Start Center

Titel:
Title:

A generic user interface for multi-process software systems

Dokumenten Typ:
Document Type: User Manual

Konfigurations-Nr.:
Configuration Item No.:

Referenz- Nr.:
Reference No.:

Klassifikations-Nr.:
Classification No.:

Lieferbedingungs-Nr.:
DRL/DRD No.:

Freigabe Nr.:
Release No.:

Gruppierung (Dok.):
Group (Doc.-related):

Gruppierung (Version):
Group (Version-related):

Thema:
Subject:

Kurzbeschreibung:
Abstract:

The *Start Center* is a Java program that serves as a configurable generic user interface for software systems consisting of several processes.

Autor: Prepared by: Franz Kruse
Org. Einh.: Organ. Unit: TE 55
Unternehmen: Company: Astrium Bremen

Geprüft: Agreed by:
Org. Einh.: Organ. Unit:
Unternehmen: Company:

Genehmigt: Approved by:
Org. Einh.: Organ. Unit:
Unternehmen: Company:

Genehmigt: Approved by:
Org. Einh.: Organ. Unit:
Unternehmen: Company:

Daten/Dokument-Änderungsnachweis/Data/Document Change Record (DCR)

Ausgabe Issue	**Datum** Date	**Betroffener Abschnitt/Paragraph/Seite** Affected Section/Paragraph/Page	**Änderungsgrund/Kurze Änderungsbeschreibung** Reason for Change/Brief Description of Change
1/-	2006-03-01		Initial version
1/A	2006-11-29		Attribute "expand" for subsystems Java default style instead of OS default style
1/B	2007-03-20		Reference to Configuration user manual

Inhaltsverzeichnis/Table of Contents (ToC)

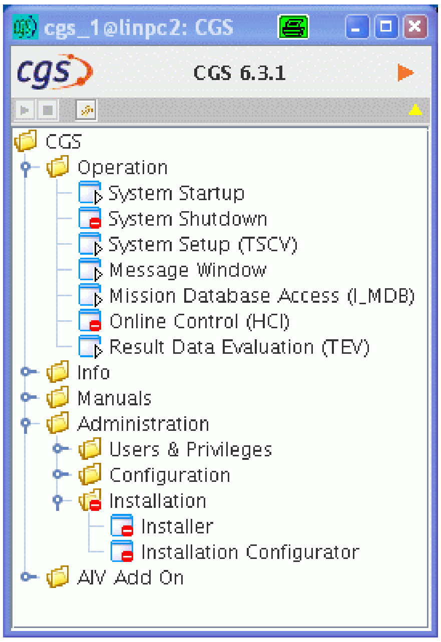
1. Reference Documents

1.1 An XML Based Configuration Concept, Configuration Editor and APIs.
CGS-RIBRE-MA-0008, 1/-, 2006-10-01

2. Usage of the Start Center

2.1 Overwiew

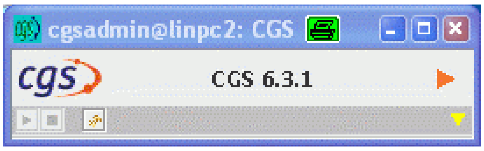
The *Start Center* is a Java program that serves as a configurable generic user interface for software systems consisting of several processes. It presents the processes of the system in a tree structure, hierarchically grouped in subsystems (process groups):



