



**Application about  
Human Machine Interface**

**applications  
& TOOLS**

WinCC flexible 2005 Recipes

**SIEMENS**



Application Description

**Note**

The Application Examples are not binding and do not claim to be complete regarding the circuits shown, equipping and any eventuality. The Application Examples do not represent customer-specific solutions. They are only intended to provide support for typical applications. You are responsible for ensuring that the described products are correctly used. These Application Examples do not relieve you of the responsibility of safely and professionally using, installing, operating and servicing equipment. When using these Application Examples, you recognize that Siemens cannot be made liable for any damage/claims beyond the liability clause described. We reserve the right to make changes to these Application Examples at any time without prior notice. If there are any deviations between the recommendations provided in these Application Examples and other Siemens publications – e.g. Catalogs – then the contents of the other documents have priority.

## Warranty, liability and support

We do not accept any liability for the information contained in this document.

Any claims against us – based on whatever legal reason – resulting from the use of the examples, information, programs, engineering and performance data etc., described in this Application Example shall be excluded. Such an exclusion shall not apply in the case of mandatory liability, e.g. under the German Product Liability Act (“Produkthaftungsgesetz”), in case of intent, gross negligence, or injury of life, body or health, guarantee for the quality of a product, fraudulent concealment of a deficiency or breach of a condition which goes to the root of the contract (“wesentliche Vertragspflichten”). However, claims arising from a breach of a condition which goes to the root of the contract shall be limited to the foreseeable damage which is intrinsic to the contract, unless caused by intent or gross negligence or based on mandatory liability for injury of life, body or health. The above provisions do not imply a change in the burden of proof to your detriment.

**Copyright© 2006 Siemens A&D. It is not permissible to transfer or copy these Application Examples or excerpts of them without first having prior authorization from Siemens A&D in writing.**

For questions about this document please use the following e-mail address:

<mailto:csweb@ad.siemens.de>

## Foreword

### Objective of the application

This application was created to...

- provide the user with information on “engineering with recipes”
- to show the different setting options for the recipe parameters and how they affect the running process.

In this application, the different requirements for the operator guidance of recipes are implemented using a SIMATIC controller and program examples.

### Main contents of this application

This application deals with the following key elements:

- Application of the standard recipe display
- Use of self-created displays for the parameterization of recipes
- Import and export of recipes
- Data backup of recipes
- Documentation of parameter changes with the aid of audit trail

### Delimitation

This application does not include a description of...

- the SIMATIC STEP 7 engineering tool.
- the used multi panel.

Basic knowledge of these topics is required.

## Structure of the document

The documentation of this application is divided into the following main parts.

Part	Description
Application Description	You are provided with a general overview of the contents. You are informed on the used components (standard hardware and software components and the created user software).
Principles of Operation and Program Structures	This part describes the detailed functional sequences of the involved hardware and software components, the solution structures and – where useful – the specific implementation of this application. It is only required to read this part if you want to familiarize with the interaction of the solution components to use these components, e.g. as a basis for own developments.
Structure, Configuration and Operation of the Application	This part takes you step by step through structure, important configuration steps, startup and operation of the application.
Appendix	The appendix provides further information such as bibliographic references, etc.

## Reference to Automation and Drives Service & Support

This entry is from the internet application portal of Automation and Drives Service & Support. Clicking the link below directly displays the download page of this document.

<http://support.automation.siemens.com/WW/view/en/23901413>

## Table of Contents

<b>Application Description .....</b>	<b>7</b>
<b>1 Automation Problem .....</b>	<b>7</b>
1.1 Overview .....	7
1.2 Requirements .....	9
<b>2 Automation Solution .....</b>	<b>10</b>
2.1 Overview of the overall solution .....	10
2.2 Description of the core functionality .....	12
2.2.1 Overview and description of the user interface .....	12
2.2.2 Partition 1, Samples of planned recipes .....	14
2.2.3 Partition 2, Import – Export Recipes .....	28
2.2.4 Partition 3, Backup of Recipes .....	33
2.2.5 Partition 4, Sample with an FDA practice .....	35
2.3 Required hardware and software components .....	43
2.4 Performance data .....	45
<b>Principles of Operation and Program Structures .....</b>	<b>46</b>
<b>3 General Functional Mechanisms .....</b>	<b>46</b>
3.1 Basics on “recipes” .....	46
<b>4 Explanations of the Example Program .....</b>	<b>49</b>
4.1 Structure of the WinCC flexible project .....	49
4.2 Configuration explanation for “Sample_1_1” .....	50
4.3 Configuration explanations “Sample_1_2 to Sample_1_5” .....	52
4.4 Configuration explanation for “Sample_1_6” .....	56
4.5 Configuration explanation for “Sample_2_1” .....	59
4.6 Configuration explanation for “Sample_2_2” .....	63
4.7 Configuration explanation for “Sample_3_1” .....	65
4.8 Configuration explanation for “Sample_4_1” .....	67
4.9 Structure of the STEP 7 project .....	76
<b>5 Modifications to the Example Program .....</b>	<b>79</b>
5.1 Adapting the storage path for the recipes / audit trail file .....	79
<b>Structure, Configuration and Operation of the Application .....</b>	<b>80</b>
<b>6 Installation and Startup .....</b>	<b>80</b>
6.1 Installation of hardware and software .....	80
6.2 Installation of the application software .....	81
<b>7 Operation of the Application .....</b>	<b>82</b>
<b>Appendix and Literature .....</b>	<b>86</b>
<b>8 Additional Notes .....</b>	<b>86</b>

---

Application Recipes	ID Number: 23901413
8.1	Import / export of recipes ..... 86
8.2	Evaluating audit trails / integrity check..... 86
8.3	Showing / hiding objects via the “layers” ..... 87
<b>9</b>	<b>Literature ..... 88</b>
9.1	Bibliographic references ..... 88
9.2	Internet links ..... 89
<b>10</b>	<b>History ..... 90</b>

## Application Description

### Content

You are informed on the used components (software components and the created user software).

The displayed performance data illustrate the performance capability of this application.

## 1 Automation Problem

### You are provided with information on...

the specific automation problem discussed in this documentation.

### 1.1 Overview

#### Introduction

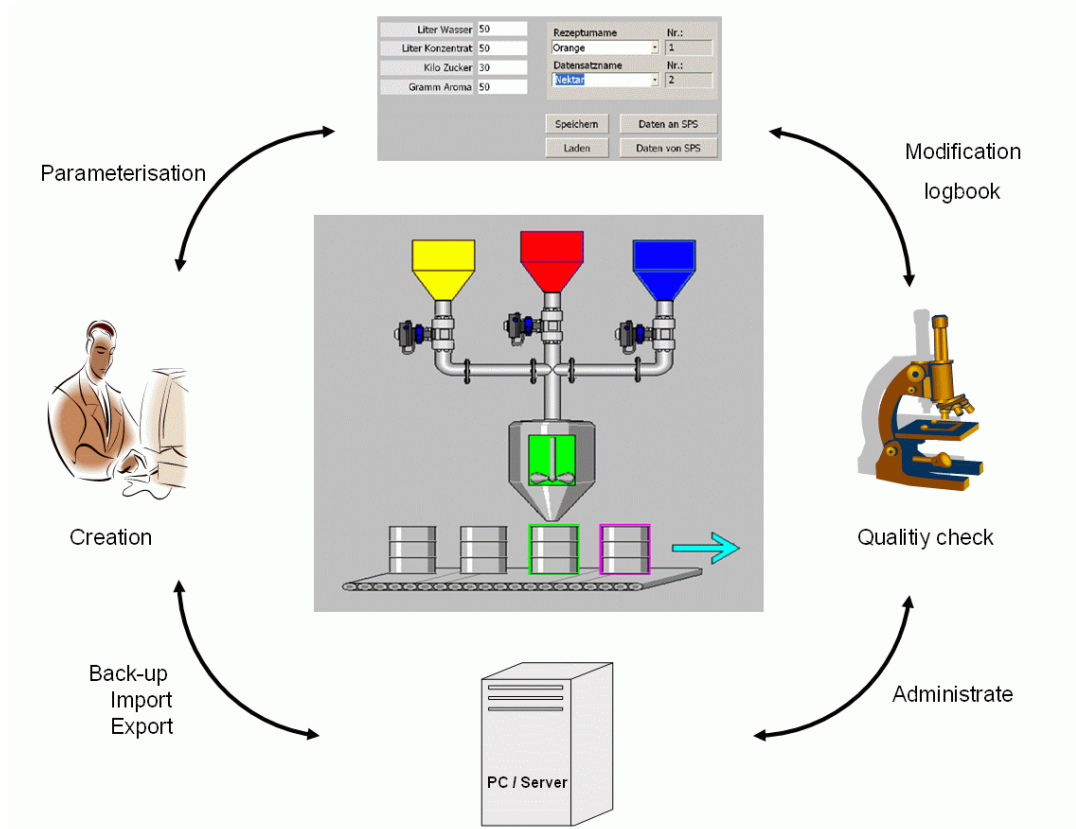
Your customer has different requirements for the operator guidance of recipes.

- Operation is to be possible via the standard recipe display and self-created image objects.
- Parameter changes are to be documented.
- Daily data backup of all or individual recipes and data records on a separate memory card or on a PC.

## Overview of the automation problem

The figure below provides an overview of the automation problem.

Figure 1-1



Copyright © Siemens AG 2006 All rights reserved  
23901413\_MinCC\_flexible\_Rezepturen\_e.doc

## Description of the automation problem

In a subprocess in the dyeing industry colors are mixed from different primary colors.

These individual colors are transported to a collecting box after an individual preprocessing. From there the color is barreled.

The individual color compositions are selected via a recipe display which includes all parameters.

If a recipe or parameter is changed, this change will be documented in a modification logbook (audit trail file). This enables to subsequently determine which person processing made this change and possible "wrong fillings" can be verified. This ensures that a quality inspection of the generated products is achieved which is as integrated as possible.

Different production requirements make it necessary that all recipes (color compositions) are centrally stored on a separate storage medium or PC / server and that they can be called individually.

New color compositions are mixed in the production development and integrated into the running process as and when required.



## 1.2 Requirements

This application is integrated into a STEP 7, it is to meet the requirements listed below:

### Controller requirements

According to the parameter sets, which are transferred to the controller via the HMI operator panel, the controller processes the values and transfers them to the corresponding valves, drives, etc.

### HMI requirements

- The HMI is to be realized by an MP370 Touch (screen diagonal 12 inches).
- The following operating options are to be available:
  - Option to change all recipe parameters
  - Writing the data records into the controller
  - Reading the data records from the controller
  - Exporting data records
  - Importing data records
  - Adding new data records
  - Deleting existing data records
  - Daily backup of the recipes
- Display of the current process progress (displaying the representation of the filling process).

## 2 Automation Solution

You are provided with information on...

the solution selected for the automation problem.

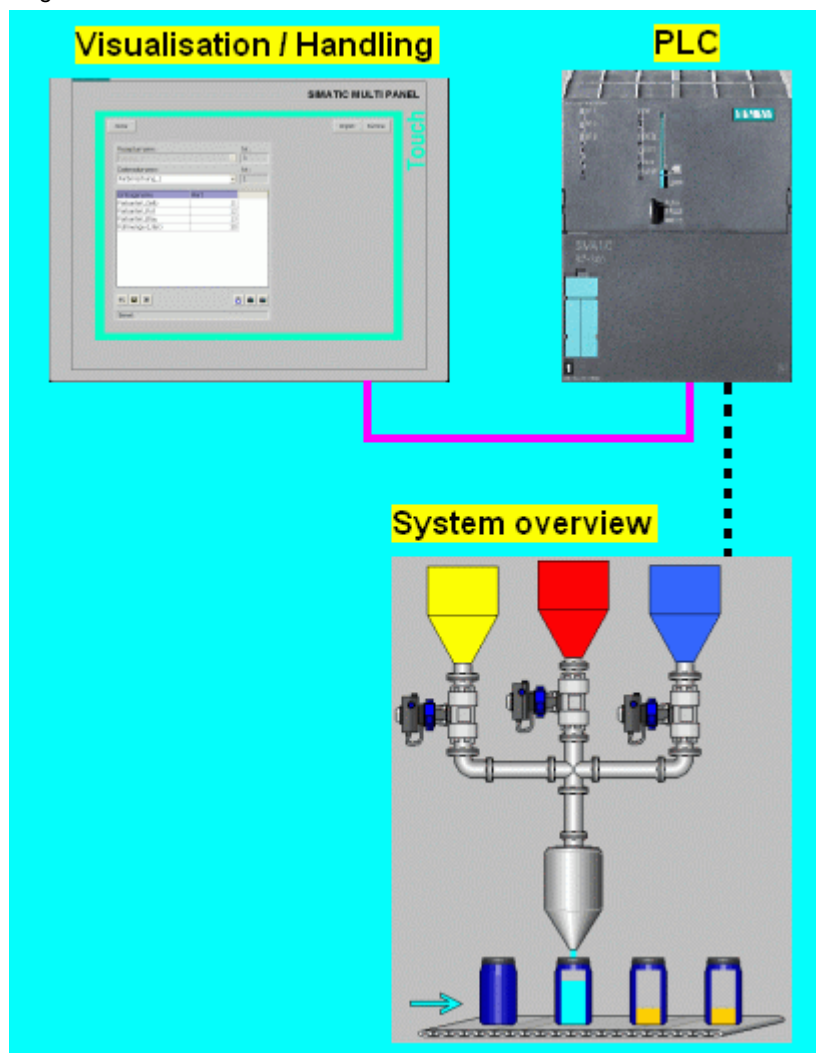
### 2.1 Overview of the overall solution

#### Schematic layout

WinCC flexible features integrated functionalities enabling to display parameters for different machine processes as recipes and data records and to log changes within these parameter sets on the Runtime system (panel / PC) (audit trail).

The following figure schematically shows the most important components of the solution:

Figure 2-1



## Structure

Different user examples of created recipes are the key element of this WinCC flexible application.

The different case studies were created with the aid of integrated WinCC flexible standard functions.

The configuration was divided into four partitions to illustrate the behavior of the configured recipes.

- Partition 1  
Samples of planned recipes
- Partition 2  
Import – Export Recipes
- Partition 3  
Backup of Recipes
- Section 4  
Sample with an FDA practice

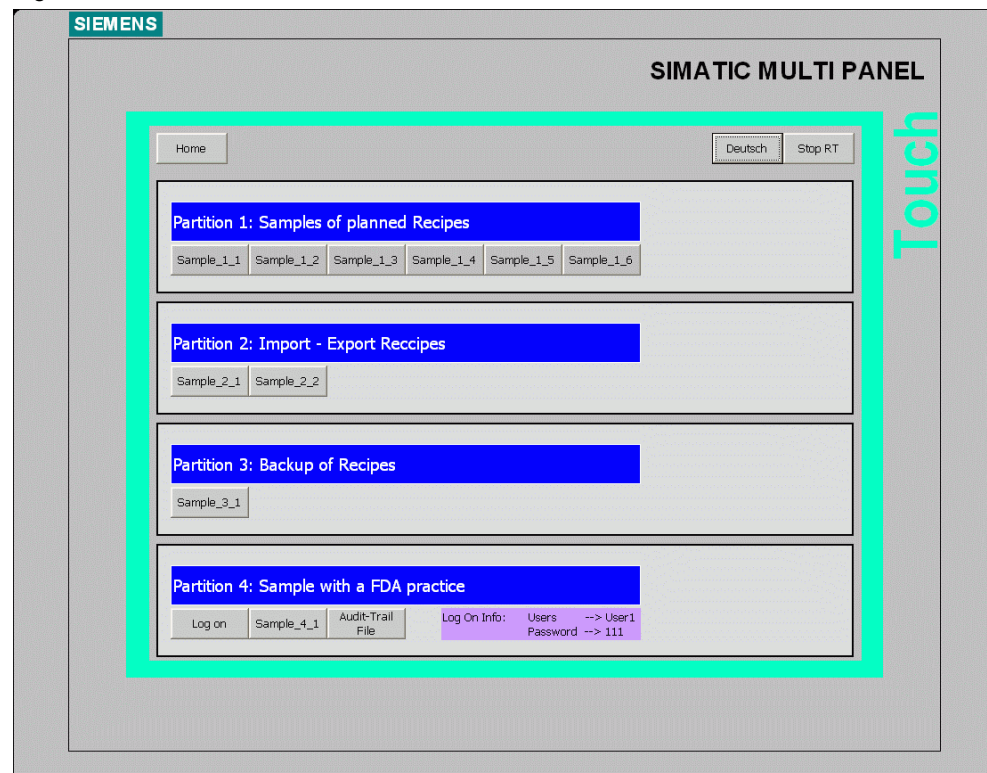
## 2.2 Description of the core functionality

This chapter provides you with a functional description of the individual configured screens. You will find a detailed configuration description in the following chapters.

### 2.2.1 Overview and description of the user interface

A 12-inch MP370 Touch 12 and a CPU 315-2 DP were used for the application. For better clarity, the configuration was divided into four partitions.

Figure 2-2



#### 1. Partition 1: Samples of planned recipes

- In partition 1, 6 examples are used to show the different parameterization and setting options you can use during the configuration phase in the properties of the respective recipe. The examples illustrate how the settings made affect the properties of the used recipe tags on the operator panel and in the controller.

#### 2. Partition 2: Import – Export Recipes

- In partition 2, two examples are used to show how you can import or export recipes and their data records.

