

在 WinCC V7.5 报表中显示多个 WinCC 的数据

— 如何在 WinCC V7.5 报表中使用链接服务器

WinCC / V7.5 SP1/ Report/ LinkedServer

<https://support.industry.siemens.com/cs/ww/en/view/109781485>

Siemens
Industry
Online
Support



This entry is from the Siemens Industry Online Support. The general terms of use (http://www.siemens.com/terms_of_use) apply.

安全性信息

Siemens 为其产品及解决方案提供了工业信息安全功能，以支持工厂、系统、机器和网络的安全运行。

为了防止工厂、系统、机器和网络受到网络攻击，需要实施并持续维护先进且全面的工业信息安全保护机制。Siemens 的产品和解决方案仅构成此类概念的其中一个要素。

客户负责防止其工厂、系统、机器和网络受到未经授权的访问。只有在必要时并采取适当安全措施（例如，使用防火墙和网络分段）的情况下，才能将系统、机器和组件连接到企业网络或 Internet。

此外，应考虑遵循 Siemens 有关相应信息安全措施的指南。更多有关工业信息安全的信息，请访问 <http://www.siemens.com/industrialsecurity>。

Siemens 不断对产品和解决方案进行开发和完善以提高安全性。Siemens 强烈建议您及时更新产品并始终使用最新产品版本。如果使用的产品版本不再受支持，或者未能应用最新的更新程序，客户遭受网络攻击的风险会增加。

要及时了解有关产品更新的信息，请订阅 Siemens 工业信息安全 RSS 源，网址为 <http://www.siemens.com/industrialsecurity>。

目录

- 1 <概述>..... 4
- 2 <报表的 ODBC 数据库>..... 6
- 3 <WinCC 链接服务器>..... 10
- 4 < WinCC 报表中显示多个 WinCC 的数据> 12
 - 4.1 <变量归档> 12
 - 4.2 <在 SQL 中添加链接归档> 12
 - 4.3 <创建画面> 15
 - 4.4 <报表组态> 17
 - 4.5 <运行结果> 20

1 <概述>

<在 WinCC 报表系统中可以以表格或曲线的形式显示自身系统的归档数据。如果需要在 WinCC 报表系统中显示多个 WinCC 的归档数据，该如何处理？在 WinCC 服务器客户机架构中，如图 1 所示，客户机可以加载多个 WinCC 服务器的数据（包括归档数据）。

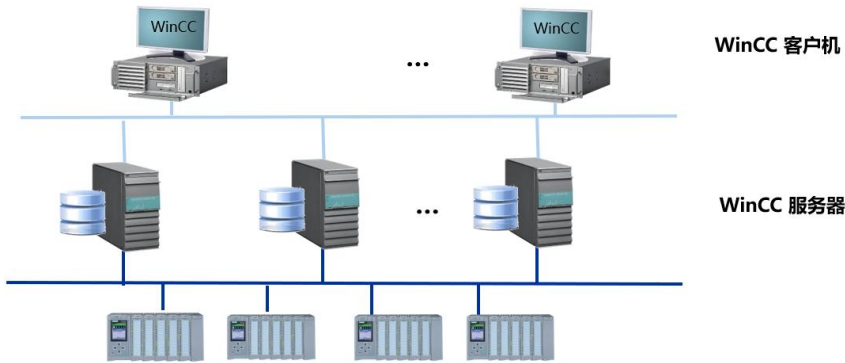


图 1

在 WinCC 客户机的报表中就可以显示多个服务器的归档数据。以在报表布局中添加在线表格为例，选择不同服务器的归档数据的步骤如图 2 所示。

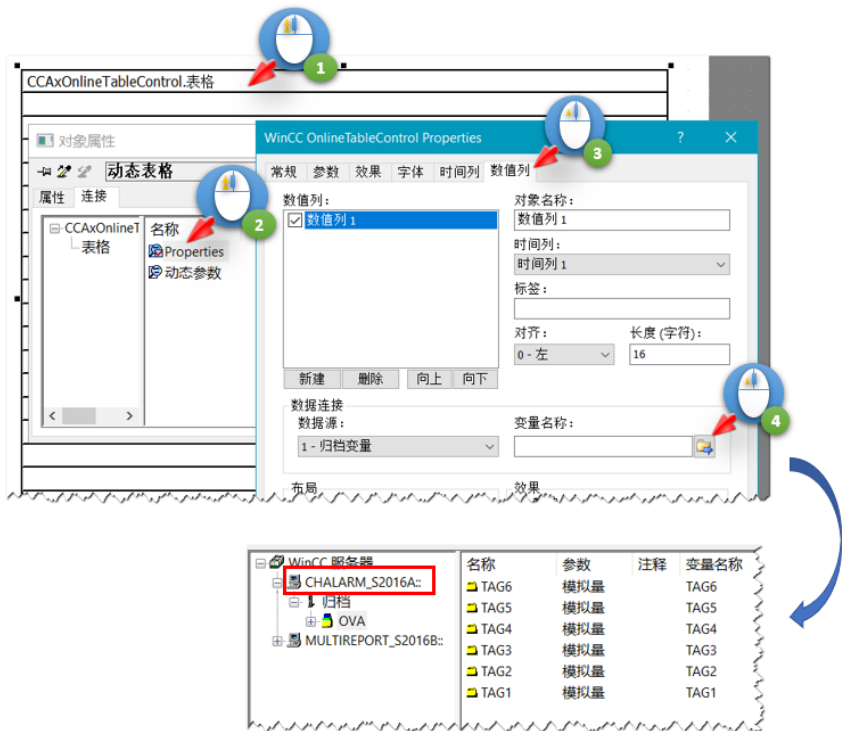


图 2

那么当单机版 WinCC 需要在报表中显示多个 WinCC 的归档数据时，需要怎么处理呢？这种情况下，就需要用到报表的 ODBC 数据库功能以及 SQL 的链接服务器（LinkedServer）功能。

本文档描述了在 WinCC 单机环境下，如何在 WinCC 报表中显示本地及远程 WinCC 的归档数据。本文档使用的操作系统及 WinCC 软件版本如下：

本地计算机：

操作系统版本：Windows Server 2016 Standard

WinCC 版本：SIMATIC WinCC V7.5 SP1

远程计算机：

操作系统版本：Windows 10 Enterprise 1809

WinCC 版本：SIMATIC WinCC V7.5 SP1

>

2 <报表的 ODBC 数据库>

<WinCC 报表布局中可以显示 ODBC 数据源。例如，数据库“laboratory”中的表“lab”中存储的是实验室检测数据，需要显示在 WinCC 报表中。如图 3 所示。

