TM800

APROL System Concept

PERFECTION IN AUTOMATION
www.br-automation.com

Prerequisites and requirements	
Training modules	TM213 – Automation Runtime
Software	APROL Automation Runtime SuSE LINUX
Hardware	1 control computer, 1 controller

Table of contents

1 Introduction

You have chosen to take B&R's APROL System Concept training module to learn about the components of this system, to get to know how to configure your own system with APROL, and to become familiar with the many different ways this process control system can be used.

Some of you may just want to learn about what types of configuration and expansion possibilities exist; others of you might actually be working with the APROL process control system and configuring your first systems after this seminar is over.

We will do our very best to provide each and every one of you with as much information as possible to meet your requirements.

Nevertheless, the technical expertise of your seminar leader is not the only factor contributing to your personal success during this seminar. Success depends on cooperation and interaction between course members and the seminar leader, as well as the attitude of individual participants towards teamwork and the course in general.

Figure 1: First time engineering

1.1 Learning objectives

The APROL System Concept seminar is meant to provide you with insight into the performance of this system while bringing you closer to how it actually works. Once you have finished this seminar, you will be in a position to make the best possible use of the advantages provided by APROL, including its complete and intuitive engineering concept.

Our goal is clear: Once you have completed this seminar, we want you to be able to solve common problems that occur in the field of automation using the standard system components and library function blocks included with this system.

This training documentation includes numerous screenshots that will help you to get a much better idea of your own system.

This seminar will also show how a project can be properly managed. This refers to the delegation of responsibilities for project engineering as well as for structuring the system. System administration can now be managed at all times.

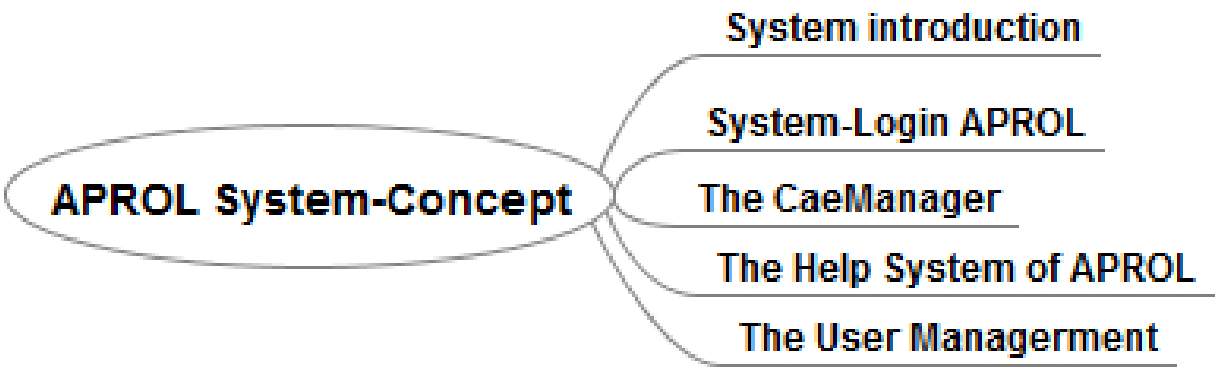


Figure 2: Overview

2 System introduction

2.1 The philosophy

Our goal with APROL is to provide a process control system that achieves the highest levels of performance.

What is a process control system?

A process control system generally refers to several controllers, usually B&R systems, connected to many different kinds of computer systems (control computers). These computers carry out a wide variety of tasks. For example, they can monitor all data which is important for executing an automated task (process) and use databases to collect it.

Other computers provide the user with graphical user interfaces that can be used to monitor running processes and analyze all of the collected data (operator station).

The connection to the controllers is established via one or more process buses.

In the past, every component in a process control system had to be programmed separately. This meant that there had to be different programmers for the control level, the graphical user interface, the databases, and the process bus system. The result is a SCADA system (Supervisory Control and Data Acquisition). The effort and expense necessary to make changes or correct any errors can easily be imagined. Furthermore, several experts (who hopefully communicate well with each other) are needed in order to successfully implement a complete system.

It was our goal to rid ourselves of this "separation of powers" and provide the user with a complete configuration interface that can be operated on all levels.

With its CaeManager, APROL provides exactly this type of configuration interface. Using function charts, you are in a position to configure and program both the control computer (with alarms, trending, visualization, etc.) as well as the controllers themselves. A connection (line) between a block level control computer and a block level controller automatically represents a communication variable. There is therefore a configuration file that is automatically created to handle communication; this prevents the possibility of oversights.

As an additional feature, APROL also makes it possible to incorporate the control computer into different controller and closed loop control tasks. This is done by simply changing the Physical View (level on which a block is processed) of a block.

The following symbolic representation shows the areas of process control engineering that are covered by the APROL CaeManager:

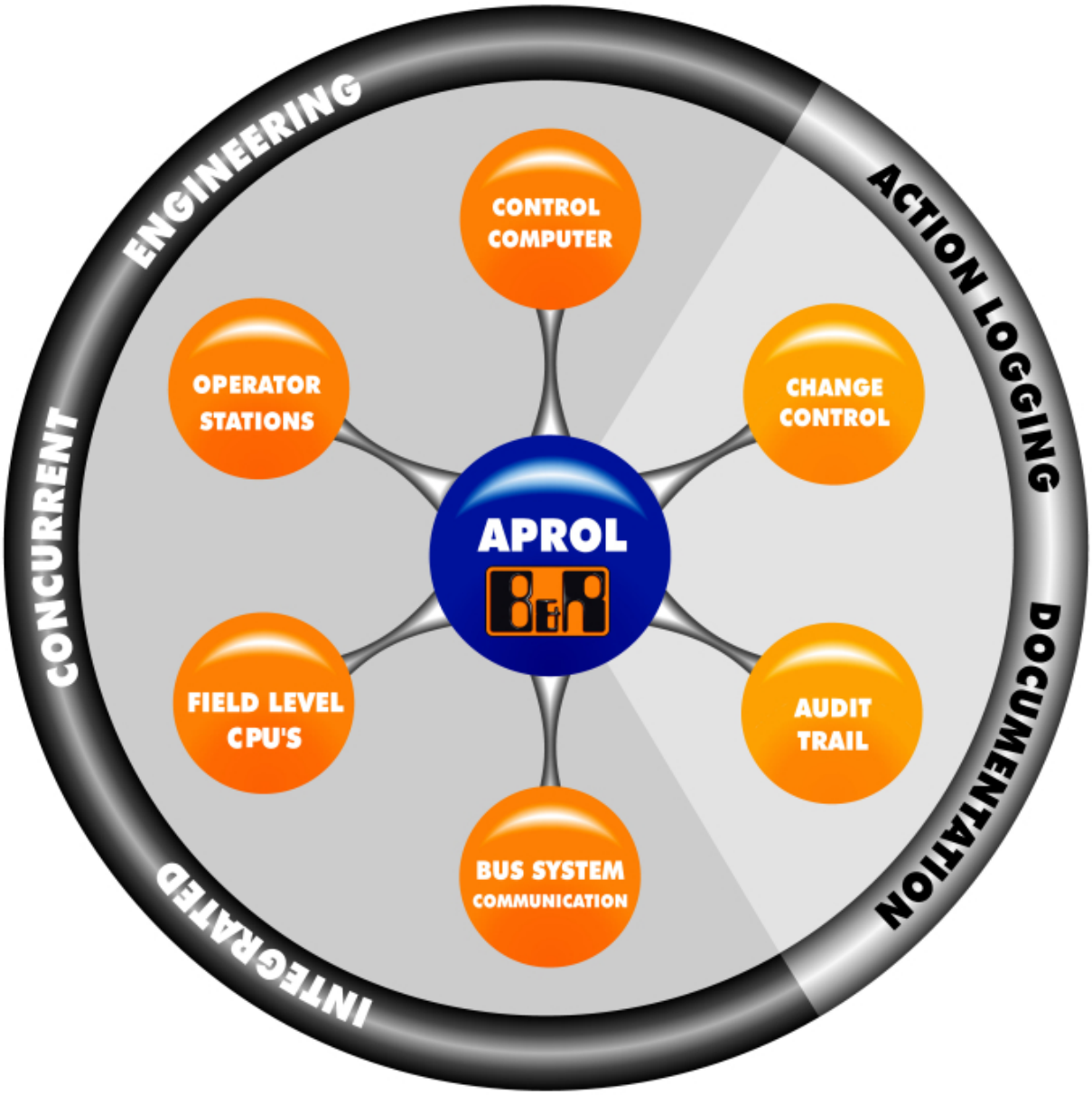


Figure 3: The APROL philosophy

2.2 The process control system components

APROL is a **system** for process control that runs on the Linux operating system. All APROL programs exchange their data using the TCP/IP communication protocol. Data is then exchanged between the programs via the Ethernet TCP/IP network that links the computers together.

Programs can be divided up between user interface programs and programs that run in the background. As the name indicates, user interface programs (or applications) are programs which use input fields, buttons, and other elements on the screen for entering data and triggering actions. As a general rule, background programs, or daemons, do not output messages to the screen. Instead, in each system they handle fixed tasks which are read from configuration files when started (drivers, etc.).

The **Engineering Server** is used to completely configure the process control system down to the control level. The **Engineering Server** handles the complete distribution of the system (downloading tasks to the controller, downloading the **Operator System**, downloading the **Runtime System** (background processes)).

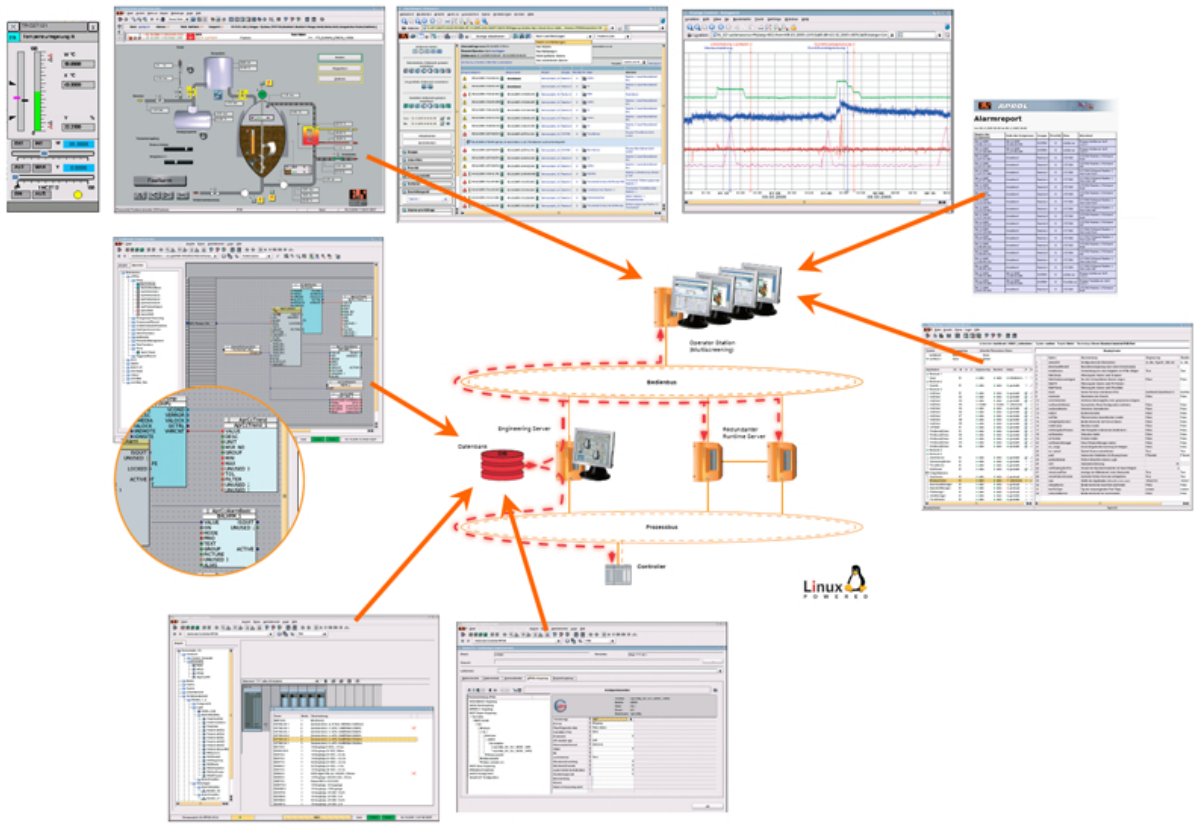


Figure 4: The APROL philosophy

The **Runtime System** uses data from the **Engineering System** and is basically the heart of the process control system. This system monitors, collects and distributes the process data configured in the **Engineering System** so that it can be processed further.

