer through the store offering suggestions and assistance, eliminating the need for the customer to ever wait for service. Personal and escorted shopping offers the most personal of shopping experiences and will bring new levels of customer satisfaction to your business.

1.5 *mPOS* Diagrams

1.5.1 Standard *mPOS* Configuration



## 1.5.2 [Printing Options](#)

### 1.5.2.1 *Zebra 2844*

- Zebra 2844 serially attaches to the Symbol Client Bridge.
- No cash drawer support.
- Use an MSR attachment to capture credit card information.

### 1.5.2.2 *Zebra QL320*

- A Symbol Spectrum24 OEM network interface card resides inside the printer and makes the printer a network IP addressable printer.
- Use the Symbol MSR attachment to capture credit card information.

### 1.5.2.3 *Zebra Comtec Cameo2 2 inch Printer*

- The Cameo2 connects via a cable to the PPT 8846/8146 via its serial interface.
- This printer has a belt clip.
- Use the printer's MSR to capture credit card information.

## 1.6 *Features and Benefits*

### 1.6.1 [Features](#)

#### 1.6.1.1 *User runs existing POS Application*

*mPOS* is a software solution that provides a migration path for existing IBM POS application users to implement an open architecture POS Terminal Solution.

Supported applications include:

- General Sales Application (GSA)
- Chain Sales Application (CSA)
- Supermarket Application (SA)
- Custom POS Applications coded to the IBM CBASIC or C-API environment

#### 1.6.1.2 *Communications*

The wireless technology is Symbol Spectrum24 802.11 Frequency Hopping (FH) or 802.11b Direct Sequencing (DS) on the Pocket PC platform. The Access Points reside on an Ethernet LAN with the TC PC, which allows you to run multiple versions.

Communications from the Symbol Device to the TC PC is accomplished using the TCP/IP API on the actual handheld.

Communications between the IBM Store Controller and the TC PC may be on an Ethernet or Token Ring LAN. The protocol is IBM's Terminal Controller Communications (TCC) protocol. If the IBM Store Controller is configured for Token Ring and Ethernet LAN support, the TC PC must reside on the IBM Store Controller's TCC LAN. Refer to the Store Controller configuration for this information.

NT or Win2K Terminal Concentrator	
Controller	<i>mPOS</i>
TCC	RPAM
DLC	TCP/IP

TC communicates with the Controller using TCC/DLC.

TC communicates with *mPOS* using Remote Peripheral Access Method over TCP/IP.

<i>mPOS</i>
RPAM
TCP/IP

*mPOS* communicates with the TC using Remote Peripheral Access Method over TCP/IP. The RPAM interface is a documented interface that allows TC to control remote POS peripherals.

### 1.6.1.3 Printer Configurations

#### 1.6.1.3.1 Zebra 2844-Z

- This Model is supported w/ a serial interface.
- It requires its own power supply
- There must be a solid green light for the printer to work properly, if the light is flashing, it is offline and you must press the button next to it to bring it back online.
- This version will only support 3" thermal paper, and contains a blade to allow it to separate the receipts.
- The POS application must support the Model 4 Printer.
- Printer errors are ignored. The operator must reprint the receipt if an error occurs.
- Use the Symbol MSR attachments to capture credit card information.
- Logo printing is supported.
  - Logo printing is supported by printing a BMP bitmap file that resides on the handheld device when a logo print command is received.
- Document insert printing is not supported.

#### 1.6.1.3.2 Zebra Comtec Cameo2 printer

- Serially attached to the bottom of the *mPOS* terminal via a cable provided by Symbol Technologies.
- Logo printing is supported.
  - Logo printing is supported by printing a PCX bitmap file that resides on the Symbol device when a logo print command is received.
- There is no physical mechanism to cut receipts so the user is prompted to remove the receipt when a CUT command is received with the following prompt:

- “REMOVE CUST RECEIPT
- <CLEAR> TO CONTINUE.”
- Use the Zebra Comtec’s MSR to capture credit card information.
- The Zebra Comtec printer attaches to the user’s belt.
- Document insert printing is not supported.

### 1.6.2 Limitations

The following are not supported in the *mPOS* solution:

- Pin pad device
- Scale support (weighted items)
- 2x20 customer display
- MICR Check reader device (only supported with IBM 4610 Native Mode)
- The transaction journal is not supported. Either electronic journaling or custom code is required to capture the transaction journal information.
- Double Byte Character Support (DBCS)
- National Language Support for *mPOS* specific prompts.

## 2 Installation & Configuration

### 2.1 Overview

Installation and configuration involves several components. This manual assumes the Store Controller is configured for running the base General Sales Application (GSA) on IBM POS registers. Also, the NT Server is attached to the Store Controller's TCC LAN.

Before installation and configuration can begin, important information must be obtained from the end-user. The first thing being the current physical keyboard layout along with the functions associated with each key. These functions are known as function codes or f-codes. An f-code simply assigns a number to any given action that the user interface (i.e. the keyboard, the touch pad buttons on the screen) exerts on the POS application, this can be some sort of ASCII input character such as a number or letter, or it could be a motor function. A motor function tells the POS application to do something with the previous f-codes entered, for example, lets say you were asked to enter your name: you would first press the hardware keys for J-O-H-N-(space)-S-M-I-T-H, these are all input function codes; after this you would most likely press the ENTER key, this would be your motor function code. Now, let's say that in addition to your name you were asked to enter in your phone number, if these two fields are on the same screen, This is an integral part of designing the user interface on the *mPOS* application.

The second bit of information that must be obtained is the state table. Each time the POS application requires an action from the user, it "unlocks" into a state. To expand on this, it actually unlocks input into a state, subsequently, only one state may be unlocked at a time, therefore it simultaneously locks the input from another state, this helps represent an input string. The information is locked until it is all submitted to the POS application. For example if the user wish's to acquire a total and take payment from their customer, they would first press some form of a TOTAL button, the POS application would then process the command, add up the prices and sales tax, and then display the final total and ask for a form of payment. Let's say the customer wishes to charge the items to their credit card. Some kind of CREDIT button would be pressed and the POS application emulator will display another screen asking for the card to be swiped. In this example, the IBM has changed states 3 times, the first state being some kind of receipt display, the second being the payment/total display, and the third being the credit display. Each state has specific buttons that can be used for that particular state. Therefore the state table defines what buttons should be shown for each state.

In addition to learning what *mPOS* requires, one may want to know what limitations the user would like to place on the application. This will vary for each roll-out. For example, some retailers may wish their mobile salespeople to not perform cash transactions. *mPOS* may be configured to not allow for this.

#### 2.1.1 Anatomy of a screen

Think of the *mPOS* application user interface as having three layers: the default, the view and the layout. Each layer is stacked on top of the other and the top two layers allow the bottom two layers to be shown through it. The area through which the POS application is to be emulated through can be defined on any layer. The bottom layer is the default layer. Think of it as your computer monitor, in order to view what is on your computer you must look at it through the monitor. Nothing can expand beyond its boundaries, it does not have any buttons,

except the picture adjust buttons of course, but we are focused on the screen. You can map out buttons in this view, but you can not assign them a function, this is done to later assist you in adding buttons to the layouts and the views.

The second layer, is known as a view, it can be thought of as being like your Windows desktop, it can only be as big as your monitor, contains a few icons that generally always remain on the screen , they can however be changed. The *mPOS* view fits in the default view, has a few buttons that remain fairly constant and are needed on many layouts, such as the Clear button as well as the Enter button.

The third and top layer is the layout. The layout should be compared to your file browser window. It appears over your desktop, and each time you click on a file the window changes to display that files content. The layout works very similarly. It contains several buttons that for the most part change with each screen change. For example when you hit the total button while making a sale, the buttons immediately change to payment option buttons, in this case the layer has changed, the view has remained the same, the buttons that did not change were part of the view.

It is here where the importance of f-codes and states are shown. F-codes are used to determine what buttons should be designed. The states are used to determine which buttons should show for each screen. Determining the distribution of buttons is the underlying basis for the views and the layouts.

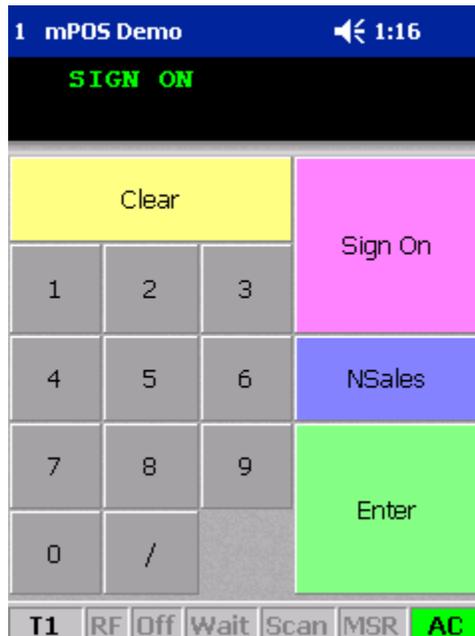
This is a general default view. Note the title bar located at the top of the snapshot, this is where you will see your state number as well as the name associated with that state.



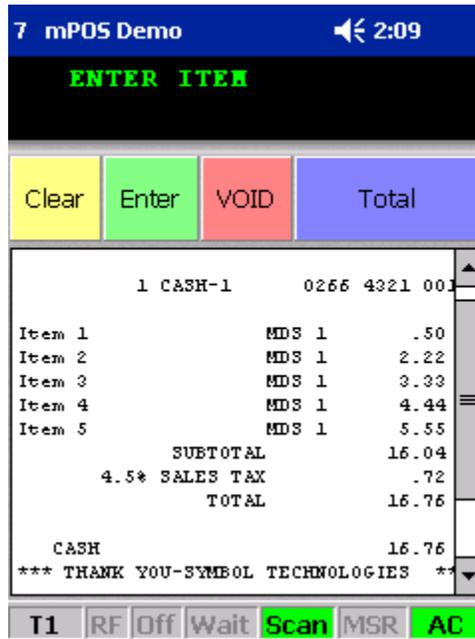
Next the **VIEW** is applied. This adds the numeric keypad, but leaves room for other keys to be added. These keys have several layouts that rely on them. They are added to the view to maintain efficiency; rather, they do not have to be written into each layout.



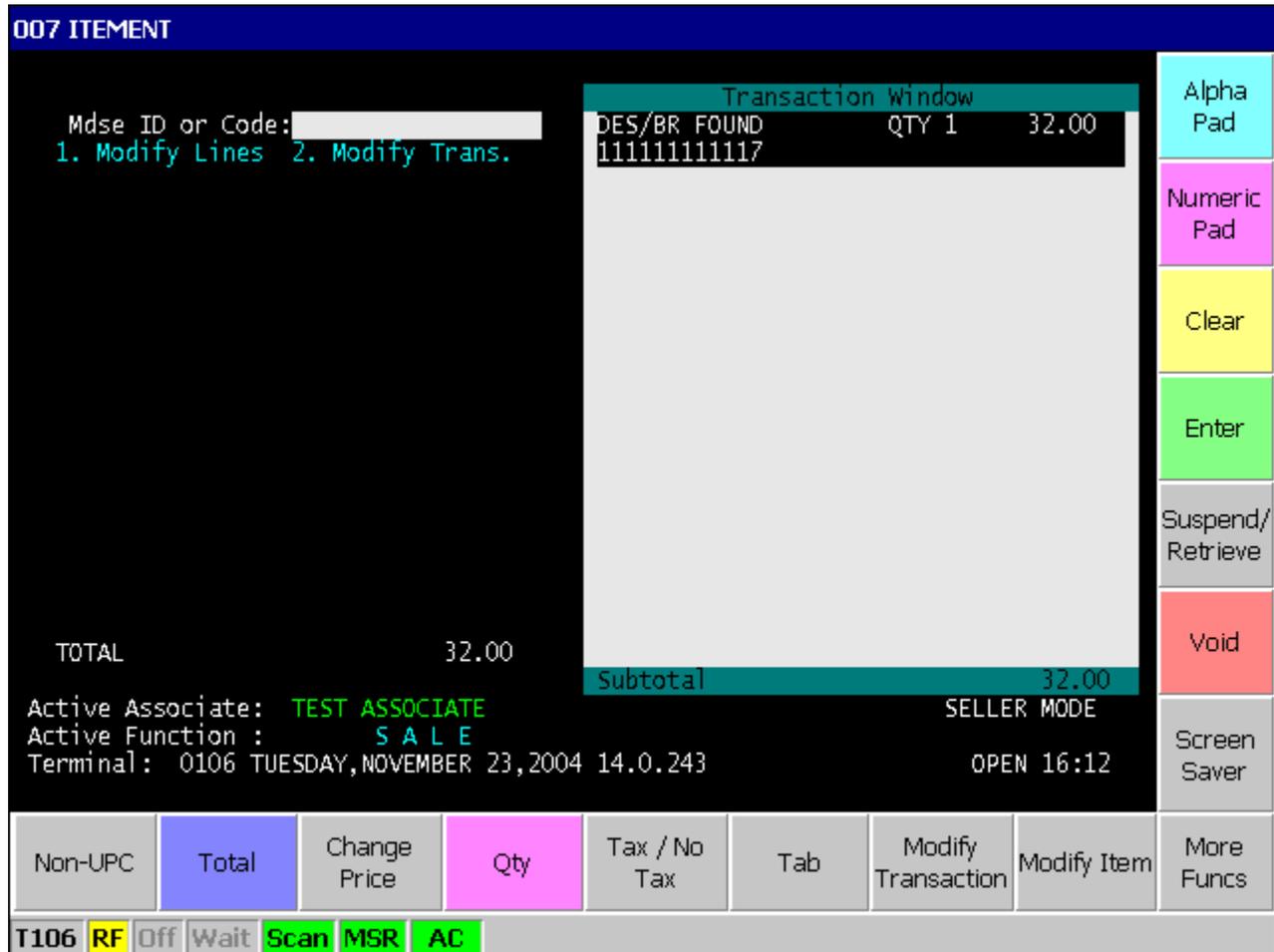
Finally, the **LAYOUT** is added which adds the function keys. The keys that have been added will change periodically each time the POS Application changes states.



The user should understand that there are two basic types of display output for POS applications. The oldest used is called a 2x20 display and the other is an enhanced full screen display. An example view of an on-line receipt for a 2x20 display is shown next. The 2x20 would be the area below with the black background and the text "ENTER ITEM" shown.



An example of an enhanced full screen display is shown next. In this POS application the largest screen possible is used which is the 25x80 mode (25 rows with 80 columns). The keys shown around the outside of the example are defined by you in the keymap file (also called the XML Configuration file) which will be discussed later in this document.



The installation steps are listed below:

- Configuring the IBM Store Controller
- Installing and Configuring the QVS Terminal Concentrator for Windows product
- Configuring the Symbol handheld device network parameters
- Building the keymap file (XML Configuration file)
- Loading the *mPOS* Terminal Client Code on the Symbol devices.
- Configuring the *mPOS* Terminal Client on the PPT Symbol devices.
- Removing the Pocket PC Welcome Screen
- Configuring the *mPOS* Printer

To run *mPOS*, the following hardware is required.

- IBM 4690 Store Controller with Token Ring or Ethernet
- PC with Windows NT, Windows 2000, or Windows XP installed
  - Ethernet Adapter
  - Token Ring Adapter if the Store Controller's TCC LAN is Token Ring
- An Ethernet LAN
- A Symbol Access Point
- Mobile POS device

The following hardware is optional

- Printer
- Wireless network card for the MK2000

## 2.2 *Configuring the IBM Store Controller*

### 2.2.1 Assumptions

- POS application OS is installed on the Store Controller
- IBM GSA is installed on the IBM Store Controller
- IBM LAN TCC Feature is purchased.
  - To verify, insert the IBM 4690 Feature Diskette and type A:\install at a command prompt. Follow the prompts until the Features Purchase screen with the LAN TCC Feature is displayed. An X indicates the feature has been purchased for the controller that will support the *mPOS* terminals.

### 2.2.2 LAN Terminal Number Definitions

- Choose terminal numbers for *mPOS* terminals that are not used existing POS LAN terminals.
- System Configuration
  - On the IBM Store Controller, from the main menu, select "4 Installation and Update Aids"
  - Select "1 Change Configuration Data"
  - Select "3 System Configuration"
  - Select "7 LAN Terminal Definition"
  - Select "1 Define LAN Terminals" and define the *mPOS* terminals as LAN terminals and assign a Primary Controller and a Backup Controller.
  - Select "4 Display the LAN Terminals" to validate the entries.
  - Press ESC to return to the primary Configuration menu.
  - Select "4 Activate Configuration"
  - Select "3 System Configuration"
  - After activation completes, verify no errors exist.
  - Reboot the Controller to activate the configuration changes.
- Controller Configuration
  - None required.
- Terminal Configuration

- On the IBM Store Controller, from the main menu, select “4 Installation and Update Aids”
- Select “1 Change Configuration Data”
- Select “1 Terminal Configuration”
- Define terminal definitions for each *mPOS* terminal.
  - Use an existing terminal as a prototype.
  - The only information that *mPOS* uses in the configuration is the terminal number and the primary application name.
- Press Esc to return to the primary Configuration menu.
- Select “4 Activate Configuration”
- Select “1 Terminal Configuration”
- After activation completes, verify no errors exist.
- Below is an example of an OS Terminal Configuration Report showing the name of the POS application name and the keyboard layout used:

```

Store 75                ACTIVE TERMINAL CONFIGURATION REPORT                03/14/01
                        ACTIVE TERMINAL LOAD DEFINITION REPORT

4693-4x1 TERMINAL NUMBER    91          APPLICATION DUMP FLAG    N
4693-xx2 PARTNER NUMBER     0
DEVICE GROUP NAME          4694_245

PRIMARY APPLICATION NAME    R::ADX_IPGM:EALTS10L.286
COMMAND TAIL
KEYBOARD LAYOUT            KBD50KEY
FULL SCREEN VIDEO SUPPORT  N
    
```

## 2.3 Installing and Configuring the QVS Terminal Concentrator for Windows product

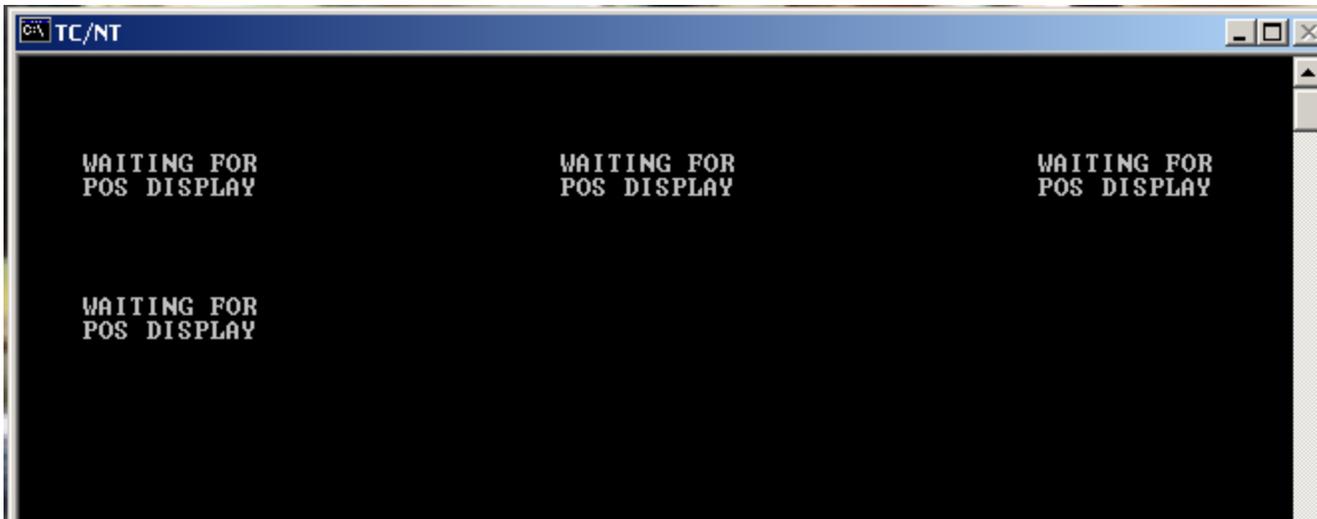
### 2.3.1 Assumptions

- Communications is set up on the Windows NT, Windows 2000 PC, Server 2003, or Windows XP
  - DLC protocol must be installed.
  - An Ethernet Adapter is installed.
    - TCP/IP must be installed
    - DLC protocol is installed if the IBM Store Controller’s TCC LAN is Ethernet.
  - A Token Ring Adapter is installed if the IBM Store Controller’s TCC LAN is Token Ring.
    - DLC protocol is installed.

