



DocuSnap X

Creating Custom IT Documentation Based on Business Structures

TITLE	Docusnap X
AUTHOR	itelio GmbH
DATE	7/12/2017
VERSION	1.0 valid from October 05, 2016

The reproduction and distribution of this document as a whole or in part as well as the utilization and disclosure of its contents to third parties without the express authorization by itelio GmbH are prohibited. Offenders will be held liable for the payment of indemnification. All rights reserved.

CONTENTS

1. Introduction	4
1.1 What is the purpose of a Framework?	4
1.2 Framework Terms and Concepts	4
1.3 Sample Documentation	5
1.4 How was this sample framework created?	7
2. Creating a Documentation Based on a Framework	9
2.1 Creating a New Framework	9
2.2 Creating Views	11
2.3 Creating Levels	12
2.4 Adding Elements	14
2.4.1 Linking elements	16
2.4.2 Moving elements	17
3. Advanced Features	18
3.1 Adding Structural Elements	18
3.2 Creating Groups	20
3.2.1 Assigning elements to a group	21
3.2.2 Linking elements to groups	22
3.3 Properties	23
3.3.1 Defining properties	23
3.3.2 Assigning a property to a level	24
3.3.3 Assigning values to properties	25
3.4 Exporting a Framework	25
3.5 Useful Tip: Customizing the Screen Display	25

1. Introduction

This document describes how use the Docusnap documentation suite to create a user-defined framework for the most diverse IT documentation types.

1.1 What is the purpose of a Framework?

A Docusnap framework can be used to define the structure of your IT documentation in Docusnap, i.e. your own set of rules taken into account when creating your IT documents. This means that you no longer need to stick to the software manufacturer's default definitions when as regards the structure of your documentation and the sequence of its creation. How can the framework support you in your work?

Depending on the objects to be represented in your documentation (OBASHI model, IT procedure, process/IT correlations, etc.), you can create the required documentation directly in the Docusnap software based on a framework. So there is no need for additional visualization software. However, creating IT documentation in this way requires a certain amount of flexibility, which is provided by a framework. This means that Docusnap offers you a "construction kit" for the flexible design of your documentation. Why is this important?

It is important because the master data created or inventoried by you can then be used consistently for all parts of the documentation. This means that it is no longer necessary to maintain duplicate data in separate systems, but you can use the same master data for all parts of your documentation. This, in turn, reduces the rate of errors caused by erroneous data input and also ensures that the data are always up-to-date.

1.2 Framework Terms and Concepts

Before we present a sample IT documentation based on a framework, it is necessary that you understand the basic concepts and terms of a framework.

Framework

A framework is the top level of a self-contained documentation unit. Within the framework, the required views, levels, elements, structural elements, groups, and properties are created and related to each other.

Views

When using a framework, you can display the same data in different views, i.e. different representations that show the same data from other perspectives, and offer new insights. So the data can be arranged in various ways, for example under functional or technical aspects.

Levels

Levels can be used to visually represent connected elements. Within a framework, a defined level and all elements assigned to it can be used in other views as well.

Structural elements

Using structural elements, you can define sub-levels for each level. These help you better organize the elements in your framework.

Elements

Each element is assigned to a certain level. Basically, you can link the elements to Docusnap inventory structure objects. This step is performed in the Editor.

Groups

Multiple elements can be grouped into visual or logical units using groups. Groups improve the visualization and, in some scenarios, ensure the technical correctness of the representation.

Properties

For each element, you can specify various properties that further describe them or document certain characteristics. Properties can be defined freely, e.g. "downtime" for a server. Properties are always assigned to a level and thus to all elements associated with it.

1.3 Sample Documentation

In this section, a sample framework will be described. The purpose of this sample documentation is to show the relation between the business processes of a company and its IT landscape. This way, it is possible to analyze which hardware and software are required for particular tasks in the business processes.

Using this framework, you can link "both worlds" existing in the company, so that the correlations and interdependencies between business and IT are documented. Thus, based on the availability requirements to business processes, requirements for the IT department with respect to the availability of their IT systems can be derived. All this is possible because the direct correlations can be analyzed and visualized.

The documentation enables you to address the following questions:

- Which business processes are affected by the failure of certain IT systems?
- Why should the company invest in certain IT systems? Which business processes are affected? Where are investments worthwhile?
- Which processes can be adapted with a reduction of the number of IT systems in mind? Which hardware can possibly be switched off, and which software maintenance contracts could be terminated?
- Which systems are responsible for bottlenecks in the IT infrastructure? Which of them should therefore be highly available so that business processes are not affected at all or as little as possible?

The following figure shows an example of such a documentation. From the top level, which is the organizational unit, it goes down to the individual hardware items that must ultimately be available for the employees to complete their tasks.

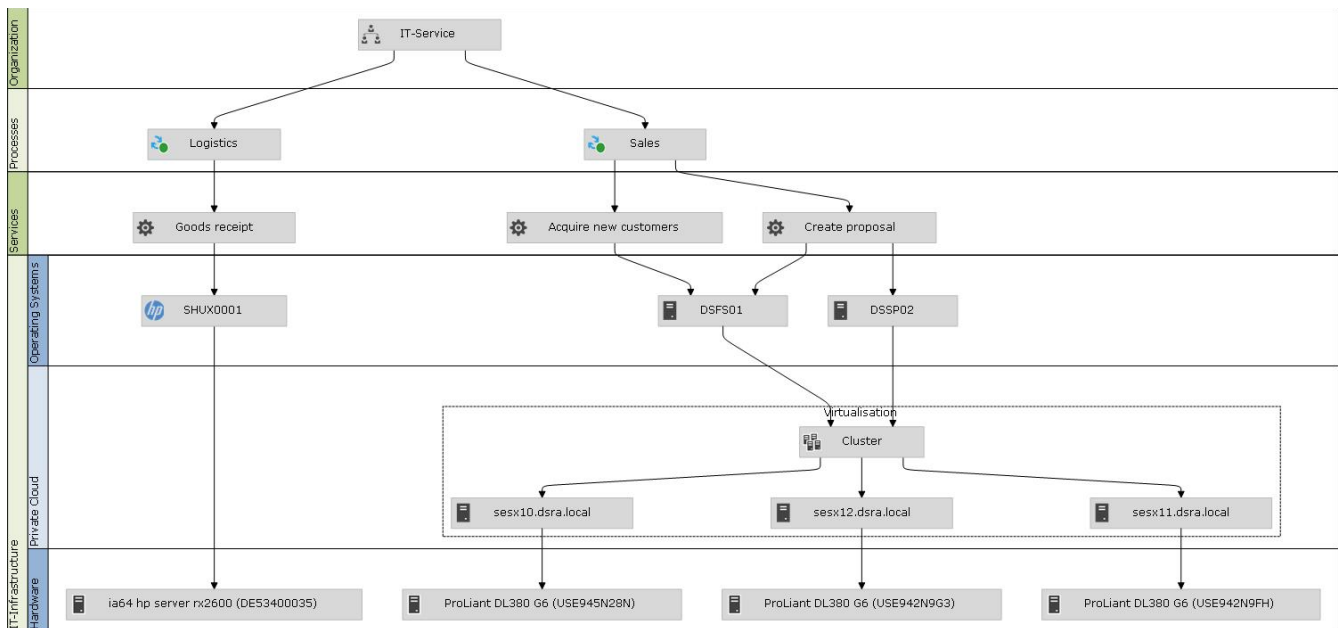


Figure 1 – Sample IT documentation

At the top level, Organization, the **IT Services** department is located. This department is involved in the two processes **Logistics** and **Sales** where it performs certain processes.

For these two processes, the services **Goods receipt** as well as **Acquire new customers** and **Create proposal** are required. These are the services that the department provides within the two processes and that affect the business activities of the company. All of them need support by IT systems.

For the "Goods receipt" service, the **SHUX0001** operating system is required, perhaps because special software, that is required for processing the goods receipt, is installed under that operating system. This OS is directly installed on the hardware item called **ia64 hp server rx2600**.

For the other two services, two further servers are required: **DSFS01** and **DSSP02**. These might for example run under Windows Server 2008 and 2012. Furthermore, the documentation shows that these servers have been virtualized and grouped into a cluster. This means that the hardware they are actually running on does not matter. The cluster is distributed to three hosts: **sesx10.dsra.local**, **sesx11.dsra.local**, and **sesx12.dsra.local**. These hosts ultimately run on three different machines. These are all **ProLiant DL380 G6** servers, for which the serial numbers are given in the figure.

If desired, the network sockets and patch fields could be added to this documentation. In addition, it would be possible to document which processes would be affected if a circuit in the electrical distribution failed. This is possible because you can see which IT systems are connected there and, in turn, which processes depend on these IT systems. As you can see, a documentation based on a framework opens up entirely new possibilities as to the degree of detail and the volume of your IT documentation.

1.4 How was this sample framework created?

Let's have a closer look at the previous example. We will now describe how the framework for the documentation of the business/IT relations in Figure 1 there was created.

Framework

The first step, of course, is to create a new framework and name it. For example, you can specify the name of your company here.

Views

Once the framework is in place, a new view can be defined. Views help you model the documentation of certain interrelations independently of other views. Since this view serves the purpose of documenting function interdependencies, this view could be called "Functional".

Levels

Using levels, you can group the elements to be created in the next step logically and functionally. Levels are a graphical grouping of elements belonging to the same category. In our example, four levels were created so that the dependencies between processes and the IT landscape can be represented in a granular way.

Organization

On this level, the organizational unit involved in the processes of the levels below it is documented. An organizational unit can be a department, a team, or an entire corporate division.

Processes

This level lists the processes for which this documentation is being created. In the example, this level contains all processes related to an organizational unit that require support by IT systems. As computerization is growing more and more, these will be the majority of processes. Hardly any business process is nowadays conceivable without the support of certain IT systems.

Services

Services functionally bundle the activities performed in the processes. This level describes which services the corresponding organizational unit performs in each process.

When defining the services, always ask yourself the following question: Which activities are carried out by the organizational unit for the different processes? A Sales department, for example, usually creates proposals, sends them out and performs the follow-up by phone or in writing. These may be individual services supplied by this organizational unit on behalf of the company.

