



Customizing – Create a Data view

Extension of the tree structure with own views

TITLE	Customizing – Create a Data view
AUTHOR	Docusnap Consulting
DATE	12/12/2023
VERSION	3.0 valid from January 11, 2023

This document contains proprietary information and may not be reproduced in any form or parts whatsoever, nor may be used by or its contents divulged to third parties without written permission of Docusnap GmbH. All rights reserved.

CONTENTS

1.	INTRODUCTION	4
2.	PREPARATIONS	5
2.1	IDENTIFY DATA SOURCES	5
2.2	CREATION OF AN SQL QUERY	7
2.3	OPTIMIZATION OF THE SQL QUERY	8
2.4	LIMITING THE SQL QUERY	8
3.	IMPLEMENTATION	9
3.1	VIEW APPLICATION	9
3.1.1	NAMESPACES	10
3.2	CREATE CAPTION OBJECT	11
3.3	CREATE DATA OBJECT	11
3.4	USING THE VIEW IN DOCUSNAP CONNECT	12
4.	FURTHER TOPICS	13
4.1	OFFER OF VIEWNAMES	13

1. INTRODUCTION

By creating your own view, you can merge data from different tables and display it in a list. SQL knowledge is required to create your own view. Furthermore, a more intensive familiarization with the table structure of the Docusnap database is necessary.

In addition to creating your own views via the Customizing area, you can also implement your own queries with Docusnap Connect. The idea behind Docusnap Connect is to select and display specific data content via the user interface quickly and easily. SQL knowledge and in-depth knowledge of the Docusnap database are not a prerequisite here. Detailed information about Docusnap Connect can be found in a separate HowTo in the [Knowledge Base](#) - Docusnap Connect - Creating your own queries and exporting data.

For requirements that cannot be covered by Docusnap Connect, this document helps you to create your own view. It describes the possibility to create own views on the data of the Docusnap database in the tree structure. It is a guide for Microsoft SQL experienced system administrators.

2. PREPARATIONS

2.1 IDENTIFY DATA SOURCES

The first step is to identify the data sources (tables and columns) that you need within the view and familiarize yourself with them.

You can manage this in Administration - Customizing - Manage Objects.

This area contains the structure of the data tree. The data tree is built from meta objects of different categories. The two most common categories are **heading** and **dates**.

The relevant category for you is **data**. On the following screenshot you can see that for the objects of the category **Data** the field **Table** is filled - **tDocu**. In this case, this is the table in which the data of the snapshots of a system are stored.

