COSMO CONSULT
Business-Software für Menschen

MICROSOFT AL REPORTS

COSMO CONSULT GmbH
Neuseiersberger Straße 157
8055 Graz

Fon +43(0) 316 |269980
Fax +43(0) 316 | 269980 - 12889
www.cosmoconsult.com
at.info@cosmoconsult.com

Steuernummer FN 164426d
UST-IdNr. ATU 43908107
Gesellschaftssitz Graz

Geschäftsführer
Reinhard Waltl
Erich Battisti

BAWAG-PSK
IBAN AT13 6000 0000 0968 4300
BIC|SWIFT BAWAATWW

COSMO CONSULT
Business-Software für Menschen

Inhalt

1 Introducing Reporting in AL

1.1 Indroduction

The Reports in AL have four main components: the reportobject, dataset, layoutfile and the requestpage.

The behavior of the components is the same as in C/AL, only the developmentprocess has changed.

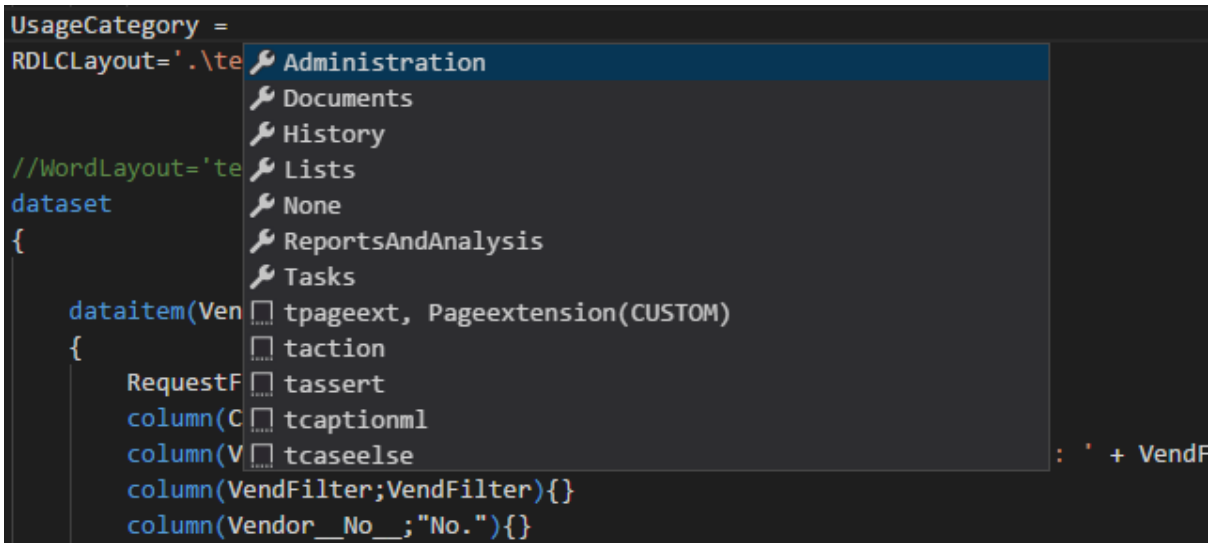
1.2 Reportobject

The reportobject is defined in a .al file. You can create it with the Keyword “report”, then you have to set an objected and an objectname. To create a reportobject you could also use the template “treport”, so that the basic structure is created automatically.

```
report 50100 "Vendor - List V2"
{
    CaptionML=ENU='Vendor - List',
              DEU='Kreditor - Liste';
    UsageCategory = Lists;
```

You can also set some properties for the whole report.

Very Important is the new Property UsageCategory, without this property the report could not be found in the menusuite of NAV 2018. The available options are “Administration”, “Documents”, “History”, “Lists”, “None”, “ReportsAndAnalysis”, “Tasks”.



1.2.1 Labels

If you want to display text in a report layout, it is recommended to use Labels. To define labels you could use the label keyword.

```
labels
{
    label(ReportCaption;ENU='ReportCaption - Test')
}
```

1.2.2 Trigger on reportobject

On the report object you can define up to three triggers which you can use to programm the behavior of the report. The triggers are: OnInitReport, OnPreReport, OnPostReport. None of them is mandatory, you could also define a report without any triggers.

```
trigger OnInitReport();
begin
end;
trigger OnPreReport();
begin
end;
trigger OnPostReport();
begin
end;
```

1.3 Dataset

The dataset is the definition of which data is used for the report. It can contain one ore more dataitems.

