

Integral Enterprise Viewer™
A member of the
Integral Enterprise Data Access Series™

Sample Solutions

Accessing Integral from SyteLine Right-Click Programs



©Copyright 2002, Decision Resources, Inc. All rights reserved.

“Integral Enterprise Viewer” and “Integral Enterprise Data Access Series” are trademarks of Decision Resources, Inc.

I. Accessing Integral from SyteLine Right-Click Programs Example

Problem: A customer running SyteLine wants to provide immediate access to certain Integral item inquiries from their item master screen – selecting data on the Integral inquiries based on the item currently displayed in SyteLine.

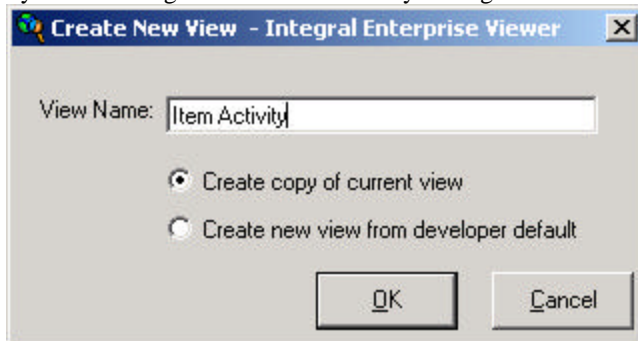
Solution: The user creates Views in Integral for each inquiry they wish to access from SyteLine. Then they add a SyteLine UDFA programs menu from the Description Field on the SyteLine Item Header view. They add a right click program for each Integral Inquiry they want to access.

The result is an easy ability to access any Integral Inquiry (including custom ones they wrote) from any SyteLine screen. The result might involve invoking and display an Integral inquiry, or might be running an Integral inquiry directly to an output option - such as an Excel Spreadsheet or a JetForm output.

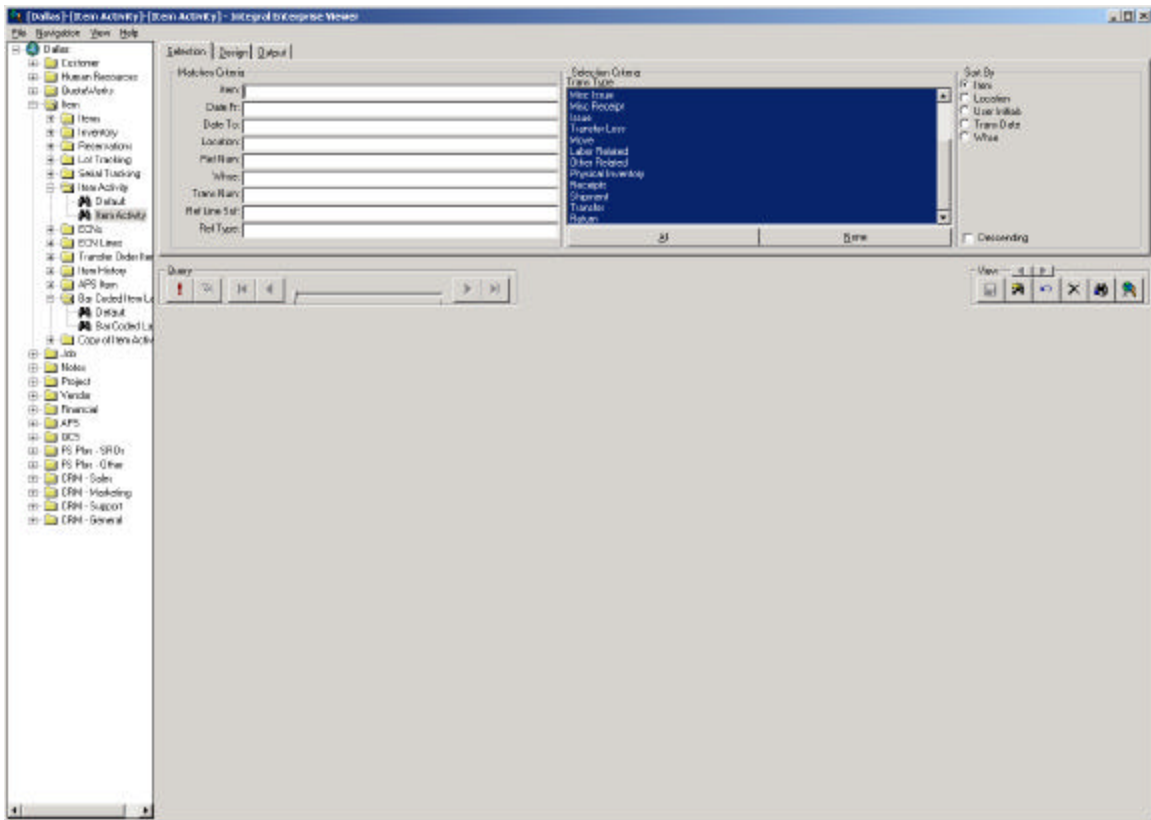
Detailed Screen Shots of the Solution:

In our example we will provide an ability to run two programs for each SyteLine item master display. First, we will permit users to Display the Integral Item Activity Inquiry for all material transactions locations for the SyteLine item. We will also provide an option that automatically prints bar coded stock labels for all locations for the item to a PDF file via JetForms (more details on setting up the Integral/JetForms interface are describe in another Solution).