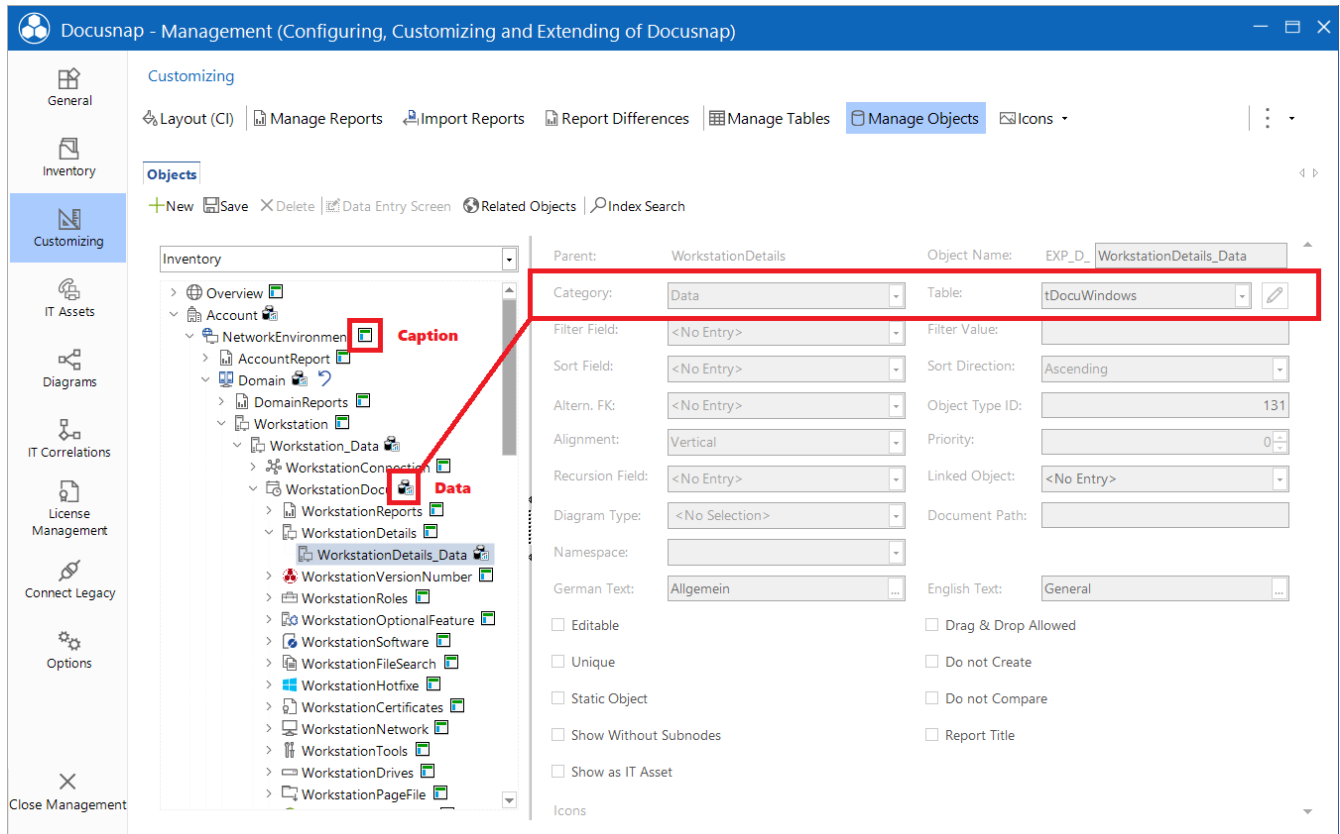


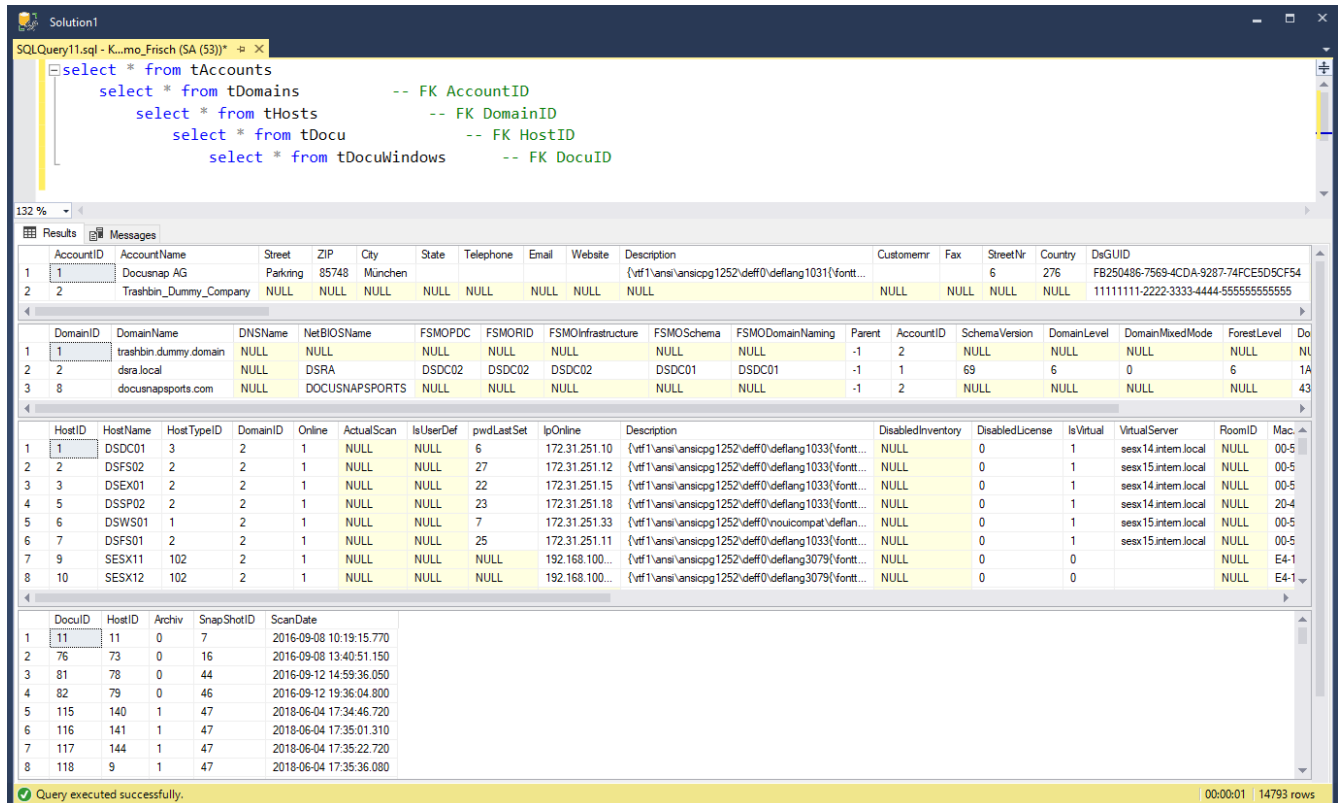
Fig. 1 - Docusnap data tree

Also check the tables for the following objects:

- Account
 - Domain
 - Server_Data
 - ServerDocu
 - ServerDetails_Data

tAccounts
tDomains
tHosts
tDocu
tDocuWindows

If you now look at these tables within SQL Management Studio, you will see the corresponding dependencies between them.