It is defined using the Keyword **dataset**, it doesn't have any properties.

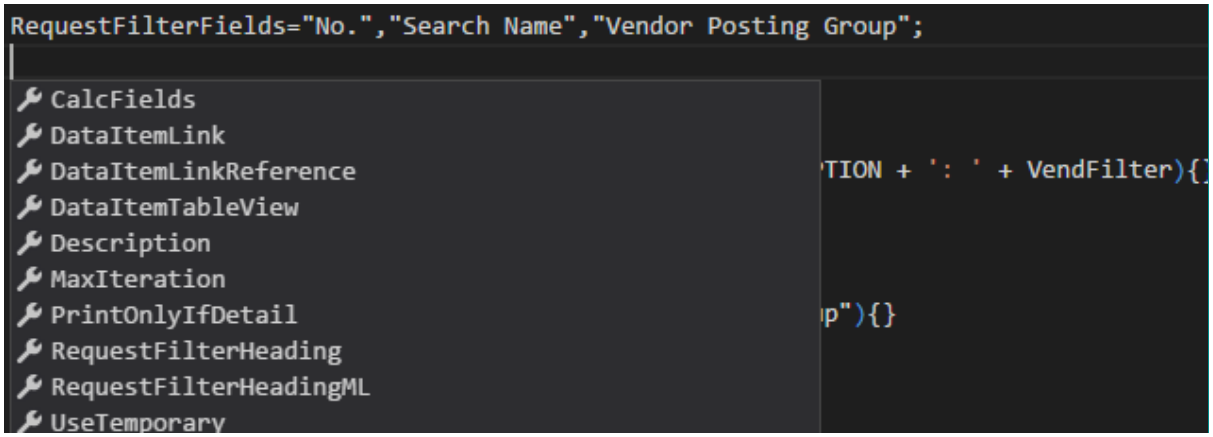
```
dataset
{

    0 references
    dataitem(Vendor; Vendor)
    {
```

1.3.1 Dataitem

With the data item you could add a new source table to the reports dataset. Each dataitem has a name and a source table

```
dataitem(Name,SourceTable)
```

The dataitem has also some properties, this properties are used to control the data which is loaded from database.



Dataitems could also be nested, each of them could contain other dataitems.

```
dataitem(Item;"Item")
{
    dataitem(ItemLedgeEntry;"Item Ledger Entry")
    {
```

Another possibility to organize them is to place them in sequence.

For Example:

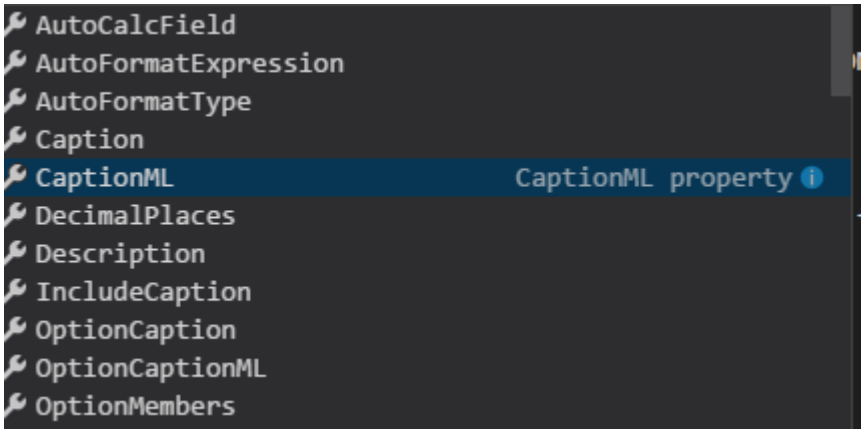
```
dataitem(Cust;Customer)
{

}
dataitem(Vend;Vendor)
{

}
```

1.3.2 Columns

If you want to display fields on the layout of the report, you have to define columns. You can only use this columns in the Layout.