### 2.3.2 Installation

Refer to the QVS Software Terminal Concentrator for Windows User Manual.

### 2.3.3 TC Screen: No Active *mPOS* Terminals



- Configuring the Symbol Access Point
  - Plug the Access Point in to the Ethernet LAN
  - Assign an ESS\_ID to the Access Point. Contact your Symbol Representative for details on installing the wireless network.

#### 2.4 *Configuring the Symbol handheld device network parameters*

- Obtain these network parameters from your network administrator.
  - DHCP settings for the network
  - If not dynamically assigned, IP Addresses and Subnet Masks for each *mPOS* terminal

##### 2.4.1 Configuring the PPT 8846/8146

All Symbol handhelds running CE .net 4.1, CE .net 4.2 or PPC 2003 should be configured via the Mobile Companion's profiles. This program is chosen from the icon tray.

##### 2.4.2 Configuring the MK2000

Depending on whether the MK2000 is attached via a hard-wired cable or wireless card will determine how the device is configured.

#### 2.5 *Building the Keymap File ( XML Configuration File)*

##### 2.5.1 Purpose of the configuration file

The configuration file defines the keys to be displayed in each state of the application. Some sample states are as follows:

- Sign On State
- Item Entry State
- Tender State
- Account Number State

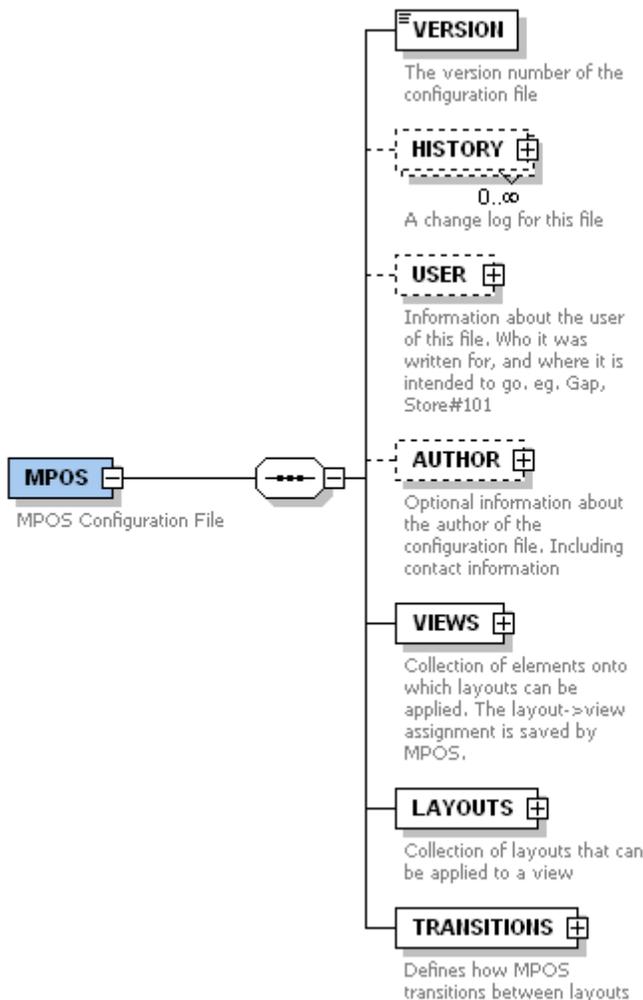
The POS Application assigns a number to each state, rather than a character string. This makes writing transitions much easier. This is referenced later in the manual. To increase the flexibility of the *mPOS* application, *mPOS* supports the use of an XML based configuration file. This file allows for the layout of the *mPOS* screen, and definition of certain *mPOS* actions such as the behavior of hardware keys.

### 2.5.2 Configuration File Structure

The *mPOS* configuration file is an XML document which is defined by the schema *mPOS.xsd* (included in the *mPOS* release package). This file allows for definition of the general user interface behavior of the application. This information is meant as an overview. For full documentation of the XML file structure refer to [Appendix B. mPOS XML Configuration Schema](#). When creating an XML configuration file, an editor that supports validation against the xml-schema should be used. An example of such an editor is Altova's XML Spy ([www.xmlspy.com](http://www.xmlspy.com)).

element **MPOS**

diagram



children **VERSION HISTORY USER AUTHOR VIEWS LAYOUTS TRANSITIONS**

notes The basic *mPOS* document contains 4 mandatory subelements: **A DEFAULT VIEW, REGULAR VIEWS, LAYOUTS, and TRANSITIONS**, generally it will contain a fifth optional element called the version number.

**VERSION** - The version tag is optional, it is a string that defines the version of *mPOS* that the keymap file was written for, e.g. 3.0. This is important for compatibility with future versions of *mPOS*.

**VIEWS**

- A default **VIEW** defines all the spaces through which the *mPOS* will interact with the user. This means all button perimeters are defined here, but not actually shown.
- A “normal” **VIEW** is a collection of basic UI elements, e.g. “keyboard” **VIEW**, that more than one **LAYOUT** will need.

**LAYOUTS** – A collection of elements which represent an additional UI “layer” applied to a **VIEW**, for the most part, each time an element associated with a **LAYOUT** is accessed, a different **LAYOUT** will then appear. Depending on which element is accessed, the **VIEW** may or may not change.

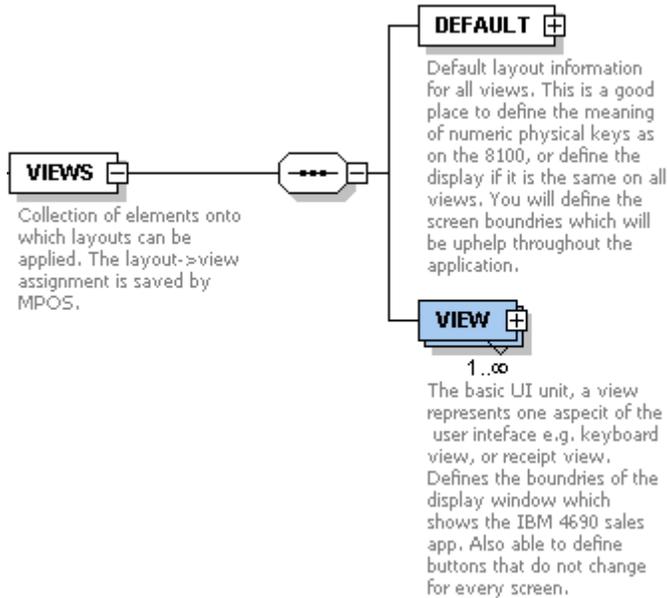
**TRANSITIONS** – Defines where and when the **VIEWS** & **LAYOUTS** are used in response to events which occur in the sales application, e.g. application state change, they can be thought of as conditional statements (if...then... statements).

### 2.5.3 Views And Default

Please note: these snapshots were taken from an XMLSpy display and should be read from left to right. An unbroken line represents a mandatory relationship, while a dotted line represents an optional relationship. Each box represents an element, the boxes connected to a specific element are known as its subelements. The small writing underneath each element describe what that element's purpose is. Also, if the element box appears to be stacked on top of another box, than there can be more than one of that particular element..

element **MPOS/VIEWS**

diagram

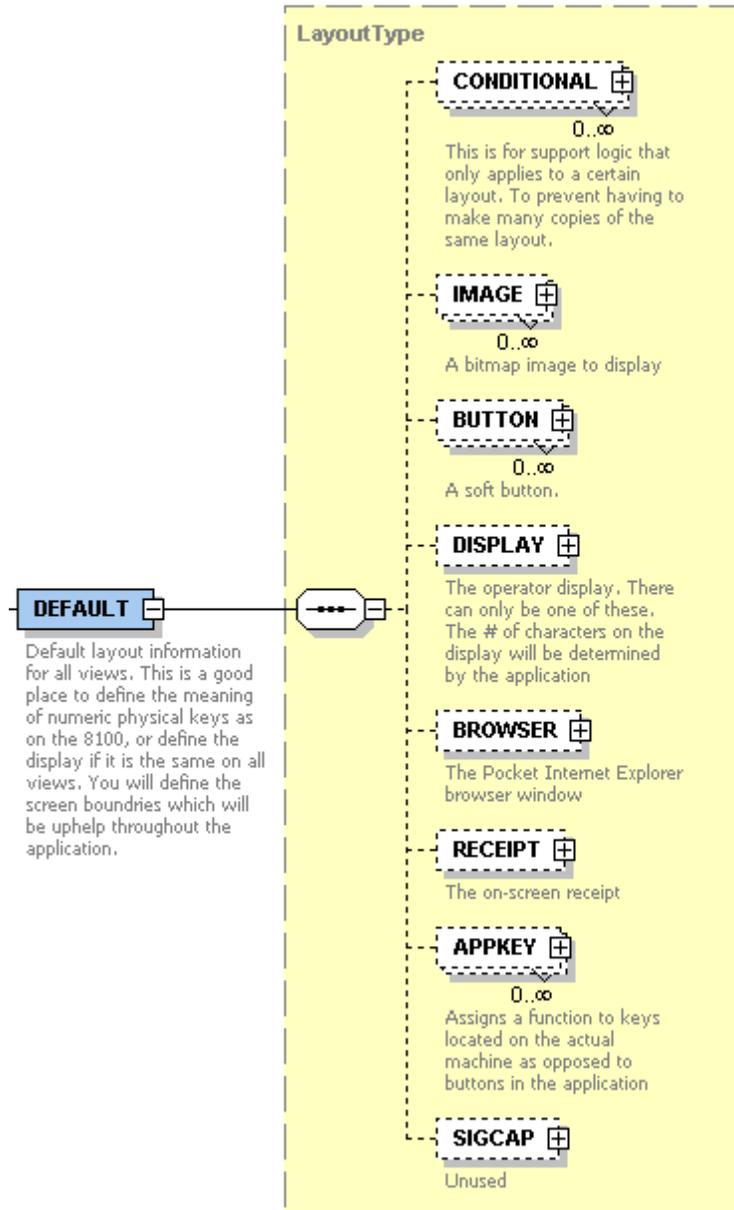


children **DEFAULT, VIEW**

notes The **VIEWS** tag will contain a **DEFAULT** tag and 1 or more **VIEW** tags.

element **MPOS/VIEWS/DEFAULT**

diagram

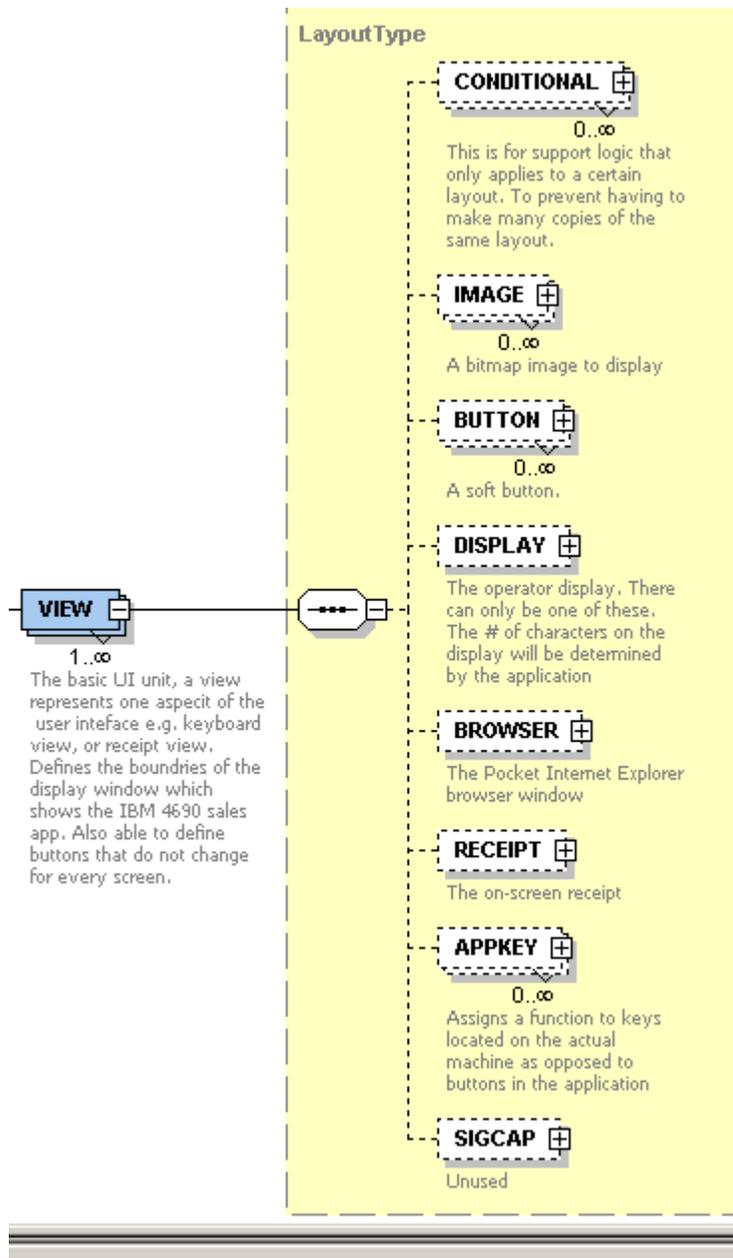


type **LayoutType**

children **CONDITIONAL IMAGE BUTTON DISPLAY BROWSER RECEIPT APPKEY**

element **MPOS/VIEWS/VIEW**

diagram



type extension of **LayoutType**

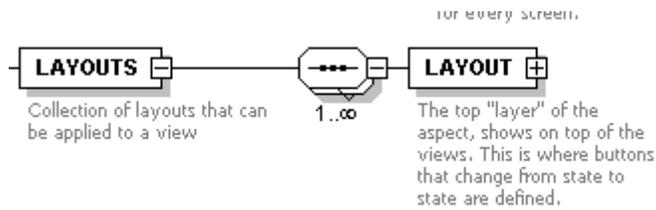
children **CONDITIONAL IMAGE BUTTON DISPLAY BROWSER RECEIPT APPKEY**

attributes	Name	Type	Use	Default	Fixed	Annotation
	viewID	xs:ID	required			

## 2.5.4 Layouts

### element **MPOS/LAYOUTS**

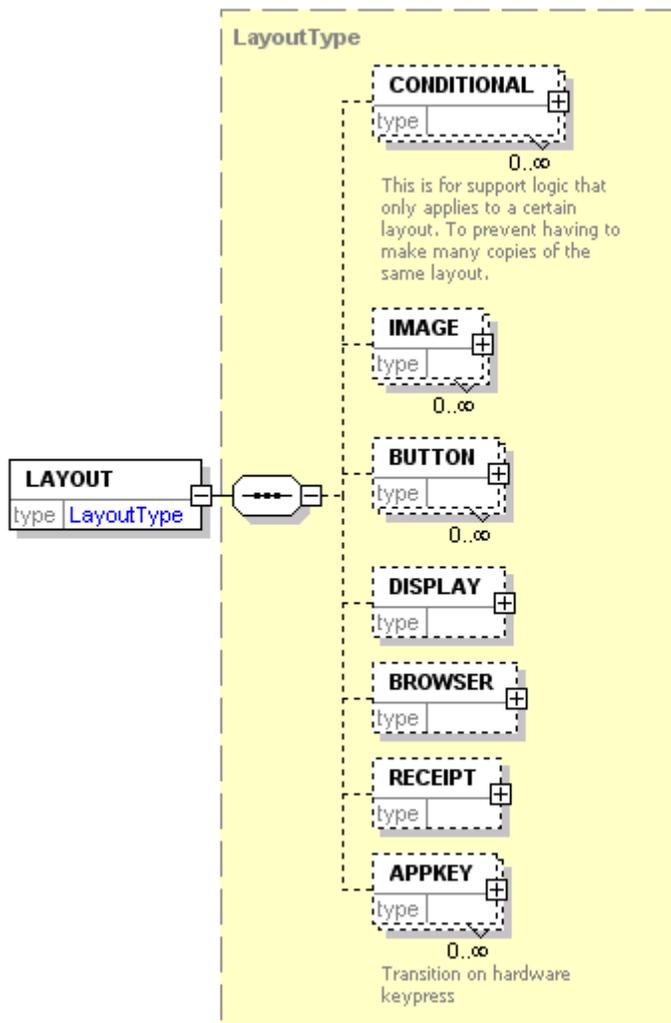
diagram



children **LAYOUT**

### element **MPOS/LAYOUTS/LAYOUT**

diagram



type extension of **LayoutType**

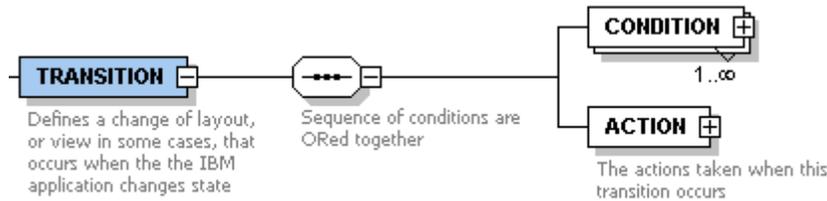
children **CONDITIONAL IMAGE BUTTON DISPLAY BROWSER RECEIPT APPKEY**