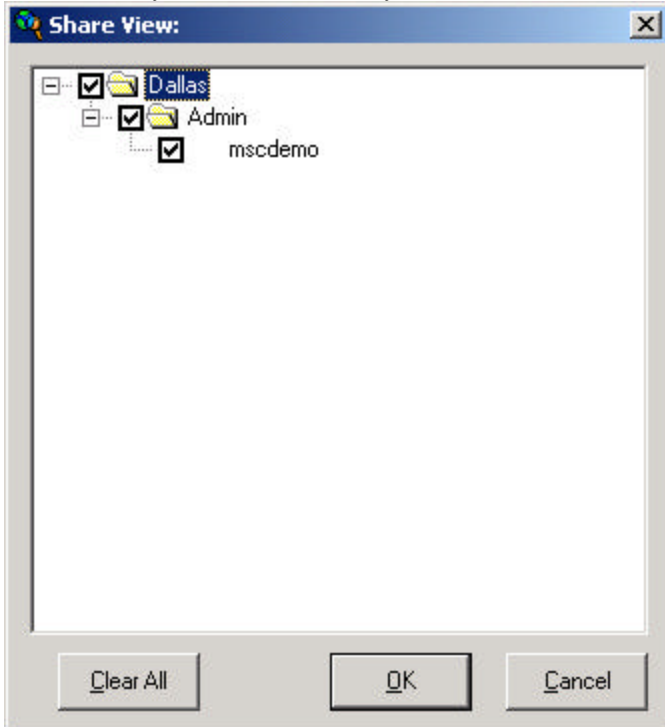
a. First call up the Integral LAN Client. Create new Views for the Inquiries we want to call from SyteLine through “View – New” or by hitting the ‘Create new view’ binoculars icon:



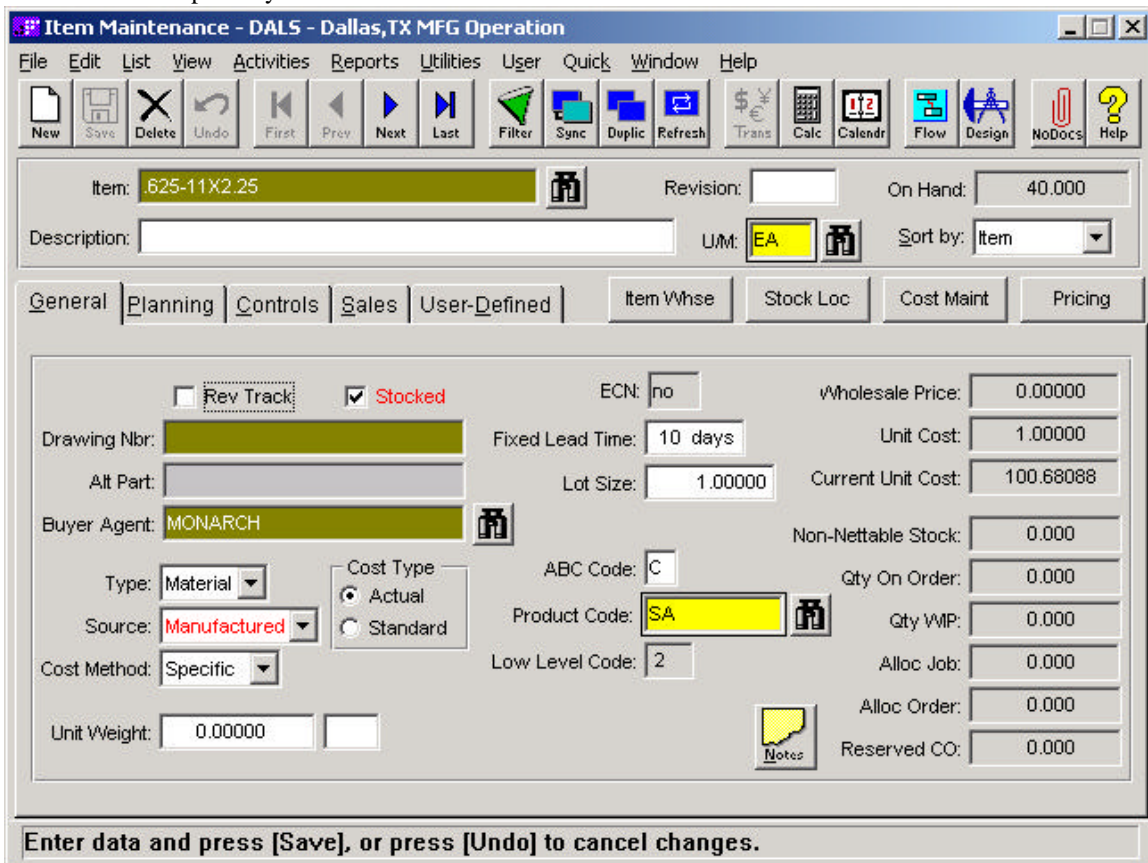
Here is the Integral Enterprise Viewer Tree view after the new Views (named “Item Activity” and “Bar Coded Item Labels” were added):