The **Operator System** forms the interface between man and machine. These programs are used to observe the process on one or more operator station monitors and can be operated with a mouse, keyboard or touch screen. The number of operator stations is limited by the network topology (TCP/IP network, netmask, etc).

The **Engineering, Runtime**, and **Operator Systems** combined make up the APROL system.

The **Engineering System** as well as the **Runtime System** and **Operator System** can be installed and used either individually or together on the same computer. Because standard hardware components are used and extend compatibility across all performance classes, we are able to implement a scalable system that has no problem handling increasing requirements as they arise.

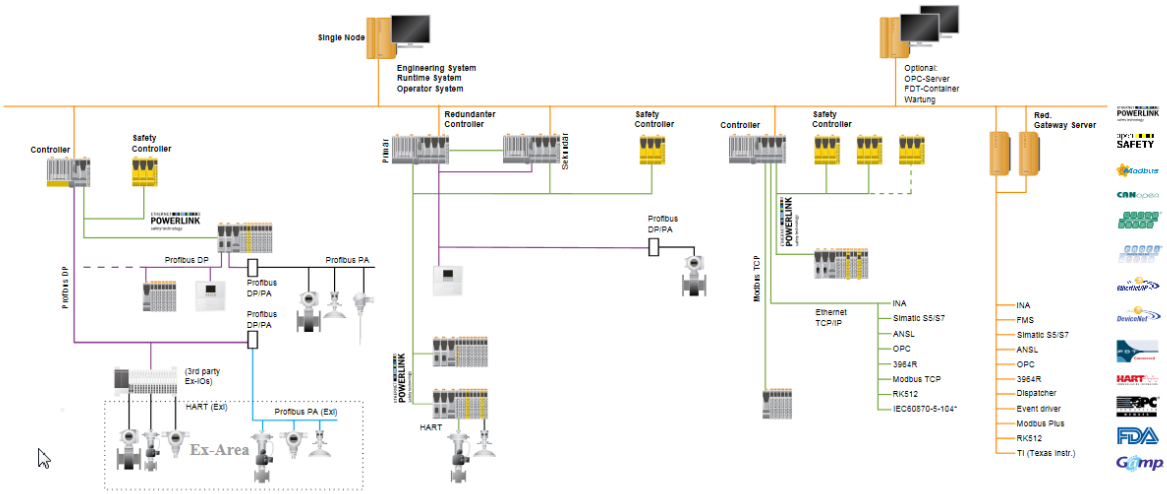


Figure 5: Scalability

Note: Redundant control computers

A computer is used as engineering server. Normally, only one runtime system is installed. However, it can be designed as a redundant system (hot standby, 2 control computers).

Every **operator station** has an engineering client which can work with the central **Engineering System** database. This functionality is referred to as "**concurrent engineering**" and allows several project designers to work with the **Engineering System** database at the same time. Parts of a project that are being worked on by others can still be opened in read-only mode. This prevents the problem of complicated data merging later on.

Note: Multiple graphical interfaces

The number of project designers is limited by the number of operating licenses. The Engineering environment can be accessed via VNC (virtual network computing, LINUX Program: krdc) or via the LINUX X-Window system.

(Possible command from console: (X :1 –ac –query IP address or computer name –dpi 75) (1 for first session)

Frequently, engineers work on a "Windows Host System" using VNC Viewer. (any computer on the network can therefore serve as an engineering client)

X programs such as "xming" provide additional options.

The information in the central engineering database is used to generate all of the user programs and to create the runtime database.

2.3 Bus systems

A wide variety of bus systems are used to connect the individual control system components.

The computers where the Runtime, Operator and Engineering System process control system applications run are connected to each other via the **system or operator bus**. The **system bus** is designed as an Ethernet TCP/IP network, which makes it possible to replicate all of the possible structures in this particular field.

Note: System bus

As a rule, the system bus is designed as a self-contained Ethernet network in order to guarantee security. A gateway server is generally used for exchanging data with other company departments. For the network structure, switches or hubs are needed to combine line and star-formed topologies.

Standard network components are used in the computers.

The **process bus** is available for connecting process control system components to the controllers. This connection is also made via **Ethernet**.

Up to 254 stations can be connected to the **Ethernet** network in an APROL configuration (netmask 255.255.255.0); the transfer rate for Intel controllers (SG4) can reach up to 100 Mbit/s. The network can be equipped with hubs or switches to achieve conventional structural network configurations. Because of this, it is important to use the settings specified by the respective IT department.

Note: Process bus physical properties

The trend has recently been more towards using Ethernet as the process bus due to its flexibility and better performance. However, it is still possible to use a serial link.

Fieldbus systems such as POWERLINK, PROFIBUS DP, Modbus, serial links, etc. can be used to connect transmitters, external hardware, or remote I/O systems. Explaining how to configure these types of connections would far exceed the bounds of this system introduction. Instead, they will be handled in later seminars.

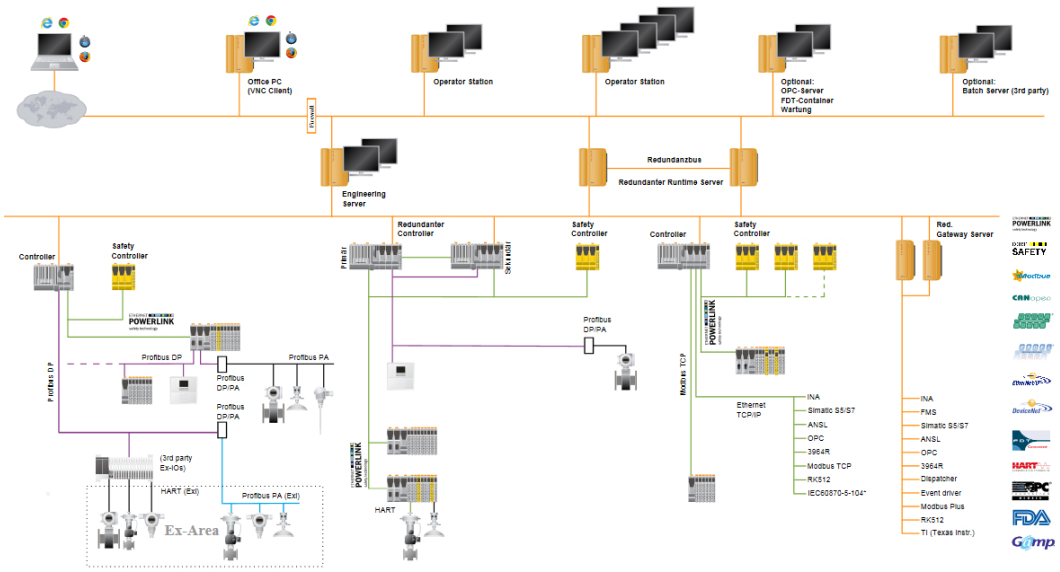


Figure 6: Bus systems

2.4 Multi Runtime System (MRS) concept

A **Multi Runtime System (MRS)** consists of multiple logical RUNTIME systems (see CaeManager "APROL system"), that are created and managed in the same project. Typically, these RUNTIME systems run on different hardware (servers), although this is not mandatory.

The reason for doing so is to create **automation islands** (subsystems), which are nevertheless able to exchange data with other **automation islands**. The project engineer does not need to worry about which variables need to be exchanged, since the system detects this automatically. This **decentralizes** configuration and downloading as well. (makes it possible to commission (modify) the subsystems separately). This automatically **improves the performance** of the Engineering and Runtime systems and **distributes the load**.

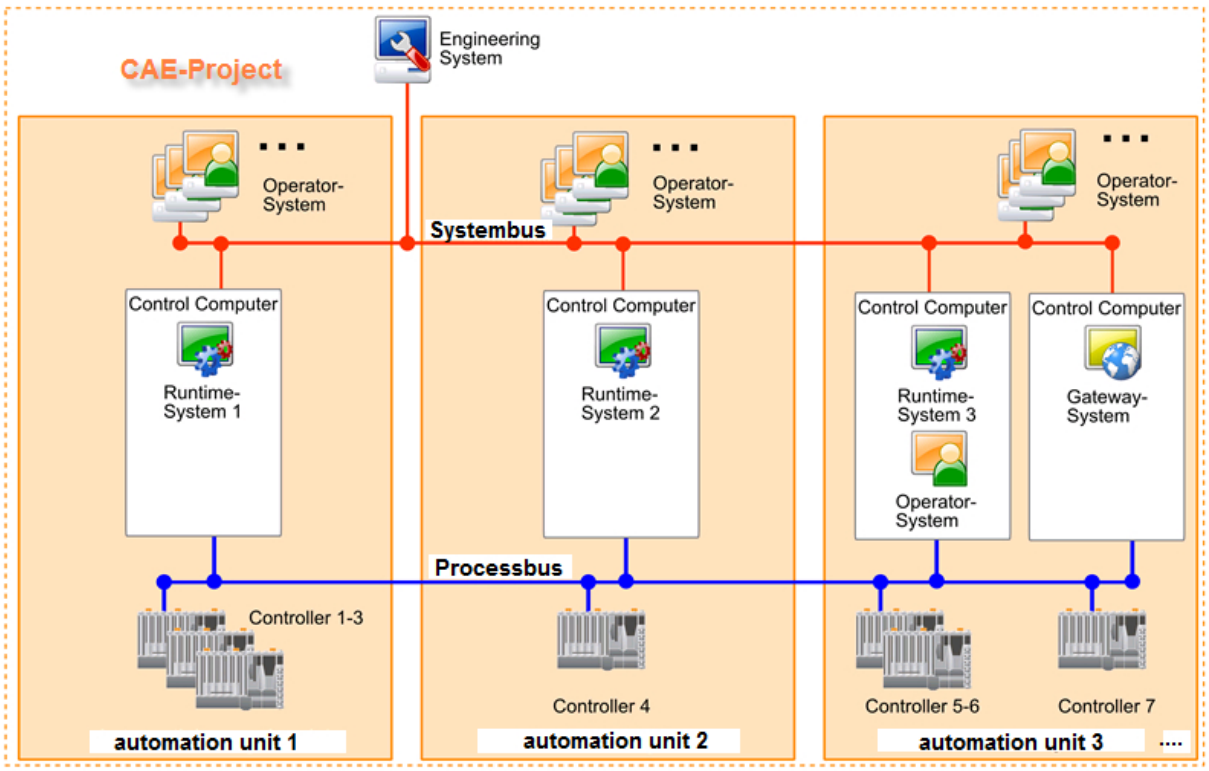


Figure 7: MRS system concept

By implementing a **"GLOBAL CLUSTER"** and routing all data to it, it is now easy perform queries (Alarm, Trend, etc.) across multiple automation islands. It is also possible to set up all or some of the subsystems as redundant.