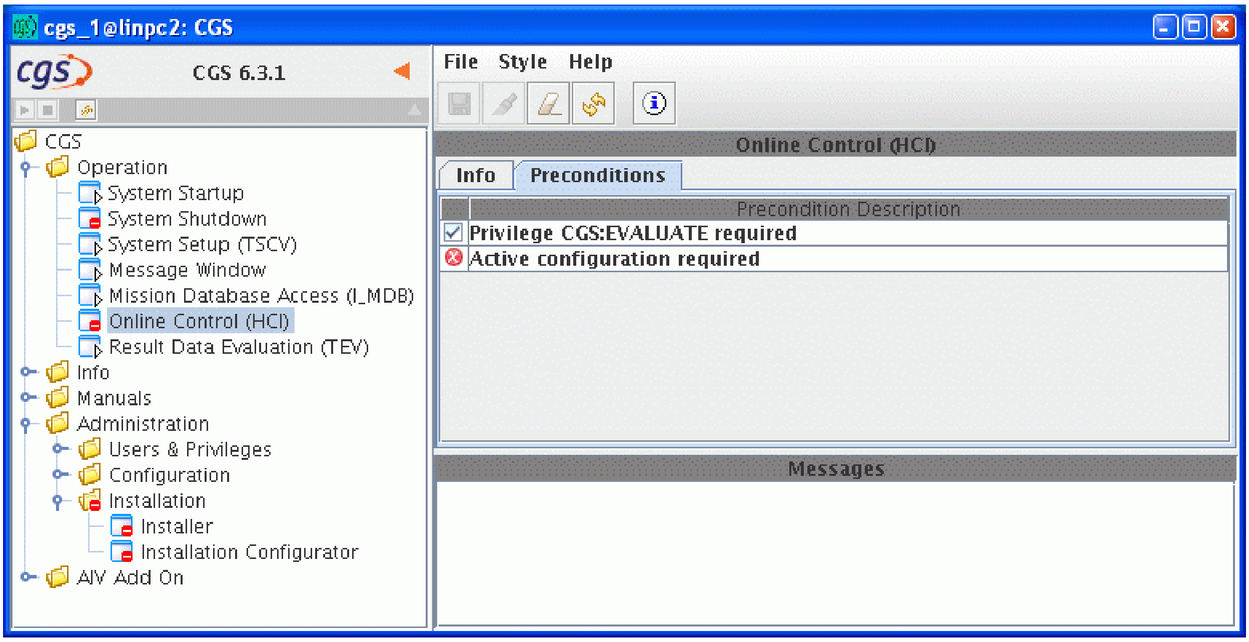
The processes (the leaves of the tree) are shown with different symbols indicating the current status of the process, whether it is currently running, has been started before, has terminated with success or failure etc.

Both subsystems and processes may be assigned *preconditions* that must be fulfilled, before a process can be started. As long as there are unfulfilled preconditions, the process is blocked (shown with the red blocked symbol). A blocked subsystem (process group) blocks all processes and sub-subsystems within that subsystem.

The left picture shows the *tree view* containing just the process tree. This is the standard view. Clicking on the vertical yellow arrow switches between the *tree view* and the *minimized view*:

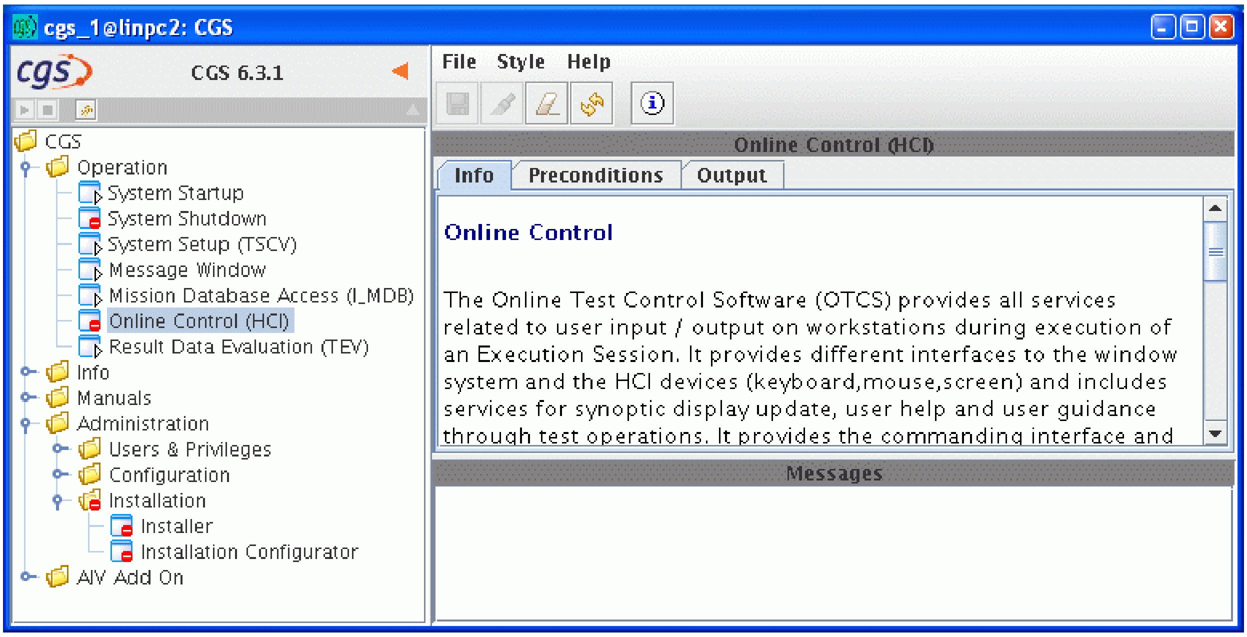


Clicking on the horizontal orange-coloured arrow opens or closes, resp., the *detail view* with different information about the selected process or subsystem. In the picture, the *Preconditions* subpage is opened, it lists all preconditions assigned to the selected item together with an indication whether a precondition is fulfilled or not.



	Precondition Description
☑	Privilege CGS:EVALUATE required
☒	Active configuration required

Each subsystem and process have a descriptive text, it is shown in the *Info* subpage (opened by clicking on the *Info* tab):



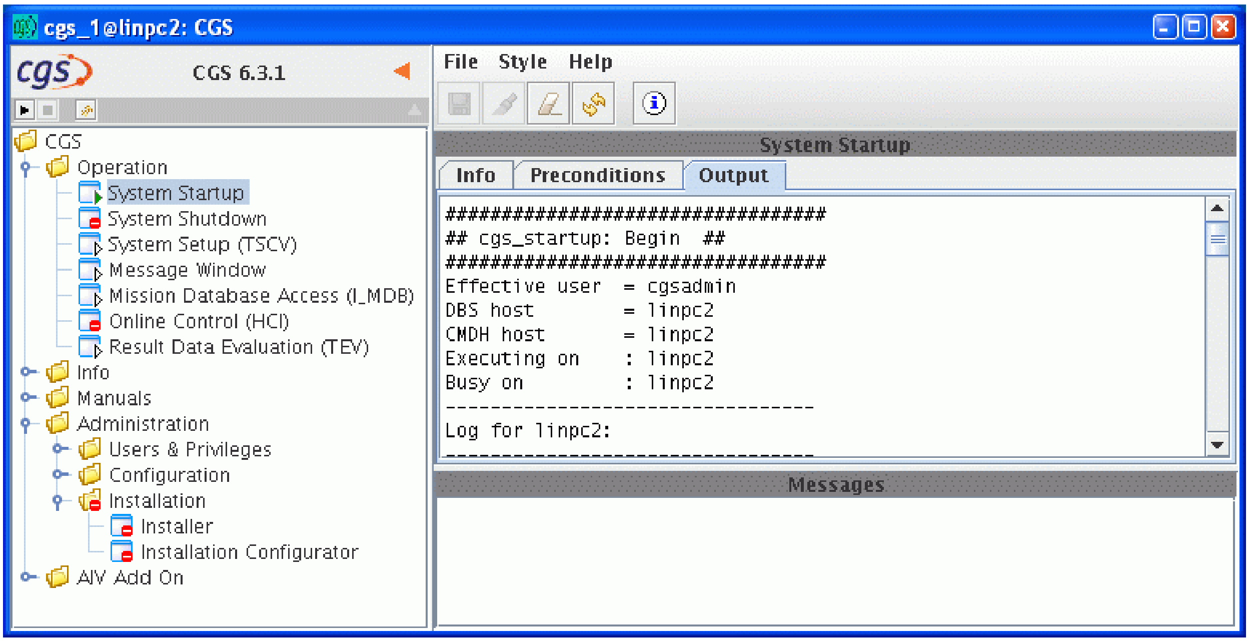
Online Control

The Online Test Control Software (OTCS) provides all services related to user input / output on workstations during execution of an Execution Session. It provides different interfaces to the window system and the HCI devices (keyboard,mouse,screen) and includes services for synoptic display update, user help and user guidance through test operations. It provides the commanding interface and

A process may be started in either of two ways:

- by double-clicking a process symbol in the process tree,
- by selecting a process symbol and clicking on the *Start* button in the tool bar of the tree subwindow.