Note – new views, like those created above begin by being accessible only to the Integral user who created them. Integral includes a capability that allows a user who creates a new view, like the two above, to share them with other users, so they can use the same custom Views. This is accomplished by hitting “View – Share” or by clicking the “Share” button (rightmost binocular icon in the share group). Select the names of other users at your site, with which you want to share the view(s) you just created.



b. Next we call up the SyteLine Item Maintenance Screen:



Item: .625-11X2.25 Revision: On Hand: 40.000

Description: U.M.: EA Sort by: Item

General Planning Controls Sales User-Defined Item Whse Stock Loc Cost Maint Pricing

Rev Track Stocked ECN: no Wholesale Price: 0.00000

Drawing Nbr: Fixed Lead Time: 10 days Unit Cost: 1.00000

Alt Part: Lot Size: 1.00000 Current Unit Cost: 100.68088

Buyer Agent: MONARCH Non-Nettable Stock: 0.000

Type: Material Cost Type: Actual Standard ABC Code: C Qty On Order: 0.000

Source: Manufactured Product Code: SA Qty WIP: 0.000

Cost Method: Specific Low Level Code: 2 Alloc Job: 0.000

Unit Weight: 0.00000 Alloc Order: 0.000

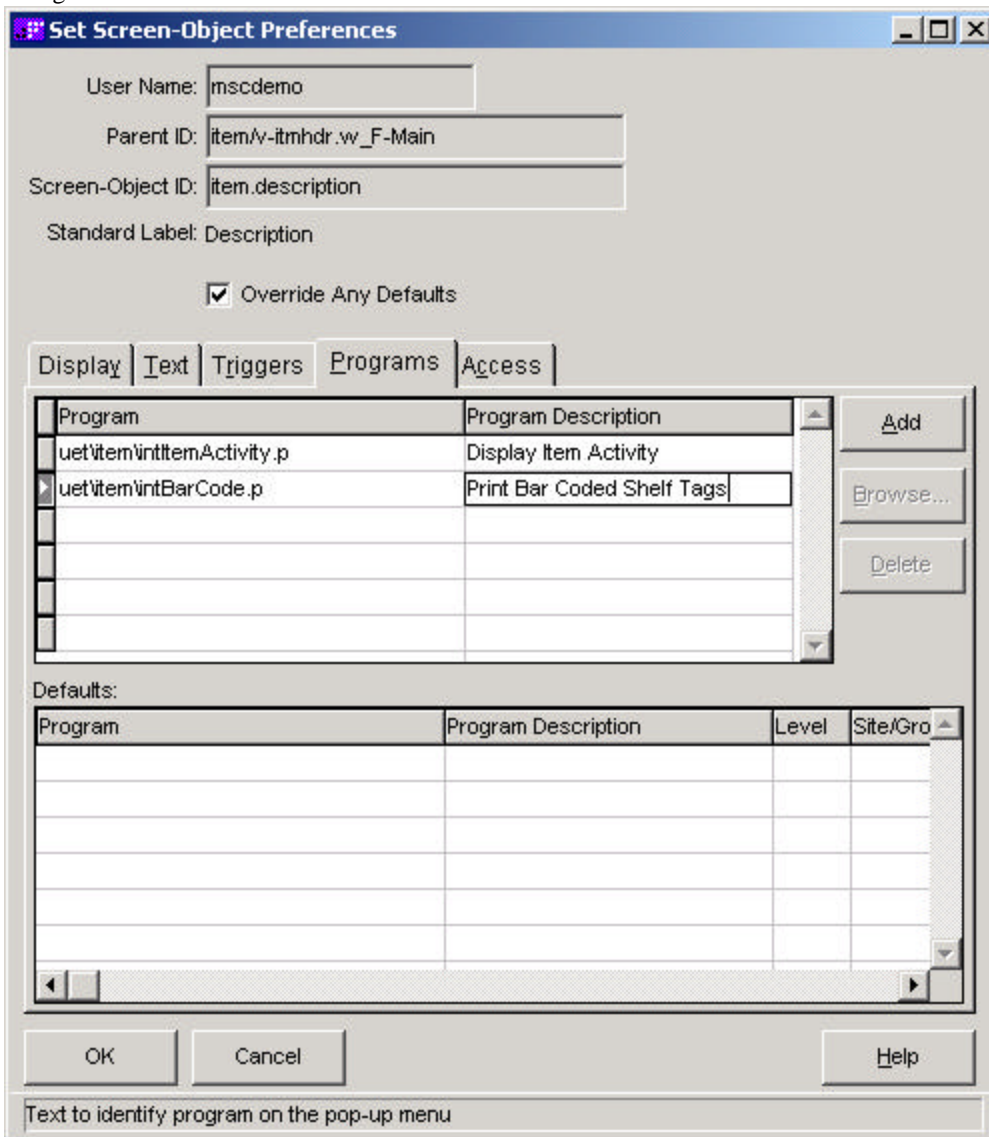
Reserved CO: 0.000

Enter data and press [Save], or press [Undo] to cancel changes.