3. Partition 3: Backup of Recipes
  - In partition 3, an example is used to explain how you can back up your recipe data.
4. Partition 4: Example application with the “Audit Trail” function
  - In partition 4, an example is used to show how you can use the “audit trail” function within a process and how it is displayed to the operator.

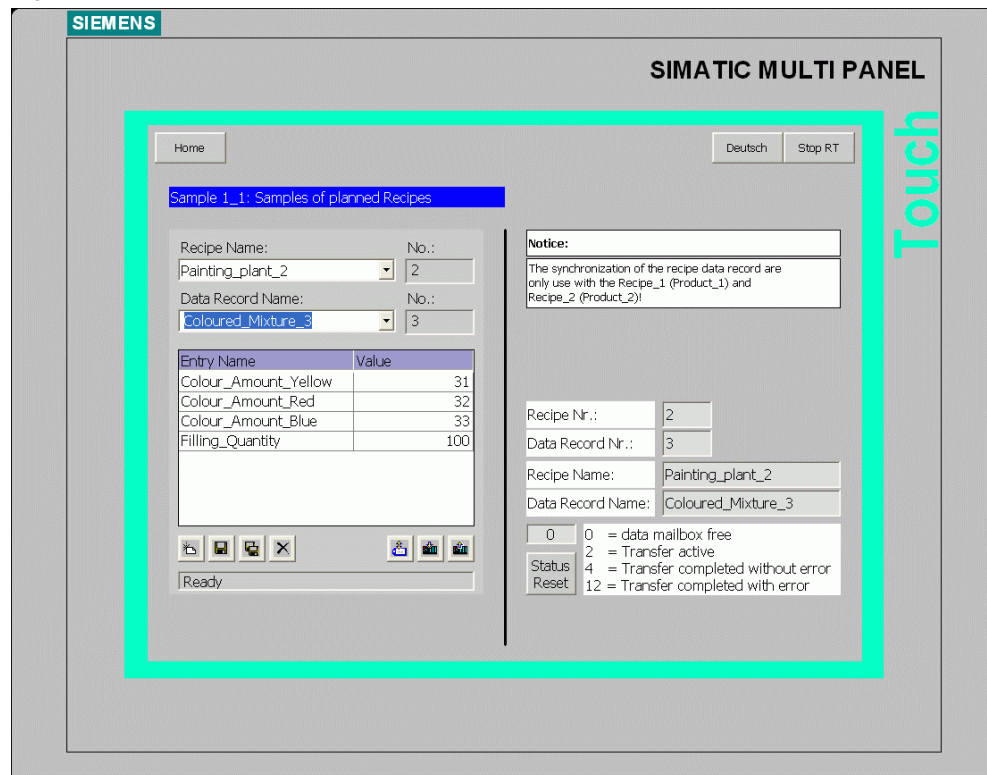
## 2.2.2 Partition 1, Samples of planned recipes

Samples of planned recipes

### Sample 1\_1






The overview display of sample 1\_1 is shown below



Figure 2-3



A standard recipe display in which different functions were implemented is located on the left side.

All "general commands / menu items" were used.

-  Add data record button
-  Save data record button
-  Save data record as... button
-  Delete data record button
-  Synchronize tags button

-  Set data record to PLC button
-  Get data record from PLC button

The individual functions will not be described in greater detail. For further information on these functions, please refer to the online help.

In addition, the parameters were defined in the configuration in such a way that each individual recipe can be selected by the operator.

**Note:**

For the recipes “Painting\_plant\_1” or “Painting\_plant\_2”, the parameters are transferred to the controller with “synchronization”.

These two recipes require that the “data mailbox” is reset after transferring a data record (the “Status Reset” button is used to reset the data mailbox via the controller).

Additionally configured functions are located on the right side.

- The current recipe number and the data record number are displayed in an external output box.
- An internal function (GetDataRecordName) is used to convert the recipe and data record numbers to a recipe name and a data record name.
- Status display  
The status display is located above the “Status Reset” button. The value of the status display indicates the current status. The explanation of the meaning of the individual values next to the display provides additional information.

**Note:**

The display is only relevant when transferring the two recipes “Painting\_plant\_1” and “Painting\_plant\_2” .

**Advantages of this solution**

- The operator can select each individual recipe.
- Additional data records can be added to the preset data records of each recipe.
- The internal function (GetDataRecordName) enables you to provide the operator with information on the currently selected recipe and data record without requiring the operator to change to the page with the configured recipe display.

## Note on the following samples 1\_2 to 1\_5

The structure of the examples “Sample 1\_2” to “Sample 1\_5” is identical.

The page is graphically divided into two parts.

A graphical representation of the used parameters defined in the properties of this recipe is located in the top left part.

Depending on these defined parameters,

- a reset of the data mailbox is necessary after transferring a data record (Synchronization).
- the behavior of the recipe tags is influenced if they are additionally used in another screen (Synchronize tag).
- the transfer of the recipe parameters from and to the controller differs (Tags offline).

A help button via which the corresponding parameter is described is located next to the graphically represented parameters.

### **Note:**

The properties of a recipe always have to be seen “as a whole”. This means that if a property is changed within a recipe, this may influence the other parameters.

A standard recipe display is located below this display.

The recipe was preset so that it cannot be modified / changed.

The right side “symbolically” displays an S7-CPU with the stored tags from the recipe display.

You are thus provided with a direct comparison of how the tags / values in the recipe display and in the controller respond in case of a value change or transfer (depending on the configured properties of this recipe).

The “External recipe tag” button is located below the displayed S7-CPU. Use this button to go to a page on which the used recipe tags were used again (external, outside the standard recipe display).

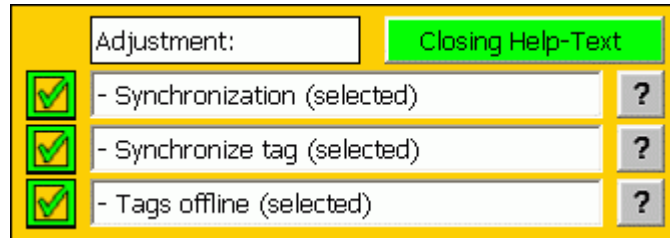
This enables you to monitor the behavior of the tags depending on the configured recipe properties.



## Sample 1\_2

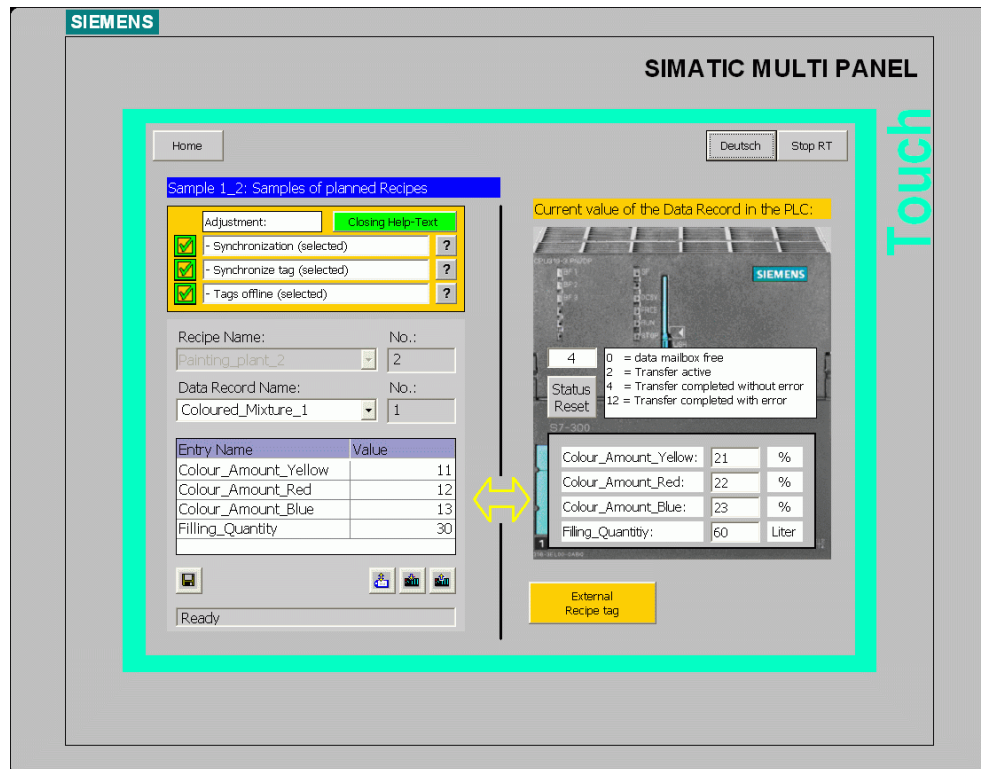
In the properties of the recipe, the following settings were defined.

Figure 2-4



The overview display of sample 1\_2 is shown below.

Figure 2-5



As indicated by the parameters, "Synchronization" is used to transfer the recipe. This requires that the data mailbox has to be reset after each transfer of a data record from or to the controller. The current status is additionally displayed in an output box.

## Advantages of this solution

- The advantage of the transfer with “Synchronization” is that you can avoid an uncontrolled mutual overwriting of the data in your control program.
- You can actively influence an enable for the transfer of data records via the controller, for example, by enabling / resetting the data mailbox only under specific conditions.
- You can trigger the transfer of data records using a system function or control command, e.g. with the system functions “SetDataRecordToPLC” and “GetDataRecordFromPLC” or the control commands “Write\_data\_mailbox\_to\_controller” and “Read\_data\_mailbox\_from\_controller”.

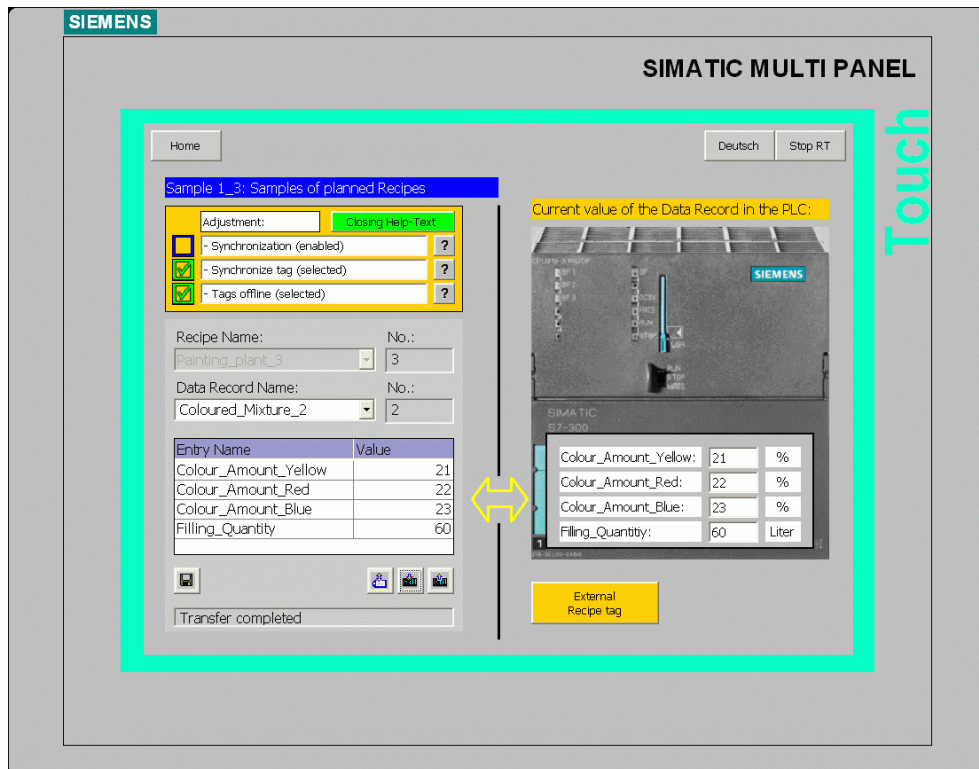
## Sample 1\_3

In the properties of the recipe, the following settings were defined.

Figure 2-6

The overview display of sample 1\_3 is shown below.

Figure 2-7



### Advantages of this solution

- No range pointer is required.
- First, a data record can be selected and – if required – edited without transferring the data immediately to the controller. The displayed data record is only written into the controller after the “Set to PLC” button has been selected. However, changes in the controller are only applied in the recipe tag after the “Read from PLC” button has been selected.

## Sample 1\_4

In the properties of the recipe, the following settings were defined.

Figure 2-8

Adjustment:	Closing Help-Text
<input type="checkbox"/> - Synchronization (enabled)	?
<input checked="" type="checkbox"/> - Synchronize tag (selected)	?
<input type="checkbox"/> - Tags offline (enabled)	?

The overview display of sample 1\_4 is shown below.

Figure 2-9

SIEMENS SIMATIC MULTI PANEL

Home Deutsch Stop RT

Sample 1\_4: Samples of planned Recipes

Adjustment: Closing Help-Text

- Synchronization (enabled) ?

- Synchronize tag (selected) ?

- Tags offline (enabled) ?

Recipe Name: No.:  
Painting\_plant\_4 4

Data Record Name: No.:  
Coloured\_Mixture\_3 3

Entry Name	Value
Colour_Amount_Yellow	31
Colour_Amount_Red	32
Colour_Amount_Blue	33
Filling_Quantity	90

Current value of the Data Record in the PLC:

Colour_Amount_Yellow:	31	%
Colour_Amount_Red:	32	%
Colour_Amount_Blue:	33	%
Filling_Quantity:	90	Liter

External Recipe tag

Touch

## Advantages of this solution

- No range pointer is required.

- When changing a data record, the data are immediately applied to the controller.  
Exception:  
If a value is changed in the recipe display, this change must either be saved or the change has to be confirmed using the “Set to PLC” button or the “Synchronize tag” button.

## Sample 1\_5

In the properties of the recipe, the following settings were defined.

Figure 2-10

The overview display of sample 1\_5 is shown below.

Figure 2-11

**SIEMENS SIMATIC MULTI PANEL**

Home Deutsch Stop RT

Sample 1\_5: Samples of planned Recipes

Adjustment: Closing Help-Text

- Synchronization (enabled) ?  
 - Synchronize tag (enabled) ?  
 - Tags offline (enabled) ?

Recipe Name: No.:  
 painting\_plant\_5 5  
 Data Record Name: No.:  
 Coloured\_Mixture\_1 1

Entry Name	Value
Colour_Amount_Yellow	11
Colour_Amount_Red	12
Colour_Amount_Blue	13
Filling_Quantity	30

Data record read

Current value of the Data Record in the PLC:

SIMATIC S7-300

Colour_Amount_Yellow:	4	%
Colour_Amount_Red:	22	%
Colour_Amount_Blue:	23	%
Filling_Quantity:	60	Liter

External Recipe tag

**Touch**

### Advantages of this solution

- No range pointer is required.
- First, a data record can be selected and – if required – edited without transferring the data immediately to the controller. The displayed data record is only written into the controller after the “Set to PLC” button has been selected. However, changes in the controller are only applied in the recipe tag after the “Read from PLC” button has been selected.

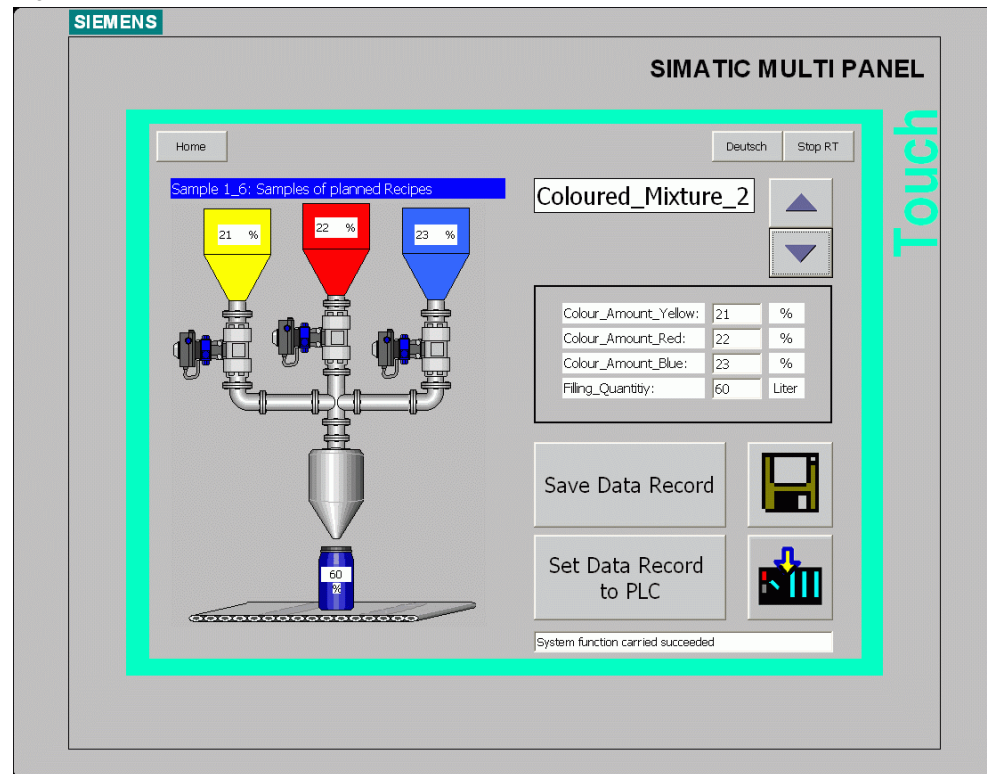
- Changes in the recipe display do not influence the recipe tags which are additionally used outside the recipe display.
- Changes of recipe tags additionally used outside the recipe display do not influence the used recipe tags in the recipe display.