The screenshot shows a SQL query executed in SQL Server Enterprise Manager (SSMS). The query is as follows:

```
select * from tAccounts
select * from tDomains      -- FK AccountID
select * from tHosts        -- FK DomainID
select * from tDocu         -- FK HostID
select * from tDocuWindows  -- FK DocuID
```

The results pane displays four tables of data:

AccountID	AccountName	Street	ZIP	City	State	Telephone	Email	Website	Description	Customerr	Fax	StreetNr	Country	DaGUID
1	Docusnap AG	Parkring	85748	München					{vtf1\ansi\ansicpg1252\deff0\deflang1033\fontt...			6	276	FB250486-7569-4CDA-9287-74FCE5D5CF54
2	Trashbin_Dummy_Company													11111111-2222-3333-4444-555555555555

DomainID	DomainName	DNSName	NetBIOSName	FSMOPDC	FSMORID	FSMORInfrastructure	FSMOSchema	FSMODomainNaming	Parent	AccountID	SchemaVersion	DomainLevel	DomainMixedMode	ForestLevel	Do
1	trashbin.dummy.domain	NULL	NULL	NULL	NULL	NULL	NULL	NULL	-1	2	NULL	NULL	NULL	NULL	NU
2	dsra.local	NULL	DSRA	DSDC02	DSDC02	DSDC02	DSDC01	DSDC01	-1	1	69	6	0	6	1A
3	docusnapsports.com	NULL	DOCUSNAPSPTS	NULL	NULL	NULL	NULL	NULL	-1	2	NULL	NULL	NULL	NULL	43

HostID	HostName	HostTypeID	DomainID	Online	ActualScan	IsUserDef	pwdLastSet	IpOnline	Description	DisabledInventory	DisabledLicense	IsVirtual	VirtualServer	RoomID	Mac
1	DSDC01	3	2	1	NULL	NULL	6	172.31.251.10	{vtf1\ansi\ansicpg1252\deff0\deflang1033\fontt...	NULL	0	1	sesx14.intern.local	NULL	00-5
2	DSFS02	2	2	1	NULL	NULL	27	172.31.251.12	{vtf1\ansi\ansicpg1252\deff0\deflang1033\fontt...	NULL	0	1	sesx15.intern.local	NULL	00-5
3	DSEX01	2	2	1	NULL	NULL	22	172.31.251.15	{vtf1\ansi\ansicpg1252\deff0\deflang1033\fontt...	NULL	0	1	sesx14.intern.local	NULL	00-5
4	DSP02	2	2	1	NULL	NULL	23	172.31.251.18	{vtf1\ansi\ansicpg1252\deff0\deflang1033\fontt...	NULL	0	1	sesx14.intern.local	NULL	20-4
5	DSWS01	1	2	1	NULL	NULL	7	172.31.251.33	{vtf1\ansi\ansicpg1252\deff0\deflang1033\fontt...	NULL	0	1	sesx15.intern.local	NULL	00-5
6	DSFS01	2	2	1	NULL	NULL	25	172.31.251.11	{vtf1\ansi\ansicpg1252\deff0\deflang1033\fontt...	NULL	0	1	sesx15.intern.local	NULL	00-5
7	SESX11	102	2	1	NULL	NULL	NULL	192.168.100...	{vtf1\ansi\ansicpg1252\deff0\deflang3079\fontt...	NULL	0	0		NULL	E4-1
8	SESX12	102	2	1	NULL	NULL	NULL	192.168.100...	{vtf1\ansi\ansicpg1252\deff0\deflang3079\fontt...	NULL	0	0		NULL	E4-1

DocuID	HostID	Archiv	SnapShotID	ScanDate
1	11	0	7	2016-09-08 10:19:15.770
2	76	73	0	2016-09-08 13:40:51.150
3	81	78	0	2016-09-12 14:59:36.050
4	82	79	0	2016-09-12 19:36:04.800
5	115	140	1	2018-06-04 17:34:46.720
6	116	141	1	2018-06-04 17:35:01.310
7	117	144	1	2018-06-04 17:35:22.720
8	118	9	1	2018-06-04 17:35:36.080

Query executed successfully. 00:00:01 14793 rows

Fig. 2 - Analysis of tables in SSMS

2.2 CREATION OF AN SQL QUERY

In a view, an SQL query is required to determine the data.

This SQL query can be created, for example, with the Query Designer in SQL Management Studio.