Go to the Description field and hit “Help – Field Schema Information.” Note the help message indicates that the description field is implemented in “item/v-itemhdr.w.”

c. Hit the SyteLine Design Button.

Double-click the Description Field to invoke the “Set Screen Object Preferences” and choose the “Programs” tab:



Note that we have set up two right-click programs.

d. Now we write our two Progress right-click programs to invoke Integral. The technique that we employ assumes that the Windows Scripting Host has been installed on the Client workstation (That is recommended for PC's running Integral. If that is not installed on a machine, the Progress programs could also invoke Integral through older/lower level Windows API calls: WinExec or ShellExecute, or via the Progress OS-COMMAND capability).

Here is the code entered into two Progress programs we created: intItemActivity.p and intBarcode.p:

```

Procedure Editor - D:\CSRP\Mscmods\uet\item\intItemActivity.p
File Edit Search Buffer Compile Tools Options Help

/* intItemActivity.p */
{lib/trig-def.i &OBJECT=ITEM/v-itmhdr}
DEF VAR strCommand AS CHARACTER.
strCommand = ""D:\CSRP\symixplus\DRI\IEV\LANClient\Integral Enterprise Viewer.exe"" +
    " /vn:Item Activity /r:-2 /on:Item," + trim(ITEM.ITEM).
DEF VAR ch AS COM-HANDLE.
CREATE "wscript.shell" ch.
ch:RUN(strCommand, 1).

```

```

/* intItemActivity.p */
{lib/trig-def.i &OBJECT=ITEM/v-itmhdr}
DEF VAR strCommand AS CHARACTER.
strCommand = ""D:\CSRP\symixplus\DRI\IEV\LANClient\Integral Enterprise Viewer.exe"" +
    " /vn:Item Activity /r:-2 /on:Item," + trim(ITEM.ITEM).
DEF VAR ch AS COM-HANDLE.
CREATE "wscript.shell" ch.
ch:RUN(strCommand, 1).

```

Notes:

- 1) {lib/trig-def.i &OBJECT=ITEM/v-itmhdr} – This line corresponds to the name of the SyteLine Viewer that contains the field to which we are attaching our right-click program.
- 2) strCommand = ""D:\CSRP\symixplus\DRI\IEV\LANClient\Integral Enterprise Viewer.exe"" - This line lists the name of the Integral LAN Client, as installed on your computer. You must update it based upon where your copy is installed.
- 3) “/vn:Item Activity” - This part lists the name of the Integral View you are invoking. We created this earlier, in step a).
- 4) “/r:-2” – This tells Integral to run the report and to display the Integral screen for the user to work with.
- 5) “/on:Item,” + trim(ITEM.ITEM)” – This part takes the value of a field in the current SyteLine Windows (the Item number, stored in the field “ITEM.ITEM,” and uses it to fill in a Schema Object in the Integral Inquiry (“item”). You can use any fields on the SyteLine screens (or that you lookup via Progress code, from other fields on the SyteLine screen) to pass values to any Schema Objects available in the Integral View.
- 6) “ch:RUN(strCommand, 1).” - This tells the Shell we just created to run the Command String we built. The “1” in the second parameter tells Windows to display the process in a normal Window.

```

Procedure Editor - D:\CSRP\Mscmods\uet\item\intBarcode.p
File Edit Search Buffer Compile Tools Options Help

/* intBarcode.p */
{lib/trig-def.i &OBJECT=ITEM/v-itmhdr}
DEF VAR strCommand AS CHARACTER.
strCommand = ""D:\CSRP\symixplus\DRI\IEV\LANClient\Integral Enterprise Viewer.exe"" +
    " /vn:Bar Coded Labels /f:d:\apps\jetform\central\server\data\x.dat" +
    " /r:3 /exit /on:Item," + trim(ITEM.ITEM).
DEF VAR ch AS COM-HANDLE.
CREATE "wscript.shell" ch.
ch:RUN(strCommand, 1, TRUE).
MESSAGE "Your Bar Codes have been printed. Hit OK to return to SyteLine."

```

```

/* intBarcode.p */
{lib/trig-def.i &OBJECT=ITEM/v-itmhdr}

```

```

DEF VAR strCommand AS CHARACTER.
strCommand = ""D:\CSRP\symixplus\DRI\IEV\LANClient\Integral Enterprise Viewer.exe"" +
  "/vn:Bar Coded Labels /f:d:\apps\jetform\central\server\data\x.dat" +
  "/r:3 /exit /on:Item," + trim(ITEM.ITEM).
DEF VAR ch AS COM-HANDLE.
CREATE "wscript.shell" ch.
ch:RUN(strCommand, 1, TRUE).
MESSAGE "Your Bar Codes have been printed. Hit OK to return to SyteLine."