IT Infrastructure

The IT Infrastructure level is subdivided into the Operating Systems, Private Cloud, and Hardware levels. This subdivision (realized by structural elements) was selected both in order to make the the documentation clearer and achieve a better representation of the functional dependencies. The subdivision here shows the server operating systems and the associated hardware and virtualization environment. Thus, the interdependencies of the IT systems can be visualized in a granular and functionally correct manner. It does not matter whether an operating system is installed directly on the hardware, virtualized in its own virtualization environment, or available as a cloud solution. It should be possible to visualize any of these options in the framework. In this context, it is important to consider the operating system and the hardware as separate entities in the documentation.

First, the operating systems were documented, and then the servers. This assumes that the operating systems are either installed directly on the hardware or on a virtual computer. Further, multiple virtual machines can run on a single host. The hosts, in turn, may be located on certain storage systems that are either deployed in the company or are hired (hardware as a service) in the Internet. The increasing trend towards virtualization eliminates the former allocation of a single IT system to a particular hardware item – now, multiple systems share a host system which, for availability reasons, can be distributed to multiple storage systems.

Elements

Each level comprises several elements, in our example, there are one organizational unit, two processes, three services as well as multiple servers and operating systems. They were added to the framework and then linked with each other.

These elements originate from the Docusnap Inventory module. Elements can be created there manually or by using the Docusnap Inventory scan feature. Elements of the Organization, Processes, and Services types can also be created directly in the Editor, i.e. there is no need to create them in the Inventory module first.

Group

In order to represent the virtualization environment, a group named Virtualization was created, and the three hosts as well as the cluster were added to it.

As you can see, it sometimes makes sense to group elements within a level or structural element. An example of a virtualization group would be to group multiple hosts into a cluster. From a technical point of view, this is acceptable as virtual machines can either be added to a cluster or to a defined host. If they are added to a cluster, it does not matter on which hardware a particular virtual machine is running. For this reason, it can also be linked to cluster.

Structural elements

As described above in the IT Infrastructure section, three structural element were created: Operating Systems, Private Cloud, and Hardware.

Depending on the complexity of your documentation, it might be helpful to create further sub-levels on each level in order to achieve a clearer representation of the relations. Structural elements can be used for this purpose.

2. Creating a Documentation Based on a Framework

2.1 Creating a New Framework

When creating the documentation, the first step will be to create a new framework. This is the top level of your documentation. Frameworks can be created in two ways: Either directly from the ribbon (user interface) or by using the Editor.

Creating the framework from the user interface

In the menu bar at the bottom of the screen, click **Documentation**. In the tree for the desired company (tenant), select "IT Documentation", click "New" and select "Framework". Enter a name for your new framework and click "Save" above the tree.

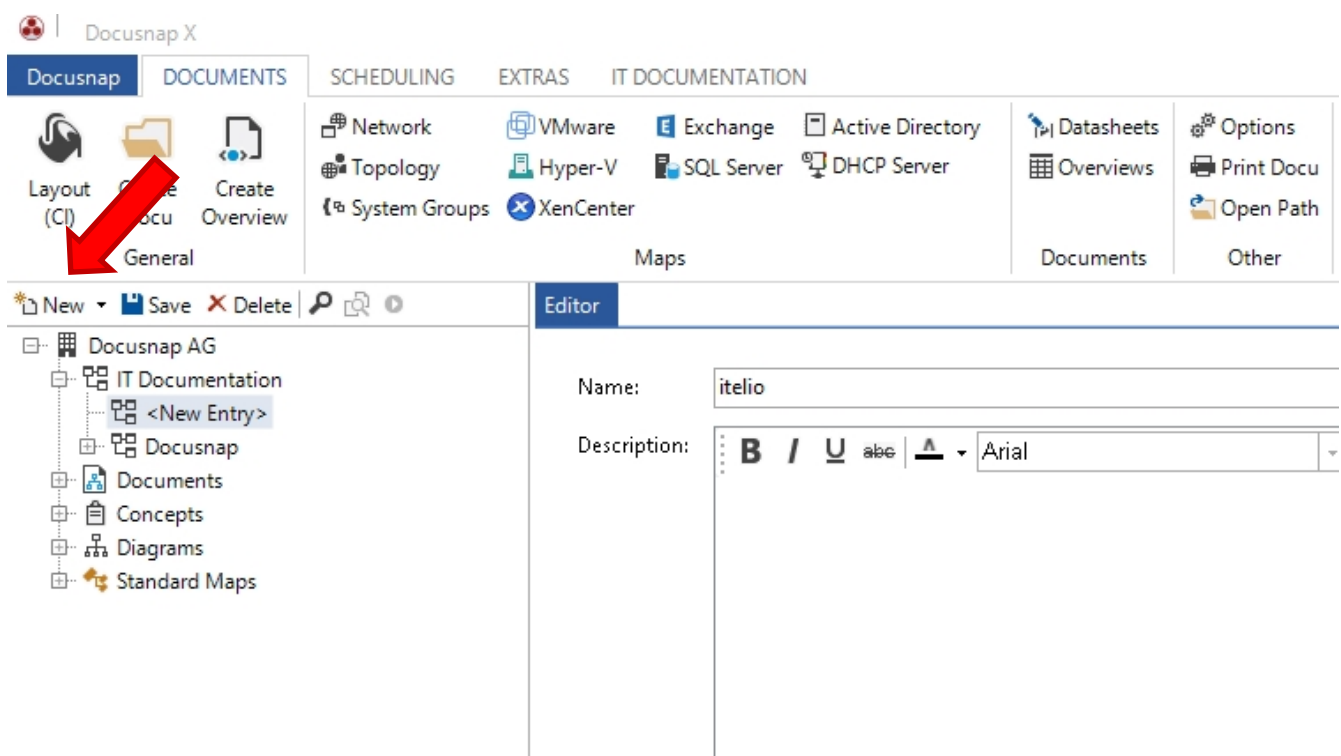


Figure 2 – Creating a framework from the Docusnap UI

Creating the framework using the Editor

As an alternative, you can create all objects in the Editor right from the start.

To do so, click “Documentation” in the menu bar at the bottom of the screen and then click **Editor** in the ribbon. This opens a new window where you can create a new framework by selecting **Add Framework**.

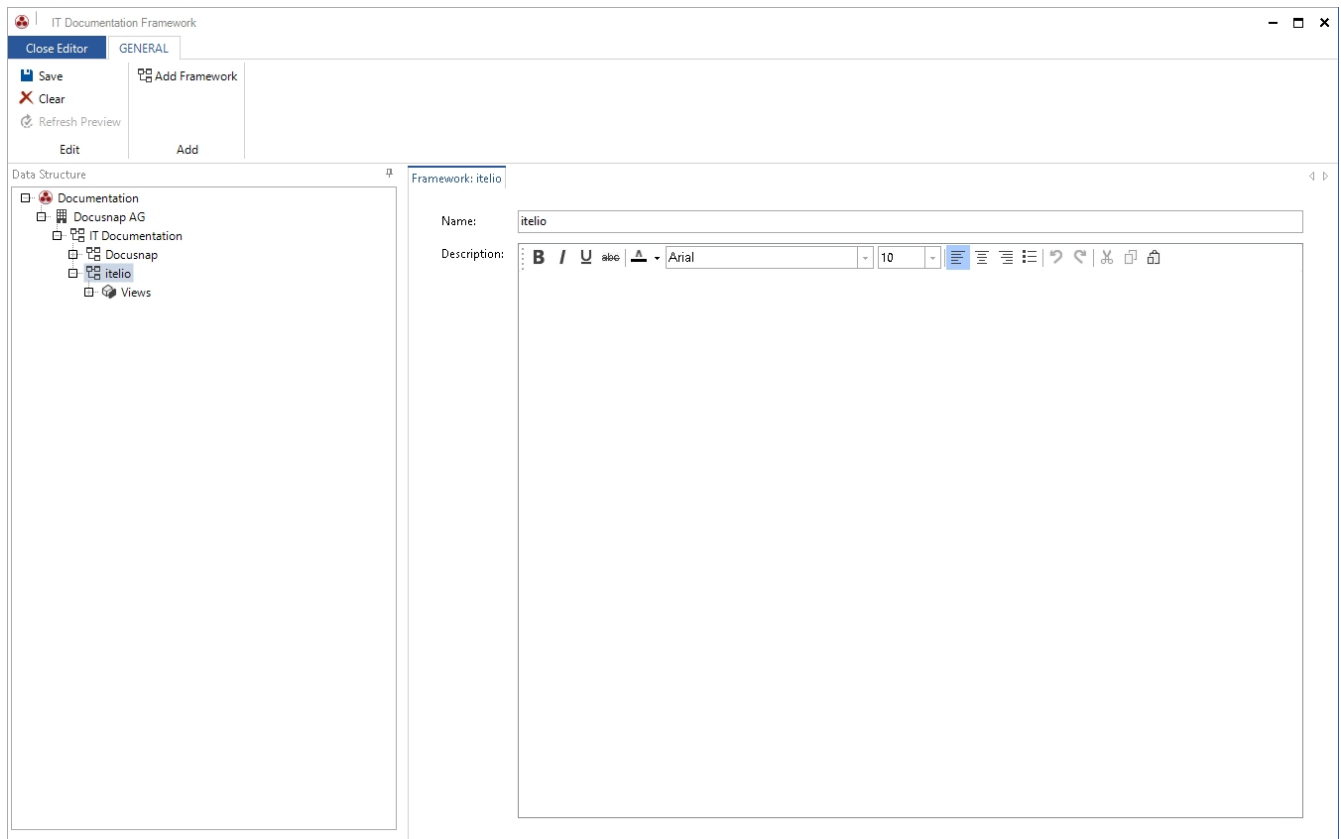


Figure 3 – Creating a framework using the Editor

2.2 Creating Views

To create views, you can also either use the UI or the Editor.

In the Editor, select “Views” from the tree and then click **Add View** in the ribbon. Enter a name for your new view and click “Save” in the ribbon.