## Sample 1\_6

In this example, the standard recipe display was replaced by an individual display. This case is referred to as a recipe screen.

The overview display of sample 1\_6 is shown below

Figure 2-12



On the left side, the used recipe tags are directly integrated into the “plant display” in order to visually show the operator the meaning of the parameters (recipe tags).

On the right side of the screen, the plant is operated / parameterized.

The following functions were configured:

- Selection of a data record using two scroll buttons.
- Display of the values in tabular form, similar to the standard recipe display.
- Saving of a data record using the “Save Data Record” button and also the button next to this button, which is provided with a graphic.
- Transfer of the data to the PLC using the “Set Data Record to PLC” button and also the button next to this button, which is provided with a graphic.



## Advantages of this solution

- The configurable functions and available image objects enable you to e.g. specify the size of the required buttons to make the operation more convenient for the operator especially if he/she, for example, has to wear gloves.
- Customer requirements can be individually adapted to the local conditions.
- The display can be individually designed.

## Screen call using the “External recipe tag” button

Figure 2-13

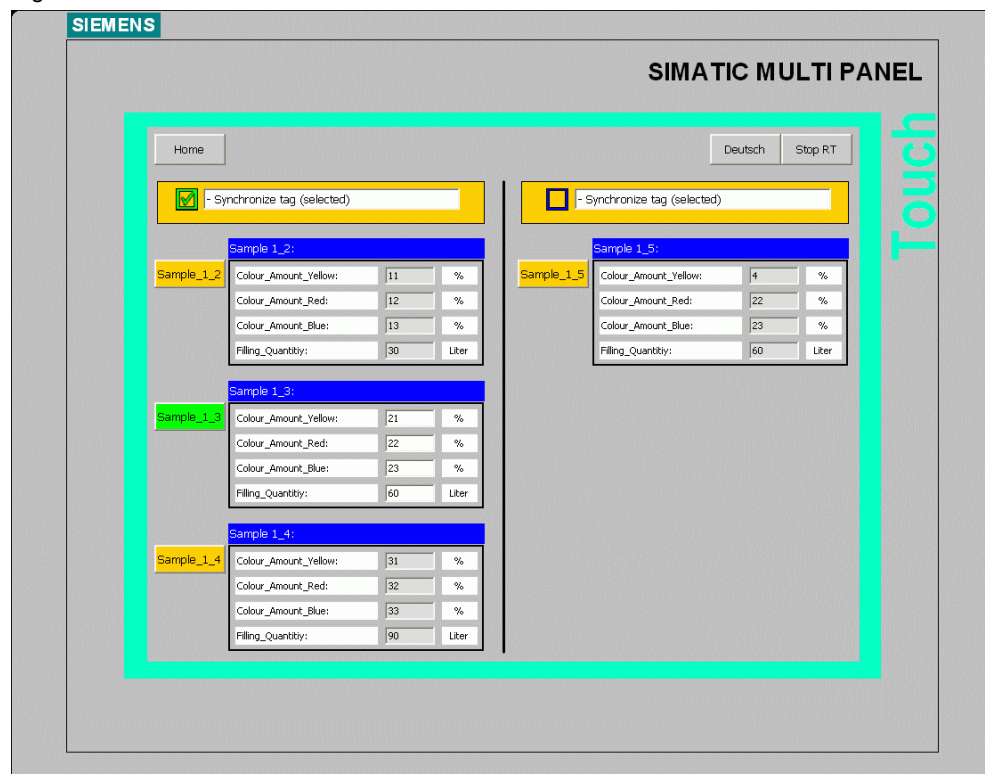


Depending on how the recipe properties have been defined, “Synchronize tag” selected or deselected, different properties ensue for the recipe tags if they are used in further screens in the project.

The program examples “Sample 1\_2 to Sample 1\_5” include the “External recipe tag” button.

The overview display called by selecting the “External recipe tag” button is shown below.

Figure 2-14



The page is graphically divided into two parts.

On the left side, the tags of the recipes are located in which the properties of the “Synchronize tag” recipe are configured.

On the right side, the tags of the recipe are located in which the properties of the “Synchronize tag” recipe are **not** configured.

**Example:**

If, for example, you select the “External tag” button on the page with sample project “Sample 1\_2”, the page with the “external” tags is displayed.

The green button correspondingly indicates the recipe you are currently editing.

You can now make changes to the respective input boxes and subsequently use the button on the right to return to the previous view.

All tags not referring to “Sample 1\_2” are grayed out and cannot be edited.

**Note:**

This page enables you to reconstruct the behavior of the “Synchronize tag” parameter with regard to the

- values in the controller
- tag values in the recipe display
- values of the additionally used recipe tags.

## 2.2.3 Partition 2, Import – Export Recipes

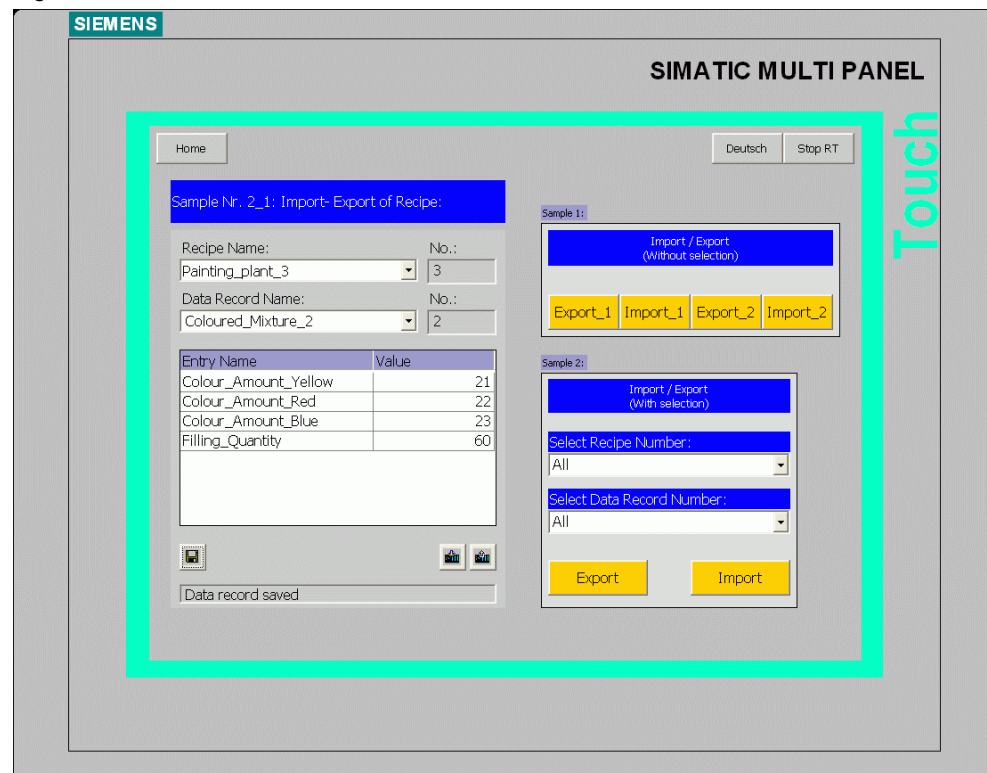
### Import – Export Recipes

#### Sample 2\_1

This screen includes two examples which show you how to import or export a recipe.

The overview display of sample 2\_1 is shown below.

Figure 2-15



The standard recipe display, which has been included in the configuration for control / testing purposes, is located on the left side.

It enables you to determine whether changes have been made in recipes/data records after an export and subsequent import of recipes.

The configured “Export / Import” system functions are located on the right side.

- Import / Export without selection (example 1)
- Import / Export with selection (example 2)

## Example 1, Export\_1 / Import\_1 (without selection)

When using the “Export\_1” button, all recipes including the data records are exported to the specified “\Storage Card” directory/storage location.

When using the “Import\_1” button, all recipes including the data records are imported from the “\Storage Card” directory/storage location into the recipe display.

### Advantages of this solution:

- It is ensured that all recipes including all data records are applied when exporting.
- Easy and quick configuration of the storage path.

## Example 1, Export\_2 / Import\_2 (without selection)

When using the “Export\_2” button, all recipes including the data records are exported to the specified “\Storage Card\Backup” directory/storage location.

This function is executed with the aid of a script.

When using the “Import\_2” button, all recipes including the data records are imported from the “Storage Card\Backup” directory/storage location into the recipe display.

### Advantages of this solution:

- It is ensured that all recipes including all data records are applied when exporting.
- Due to the use of a script, additional subfolders are automatically generated on the storage medium when executing the function.

#### Note:

If you do not use a script, the files will be stored as follows.

If, for example, the storage location

“\Storage Card\Backup” is configured for the Export function, the “Backup” file folder must exist on the storage card so that the exported recipes can be written into the “Backup” subfolder with the configured name.

If the folder does not exist, the name of the “subfolder” is prefixed for the configured recipe names.

Relating to this example, the name would, for example, be

“Backup\_painting\_plant\_2.csv” instead of “Painting\_plant\_2.csv”.

- Additional subfolders enable to create a better data storage structure.

## Example 2, Export / Import

In example 2, a special selection can be made. The operator can preselect whether he/she wants to export or import all recipes or an individual recipe. The same applies to the selection of the data record. The operator can select all or an individual data record of a recipe.

Use the "Export" button to execute the Export function.

Use the "Import" button to execute the Import function.

### **Note:**

When exporting, **one** Excel file is generated in csv file format for each recipe independently of whether all data records of this recipe were to be transferred or whether, for example, only one data record from this recipe was transferred.

The Excel file can thus include all data records of this recipe or only, for example, data record 1 from this recipe if it has previously been selected correspondingly.

## Advantages of this solution

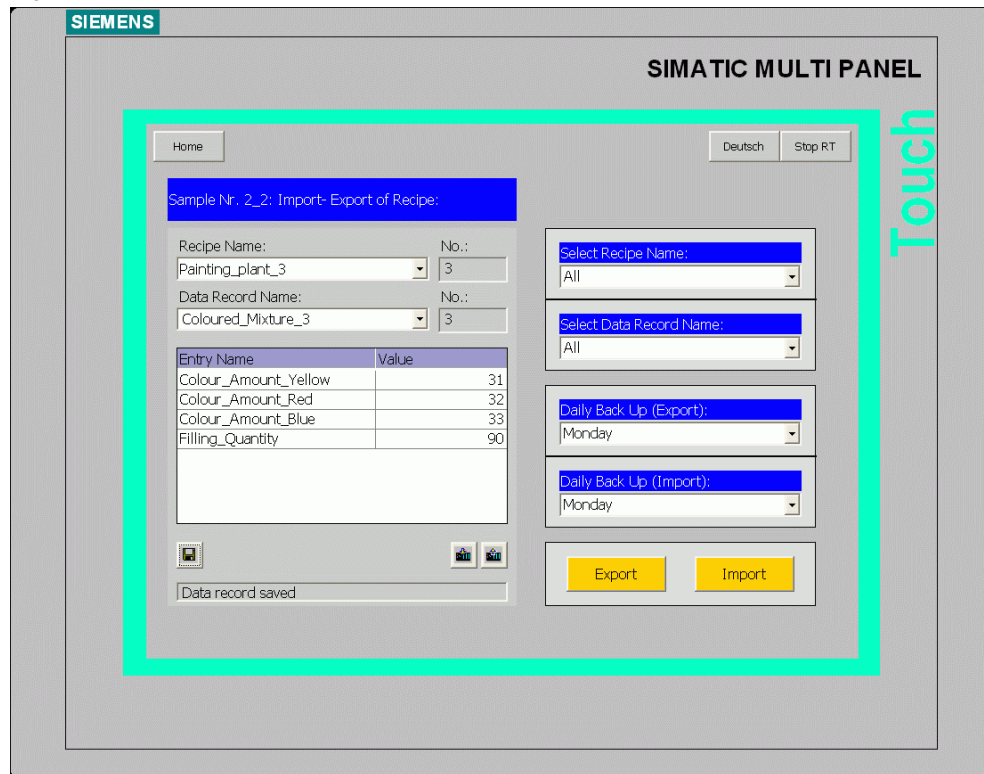
- It is possible to select all but also to specifically select individual recipe and data records.

## Sample 2\_2, Export / Import

This screen includes another example which shows you how to import or export a recipe.

The overview display of sample 2\_2 is shown below.

Figure 2-16



The standard recipe display, which has been included in the configuration for control / testing purposes, is located on the left side.

It enables you to determine whether changes have been made in recipes/data records after an export and subsequent import of recipes.

The configured “Export / Import” system function is located on the right side.

In this example, a special selection can be made:

- The operator can use a text list to select whether he/she wants to export or import all recipes or an individual recipe. The same applies to the selection of the data record. The operator can select all or an individual data record of a recipe.
- The operator can select a subfolder via another text list. In this case, the selection of a “day of the week” is used as a “variable” subfolder.

Use the “Export” button to execute the “Export” function via a script.

Use the “Import” button to execute the “Import” function via a script.

**Note:**

When exporting, **one** Excel file is generated in csv file format for each recipe independently of whether all data records of this recipe were to be transferred or, for example, only one data record from this recipe. The Excel file can thus include all data records of this recipe or only, for example, data record 1 from this recipe if it has previously been selected correspondingly.

**Advantages of this solution**

- The variable storage locations (subfolders) enable to perform “daily backups” of the individual recipes. Depending on the weekday, this enables to easily export or import different recipes / data records.



## 2.2.4 Partition 3, Backup of Recipes

### Backup of Recipes

#### Sample 3\_1

This screen uses graphical information to show how data can be backed up with the aid of ProSave.

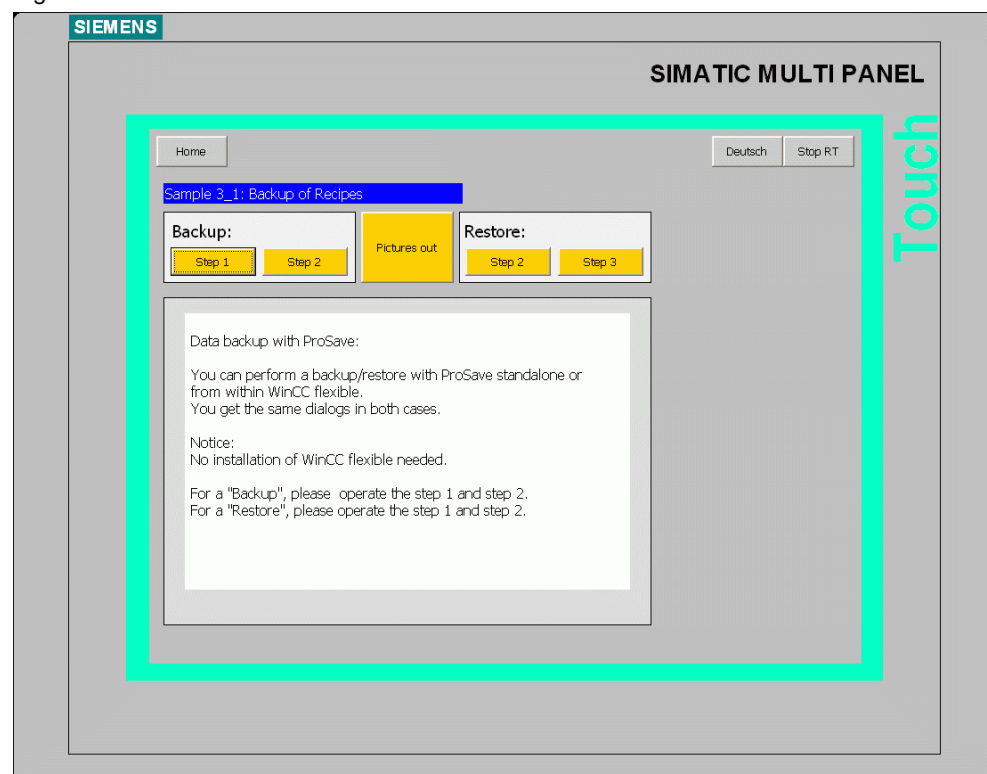
This backup file is only used for data backup/archiving of the existing recipes.

The advantage over a data backup using the “Export / Import of Recipes” system function is that it is not required to program an additional system function in the configuration.

The recipes can be backed up later. However, it has to be considered that a subsequent editing of the data is not possible. In addition, all recipes are always backed up. Individual recipes / data records cannot be selected.

The overview display of sample 3\_1 is shown below

Figure 2-17



In the center of the screen, you find a text description of how to perform a data backup (backup / restore).

**Backup:**

In the “Backup” field, select the “Step 1” button and subsequently the “Step 2” button. A graphical description of how to perform a backup will be displayed.

**Restore:**

In the “Restore” field, select the “Step 1” button and subsequently the “Step 2” button. A graphical description of how to perform a restore process will be displayed.

To close the screens, please use the “Pictures out” button.