```

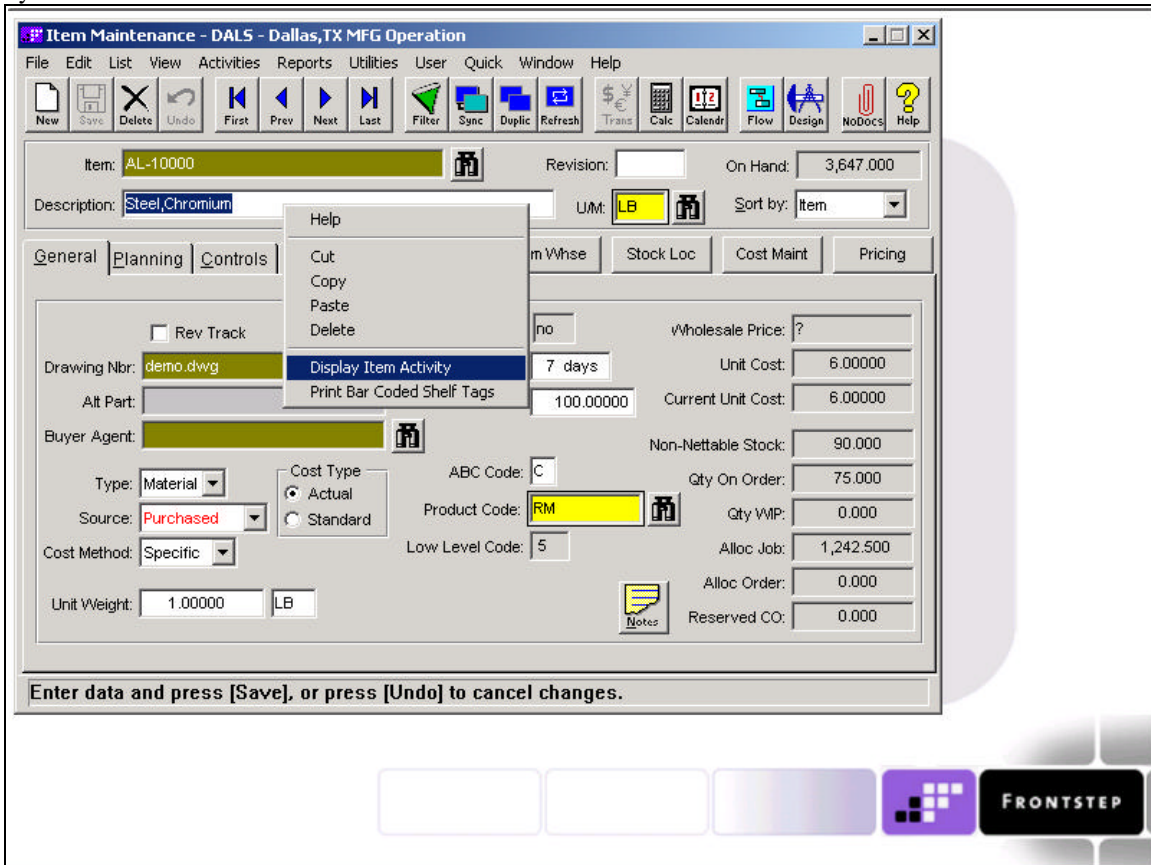
Notes:

- 1-3,6) Same as above.
- 4) “/r:3 /exit” – This tells Integral to run the report and output the results to JetForms, exiting after the report has been completed.
- 5) “/f:d:\apps\jetform\central\server\data\x.dat” – This tells Integral where to put its Field-Nominated output file, which represents a JetForms collector directory on our system.
- 7) “ch:RUN(strCommand, 1, TRUE).

MESSAGE "Your Bar Codes have been printed. Hit OK to return to SyteLine." -

This tells the Shell we just created to run the Command String we built. The “TRUE” in the third parameter (which is optional) tells the calling application (SyteLine) to wait until Integral is finished, before allowing users to interact further with it. This leads to the report being created and the message being presented to the user, to hit OK, before they can continue with SyteLine.