attributes	Name	Type	Use	Default	Fixed	Annotation
	layoutID	xs:ID	required			

## 2.5.5 Transitions

### element *MPOS*/TRANSITIONS/TRANSITION

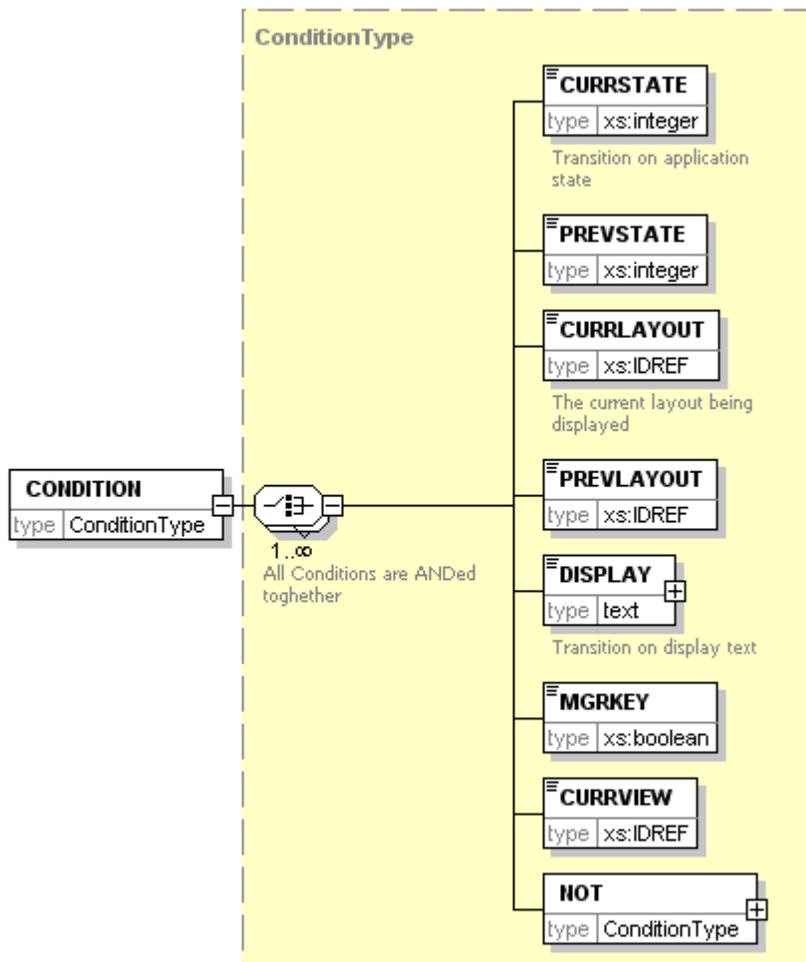
diagram



children **CONDITION ACTION**

### element *MPOS*/TRANSITIONS/TRANSITION/CONDITION

diagram



type **ConditionType**

children **CURRSTATE PREVSTATE CURRLAYOUT PREVLAYOUT DISPLAY MGRKEY CURRVIEW NOT**

notes **CURRSTATE is 0 when *mPOS* is first started. This is useful for setting the initial view.**

example

1. In this example, the state 3 is checked for in the condition, this is done using the `<XSTATE>` tag, this is saying, if the state before this one was not 3, then execute the following actions. If the condition is met, the view is set to “KEYBOARD”, the “RECEPT\_STD” layout is applied to the “RECEIPT” view. The browser is told to navigate to the home page URL. The application title bar text is also set to “Please Wait...”

```
<TRANSITION>
  <CONDITION>
    <XSTATE>3</XSTATE>
  </CONDITION>
  <ACTION>
    <VIEW viewID="KEYBOARD"/>
    <TITLE>003 RECEIPT</TITLE>
    <LAYOUT viewID="RECEIPT" layoutID="RECEIPT_STD"/>
    <URL>file://\application\mPOS_start.htm</URL>
  </ACTION>
</TRANSITION>
```

2. In this example, the XML is telling us that when the IBM application enters into state 111, it must execute the actions following the `<action>` tag. In other words, it will set the title bar to read “111 IDREQ”, the view will be changed to “NUMERIC\_VIEW.”

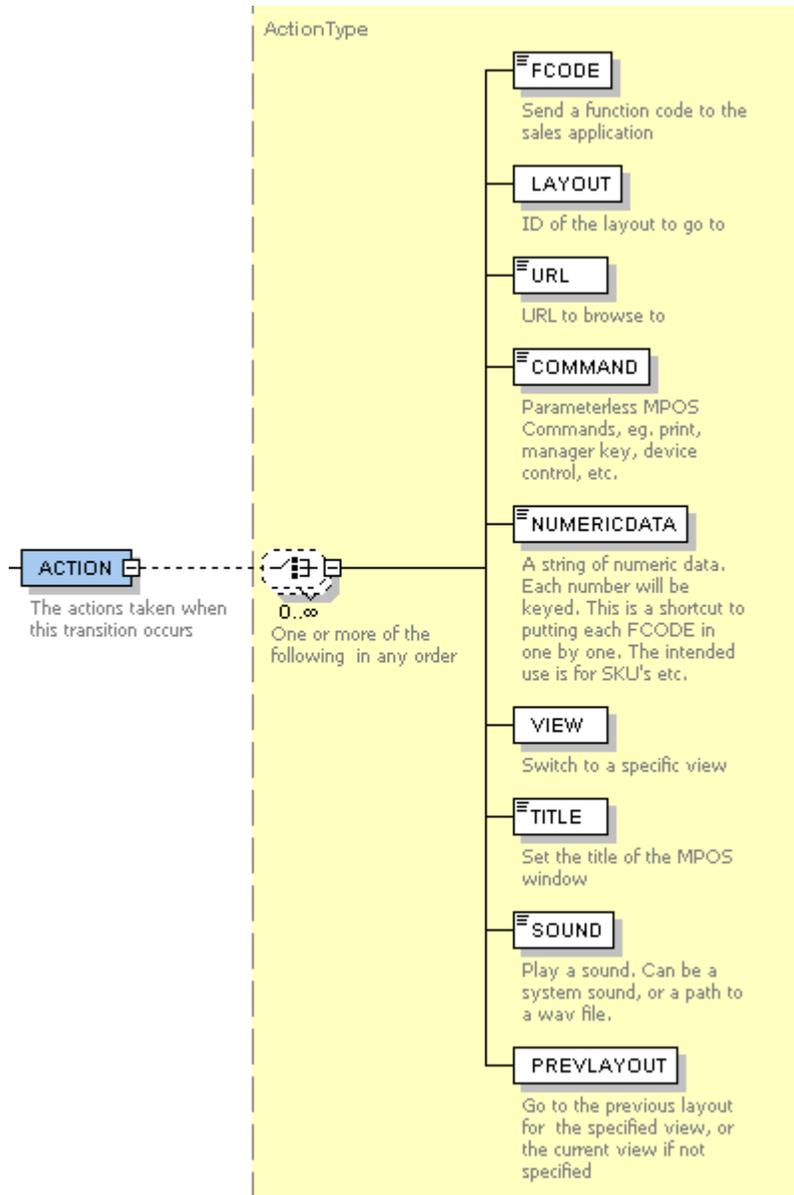
```
<TRANSITION>
  <CONDITION>
    <XSTATE>111</XSTATE>
  </CONDITION>
  <ACTION>
    <TITLE>111 IDREQ</TITLE>
    <VIEW viewID="NUMERIC_VIEW"/>
  </ACTION>
</TRANSITION>
```

3. This example is a little bit more complicated, it says that when the IBM application enters state 195, the *mPOS* application should execute these actions: change the title to read “195 EXPDATE”, change the view to “NUMERIC\_VIEW”, and change the layout to “MK2000\_MAIN”. The reason for two lines to ask the program to do this, are to make sure that no matter what the view is, whether it be “NUMERIC\_VIEW” or “MAIN”, it should still change to the “MK2000\_MAIN” layout.

```
<TRANSITION>
  <CONDITION>
    <XSTATE>195</XSTATE>
  </CONDITION>
  <ACTION>
    <TITLE>195 EXPDATE</TITLE>
    <VIEW viewID="NUMERIC_VIEW"/>
    <LAYOUT viewID="MAIN" layoutID="MK2000_MAIN"/>
    <LAYOUT viewID="NUMERIC_VIEW" layoutID="MK2000_MAIN"/>
  </ACTION>
</TRANSITION>
```

element **MPOS/TRANSITIONS/TRANSITION/ACTION**

diagram



type **ActionType**

## 2.6 Loading the *mPOS* Terminal Client Code

There are several methods of transferring files from a host machine to the mobile unit, most units have several different choices such as: USB, serial, FTP and the usage of one or more AirBEAM packages.

### 2.6.1 [ZIP file](#)

#### 2.6.1.1 Transfer using a Direct Connection

- Create a directory on your local PC (**MPOSInstall**) and unzip the provided **mPOS** zip file.
- Connect the Cradle or the Synchronization Cable to your desktop computer and attach the device to which you wish to make a direct connection.
- Launch the Microsoft ActiveSync (MSAS) tool and validate that the status displayed is “Connected.”
- Click the EXPLORE device icon to view the directories on the Mobile Device.
- Navigate to the \Application directory on the Mobile Device
- Navigate to your newly created local PC directory (**MPOSInstall**).
- Copy all the files from the \bin subdirectory of the (**MPOSInstall**) directory that you created on your PC to the target directory on the MPOS unit that is listed in the table that follows.

#### 2.6.1.2 Transfer using FTP on the MK2000

- Attain your Symbol device’s IP address from your network administrator
- On any computer connected to your network, open up Internet Explorer and type in the mobile device’s IP address in the address bar. A view of the mobile device’s directory will then be displayed.
- Navigate to the \Application directory on the mobile device.
- On the local computer, create a directory entitled “**MPOS**” and unzip the provided **mPOS** files into this directory.
- Create a folder on the device under \Application entitled “mpos”
- Copy all files from the **MPOS** directory directly into the \Application\mpos directory on the mobile device, simply highlight all the files and “drag-and-drop” to the mpos folder.

The following files need to be transferred to the MPOS unit:

Filename	Target Directory	Description
MPOS.CPY	Application	Symbol’s unique file for copying files from the Application directory to other locations. Only file placed in the \Application directory as apposed to the \Application\mpos directory
MSR_SYMCLIP.QDV	Application\MPOS	Plug-in MSR DLL needed to use the Symbol clip-on MSR.
<b>MPOS</b> Application.lnk	Application\MPOS	Shortcut to the <b>mPOS</b> Application EXE, which is moved to the Windows\Start Menu directory
<b>MPOS</b> Configuration.lnk	Application\MPOS	Shortcut to the <b>mPOS</b> Configuration EXE, which is moved to the Windows\Start Menu directory

MPOS.EXE	Application\MPOS	The <i>mPOS</i> program.
MPOS_CFG.EXE	Application\MPOS	The <i>mPOS</i> Configuration program. This program is used to set the store number, filename of the keymap file, etc.
RPAMFUNC.DLL	Application\MPOS	QVS DLL, which handles the communications between the Portable POS Application and the Terminal Concentrator. Communications is accomplished using the TCP/IP API.
QVSDBG.DLL	Application\MPOS	QVS debugging DLL to allow tracing and problem determination at the handheld device.
QVSRPAMARM.DLL	Application\MPOS	QVS's Remote Peripheral Access Method (RPAM) protocol that receives the POS application requested device I/O and passes it on to the MPOS application.
S24UTIL.DLL	Application\MPOS	RF driver DLL which utilizes the Symbol Spectrum24 wireless LAN protocol to report connectivity status to the MPOS program.
PWRDLL2.DLL	Application\MPOS	Power on/off driver DLL which is used by the MPOS application to know when the device is powered on or off.
PRN_ZEBRA320.QDV	Application\MPOS	Printer plug-in driver DLL for the Zebra QL320 wireless printer.
PRN_ZEBRA2844.QDV	Application\MPOS	Printer plug-in driver DLL for the serially attached Zebra 2844 printer.
MPOS.REG	Application	This registry file sets the default configuration needed. This file is updated automatically by the <i>mPOS</i> application, and allows for persistence of configuration data across a cold boot, should also be placed in the \Application directory with MPOS.CPY
MPOS_START.XML	Application\MPOS	Key map to display while mPOS is loading

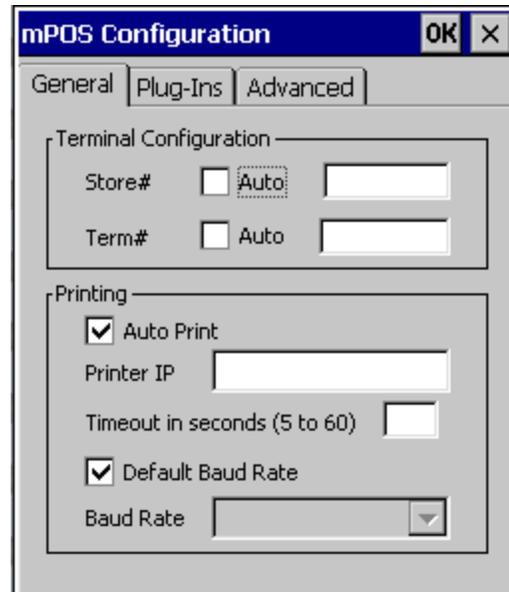
It is assumed that the Symbol devices will be preloaded with the proper Microsoft Pocket PC or Pocket PC 2002 operating system and the software that allows the user to launch their Demo programs.

## **2.7 *Configuring the mPOS Terminal Client***

Once the installation is complete you must cold boot the device. Follow the instructions in the Symbol Product Reference guide for instructions on how to cold boot your device.

## 2.7.1 *mPOS* Configuration Application

Assign the proper Mobile POS Configuration parameters by launching the *mPOS* Configuration application via the Start menu. A sample *mPOS* Configuration screen is seen here:



### 2.7.1.1 General Settings

#### 2.7.1.1.1 Store#

Auto (default)

Set the Store# to Auto unless the environment variable TSF\_STORE\_NUM is defined on the TC PC.

If TSF\_STORE\_NUM is defined, uncheck Auto and set Store# to the same value. The value should match the store number of the IBM Store Controller. The main use for this field is in a lab environment where multiple store controllers exist on the same LAN.

#### 2.7.1.1.2 Term # (POS Terminal number)

Set Term# to a valid Terminal address. Refer to the “LAN Terminal Number Definitions” section under “Configuring the IBM Store Controller”. Auto will set the terminal number to the last octet of the IP address MOD 100. So 10.1.1.204 would have a terminal ID of 4.

#### 2.7.1.1.3 Auto Print / Printer IP

Unchecked (default)

If Auto Print is checked, the *mPOS* application will automatically print the cash receipt at the end of each transaction. Printer IP is an IP address value for an IP addressable printer. This IP address should be the IP address of the Zebra QL320 wireless printer. Here you will enter the entire IP address with no leading or

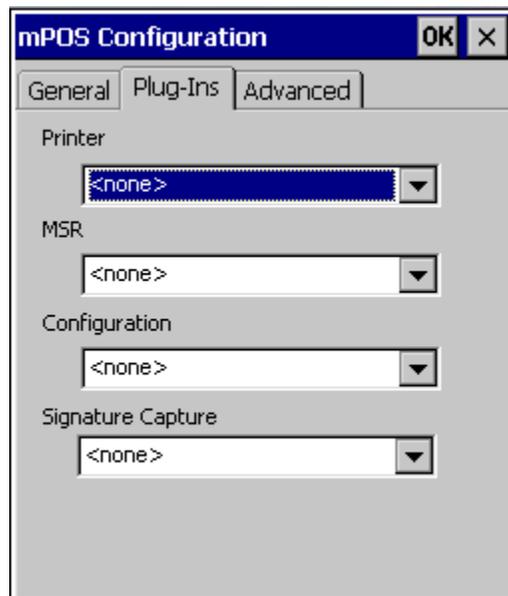
trailing text. The Zebra series of printers will print out a UPC code if the feed button is pressed, while running *mPOS* have it scan the UPC and the IP address will be automatically entered. The *mPOS* application will load without a printer, however if a print command is executed by you or the application, *mPOS* will timeout looking for a printer.

If the no auto-print option is set in the config field, this parameter is ignored.

#### 2.7.1.1.4 Baud Rate

Auto (default). Generally set to auto to utilize the default baud rate for printer connections. If another baud rate is desired, uncheck Auto and set the baud rate manually.

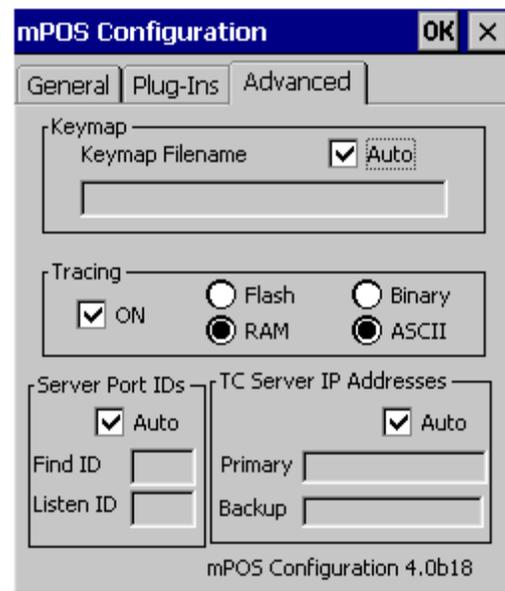
#### 2.7.1.2 Plug-ins



Device and configuration selections can be made using this page. Each drop-down will contain a list of all the installed drivers for that device type e.g. printer, MSR etc. Make sure you select the plug-ins that are specific to your device.

To initially get your application running you should not need to choose any of the plugins and leave them all as <none> until you are up and running your POS application.

#### 2.7.1.3 Advanced Settings



### 2.7.1.3.1 Keymap Filename

The Keymap Filename defines the keymap file for the POS application. A sample keymap file *mPOS.000* is included with the install package. It works with the base IBM GSA POS application.

Examples:

“\Application\mpos\keymap.xml” will load the keymap file from the Application\mpos directory of the mobile device.

“r::ADX\_UDT1\keymap.000” will load the keymap file from the Store Controller’s c:\ADX\_UDT1 directory.