If the detail view is open, it now switches to the *Output* subpage that shows all output of the process. Text written to its standard output channel is shown in black, and text written to its standard error channel is shown in red:



```
###################################
## cgs_startup: Begin  ##
###################################
Effective user  = cgsadmin
DBS host        = linpc2
CMDH host       = linpc2
Executing on    : linpc2
Busy on         : linpc2
----------------------------------
Log for linpc2:
----------------------------------
```

It depends on the configuration, whether a running process may be stopped by clicking the *Stop* button in the tool bar of the tree subwindow. For processes that cannot be stopped this way the *Stop* button is disabled.

2.2 Preconditions

Processes and subsystems may have one or more preconditions assigned. All of the preconditions must be fulfilled, before a process can be started. As long as there are unfulfilled preconditions, the process is blocked. Blocked processes are shown with a blocked symbol in the process tree (see 2.4.1). If a subsystem is blocked, all of its contents (processes, sub-subsystems) are implicitly blocked, too.

Behind each precondition, there is a dedicated process (a script or program) that checks whether the precondition is fulfilled and reports this via its exit status to the Start Center. Preconditions are automatically checked in the following situations:

- When the Start Center is started, all preconditions of the initially visible processes and subsystems are checked.
- When a branch of the process tree is opened, all preconditions of all visible processes and subsystems within the newly opened branch are checked.
- When a process is started, its preconditions are checked. The process is prevented from being started, if any of its preconditions is not fulfilled.
- Whenever a running process terminates, all preconditions of the currently visible processes and subsystems are checked, since the process may have changed the status of a precondition.

This shows, that precondition checking is synchronous, i. e. triggered by certain events in the Start Center. The status of preconditions may,however, change ansynchronously, as well. As an example, a precondition depending on the existence of a file will become unfulfilled, if the file is deleted for some reason. These asynchronous events are not detected automatically by the Start Center, in order to avoid heavy CPU load for periodic event checking. Instead, the user may trigger additional precondition checking at any time by clicking on the *Refresh* button in either of the tree subwindow tool bar (2.4.1) or the detail subwindow toolbar (2.4.2.4).