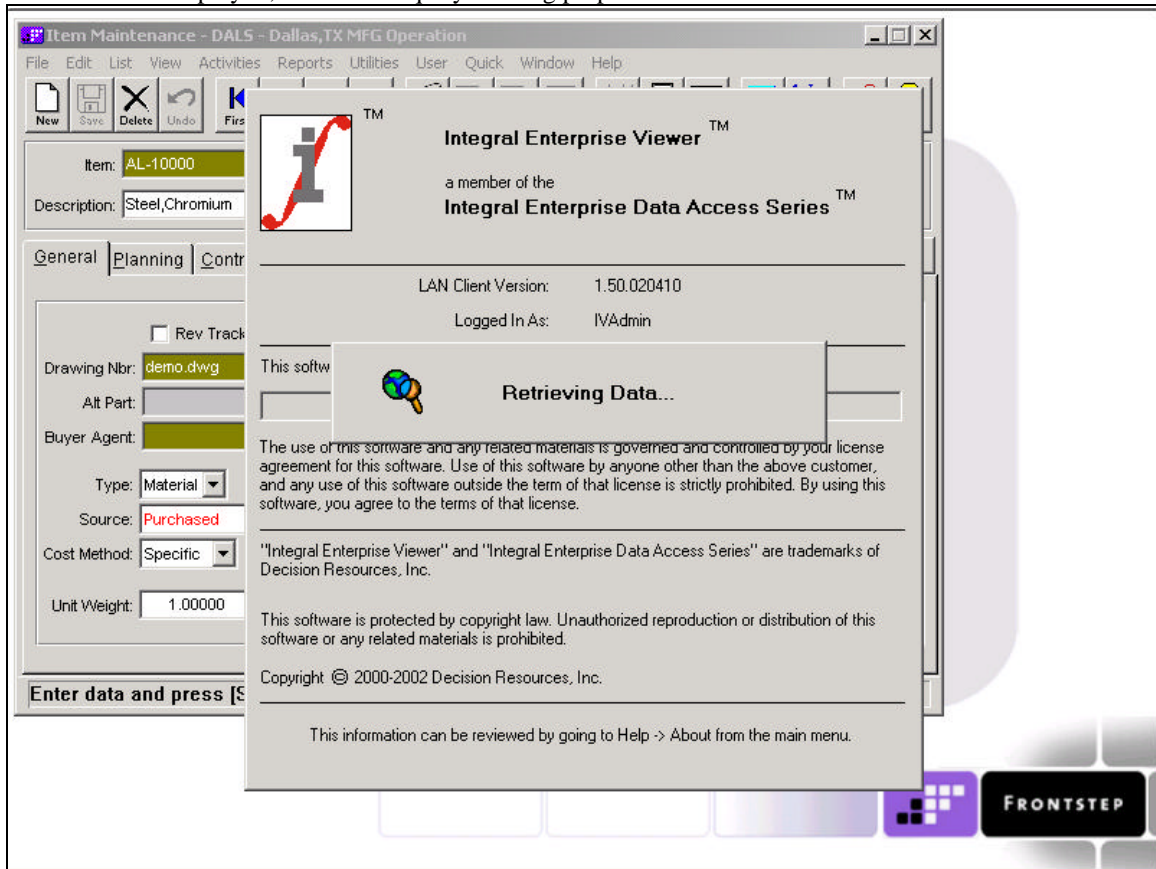
e. Now look what happens when I right-click the Description field on the Item Maintenance Window in SyteLine:



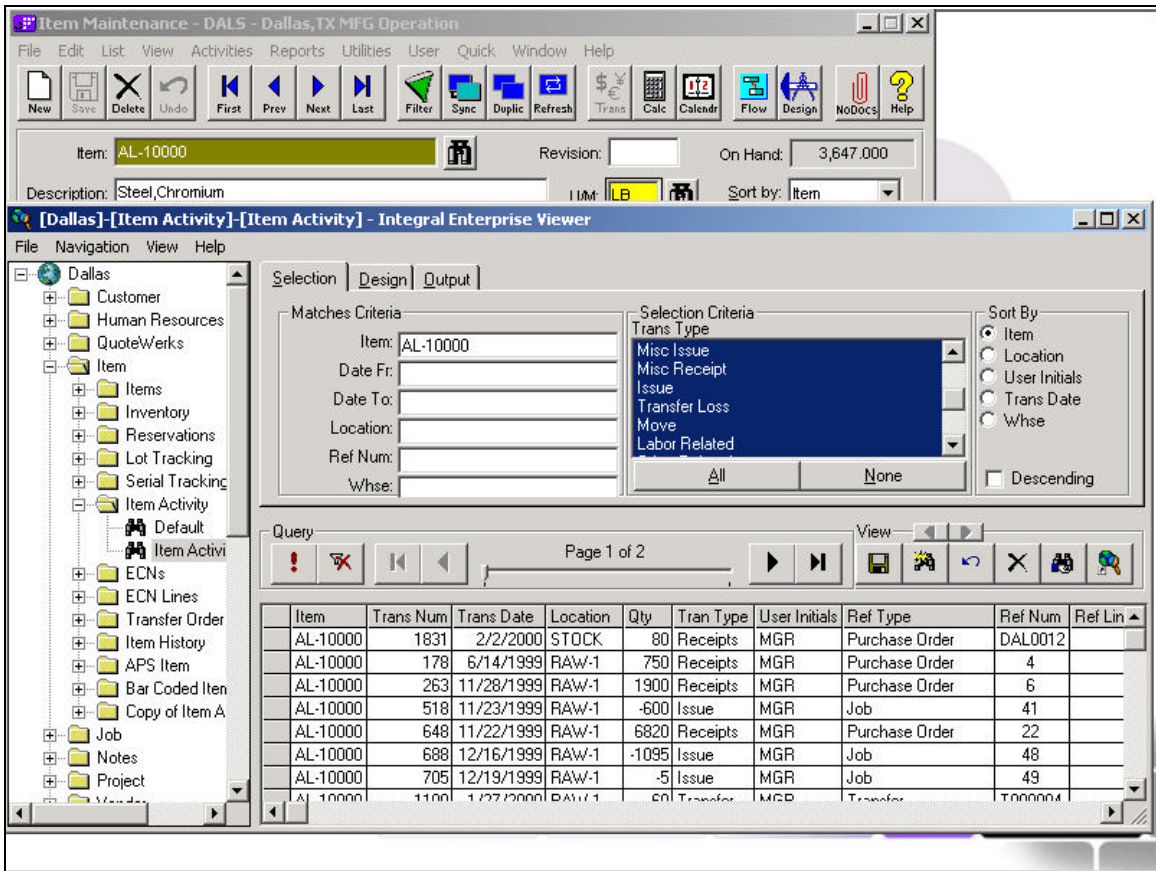
This provides us the option to run either of our Integral inquiries from SyteLine. Here are examples of both.