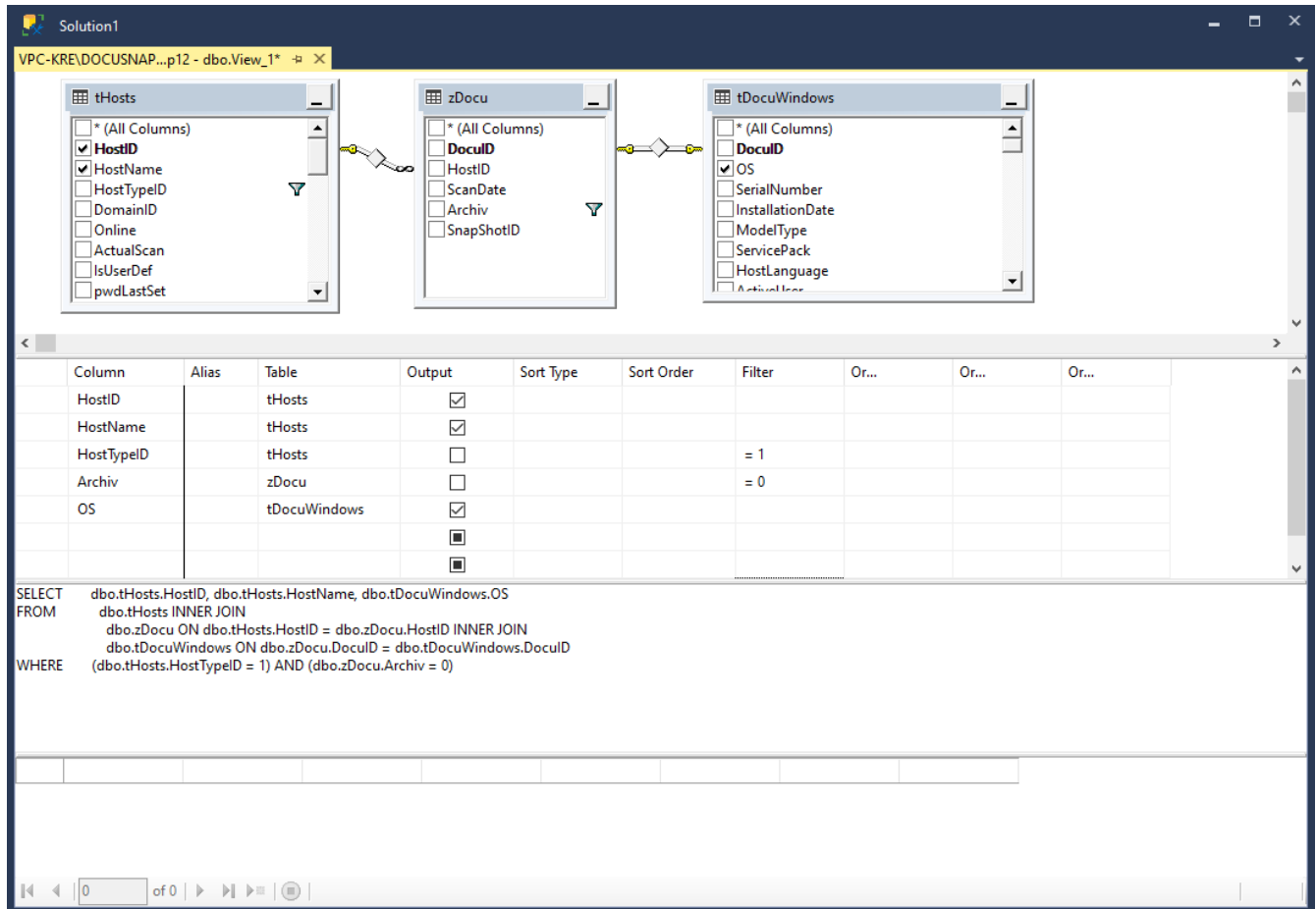


Fig. 3 - Query Designer

The tables required for the application example are

- **tHosts** contains information about the hosts inventoried (e.g. the host name)
- **tDocuWindows** contains general information about a host (e.g. the operating system)
- **tDocu** contains information about the different snapshots of the hosts

Since there can be several snapshots (inventoried) for one host, the query must be restricted to the most recent snapshot by **WHERE tDocu.Archiv = 0**. With the specification **tHosts.HostTypeID = 1** only workstation will be selected.

An overview for the different host types can be found in the table tHostTypes.

2.3 OPTIMIZATION OF THE SQL QUERY

The SQL query generated by the Query Designer:

```
SELECT      tHosts.HostID, tHosts.HostName, tDocuWindows.OS
FROM        tHosts INNER JOIN
            tDocu ON tHosts.HostID = tDocu.HostID INNER JOIN
            tDocuWindows ON tDocu.DocuID = tDocuWindows.DocuID
WHERE       (tHosts.HostTypeID = 1) AND (tDocu.Archiv = 0)
```

can be optimized / made easier to read as follows:

```
SELECT tHosts.HostID, tHosts.HostName, tDocuWindows.OS
FROM tHosts
    INNER JOIN tDocu ON tHosts.HostID = tDocu.HostID
    INNER JOIN tDocuWindows ON tDocu.DocuID = tDocuWindows.DocuID
WHERE tHosts.HostTypeID = 1
    AND tDocu.Archiv = 0
```

2.4 LIMITING THE SQL QUERY

The SQL query determined so far covers all entries in the Docusnap database, regardless of whether they belong to a client or a domain.