If Auto is checked the *mPOS* unit will try and load a keymap file from the Store Controller’s c:\ADX\_UDT1 directory called *MPOS.xxx* where “xxx” is your terminal number. If this file is not found then the file *MPOS.000* will attempt to be loaded. While the extension is different, this is actually an XML file.

Three sample keymap files are provided with the MPOS installation package. They are as follows:

MPOS\_GSA.XML - Sample for a General Sales Application (GSA) 2x20 POS Application.

MPOS\_CSA.XML – Sample for a Chain Drug Sales Application (CSA) 2x20 POS Application.

MPOS\_SA.XML – Sample for a Supermarket Application (SA) 2x20 POS Application.

To ease your effort simply use one of the sample XML keymap files listed above and rename it to MPOS.000 and put it on your store controller in the ADX\_UDT1 directory and check “Auto” so that it is loaded. This should allow you to at least sign on to your application.

Please contact your Symbol technical representative for a sample full screen keymap file.

### 2.7.1.3.2 Tracing

This option turns on diagnostic logging for the *mPOS* application. The Flash option creates a log that is permanent; unfortunately it tends to slow down the application itself. The RAM option creates a file that is deleted once the device is turned off or runs *mPOS* again. Checking Binary will write using the Binary computer language; however it is less recoverable because it writes to the log file less frequently. ASCII is a character output which is easily readable and is constantly written to the log file but it tends to slow down the application.

### 2.7.1.3.3 Server Port ID

If your device has a firewall or similar security, you will set the port ID's through which you want the **mPOS** application to listen through. The default server find port is 7503 and the default server listen port is 7502.

### 2.7.1.3.4 TC Server IP Addresses

Here you can specify the IP address of the TC server, if Auto is checked the handheld will broadcast over the network for a TC server. This entry is used in environments where multiple computers are running TC and are on the same network or the LAN connection requires crossing routers.

## 2.8 Removing the Pocket PC Welcome Screen

Many customers do not like the Pocket PC welcome screen with its "tap and hold" tutorial. Because of this, the **mPOS** distribution contains the unWelcome program. This program consists of 3 files: unwelcome.exe, unwelcome.cpy, and unwelcome.lnk. When placed in the handheld's application directory, these files will remove the "tap and hold" tutorial & time zone setting screens, while leaving the pen calibration process intact. At startup, the user will be asked to calibrate the pen, and then will immediately be taken to the **mPOS** application (or the desktop if **mPOS** has not been set up to run automatically).

## 2.9 Configuring the mPOS Printer

### 2.9.1 Zebra QL320 wireless printer

To configure the Zebra QL320 printer, you will need the following items:

- Serial printer cable connecting a standard Windows PC with the Zebra QL320 Printer.
- Copy of Label Vista running on the Windows PC.
- The printer files Luc08pt.cpf, Luc10cp.ptf, Luc16cp.ptf and the ff.bat file.

To configure the printer, several font files need to be downloaded to the printer as follows:

1. Select "Printer->Send Font". Browse to file Luc08pt.cpf. Make sure that "Store to Flash File System" is checked. Press "Send". Status bar will update with progress of sending file.
2. Select "Printer->Send Font". Browse to file Luc10pt.cpf. Make sure that "Store to Flash File System" is checked. Press "Send". Status bar will update with progress of sending file.
3. Select "Printer->Send Font". Browse to file Luc16pt.cpf. Make sure that "Store to Flash File System" is checked. Press "Send". Status bar will update with progress of sending file.

Also, the FormFeed button on the front of the printer needs to be remapped so that when it is pressed, it will print a barcode with the printers current IP Address. To accomplish this, perform the following step:

1. Select "Printer->Send as FF.bat". Browse to file FF.BAT. Make sure that "Store to Flash File System" is checked. Press "Send". Status bar will update with progress of sending file.

Also, "Printer->Network Settings" can be used to configure the Printer's IP address and ESSID.

### 2.9.2 [Comtec 2 inch printer](#)

Contact Comtec for operating information. There are no known configuration requirements.

## 3 General Instructions

### 3.1 *Symbol Handheld Device Information*

Refer to the Symbol PPT 8846/8146 and the MK2000 Series QRG (Quick Reference Guide) and the Symbol MSR 3000 Magnetic Stripe Reader QRG for general information. Specific topics covered include:

- Symbol device parts
- Recharging the Battery
- Calibrating the Screen
- Using the device
- Typing and Writing with the Stylus
- Barcode scanning
- Resetting the device
- Maintenance
- Troubleshooting
- Using the MSR 3000

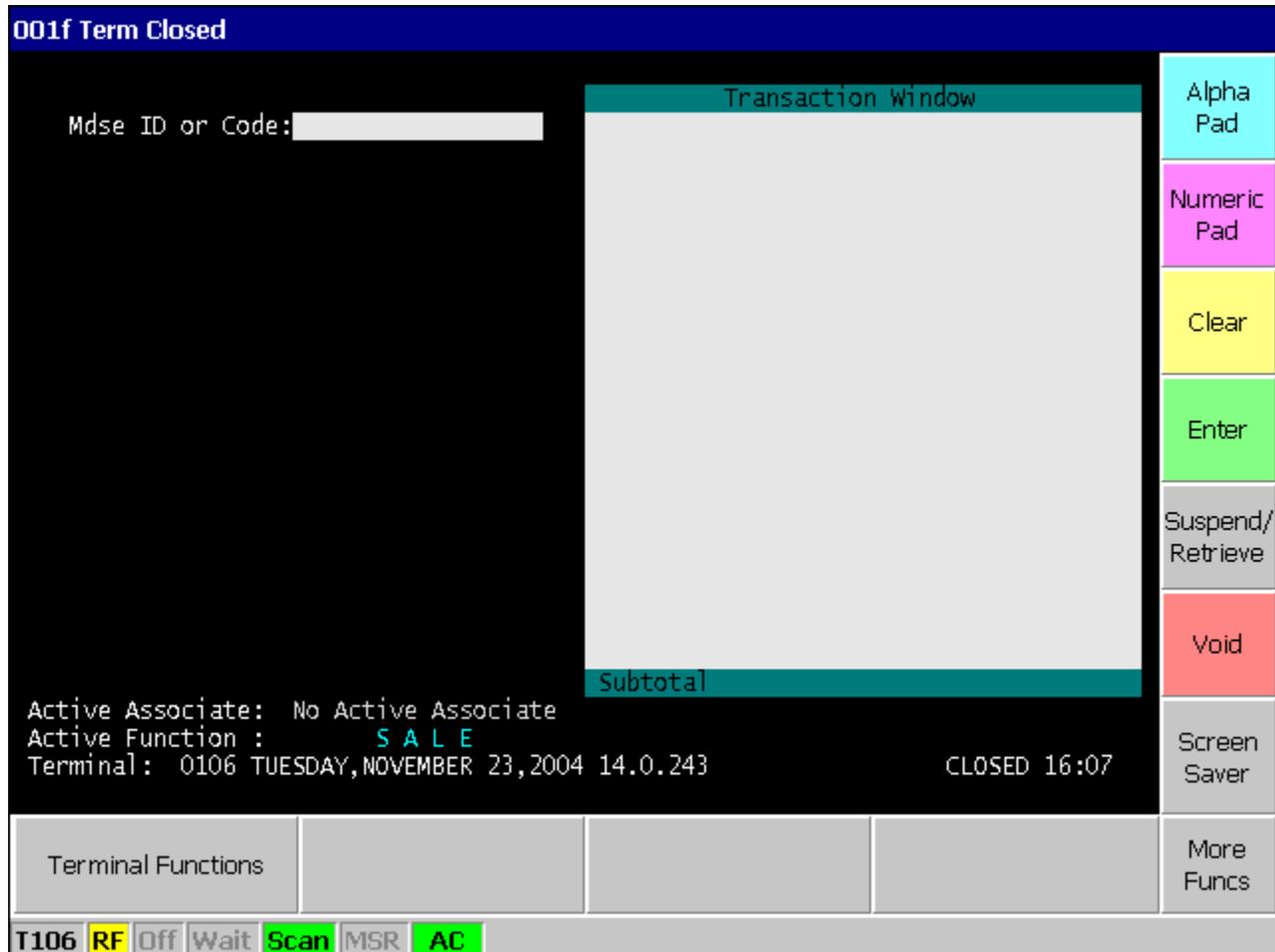
### 3.2 *Running the mPOS Application*

#### 3.2.1 [mPOS Screen Layout](#)

This is the opening screen for the PPT 8846/8146.



This is a snapshot from the MK2000, its screen is much larger allowing for more buttons to be displayed at once.



The screen contains the following components:

- 1) Title: The title displays at the top of the screen. A message displays on the left and the time displays on the right. Once the POS application is loaded the message will be the Input State Number and its associated text from the keymap file.
- 2) 2x20 Display: When the POS application is running, the 2x20 Operator Display messages are displayed. Informational messages also appear. For example, when the *mPOS* application starts, the *mPOS* application version number and other informational messages are displayed. When using the full-screen mode, the 2x20 display is much larger.
- 3) Application keys. Since all POS keyboard keys can't be mapped to the screen, a keymap file defines the keys to be displayed in each state of the application. Some sample states are as follows:
  - Sign On State
  - Item Entry State
  - Tender State
  - Account Number State

- 4) Status Bar: The status bar displays at the bottom of the screen and contains the following:
- a. Terminal Number
  - b. RF (TC Connection Status) When “RF” is red there is no wireless card present or the card is not talking to TC, it will be yellow if the connection signal is getting weak (warning user that they are about to be out of range), and green when a good signal is received and talking to TC.
  - c. Controller Status: Not yet implemented.
  - d. Keyboard Status: If “Wait” displays the keyboard is locked, blank when unlocked.
  - e. Scanner Status: “Scan” is green if the scanner is enabled or shaded gray if disabled.
  - f. MSR Status: “MSR” is green if enabled or shaded gray if disabled. If yellow then it is in retry mode and should then turn red or green. If red then the MSR device has a hard error and the device should be reset.
  - g. Battery Status: Remaining battery life in percent format is displayed. If battery life is 20% or less the display will be red. If the battery life is between 21% and 40% the display will be yellow. All other values are displayed with a shaded gray background.

### 3.2.2 *mPOS* Application Messages

The *mPOS* application will display a variety of user informative messages. The following table lists each of these messages, where it is located and a description.

Location	Message	Description
Status Bar	Txxx	Displayed in the Terminal number window where xxx is the terminal number assigned to the <i>mPOS</i> unit. Eg. T123 would be terminal 123.
Status Bar	RF	Displayed in the TC Connection Status window when the unit is offline or reconnecting to the TC PC.
Status Bar	“Off”, ”A”, or “a”	This status window is used to know if the device is in alpha mode or not. The capital “A” says that capitalization is on while “a” says that it is not. “Off” is not used at this time.
Status Bar	Wait	Displayed in the Keyboard status window when the keyboard is locked.
Status Bar	Scan	Displayed in the Scanner status window when the scanner is enabled.
Status Bar	MSR	Displayed in the MSR status window when the MSR is enabled.
Status Bar	xx%	Displayed in the Battery status window where xx is the battery status as a percentage value.
Status Bar	AC	Displayed in the Battery status window when the <i>mPOS</i> device is attached to a cradle and charging.
Title Bar	Terminal Sales	Displayed when the <i>mPOS</i> application is first started.
Title Bar	xxx...	Displayed on the title bar is the description information for each state defined in the keymap file.
2x20 Display	Version x.x.x	Displayed when the application is started is a version number. Eg. Version 2.5.1

2x20 Display Doesn't appear in new versions	PRINTER: Version xxx	When the <i>mPOS</i> application is loaded the printer driver version will be displayed on the second line.
2x20 Display Doesn't appear in new versions	CASH DRAWER: Version xxx	When the <i>mPOS</i> application is loaded the cash drawer driver version will be displayed on the second line.
2x20 Display Doesn't appear in new versions	MSR: Version xxx	When the <i>mPOS</i> application is loaded the MSR driver version will be displayed on the second line. Eg. Sym MSR3000 v2.0
2x20 Display Doesn't appear in new versions	CONFIG: Version xxx	When the <i>mPOS</i> application is loaded the Configuration driver version will be displayed on the second line. Eg. GSA Demo v1.0
2x20 Display	Connecting...	Displayed when the <i>mPOS</i> application is connecting to the TC PC.
2x20 Display	Reconnecting...	Displayed when the <i>mPOS</i> application is reconnecting to the TC PC. This will happen if the <i>mPOS</i> device is powered off and then on.
2x20 Display	Connection Failed...	Displayed when the <i>mPOS</i> unit loses connection with the TC PC.
2x20 Display	Loading Keymap...	Displayed when the application is started and loads the keymap file.
2x20 Display	CHECK PRINTER	Displayed if an error occurs while printing to the printer.
2x20 Display	REMOVE CUST RECEIPT <CLEAR> TO CONTINUE	Displayed when a CUT command is performed on a printer that does not physically CUT the receipt. This allows the user to tear the receipt before the next receipt is printed.
Online Receipt	----- CUT -----	A marker to show when a CUT command was received.
Online Receipt	----- INSERT -----	A marker to show that a document insert command was received.
Online Receipt	----- LOGO -----	A marker to show that a logo command was received.

### 3.2.3 [Starting the \*mPOS\* Application](#)

Launch the Mobile POS Application by pressing Start Menu and selecting the *mPOS* Application. If the auto-start option is on, soft boot the PPT 8846/8146 by holding down both the up and down yellow scroll buttons on the side of the device and pressing the FUNC button; the MK2000 can be cold booted by pressing the two middle hardware buttons. The Store Controller and the TC PC must be up and running.

The *mPOS* Application is a thin client and requires the Store Controller and the TC PC to load. Upon starting, a generic *mPOS* screen will display and messages will appear in the 2x20 Display.

- The *mPOS* application version displays.
- A series of informational messages display.
- “Connecting...” displays.
- If “Connection Failed” displays, see the problem determination section.
- “W008 LOADING TERMINAL” displays after the *mPOS* application has successfully

communicated with the TC application. The TC application is loading the POS application from the Store Controller. A sample TC screen follows. When a message displays on the TC 2x20 display for this *mPOS* terminal, it also displays on the *mPOS* 2x20 screen.



```
C:\TC_NT\BIN\tprocess.exe
=====T094
W008 PROGRAM
IS LOADING...
P1528-----L0002
                WAITING FOR POS DISPLAY
                WAITING FOR POS DISPLAY
                WAITING FOR POS DISPLAY
                WAITING FOR POS DISPLAY
```

#### Notes:

- 1) The text “T094” represents terminal number 94.
- 2) The text “P1528” represents the Process Identifier (PID) 1528 where 1528 is the numerical ID assigned to the process while it runs on the TC PC.
- 3) The text “L0002” represents the main log file that is used by this terminal if logging is enabled. The actual log file would be named TSF2.BIN and is located in the TC\_NT\LOG directory on the TC PC.

### 3.2.4 Running the *mPOS* Application

Once the POS application is loaded, the keying sequences are identical to the keying sequences on an IBM POS terminal. The operator may press the screen with a finger or the stylus. To scan, press one of the two yellow buttons on the PPT 8846/8146, in general the scanner should always be on with the MK2000.

To the store controller, the *mPOS* terminal is a regular POS terminal. Whatever rules apply to traditional POS terminals; the same rules apply to *mPOS* terminals. With GSA, the *m* terminal must be opened to run sales and it must be closed to run the store closing reports.

If the *mPOS* terminal is powered off and on, the *mPOS* application will reconnect to TC and resume the POS application.

If the *mPOS* terminal is soft reset, the *mPOS* application restarts and the POS application is reloaded.

### 3.2.5 [Buttons on the PPT 8846/8146 Terminal](#)



#### 3.2.5.1 *Scan Buttons*

The yellow buttons are the scan buttons. They are active when scanner input is allowed by the POS application.

#### 3.2.5.2 *Power Button*

The Power button is the red button on the lower left-hand side of the terminal. Press it to turn the device off and on. Turning it off does not stop the *mPOS* application. When the terminal is turned on, the *mPOS* application will reconnect to TC and resume the POS application.

#### 3.2.5.3 *Application Buttons*

The buttons on the bottom are the hardware buttons, they perform the same function through *mPOS* as what appears on their label. The number keys will enter in numbers on *mPOS*, letter will enter letters and so forth.

3.2.6 [MK2000 Layout](#)



3.2.6.1 **Power Off**

The only way to power off the device is to remove power from the device

3.2.7 [Turn the Manager Key On and Off](#)

The PPT 8846/8146 doesn't have a manager key. If the POS application requires the manager key to be turned on, the operator can scan the following barcodes to simulate the manager key. User programming can simulate the manager key other ways.

Scan MPOSM1 to turn the Manager key on, MPOSM0 to turn it off.



MPOSM1



MPOSM0

3.2.8 [Exit the \*mPOS\* Application](#)

Scan MPOSEXIT to exit the MPOS application.



### 3.2.9 Clearing the NVRAM for the *mPOS* terminal

NVRAM is stored in a disk file on the TC PC in the file \TC\_NT\BATPRMEM.DAT.

Two methods exist to clear NVRAM on the TC PC.

Method 1:

- 1) Press the Start Button on the Windows toolbar.
- 2) Select Terminal Concentrator
- 3) Select “Reload Terminal Configuration”

Method 2:

- 1) Stop TC
- 2) Delete \TC\_NT\BATPRMEM.DAT
- 3) Restart TC

## 4 Problem Determination

### 4.1 mPOS Application starts but the “Connection Fails”

Soft reset the Symbol device(s) and restart the *mPOS* application.

Ping the Symbol device(s) from the TC PC.

If ping fails, access the AP’s configuration screen and validate the device has associated with the AP.

If no association,

- Validate the ESSID in the Symbol device(s) matches the ESSID in the AP. ESSID is case sensitive.
- Turn off all other AP’s with the same ESSID to ensure the Symbol device(s) is not associating with another AP.
- Validate the AP and Symbol device(s) are compatible
  - Both are FH units or DS units.
  - If DS, verify the radio in the Symbol device(s) is version compatible with the AP’s version. Example, a V1.0 radio in a Symbol device(s) will not associate with an AP running V2.0.

If association,

- Validate the IP addresses and subnet masks in the TC PC and the Symbol device(s).
- Validate the Ethernet connections between the AP and the TC PC.
- If the AP’s IP address is set, try pinging the AP from the TC PC. Note: After setting the IP address in an AP, the AP must be rebooted for the IP address to take effect.