Item Activity:

Here's what is displayed, while the Inquiry is being prepared:



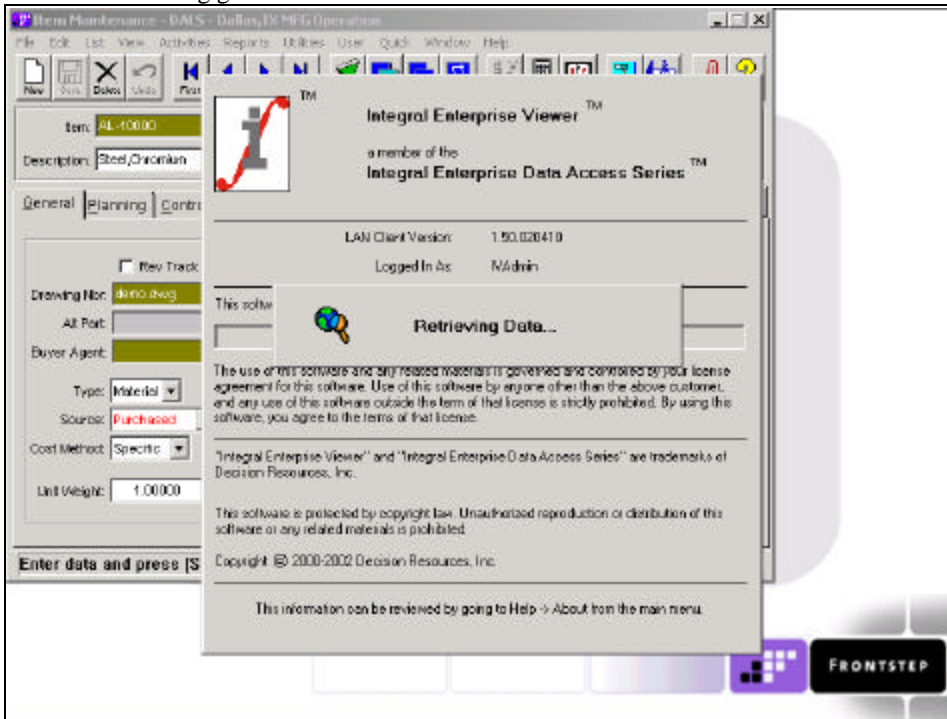
Then, the inquiry appears:



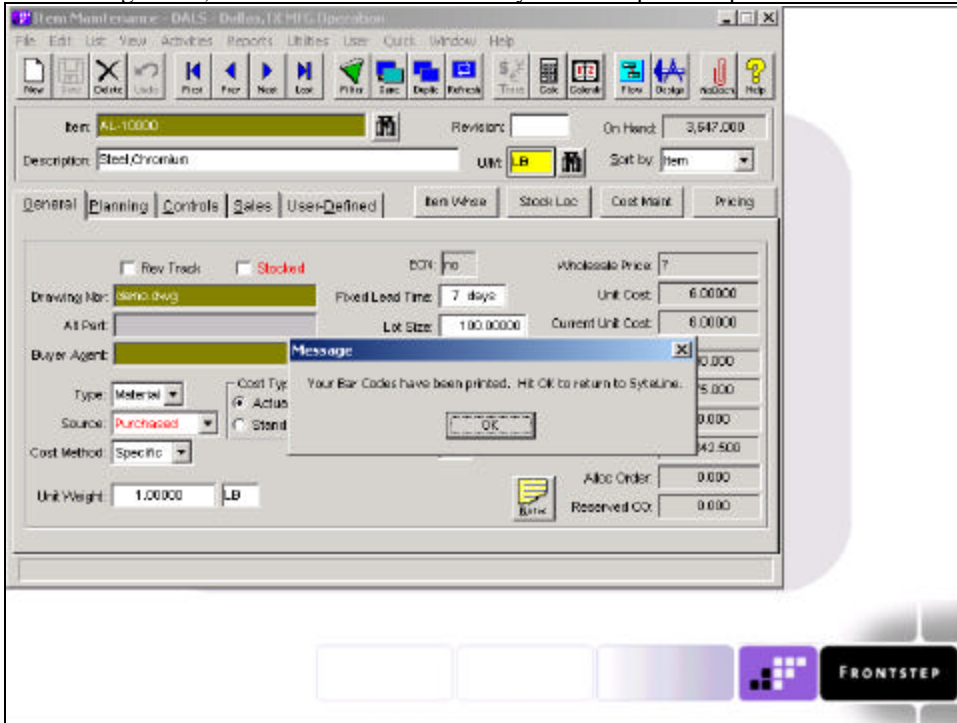
The user may continue working with both SyteLine and Integral.