To ensure that the view only contains the devices that belong to the respective area in the tree structure (e.g. a domain), the SQL query must be extended by **AND tHosts.DomainID = {FilterID}**.

Advanced SQL query:

```
SELECT tHosts.HostID, tHosts.HostName, tDocuWindows.OS
FROM tHosts
    INNER JOIN tDocu ON tHosts.HostID = tDocu.HostID
    INNER JOIN tDocuWindows ON tDocu.DocuID = tDocuWindows.DocuID
WHERE tHosts.HostTypeID = 1 -- Filtering on workstation
    AND tDocu.Archiv = 0     -- Filtering on last inventory
    AND tHosts.DomainID = {FilterID} -- Filtering on Domain
```

The **FilterID** is a Docusnap internal variable that can be used to restrict a query to the parent object in the tree structure. In the application example, the view is mounted below the domain, whereby the **FilterID** dynamically contains the value of the **domainID** of the respective domain where it is called.

If, for example, the view was to be integrated directly under a company, the **FilterID** would dynamically have the value of the **AccountID** under which it is located. A comparison of the **AccountID** of a company with the **DomainID** of a host leads to a faulty SQL query, which returns either no result or a wrong result.

If you want to integrate the view underneath the account, you would filter like this:

```
SELECT tHosts.HostID, tHosts.HostName, tDocuWindows.OS
FROM tHosts
    INNER JOIN tDocu ON tHosts.HostID = tDocu.HostID
    INNER JOIN tDocuWindows ON tDocu.DocuID = tDocuWindows.DocuID
    INNER JOIN tDomains ON tHosts.DomainID = tDomains.DomainID -- join table tDomains
WHERE tHosts.HostTypeID = 1
    AND tDocu.Archiv = 0
    AND tDomains.AccountID = {FilterID} -- Filtering on Account
```


3. IMPLEMENTATION

3.1 VIEW APPLICATION

The new view can be created in Docusnap **Administration – Customizing – Manage Tables** using the **New** button.

The following information must be carried out:

- | | |
|-----------------|--|
| • Table Type | View |
| • Tablename | The tablename is a combination of the namespace and the tablename
<i>xv</i>
<i>HowToView</i>
<i>WorkstationOS</i> |
| • Primary key | cannot be selected until the fields are created |
| • Display field | cannot be selected until the fields are created |
| • German Name | <i>Arbeitsstationen mit Betriebssystem</i> |
| • English Name | <i>Workstations with OS</i> |
| • SQL statement | the actual SQL statement |

After creating the view, the next step is to require the appropriate fields:

- HostID
- Hostname
- OS

Via the button **Edit fields** they can be created or added from other tables.

Adding fields from other tables means that you select and add the fields to display from existing tables or views. This saves time, especially if you are creating a richer view. In this case, you can add the fields from the **tHosts** and **tDocuWindows** table:

- tHosts
 - HostID
 - HostName
- tDocuWindows
 - OS

Alternatively, you can create the fields manually.

Field Name	Data type	German Name	English Name	Show field in lists	Show field in web client	Sorting	Display Length
HostID	Int	HostID	HostID	No	No	0	
HostName	String	Name	Name	And	And	10	200
THE	String	Operating system	Operating System	And	And	20	250

For more information on the available options, refer to the User's Guide, which you can access via the F1 key.

After creating the fields, the information regarding the primary key and the display field must be entered for the view. The primary key does not necessarily have to be unique. However, you should be careful to use a numeric value (INT, BigINT) as the primary key!

3.1.1 NAMESPACES

The namespace serves the assignment of the view, for example to a more extensive customizing. In this way, the view and the metaobjects to be created later can be related to each other. This is especially useful if a specific customizing is to be exported from a Docusnap environment.

3.2 CREATE CAPTION OBJECT

To display the view in the Docusnap tree, an object of type **Caption** must be created in Customizing - Manage Objects. The caption is included below the domain.

- To do this, click **New**
- Object Name - WorkstationOS_Header
- Category - Caption
- Namespace - View - Workstations with Operating System
- German Text - Arbeitsstationen mit Betriebssystem
- English Text - Workstations with OS
- In the Icons area, you can store a standard icon (16x16) and a preview icon (100x100)
 - You can download an icon pack in our [community](#)

For more information on the additional options available, refer to the User's Guide, which you can access via the F1 key.

3.3 CREATE DATA OBJECT

The next step is to create a new metaobject of type Data.