2.3 Command Line Invocation

The Start Center is started via the command line in the form

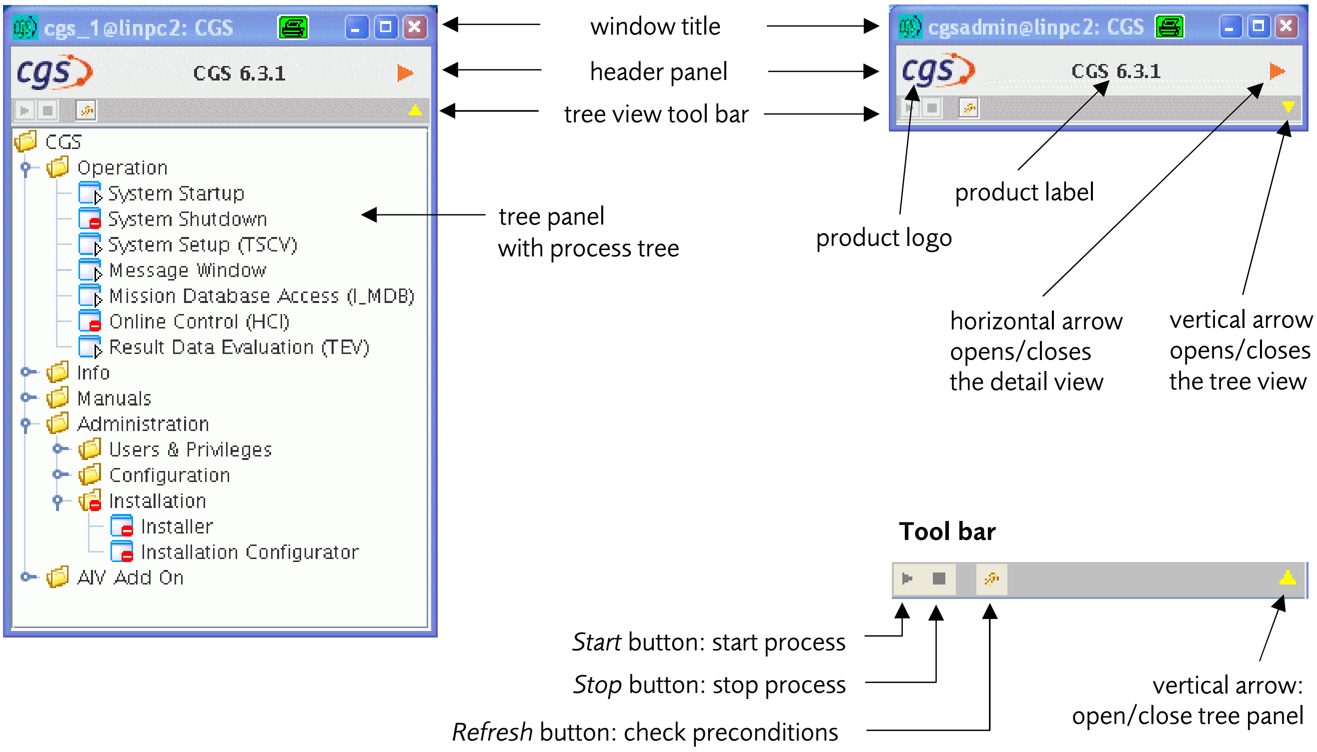
```
start_center file_name ...
```

The parameter is the name of an XML file containing the configuration of the Start Center. The configuration describes the specific appearance of the Start Center: title, logo, subwindow sizes etc., and in particular the process tree, see chapter 3 for details.

More than one XML file may be given. These files are merged into one configuration, such that elements from succeeding files are added to the elements from preceding files, or overlay and hide these elements, if they are already present. This makes it possible to build multiple-level configurations, e. g. in the form that a basic configuration on system level may be extended and modified by separate configurations on group and user level. This way, a user may not only add his own processes to the process tree, but also modify e. g. the product logo and the subwindow sizes.

2.4 Detailed Description

2.4.1 Tree View and Minimized View



Subsystem/process symbols

Subsystem (process group)

Subsystem (process group), blocked (precondition not fulfilled)

Process, not started yet, ready to be started

Process, sucessfully terminated, ready to be started again

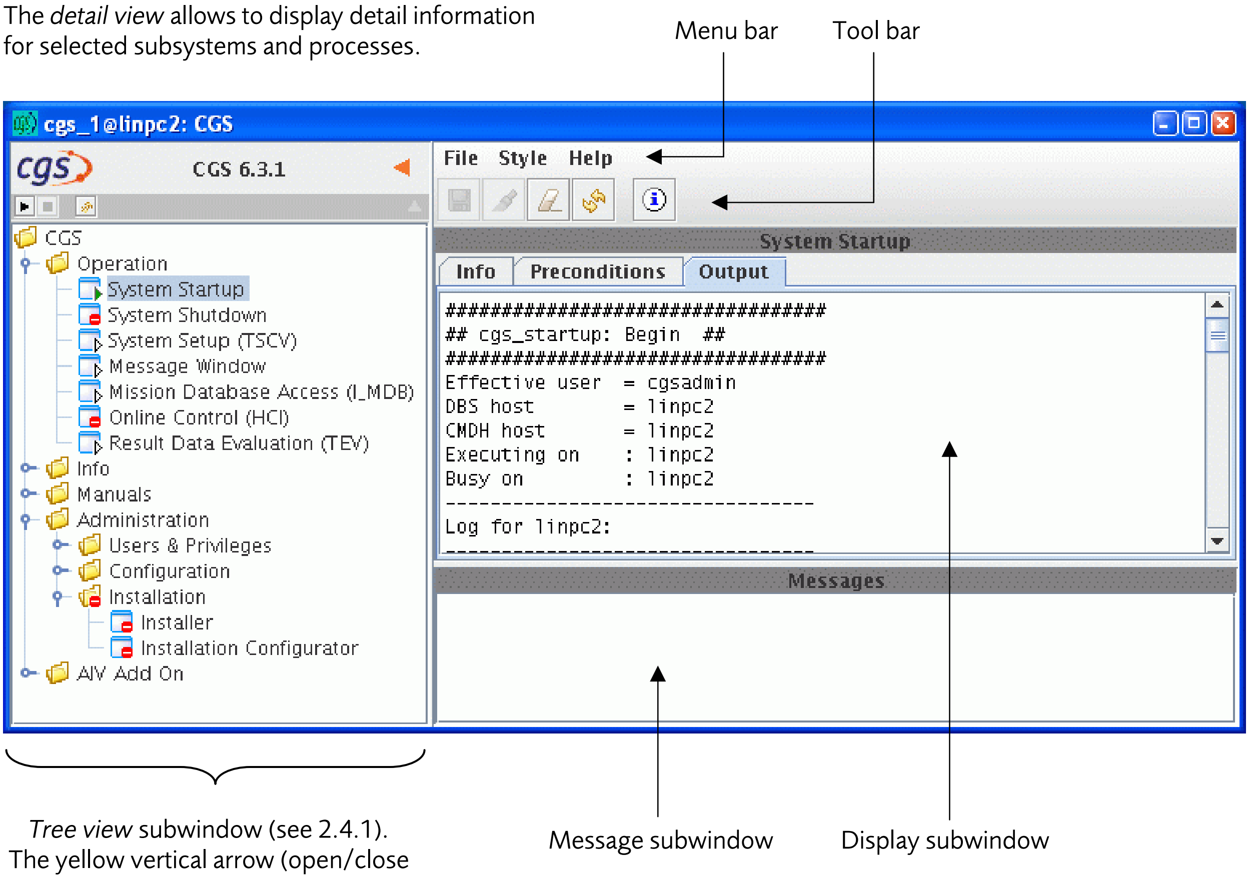
Process, terminated with failure, ready to be started again