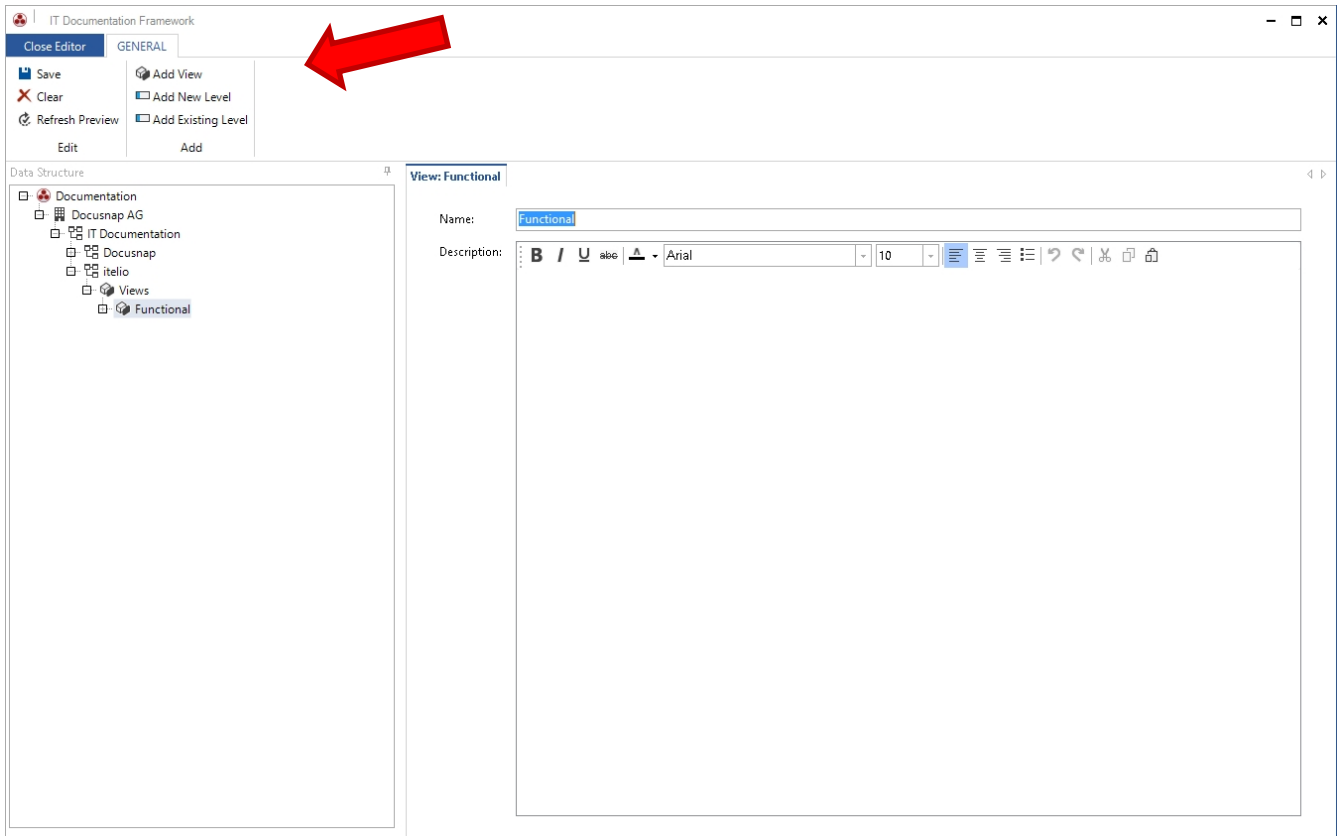


Figure 4 – Creating a view

2.3 Creating Levels

In the Editor, click the previously created view from the tree and then select **Add New Level** in the ribbon. Create all required levels. You can specify the order in which the levels will be displayed by specifying a value in the "Sorting" field for each level. Start with the value 1.

Tip: Enter increments of 10 instead of 1 for sorting. This makes re-sorting easier if multiple levels have been created.

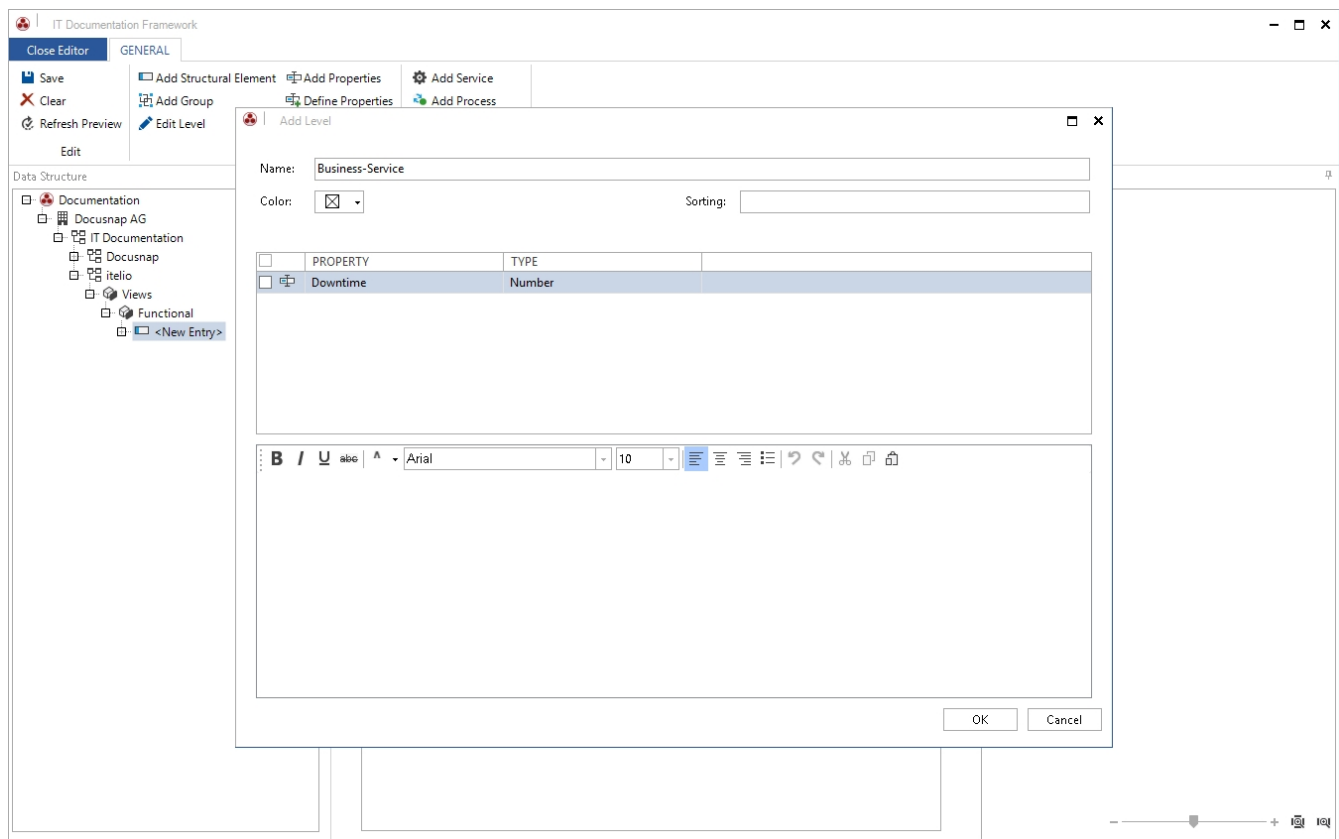


Figure 5 – Creating a new level

When you create another level, the **Visualization** preview window (shown in Figure 6 at the right) shows you what the view will look like. Here, you can check if the sorting order is of the levels correct.

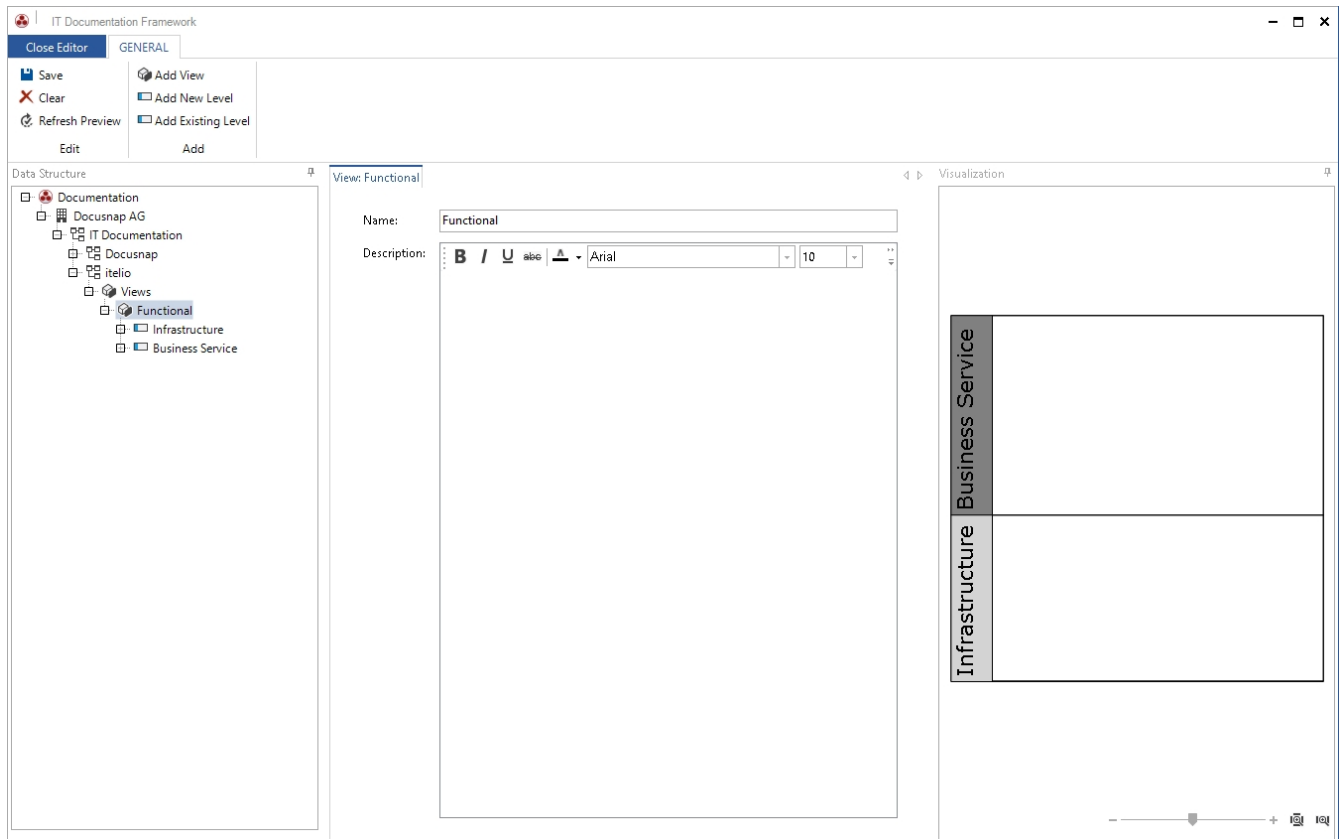


Figure 6 – Adding more levels

2.4 Adding Elements

Click **Add Elements** to open a window where you can select previously created elements from the inventory tree. These elements will be added to the current level when you click "Save". If you want to assign these elements to another level later, they can be moved to the desired location.