S2016A\WINCC.laboratory - dbo.lab						
	Batchname	Waterpre	Sugardegree	Ph	Acidity	Labtime
	BATCH01	...	16	11	5.1	6
	BATCH02	...	19	12	5	5.8
	BATCH03	...	17	10	4.7	6.1
▶*	NULL	NULL	NULL	NULL	NULL	NULL

图 3

首先需要创建 ODBC 数据源。在 Windows 控制面板下管理工具中，双击“ODBC 数据源 (32 位)”，打开 ODBC 数据库配置窗口，在“User DSN”下添加数据源。如图 4 所示。

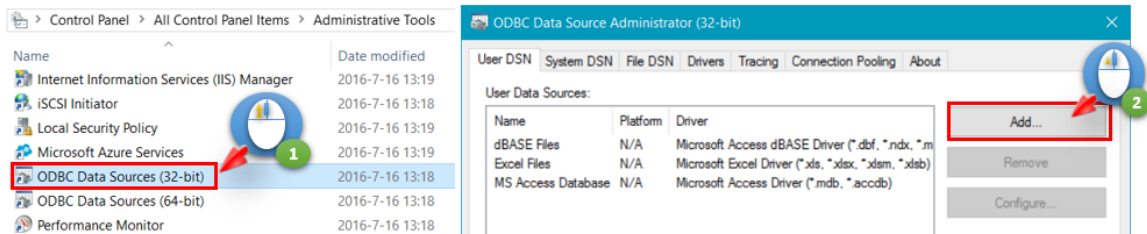


图 4

添加 ODBC 数据源的步骤参考图 5 所示。

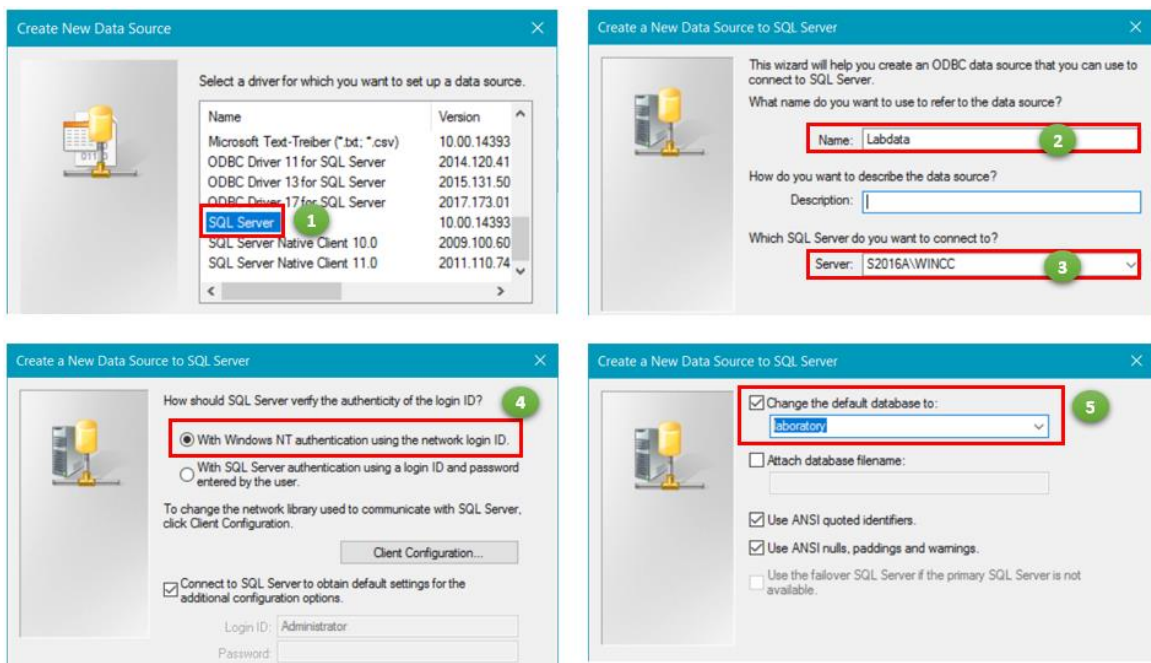


图 5

ODBC 数据源创建完成后，可以测试工作是否正常。如图 6 所示。

2 <报表的 ODBC 数据库>

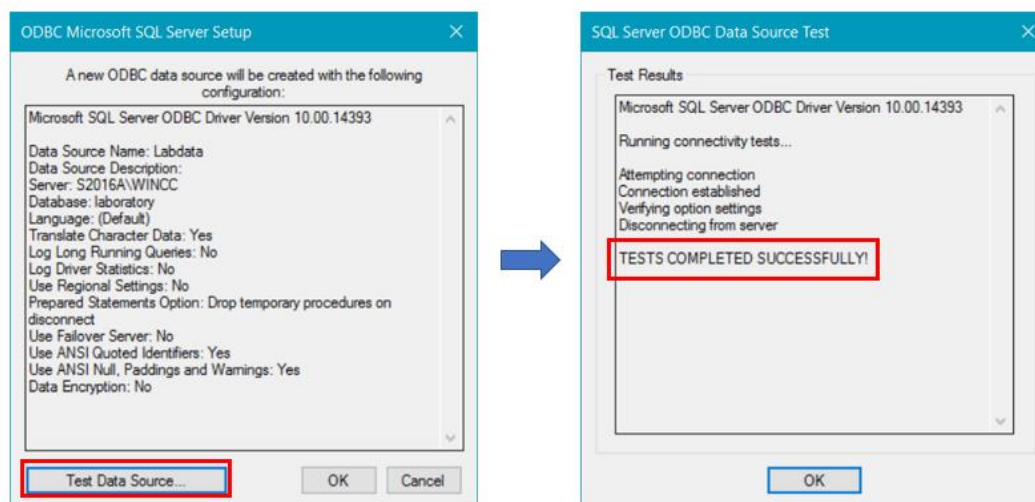


图 6

打开 WinCC 报表布局编辑器，选择“动态对象->ODBC 数据库”，把下面的“数据库表”拖拽到报表编辑器中，如图 7 所示。

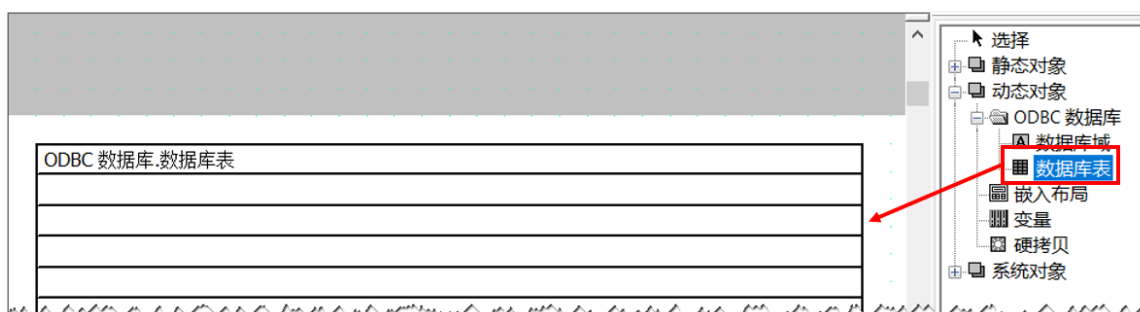


图 7

双击 WinCC 报表布局编辑器中的“数据库表”，在属性窗口中，选择“连接”，双击“数据库链接”①。然后为“ODBC 数据源”选择刚才创建的 ODBC 数据源“Labdata”②，，使能“显示列的标题”③，并输入查询语句“select * from lab”④。可以点击“测试 SQL 语句”按钮来测试数据链接工作是否正常。如图 8 所示。