If pinging the Symbol device(s) from the TC PC works,

- Verify TC is active.
- Verify the LAN connections between the TC PC and the Store Controller.
- If the Store Controller has a TCP/IP address and only uses Ethernet, try pinging the Store Controller from the TC PC.
- Clear the Terminal’s CMOS. This is required if the Controller’s store number changes.
  - On the TC PC
    - Press the Start Button on the Windows toolbar.
    - Select Terminal Concentrator
    - Select “Reload Terminal Configuration”
  - Restart the *mPOS* application on the Symbol device(s).
- If the environment variable TSF\_STORE\_NUM exists on the TC PC, use the *mPOS* Configuration application to verify the Store# field is equal to TSF\_STORE\_NUM. If the field is non-zero, use the Controller to verify the TSF\_STORE\_NUM matches the store number in System Configuration. The System Configuration must be active. If not sure, activate the System Configuration and reboot the Controller.
- Verify the Store Controller is controlling the LAN.
  - On the Store Controller,
    - Press Alt-SysRq

- Press c to access the Store Control Functions Screen
- Select “2 Controller Functions”
- Select “1 Display Controller Status”
- Verify the LAN is configured as PRIMARY and the status is CONTROLLING. If PRIMARY but not CONTROLLING, try Resuming Store TCC Control which is under TCC Functions under Store Control Functions. If the LAN is not configured as PRIMARY, the Controller Configuration is incorrect.
- Verify the terminal number is defined as a LAN terminal.
  - On the Store Controller,
    - Access the System Main Menu
    - Select “4 Installation and Update Aids”
    - Select “1 Change Configuration Data”
    - Select “3 System Configuration”
    - Select “7 LAN Terminal Definition”
    - Select “4 Display the LAN Terminals”
    - Enter the terminal number and verify it is assigned to this controller.
    - The System Configuration must be active. If not sure, activate the System Configuration and reboot the Controller.
- Try using a 4694 or SurePOS with the same terminal number on the LAN.
  - If it doesn’t load, check the Store and Terminal Configurations.
- If using a DDS/CSF and running DDS/CSF and TC on the same PC, the Store Controller configuration must define store loop 1 as active. This becomes a “pseudo” store loop through which TC and CSF communicate. A store loop adapter is not required.

#### **4.2 The Symbol device(s) Powers Off Too Frequently**

Change the power settings on the Symbol device(s).

- Access the main screen on the Symbol device(s)
- Select Settings
- Select System
- Select Power
- Under “On Battery Power”, uncheck the box labeled “Turn off device if not used for”.
- Click OK
- This will keep the unit from powering off. It should run all day in this mode.

#### **4.3 The Comtec printer prints ‘not’ characters.**

Change the connection settings on the Symbol device(s).

- Access the main screen on the Symbol device(s)
- Select Settings
- Select Connections
- Select PC
- Uncheck the box labeled “Automatically synchronize when serial cable is connected, using”.
- Click OK

Appendix A. Sample Barcodes

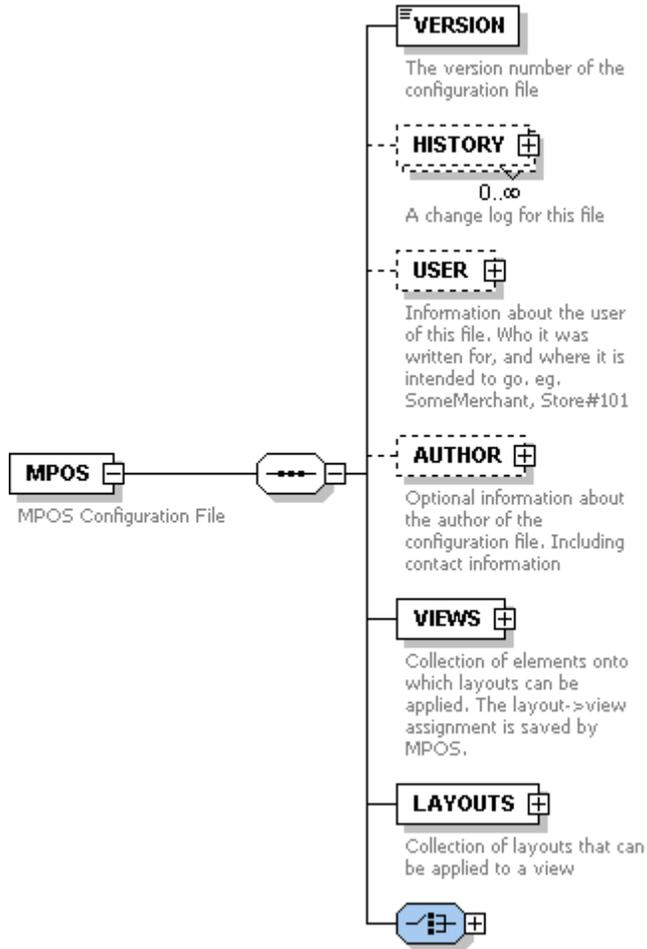
MPOS Demo Barcodes  
(Code128 barcodes)

Function	
Tracing ON Barcode Code128	 <b>MPOST9</b>
Tracing OFF Barcode Code128	 <b>MPOST0</b>
Enable Message Pump Probing Barcode Code128	 <b>MPOSTP1</b>
Disable Message Pump Probing Barcode Code128	 <b>MPOSTP0</b>
Exit the Application Barcode Code128	 <b>MPOSEXIT</b>
Load Keymap Barcode Code128	 <b>MPOSKeymap</b>

Appendix B. mPOS XML Configuration Schema

element **MPOS**

diagram



children [VERSION](#) [HISTORY](#) [USER](#) [AUTHOR](#) [VIEWS](#) [LAYOUTS](#) [TRANSITIONS](#)

annotation documentation **MPOS** Configuration File

element **MPOS/VERSION**

diagram



The version number of the configuration file

type **xs:string**

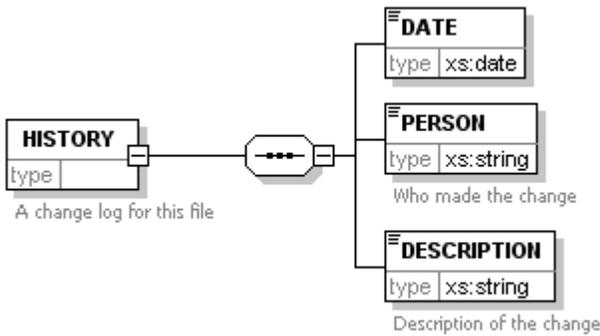
annotation documentation The version number of the configuration file

sample

<VERSION>3.0</VERSION>

element **MPOS/HISTORY**

diagram



A change log for this file

Description of the change

children [DATE PERSON DESCRIPTION](#)

annotation documentation A change log for this file

element **MPOS/HISTORY/DATE**

element **MPOS/HISTORY/PERSON**

diagram



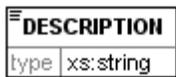
Who made the change

type **xs:string**

annotation documentation Who made the change

element **MPOS/HISTORY/DESCRIPTION**

diagram



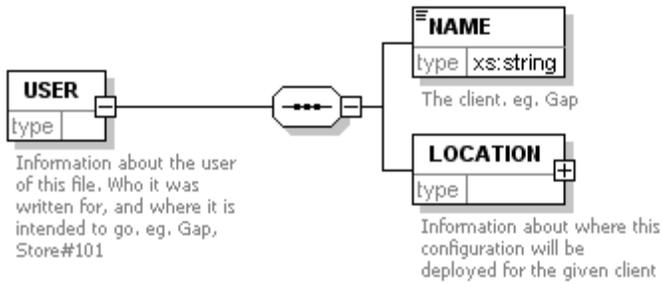
Description of the change

type **xs:string**

annotation documentation Description of the change

element **MPOS/USER**

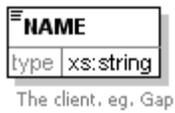
diagram



children [NAME](#) [LOCATION](#)  
 annotation documentation Information about the user of this file. Who it was written for, and where it is intended to go. eg. Gap, Store#101

element **MPOS/USER/NAME**

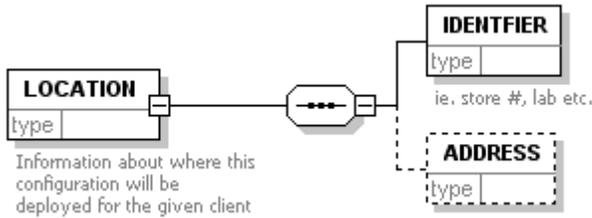
diagram



type **xs:string**  
 annotation documentation The client. eg. Gap

element **MPOS/USER/LOCATION**

diagram



children [IDENTIFIER](#) [ADDRESS](#)  
 annotation documentation Information about where this configuration will be deployed for the given client

element **MPOS/USER/LOCATION/IDENTIFIER**

diagram



annotation documentation ie. store #, lab etc.

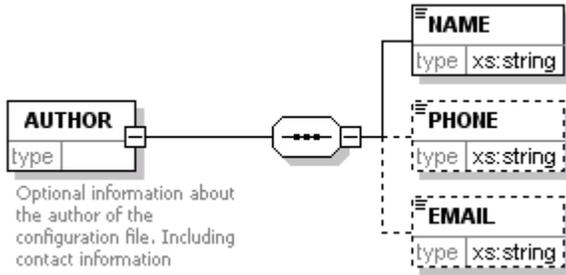
element **MPOS/USER/LOCATION/ADDRESS**

diagram



element **MPOS/AUTHOR**

diagram



children [NAME](#) [PHONE](#) [EMAIL](#)

annotation documentation Optional information about the author of the configuration file. Including contact information

element **MPOS/AUTHOR/NAME**

diagram



type **xs:string**

element **MPOS/AUTHOR/PHONE**

diagram



type **xs:string**

element **MPOS/AUTHOR/EMAIL**

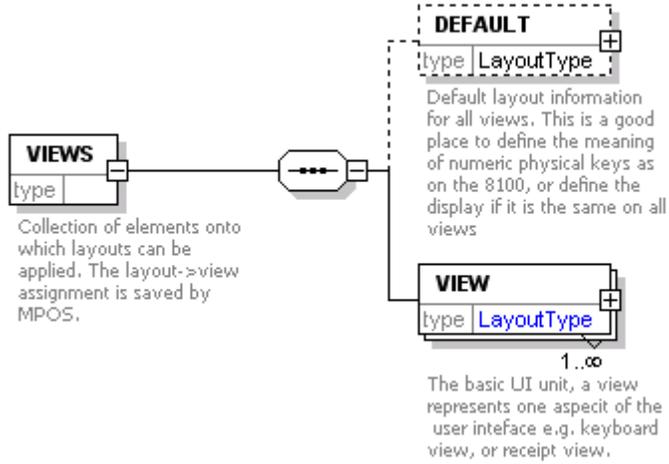
diagram



type **xs:string**

element **MPOS/VIEWS**

diagram



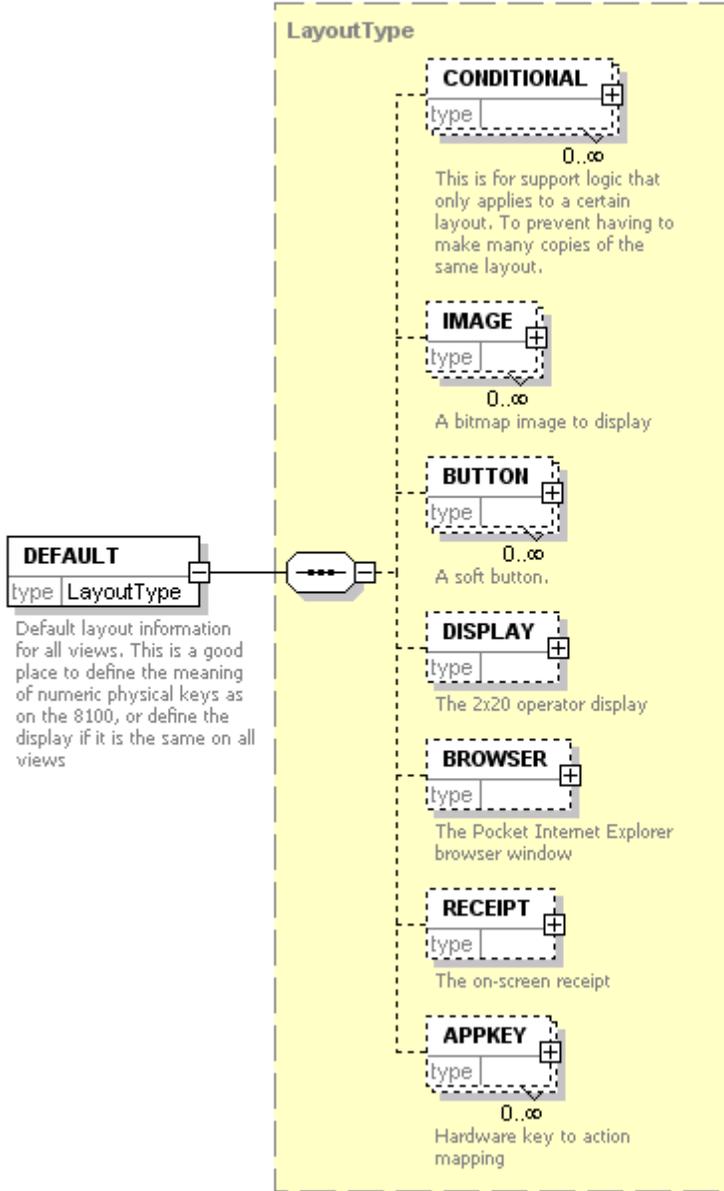
children [DEFAULT VIEW](#)

annotation documentation Collection of elements onto which layouts can be applied. The layout->view assignment is saved by **MPOS**.

documentation Collection of elements onto which layouts can be applied. The layout->view assignment is saved by **MPOS**.

element **MPOS/VIEWS/DEFAULT**

diagram



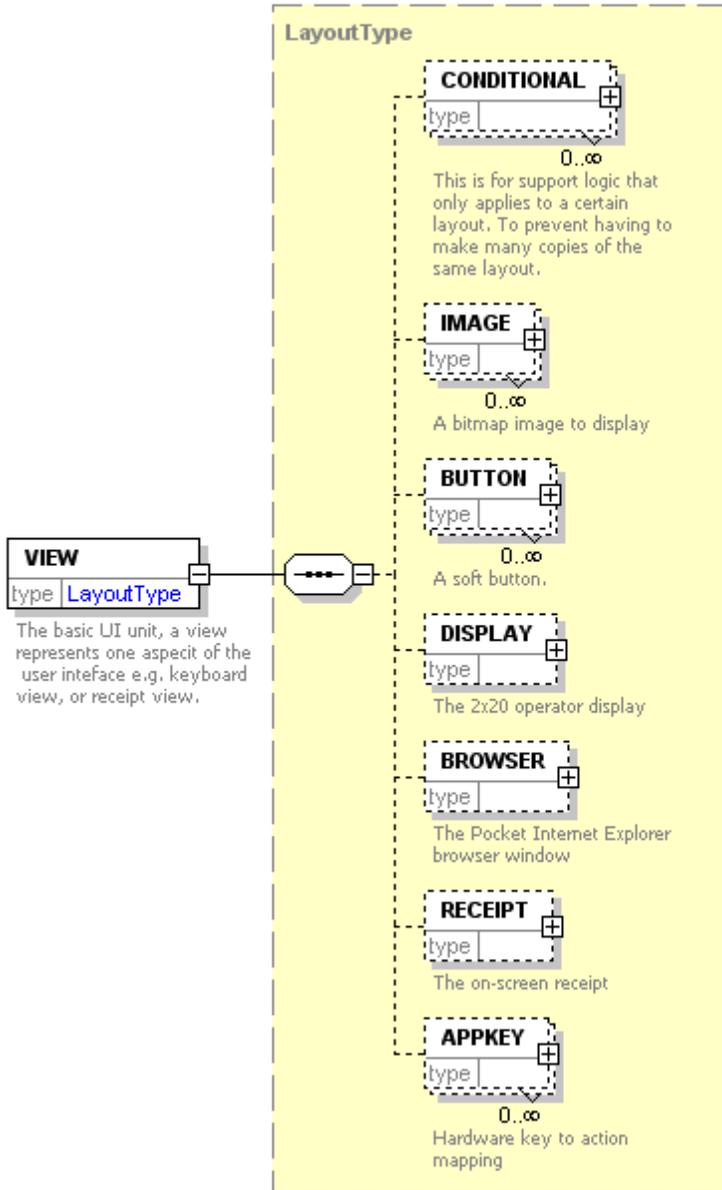
type [LayoutType](#)

children [CONDITIONAL](#) [IMAGE](#) [BUTTON](#) [DISPLAY](#) [BROWSER](#) [RECEIPT](#) [APPKEY](#)

annotation documentation Default layout information for all views. This is a good place to define the meaning of numeric physical keys as on the 8100, or define the display if it is the same on all views

element **MPOS/VIEWS/VIEW**

diagram



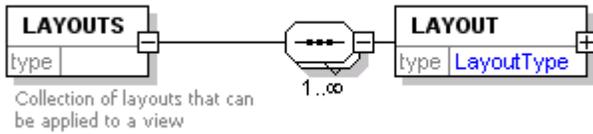
type extension of [LayoutType](#)

children [CONDITIONAL](#) [IMAGE](#) [BUTTON](#) [DISPLAY](#) [BROWSER](#) [RECEIPT](#) [APPKEY](#)

attributes	Name	Type	Use	Default	Fixed	Annotation
	viewID	xs:ID	required			
annotation	documentation	The basic UI unit, a view represents one aspect of the user interface e.g. keyboard view, or receipt view.				