Each Column has a Name and SourceExpression, the SourceExpression could be a global Variable or a function or a field in the sourcetable. You could also define some Properties for each column.

```
column(Name,SourceExpression)
{
    ...Properties...
}
```



1.3.3 Trigger on dataitem

Foreach dataitem you could define up to three triggers: OnPreDataItem, OnAfterGetRecord, OnPostDataItem

```
trigger OnPreDataItem();
begin
end;
trigger OnAfterGetRecord();
begin
end;
trigger OnPostDataItem();
begin
end;
```

1.4 Layout

The Layout is saved in a a separated file. You have to link the Layout with the code. The support Layout extensions are ".rdl", ".rdlc" and ".docx".

There are two properties for linking the Layout, RDLCLayout (used for ".rdl" and ".rdlc") and WordLayout (used for ".docx").

```
    RDLCLayout='.\Reports\Layout\CustomerTop10V2.rdlc';
    WordLayout='.\Reports\Layout\word.docx';
```

If the file does not exist, you create it by building your solution (ctrl+shift+B)

For those reports where you have declared both (word and rdlc) you could define a DefaultLayout.

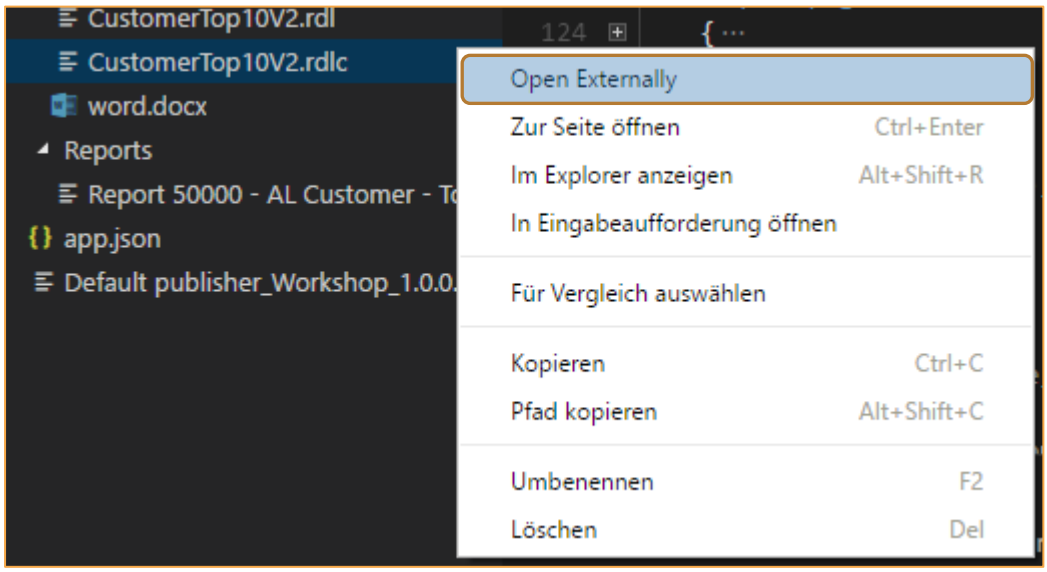
```
    DefaultLayout=RDLC;
```

or

```
    DefaultLayout=Word;
```

The customer could change the default layout in the Report Layout Selection Page.

To open the Layout you could use the „Open Externally“ menuitem.



1.5 Requestpage

Designing a Requestpage is similar to the pageobject, the only difference is the key word "requestpage"

```
requestpage
{
    …Properties…

    layout
    {
    }

    actions
    {
    }
}
```

2 Build new reports in AL

2.1 Build a small Report 50130 "Simple Document"

Use with some fields from Sales Header:

No.

- Selt-To Customer Name
- Selt-To Customer Address
- Selt-To Customer Post Code
- Selt-To Customer City