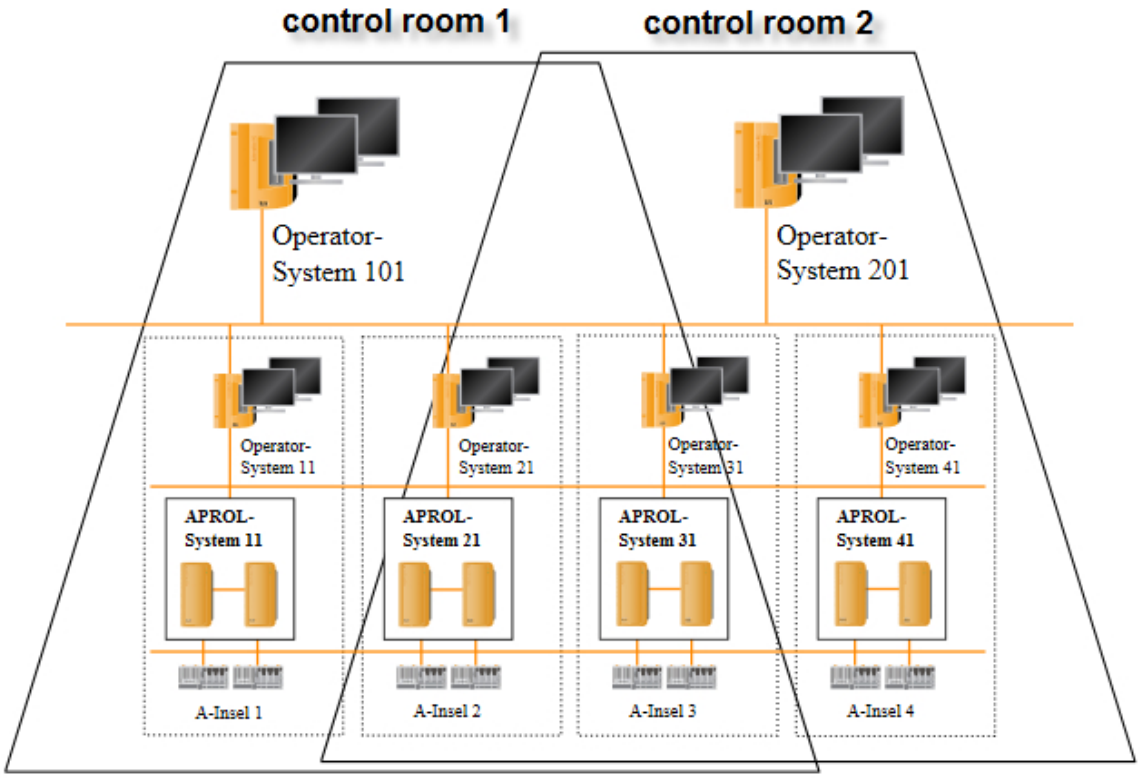


Figure 8: Plantwide constellation

Additional information can be found in the APROL product documentation.

2.5 Redundant controllers

In systems where high-availability is required, redundancy must be extended to the controller level. This involves using redundant controllers with a redundant process bus and a redundant fieldbus (POWERLINK).

One is the primary controller and the other is the secondary controller, and at any given time one of them is active while the other is in standby.

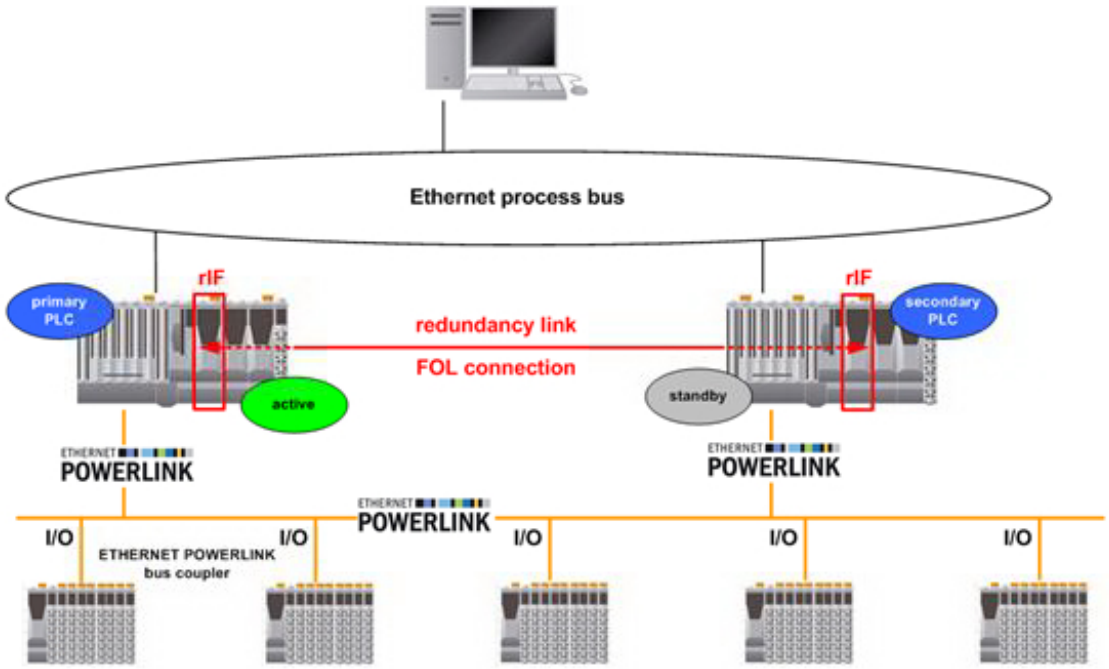


Figure 9: The concept of redundant controllers

Advantages of using redundant controllers:

Prevents downtime due to hardware failure.

Hot standby ensures immediate failover in the event of a "single point of failure".

Makes it easy to replace controllers during operation.

Redundant fieldbus networks (separately routed cables) to prevent I/O failures.

Redundant process bus for reliable communication with process control system.

.....

Additional information can be found in the APROL product documentation.

3 APROL system login

3.1 Logging on to the engineering system

At your training desk, you should find a control computer which has been set up with the Linux operating system. In addition, APROL has been installed with the Engineering System (**login "engin" and and password ".engin"**) and the Runtime/Operator System (**login "runtime" and password ".runtime"**).

Note: FAT, etc.

This nearly corresponds with how APROL is delivered. B&R delivers completely installed systems which have undergone testing before delivery. This testing checks to make sure that all components and their functions on the system are working. The only thing that remains to be done is configure the system (Operator, RUNTIME, Engineering, etc.). If desired, a FAT (Factory Acceptance Test) can also be carried out at B&R.

The Linux operating system is a multi-user and multitasking-capable operating system known for its high degree of stability. Another considerable advantage is that it can manage several graphical user interfaces.

The **Engineering and Runtime/Operator systems** can be likened to **independent Linux users** who start in separate directory structures with adapted environments. All necessary process control system components for the user are preinstalled at the factory.

If a FAT is requested, the entire system including servers, operator stations and controllers can be put together and pre-configured. A project version can also be "broken in" if one is provided.

The following image appears after the computer finishes booting (example):

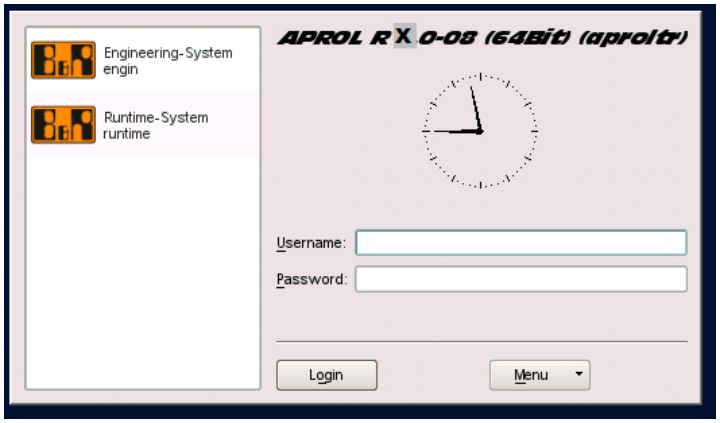


Figure 10: System login

APROL system login

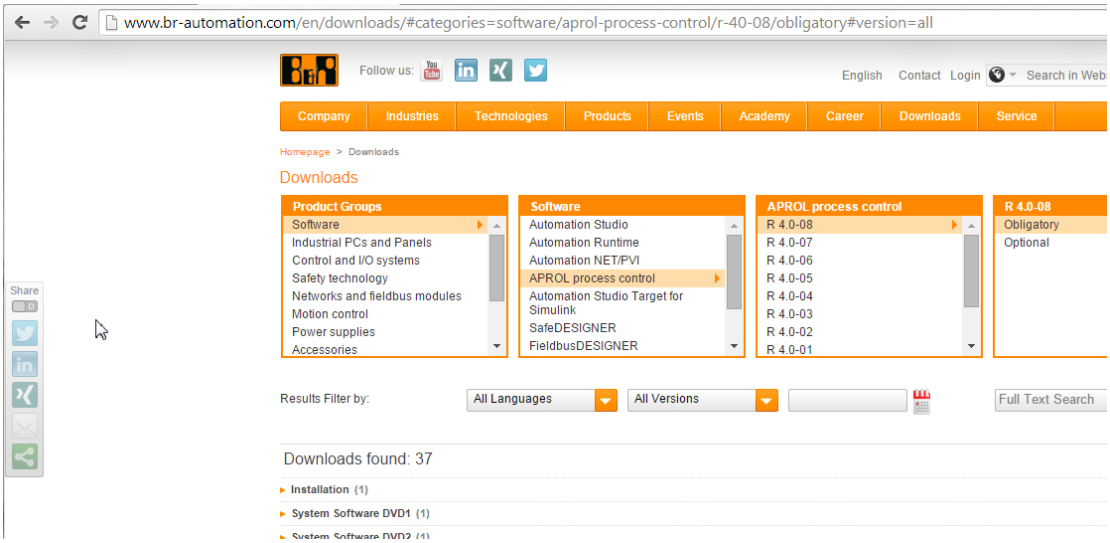
You identify yourself on the system by logging in as either an "engin" or "runtime" user.

Note: LINUX login

The login makes sure that the system is protected from unauthorized access from the very beginning. Unlike other systems, however, Linux is able to graphically represent several users at the same time.

Note: Installation

Just in case, installation media are available for each computer. This includes both the Linux operating system, which contains the necessary packages for use with APROL, as well as the APROL system software itself. The user is guided through the installation by a series of dialog boxes. The installation files (ISO format) are also available on the B&R website.