## 2.2.5 Partition 4, Sample with an FDA practice

### FDA / GMP compliant projects with WinCC flexible 2005 and audit

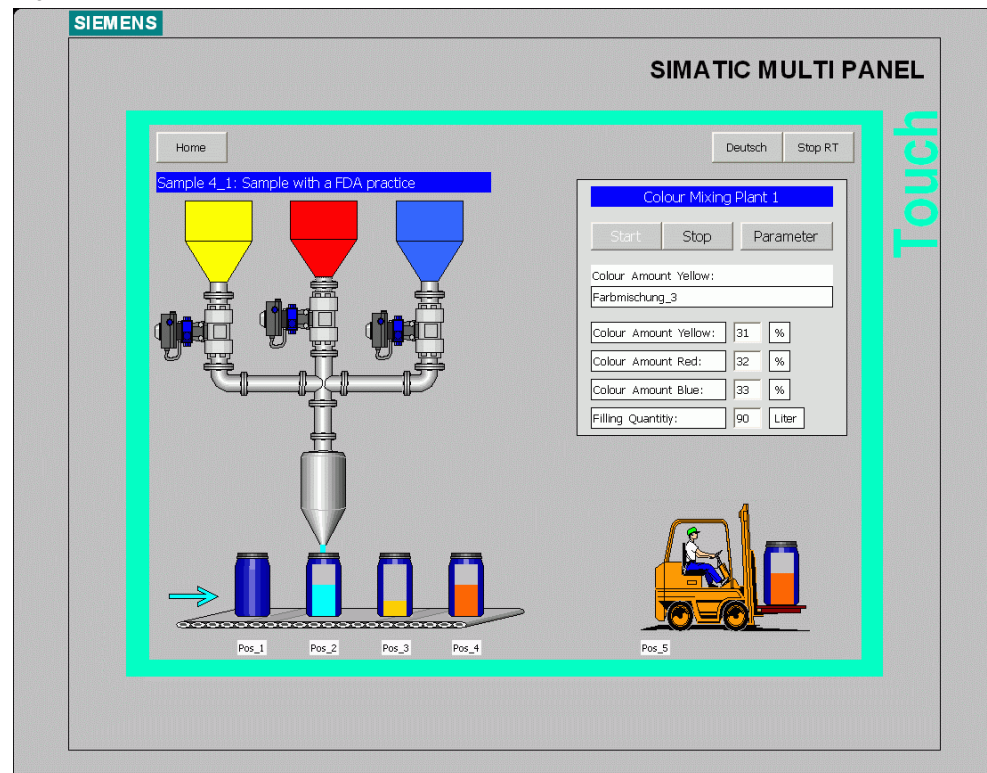
In many industries, e.g. in the pharmaceutical industry, the traceability and thus the documentation of production data is becoming increasingly important. Detailed information on this topic is available in the WinCC flexible online help.

#### Sample 4\_1

This screen uses a simulated production plant to show the application of a recipe in connection with an FDA application.

The overview display of sample 4\_1 is shown below

Figure 2-18



The plant overview shows a color mixing plant.

Depending on the recipe composition, a color is mixed from the containers "yellow / red / blue" and barreled. A conveyor belt is used to transport the barrel until it is finally picked up by the lifting truck and transported to the warehouse or directly to the painting plant.

To provide the operator with an overview, the current mixture of colors (current data record) is displayed in the right part of the screen.

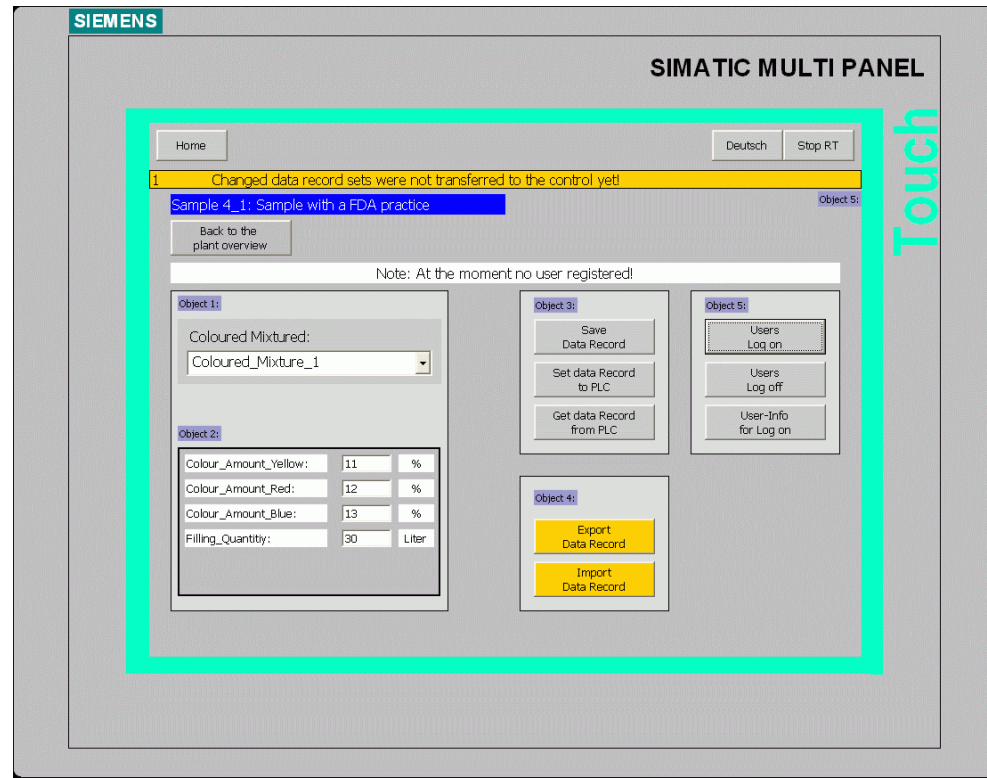
The operator can use the “Start” button to start the filling process and the “Stop” button to stop it.

The function is configured as an FDA-relevant function.

When the operator selects the “Parameter” button, the page which is used to parameterize the plant will be displayed.

Below you see the overview display of the “Parameter” screen, which is displayed to you after using the “Parameter” button from sample 4\_1.

Figure 2-19



### **Object 1:**

Object 1 shows a standard recipe display which is configured in such a way that only the corresponding data record can be selected.

### **Object 2:**

The recipe parameters are output via object 2.

In addition, it can be used to edit the parameters of the recipe.

The recipe tags were configured as “GMP-relevant” in the properties so that a safety query in which you have to confirm the “change” is displayed after a value has been changed.

### **Note:**

The buttons can only be operated if a user is logged on. When the text is silhouetted in black, a user is logged on. When no user is logged on, a text box with corresponding information is displayed above the object.

### **Object 3:**

The buttons can be used to execute the following functions via configured system functions:

- Save data record

- Set data record to PLC
- Get data record from PLC

In addition, these functions were classified / configured as “GMP-relevant” so that a safety query in which you have to confirm the execution is displayed after using one of these buttons.

**Note:**

With regard to the operation, the same restrictions as for object 2 apply.

**Object 4:**

The buttons can be used to execute the following configured system functions:

- Export Data Record
- Import Data Record

In addition, these functions were classified as “GMP-relevant” so that a safety query in which you have to confirm the execution is displayed after using one of these buttons.

**Note:**

With regard to the operation, the same restrictions as for object 2 apply.

**Object 5:**

The buttons can be used to execute the following configured system functions:

- Users log on
- Users log off
- User info for logon

**Note:**

Due to the GMP functionality, all GMP-relevant functions require that a “user” is logged on.

A logoff time of 5 minutes was configured so that you have to log on again after this time has elapsed.

**Object 6:**

If a change is made within the recipe (e.g. a data record was modified/changed), a message is displayed, which indicates that the data in the recipe display and in the PLC differ.

**FDA (GMP-relevant) operational sequence description:**

- Example of changing a data record

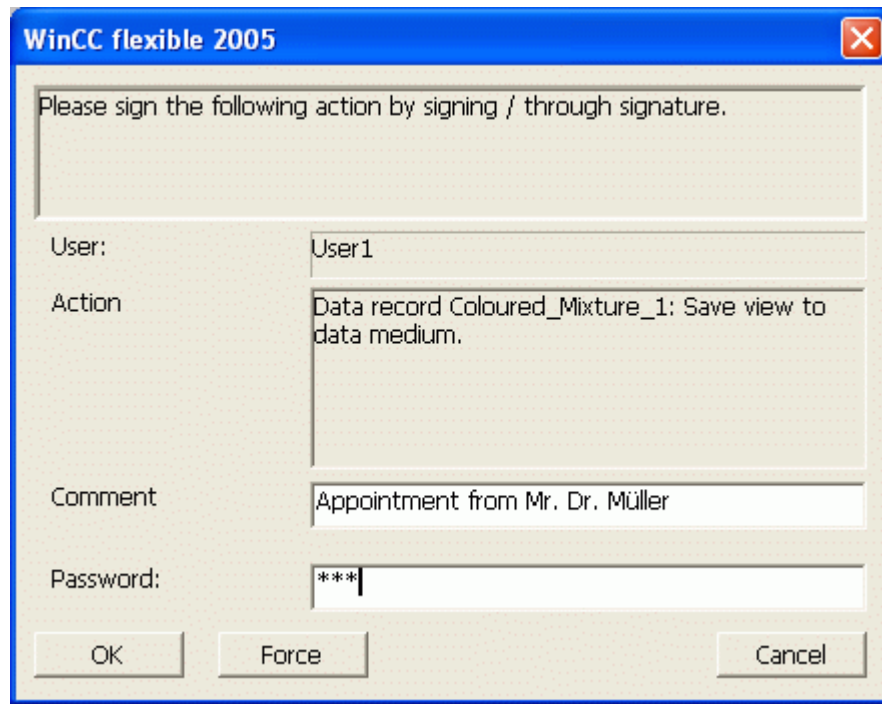
Make sure that you are logged on as “user”.  
A logon can be performed via “object 5”.  
Use the “User info for logon” button to call the corresponding user names and passwords.

You can select three different data records via “object 1”. If you have performed a data record change, a message box is displayed, which indicates that the recipe data have not yet been transferred to controller. To update the data, please execute the “Set data record to PLC” function from “object 3”.

A prompt is displayed.  
You can store a comment, e.g. instruction by Dr. Müller; in addition, enter your password in “Password:” and select “OK” to confirm your input. You have now given an electronic signature. The changed data record is transferred to the controller and the message box is automatically reset.

The prompt is shown below.

Figure 2-20



**Note:**

This prompt appears for all functions which have been configured as GMP-relevant.

The individual functions are listed in the following:

- When a value is changed in "object 2".
- When a data record is saved via "object 3".
- When a data record is to be transferred from the recipe display to the controller via "object 3".
- When a data record is to be transferred from the controller to the recipe display via "object 3".
- When the data records are exported in "object 4".
- When the data records are imported in "object 4".



The figure below is an audit trail recording excerpt in which all configured GMP-relevant data are logged.

Figure 2-21

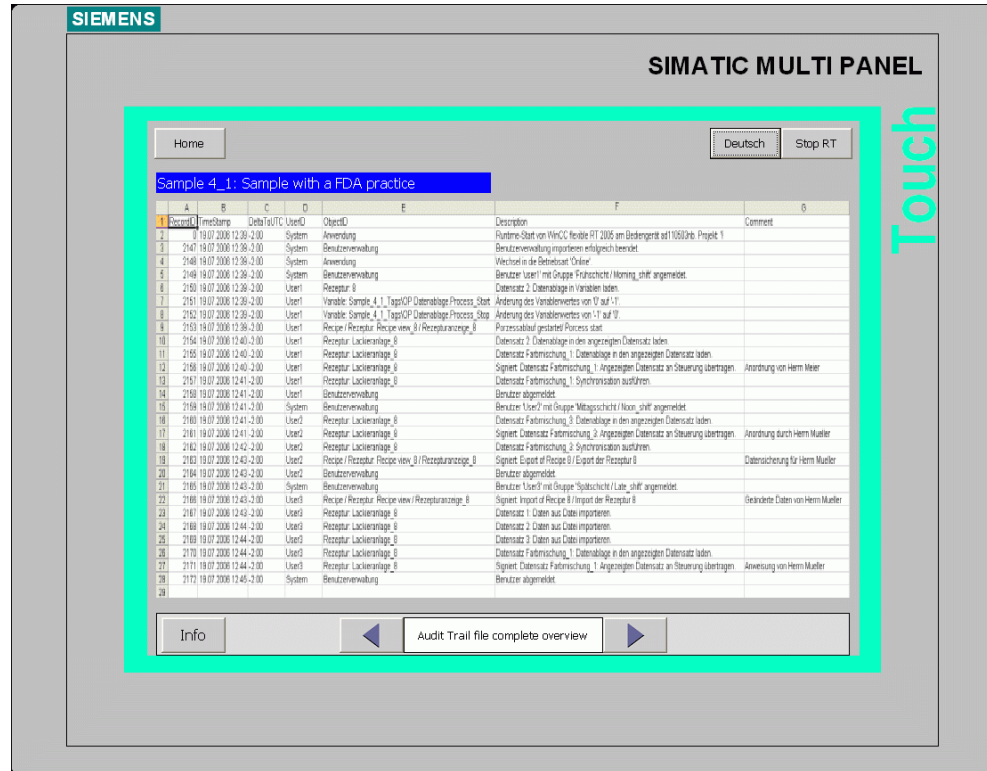
User1	Recipe: Painting_plant_8	Signed: Data record Coloured_Mixture_1: Transfer displayed data record to PLC.
User1	Recipe: Painting_plant_8	Signed: Data record Coloured_Mixture_1: Save view to data medium.

### General note on the configuration:

Changing the value of the color composition does not influence the “graphical representation” of the contents when “barreling”. Merely the filling quantity (Filling\_Quantity) can be “actively” influenced and is also graphically displayed.

Below you see an overview display of a graphically displayed audit trail file. The screen is called by selecting the “Audit Trail File” button from the “Screen\_Sample\_Overview” screen (start screen).

Figure 2-22



The screen center graphically displays an excerpt of the audit trail file stored on the storage card.

Use the “Info” button to display information on this screen.

The limited option to display the entire file as an image as a whole again divides the screen into three sections.

The arrow buttons are used to call the sections.

The text between the arrow buttons informs you of the page (section) that has been called.

## 2.3 Required hardware and software components

### Hardware components

Table 2-1

Component	No.	MLFB / Order number	Note
SIMATIC S7-300, DIN RAIL L=480MM	1	6ES7390-1AE80-0AA0	= minimum length
SIMATIC S7-300, PS 307 LOAD POWER SUPPLY, AC 120/230V, DC 24V, 2A	1	6ES7307-1BA00-0AA0	Or similar
SIMATIC S7-300, CPU 315-2 DP	1	6ES7 315-2AG10-0AB0	Alternatively, any other 300/400 CPU can also be used.
SIMATIC S7, MICRO MEMORY CARD FOR S7-300/C7/ET 200S IM151 CPU, 3.3 V NFLASH, 64 KB	1	6ES7953-8LF11-0AA0	Or larger
12-inch MP370 Touch Multi Panel	1	6AV6 545-0DA10-0AX0	Alternatively, another panel can be used. (xP 270 or higher)
SIMATIC NET, CONNECTING CABLE 830-2 FOR PROFIBUS, PREASSEMBLED CABLE WITH TWO 9-PIN SUB D PLUGS,  TERMINATING RESISTORS REVERSIBLE, 3 M	1	6XV1830-2AH30	2-wire shielded cable with PROFIBUS connectors for connecting the MP370 Touch to the CPU.
Memory card CF card 32 MB  or  memory card PC card (ATA flash) 64 MB	1   1	6AV6 574-2AC00-2AA0   6AV6 574-2AC00-2AF0	

## Standard software components

Table 2-2

Component	No.	MLFB / Order number	Note
SIMATIC S7, STEP7 V5.3	1	6ES7810-4CC07-0YA5	For additional ordering information, system requirements and compatibility of STEP7, please visit our Customer Support pages on the internet.  <a href="http://support.automation.siemens.com">http://support.automation.siemens.com</a>
WinCC flexible 2005 standard SP1	1	6AV6 612-0AA01-1CA5	At least WinCC flexible 2005 standard SP1 is required to open the project.
WinCC flexible /Audit for SIMATIC Panel Single license only license key	1	6AV6 618-7HB01-1AB0	
WinCC flexible /Audit for WinCC flexible RT 2005 Single license only license key	1	6AV6 618-7HD01-1AB0	Only required if the multi panel project is migrated to a PC project.

Copyright © Siemens AG 2006 All rights reserved  
23901413\_WinCC\_flexible\_Rezepturen\_e.doc

## Example files and projects

The following list includes all files and projects used in this example.

Table 2-3

Component	Note
AppIrez1.zip	This zip file contains the STEP 7 project and the WinCC flexible project integrated in STEP 7.

## 2.4 Performance data

### Application software

Table 2-4: Performance data of the application software

Criterion	Performance data	Additional note
Program size	Project: 36 MB Project (.zip) 6.7 MB MMC: 2746 bytes Main memory: 1306 bytes	
Maximum cycle time	3ms	For CPU 315-2 DP
Number of HMI screens	14	

## Principles of Operation and Program Structures

### Content

This part describes the detailed functional sequences of the involved hardware and software components, the solution structures and – where useful – the specific implementation of this application.

It is only required to read this part if you are interested in the interaction of the solution components.

### 3 General Functional Mechanisms

#### You are provided with information on...

the general functional mechanisms which apply with regard to “recipes”.

#### 3.1 Basics on “recipes”

To provide also readers with little or no experience in handling recipes with an introduction to the topic, some basic information on “recipes” will be listed in the following.

#### Basic properties

Data which belong together such as machine parameterizations or production data are combined in recipes. A typical application example is the manufacturing industry and the engineering industry.

#### What is a recipe?

At first, the term recipe only represents the name of a “product”, which, for example, stands for the manufacturing of a brake disc or a specific color type for spraying a car.

##### Example 1:

Different brake discs are to be manufactured on a milling machine.

The name of the recipe is “recipe\_brake\_disc\_front\_wheel”.

