

RDP, Contract, UI Builder and Controller classes for SSRS report Development



RSS



Faisal Fareed on 27 Dec 2014 7:24 PM



Without going into more details over individual classes (RDP, Contract, UI Builder and Controller) I am writing this post to develop an SSRS report using all these classes (some of them are optional). Let's find out how?

RDP Class

Details of this class are here;

<http://www.dynamics101.com/2013/09/developing-ssrs-report-using-report-data-provider-microsoft-dynamics-ax-2012/>

Create a new class in AOT, I have named it FF_ReportDP and **extends** it from **SRSReportDataProviderBase**

```

1  [
2      SRSReportQueryAttribute(queryStr(CustBalanceList)),
3      SRSReportParameterAttribute(classStr(FF_ReportContract))
4  ]
5  class FF_ReportDP extends SRSReportDataProviderBase
6  {
7      CustTable          custTable;
8      CustTrans          custTrans;
9      FF_ReportContract  ff_ReportContract;
10     RecordInsertList    recordInsertListTmpTable;
11     FF_ReportTmpTable   ff_ReportTmpTable;
12 }

```

SRSReportQueryAttribute: specifies which AOT query will be used in this report. If the RDP class uses an AOT query to process data, define this attribute at the beginning of the class.

SRSReportParameterAttribute: defines the data contract class that will be used by this report to prompt for parameter values. If the RDP class contains any parameters this define this attribute at the beginning of the class.

NOTE: Both the attributes are optional. If the report does not use any query or does not want any parameter to filter report data, these attributes do not need to be used.

Add a new method and name it `getFF_ReportReportTmp`.

This method is mandatory because reporting services uses this method to get the table buffer containing the processed data. The **SRSReportDataSetAttribute** is used to indicate the temporary table name and also tells the reporting services to use this method to retrieve the processed data.

```
1 [SRSReportDataSetAttribute(tablestr('FF_ReportTmpTable'))]
2 public FF_ReportTmpTable getFF_ReportReportTmp()
3 {
4     select ff_ReportTmpTable;
5     return ff_ReportTmpTable;
6 }
```

Add a
new

method and name it `processReport`. This method contains the business logic and is called by reporting services to generate data.

```
1 [SysEntryPointAttribute]
2 public void processReport()
3 {
4     Query          query;
5     QueryRun       queryRun;
6
7     ;
8
9     query = new query();
10    ff_ReportContract = this.parmDataContract() as ff_ReportContract;
11
12
13    recordInsertListTmpTable = new RecordInsertList(tableNum(FF_ReportTmpTable), // table id
14                                                    false, // skip insert
15                                                    false, // skip database log
16                                                    false, // skip events
17                                                    false, // skip aos validation
18                                                    false, // skip RLS validation
19                                                    ff_ReportTmpTable); // buffer where records will be inserted
20
21    while (queryRun.next())
22    {
23        custTable = queryRun.get(tableNum(custTable));
24        custTrans = queryRun.get(tableNum(custTrans));
25
26        select custTable where custTable.AccountNum == ff_ReportContract.parmCustomerAccount();
27
28        ff_ReportTmpTable.AccountNum = custTable.AccountNum;
29        ff_ReportTmpTable.BankAccount = custTable.BankAccount;
30        ff_ReportTmpTable.AmountMST = custTrans.AmountMST;
31        ff_ReportTmpTable.Invoice = custTrans.Invoice;
32
33        recordInsertListTmpTable.add(ff_ReportTmpTable);
34    }
35    recordInsertListTmpTable.insertDatabase();
36 }
37 }
```

Initialise and retrieve contract class parameters in `processReport` method to get filtered records from tables based on parameter inputs. e.g. Customer account is added as a range to retrieve data only for selected customer.

Tip: For better performance `RecordInsertList` is used in `processReport` method to write data into database in one call.

Contract class

Create a new class and name it `FF_ReportContract`

In this class I have added two parameters `Company` and `CustomerAccount` which can be used to filter data on report either passing values from a menuitem or from a class or while calling report from a form. I will address this later in this or next post.

```

1  [
2      DataContractAttribute,
3      SysOperationContractProcessingAttribute(classstr(FF_ReportUIBuilder))
4  ]
5  class FF_ReportContract
6  {
7      str          company;
8      CustAccount  customerAccount;
9  }

```

It will tell the contract class to build the parameter dialog. In other words, it will link the UI Builder Class with the contract class.

Parameter methods:

```

1  [
2      DataMemberAttribute('Customer Account'),
3      SysOperationLabelAttribute(literalStr("@SYS7149"))
4  ]
5  public CustAccount parmCustomerAccount(str _customerAccount = customerAccount)
6  {
7      customerAccount = _customerAccount;
8      return customerAccount;
9  }|

```

```

1  [
2      DataMemberAttribute('company'),
3      SysOperationLabelAttribute(literalStr("@SYS177192"))
4  ]
5  public FreeText parmCompany(str _company = company)
6  {
7      company = _company;
8      return company;
9  }|

```

UI Builder Class

Details of this class are here;

<http://www.dynamics101.com/2014/04/using-ui-builder-class-to-develop-ssrs-reports-microsoft-dynamics-ax-2012/>

Create a new class and extends it from [SysOperationAutomaticUIBuilder](#)

```

1  class FF_ReportUIBuilder extends SysOperationAutomaticUIBuilder
2  {
3      FF_ReportContract  contract;
4
5      DialogField        dialogCompany;
6      DialogField        dialogCustomerAccount;
7  }