- To do this, click **New**
- Object Name - WorkstationOS_Data
- Category - Data
- Table - xvWorkstationOS
- Sort Field - Hostname
- Namespace - View - Workstations with Operating System
- German Text - Arbeitsstationen mit Betriebssystem
- English Text - Workstations with OS
- Re-enter the icons

3.4 USING THE VIEW IN DOCUSNAP CONNECT

If you now want to use the view in a Docusnap Connect package, you will receive the following error message:

Cannot connect to parent table. Please insert the foreign key 'DomainID' for the customized table 'xvWorkstationOS' (EXP_U_WorkstationOS_Data)!

In this case, the foreign key is the DomainID (tHosts.DomainID) since the view was included under the domain – see chapter [Limiting the sql query](#). The foreign key – tHosts.DomainID – simply must be queried within the view. Adapt the SQL statement to the previously created view. The view can then be used in Docusnap Connect.

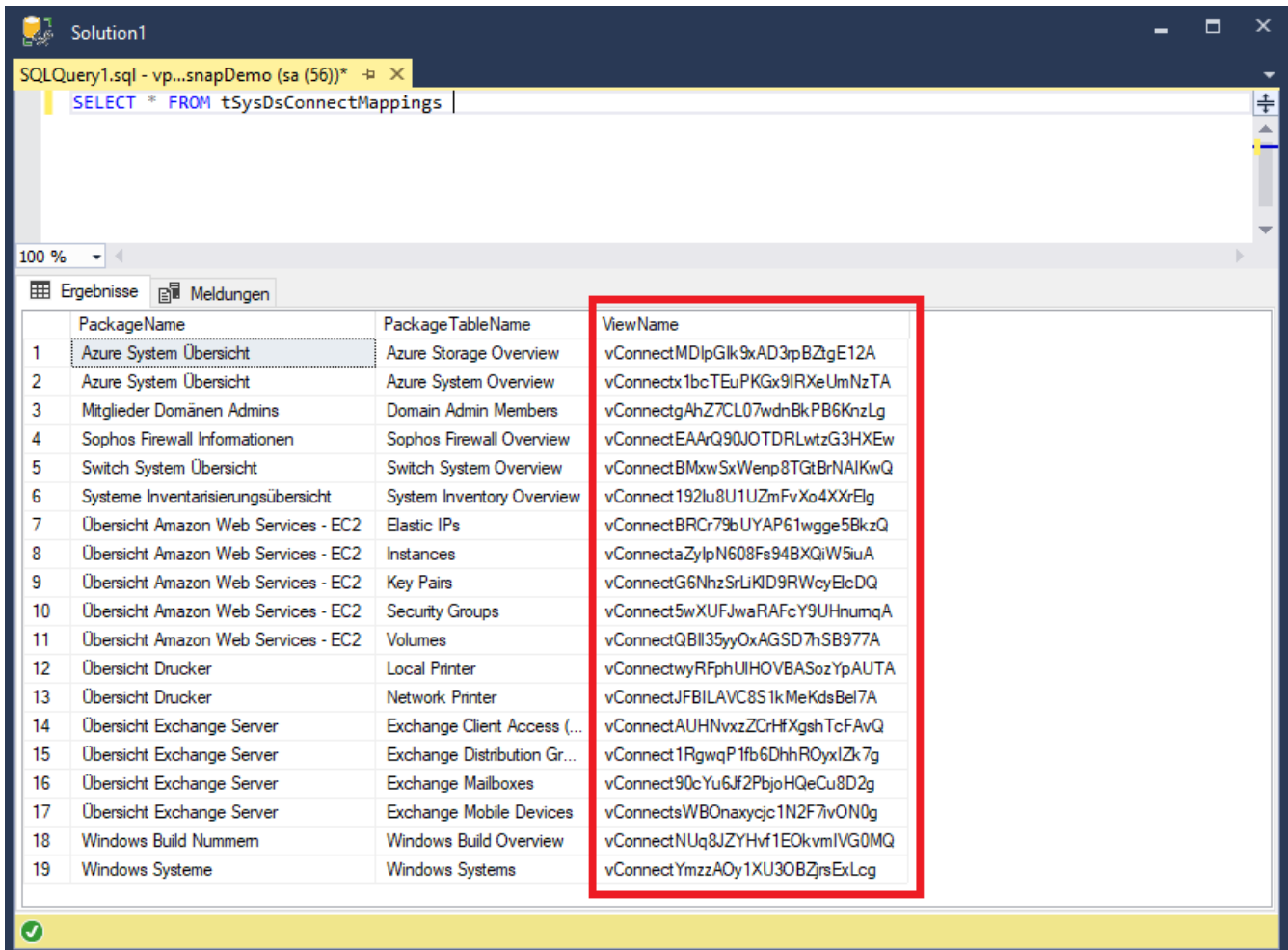
```
SELECT tHosts.HostID, tHosts.HostName, tDocuWindows.OS, tHosts.DomainID
FROM tHosts
    INNER JOIN tDocu ON tHosts.HostID = tDocu.HostID
    INNER JOIN tDocuWindows ON tDocu.DocuID = tDocuWindows.DocuID
WHERE tHosts.HostTypeID = 1
    AND tDocu.Archiv = 0
    AND tHosts.DomainID = {FilterID}
```

4. FURTHER TOPICS

4.1 OFFER OF VIEWNAMES

As already described in the introduction, individual requirements can also be easily implemented using Docusnap Connect.