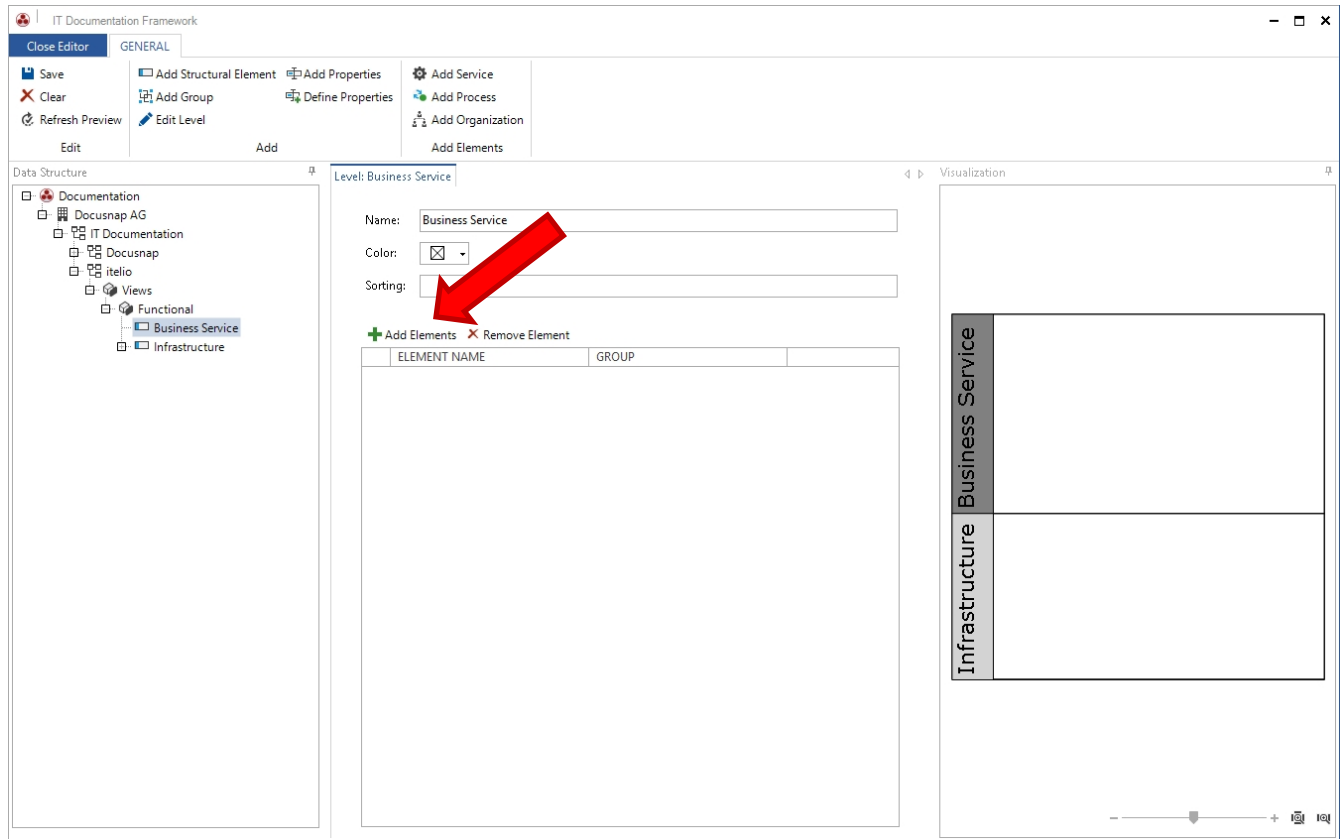


Figure 7 – Adding an element

Once you have added an element to a level, the preview is updated accordingly so that you can see what your documentation will look like.

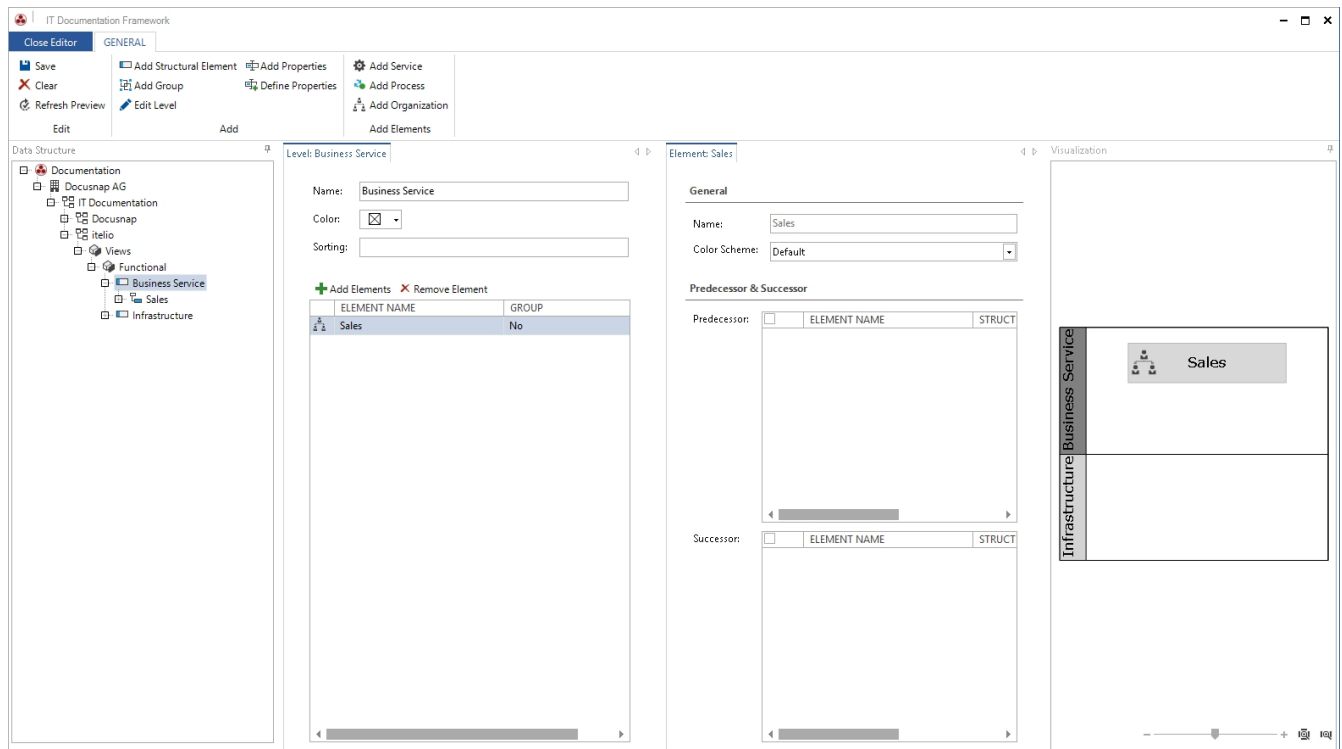


Figure 8 – Preview with the element you created

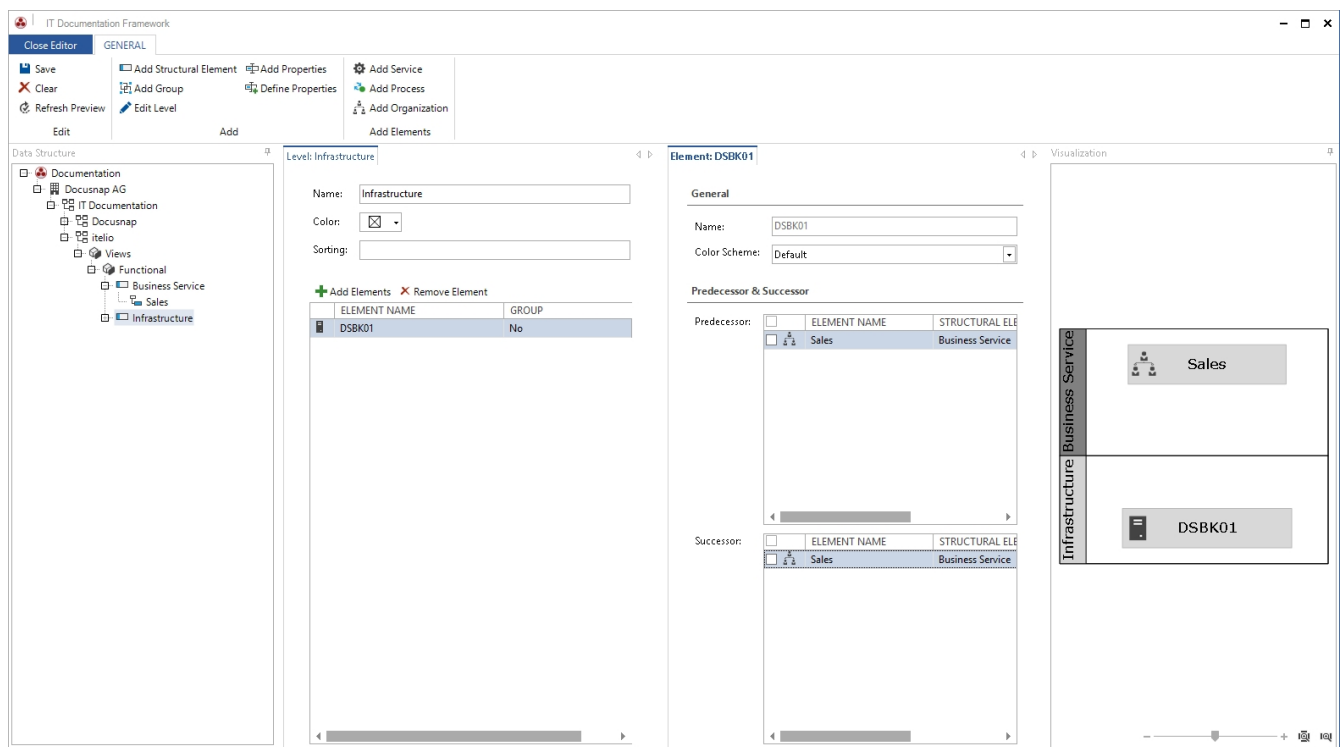


Figure 9 – Another element has been added

2.4.1 Linking elements

Now link the elements in order to show the dependencies between them. To do so, click the desired element and select its predecessor or successor in the "Predecessor & Successor" panel. Depending on whether you select a predecessor or successor for an element, the direction of the linking arrow will change.