2 <报表的 ODBC 数据库>

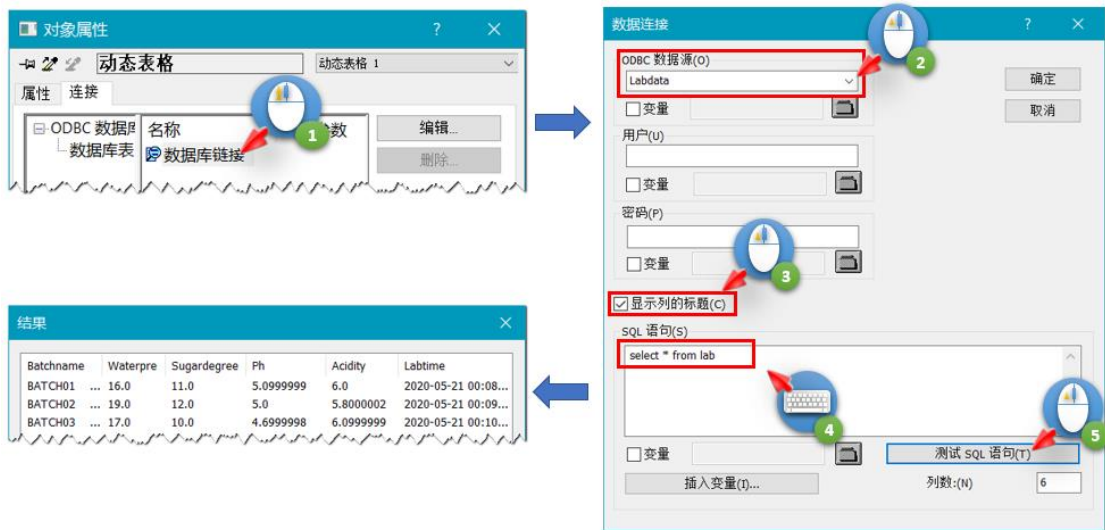


图 8

在 WinCC 报表中创建打印作业，并连接到前面创建的报表布局，如图 9 所示。

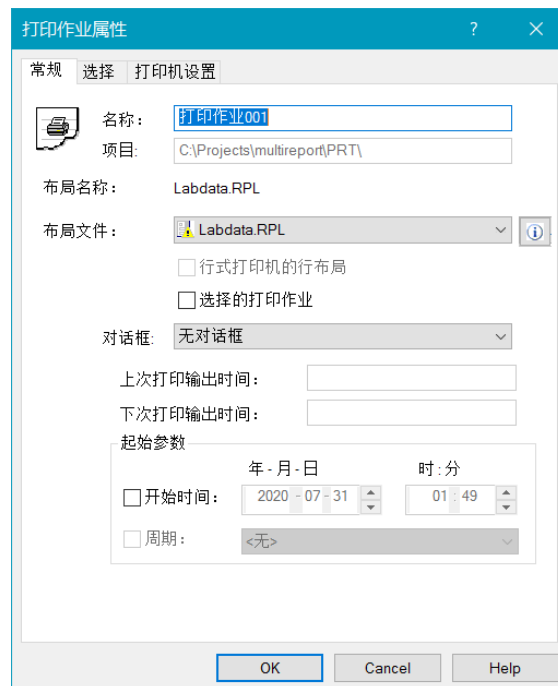


图 9

右键打印作业，在弹出菜单中选择“打印”，如图 10 所示。

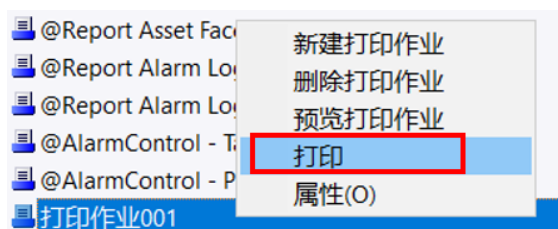
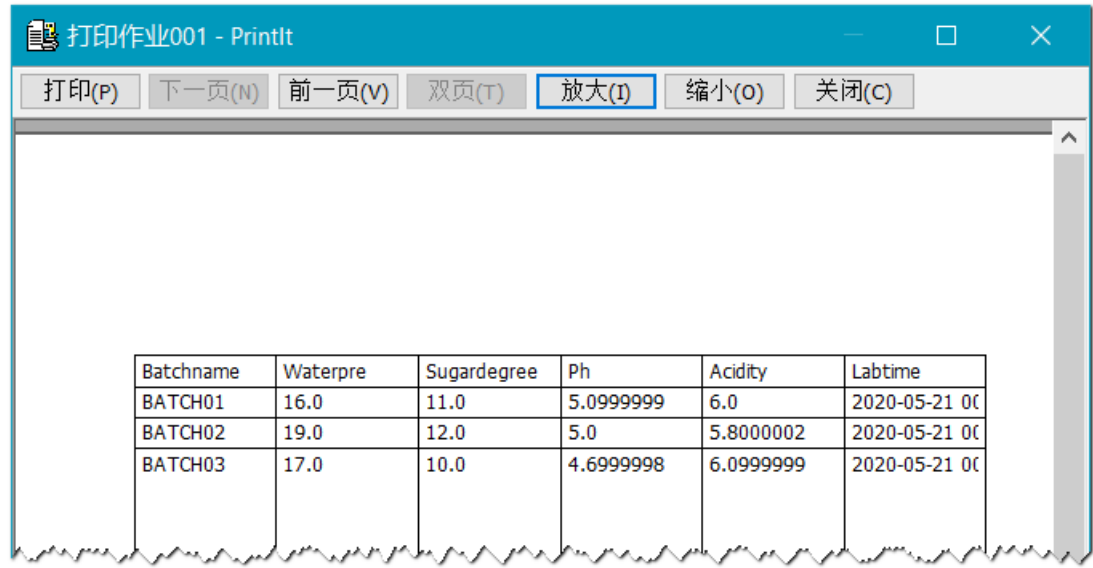