Order No. 101001

Adatum Corporation
Station Road, 21
1230 Wien

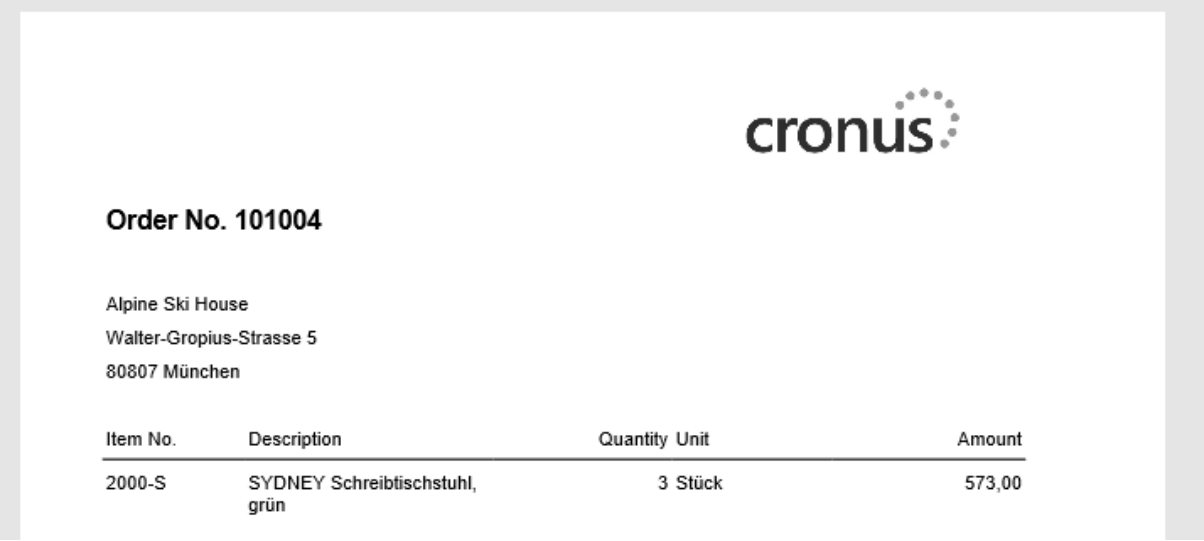
2.2 Add sales Lines to the Report “Simple Document”

Order No. 102200

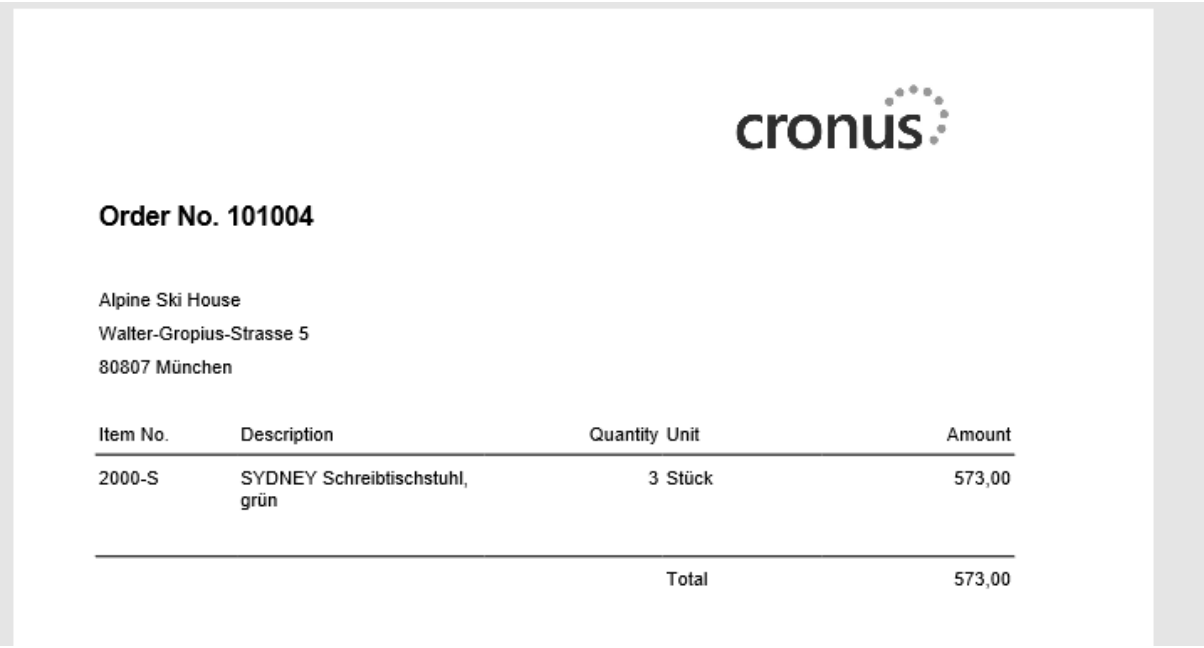
Adatum Corporation
Station Road, 21
1230 Wien