element **MPOS/LAYOUTS**

diagram

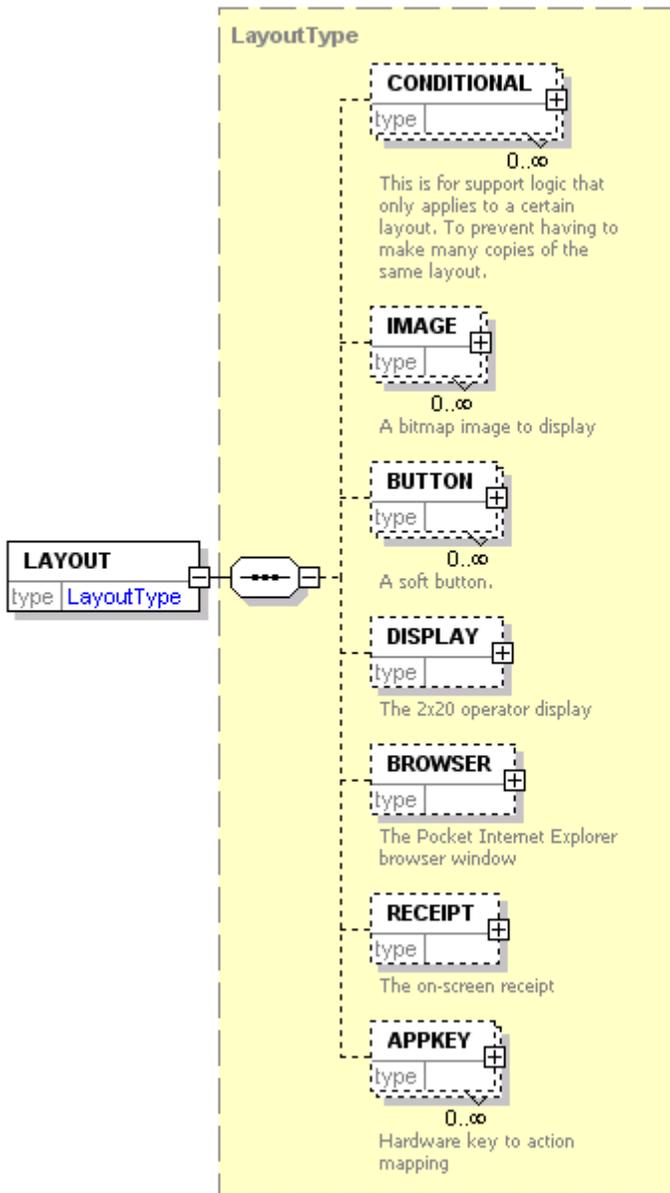


children [LAYOUT](#)

annotation documentation Collection of layouts that can be applied to a view

element **MPOS/LAYOUTS/LAYOUT**

diagram



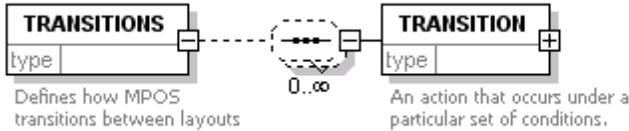
type extension of [LayoutType](#)

children [CONDITIONAL](#) [IMAGE](#) [BUTTON](#) [DISPLAY](#) [BROWSER](#) [RECEIPT](#) [APPKEY](#)

attributes	Name	Type	Use	Default	Fixed	Annotation
	layoutID	xs:ID	required			

element **MPOS/TRANSITIONS**

diagram

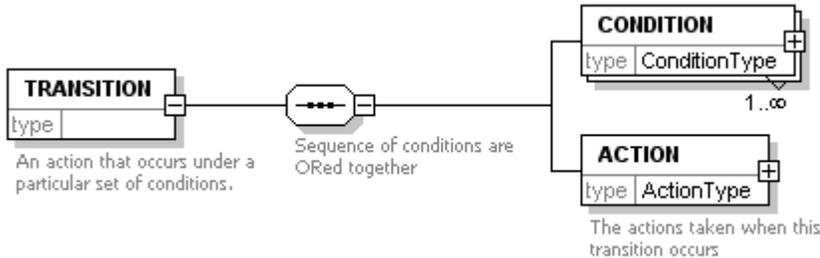


children [TRANSITION](#)

annotation documentation Defines how **MPOS** transitions between layouts

element **MPOS/TRANSITIONS/TRANSITION**

diagram



children [CONDITION](#) [ACTION](#)

annotation documentation An action that occurs under a particular set of conditions.

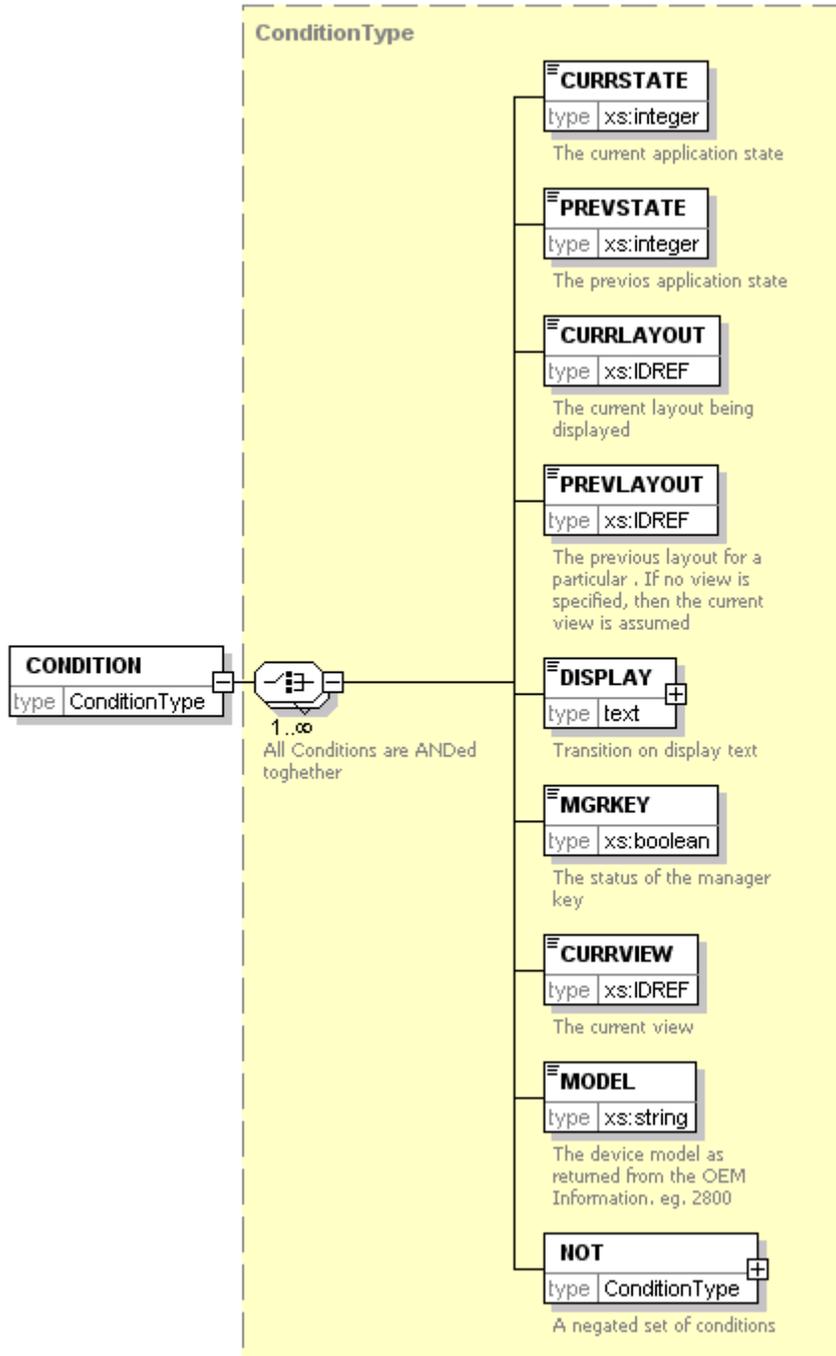
example

In this example, the startup state 0 is checked for in the condition. If the condition is met, the **VIEW** is set to “KEYBOARD”, the “RECEIPT\_STD” **LAYOUT** is applied to the “RECEIPT” **VIEW**. The **BROWSER** is told to navigate to the home page **URL**, and the application **TITLE** bar text is also set to “Please Wait...”. If multiple **CONDITION** tags are used then each tag is checked. If any one **CONDITION** is true, then the **ACTION** is performed, this means that the Condition Tag can be thought of as an OR statement.

```
<TRANSITION>
  <CONDITION>
    <CURRSTATE>0</CURRSTATE>
  </CONDITION>
  <ACTION>
    <VIEW viewID="KEYBOARD"/>
    <TITLE>Please Wait...</TITLE>
    <LAYOUT viewID="RECEIPT" layoutID="RECEIPT_STD"/>
    <URL>file://\application\mPOS_start.htm</URL>
  </ACTION>
</TRANSITION>
```

element **MPOS/TRANSITIONS/TRANSITION/CONDITION**

diagram



type [ConditionType](#)

children [CURRSTATE](#) [PREVSTATE](#) [CURRLAYOUT](#) [PREVLAYOUT](#) [DISPLAY](#) [MGRKEY](#) [CURRVIEW](#) [MODEL](#) [NOT](#)

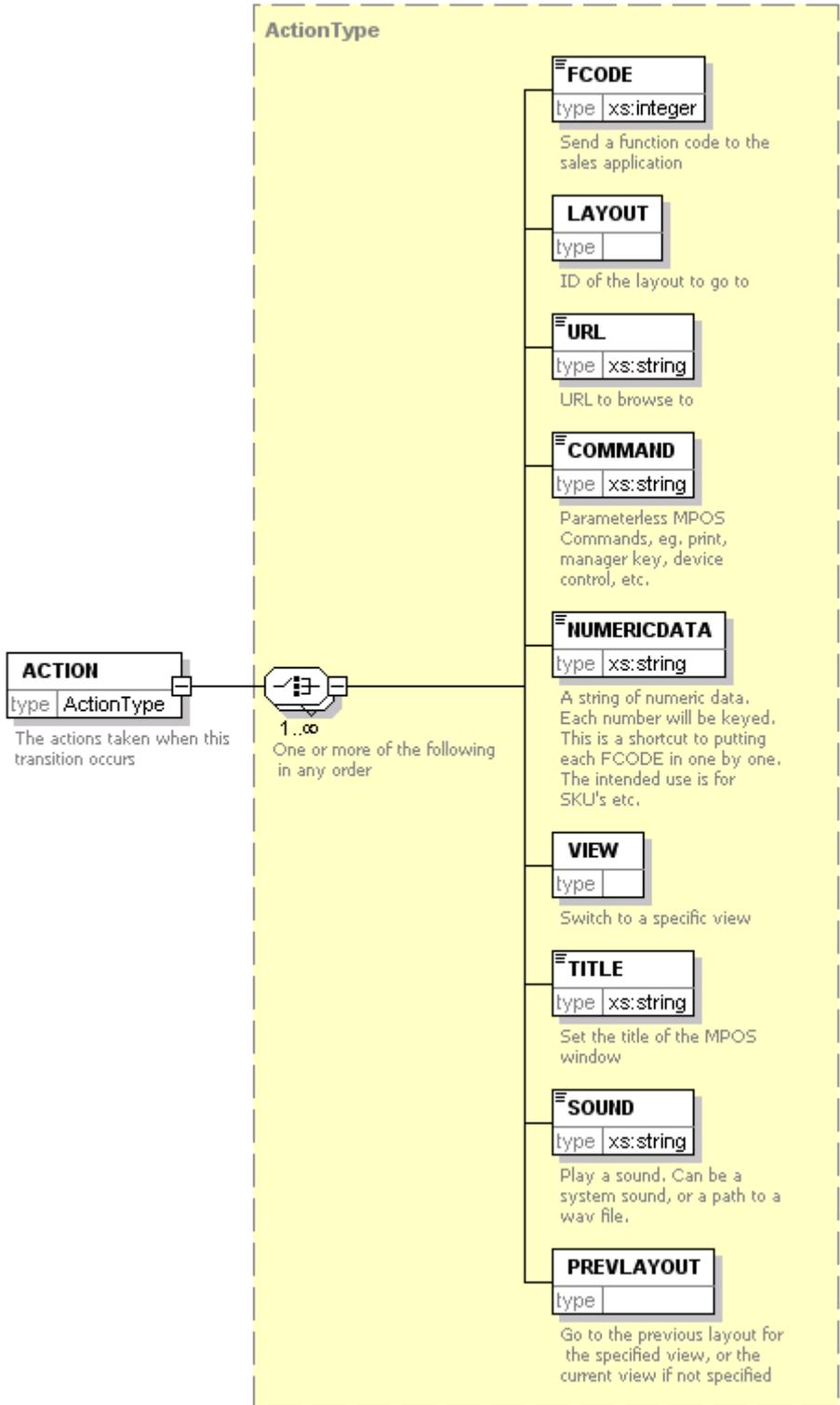
sample

```

<CONDITION>
  <CURRSTATE>1</CURRSTATE>
  <NOT>
    <CURRLAYOUT>GROUP1</CURRLAYOUT>
  </NOT>
</CONDITION>
    
```

element **MPOS/TRANSITIONS/TRANSITION/ACTION**

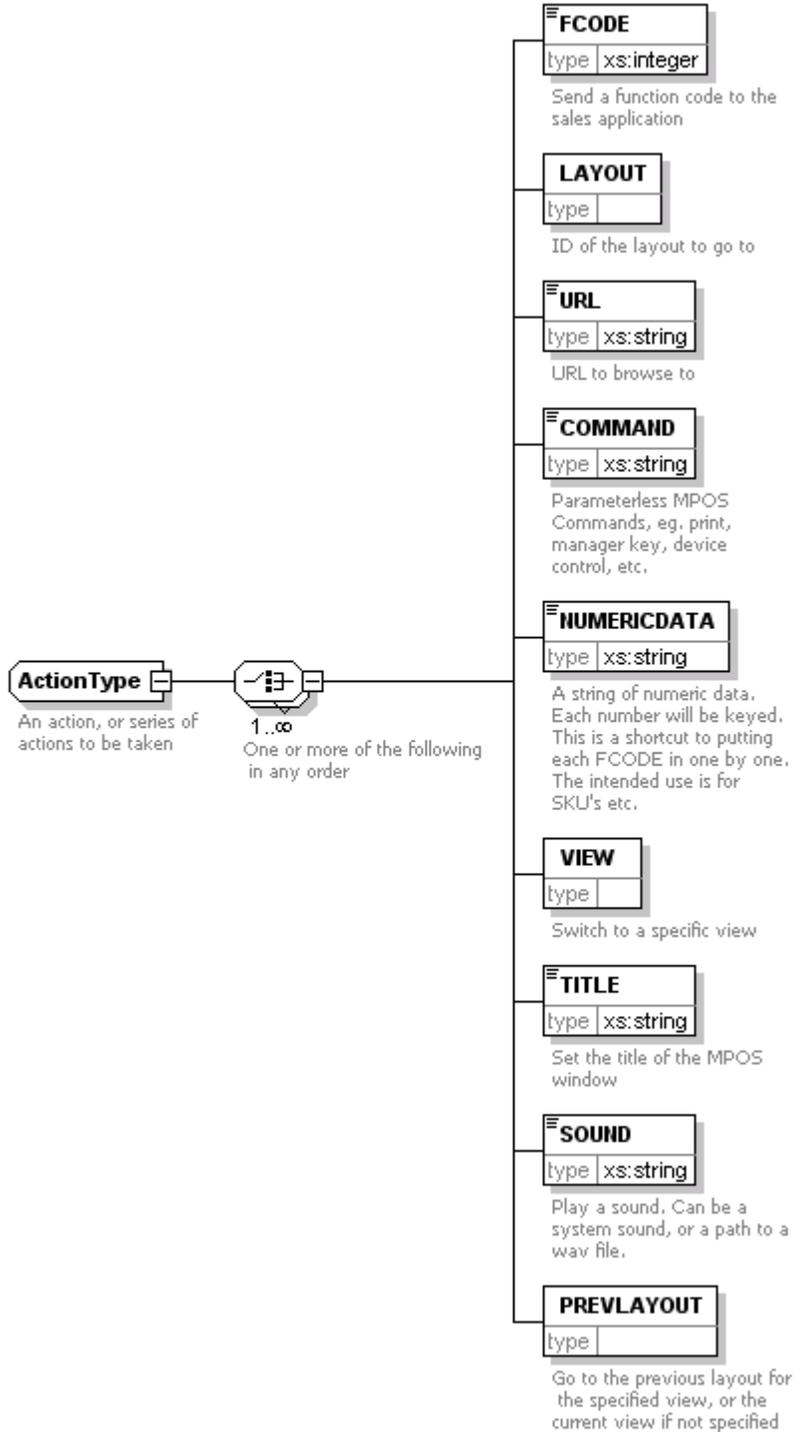
diagram



type [ActionType](#)

children [FCODE](#) [LAYOUT](#) [URL](#) [COMMAND](#) [NUMERICDATA](#) [VIEW](#) [TITLE](#) [SOUND](#) [PREVLAYOUT](#)  
 annotation documentation The actions taken when this transition occurs

complexType **ActionType**  
 diagram



children [FCODE](#) [LAYOUT](#) [URL](#) [COMMAND](#) [NUMERICDATA](#) [VIEW](#) [TITLE](#) [SOUND](#) [PREVLAYOUT](#)

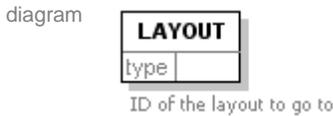
used by elements [MPOS/TRANSITIONS/TRANSITION/ACTION](#) [LayoutType/BUTTON/ACTION](#)  
[LayoutType/APPKEY/ACTION](#)  
 annotation documentation An action, or series of actions to be taken

**element ActionType/FCODE**



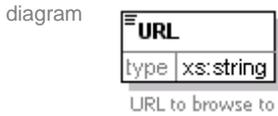
type **xs:integer**  
 annotation documentation Send a function code to the sales application

**element ActionType/LAYOUT**



attributes	Name	Type	Use	Default	Fixed	Annotation
	layoutID	xs:IDREF	optional			
	viewID	xs:IDREF	optional			
annotation	documentation	ID of the layout to go to				

**element ActionType/URL**



type **xs:string**  
 annotation documentation URL to browse to

**element ActionType/COMMAND**



type **xs:string**  
 annotation documentation Parameterless **MPOS** Commands, eg. print, manager key, device control, etc.

element **ActionType/NUMERICDATA**

diagram



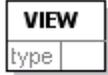
A string of numeric data. Each number will be keyed. This is a shortcut to putting each FCODE in one by one. The intended use is for SKU's etc.

type **xs:string**

annotation documentation A string of numeric data. Each number will be keyed. This is a shortcut to putting each FCODE in one by one. The intended use is for SKU's etc.

element **ActionType/VIEW**

diagram



Switch to a specific view

attributes	Name	Type	Use	Default	Fixed	Annotation
	viewID	xs:IDREF	required			
annotation	documentation	Switch to a specific view				

element **ActionType/TITLE**

diagram



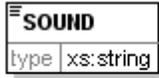
Set the title of the MPOS window

type **xs:string**

annotation documentation Set the title of the **MPOS** window

element **ActionType/SOUND**

diagram



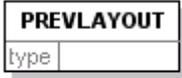
Play a sound. Can be a system sound, or a path to a wav file.

type **xs:string**

annotation documentation Play a sound. Can be a system sound, or a path to a wav file.