To inform the milling machine on

- the drill to be used
- the speed at which the drill is to rotate
- whether a cooler is to be connected or not,

these and, if required, additional parameters are entered as individual **elements** (parameters) in this recipe.

We recommend creating an individual **data record** for each brake disc type within this recipe.

## What is a data record?

The term **data record** is now basically again only the name for the product to be created, in this case a very specific brake disc type.

Each created **data record** automatically includes all parameters which have been previously defined in the **elements**.

The creation of data records later results in a quicker reparameterization of the machine and consequently a considerable saving of time if, for example, the brake disc type is to be consistently changed.

### Example 2:

In a paint mixing plant, differently composed paints are to be mixed for a painting plant and filled into a container.

To inform the paint mixing plant on

- how large the yellow percentage of the paint to be mixed is to be
- how large the red percentage of the paint to be mixed is to be
- how large the blue percentage of the paint to be mixed is to be
- how many liters of the paint are to be filled,

these and, if required, additional parameters are entered as individual **elements** (parameters) in this recipe.

We recommend creating an individual **data record** for each color type within this recipe.

The term **data record** is now basically again only the name for the product to be created, in this case a very specific color type (e.g. dark green, light green, etc.).

Each created **data record** automatically includes all parameters which have been previously defined in the **elements**.

The creation of data records later results in a quicker reparameterization of the machine and consequently a considerable saving of time if, for example, the color composition has to be consistently changed.

To display the recipe on the operator panel, you can:

- Use the standard recipe display
- configure your own recipe screen.

For a detailed description of how to create a recipe and how to configure a recipe display, refer, for example, to the "WinCC flexible 2005 Compact/Standard/Advanced" user guide or to the "Getting Started First Time Users" manual for WinCC flexible.

Please observe chapter 9 "Literature".

This chapter includes links to the corresponding manuals you can download.

In the course of this application description, the created program examples will be described in greater detail.



## 4 Explanations of the Example Program

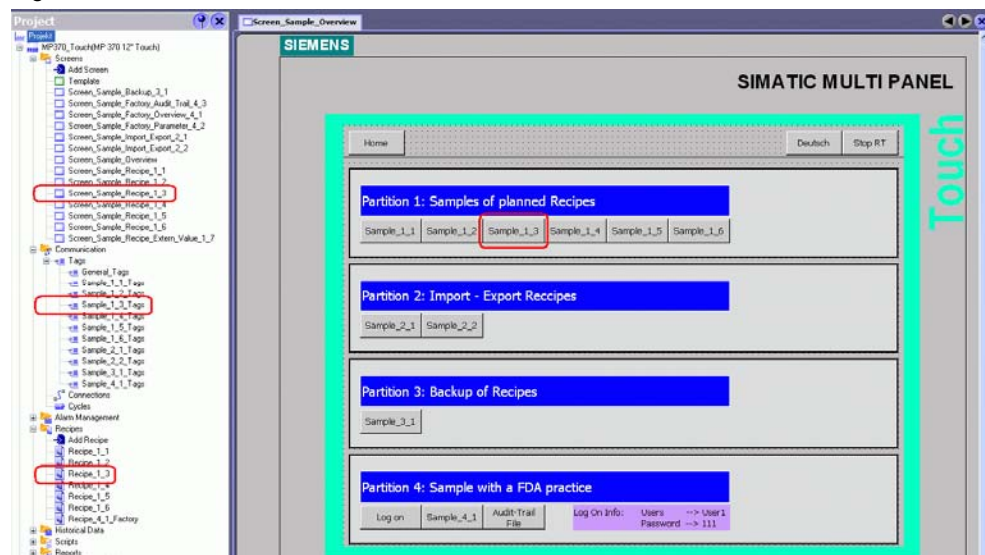
You are provided with...

- valuable information on the structure of the WinCC flexible program
- valuable information on the structure of the STEP7 program
- information on the functions of the individual blocks and networks

### 4.1 Structure of the WinCC flexible project

The WinCC flexible project is structured in such a way that an individual screen, an individual recipe and an individual tag folder were created for each program example.

Figure 4-1



#### Partition 1, sample 1\_3

You find all elements of sample 1\_3 which are relevant to the configuration under the following names:

- “Screen\_Sample\_Recipe\_1\_3” (configured screen)
- “Sample\_1\_3\_Tags” (includes all used tags)
- “Recipe\_1\_3” (includes the recipe for example 3)

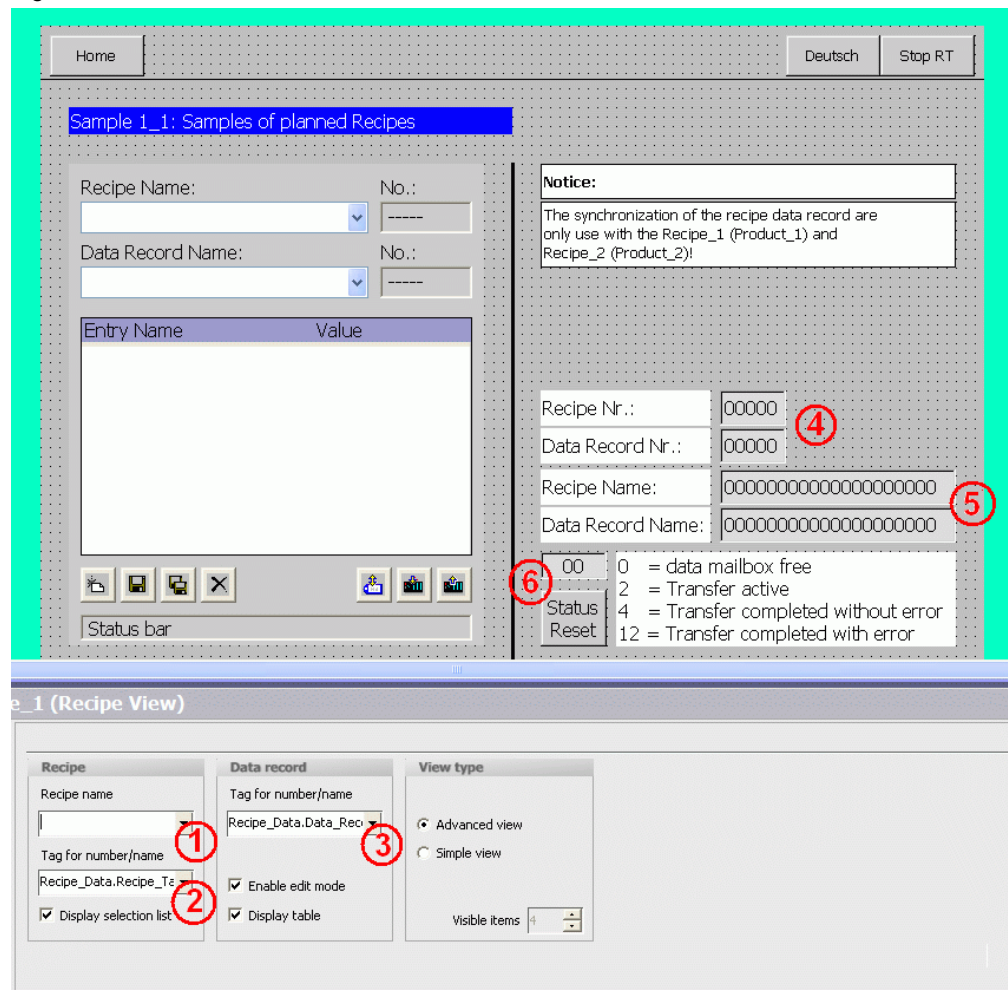
This structure enables you to simply copy individual program parts from the overall program.

In addition, you can reconstruct more easily which tag was used where.

## 4.2 Configuration explanation for “Sample\_1\_1”

Unlike the program examples “Sample\_1\_2 to Sample\_1\_5”, no fixed recipe name is preset in program example “Sample\_1\_1”.

Figure 4-2



Copyright © Siemens AG 2006 All rights reserved  
 23901413\_MinCC\_flexible\_Rezepturen\_e.doc

### 1-3: Parameters of the standard recipe display

If a separate selection of configured recipes is to be possible on the operator panel, nothing is entered in “1”.

Entries in “2 and 3” are only required if you want to evaluate the displayed recipe number or data record number in your PLC program or if you need it for other purposes, for example to be able to display the recipe / data record number in a separate output box.

Otherwise, no entry is required in “2 or 3”.

## 4-6: Additional information

The current recipe and data record number is output in 4. The tag from “2 and 3” was used. In the properties of these two tags, the “GetDataRecordName” function was additionally configured. The function writes the name of the specified recipe and of the specified recipe data record into the specified tag of “5”.

Via “6”

- the current status is displayed.
- the current status is reset by selecting a button.

If data records of “Recipe\_1\_1” and “Recipe\_1\_2” are transferred from or to the controller, the data mailbox has to be reset before another transfer.

The “data mailbox” range pointer with a length of 5 words is used to for the synchronization to the controller. The fourth word of the “data mailbox” range pointer includes the current status. This value is output via an output box.

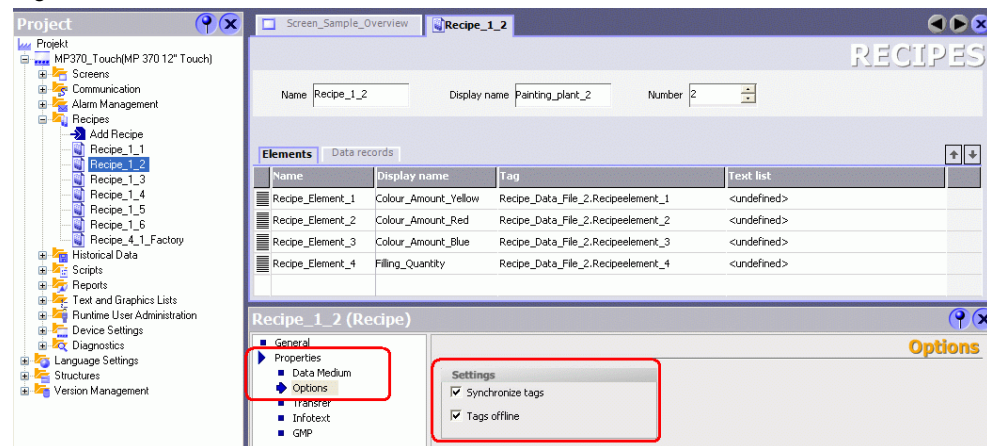
The “status” is reset via the STEP 7 program in OB1 network 1.

For detailed information on the structure of the “data mailbox” range pointer, please refer to the communication manual for Windows-based systems part 1 (related information in chapter 9, Literature).

## 4.3 Configuration explanations “Sample\_1\_2 to Sample\_1\_5”

In place of the program examples “Sample\_1\_2” to “Sample\_1\_5”, sample\_1\_2 is used to explain the essential configuration steps.

Figure 4-3

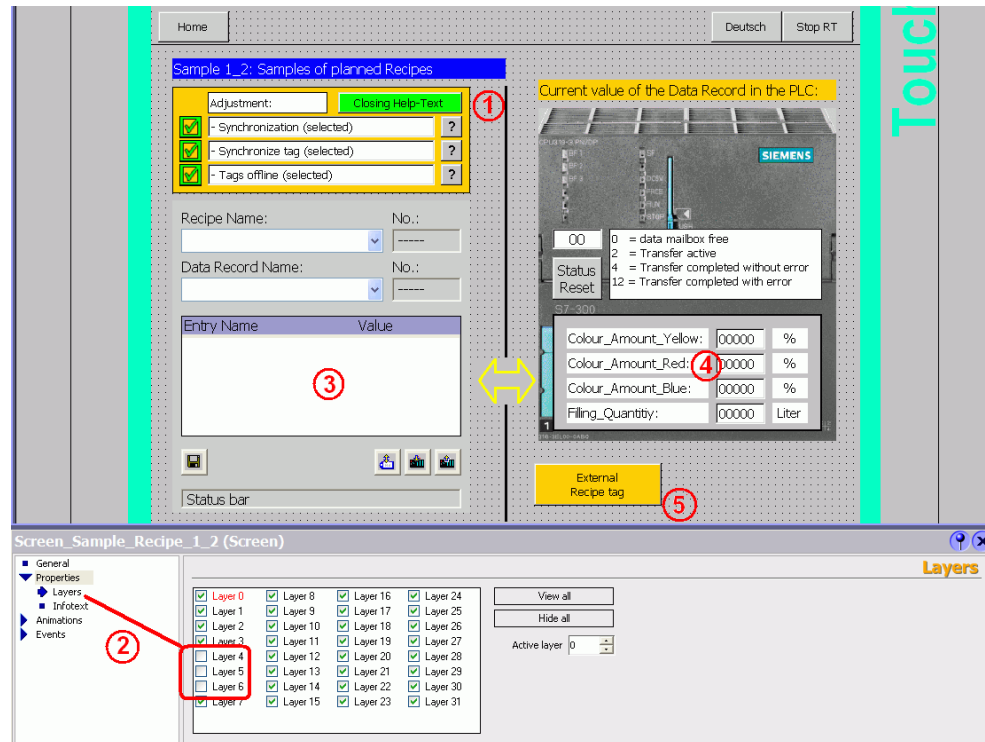


The structure of all recipe examples (examples 1\_2 to 1\_5) is identical. They all have the same number of recipe elements and data records.

They differ in the configured recipe properties. The configuration of these properties differs in each example.

## Structure of the configuration examples

Figure 4-4



→ 1:

“1” graphically shows the configured property of the recipe. You immediately see with which property the called recipe was configured.

A help button via which a graphic screen is called is located next to the graphically represented parameters.

To “visualize” the stored graphic screen elements in the configuration view, please observe the information on “2. The graphic elements were assigned to layers 4, 5 and 6.

In Runtime the “help buttons” are used to show or hide the graphic elements.

→ 2:

Select “Properties > Layers” in the properties of a screen to show or hide the corresponding screen layers. All objects of the respective layers will be shown or hidden.

**Example:**

For instance, layers 4, 5 and 6 were assigned to the “invisible” graphics in “1”. If these layers are deselected, these objects are not displayed.

The advantage of this type of configuration is that parts of a screen (objects) can be shown and hidden to obtain greater clarity.

**Note:**

Showing or hiding the objects via the “layers” **only** has effects in the development environment view.

If objects are to be shown or hidden in Runtime, the “Visibility” system function has to be activated in the properties of the corresponding object.

→ 3:

3 is a standard recipe display which can be selected via “Tools > Enhanced Objects”. A fixed parameter name was selected for each example in the recipe properties. The further properties are listed in the self-explanatory configuration.

→ 4:

4

- symbolically shows an S7-CPU.  
The intention is to make clear that the values between the values of the displayed recipe display and the values in the S7-CPU can differ (depending on the configured recipe properties).
- displays the current values of the loaded recipe in the S7-CPU.

**Note:**

For this purpose, the recipe tag was created twice in the WinCC flexible configuration (same address but different names!)

- The “original” recipe tags were used in the parameterized recipe.
- The “second” tag was used in the display for the “S7-CPU”(same address but with a different name)

**Background:**

Recipes have internal functional mechanisms which, for example, prevent a direct writing of the data into the S7 controller. (Depending on the configured properties)

If the recipe tags were used for the view of the values within the S7-CPU without modification, always the same values as in the recipe display would be displayed in this view.

(Only valid if values are changed within the recipe. Conversely, the recipe tags would behave as additionally used recipe tags (see description for 5!)).

→ 5:

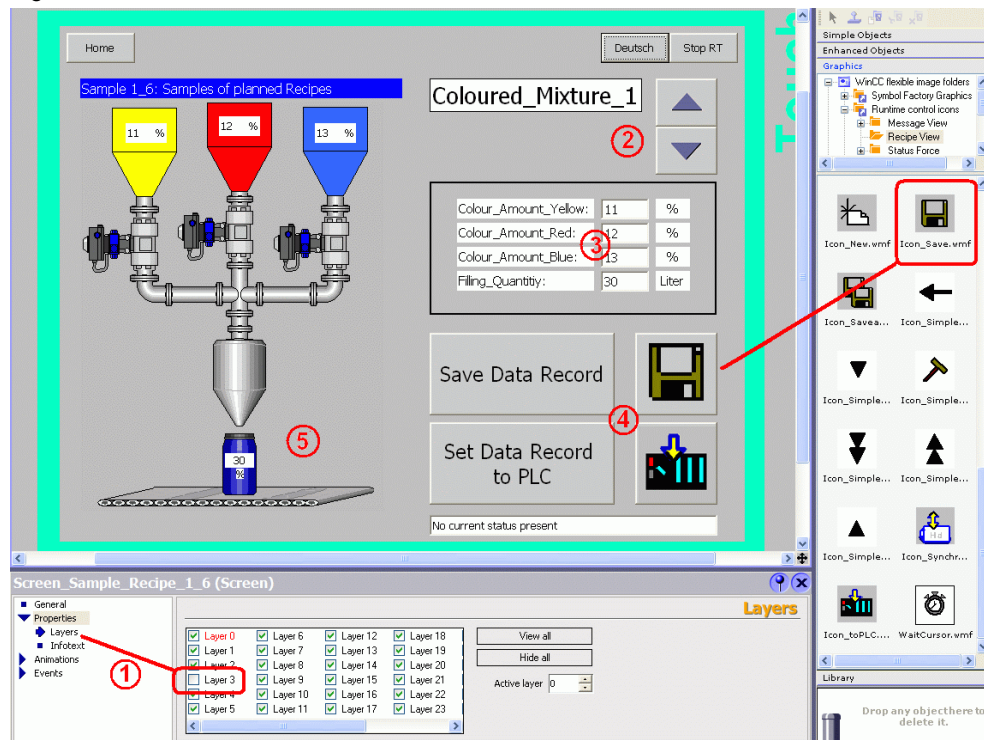
Use the “External recipe tag” button to display the page with the recipe tags which are used in both the recipe display and additionally in a “further” screen named “Screen\_Sample\_Recipe\_Extern\_Value\_1\_7”.