```

Override build method to create own dialog box

```

1 public void build()
2 {
3     ;
4
5     if (this.controller() as FF_ReportController)
6         return;
7
8     contract = this.dataContractObject();
9
10    dialogCompany      = this.addDialogField(methodStr(FF_ReportContract, parmCompany), contract);
11    dialogCustomerAccount = this.addDialogField(methodStr(FF_ReportContract, parmCustomerAccount), contract);
12 }

```

Override postBuild method

```

1 public void postBuild()
2 {
3     super();
4
5     if(this.controller() as FF_ReportController)
6         return;
7
8     dialogCompany.registerOverrideMethod(methodStr(FormStringControl, lookup), methodStr(FF_ReportUIBuilder, companyLookup), this);
9 }

```

New method to create custom lookup of all companies in AX

```

1 private void companyLookup(FormStringControl _lookup)
2 {
3     Query          query = new Query();
4     QueryBuildDataSource qbds;
5     SysTableLookup sysTableLookup;
6
7     if (_lookup != null)
8     {
9         sysTableLookup = SysTableLookup::newParameters(tableNum(CompanyView), _lookup);
10
11         qbds = query.addDataSource(tableNum(CompanyView));
12         sysTableLookup.addLookupfield(fieldnum(CompanyView, id), true);
13         sysTableLookup.addLookupfield(fieldnum(CompanyView, Name), false);
14         sysTableLookup.parmUseLookupValue(false);
15
16         sysTableLookup.parmQuery(query);
17
18         // Perform the lookup.
19         sysTableLookup.performFormLookup();
20     }
21 }

```

Controller class

Create a new class and extends it from SrsReportRunController

```

1 class FF_ReportController extends SrsReportRunController
2 {
3 }

```

Create a new Main method to call this class from a menu item which will internally call SSRS report

```

1 public static client void main(Args args)
2 {
3     FF_ReportController ff_ReportController;
4     ;
5
6     ff_ReportController = new FF_ReportController();
7
8     ff_ReportController.parmArgs(args);
9
10    ff_ReportController.parmReportName(ssrsReportStr(FF_Report, PrecisionDesign));
11    ff_ReportController.parmDialogCaption('FF Report');
12 }

```

```

13     ff_ReportController.startOperation();
14 }

```

Add a new output menu item in AOT with name FF_ReportController with property Object type = Class and Object = FF_ReportController.

Run menu item and this will be the output dialog screen

Accounts receivable group parameters are three ranges added in CustBalanceList query which is used in this report example.

To hide these group parameters override showQueryValues method and return false.

```

1  public boolean showQueryValues(str parameterName)
2  {
3      return false;
4  }

```

A
 t
 r
 h
 d
 r
 c
 p
 a
 e
 n
 e
 s
 <>

[Read Complete Post and Comments >](#)

Community Forums

[Ask a Question >](#)

Personalize your experience!

Personalized Community is here!

Quickly customize your community to find the content you seek.

[Personalize Community Now >](#)



Top Tags

[View All](#)

[AX 2012 R3](#)

[AX General](#)

[X++](#)

[AX 2012](#)

[D365FinOps](#)

[Dynamics 365 for Operations](#)

[SSRS](#)

[dual write](#)

[AX 7](#)

FEATURED CONTENT



Latest TechTalk Videos

[View the latest TechTalks](#)

[View the Finance TechTalks](#)

[View all TechTalks](#)

Have questions on moving to the cloud? Visit the [Dynamics 365 Migration Community](#) today! Microsoft's extensive network of Dynamics AX and Dynamics CRM experts can help.



2022 Release Wave 2

Check out the latest updates and new features of Dynamics 365 released from October 2022 through March 2023

[Release Overview Guides and Videos](#)

[Release Plan](#)

[Preview 2022 Release Wave 2](#)

FASTTRACK



The FastTrack program is designed to help you accelerate your Dynamics 365 deployment with confidence.

[FastTrack Community](#) | [FastTrack Program](#) | [Finance and Operations TechTalks](#) | [Customer Engagement TechTalks](#) | [Upcoming TechTalks](#) | [All TechTalks](#)

Business Applications communities

Select Community ▾

What's new

Surface Laptop Go 2

Surface Pro 8

Surface Laptop Studio

Surface Pro X

Surface Go 3

Surface Duo 2

Surface Pro 7+

Windows 11 apps

Microsoft Store

Account profile

Download Center

Microsoft Store support

Returns

Order tracking

Virtual workshops and training

Microsoft Store Promise

Flexible Payments

Education

Microsoft in education

Devices for education

Microsoft Teams for Education

Microsoft 365 Education

Education consultation appointment

Educator training and development

Deals for students and parents

Azure for students

Business

Microsoft Cloud

Developer & IT

Azure

Company

Careers

[Microsoft Security](#)

[Developer Center](#)

[About Microsoft](#)

[Dynamics 365](#)

[Documentation](#)

[Company news](#)

[Microsoft 365](#)

[Microsoft Learn](#)

[Privacy at Microsoft](#)

[Microsoft Power Platform](#)

[Microsoft Tech Community](#)

[Investors](#)

[Microsoft Teams](#)

[Azure Marketplace](#)

[Diversity and inclusion](#)

[Microsoft Industry](#)

[AppSource](#)

[Accessibility](#)

[Small Business](#)

[Visual Studio](#)

[Sustainability](#)



[English \(United States\)](#)

[Contact Us](#) [Privacy](#) [Terms of Use](#) [Trademarks](#) [© Microsoft 2022](#)