element **ActionType/PREVLAYOUT**

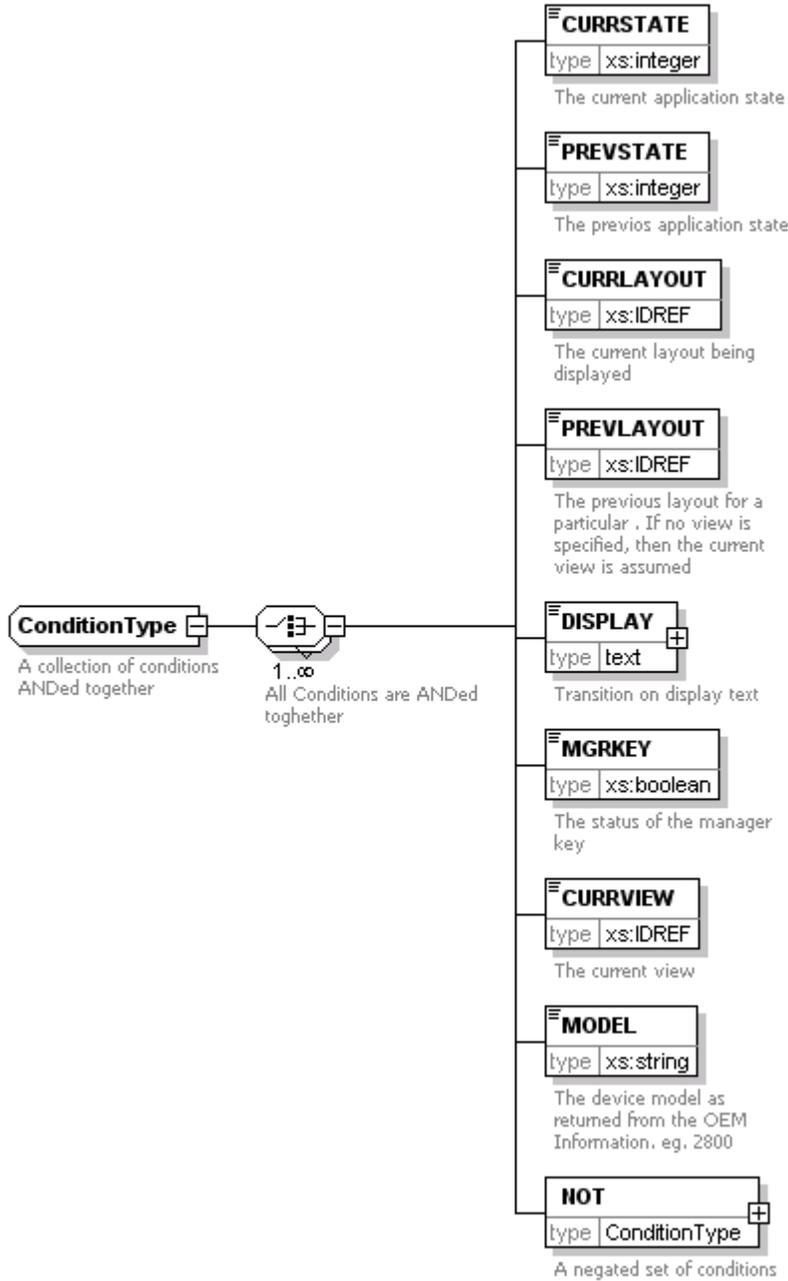
diagram



Go to the previous layout for the specified view, or the current view if not specified

attributes	Name	Type	Use	Default	Fixed	Annotation
	viewID	xs:IDREF	optional			
annotation	documentation	Go to the previous layout for the specified view, or the current view if not specified				

complexType **ConditionType**  
 diagram



children [CURRSTATE](#) [PREVSTATE](#) [CURRLAYOUT](#) [PREVLAYOUT](#) [DISPLAY](#) [MGRKEY](#) [CURRVIEW](#) [MODEL](#) [NOT](#)  
 used by elements [MPOS/TRANSITIONS/TRANSITION/CONDITION](#) [LayoutType/CONDITIONAL/CONDITION](#)  
 annotation documentation [ConditionType/NOT](#) A collection of conditions ANDed together

element **ConditionType/CURRSTATE**

diagram



The current application state

type **xs:integer**

annotation documentation The current application state

sample

<CURRSTATE>5</CURRSTATE>

element **ConditionType/PREVSTATE**

diagram



The previous application state

type **xs:integer**

annotation documentation The previous application state

sample

<PREVSTATE>5</PREVSTATE>

element **ConditionType/CURRLAYOUT**

diagram



The current layout being displayed

type **xs:IDREF**

annotation documentation The current layout being displayed

sample

<CURRLAYOUT>GROUP1</CURRLAYOUT>

element **ConditionType/PREVLAYOUT**

diagram



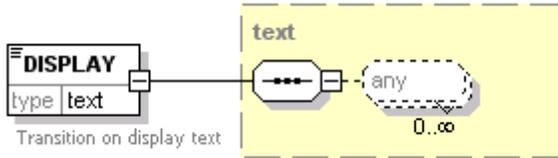
The previous layout for a particular . If no view is specified, then the current view is assumed

type **xs:IDREF**

annotation documentation The previous layout for a particular . If no view is specified, then the current view is assumed

element **ConditionType/DISPLAY**

diagram



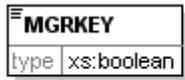
Transition on display text

type [text](#)

attributes	Name	Type	Use	Default	Fixed	Annotation
annotation	documentation		Transition on display text			
sample	<pre>&lt;DISPLAY&gt;CLOSE AT CONTROLLER&lt;/DISPLAY&gt;</pre>					

element **ConditionType/MGRKEY**

diagram



The status of the manager key

type **xs:boolean**

annotation	documentation	The status of the manager key			
sample	<pre>&lt;MGRKEY&gt;false&lt;/MGRKEY&gt;</pre>				

element **ConditionType/CURVIEW**

diagram



The current view

type **xs:IDREF**

annotation	documentation	The current view			
------------	---------------	------------------	--	--	--

element **ConditionType/MODEL**

diagram



The device model as returned from the OEM Information, eg. 2800

type **xs:string**

annotation	documentation	The device model as returned from the OEM Information. e.g. 2800			
------------	---------------	--	--	--	--

sample The model tag will perform a substring match against the platform information returned by the system. To view the platform information, follow the instructions to enable full tracing as described in the *mPOS* Configuration Utility's [Advanced Settings](#). Run the *mPOS* application, then locate the file qmp1.log. There will be a line at the beginning of the log similar to:

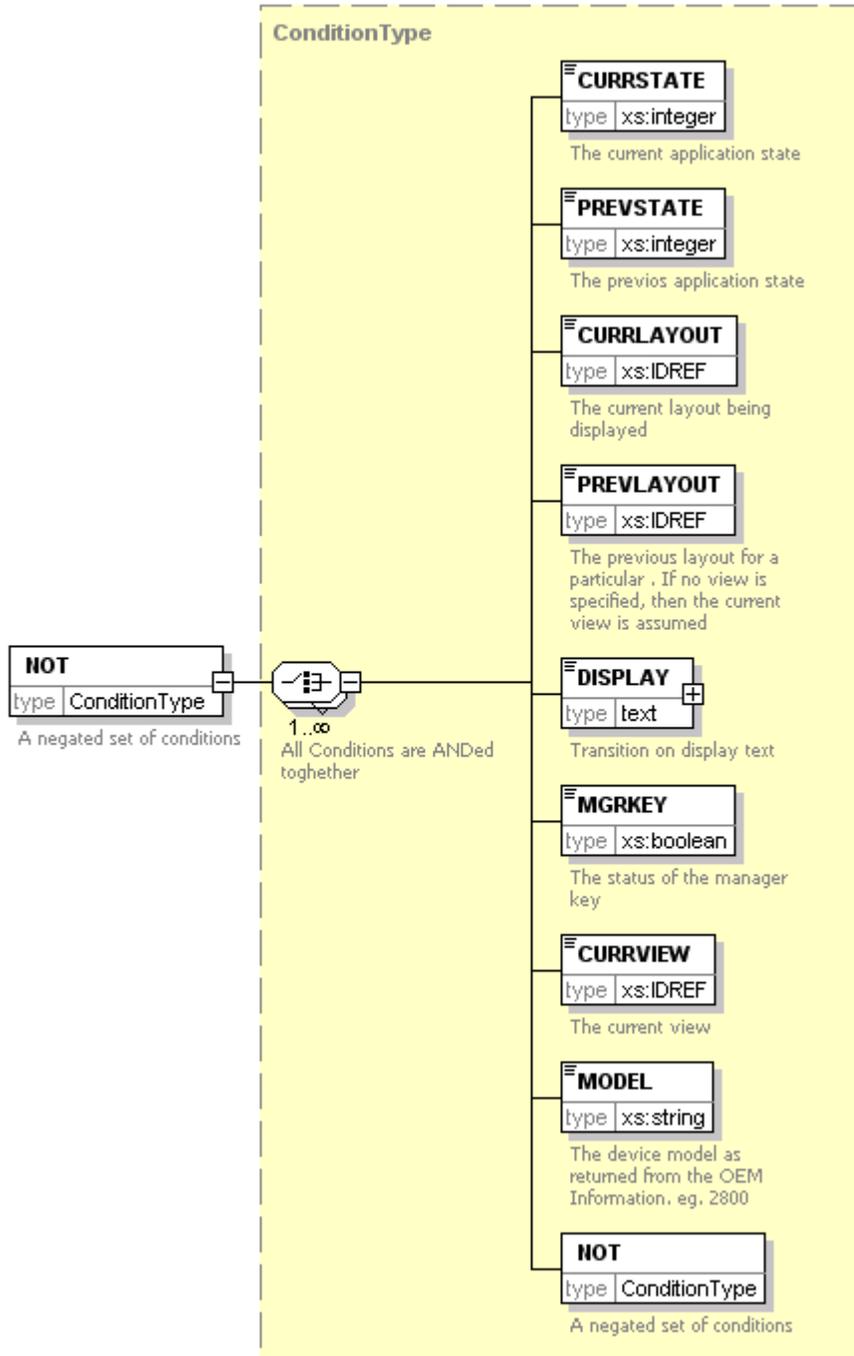
```
5542.211 1 CMposFrame::CMposFrame() - Platform <PPT8800>
```

So the corresponding model tag would be:

```
<MODEL>8800</MODEL>
```

element **ConditionType/NOT**

diagram



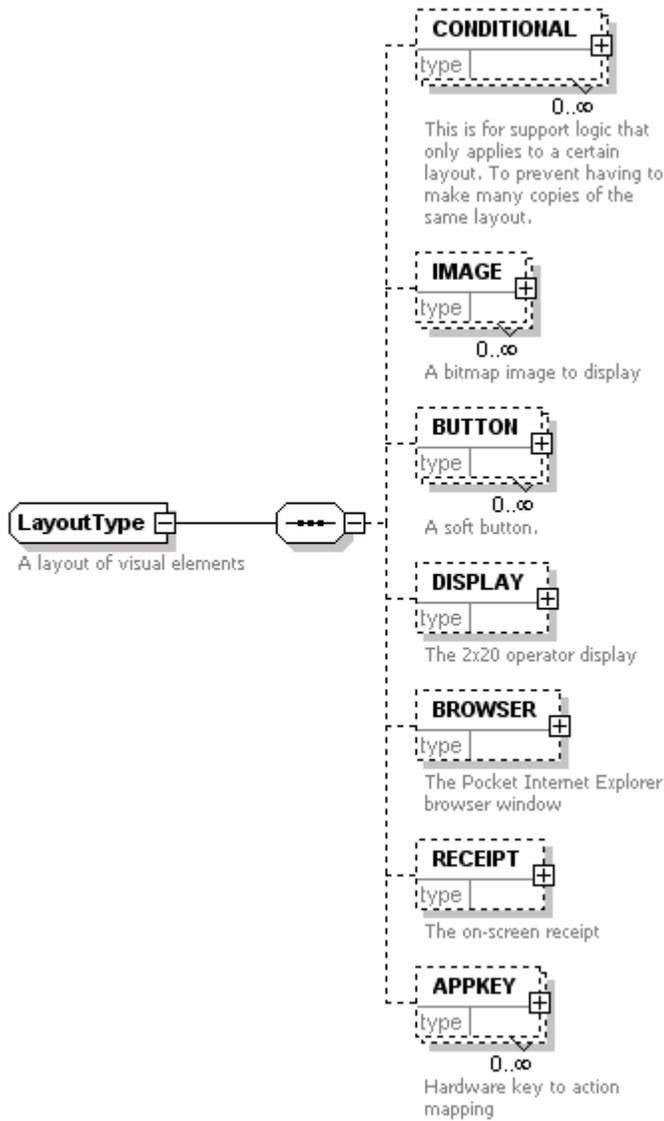
type [ConditionType](#)

children [CURRSTATE](#) [PREVSTATE](#) [CURRLAYOUT](#) [PREVLAYOUT](#) [DISPLAY](#) [MGRKEY](#) [CURRVIEW](#) [MODEL](#) [NOT](#)

annotation documentation A negated set of conditions

complexType **LayoutType**

diagram



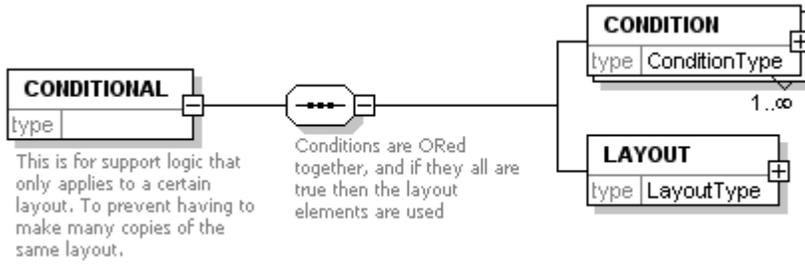
children [CONDITIONAL](#) [IMAGE](#) [BUTTON](#) [DISPLAY](#) [BROWSER](#) [RECEIPT](#) [APPKEY](#)

used by elements [MPOS/VIEWS/DEFAULT](#) [MPOS/LAYOUTS/LAYOUT](#) [LayoutType/CONDITIONAL/LAYOUT](#) [MPOS/VIEWS/VIEW](#)

annotation documentation A layout of visual elements

element **LayoutType/CONDITIONAL**

diagram

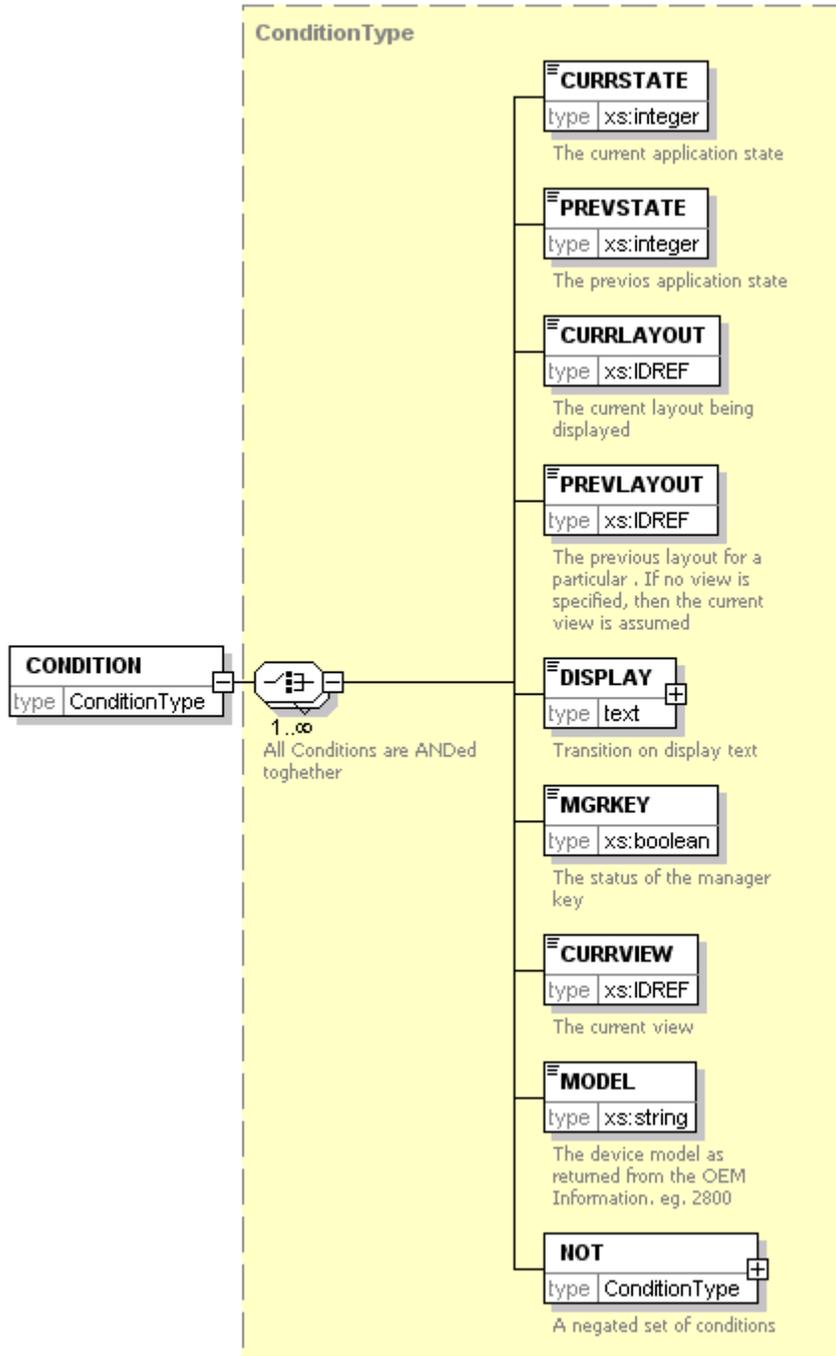


children [CONDITION](#) [LAYOUT](#)

annotation documentation This is for support logic that only applies to a certain layout. To prevent having to make many copies of the same layout.