Depending on the properties configured for the recipe, the behavior of the used “external recipe tags” differs with reference to the same recipe tags in the recipe display.

## 4.4 Configuration explanation for “Sample\_1\_6”

In configuration example “Sample\_1\_6”, an individually composed recipe screen is used instead of the standard recipe display for operation.

Figure 4-5



Copyright © Siemens AG 2006 All rights reserved  
23901413\_WinCC\_flexible\_Rezepturen\_e.doc

→ 1:

Select “Properties > Layers” in the properties of a screen to show or hide the corresponding screen layers. All objects of the respective layers will be shown or hidden.

### Note:

Showing or hiding the objects via the “layers” **only** has effects in the development environment view.

If objects are to be shown or hidden in Runtime, the “Visibility” system function has to be activated in the properties of the corresponding object.

In this example, the standard recipe display is shown or hidden via layer 3. This ensures greater clarity. (The visible screen elements are reduced in the configuration view.)

In either case, you have to integrate a standard recipe display into the screen to enable to assign the respective recipe parameters.

A fixed recipe tag and a tag for the data record were assigned in the standard recipe display properties.



A text list is later used to select the corresponding data record via a tag for the data record.

To ensure that the standard recipe display is later not visible on the operator panel, the recipe display was hidden in “Properties > Animations > Visibility”.

**→ 2:**

A text list was configured (“Recipe\_Data\_Record\_Choices”), which includes the same texts as the ones used in the recipe used in the “data record” parameter.

The buttons on the right are used to scroll through this text list. When pressing the scroll buttons, a script is called, which assigns the corresponding data record number to the text list (“Recipe\_Data\_Record\_Choices”).

**→ 3:**

All tags of the used recipe were listed as a “table”. The input/output boxes can be individually adjusted to the required size.

**→ 4:**

Two different types of display are shown:

- One button with describing text
- One button with graphical representation

The same function was assigned to both adjacent buttons.

A large number of already predefined graphics as, for example, used in the standard recipe display are available to you in “Tools > Graphics > Runtime control icons > Recipe View”.

→ 5:

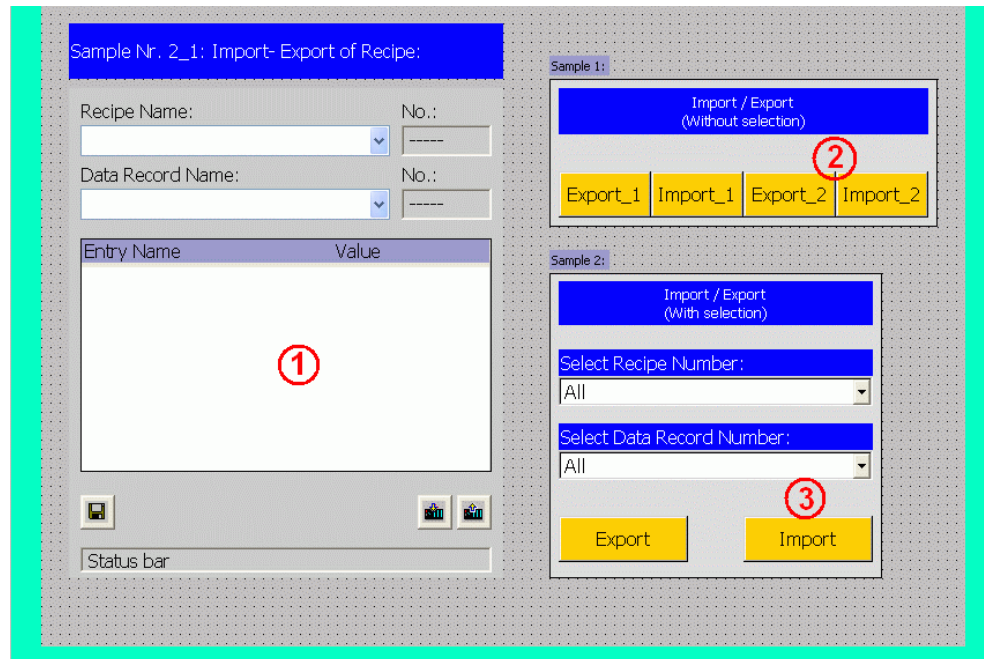
All recipe tags of the configured recipe were not configured as a “table” but directly into the plant display.

A graphic display with the corresponding plant display was inserted as plant overview.

## 4.5 Configuration explanation for “Sample\_2\_1”

This example uses a standard function and a script to explain the Import and Export function of recipes.

Figure 4-6



→ 1:

A standard recipe display was configured. It is not required for the actual function (import / export of recipes). It is only used as a “control function”.

→ 2:

An Import / Export function was configured to the respective button with the aid of the “ExportDataRecords” or “ImportDataRecords” system function.

### Export\_1 button:

The “ExportDataRecords” function was configured to the “Export\_1” button. For information on the individual parameters, please refer to the tool tips / online help.

For information on the storage path your operator panel supports, please refer your operator panel manual.

The following storage path was configured.

\Storage Card

- Storage card = storage location MP370 “PC card”

**Notes:**

- If additional subfolders are created, e.g. “\Storage Card\**Backup**”, it is required that these subfolders have been created/exist on the data medium before executing the Export function.
- If the value “zero” is preset for the “Recipe Number / Name” parameter (all recipes), no additional name has to be created for the recipe. The system then automatically uses the configured “display name” of the recipe.
- If only a very specific recipe is specified in the Export function, for example, recipe\_2 was specifically selected on the “Recipe Number / Name” parameter, it is absolutely necessary to specify the complete file name including the “csv” file extension on the “File name” parameter.

**Example:** \Storage Card\**Backup**\Painting Plant\_2.csv

**Import\_1 button:**

The “ImportDataRecords” function was configured to the “Import\_1” button. For information on the individual parameters, please refer to the tool tips / online help.

To import all previously exported recipe data records, the “ImportDataRecords” function has to be configured for each individual recipe file.

In the “File name” parameter, the complete recipe name has to be entered.

The following file name was configured.

\Storage Card\Painting Plant\_1.csv

- Storage Card = storage location MP370 “PC card”
- Painting Plant\_1.csv = name of the file to be imported

**Export\_2 button:**

Use the “Export\_2 ” button to execute the “ExportDataRecords” function via the “Recipe\_Function\_Export” script.

The advantage of using this script is that the configured “subfolder” is also automatically generated on the memory medium.

### **Import\_2 button:**

The "ImportDataRecords" function was configured to the "Import\_2" button. For information on the individual parameters, please refer to the tool tips / online help.

To import all previously exported recipe files, the "ImportDataRecords" function has to be configured for each individual recipe file.

In the "File name" parameter, the complete recipe name has to be entered.

The following storage path was configured.

\\Storage Card\Painting Plant\_1.csv

- Storage Card= storage location MP370 "PC card"
- Backup = subfolder
- Painting Plant\_1.csv = name of the file to be imported

### **→ 3:**

In this example, individual recipes and data records can be selected via text lists.

#### **"Recipe\_Number" text list:**

- The "Recipe\_Number" text list lists the names of all configured recipes. When a recipe is selected, a value is assigned to the "Textlist\_Recipe\_Number" tag. This value is transferred to the "ExportDataRecords" or "ImportDataRecords" system function. The actual evaluation is performed via the scripts "Recipe\_Export\_Sample\_1" and "Recipe\_Import\_Sample\_1".

#### **"Data\_Record\_Number" text list:**

- The "Data\_Record\_Number" text list lists the names of all configured data records. When one of these data records is selected, a value is assigned to the "Textlist\_Data\_Record\_Number" tag. This value is transferred to the "ExportDataRecords" or "ImportDataRecords" system function. The actual evaluation is performed via the scripts "Recipe\_Export\_Sample\_1" and "Recipe\_Import\_Sample\_1".

**“Export” button:**

Use the “Export” button to call the “Recipe\_Export\_Sample\_1” script.

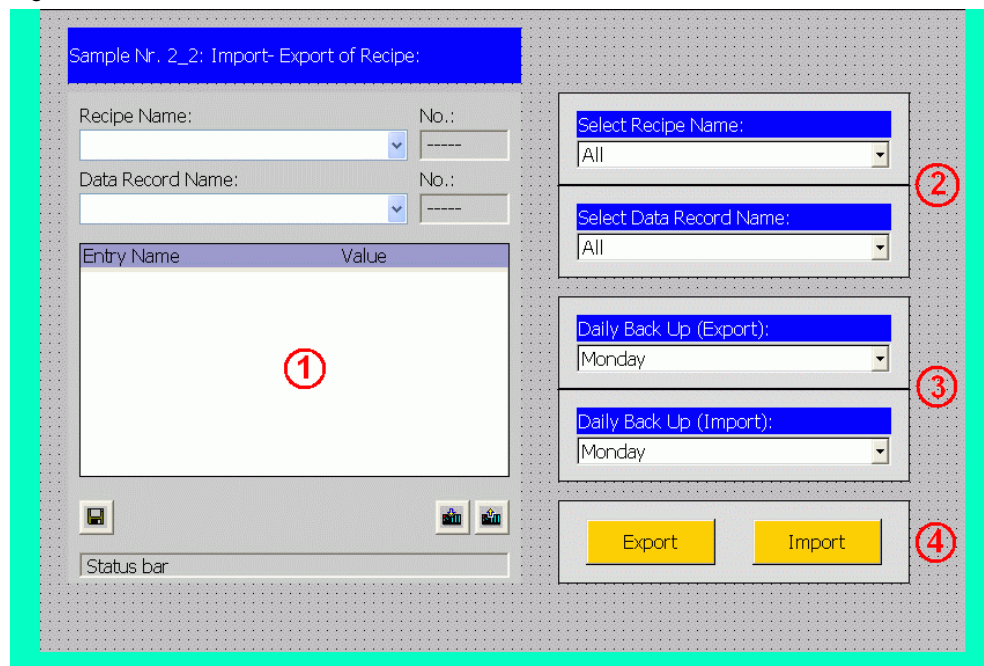
**“Import” button:**

Use the “Import” button to call the “Recipe\_Import\_Sample\_1” script.

## 4.6 Configuration explanation for “Sample\_2\_2”

In comparison with the previous example (sample\_2\_1), the files to be exported can additionally be exported to a folder which can be selected via a text list and reimported from the corresponding folder in this additional example.

Figure 4-7



### → 1:

A standard recipe display was configured. It is not required for the actual function (import / export of recipes). It is only used as a “control function”.

### → 2:

Two text lists were created for the selection of individual recipes and data records.

#### “Recipe\_Number” text list (Select Recipe Name box):

- The “Recipe\_Number” text list lists the names of all configured recipes. When a recipe is selected, a value is assigned to the “Textlist\_Recipe\_Number” tag. This value is transferred to the “ExportDataRecords” or “ImportDataRecords” system function. The actual evaluation is performed via the scripts “Recipe\_Export\_Sample\_2” and “Recipe\_Import\_Sample\_2”.

**“Data\_Record\_Number” text list** (Select Data Record Name box):

- The “Data\_Record\_Number” text list lists the names of all configured data records.  
When one of these data records is selected, a value is assigned to the “Textlist\_Data\_Record\_Number” tag.  
This value is transferred to the “ExportDataRecords” or “ImportDataRecords” system function.  
The actual evaluation is performed via the scripts “Recipe\_Export\_Sample\_2” and “Recipe\_Import\_Sample\_2”.

→ 3:

Two text lists were created for the selection of individual subfolders (weekdays).

**“Path\_Recipe\_Export” text list** (Select Daily Backup (Export) box):

- The “Path\_Recipe\_Export” text list lists the names of the weekdays from “Monday” to “Saturday”.  
When one of these weekdays is selected, a value is assigned to the “Textlist\_Path\_Recipe\_Export” tag.  
This value is transferred to the “ExportDataRecords” or “ImportDataRecords” system function.  
The actual evaluation is performed via the scripts “Recipe\_Export\_Sample\_2” and “Recipe\_Import\_Sample\_2”.

**“Path\_Recipe\_Import” text list** (Select Daily Backup (Import) box):

- The “Path\_Recipe\_Import” text list lists the names of the weekdays from “Monday” to “Saturday”.  
When one of these weekdays is selected, a value is assigned to the “Textlist\_Path\_Recipe\_Import” tag.  
This value is transferred to the “ExportDataRecords” or “ImportDataRecords” system function.  
The actual evaluation is performed via the scripts “Recipe\_Export\_Sample\_2” and “Recipe\_Import\_Sample\_2”.

→ 4:

**“Export” button:**

The “Export” button is used to call the “Recipe\_Export\_Sample\_2” script. The path composition for the export is performed in this script.

**“Import” button:**

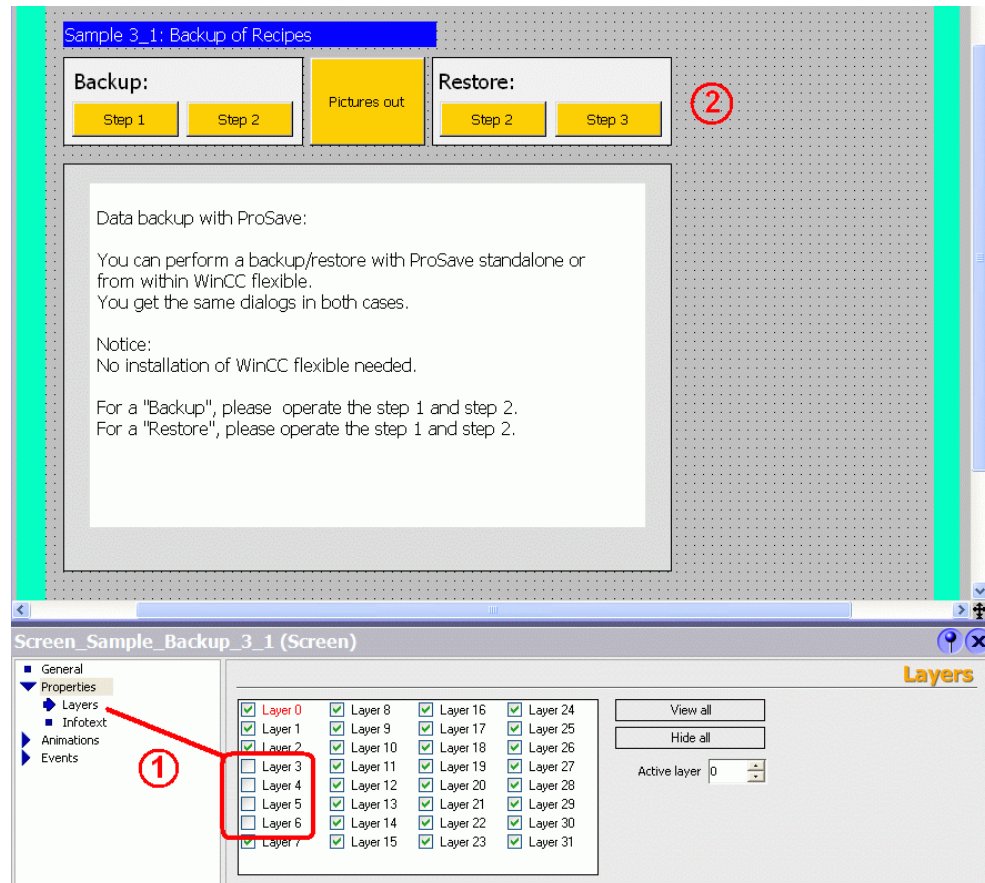
The “Import” button is used to call the “Recipe\_Import\_Sample\_2” script. The path composition for the import is performed in this script.



## 4.7 Configuration explanation for “Sample\_3\_1”

In sample 3\_1, graphics lists are shown / hidden using buttons.

Figure 4-8



→ 1:

Select “Properties > Layers” in the properties of a screen to show or hide the corresponding screen layers. All objects of the respective layers will be shown or hidden.

**Note:**

Showing or hiding the objects via the “layers” **only** has effects in the development environment view.  
If objects are to be shown or hidden in Runtime, the “Visibility” system function has to be activated in the properties of the corresponding object.

In this example, different “layers” have been assigned to four graphics lists (layer 3 to 6). By selecting one of the layers deselected in the screen, the corresponding graphics list is displayed. This type of configuration ensures greater clarity.

→ 2:

By pressing the buttons “Step 1” or “Step 2”, an internal tag of the Bool type is set.

A graphics list associated to the button is shown via this tag.

Use the “Pictures out” button to close all screens.