图 10

报表 ODBC 数据打印结果如图 11 所示。



Batchname	Waterpre	Sugardegree	Ph	Acidity	Labtime
BATCH01	16.0	11.0	5.0999999	6.0	2020-05-21 00:00:00
BATCH02	19.0	12.0	5.0	5.8000002	2020-05-21 00:00:00
BATCH03	17.0	10.0	4.6999998	6.0999999	2020-05-21 00:00:00

图 11

>

3 <WinCC 链接服务器>

<WinCC 提供 WinCC OLE DB 接口，通过此接口可以访问 WinCC 的归档数据，其查询语句和标准的 SQL 语不同。其查询语句结构如下：

TAG:R,<ValueID or ValueName>,<TimeBegin>,<TimeEnd>[,<SQL_clause>][,<TimeStep>]

例如查询归档变量“PVArchive\Tag1”最近 10 分钟并且数值大于 100 的数据的查询语句如下：

TAG:R, 'PVA\tag1', '0000-00-00 00:10:00.000','0000-00-00 00:00:00.000','Where Realvalue>100'

WinCC OLE DB 的查询语句在报表的 ODBC 数据库接口中是无法执行的，如图 12 所示。



图 12

如果改成 select * from openquery (LnkRtdb_WinCCOLEDB,' TAG:R, "PVA\tag1", "0000-00-00 00:10:00.000", "0000-00-00 00:00:00.000")', 则可以正常执行，如图 13 所示。

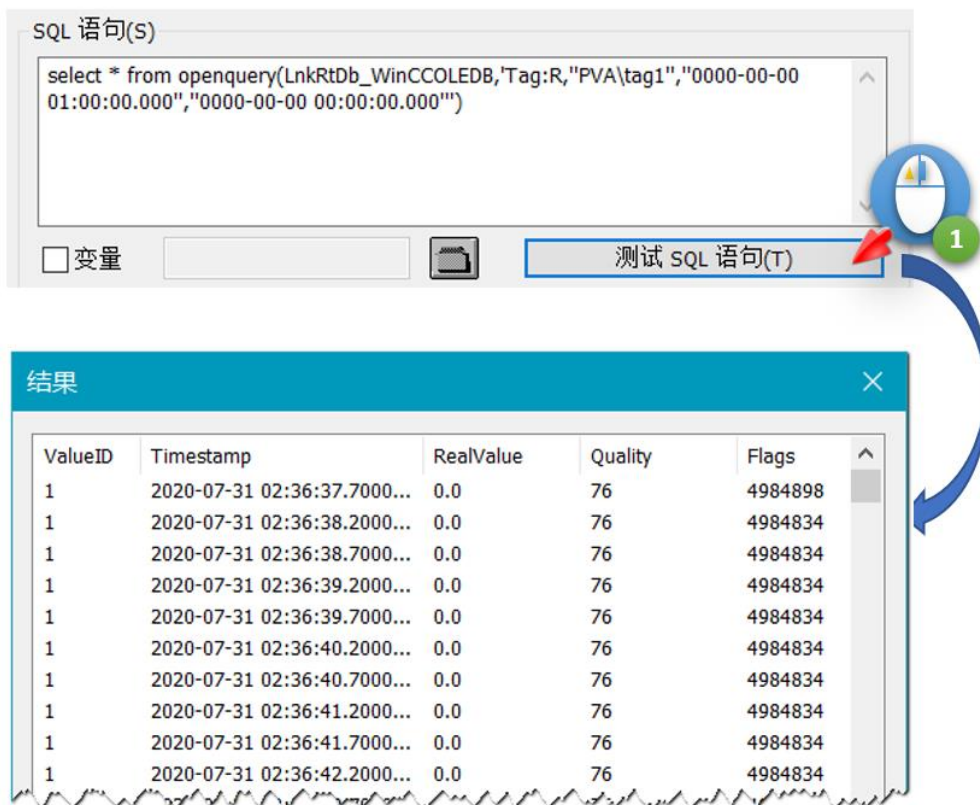


图 13

其中“LnkRtDb_WinCCOLEDB”就是系统自带的 WinCC 链接服务器，通过“LnkRtDb_WinCCOLEDB”可以使用标准的 SQL 语句进行查询。也可以自定义链接服务器，当将 WinCC OLE DB 提供程序组态为“链接的服务器”时，必须在“链接的服务器属性”对话框的“提供程序名”域中选择“归档的 WinCC OLE DB 提供程序”条目。同时也可以为链接服务器指定远程数据库。如图 14 所示。

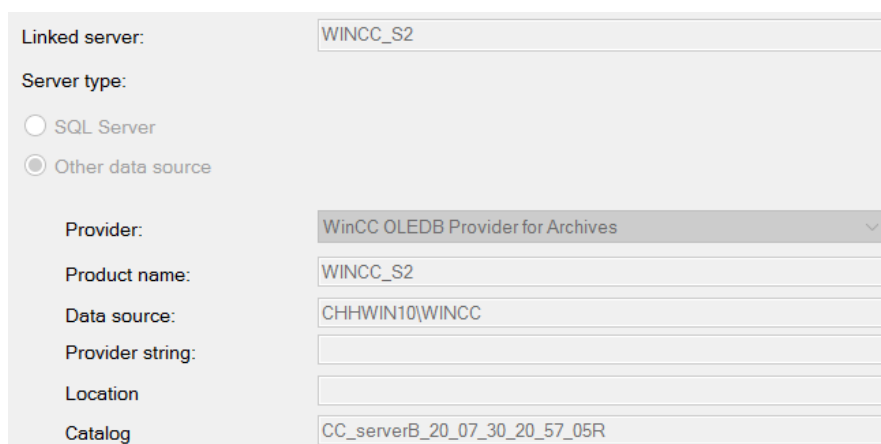


图 14

>

4 < WinCC 报表中显示多个 WinCC 的数据>

<下面介绍在 WinCC 报表中显示多个 WinCC 的归档数据的组态过程。>

4.1 <变量归档>

<本地 WinCC 的变量归档组态如图 15 所示。>



	过程变量	变量类型	变量名称	归档名称	采集类型
1	tag1	模拟量	tag1	PVA	周期 - 连续
2	tag2	模拟量	tag2	PVA	周期 - 连续
3	tag3	模拟量	tag3	PVA	周期 - 连续
4	tag4	模拟量	tag4	PVA	周期 - 连续
5	tag5	模拟量	tag5	PVA	周期 - 连续
6	tag6	模拟量	tag6	PVA	周期 - 连续
7	tag7	模拟量	tag7	PVA	周期 - 连续
8	tag8	模拟量	tag8	PVA	周期 - 连续
9	tag9	模拟量	tag9	PVA	周期 - 连续