If an AutoLogin (automatic login into a graphical environment) is required, this can easily be implemented using standard Linux functions. This would mean that the desired environment is displayed right away when the server is started.

3.2 Window manager

KDE (K Desktop Environment) is used as the window manager not only because it is similar to the look and feel of Windows, but because it is a very comfortable interface for helping users get their work done.

The window manager is configured in such a way that all applications needed in the Engineering and Runtime systems can be started.

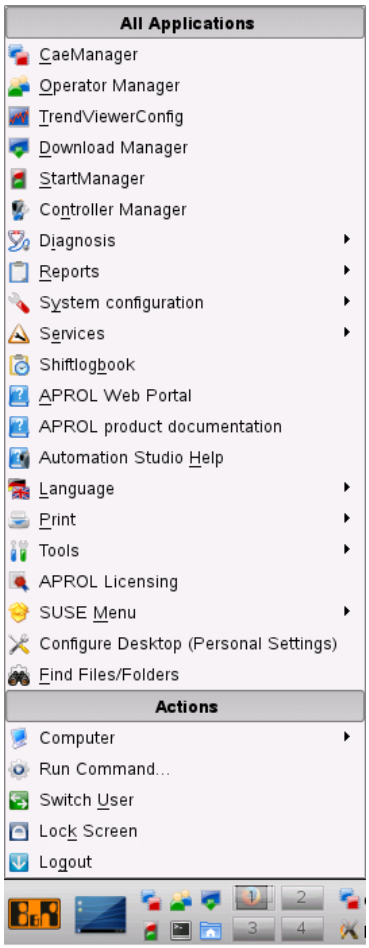


Figure 11: Window manager

The most important application for configuring systems with APROL is **CaeManager**. As is familiar with most Windows applications, it can be started using from the window manager's "Start" menu.

Note: KDE menu

All important applications for operating APROL are located under the KDE menu and the respective submenus. The other menus contain LINUX-specific programs like the ones in Windows environments.

3.3 Login mechanism

The Engineering System contains an integrated user management system. The **LoginServer** makes it possible to access all of the other applications available to the current user. When the system is delivered, only one user is present who has all rights of an administrator. This **user** is called **"Startup"** and has the **password "admin"**. To be able to start an application, it is necessary for at least one valid user to login. The user should change the password immediately after logging in for the first time. (See User Management in APROL.)

Note: Login concept

The advantage of the LoginServer is that a user only has to login once, and all applications have access to this information. This centralization saves time since the user doesn't have to re-identify himself to every single application.

The following dialog box appears when starting CaeManager for the first time:

Figure 12: APROL login

This dialog box is always opened whenever a program belonging to APROL is started before the LoginServer is running.

Exercises:

Exercise 1:

The Engineering System user interface should be started with user "engin" and password ".engin". You may use a combination of the mouse and keyboard, or just the keyboard alone to enter the information.

Play around with the LINUX environment and just try starting some of the many programs.

Exercise 2:

CaeManager should be started by each course participant.

Exercise 3:

Starting CaeManger automatically opens the LoginServer, which then continues to run in the background.

Log in to the system using "aprol" as the login name and "BuR1.aprol" as the password. This login has been created in advance; it is not a default APROL login.

Navigate through CaeManager.

Functionality of the LoginServer

The LoginServer is started and then runs continually in the background as soon as the very first application that wants information about the user is opened. The currently logged in user is shown in the window manager menu as a "tooltip." Place the cursor over the LoginServer icon for a second or two to display the username.

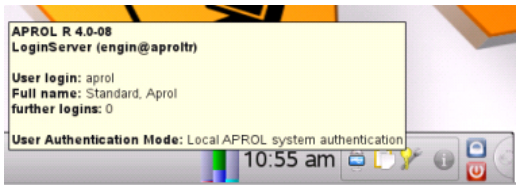


Figure 13: LoginServer

There are several possibilities to find out more information about the current login status. Right-clicking on the LoginServer icon will display the following menu.

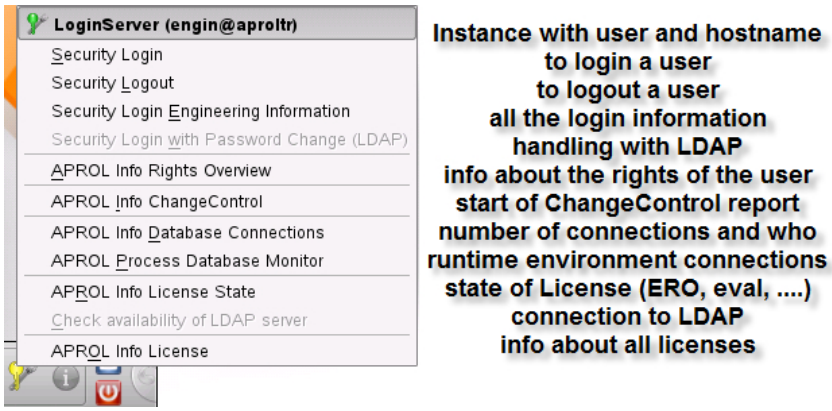


Figure 14: Login management

These commands can be used to control the login process and access to the Engineering environment and retrieve information which and how many users are logged in, info about the current user, opened reports, licensing status, etc.).

Clicking on the "**Security Login Information**" menu item opens the following window, which provides more information about the user who is currently logged in.

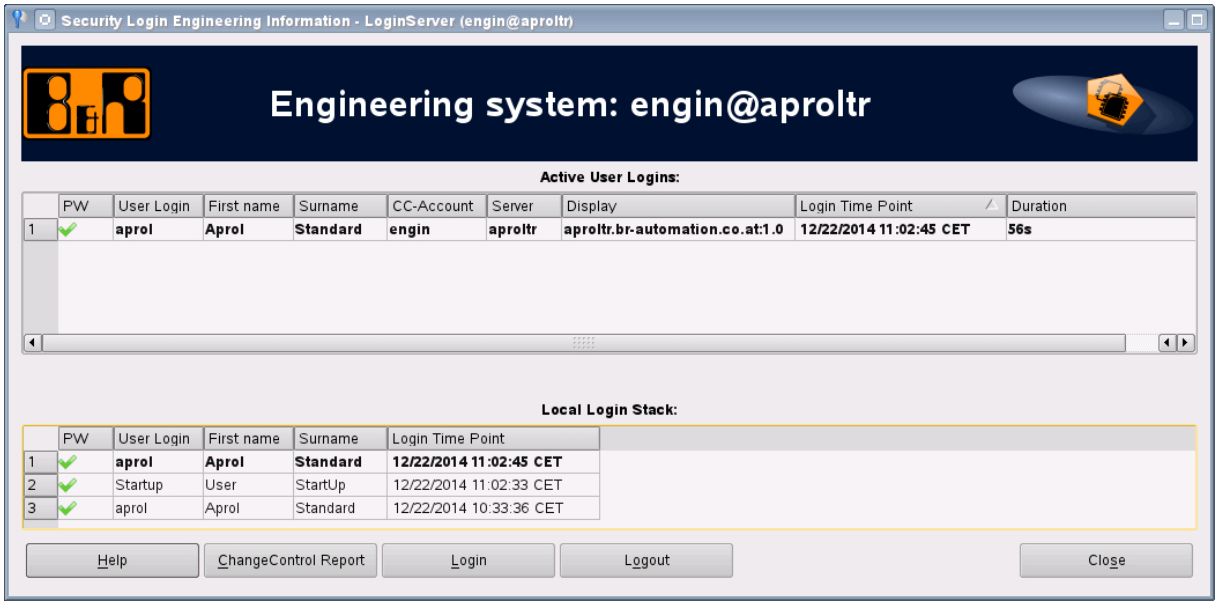


Figure 15: Login info

Note: Login concept

It is possible for more than one user to be logged on at the same time. Although the number of users can increase (stacked users), only the rights of the current user are taken into consideration. Rights are not added to each other if several users are logged on. Users are logged out in the reverse order of when they logged in. It is not possible to select the user to be logged off.

Important: Passwordassignment

Please make sure that you are careful with passwords! It is very easy for someone to watch you enter your password while you are typing it on the keyboard. Any changes to the projects and databases are tagged with the active login.

3.3.1 Licensing mechanism

The licensing mechanism is also shown in the menu above. There are three types of licenses available:

- E - Engineering
- R - Runtime
- O - Operating

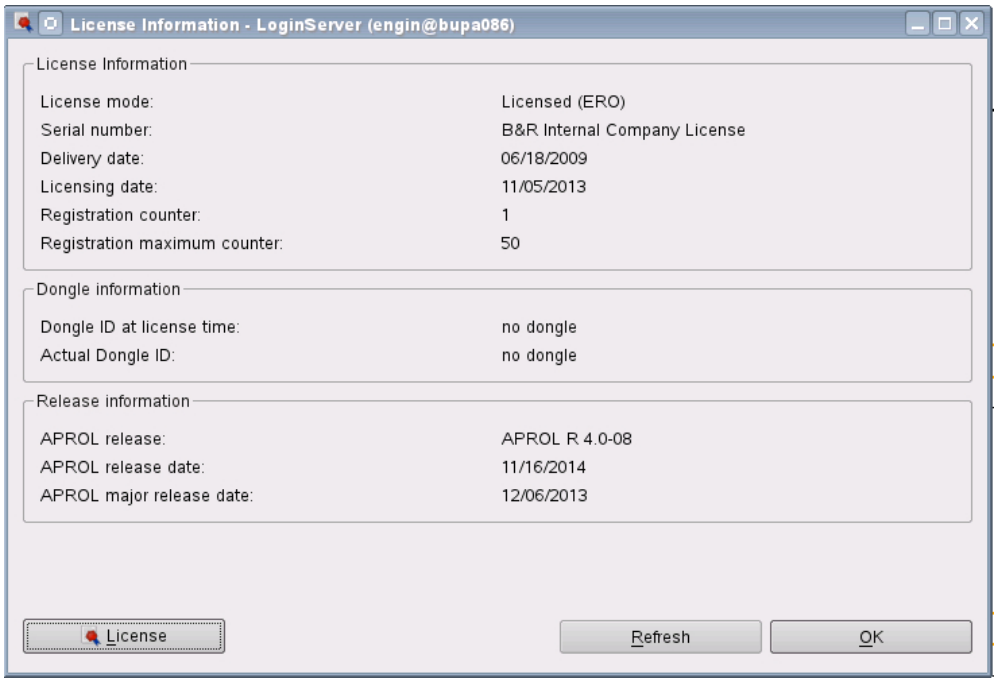


Figure 16: Licensing information

By default, the APROL system can be operated using the evaluation license after installation. This is valid for 90 days. Engineering is greatly hindered after the evaluation license has expired. However, a RUNTIME environment will continue to run for safety reasons (i.e. system is not shut down). In this case, the system triggers alarms and displays messages.

In addition, "I/O licenses" are also available. The quantity and usage thereof are determined by the sales department. Packages of 250, 1000, 2500 and 5000 I/Os are available. The same applies for gateway I/Os. These are defined via the extensive range of couplings available in APROL.

The various libraries, drivers and redundant controllers that require licenses are all managed via "Technology Guarding". Additional information can be found in the APROL product documentation.

4 CaeManager

APROL engineering can be divided into two main parts: general and project-specific. The general part is known as **basic engineering**. In the best case, basic engineering only has to be carried out once and is used to configure what follows below.