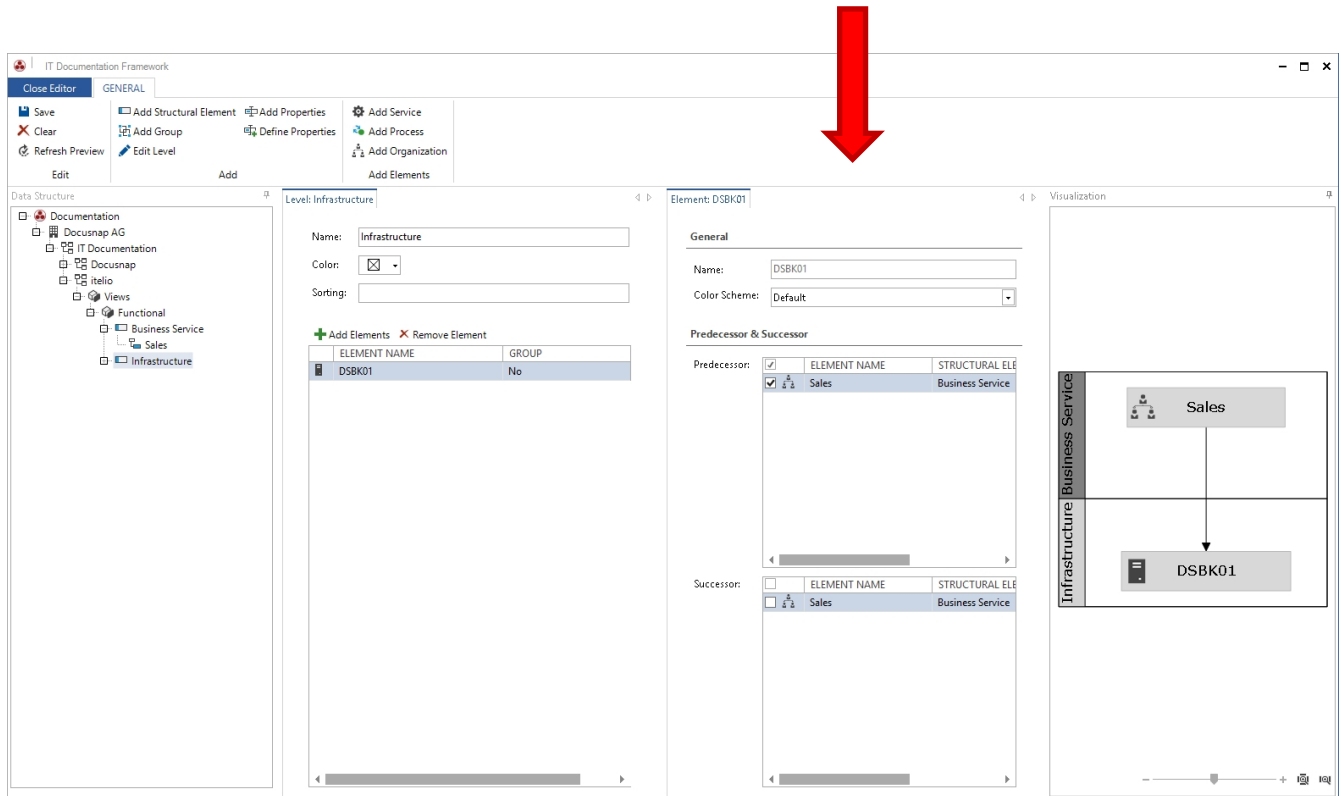


Figure 10 – Linking elements

2.4.2 Moving elements

If you later find that an element should be located on a different level, you can reassign it using the **Move Element** option.

To do so, select the element to be moved (Services, Processes, or Organization) and then click **Move Element** in the ribbon. The preview updates immediately to reflect the changed assignment.

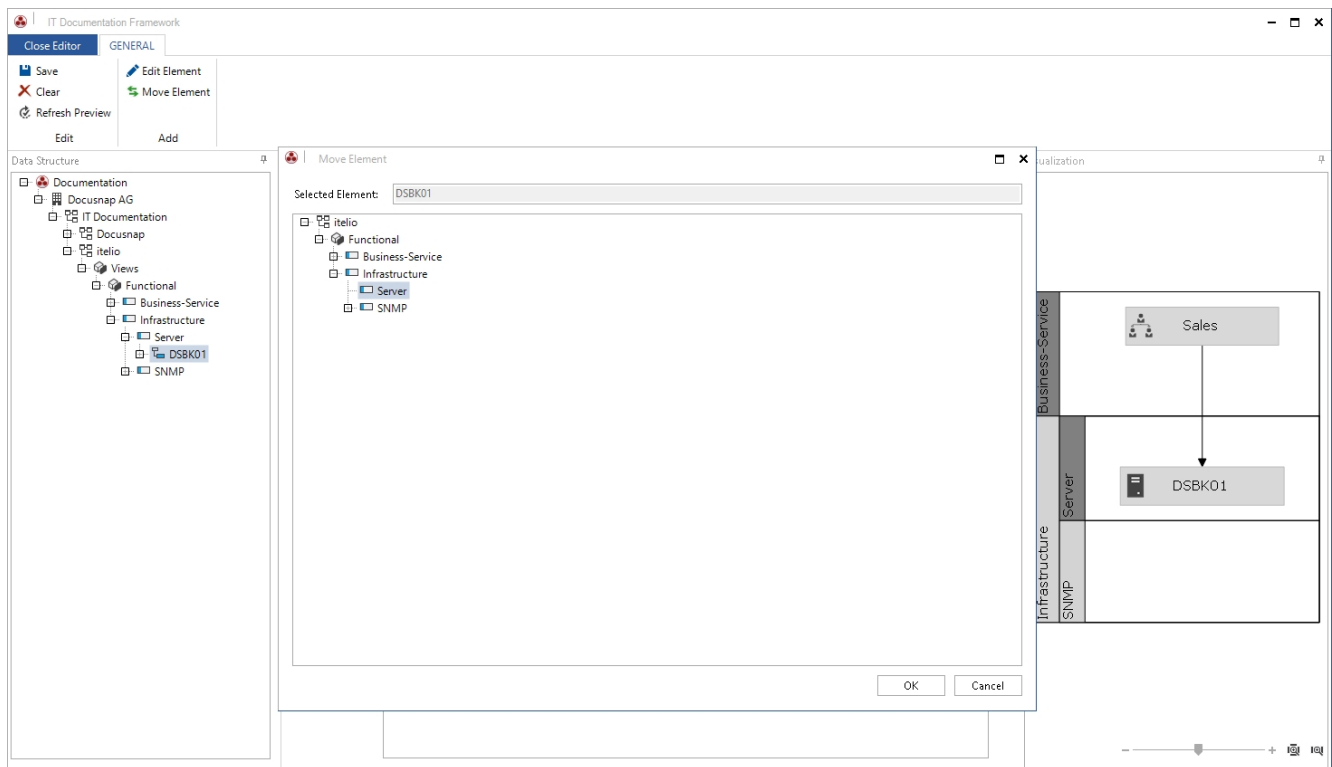


Figure 11 – Moving an element

3. Advanced Features

3.1 Adding Structural Elements

Select the level you want to subdivide, and click **Add Structural Element** in the ribbon.