图 15

远程 WinCC 的变量归档组态如图 16 所示。



	过程变量	变量类型	变量名称	采集类型
1	serB_tag1	模拟量	serB_tag1	周期 - 连续
2	serB_tag2	模拟量	serB_tag2	周期 - 连续
3	serB_tag3	模拟量	serB_tag3	周期 - 连续
4	serB_tag4	模拟量	serB_tag4	周期 - 连续
5	serB_tag5	模拟量	serB_tag5	周期 - 连续
6	serB_tag6	模拟量	serB_tag6	周期 - 连续
7	serB_tag7	模拟量	serB_tag7	周期 - 连续
8	serB_tag8	模拟量	serB_tag8	周期 - 连续
9	serB_tag9	模拟量	serB_tag9	周期 - 连续

图 16

>

4.2 <在 SQL 中添加链接归档>

<打开本地 SQL 管理器，选中“Server Objects”下的“Linked Servers”，右键，选中菜单中的“New Linked Server”，如图 17 所示。>

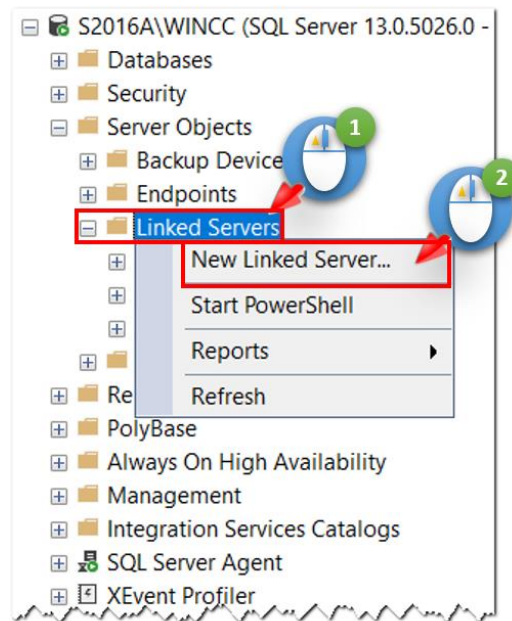


图 17

在弹出窗口中，选中“General”，设置链接服务器的属性。如图 18 所示。

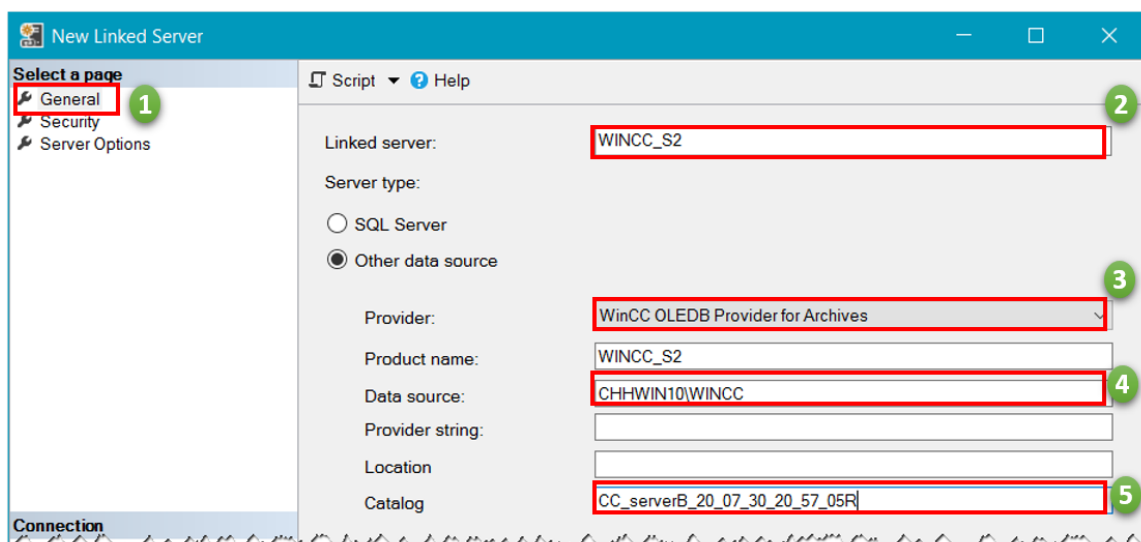


图 18

②设置链接服务名称。

③选择 Provider 为“WinCC OLEDB Provider for Archives”。

④本地数据源，Data source 可以直接写为“.\WINCC”。远程数据源，Data source 格式为“远程计算机名称\WINCC”。

⑤远程 WINCC 运行数据库名称。可以通过 WinCC 系统变量 “@DatasourceNameRT” 获得 WINCC 运行数据库名称，如图 19 所示。

4 < WinCC 报表中显示多个 WinCC 的数据>

变量 [内部变量]		
	名称	值
3	@CurrentUserName	
4	@DatasourceNameRT	CC_serverB_20_07_30_20_57_05R
5	@DeltaLoaded	0

图 19

选中“Security”，设置远程数据库的登录方式。如图 20 所示。

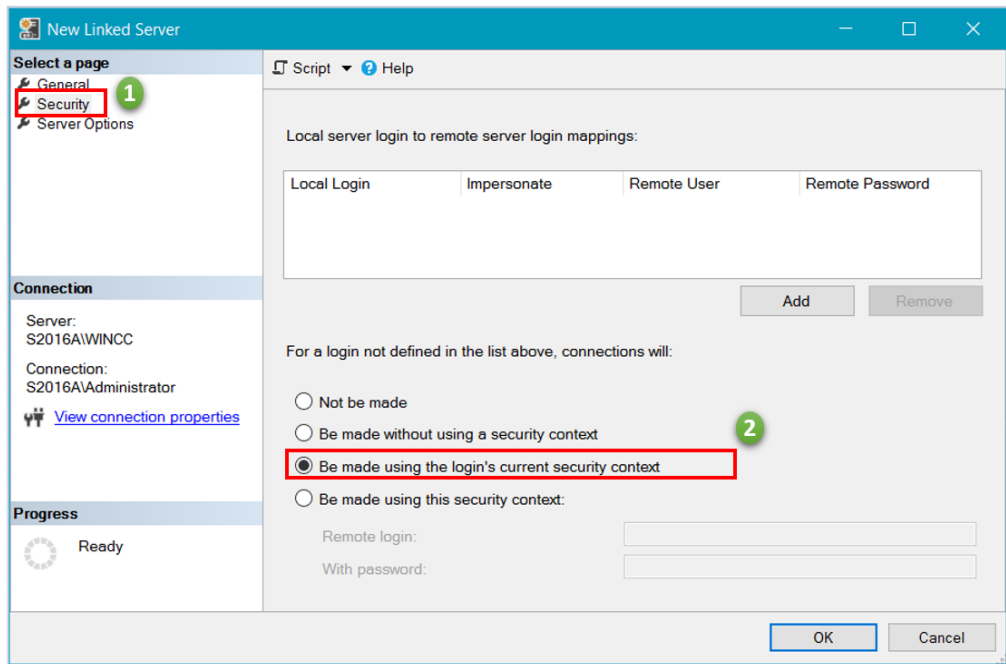


图 20

当本地计算机和远程计算机操作系统使用相同的用户名和密码登录时，此处选择“Be made using the login’s current security context”，否则需要在本地计算机用户管理下创建与远程计算机使用的用户名和密码相同的用户，然后此处选择“Be made using this security context”，并输入用户名和密码。