**Note:**

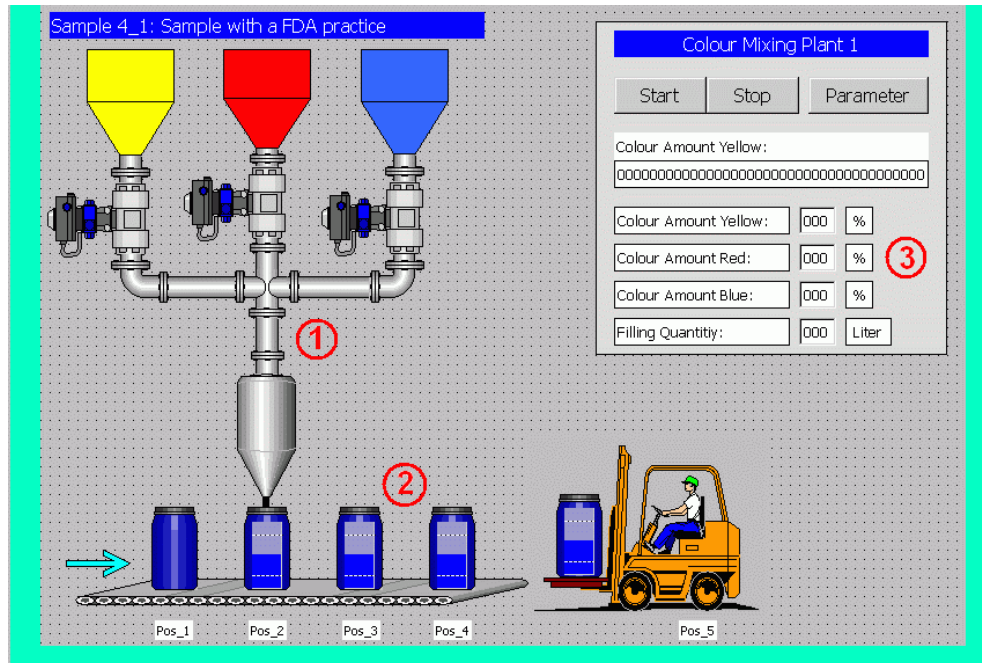
Whether a graphics list is “visible” is defined in the graphics list properties (=> “Properties > Animations > Visibility”).

## 4.8 Configuration explanation for “Sample\_4\_1”

Sample 4\_1 shows two functions of which the configuration was connected:

- A simulated plant is used to illustrate a possible option to represent and display a recipe.
- An FDA compliant configuration.  
 Each change in the production sequence, e.g. changing or modifying a data record, is logged in an audit trail file and saved in an archive as “csv” file.

Figure 4-9



→ 1:

1 displays different graphic objects made available by WinCC flexible in a symbol library in “Tools > Enhanced Objects > Symbol Library”. The symbols can be adjusted to the conditions.

→ 2:

2 displays the filling process and the subsequent transport of the barrels.

The “barrels” are composed of several objects (grouped) (a barrel symbol from the symbol library and a bar chart).

To ungroup a barrel, select the barrel with the right mouse button.

You will find the “Ungroup” function in the opening context menu. You can now edit the individual elements and view the configured properties.

**Note:**

Before ungrouping an object, first create a copy of this object or note down the configured properties of the grouped object.

If an object is ungrouped, the configured properties of the grouped object are lost.

→ 3:

3 displays an operator block via which the plant is operated and system parameters are displayed.

The displayed values are from the recipe tags of “Recipe\_4\_1\_Factory”.

By selecting the “Start” button, a bit is set in the controller via which the filling process is started. In addition, the “NotifyUserAction” function was configured to the button. Each operation is logged in an audit trail file.

Operation requires that a data record is selected. If no data record is selected, a bit is set to “True” by the controller. The operation of the button is suppressed via this “bit”. The font is also displayed differently via the bit (white font).

By selecting the “Stop” button, a bit is set in the controller via which the current process is stopped. In addition, the “NotifyUserAction” function was configured to the button.

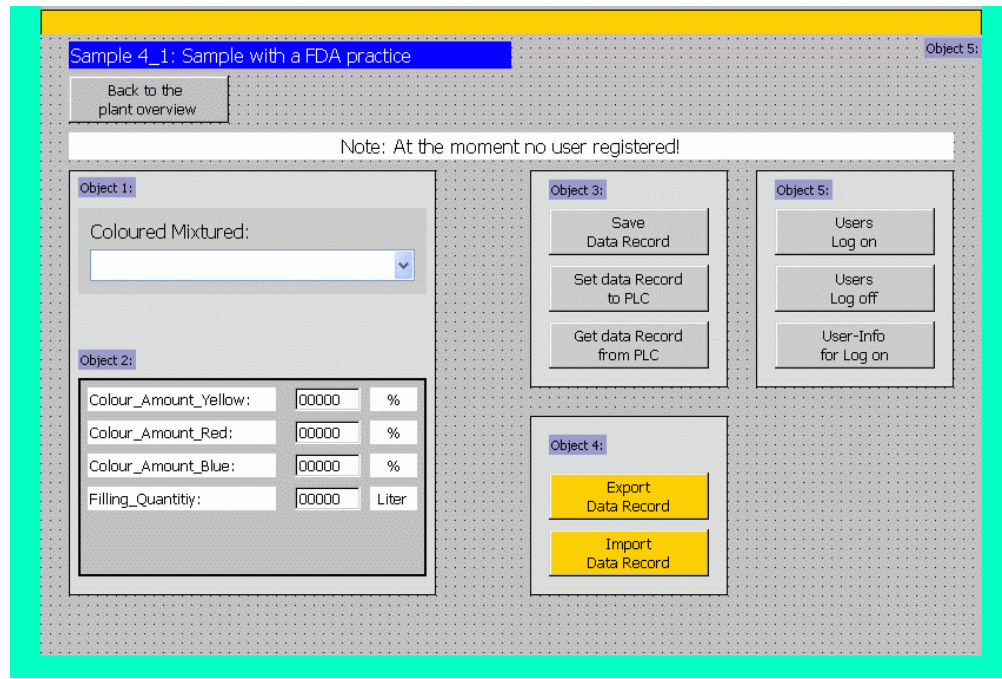
Each operation is logged in an audit trail file.

The “Screen\_Sample\_Factory\_Parameter\_4\_2” screen is called by selecting the “Parameter” button. The plant can be parameterized in the called page.

## Overview “Screen\_Sample\_Factory\_Parameter\_4\_2” screen parameters page

The “Screen\_Sample\_Factory\_Parameter\_4\_2” screen includes four different “objects”, which will be described in greater detail in the following: These buttons / functions are used to parameterize the plant.

Figure 4-10

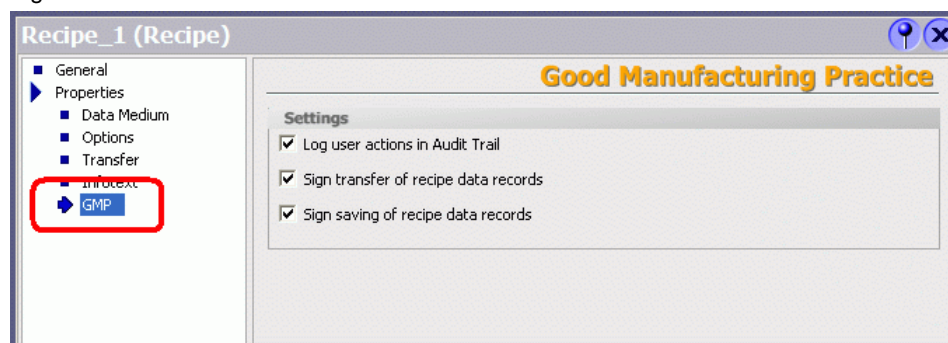


### Object 1:

Object 1 is a standard recipe display configured in such a way that only the data records of “Recipe\_4\_1\_Factory” can be selected. “Recipe\_4\_1\_Factory” is preset.

In the recipe properties in “GMP” (Good Manufacturing Practice), all options were activated.

Figure 4-11



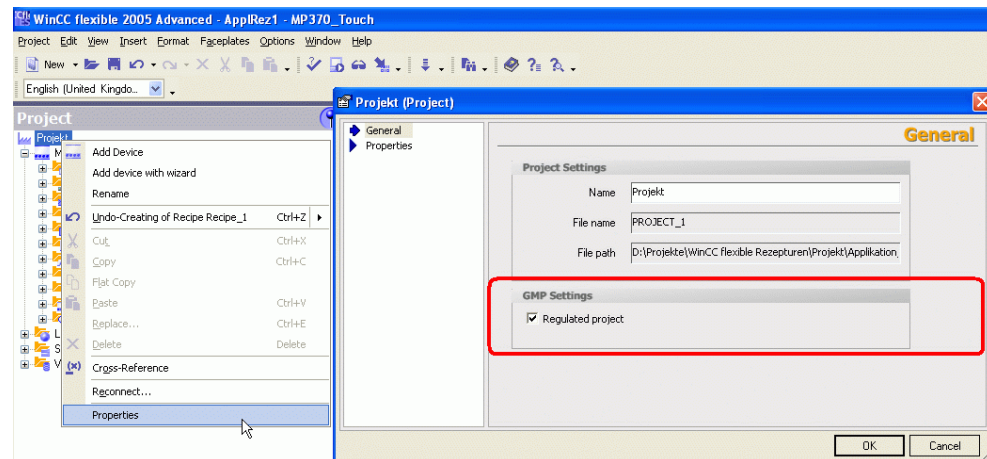
### Note:

The GMP functionality can only be selected if the project is a “regulated project”.

## Creating a “regulated project”:

In the project tree, use the right mouse button to select the “project”.  
In the opening context menu, click “Properties”. In “General”, select the “Regulated project” function.  
The /Audit option is now activated for the project.

Figure 4-12



For all operator panels supporting /Audit WinCC flexible is expanded by the following configuration options:

- Project window, “Historical Data” group: “Audit Trail” entry  
With “Audit Trail” an archive is configured in which all user actions and system processes with Audit relevance are logged in Runtime. The following storage path was defined for this project  
“Storage Card\Factory\AuditTrail”.
- Properties window of the tags, Properties group: “GMP Settings” entry.  
In this setting, you define which tags are GMP-relevant and whether the user has to confirm a value change of the tag with his/her electronic signature in Runtime.

- Properties window of the recipes, Properties group: “GMP Settings” entry  
You define for which recipes the operator actions are logged in the audit trail and whether the operator has to confirm these actions with his/her electronic signature.
- “NotifyUserAction” system function  
With this system function you detect those user actions in the audit trail which are not automatically entered in the audit trail. In addition, you can use this system function to request an acknowledgement or an electronic signature for the operator action and a comment from the user.

**Note:**

The “AuditTrail” function can also be activated later.

This enables you to subsequently integrate this function into already existing projects.

Detailed information on “GMP” is available in the WinCC flexible online help.

**Object 2:**

All recipe tags of the used “Recipe\_4\_1\_Factory” were listed as a table in object 2.

A user has to be logged on to change a value in object 3.

For this purpose, “enabled” was activated in the properties of the I/O boxes “Properties > Enable object > Enabled”.

For the visual display of “enabled”, “appearance” was activated in the properties of the I/O boxes “Properties > Animations > Appearance”. If no user is logged on, the foreground color changes from black to white.

**Object 3:**

In object 3, recipe-typical functions are executed via buttons.

The configured functions are from the “Function List > Keyboard operation for screen objects” folder.

The “NotifyUserAction” function was additionally configured to the “Get data record from PLC” button.

Each operation of this button is logged in an audit trail file.

The buttons in object 3 can only be operated if a user is logged on.

For this purpose, “enabled” was activated in the properties of the buttons “Properties > Enable object > Enabled”.

For the visual display of “enabled”, “appearance” was activated in the properties of the buttons “Properties > Animations > Appearance”. If no user is logged on, the foreground color changes from black to white.

**Note:**

The “NotifyUserAction” function is not required for the buttons “Save Data Record” and “Set Data Record to PLC”. The functionality already exists automatically due to the GMP functionality activated for “Recipe\_4\_1\_Factory”.

**Object 4:**

In object 4, recipe-typical functions are executed via buttons.

The “Export Data Record” button is used to copy all data records of “Recipe\_4\_1\_Factory” to the “\Storage Card\Factory\_Data\Recipe\_8.csv” directory.

The “Import Data Record” button is used to copy all data records of “Recipe\_4\_1\_Factory” from the “\Storage Card\Factory\_Data\Recipe\_8.csv” directory to the recipe display.

The buttons in object 4 can only be operated if a user is logged on. For this purpose, “enabled” was activated in the properties of the buttons “Properties > Enable object > Enabled”.

For the visual display of “enabled”, “appearance” was activated in the properties of the buttons “Properties > Animations > Appearance”. If no user is logged on, the foreground color changes from black to white.

**Object 5:**

The GMP functionality requires that a user is logged on when data are changed.

The “ShowLogonDialog” function is configured to the “Users log on” button. After selecting this button, a logon dialog box is displayed via which the user can log on.

The “Logoff” function is configured to the “Users log off” button. The current user is logged off after operating this button.

An internal bit is set via the “User info for logon” button. This bit is used to show or hide a text box. The data for the three configured “users” are stored in the text box.

The actual text box is shown or hidden in the configuration user interface by “layer 3”, which can be activated in the screen properties “Properties > Layers”.



**Notes:**

- Three users were configured. Information on how to configure a user management is available in the online help.
- The “GetGroupNumber” standard function is used in the scheduler. The number of the logged on group to which the user logged on at the operator panel belongs is written into the specified tag via this function. The evaluation whether the buttons used in objects 3 and 4 can be operated is then performed via this tag.

**Object 6:**

Object 6 represents a message display.

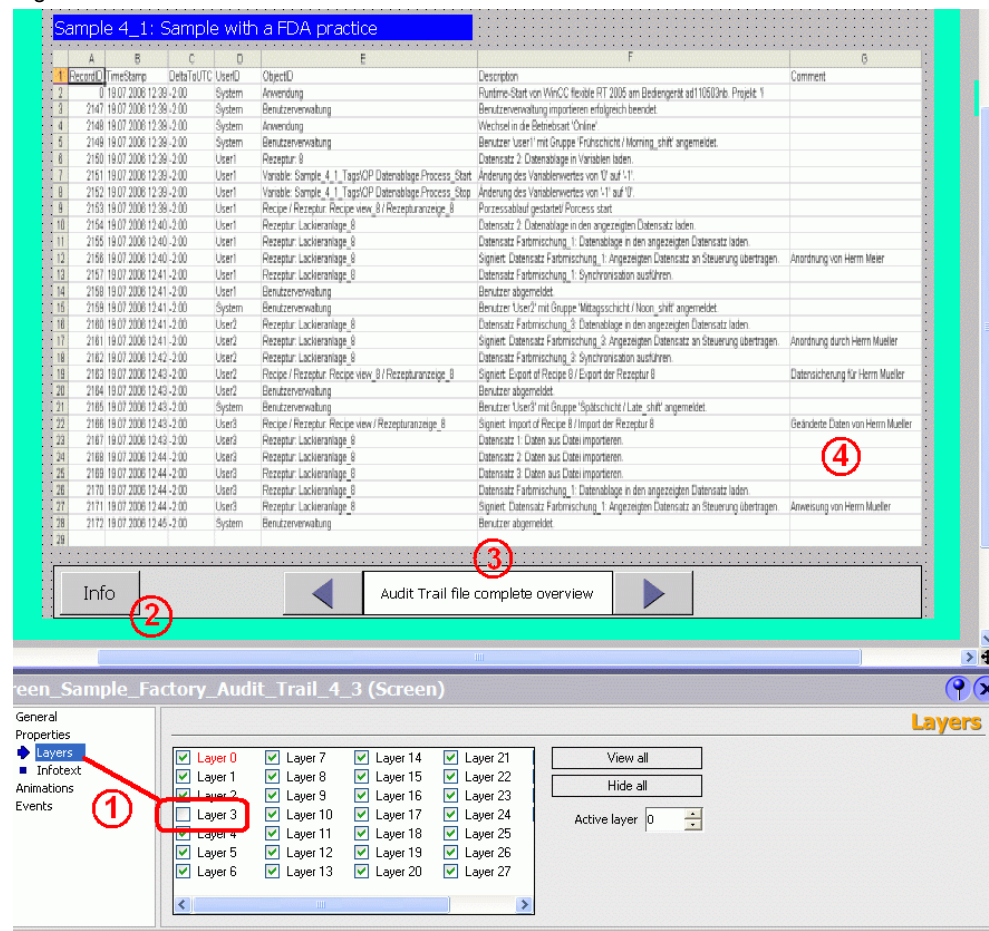
The message display is only shown on the operator panel when a message is present.

The visibility of the message display is defined in the message display properties in “Properties > Animations > Visibility”.

### Overview “Screen\_Sample\_Factory\_Audit\_Trail\_4\_3” screen

The “Screen\_Sample\_Factory\_Audit\_Trail\_4\_3” screen graphically shows an excerpt from an audit trail file.

Figure 4-13



**1:** Select “Properties > Layers” in the properties of a screen to show or hide the corresponding screen layers. All objects of the respective layers will be shown or hidden.

**Note:** Showing or hiding the objects via the “layers” **only** has effects in the development environment view. If objects are to be shown or hidden in Runtime, the “Visibility” system animation function has to be activated in the properties of the corresponding object.

In this example, a text box has been assigned to “layer 3”. By selecting layer 3, a text list is displayed.

In Runtime this text list is shown or hidden using the “Info” button.

**2:**

The “InvertBit” function is configured to the “Info” button. A text list is shown or hidden via the connected tag.

The text list includes an “info text” on this page.

**3:**

The “arrow buttons” are used to call the scripts “Graphic\_List\_Minus” and “Graphic\_List\_Plus”.

In the scripts, the current value of the “Audit\_Trail\_Graphic\_List” tag is added or subtracted with the factor “1”.

The evaluation / selection of the screen to be displayed in the “graphical I/O box” in “4” is performed via the “Audit\_Trail\_Graphic\_List” tag.

**4:**

4 is a “graphical I/O box”.

According to the value of the “Audit\_Trail\_Graphic\_List” tag, graphical excerpts from the audit trail file are shown via the “Audit\_Trail\_Graphic” graphics list.