- The **User Management** system determines which company employees can work with APROL and which rights they have to work with the applications and implement the project.

4.1 General CaeManager functions

CaeManager is the central program of the Engineering System. It can be used to perform almost all engineering and configuration tasks necessary for creating a process control system application. **CaeManager** makes it possible to develop a process control system as a team.

It is a **structured configuration system** for both **concurrent engineering** and **offline engineering** based on **function block language conforming to IEC 1131-3**.

We will now take a look at the terms above:

Concurrent engineering

Engineering teams work **simultaneously** on the same project **without impeding each other** and without having to resort to complicated data exchange between one another. All data is stored in a single database. Access to this data is managed centrally by a database server which prevents competing access to the same data ("competitor" only granted read-only access). Only commissioning functions like **Compile, Build**, and **Download** may not overlap. If one of these processes is executed, then various other possible actions are blocked.

Structured configuration

Configuration data is divided into project-specific reusable parts. Function blocks from standard libraries and information for user management can be reused. Project-specific parts are stored locally in the projects (specifically for each respective task). Project-specific data include the hardware configuration, function charts, process graphics, etc.

Function block language conforming to IEC 1131-3

It is not necessary to have knowledge of specific programming languages to configure the process control system. Interconnecting function blocks with a graphical editor is sufficient to configure both the logic of the control level and the control system components (alarm system, graphic display of system data, etc.).

4.2 General information for operating CaeManager

After starting CaeManager and logging in as a valid user, the following window appears by default on the desktop:

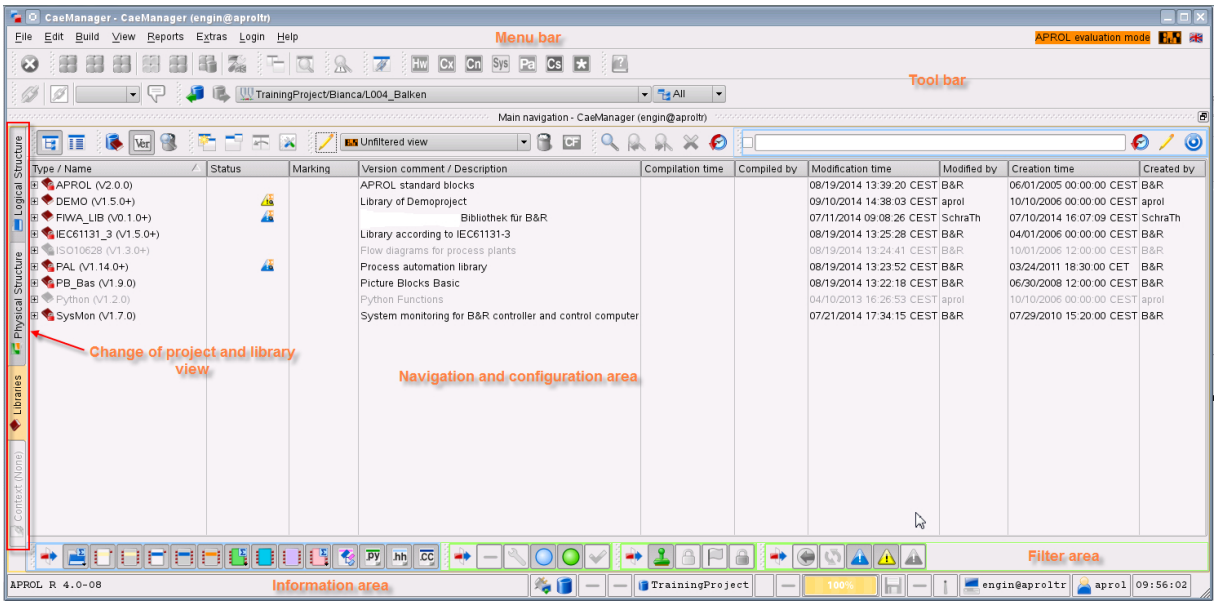


Figure 17: CaeManager

Menu area:

All engineering and configuration work can be done using these menus. In addition, there are a few shortcut buttons that handle file operations and editing. The icons are from the Windows world and should be self-explanatory.

Project / Library selection area:

Here you can select the area (in different views) where editing will take place. With a project, this comprises both the **logical structure** and the **physical structure**. The logical structure is much like an Explorer view with the corresponding directories. The physical structure displays the project's hardware view. This can be very helpful later (moving function charts or tasks, etc.). User libraries can be created under **Library**. The **Context view** opens the objects required in a respective configuration element.

Navigation and configuration area

The structure of engineering data you create is displayed here in the form of a directory structure. You are able to navigate inside of your engineering data while editing it and work with the various elements presented here.

This work area is used to generate and edit objects necessary for creating your process control application. These include function charts, process graphics, controllers, control computers, etc.

If you haven't clicked on a particular process control system object to edit it, then this area will contain the contents of the directory selected on the left. Double-clicking on an object opens it for editing.

The following objects can be created and edited in the configuration area.

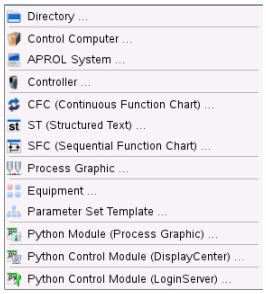


Figure 18: Objects to be created

- **Directory** — Directory for an additional level in the project tree.
- **Control computers** — The hardware systems at the process control level, including: runtime server, operator station, gateway server etc.
- **APROL system** — Configuration for the corresponding hardware (RUNTIME system with background processes (drivers), OPERATOR system, etc.)
- **Controller** — The hardware systems of the controller components (System X20 and 2005) such as a CPU with I/O modules, POWERLINK I/O devices, etc.
- **Continuous Function Chart** — Function charts for the controller as well as the runtime system or control computer. These may be INIT, CYCLIC, or EXIT function charts.
- **Structured Text** — Logic implemented in ST. (standardized high-level textual language)
- **Sequential Function Chart** — Allows the use of SFC to implement sequential processes (e.g. select statements, etc.).
- **Process graphic** — Process graphics for the operator stations and operator systems.
- **Equipment** — Equipment such as motors, pumps, valves, etc.
- **Parameter Set Template** — Parameter templates for the subsystems listed in this template (commonly: subsystems)
- **Python modules** — Different types of Python modules can be created to implement additional functions.

Filter area:

Several different filter settings can be made here in order to pinpoint the location of a desired object. The user can customize the arrangement of the filter areas in whatever way is most convenient. With the ability to create custom text filters and define links, it's easy to find everything, even in a project you didn't create yourself.

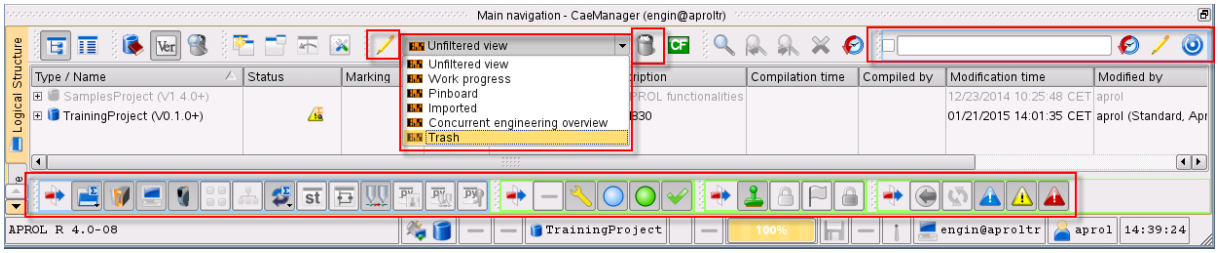


Figure 19: Various filter options

Information area:

This area shows information about the object name (version), editing mode, current user, date, and time. Hovering the cursor over a field displays the corresponding tooltip. Please use this functionality to become familiar with the functionality. Of course it is also possible to look it up in the product documentation as well.

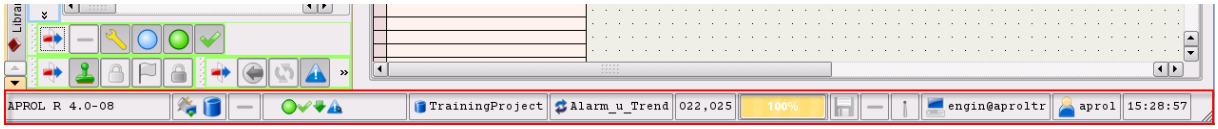


Figure 20: Information bar

5 CaeManager help documentation

All manuals currently available are installed as HTML files on the control computers. These documents can be accessed at any time with the "**Help / Product documentation**" menu item or from the KDE menus. Of course, an APROL installation is not necessary for the simple purpose of gaining an overview. Documentation can also be found in the APROL package (DVD case delivery). In addition, the documentation can be found on the **B&R website**.

A starting index page branches off to the individual manuals.

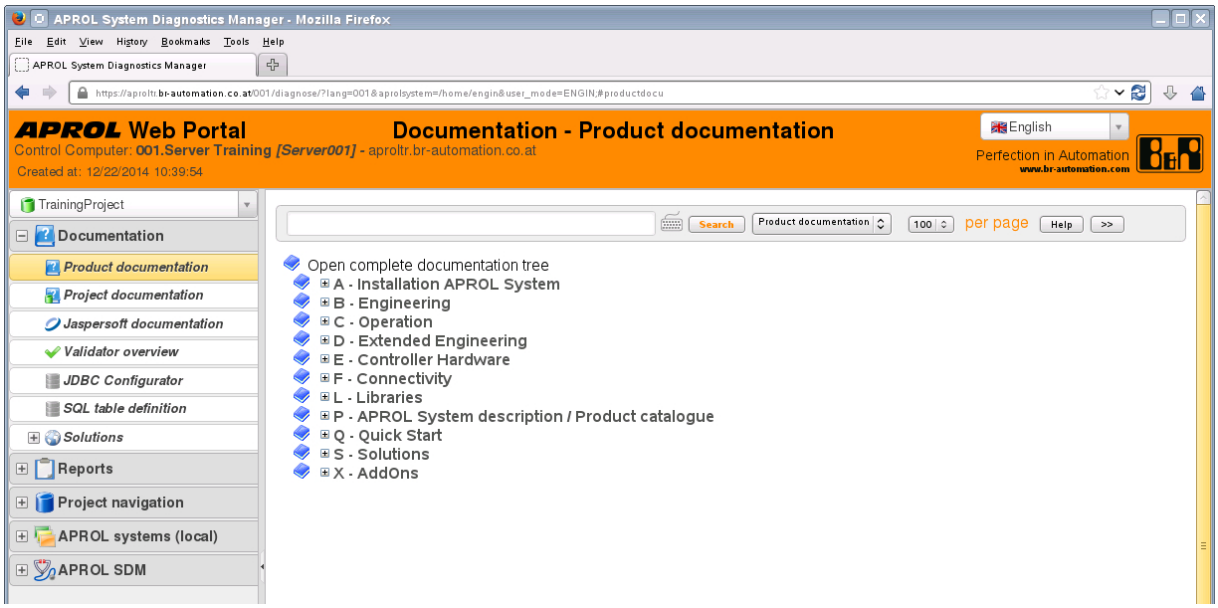


Figure 21: Help index page

A convenient search bar is implemented at the top. Feel free to use it whenever you feel that it is necessary. The **"Help"** button opens a detailed description of how to use the search tool.