选中“Server Options”，设置链接服务器的选项。如图 21 所示。

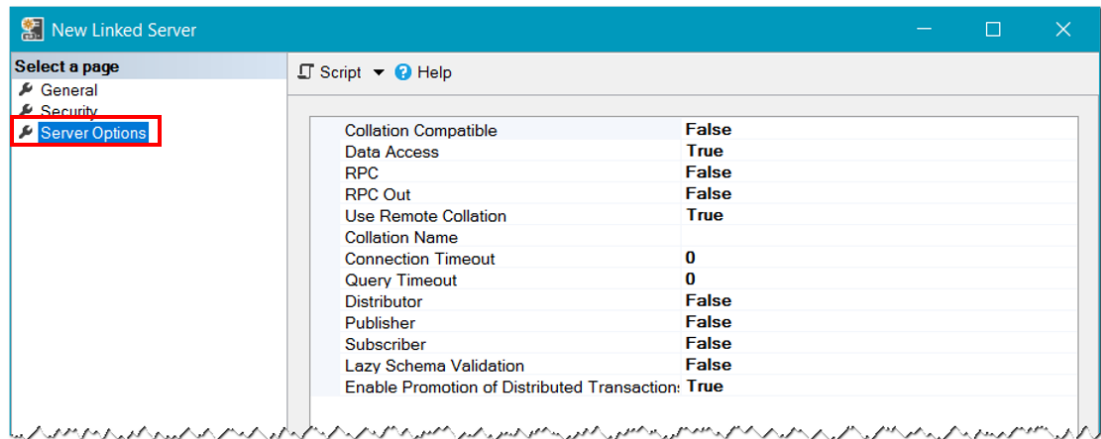


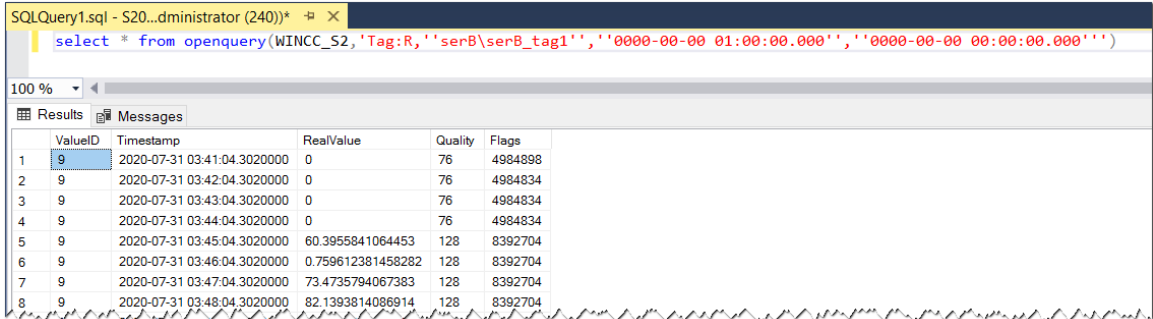
图 21

4 < WinCC 报表中显示多个 WinCC 的数据>

链接服务器创建完成后，可以在 SQL 中创建查询，使用以下的查询语句进行查询：

```
select * from openquery(WINCC_S2,'Tag:R,"serB\serB_tag1","0000-00-00 01:00:00.000","0000-00-00 00:00:00.000"')
```

查询结果如下图 22 所示。



ValueID	Timestamp	RealValue	Quality	Flags
9	2020-07-31 03:41:04.3020000	0	76	4984898
9	2020-07-31 03:42:04.3020000	0	76	4984834
9	2020-07-31 03:43:04.3020000	0	76	4984834
9	2020-07-31 03:44:04.3020000	0	76	4984834
9	2020-07-31 03:45:04.3020000	60.3955841064453	128	8392704
9	2020-07-31 03:46:04.3020000	0.759612381458282	128	8392704
9	2020-07-31 03:47:04.3020000	73.4735794067383	128	8392704
9	2020-07-31 03:48:04.3020000	82.1393814086914	128	8392704

图 22

>

4.3 <创建画面>

<创建以下内部变量，用来存储查询语句。如图 23 所示。



名称	注释	数据类型	长度
SQL_L1	SQL 查询语句1	文本变量 8 位字符集	255
SQL_L2	SQL 查询语句2	文本变量 8 位字符集	255
SQL_L3	SQL 查询语句3	文本变量 8 位字符集	255