Additionally, it is possible to view the previously created Connect packages via the table `tSysDsConnectMappings`, so that the corresponding name of the Connect package can be used for a view, for example.



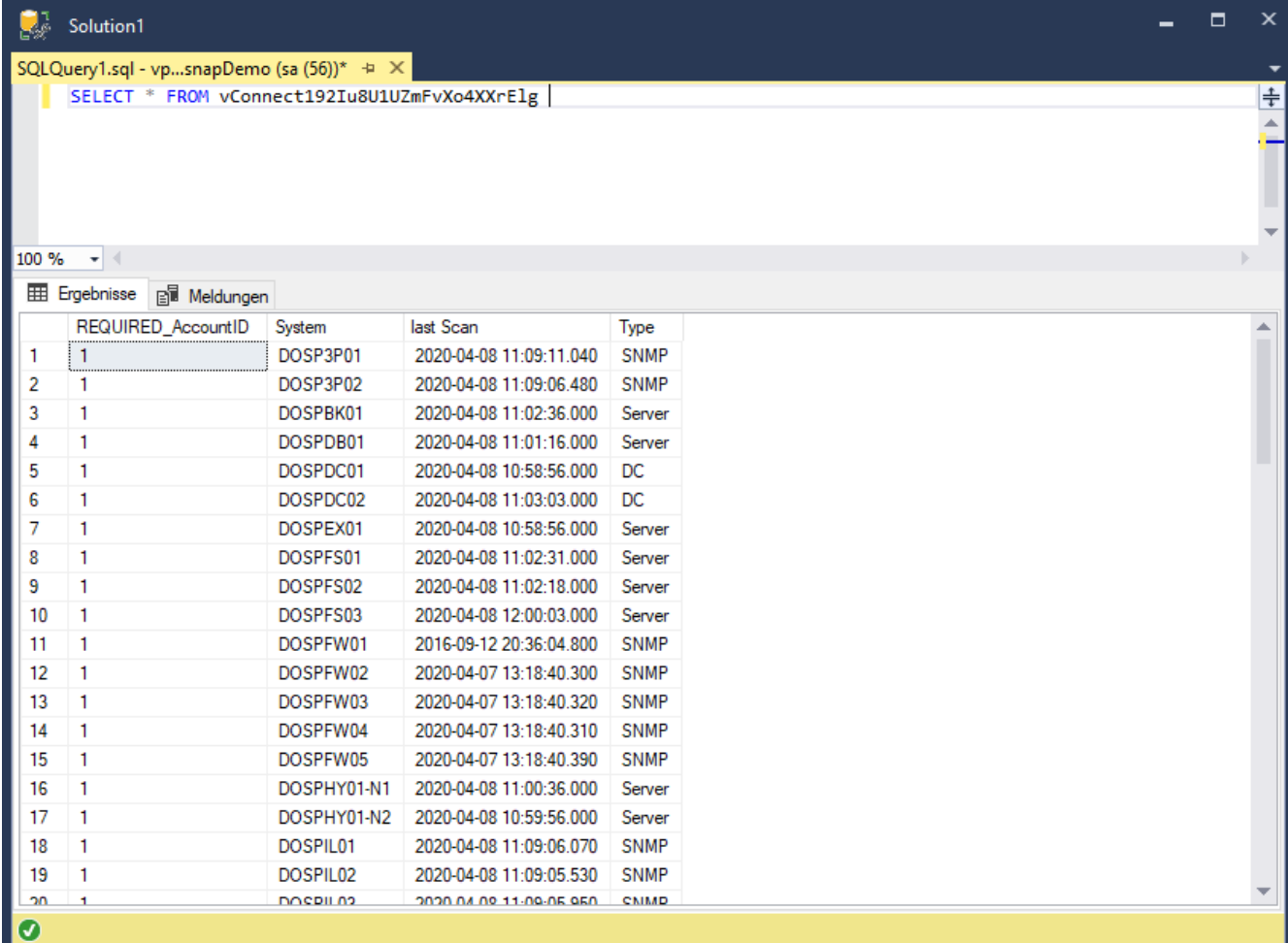
SQLQuery1.sql - vp...snapDemo (sa (56))*

```
SELECT * FROM tSysDsConnectMappings
```