Bar-coded labels:

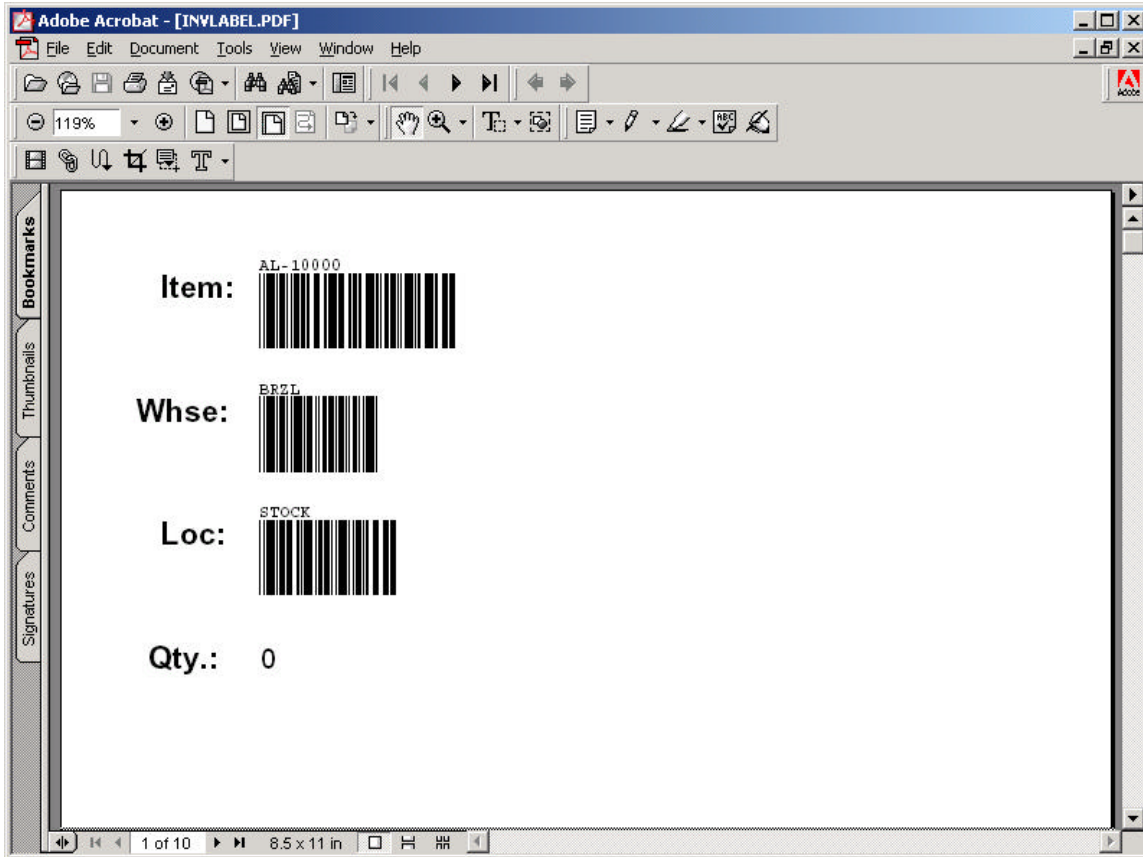
While data is being gathered:



While this goes on, the user cannot work with SyteLine. Upon completion we see:



After OK, the user continues with SyteLine, and the PDF-label file has been created:



These techniques are very versatile. Utilizing other Integral and other Solutions provided here, you can attach Integral programs to any SyteLine window, and can select information within the Integral Inquiry based upon any fields on the SyteLine screen. This might be one of the built-in inquiries that comes with Integral, or it may be one that you built, that might include data in SyteLine with other data from other databases at your site that contain related information. Once your inquiry is invoked, it can be display to the user, printed out, or exported in formats such as Excel spreadsheets or comma-separated values files. The result – information at your users’ fingertips.