图 23

在画面中添加两个 DT Picker 控件，用来选择报表的开始和结束时间，并分别为两个控件命名为“begintime”和“endtime”。如图 24 所示。

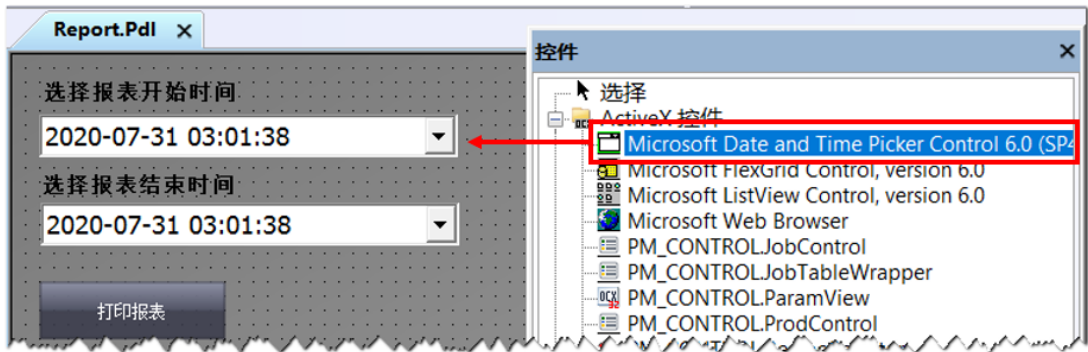


图 24

如果 WinCC 控件中找不到 DT Picker 控件，可以在“Active X 控件”上右键并选择“添加/删除”，以添加控件，如图 25 所示。

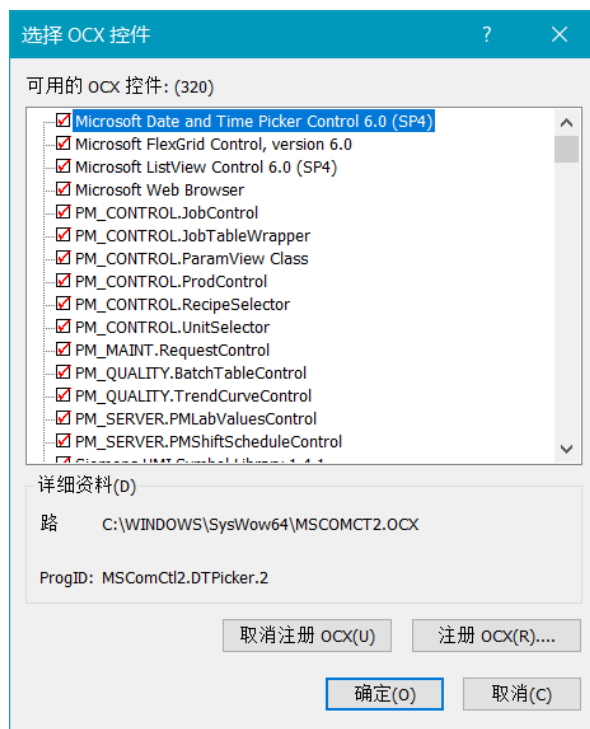


图 25

设置两个 DT Picker 控件的属性，“Format”设为“3-dtpCustom”，“CustomFormat”设为“yyyy-MM-dd HH:mm:ss”，如图 26 所示。

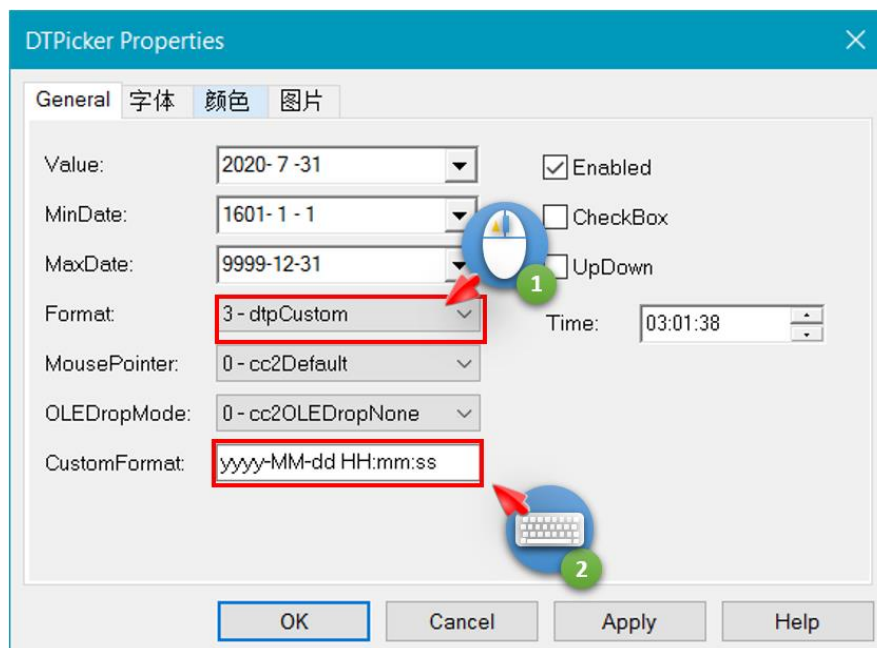


图 26

在两个 DT Picker 控件的“Change”事件（如图所示）下都编写生成查询语句的脚本。如图 27 所示。

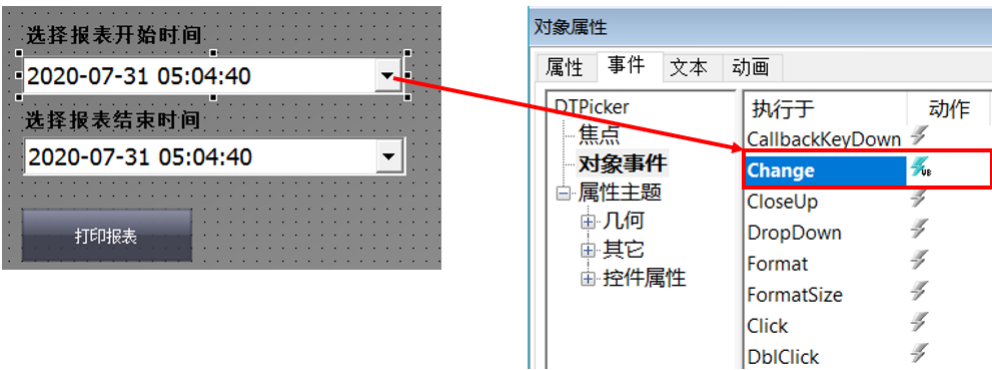


图 27

脚本如下图 28 所示：

```
Sub Change(ByVal Item)
On Error Resume Next
Dim timepicker, timepicker2
Set timepicker = ScreenItems("begintime")'获取报表开始时间
Set timepicker2 = ScreenItems("endtime")'获取报表结束时间
Dim SQL_L1
Set SQL_L1 = HMIRuntime.Tags("SQL_L1")'本地数据查询语句
Dim SQL_L2
Set SQL_L2 = HMIRuntime.Tags("SQL_L2")'远程数据查询语句
SQL_L1.Write "Select * from openquery (LnkRtdb_WinCCOLEDB,'Tag:R,('PVA\tag1';'PVA\tag2';'PVA\tag3'),'"& timepicker.Value & ".000'",'"& timepicker2.Value & ".000'")"
SQL_L2.Write "select * from openquery (WINCC_S2,'Tag:R,('serB\serB_tag1';'serB\serB_tag2';'serB\serB_tag3'),'"& timepicker.Value & ".000'",'"& timepicker2.Value & ".000'")"
End Sub
```

图 28

具体的脚本：

```
On Error Resume Next
Dim timepicker, timepicker2
Set timepicker = ScreenItems("begintime")'获取报表开始时间
Set timepicker2 = ScreenItems("endtime")'获取报表结束时间
Dim SQL_L1
Set SQL_L1 = HMIRuntime.Tags("SQL_L1")'本地数据查询语句
Dim SQL_L2
Set SQL_L2 = HMIRuntime.Tags("SQL_L2")'远程数据查询语句
SQL_L1.Write "Select * from openquery
(LnkRtdb_WinCCOLEDB,'Tag:R,('PVA\tag1";"PVA\tag2";"PVA\tag3"),'"& timepicker.Value
& ".000",'"& timepicker2.Value & ".000'")"
SQL_L2.Write "select * from openquery
(WINCC_S2,'Tag:R,('serB\serB_tag1";"serB\serB_tag2";"serB\serB_tag3"),'"&
timepicker.Value & ".000",'"& timepicker2.Value & ".000'")"
>
```