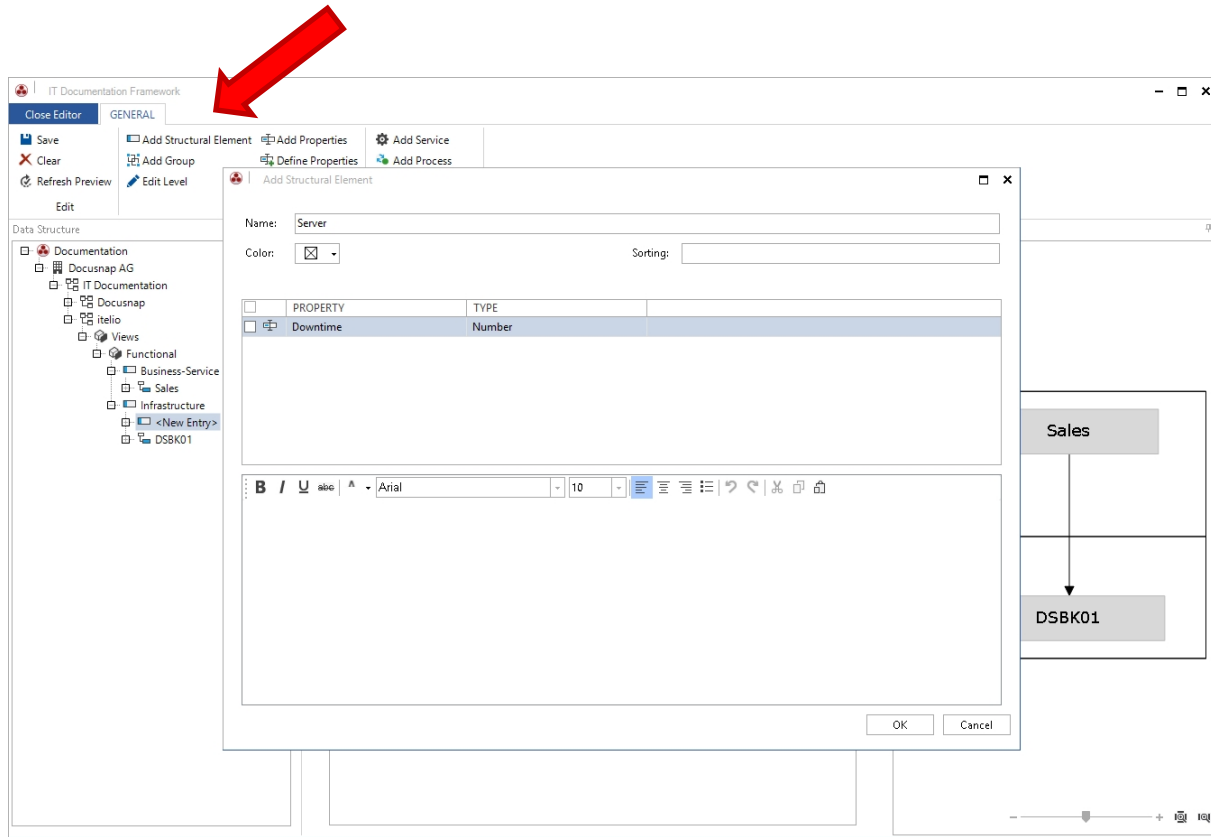


Figure 12 – Adding a structural element

Here again, the preview window will immediately reflect the impact of the change on the documentation (see figure 13). As you can see, the structural elements Server and SNMP were added to the IT Infrastructure level.

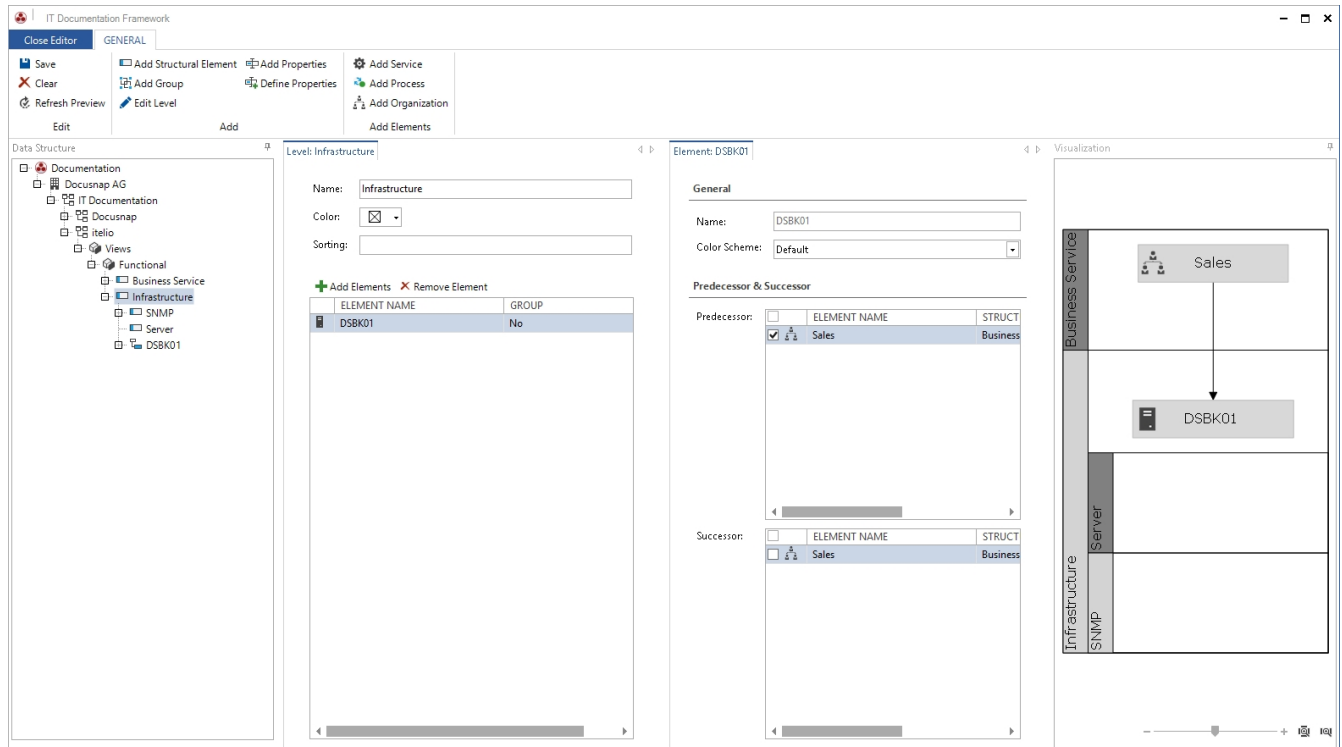


Figure 13 – Representation of structural elements

3.2 Creating Groups

Select a view from the tree and then click **Add Group** in the ribbon. A new window opens where you can enter a name and description for the group.

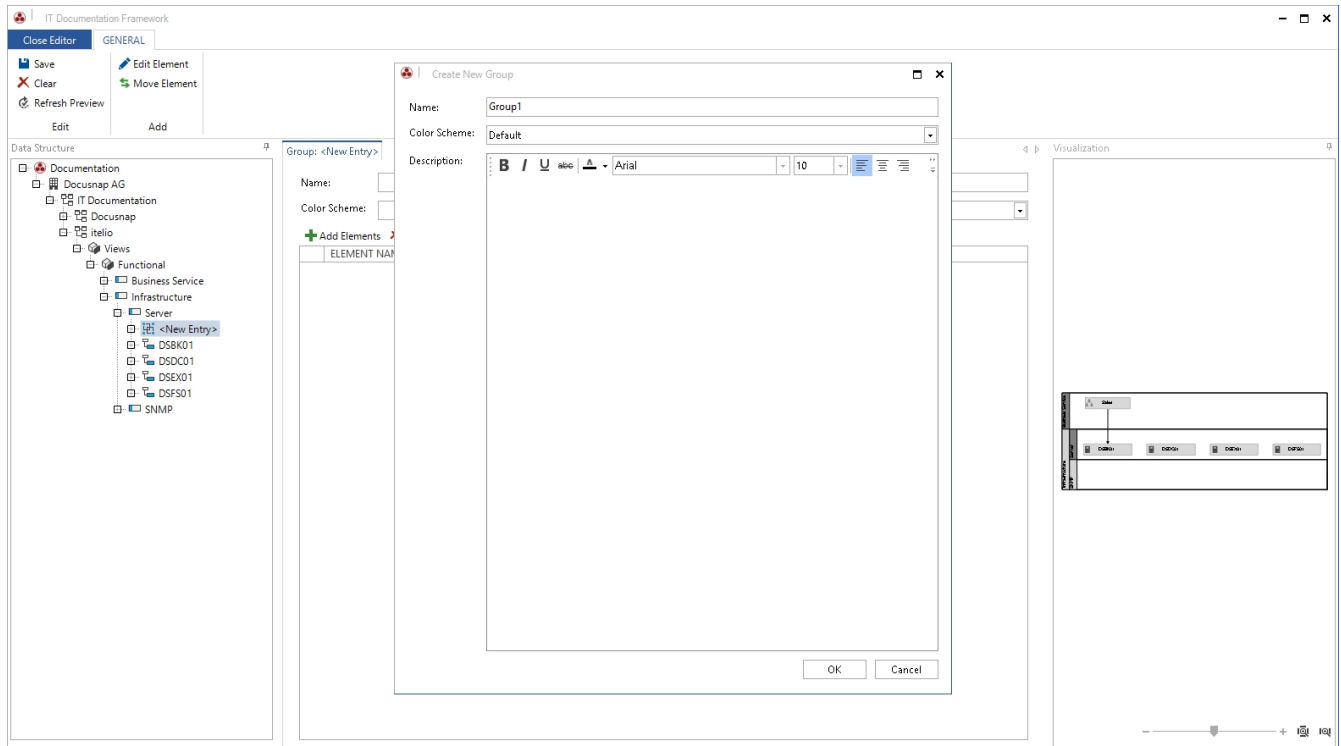


Figure 14 – Creating a group

3.2.1 Assigning elements to a group

Once you have created the group, you can assign elements to it. To do so, select the group from the tree and then click **Add Elements**. This opens a selection window where you can select the desired elements to be added to the group.

Click "Save". The preview window will immediately reflect the impact of the change on the documentation.