Item No.	Description	Quantity	Unit	Amount
2000-S	SYDNEY Schreibtischstuhl, grün	2	Stück	382,00
1996-S	ATLANTA Whiteboard, Basis	5	Stück	7 021,50

2.3 Add a picture

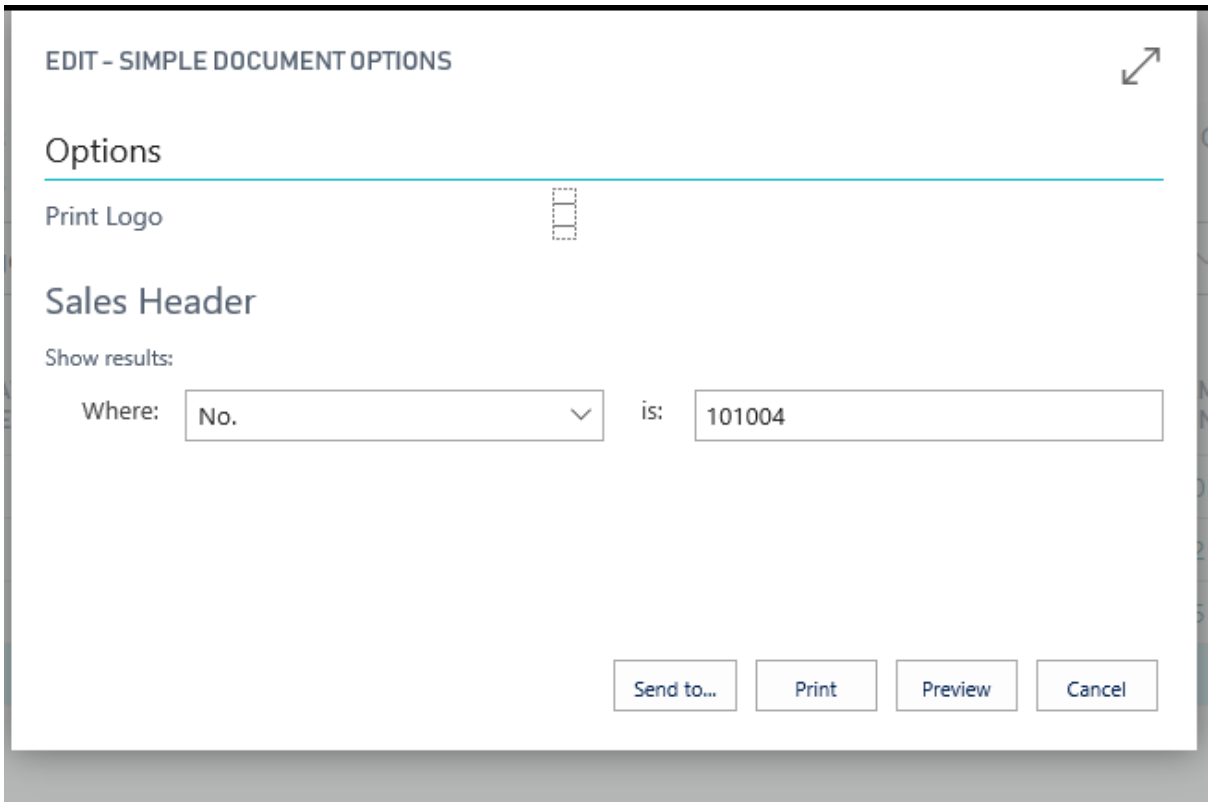


2.4 Add Totals



2.5 Add a requestpage

Make Request page with an option “Print logo”



