

Crystal Reports Modification

Technical Specification and Development Request Estimate

Client: Your Corporation
Application: Dynamics AX 4.0

5/7/2009
eChain Technology
Lewis Kilby
lkilby@echaintechnology.com

Document History:

ID	Date	Description	Author(s)
05070801	5-7-09	Crystal Reports Multi-Print Mod	Lewis Kilby

Customization Estimate:

Detailed Technical Specifications: (Actual)	1.0 Hrs
Programming: (Est)	6.0 Hrs
Testing & Documentation: (Est)	0.5 Hrs
Total:	7.5 Hrs

* Client will be invoiced for actual hours worked.

Acceptance and authorization – DEVELOPMENT REQUEST

This Technical Specification and Development Request Estimate is expected to be accurate to +/- 2 hours. However, unexpected complications can arise that may significantly increase the amount of time required to complete the work outlined in this document. eChain will obtain authorization in writing before exceeding this estimate by more than 2 hours. Signoff by the parties below constitutes authorization and approval to begin work on this development request.

Your Corporation

eChain Technology

Signature

Signature

Title

Title

Full Name

Full Name

Date

Date

Please Fax Signed Documents to:

eChain Fax: 800-665-3176

1. Overview

Phillips Corporation created reports in Crystal Reports to replace the generic Ax reports. They made changes to the appropriate FormLetter classes within Ax to generate the Crystal Report. The remaining problem with this modification occurs when multiple records are selected to print at the same time. When this scenario occurs, a Crystal Reports window is opened for each record. They would like only one Crystal Reports window open containing a multi-page document.

2. Enhancement Summary

Make changes which will support one call to Crystal Reports with each record "id" being passed into Crystal reports as a comma delimited string.

3. Assumptions

- Phillips Corporation uses Proforma reports
- Technical design changes below will be made to support the following reports:
 - Production Pick List
 - Sales Confirmation
 - Sales Invoice
 - Purchase Order

4. Observations and Considerations

- Proforma reports will be a problem based on the existing code design.
 - Ax currently creates the appropriate records in the required tables for the proforma reports but immediately deletes the records after printing the report.
 - In order to make Crystal Reports generate all reports in one window, we will need to capture each record "id" to be passed into Crystal Reports. In addition to that we will need to capture each record "id" of each proforma report. We will then move the existing proforma "delete" code to run through each record "id" and delete the proforma records after printing the Crystal Reports.
- If you don't use pro-forma documents then you have the option to leave that part out of the change and the programming time could be reduced. I'm not necessarily a believer in leaving something broken because of future implementations or procedural changes but I didn't know if we wanted to throw that out there or not.

5. Technical Design

Classes

Modify existing class – *pciCrystalReports*

- Add new string variable in class declaration named *docIdStr*

- Add new method named *AddDocument(str_docId)* . This new method will build the string containing each of the document id's that need to be printed.
- Modify the existing methods that are currently building the paramStr1 variable to use the new variable docIdStr.

Modify existing classes – *SalesFormLetter_Confirm, SalesFormLetter_Invoice, PurchFormLetter_PurchOrder*

- Instantiate the pciCrystalReports class at the beginning of the query.
- In the existing printJournal method, call the new new pciCrystalReports.addDocument method passing in the appropriate document id.
- Add a new container that will hold each document id if the reports are being generated as proforma reports and add the document id to the container in the printJournal method.
- Comment the code to delete the proforma records in the printJournal method.
- Add the code to call pciCrystalReports.run method after the query has completed the generation of all the records.
- Add the code to delete the proforma records by scanning through the container after the crystal reports have been generated.
- Modify removeProForma method to work with the above changes.

Forms

ProdTable

- Modify the click event of the button pci_print_picklist.
- Instantiate the pciCrystalReports class.
- Query for each production order that was selected and pass each production order id into the new pciCrystalReports.addDocument method.
- Call the pciCrystalReports.run method.
- Comment the super() call.