## 4.9 Structure of the STEP 7 project

The figure below shows the program structure / program blocks of the STEP 7 project.

Figure 4-14

Object name	Symbolic name	Created in language	Size in th...	Type
Systemdaten	---	---	---	SDB
OB1	---	STL	220	Organization Block
FB2	Progr_General	STL	828	Function Block
FB3	Progr_Visible	STL	166	Function Block
FB4	Progr_Filling	STL	542	Function Block
FB5	Progr_Transport	STL	1774	Function Block
FB6	Progr_Fork_Lift	STL	468	Function Block
DB1	OP Datenablage	DB	98	Data Block
DB2	Instanz_DB_FB2	DB	64	Instance data block for FB 2
DB3	Instanz_DB_FB3	DB	38	Instance data block for FB 3
DB4	Instanz_DB_FB4	DB	64	Instance data block for FB 4
DB5	Instanz_DB_FB5	DB	64	Instance data block for FB 5
DB6	Instanz_DB_FB6	DB	64	Instance data block for FB 6
DB10	Recipe_Data	DB	98	Data Block
DB12	Recipe_Data_File_1	DB	92	Data Block
DB14	Recipe_Data_File_2	DB	94	Data Block
DB16	Recipe_Data_File_3	DB	94	Data Block
DB18	Recipe_Data_File_4	DB	94	Data Block
DB20	Recipe_Data_File_5	DB	94	Data Block
DB22	Recipe_Data_File_6	DB	96	Data Block
DB100	Filling_Line	DB	54	Data Block
DB101	Position_1	DB	54	Data Block
DB102	Position_2	DB	68	Data Block
DB103	Position_3	DB	54	Data Block
DB104	Position_4	DB	54	Data Block
DB105	Position_5	DB	54	Data Block
DB110	Recipe_Data_Factory	DB	94	Data Block
VAT_1	VAT_1	---	---	Variable Table
SFB4	TON	STL	---	System function block
SFC20	BLKMOV	STL	---	System function
SFC46	STP	STL	---	System function

Copyright © Siemens AG 2006 All rights reserved  
23901413\_WinCC\_flexible\_Rezepturen\_e.doc

The following table lists and explains the used STEP 7 blocks.

Table 4-1

Block	Explanation
OB1	Organization block (called by the operating system) for the cyclic program processing.
FB2	Simulation program: Includes functions generally valid for the simulation program.
FB3	Simulation program: The evaluation when an object is to be shown or hidden in Runtime is performed in the FB.
FB4	Simulation program: Includes the sequence (step sequence) for the filling process.
FB5	Simulation program: Includes the sequence (step sequence) for the transport process.
FB6	Simulation program: Includes the sequence (step sequence) for the lifting truck.

Block	Explanation
DB1	Simulation program: Includes the generally valid data for the simulation program.
DB2	Simulation program: Instance DB for FB2.
DB3	Simulation program: Instance DB for FB3.
DB4	Simulation program: Instance DB for FB4.
DB5	Simulation program: Instance DB for FB5.
DB6	Simulation program: Instance DB for FB6.
DB10	Examples recipes: Includes the data for generally valid functions as, for example, the "data record range pointer".
DB12	Examples recipes: Includes the data for recipe 1 (Recipe_1_1).
DB14	Examples recipes: Includes the data for recipe 2 (Recipe_1_2).
DB16	Examples recipes: Includes the data for recipe 3 (Recipe_1_3).
DB18	Examples recipes: Includes the data for recipe 4 (Recipe_1_4).
DB20	Examples recipes: Includes the data for recipe 5 (Recipe_1_5).
DB22	Examples recipes: Includes the data for recipe 6 (Recipe_1_6).
DB100	Simulation program: Includes the data for the filling process (filling stream).
DB101	Simulation program: Includes the data for position 1 (Pos 1). (See "Screen_Sample_Factory_Overview_4_1" screen) The data are always transferred to the next station for further transport.
DB102	Simulation program: Includes the data for position 2 (Pos 2) (see "Screen_Sample_Factory_Overview_4_1" screen). The data are always transferred to the next station for further transport.
DB103	Simulation program: Includes the data for position 3 (Pos 3) (see "Screen_Sample_Factory_Overview_4_1" screen). The data are always transferred to the next station for further

Block	Explanation
	transport.
DB104	Simulation program: Includes the data for position 4 (Pos 4) (see "Screen_Sample_Factory_Overview_4_1" screen). The data are always transferred to the next station for further transport.
DB105	Simulation program: Includes the data for position 5 (Pos 5) (see "Screen_Sample_Factory_Overview_4_1" screen). The data are always transferred to the next station for further transport.
DB110	Simulation program: Includes the data for "Recipe_4_1_Factory".
SFB4	Switch-on delay: SFB 4 "TON" delays a rising edge by the time PT
SFC20	Copying memory area (BlockMove) You use SFC20 "BLKMOV" (block move) to copy the contents of one memory area (= source area) to another memory area (= target area).
SFC46	Setting CPU to STOP. If an error occurs in the configured step sequence, the CPU goes to STOP (only required for the configuration phase).

## 5 Modifications to the Example Program

You are provided with information on...

what you can do to adapt the sample project to your project environment.

### 5.1 Adapting the storage path for the recipes / audit trail file

“\Storage Card\...” was selected as storage path for all data to be stored (recipes and audit trail file).

For the MP370 this corresponds to the storage location for the “PC card”.

If you want to store your data in another storage location, e.g. CF card, network or a corresponding drive for a PC project, please change the preset storage location accordingly.

Information on the specific storage locations supported by your operator panel is available in the respective manual.

Example MP370:

- PC card: Slot A, \Storage Card\...
- CF card: Slot B, \Storage Card2\...
- Network: \\PC\_Name\Share Name\....

The storage location / path has to be adapted in the following objects.

- Recipes
- Archive of the audit trail file
- Scripts

# Structure, Configuration and Operation of the Application

## Content

This part takes you step by step through structure, important configuration steps, startup and operation of the application.

## 6 Installation and Startup

### You are provided with information on...

the hardware and software you have to install and the steps necessary to start up the example.

### 6.1 Installation of hardware and software

This chapter describes which hardware and software components have to be installed. The descriptions and manuals as well as delivery information included in the delivery of the respective products should be observed in any case.

#### Hardware installation

For details on the hardware components, please refer to chapter 2.1. For the hardware configuration, please follow the instructions listed in the table below:

Table 6-1

No.	Instruction	Comment
1.	Install your used S7-300/400 controller according to the installation regulations. The following FAQ provides more information on this topic. <a href="http://support.automation.siemens.com/WW/view/en/15390415">http://support.automation.siemens.com/WW/view/en/15390415</a>	For the installation regulations, please refer to the manual.
2.	Install your used xP 270 / 370 operator panel according to the installation regulations. The following FAQ provides more information on this topic (e.g. MP370 manual). <a href="http://support.automation.siemens.com/WW/view/en/19106667">http://support.automation.siemens.com/WW/view/en/19106667</a>	Pin assignment / power supply according to manual
3.	Use a PROFIBUS cable to connect the PROFIBUS interface of the S7 controller to the PROFIBUS interface of your operator panel.	



**Note** In general, the installation guidelines for have to be observed.

## Installation of the standard software

Table 6-2

No.	Instruction	Comment
1.	<p>It is required that the software specified in chapter 2.3 table 2-2 is installed on your PG/PC. If you use a PC or notebook as development system, it is required that it is equipped with a communications processor (e.g. CP5611 / CP5512 PC card for notebooks).</p> <p>Please observe the system requirements in any case.</p> <p>For more information, please refer to the Customer Support pages on the internet.</p>	

## 6.2 Installation of the application software

Table 6-3

No.	Instruction	Comment
1.	<p>Transfer the STEP 7 configuration to your S7 controller.</p> <p>For information on how to transfer a STEP 7 configuration to a controller, please refer to the information provided on the Customer Support pages on the internet.</p>	<p>Link to the Customer Support pages.</p> <p><a href="http://support.automation.siemens.com">http://support.automation.siemens.com</a></p>
2.	<p>Transfer the WinCC flexible configuration to your operator panel.</p> <p>For information on how to transfer a configuration to an operator panel, please refer to the corresponding manual.</p>	<p>Link to the Customer Support pages.</p> <p><a href="http://support.automation.siemens.com">http://support.automation.siemens.com</a></p>

## 7 Operation of the Application

### You are provided with information on...

how to operate the functions of this application.

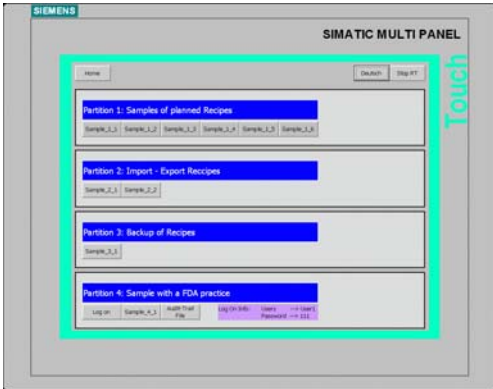
### Overview

The most essential functions have already been described in the previous chapters.

It is required that you have already transferred the STEP 7 configuration and the WinCC flexible configuration to the corresponding hardware and that a connection exists between S7 controller and operator panel.

### Selecting configured examples

Table 7-1

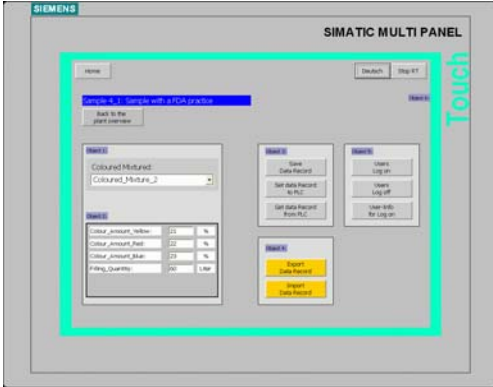
No.	Instruction	Comment
1.	<p>After starting Runtime on the operator panel, the following start screen is displayed.</p> <p><b>Buttons in the permanent window:</b>            (Buttons visible on all screens)</p> <ul style="list-style-type: none"> <li>• <b>“Home” button</b>                Use the “Home” to return to the start screen.</li> <li>• <b>“English (Deutsch)” button</b>                This button is used to change the language. The project is configured in “German” and in “English”.</li> <li>• <b>“Stop RT” button</b>                Use this button to stop the Runtime of the operator panel.</li> </ul>	

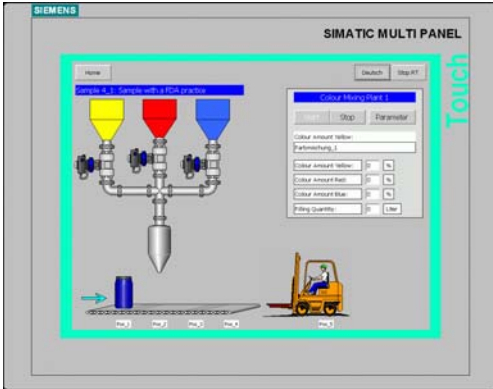
No.	Instruction	Comment
2.	<p>The configuration examples are divided into four partitions. Use the corresponding buttons to display the individual program examples.</p> <p><b>Characteristic features:</b></p> <ul style="list-style-type: none"> <li>Partition 1: If a button in partition 1 is selected, the button color changes to green. This helps you better see the last page you were on.</li> <li>Partition 4: An FDA function is integrated in sample 4_1. It requires the logon of a user. If no user is logged on, a system message is displayed when calling the example. This can be prevented by a previous logon using the "Log on" button.</li> </ul>	

## Partition 4, selecting sample\_4\_1

Table 7-2

No.	Instruction	Comment
1.	<p>Before calling sample 4_1, please use the "Log on" button to log on with one of the following user names.</p> <ul style="list-style-type: none"> <li>User1 111</li> <li>User2 222</li> <li>User3 333</li> </ul>	

No.	Instruction	Comment
2.	<p><b>Starting the process:</b></p> <p>To be able to start the process,</p> <ul style="list-style-type: none"> <li>• a data record must always be in the controller. (It is required that values have been entered for the parameters).</li> <li>• a user must be logged on.</li> <li>• If the process was interrupted and if the current data record has been changed in the meantime or if another data record has been selected, this data record first has to be transferred to the controller before the process can be restarted.</li> <li>• If the process was interrupted, the process can only be restarted after the barrels have been automatically transported by one position.</li> </ul> <p>If necessary, use the "Parameter" button to call the page for parameterizing the plant.</p>	
3.	<p><b>Stopping / interrupting the process:</b></p> <p>Use the "Stop" button to stop / interrupt the process.</p> <p><b>Note:</b></p> <p>The process is not immediately stopped with the Stop command. The last step to be processed is completed (e.g. transport of a barrel).</p>	

No.	Instruction	Comment
4.	<p><b>Selecting data record:</b> You can select between three different data records via object 1.</p> <p><b>Entering / changing values:</b> You can edit the preset data record values via object 2.</p> <p><b>Transferring data:</b> You can execute the functions shown on the buttons via object 3.</p> <p><b>Export / Import:</b> You can export the recipe data records to a storage card and import them from the card via object 4.</p> <p><b>User logon:</b> You can log on as "user" via object 5. A logon is required when you want to execute the functions in objects 2 and 3. At first, changes within objects 1 and 2 do not affect the data in the controller. A message box is displayed, which indicates that changed data have not yet been updated with the controller.</p>	 <p>The screenshot displays the SIMATIC MULTI PANEL interface. On the left, there is a 3D model of a mixing station with three colored hoppers (yellow, red, blue) and a central mixing nozzle. On the right, there is a control panel titled 'Color Management 1' with several input fields for 'Color amount yellow', 'Color amount red', and 'Color amount blue', each with a numeric keypad. Below these are buttons for 'Stop' and 'Transfer'. At the bottom of the interface, there are icons for a storage card and a forklift. The word 'Touch' is written vertically on the right side of the panel.</p>

## Appendix and Literature

### 8 Additional Notes

#### 8.1 Import / export of recipes

The “Sm@rt Service ” option provides an additional possibility to import or export recipes.

Sm@rt Service enables remote maintenance of operator panels for service purposes via the internet.

The service technician starts Internet Explorer on the service PC and enters the IP address of the operator panel to connect to the operator panel home page.

From the start page the service technician can call the HTML page for the import / export of recipes.

For detailed information on the installation / operation of “Sm@rt Service”, please refer to the WinCC flexible online help or, for example, to the “Getting Started Options” manual.

#### 8.2 Evaluating audit trails / integrity check

All configured FDA / GMP-relevant processes are saved in an audit trail file in CSV format.

Each entry in this file has a separate checksum.

The checksum can be used to determine whether entries have subsequently been changed / manipulated.

Using an “Audit Viewer” you can determine whether an “audit trail file” has subsequently been changed / manipulated or not.

An “Audit Viewer” is available on CD2 of WinCC flexible 2005. (CD2 > Support > Audit Viewer).

In addition, you can download the Audit Viewer from the internet at [www.siemens.de/wincc-flexible-audit](http://www.siemens.de/wincc-flexible-audit).

Detailed information on “Audit Viewer 2005” is available in the “Audit Viewer 2005” online help.

The “WinCC flexible 2005 Getting Started Options” manual describes an additional option for evaluating the audit trail file via a prompt on your configuration computer.

## 8.3 Showing / hiding objects via the “layers”

Showing or hiding objects via the “layers” **only** has effects in the development environment view.

If objects are to be shown or hidden in Runtime, the “Visibility” system function has to be activated in the properties of the corresponding object.

## 9 Literature

### 9.1 Bibliographic references

This list is by no means complete and only provides a selection of appropriate sources.

Table 9-1

	Topic	Title
/1/	WinCC flexible 2005	User's manual WinCC flexible 2005 Compact / Standard / Advanced Order No. 6AV6691-1AB01-0AA0 <a href="http://support.automation.siemens.com/WW/view/en/18796010">http://support.automation.siemens.com/WW/view/en/18796010</a>
/2/	WinCC flexible 2005	Communication manual WinCC flexible 2005 Communication Part 1 Order No. 6AV6691-1CA01-0AA0 <a href="http://support.automation.siemens.com/WW/view/en/18797552">http://support.automation.siemens.com/WW/view/en/18797552</a>
/3/	WinCC flexible 2005	WinCC flexible Getting Started First Time User <a href="http://support.automation.siemens.com/WW/view/en/18660846">http://support.automation.siemens.com/WW/view/en/18660846</a>
/4/	WinCC flexible 2005	WinCC flexible Getting Started Options <a href="http://support.automation.siemens.com/WW/view/en/18657078">http://support.automation.siemens.com/WW/view/en/18657078</a>
/5/	WinCC flexible 2005	MP370 manual Order No. 6AV6691-1DE01-0AA0 <a href="http://support.automation.siemens.com/WW/view/en/19106667">http://support.automation.siemens.com/WW/view/en/19106667</a>
/6/	STEP7	Automating with STEP7 in STL and SCL Hans Berger Publicis MCD Verlag ISBN 3-89578-113-4



## 9.2 Internet links

This list is by no means complete and only provides a selection of appropriate sources.

	Topic	Title
\1\	Reference to the entry	<a href="http://support.automation.siemens.com/WW/view/en/23901413">http://support.automation.siemens.com/WW/view/en/23901413</a>
\2\	Siemens A&D Customer Support	<a href="http://www.ad.siemens.de/support">http://www.ad.siemens.de/support</a>

## 10 History

Table 10-1 History

Version	Date	Modification
V1.0	09/01/06	First edition