4.4 <报表组态>

<打开 WinCC 报表布局编辑器，在报表静态部分加入报表标题，在报表动态部分加入两个文本、两个 ODBC 数据表（具体参考图），如图 29 所示。

本地归档数据

[illegible]

Server归档数据

[illegible]

图 29

本地归档数据 ODBC 数据表的数据连接属性如图 30 所示。这里的 ODBC 数据源选择本地 WinCC 的运行数据库，查询语句使用 WinCC 变量“SQL_L1”，并使能“显示列的标题”。



图 30

远程归档数据 ODBC 数据表的 ODBC 数据源也是选择本地 WinCC 的运行数据库，查询语句使用 WinCC 变量“SQL_L2”，使能“显示列的标题”。如图 31 所示。



图 31

4 < WinCC 报表中显示多个 WinCC 的数据>

设置两个 ODBC 数据表控件的列宽，设置结果如图 32 所示。

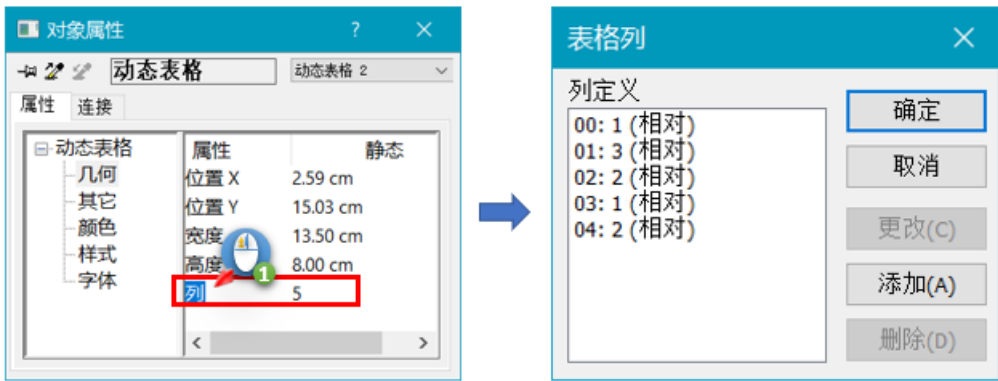


图 32

创建 WinCC 报表打印作业，选择前面创建的报表布局及打印机，如图 33 所示。



图 33

在 WinCC 画面中添加打印按钮，使用 PRTJobPrint 函数打印报表。如图 34 所示。

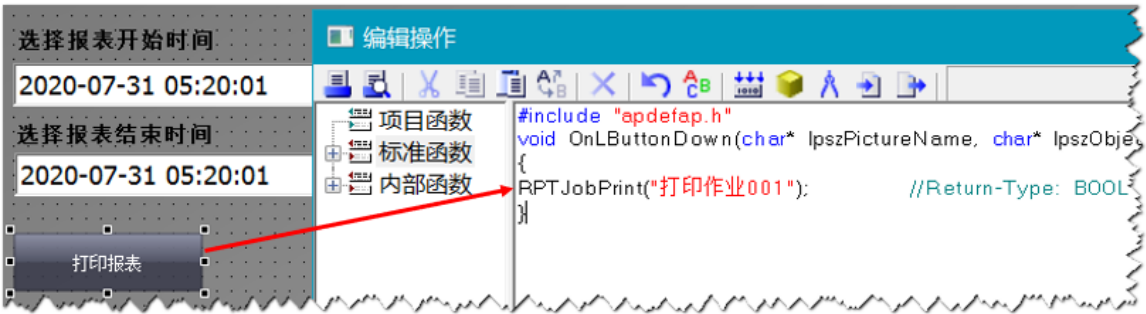


图 34

>

4.5 <运行结果>

<在 WinCC 启动列表中使能“WinCC 变量记录系统”，并激活 WinCC 运行系统。

在 WinCC 画面中分别设置报表开始时间和结束时间，如图 35 所示。点击“打印报表”按钮。



图 35

打印结果如下图 36 所示。在 WinCC 单机报表中同样可以显示多个 WinCC 的数据。

Linked Server 数据报表

本地归档数据

ValueID	Timestamp	RealValue	Quality	Flags
1	2020-07-31 05:00:32.333000	0.0	76	4984834
1	2020-07-31 05:01:32.333000	61.0	96	6295552
1	2020-07-31 05:02:32.333000	65.0	96	6295552
1	2020-07-31 05:03:32.333000	68.0	96	6295552
1	2020-07-31 05:04:32.333000	68.0	96	6295552
9	2020-07-31 05:00:32.333000	0.0	76	4984834
9	2020-07-31 05:01:32.333000	192.0	96	6295552
9	2020-07-31 05:02:32.333000	176.0	96	6295552
9	2020-07-31 05:03:32.333000	160.0	96	6295552
9	2020-07-31 05:04:32.333000	97.0	96	6295552
10	2020-07-31 05:00:32.333000	0.0	76	4984834
10	2020-07-31 05:01:32.333000	188.0	96	6295552
10	2020-07-31 05:02:32.333000	128.0	96	6295552
10	2020-07-31 05:03:32.333000	68.0	96	6295552
10	2020-07-31 05:04:32.333000	8.0	96	6295552

ServerB归档数据

ValueID	Timestamp	RealValue	Quality	Flags
9	2020-07-31 05:00:04.302000	60.395584106445	128	8392704
9	2020-07-31 05:01:04.302000	0.7596123814582	128	8392704
9	2020-07-31 05:02:04.302000	73.473579406738	128	8392704
9	2020-07-31 05:03:04.302000	82.139381408691	128	8392704
9	2020-07-31 05:04:04.302000	2.4471740722656	128	8392704
8	2020-07-31 05:00:04.302000	44.413673400878	128	8392704
8	2020-07-31 05:01:04.302000	44.413673400878	128	8392704
8	2020-07-31 05:02:04.302000	44.413673400878	128	8392704
8	2020-07-31 05:03:04.302000	44.413673400878	128	8392704
8	2020-07-31 05:04:04.302000	44.413673400878	128	8392704
7	2020-07-31 05:00:04.302000	65.0	128	8392704
7	2020-07-31 05:01:04.302000	79.0	128	8392704
7	2020-07-31 05:02:04.302000	35.0	128	8392704
7	2020-07-31 05:03:04.302000	48.0	128	8392704
7	2020-07-31 05:04:04.302000	168.0	128	8392704

图 36