3 Extend/change existing reports in AL

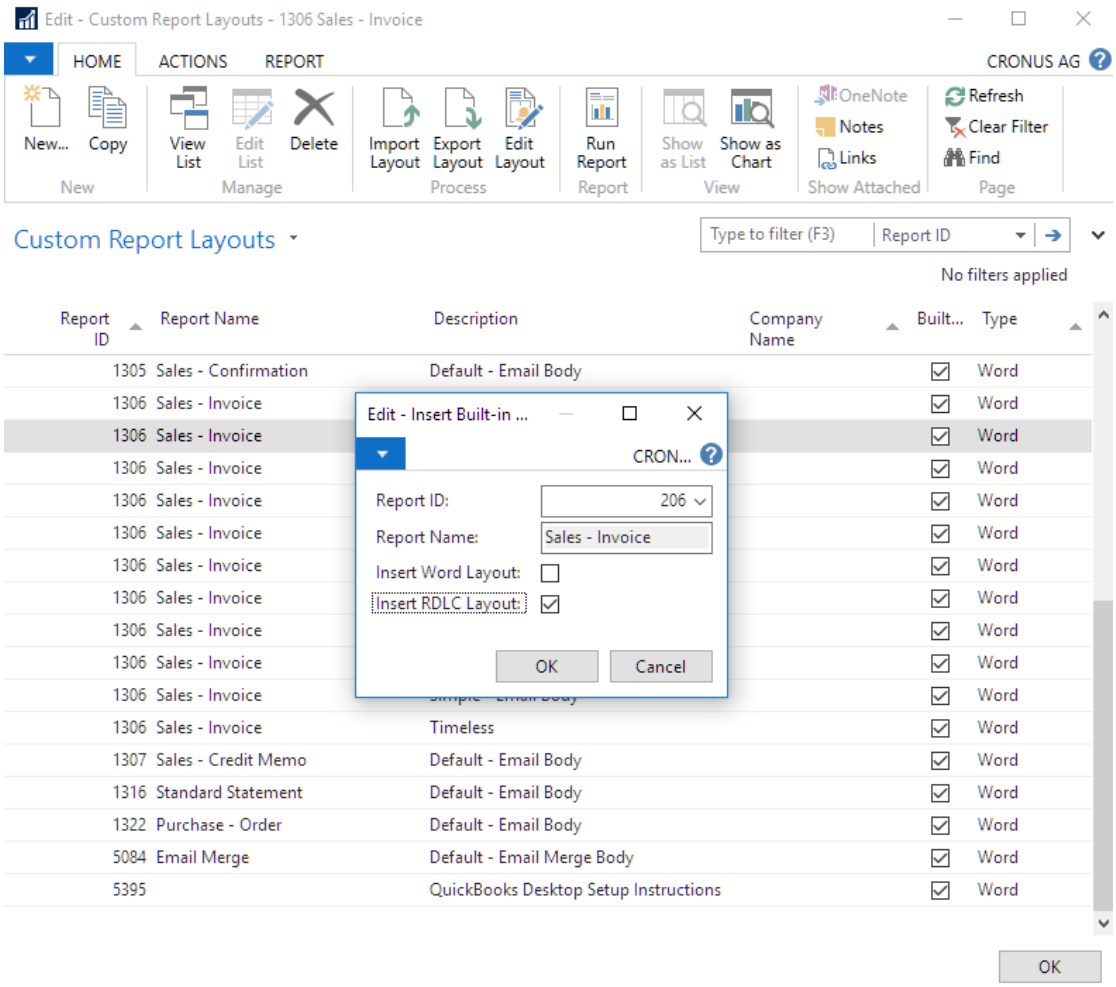
3.1 General

At the moment it is nearly impossible to extend or change existing reports, but there are some features to make some small changes. It is not possible to change the dataset of an existing report. If the report has some event publishers you could subscribe to them, but the standard code now hasn't any event in reports.

3.2 Custom Layout

With custom Layouts you can change the Layout of an Report and replace it with a new layout. You cannot change the dataset of an existing report.

To define a custom layout, you could use the „Custom Report Layouts“ Page



3.3 Replace Report using the Report Selection

For some Reports it's possible to use the Report Select, with this Feature you can Replace a whole report against another one. This effects every logic of the report. This could be used to make changes to a default report. You could copy a default report and make your changes, after this you define the new report in the report selection.

3.4 Make Reports extensible

Do make Reports extensible place an Event Publisher on the beginning and ending of each function and trigger. It's recommended to do this for product development.

```
trigger OnPreDataItem();
  begin
      OnBeforeCustomerPreDataItem();
      Window.Open(Text000);
      i := 0;
      CustAmount.DeleteAll;
      OnAfterCustomerPreDataItem();
  end;
```