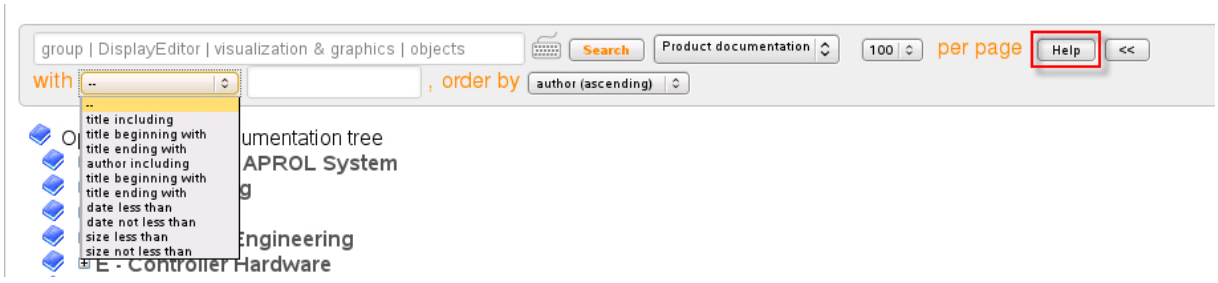


Figure 22: Details regarding the search tool

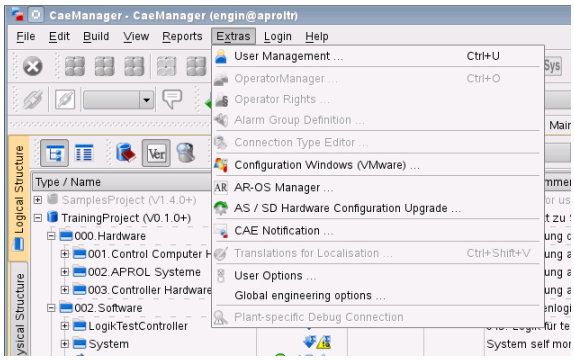
6 User Management in APROL

Users include any persons who have anything whatsoever to do with APROL. For example, these can be engineers, service technicians, or system operators (relatively uncommon in the engineering environment). The **User Management** system is used to create **templates** for different areas of responsibilities and qualifications. These **templates** are then assigned to different people. (drag and drop)

Note: How it works

It is only necessary to make changes here if new employees are hired by the company, employees leave the company, or employees should receive different rights due to the tasks they need to fulfill.

The **User Management** system can be opened using the "**Extras / User Management**" menu item.



The integrated **User Management** system is used to coordinate and divide the planned work while supporting the formation of a project team. This also ensures access protection for all objects in the Engineering system.

Note: Permissions

In the Engineering System, each user should "know everything" (have read access). However, many tasks can only be taken care of by employees that are actually qualified to do so. These employees then receive write authorization in their areas of responsibility. Other employees are responsible for the editing of the project itself (or individual parts thereof) and are granted the corresponding rights as a result.

If it is necessary to validate a system, there would be an extra group that would only have read access for all objects and the ability to confirm tested versions.

6.1 Templates (user profiles in general)

Templates affect every project since they determine the authorization levels for using Engineering System applications in APROL, therefore also affecting the engineering of each individual project.

A **template** defines certain access rights for the Engineering system's different functions. The user management system integrated in CaeManager is able to manage a large number of templates.

Note: Using templates

It would be pointless to create a separate profile for every subsequent user since this would virtually eliminate the advantage of the profile creation mechanism. Templates should be created skillfully so that planners (users) are assigned to individual templates in such a way that they receive the rights needed to carry out their work.

Note: Template assignments via drag and drop

The advantage of this procedure is that employees can be quickly assigned to one or more templates without having to assign every single user permission each time.

6.2 Creating a template

The following describes how to create a **template** in the **user management system** (right-click to open submenus):

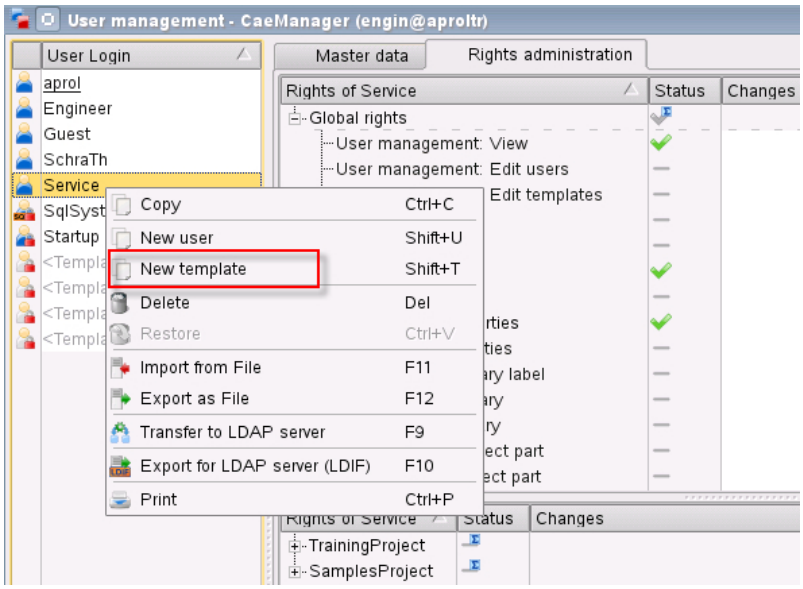


Figure 23: Creating a template

Select the "**New template**" menu item. This creates a template that can be edited and configured as needed.

Define a name. In order to structure work as effectively as possible, an **existing template can be assigned using drag and drop**.

The **"Help"** button immediately links to documentation regarding this procedure.

Figure 24: Naming the template

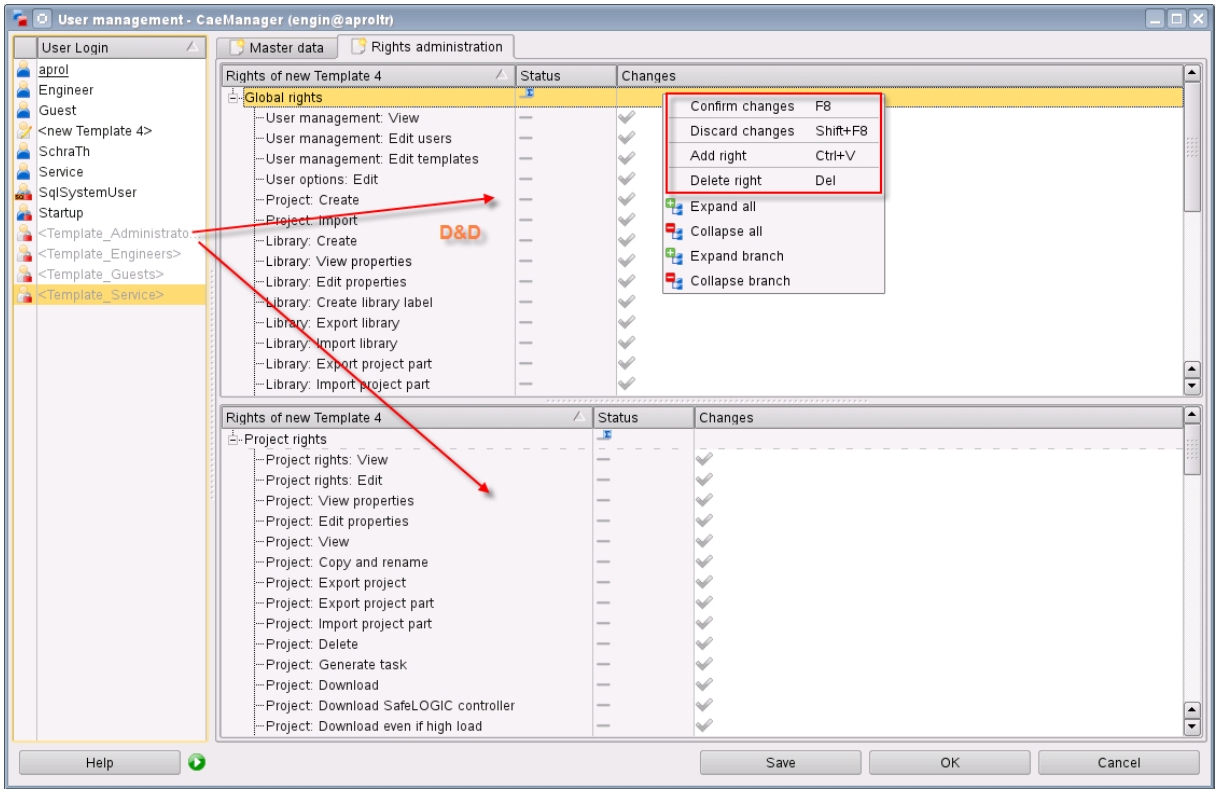


Figure 25: Editing the template

Note: Naming conventions

Names should/must be as clear and unique as possible in a system.

Operation:

In the submenu, which is opened by right-clicking, you have the ability to assign and confirm permissions. It is possible to make multiple selections or to work with a root in order to edit all the entries below it. Any changes must be saved when you are done.

Meaning of the individual permissions:

Individual permissions can now be granted in the template. There are two types of rights: **global rights** and **project rights**.

Global rights:

Global rights have to do with system operations that span the entire project and libraries.

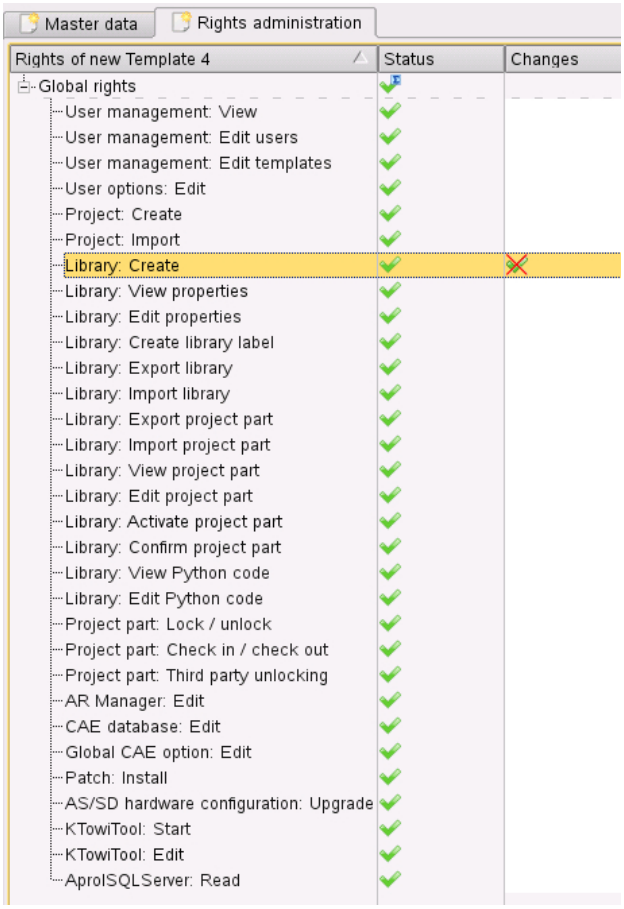


Figure 26: Global rights

As you can see in the screenshot, this area is used to manage access to project engineering, library engineering, user management and a few other activities. This generally involves permission to read, write or confirm. Access vie import and export mechanisms is also managed here. A detailed description of the individual permissions can be found in the APROL help documentation.