	PackageName	PackageTableName	ViewName
1	Azure System Übersicht	Azure Storage Overview	vConnectMDlpGik9xAD3pBZtgE12A
2	Azure System Übersicht	Azure System Overview	vConnectx1bcTEuPKGx9IRXeUmNzTA
3	Mitglieder Domänen Admins	Domain Admin Members	vConnectgAhZ7CL07wdnBkPB6KnzLg
4	Sophos Firewall Informationen	Sophos Firewall Overview	vConnectEAArQ90JOTDRLwtzG3HXEw
5	Switch System Übersicht	Switch System Overview	vConnectBMxwSxWenp8TGtBrNAIKwQ
6	Systeme Inventarisierungsübersicht	System Inventory Overview	vConnect192lu8U1UZmFvXo4XXrElg
7	Übersicht Amazon Web Services - EC2	Elastic IPs	vConnectBRCr79bUYAP61wgge5BkzQ
8	Übersicht Amazon Web Services - EC2	Instances	vConnectaZylpN608Fs94BQiW5iuA
9	Übersicht Amazon Web Services - EC2	Key Pairs	vConnectG6NhzSrLiKD9RWcyElcDQ
10	Übersicht Amazon Web Services - EC2	Security Groups	vConnect5wXUFJwaRAFcY9UHnumqA
11	Übersicht Amazon Web Services - EC2	Volumes	vConnectQBII35yyOxAGSD7hSB977A
12	Übersicht Drucker	Local Printer	vConnectwyRFphUIHOVBASozYpAUTA
13	Übersicht Drucker	Network Printer	vConnectJFBILAVC8S1kMeKdsBel7A
14	Übersicht Exchange Server	Exchange Client Access (...)	vConnectAUHNvxzZCrHfXgshTcFAvQ
15	Übersicht Exchange Server	Exchange Distribution Gr...	vConnect1RgwqP1fb6DhhROyxIZk7g
16	Übersicht Exchange Server	Exchange Mailboxes	vConnect90cYu6Jf2PbjoHQeCu8D2g
17	Übersicht Exchange Server	Exchange Mobile Devices	vConnectsWBOxaycjc1N2F7ivON0g
18	Windows Build Nummern	Windows Build Overview	vConnectNUq8JZYHvf1EOkvmIVG0MQ
19	Windows Systeme	Windows Systems	vConnectYmzzAOy1XU30BZjrsExLcg

Fig. 4 - Table with Connect Packages

The view name can be used to list the content of the Connect package and to edit or extend it at any time.



SQLQuery1.sql - vp...snapDemo (sa (56))*

`SELECT * FROM vConnect192Iu8U1UZmFvXo4XXrElg`

100 %

Ergebnisse Meldungen

	REQUIRED_AccountID	System	last Scan	Type
1	1	DOSP3P01	2020-04-08 11:09:11.040	SNMP
2	1	DOSP3P02	2020-04-08 11:09:06.480	SNMP
3	1	DOSPBK01	2020-04-08 11:02:36.000	Server
4	1	DOSPD01	2020-04-08 11:01:16.000	Server
5	1	DOSPDC01	2020-04-08 10:58:56.000	DC
6	1	DOSPDC02	2020-04-08 11:03:03.000	DC
7	1	DOSPEX01	2020-04-08 10:58:56.000	Server
8	1	DOSPFS01	2020-04-08 11:02:31.000	Server
9	1	DOSPFS02	2020-04-08 11:02:18.000	Server
10	1	DOSPFS03	2020-04-08 12:00:03.000	Server
11	1	DOSPFW01	2016-09-12 20:36:04.800	SNMP
12	1	DOSPFW02	2020-04-07 13:18:40.300	SNMP
13	1	DOSPFW03	2020-04-07 13:18:40.320	SNMP
14	1	DOSPFW04	2020-04-07 13:18:40.310	SNMP
15	1	DOSPFW05	2020-04-07 13:18:40.390	SNMP
16	1	DOSPHY01-N1	2020-04-08 11:00:36.000	Server
17	1	DOSPHY01-N2	2020-04-08 10:59:56.000	Server
18	1	DOSPIL01	2020-04-08 11:09:06.070	SNMP
19	1	DOSPIL02	2020-04-08 11:09:05.530	SNMP
20	1	DOSPIL02	2020-04-08 11:09:05.950	SNMP

Fig. 5 - Table with Connect Packages

LIST OF FIGURES

FIG. 1 - DOCUSNAP DATA TREE.....	5
FIG. 2 - ANALYSIS OF TABLES IN SSMS	6
FIG. 3 - QUERY DESIGNER.....	7
FIG. 4 - TABLE WITH CONNECT PACKAGES	13
FIG. 5 - TABLE WITH CONNECT PACKAGES	14

VERSION HISTORY

Date	Description
April 29, 2020	Version 2.0 - Revision of the HowTos for Docusnap 11
November 30, 2022	Version 2.1 - Adding the namespaces during creation / New chapter Using the view in Docusnap Connect
January 11, 2023	Version 3.0 – Revision of the HowTo for Docusnap 12