element **LayoutType/CONDITIONAL/CONDITION**

diagram

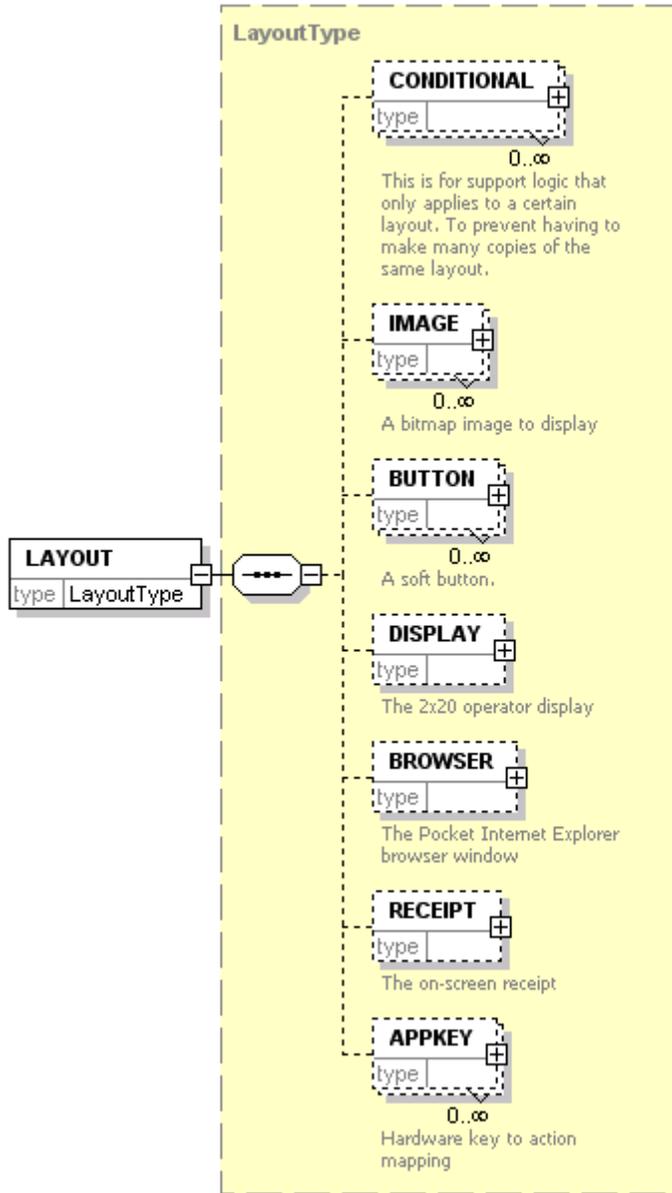


type [ConditionType](#)

children [CURRSTATE](#) [PREVSTATE](#) [CURRLAYOUT](#) [PREVLAYOUT](#) [DISPLAY](#) [MGRKEY](#) [CURRVIEW](#) [MODEL](#) [NOT](#)

element **LayoutType/CONDITIONAL/LAYOUT**

diagram

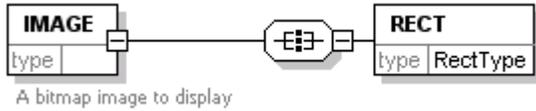


type [LayoutType](#)

children [CONDITIONAL](#) [IMAGE](#) [BUTTON](#) [DISPLAY](#) [BROWSER](#) [RECEIPT](#) [APPKEY](#)

element **LayoutType/IMAGE**

diagram



children

[RECT](#)

attributes	Name	Type	Use	Default	Fixed	Annotation
	SRC	xs:string	required			
annotation	documentation	A bitmap image to display				

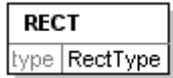
sample

```

<IMAGE SRC="mPOS_background.bmp">
  <RECT L="0" T="13" R="240" B="286"/>
</IMAGE>
    
```

element **LayoutType/IMAGE/RECT**

diagram



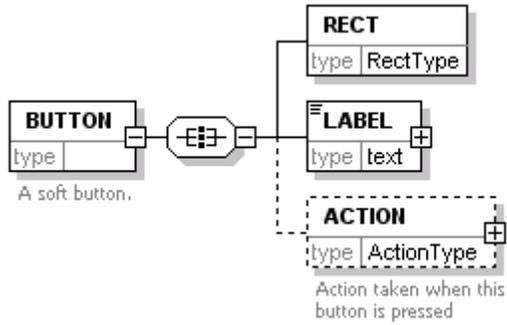
type

[RectType](#)

attributes	Name	Type	Use	Default	Fixed	Annotation
	L	xs:integer	required			
	T	xs:integer	required			
	R	xs:integer	required			
	B	xs:integer	required			

element **LayoutType/BUTTON**

diagram



children [RECT LABEL ACTION](#)

attributes	Name	Type	Use	Default	Fixed	Annotation
	ONBACKCOLOR	xs:hexBinary	optional			
	OFFBACKCOLOR	xs:hexBinary	optional			
	ONTEXTCOLOR	xs:hexBinary	optional			
	OFFTEXTCOLOR	xs:hexBinary	optional			
	BOLD	xs:boolean	optional			
	FONTSIZE	xs:string	optional			
	FONTNAME	xs:string	optional			
annotation	documentation	A soft button.				

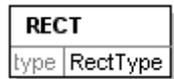
sample

```

<BUTTON OFFBACKCOLOR="0080FFFF" ONBACKCOLOR="0000FFFF">
  <RECT L="1" T="63" R="47" B="106"/>
  <LABEL>Clear</LABEL>
  <ACTION>
    <FCODE>73</FCODE>
  </ACTION>
</BUTTON>
    
```

element **LayoutType/BUTTON/RECT**

diagram



type [RectType](#)

attributes	Name	Type	Use	Default	Fixed	Annotation
	L	xs:integer	required			
	T	xs:integer	required			
	R	xs:integer	required			
	B	xs:integer	required			

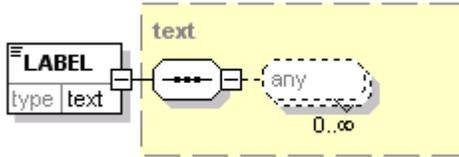
sample

```

<RECT L="1" T="63" R="47" B="106"/>
    
```

element **LayoutType/BUTTON/LABEL**

diagram



type [text](#)

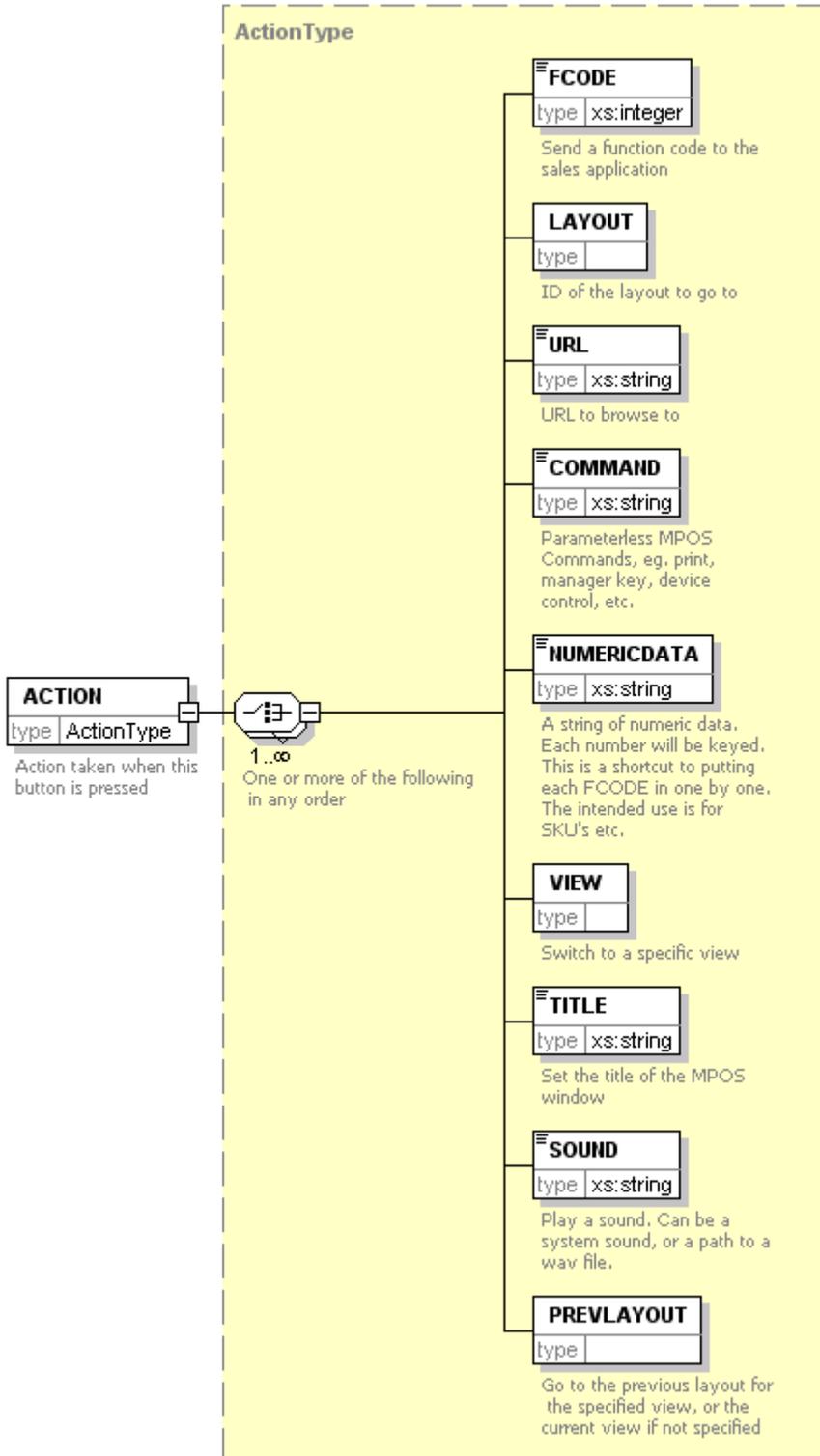
attributes	Name	Type	Use	Default	Fixed	Annotation
------------	------	------	-----	---------	-------	------------

sample

```
<LABEL>Clear</LABEL>
```

element **LayoutType/BUTTON/ACTION**

diagram



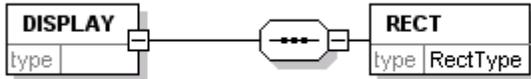
type [ActionType](#)

children [FCODE](#) [LAYOUT](#) [URL](#) [COMMAND](#) [NUMERICDATA](#) [VIEW](#) [TITLE](#) [SOUND](#) [PREVLAOUT](#)

annotation documentation Action taken when this button is pressed

element **LayoutType/DISPLAY**

diagram



The 2x20 operator display

children [RECT](#)

attributes	Name	Type	Use	Default	Fixed	Annotation
	BACKCOLOR	xs:hexBinary	optional	00000000		00BBGGRR
	TEXTCOLOR	xs:hexBinary	optional	0000FF00		00BBGGRR
annotation	documentation	The 2x20 operator display				

element **LayoutType/DISPLAY/RECT**

diagram

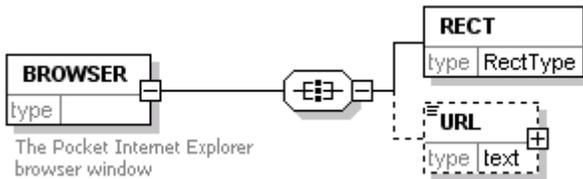


type [RectType](#)

attributes	Name	Type	Use	Default	Fixed	Annotation
	L	xs:integer	required			
	T	xs:integer	required			
	R	xs:integer	required			
	B	xs:integer	required			

element **LayoutType/BROWSER**

diagram



The Pocket Internet Explorer browser window

children [RECT](#) [URL](#)

annotation documentation The Pocket Internet Explorer browser window

element **LayoutType/BROWSER/RECT**

diagram

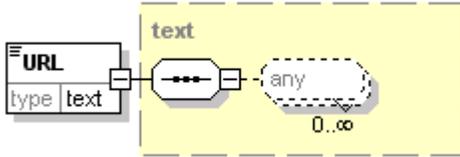


type [RectType](#)

attributes	Name	Type	Use	Default	Fixed	Annotation
	L	xs:integer	required			
	T	xs:integer	required			
	R	xs:integer	required			
	B	xs:integer	required			

element **LayoutType/BROWSER/URL**

diagram

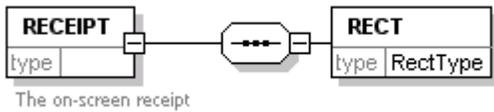


type [text](#)

attributes	Name	Type	Use	Default	Fixed	Annotation
------------	------	------	-----	---------	-------	------------

element **LayoutType/RECEIPT**

diagram



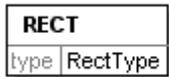
The on-screen receipt

children [RECT](#)

annotation documentation The on-screen receipt

element **LayoutType/RECEIPT/RECT**

diagram



type [RectType](#)

attributes	Name	Type	Use	Default	Fixed	Annotation
	L	xs:integer	required			
	T	xs:integer	required			
	R	xs:integer	required			
	B	xs:integer	required			

element **LayoutType/APPKEY**

diagram



Hardware key to action mapping

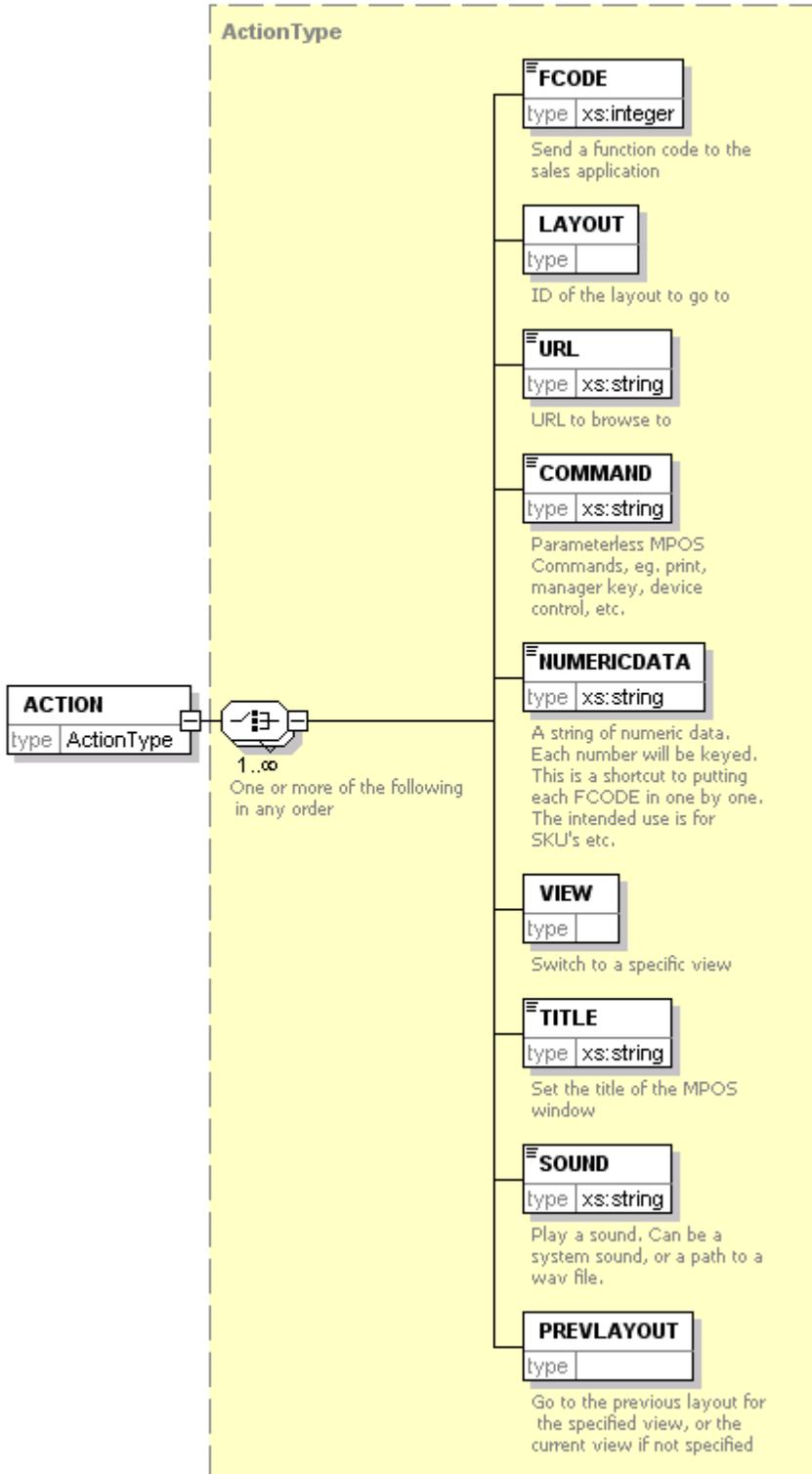
children [ACTION](#)

attributes	Name	Type	Use	Default	Fixed	Annotation
	Vkey	xs:hexBinary	required			

annotation documentation Hardware key to action mapping

element **LayoutType/APPKEY/ACTION**

diagram



type [ActionType](#)

children [FCODE](#) [LAYOUT](#) [URL](#) [COMMAND](#) [NUMERICDATA](#) [VIEW](#) [TITLE](#) [SOUND](#) [PREVLAYOUT](#)

**complexType RectType**



used by elements [LayoutType/IMAGE/RECT](#) [LayoutType/BUTTON/RECT](#) [LayoutType/DISPLAY/RECT](#) [LayoutType/BROWSER/RECT](#) [LayoutType/RECEIPT/RECT](#)

attributes	Name	Type	Use	Default	Fixed	Annotation
	L	xs:integer	required			
	T	xs:integer	required			
	R	xs:integer	required			
	B	xs:integer	required			
annotation	documentation	A screen rectangle				

**complexType text**



type restriction of **xs:anyType**

used by elements [ConditionType/DISPLAY](#) [LayoutType/BUTTON/LABEL](#) [LayoutType/BROWSER/URL](#)

attributes	Name	Type	Use	Default	Fixed	Annotation
annotation	documentation	A replacement for the internal string type				

XML Schema documentation generated with Error! Hyperlink reference not valid. Schema Editor Error! Hyperlink reference not valid.

*Appendix C. Acronyms*

4610	An IBM 4610 thermal printer.
4690	The IBM 4690 Operating System
AP	Symbol Access Point
CB	Symbol Client Bridge
CSF	DDS/CSF
DDS/CSF	IBM Distributed Data Services with the Controller Services Facility
GSA	IBM 4690 General Sales Application
<b>mPOS</b>	Mobile Point of Sale
MSAS	Microsoft Active Sync
MSR	Magnetic Stripe Reader
PC	Personal Computer
POS	Point of Sale
TC	Terminal Concentrator
TCC	Terminal Controller Communications, the communications protocol between IBM LAN based POS and the IBM Store Controller or between TC and the IBM 4690 Store Controller

***Appendix D. Related Publications***

*mPOS* Programmer's Guide

Symbol PPT 8846/8146 Series QRG (Quick Reference Guide)

Symbol MSR 3000 Magnetic Stripe Reader QRG (Quick Reference Guide)

QVS 4690 Terminal Concentrator for Windows User Guide

QVS Software Mobile POS Developer's Guide

Symbol Spectrum24 CB1000 User Guide

IBM 4680-4690 General Sales Application: Programming Guide

IBM 4680-4690 General Sales Application: Guide to Operations