Project rights:

Project rights deal exclusively with operations connected to project engineering.

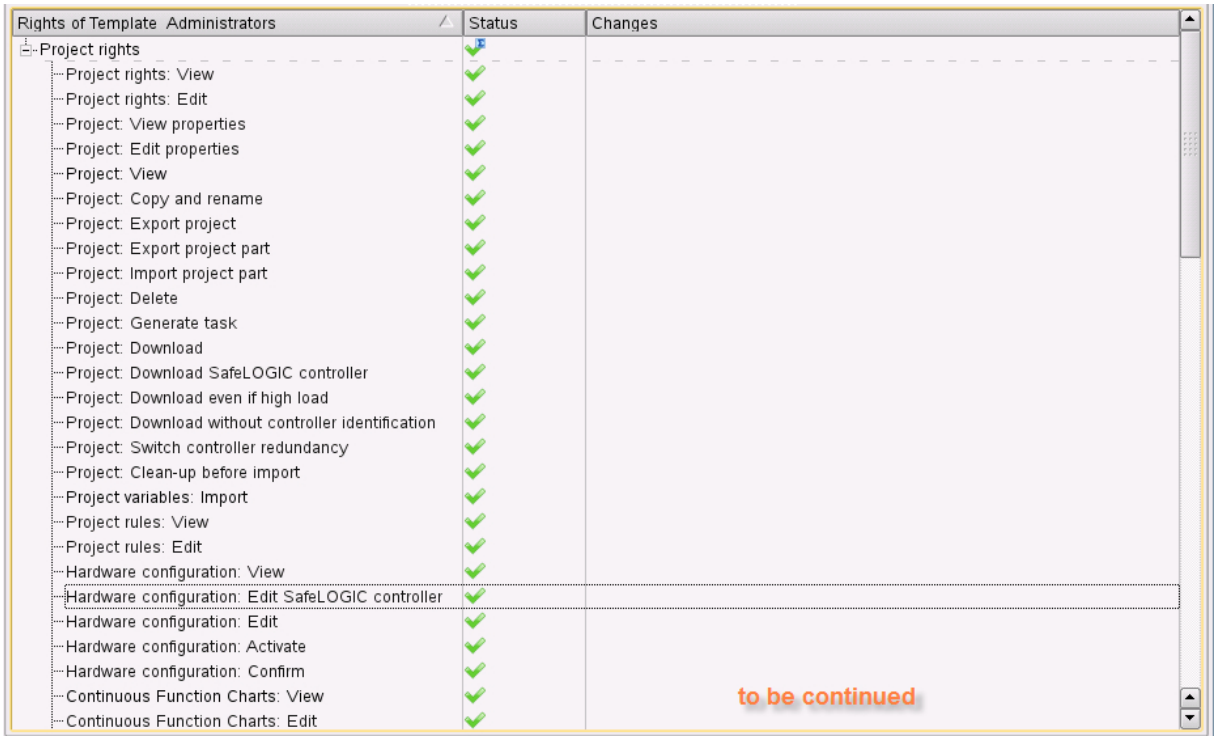


Figure 27: Project rights

It would be possible to create templates for each of the roles involved in developing the various parts of a project.

A graphic designer, for example, would receive access to the process graphics. The hardware designer would have access to the controllers and control computers. These roles can be defined and organized with any level of detail.

Editing individual rights.

Rights can be assigned as necessary and confirmed by right-clicking on an entry.

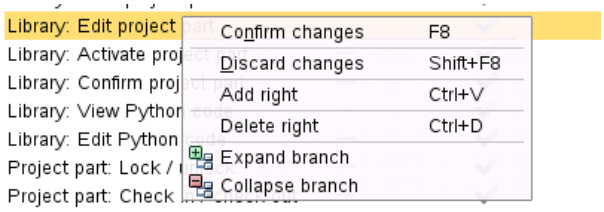


Figure 28: Menu for editing rights

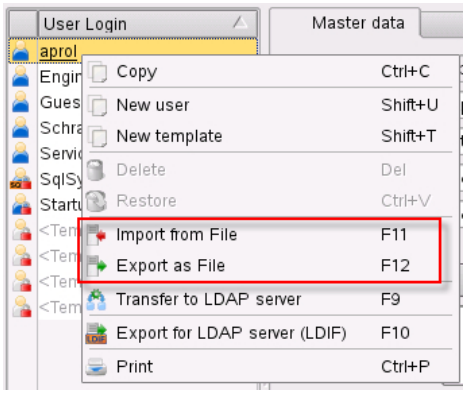
After all settings have been made satisfactorily, they can be saved. Users can now be added once this is completed.

Note: References

In general, objects with references cannot be deleted before all of the references are removed.

Important: References

Templates which have already been assigned users (i.e. references exist) cannot be deleted. (the "Delete" option is grayed out in the menu)



6.3 Creating users

The submenu **"New user"** can be used to create new users. This is how the project team is formed.

> **Note: User organization**
>
> It makes sense to enter all employees that will work with APROL and CaeManager as users. The users entered can be assigned a template, or the assignment can be changed at any time.
>
> User information is needed in various reports. Change control, build and target control reports make it possible to see what has been changed, built and loaded to the target at any time. Of course, it's also important to know who performed the respective actions.

6.3.1 Creating users

In order to create a new user, the **User Management** system needs to be started. Right-clicking on the left side then allows a **new user** to be created or even copied.

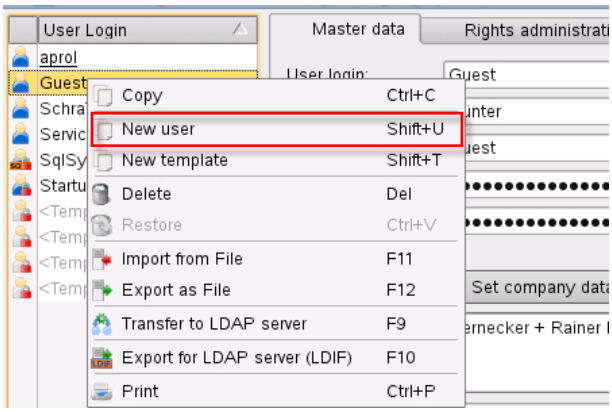


Figure 29: New user

A user can be deleted using the shortcut menu described above. Deleting is only possible, however, if all references to projects and templates have been removed.

The attributes for created users can then be edited on the right side.

The appropriate information can then be entered in the **User login, First name, Last name, Call number, Contact address, Password, Repeat password** and **Description** fields. The **"Set company data"** button can be used to apply the company data defined in the project. Whatever project is active at the time will be used. The same applies for the data of the engineering partner. This data is entered and edited under the project properties.

In addition, a default browser can also be defined for the created users.

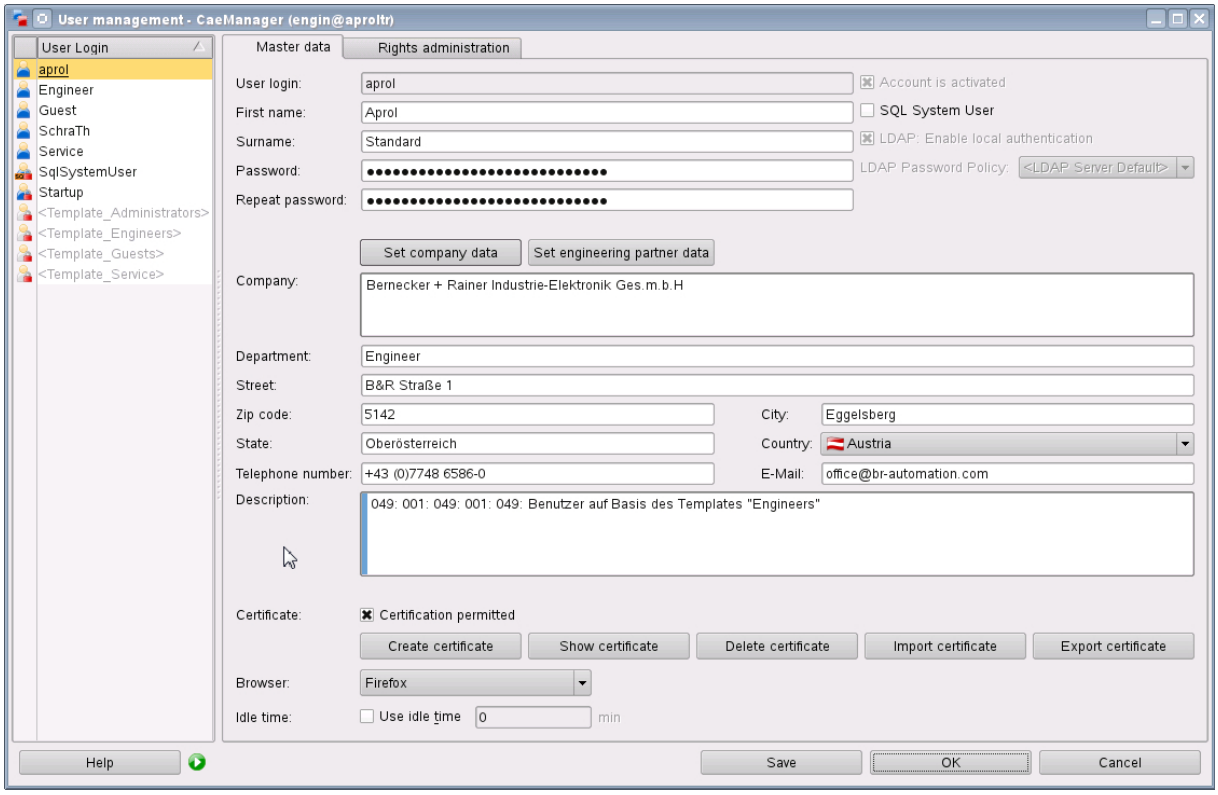


Figure 30: User data

Note: Login

"User login" refers to the username which is used to log in in the LoginServer dialog box.

Note: SQL system user

This user is only for authenticating data queried via SQL.

There must be a global SQL system user in order to have the functionality of standard reports.

Information: LDAP

In APROL, authentication of users can take place via a separate LDAP server or active directory server.

This makes it possible to organize and modify user data (password, password expiration, etc.) via the external LDAP server.

(LDAP : Lightweight Directory Access Protocol)

You can use the **"certificate"** area to create and manage your own certificates. These are need to sign PDF documents.

The "Idle time" is a safety function that can be defined separately for each user. A user who does not use the keyboard or mouse in this time will be logged out.

6.3.2 Assigning templates

The **templates** described above can be assigned to the newly created users using **drag and drop**.

It is possible to make changes after the templates have been assigned. (rights that deviate from the templat can be added or removed.)

Once this has been done, the changes need to be confirmed. This applies to both global rights as well as project rights.