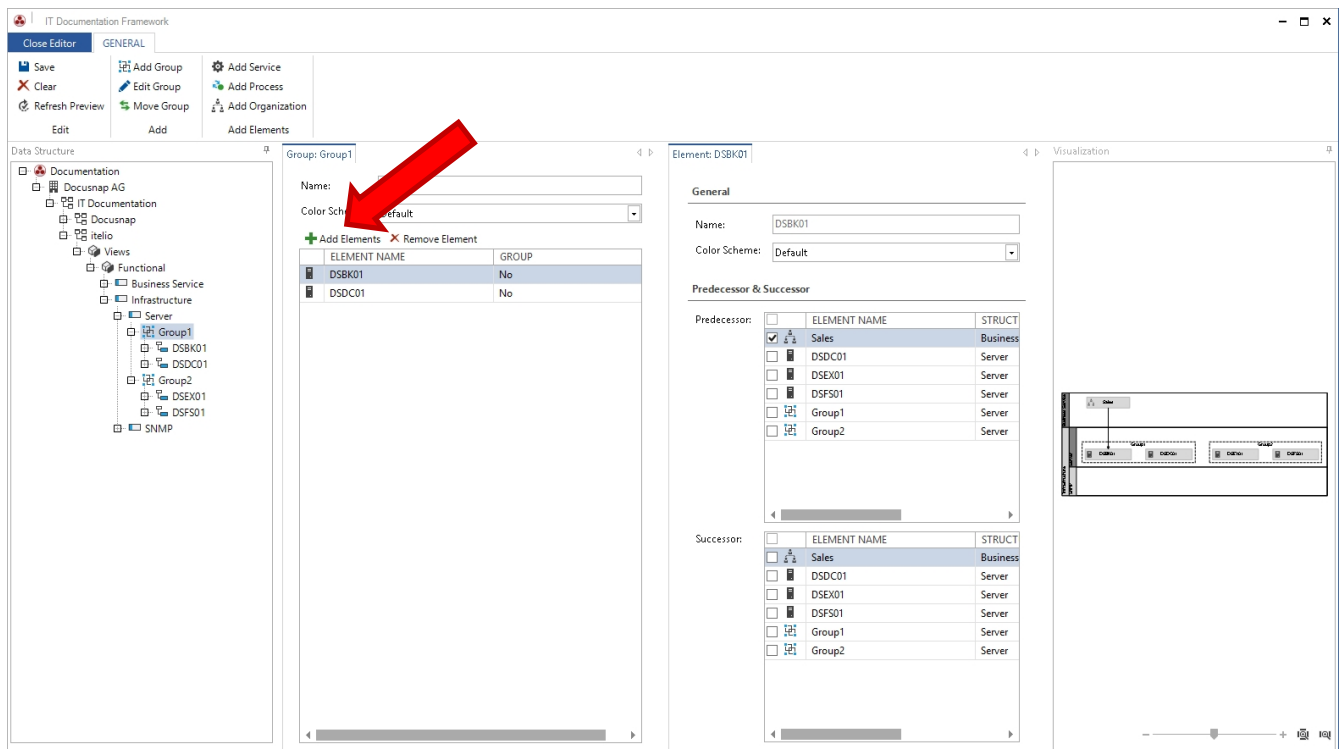
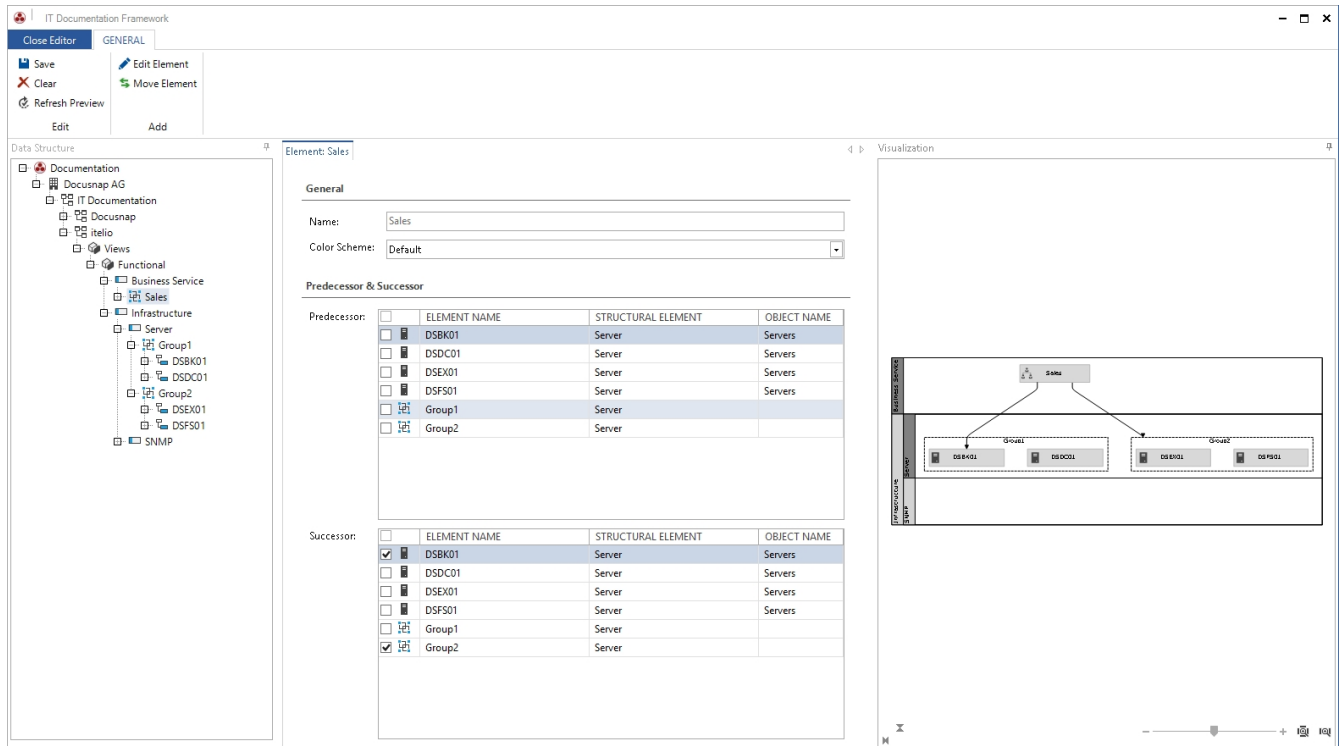


Figure 15 – Adding elements to the group

3.2.2 Linking elements to groups

You can specify the dependencies between individual elements or between an element and a group. The effect is shown in figure 16. At the left, the arrow points to an element, at the right, it points to the group.



IT Documentation Framework

Close Editor | GENERAL

Save | Edit Element
Clear | Move Element
Refresh Preview
Edit | Add

Data Structure

- Documentation
 - DocuSnap AG
 - IT Documentation
 - DocuSnap
 - itelio
 - Views
 - Functional
 - Business Service
 - Sales
 - Infrastructure
 - Server
 - Group1
 - DSBK01
 - DSDC01
 - DSEX01
 - DSFS01
 - Group2
 - DSEX01
 - DSFS01

Element: Sales

General

Name: Sales

Color Scheme: Default

Predecessor & Successor

Predecessor:

| <input type="checkbox"/> | ELEMENT NAME | STRUCTURAL ELEMENT | OBJECT NAME |
|--------------------------|--------------|--------------------|-------------|
| <input type="checkbox"/> | DSBK01 | Server | Servers |
| <input type="checkbox"/> | DSDC01 | Server | Servers |
| <input type="checkbox"/> | DSEX01 | Server | Servers |
| <input type="checkbox"/> | DSFS01 | Server | Servers |
| <input type="checkbox"/> | Group1 | Server | Servers |
| <input type="checkbox"/> | Group2 | Server | |

Successor:

| <input type="checkbox"/> | ELEMENT NAME | STRUCTURAL ELEMENT | OBJECT NAME |
|-------------------------------------|--------------|--------------------|-------------|
| <input checked="" type="checkbox"/> | DSBK01 | Server | Servers |
| <input type="checkbox"/> | DSDC01 | Server | Servers |
| <input type="checkbox"/> | DSEX01 | Server | Servers |
| <input type="checkbox"/> | DSFS01 | Server | Servers |
| <input type="checkbox"/> | Group1 | Server | |
| <input checked="" type="checkbox"/> | Group2 | Server | |

Visualization

Diagram showing the hierarchy of the 'Sales' element, with arrows pointing to 'Group1' and 'Group2'.

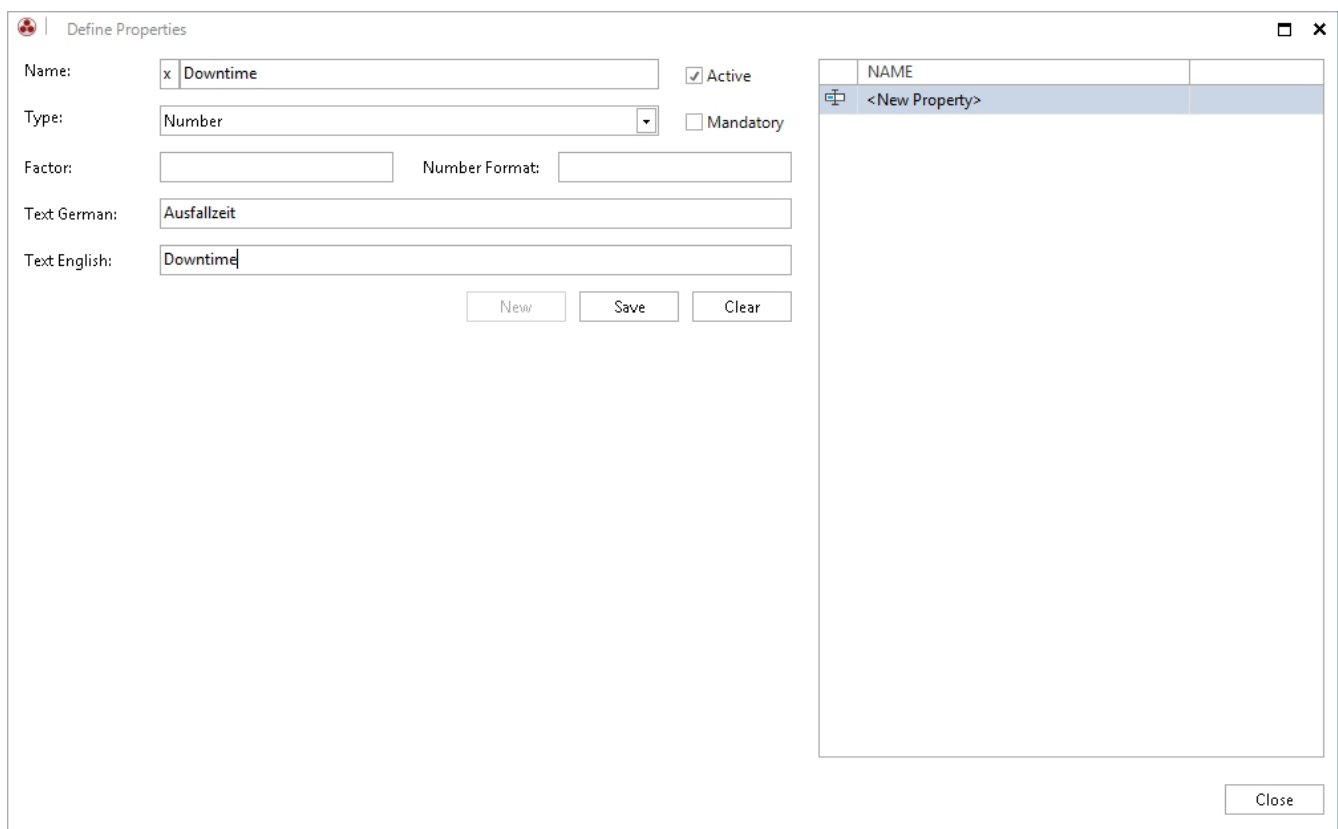
Figure 16 – Linking objects to a group

3.3 Properties

You can define properties for levels and structural elements. Thus, it is possible to store different pieces of information for these objects. In the next section, the “Downtime” property is defined. This property can then be assigned to every server element.

3.3.1 Defining properties

Select a level or structural element from the tree and then click **Define properties** in the ribbon. A new window opens as shown in figure 17. Here, enter a name for the new property, select the desired field type, and enter text into the “Text German” and “Text English” fields (if you do not speak German, you can repeat the English label in the “Text German” field). In addition, you can specify whether this field is mandatory or not.



| NAME |
|----------------|
| <New Property> |

Figure 17 – Defining the “Downtime” property

3.3.2 Assigning a property to a level

Once a property has been defined, you can assign it to a level or structural element. This assigns the property to all elements associated with the level or structural element.

Figure 18 below shows how to assign the “Downtime” property to the Server structural element on the “Infrastructure” level.

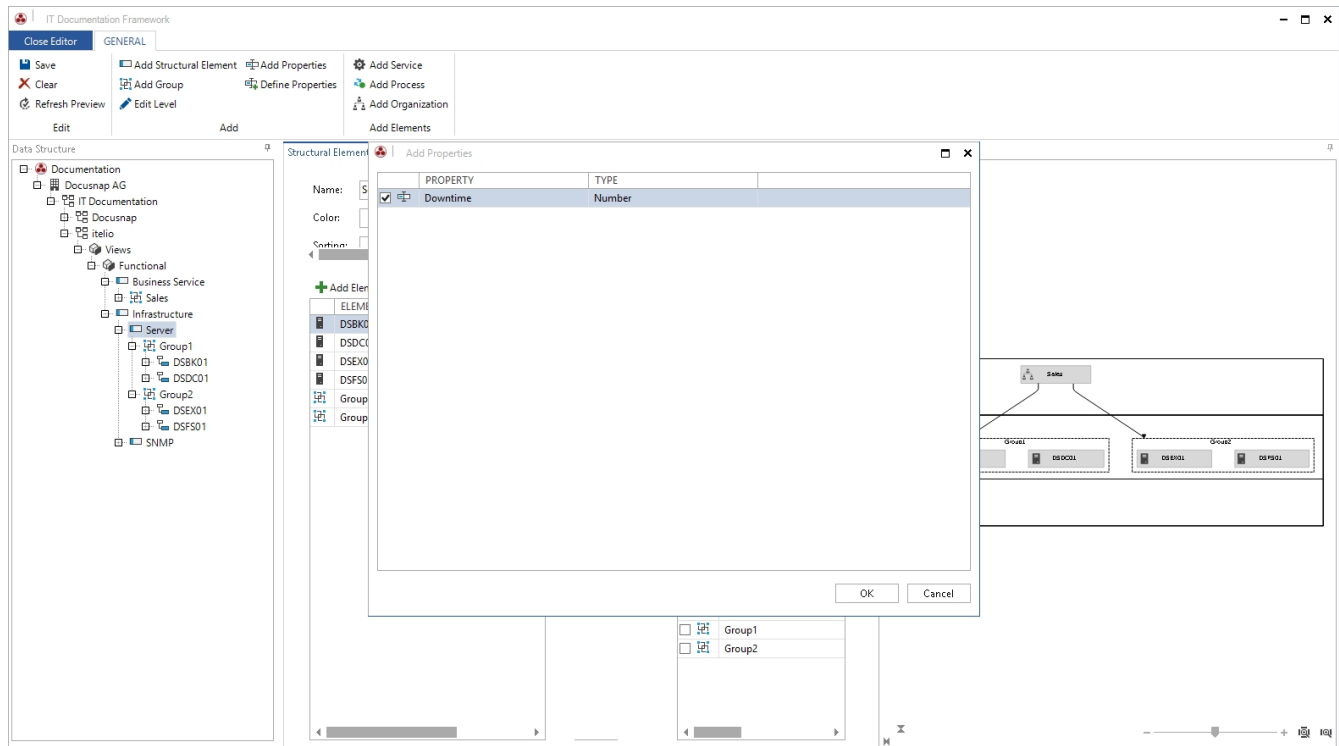


Figure 18 – Assigning a property

3.3.3 Assigning values to properties

Now, you can specify a downtime value for each element on this level.

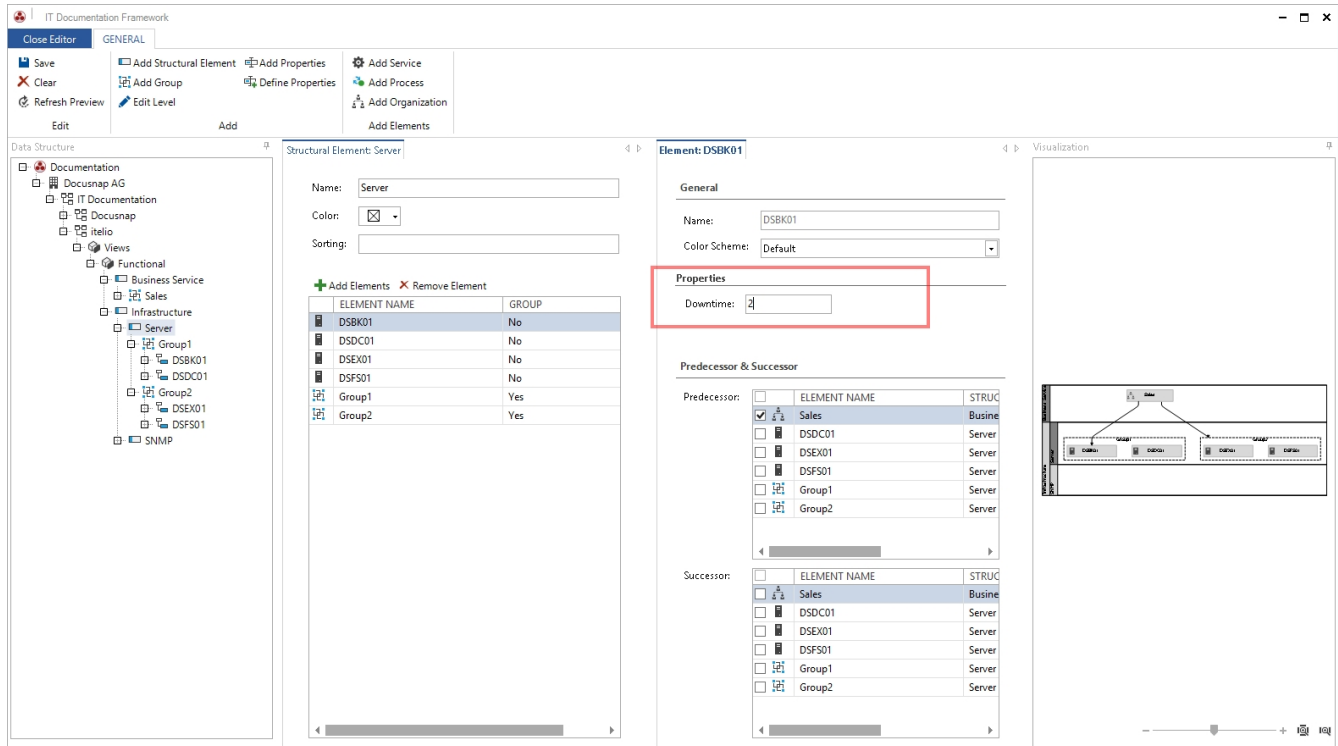


Figure 19 – Specifying a value for a property

3.4 Exporting a Framework

You can export the framework created in Docusnap and import it to other Docusnap installations. To do so, click **Export Structure** in the ribbon of the main window and then select the desired framework.

3.5 Useful Tip: Customizing the Screen Display

Working in the Editor is easier if you use two monitors: If you move the preview window to the second monitor while working in the main window, it is possible to see all required data and the preview at the same time.

Simply click the title bar of the Visualization window, press and hold the left mouse button and drag the mouse to the side. You can arrange the preview window at one of the predefined locations or display it as a separate full-screen window. This window can be moved to the second monitor. Please note that this is also possible for many other window panels in Docusnap.

LIST OF FIGURES

| | |
|-----------------------------------------------------------|----|
| FIGURE 1 – SAMPLE IT DOCUMENTATION..... | 6 |
| FIGURE 2 – CREATING A FRAMEWORK FROM THE DOCUSNAP UI..... | 9 |
| FIGURE 3 – CREATING A FRAMEWORK USING THE EDITOR..... | 10 |
| FIGURE 4 – CREATING A VIEW | 11 |
| FIGURE 5 – CREATING A NEW LEVEL | 12 |
| FIGURE 6 – ADDING MORE LEVELS | 13 |
| FIGURE 7 – ADDING AN ELEMENT | 14 |
| FIGURE 8 – PREVIEW WITH THE ELEMENT YOU CREATED | 15 |
| FIGURE 9 – ANOTHER ELEMENT HAS BEEN ADDED | 15 |
| FIGURE 10 – LINKING ELEMENTS | 16 |
| FIGURE 11 – MOVING AN ELEMENT..... | 17 |
| FIGURE 12 – ADDING A STRUCTURAL ELEMENT | 18 |
| FIGURE 13 – REPRESENTATION OF STRUCTURAL ELEMENTS..... | 19 |
| FIGURE 14 – CREATING A GROUP | 20 |
| FIGURE 15 – ADDING ELEMENTS TO THE GROUP | 21 |
| FIGURE 16 – LINKING OBJECTS TO A GROUP..... | 22 |
| FIGURE 17 – DEFINING THE “DOWNTIME” PROPERTY | 23 |
| FIGURE 18 – ASSIGNING A PROPERTY | 24 |
| FIGURE 19 – SPECIFYING A VALUE FOR A PROPERTY | 25 |

VERSION HISTORY

| Date | Description |
|------|------------------|
| 1.0 | Document created |



Docusnap[®]

support@docusnap.com | www.docusnap.com/support
© itelio GmbH - www.itelio.com