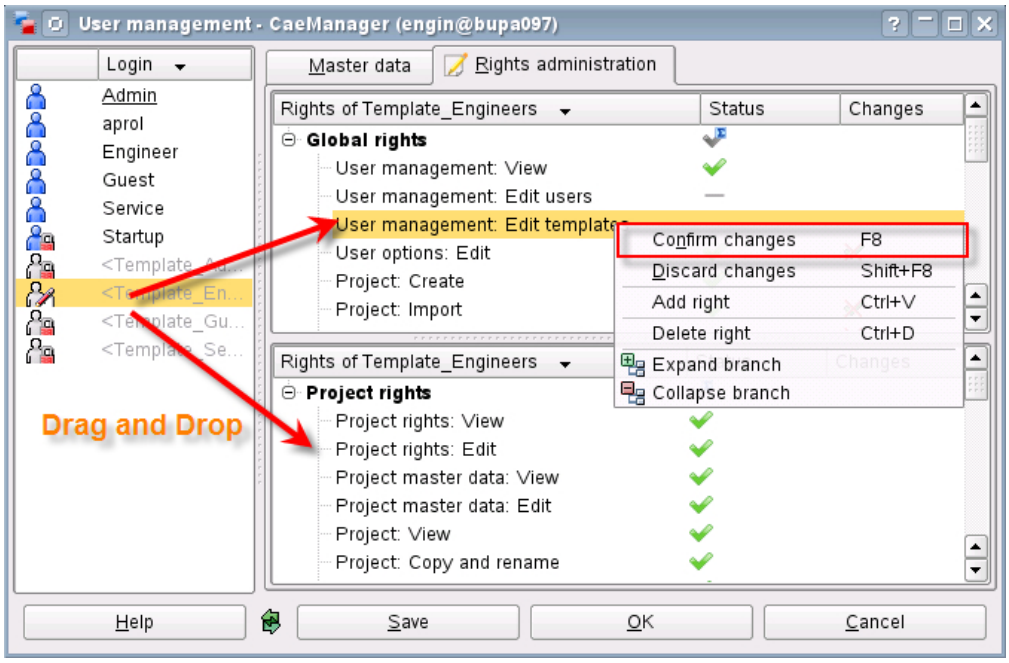


Figure 31: Assigning and confirming a template

To finish up, all of the changes must then be saved. This completes the User Management procedure.

6.4 Exporting/Importing users and templates

An **export/import mechanism** has been created so that templates and users can be managed throughout a project. This is particularly necessary when a single engineering team is responsible for several systems. For example, data can be exported from System A (XML format) and then imported into System B. This accelerates the organization of users enormously. The password is encrypted in the export file as well.

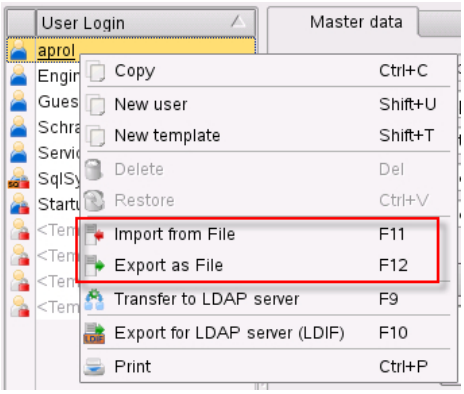


Figure 32: Export/Import mechanism

Exercise 1:

Create two new templates. Give one all the rights and the other only a few very limited rights.

Exercise 2:

Now create two users with first names and last names, and use their last names as their login (username). Please use "aprol" as the password. Assign the templates you created to these two developers. Check the access rights by logging in first as user 1 and then as user 2.

Note: Organization of rights

When starting a project, many users assign superuser rights to all designers who are assigned to the project. This can be helpful to get the project structure set up as quickly as possible without having to worry about the limitations of access rights. However, an administrative level should be established once the project has reached a certain size (by the start of commissioning at the latest).

7 Summary

What have we achieved so far?

We now know how an APROL process control system can be structured and are familiar with key features that range from a small system to a fully redundant solution. Multi Runtime Systems and redundant control solutions facilitate scalability and equipment availability. All of this is handled under the comfort and security of LINUX.

The resulting advantages are obvious.

We have also seen how an APROL system can be expanded or scaled as needed.

As the central engineering tool, CaeManager is the core of the engineering environment. It is therefore important to understand its structure thoroughly. The APROL help system (product documentation) should always be the first place to look if you have any questions.

The different areas of responsibility are clearly defined, but you will mostly only need the project engineering area in the future. Ideally, the engineering user management system only needs to be configured correctly once to make simple and intuitive engineering possible.

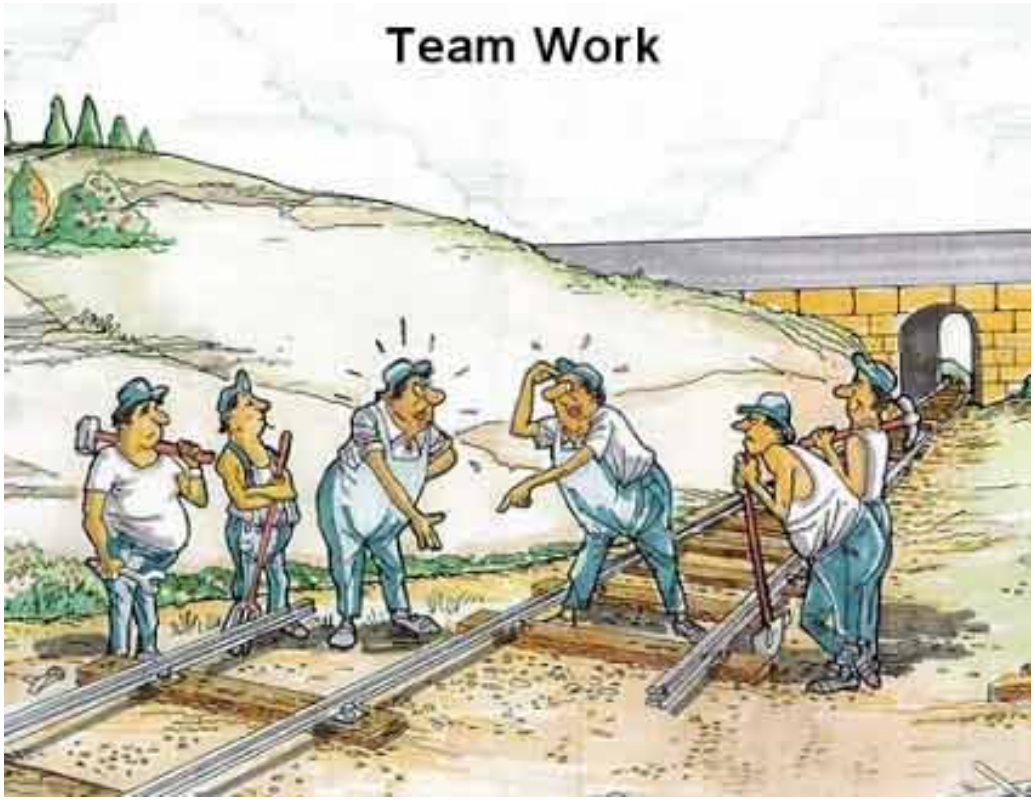


Figure 33: Cooperation

8 Appendix

8.1 Objectives checklist (a little review)

What is the difference between a RUNTIME and an OPERATOR system?

Which bus systems are significant?

What exactly is an operator bus?

What exactly is a process bus?

What exactly is a fieldbus?

How does "concurrent engineering" work?

What are the advantages of LINUX?

How is it possible to scale an APROL system?

What login mechanisms are needed to work with CaeManager?

How does the licensing mechanism work?

What combinations are useful and logical for the User Management system?

8.2 Responses to the objectives checklist

Please take down any necessary additional information.

8.3 Glossary

APROL	Name of B&R's process control system
CaeManager	Global engineering and configuration tool from APROL. (Computer-aided engineering)
LINUX	Operating system used by APROL. For B&R and APROL, this is the Novell SuSE distribution.
TCP/IP	Transmission Control Protocol / Internet Protocol
Ethernet	Network standard developed by Intel, Dec, and Xerox.
Demon	Background process that is only found in a process list. This includes all drivers and application servers (AlarmServer, etc.).
Task	Collection of logic that is executed on the controller or control computer as machine language.
Concurrent engineering	Refers to the possibility for several users to access a database without interfering with others.
Environment	For example, an engineering environment or runtime environment. Different environments naturally behave differently.
KDE	K Desktop Environment: Graphical user interface with a Windows look and feel.
LoginServer	Central server that manages logins for each environment.

Seminars and training modules

Automation Studio seminars and training modules

Programming and configuration	Diagnostics and service
SEM210 – Basics SEM246 – IEC 61131-3 programming language ST* SEM250 – Memory management and data storage SEM410 – Integrated motion control* SEM441 – Motion control (multi-axis systems) ** SEM480 – Hydraulics** SEM1110 – Axis groups and path-controlled movements** SEM510 – Integrated safety technology* SEM540 – Safe motion control*** SEM610 – Integrated visualization*	SEM920 – Diagnostics and service for end users SEM920 – Diagnostics and service with Automation Studio SEM950 – POWERLINK configuration and diagnostics* If you don't happen to find a seminar on our website that suits your needs, keep in mind that we also offer customized seminars that we can set up in coordination with your sales representatives: SEM099 – Individual training day Visit our website for more information: **www.br-automation.com/academy**

Overview of training modules	
TM210 – Working with Automation Studio TM213 – Automation Runtime TM223 – Automation Studio Diagnostics TM230 – Structured Software Development TM240 – Ladder Diagram (LD) TM241 – Function Block Diagram (FBD) TM242 – Sequential Function Chart (SFC) TM246 – Structured Text (ST) TM250 – Memory Management and Data Storage TM400 – Introduction to Motion Control TM410 – Working with Integrated Motion Control TM440 – Motion Control: Basic Functions TM441 – Motion Control: Multi-axis Functions TM1110 – Integrated Motion Control (Axis Groups) TM1111 – Integrated Motion Control (Path Controlled Movements) TM450 – Motion Control Concept and Configuration TM460 – Initial Commissioning of Motors TM500 – Introduction to Integrated Safety TM510 – Working with SafeDESIGNER TM540 – Integrated Safe Motion Control	TM600 – Introduction to Visualization TM610 – Working with Integrated Visualization TM630 – Visualization Programming Guide TM640 – Alarm System, Trends and Diagnostics TM670 – Advanced Visual Components TM920 – Diagnostics and service TM923 – Diagnostics and Service with Automation Studio TM950 – POWERLINK Configuration and Diagnostics TM1010 – B&R CNC System (ARNC0) TM261 – Closed-loop Control with LOOPCONR TM280 – Condition Monitoring for Vibration Measurement TM480 – The Basics of Hydraulics TM481 – Valve-based Hydraulic Drives TM482 – Hydraulic Servo Pump Drives In addition to a printed version, our training modules are also available on our website for download as electronic documents (login required): Visit our website for more information: **www.br-automation.com/academy**

Process control seminars and training modules

Process control standard seminars	Process control training modules
SEM830 – Process Control Training: Basic SEM800 – Process Control Training: Basic/Intensive SEM840 – Process Control Training: Advanced SEM820 – Solutions Training: Basic SEM821 – Solutions Training: Advanced	TM800 – APROL System Concept TM811 – APROL Runtime System TM812 – APROL Operator Management TM813 – APROL XML Queries and Audit Trail TM830 – APROL Project Engineering TM890 – The Basics of LINUX Visit our website for more information: **www.br-automation.com/academy**

* SEM210 - Basics is a prerequisite for this seminar.
** SEM410 - Integrated motion control is a prerequisite for this seminar.
*** SEM410 - Integrated motion control and SEM510 - Integrated safety technology are prerequisites for this seminar.