Process, currently running, cannot be started now

Process, blocked (precondition not fulfilled)

2.4.2 Detail View

The *detail view* allows to display detail information for selected subsystems and processes.



```
##################################
## cgs_startup: Begin  ##
##################################
Effective user  = cgsadmin
DBS host        = linpc2
CMDH host       = linpc2
Executing on    : linpc2
Busy on         : linpc2
---------------------------------
Log for linpc2:
---------------------------------
```

Tree view subwindow (see 2.4.1).
The yellow vertical arrow (open/close tree panel) is disabled in detail view.

2.4.2.1 Display Subwindow

The *Display* subwindow is a tabbed pane consisting of several subpages to be opened by clicking on the respective tab:

Info A textual description of the subsystem or process
This may be simple text or formatted text containing graphical elements and links.

Preconditions The table of preconditions assigned to the subsystem or process.
Each precondition is shown with a short descriptive text and a symbol indicating its current status:

☑ fulfilled

☒ not fulfilled

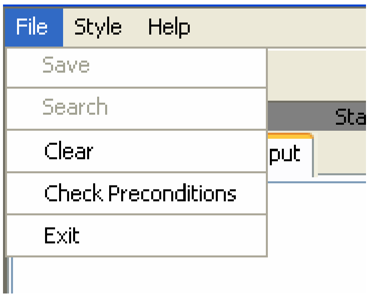
? status unknown

Output All output of the process (this tab is shown for processes only)
Text written to standard output is shown in black, text written to standard error is shown in red.

2.4.2.2 Message Subwindow

The *Message* subwindow displays messages issued by the Start Center itself. Informal messages are shown in black, error messages are shown in red.

2.4.2.3 The Menu Bar



The *File* Menu

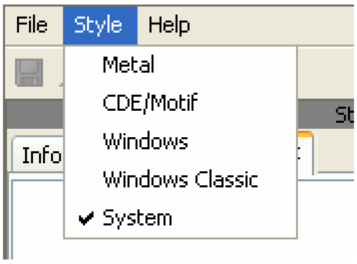
Save Store the contents of the Output subpage in a file (not implemented yet)

Search Search for text patterns in the Output subpage (yet implemented yet)

Clear Delete the Output subpage contents

Check Preconditions Check preconditions for all currently visible subsystems and processes

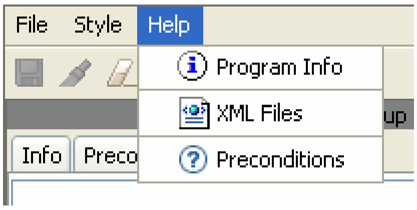
Exit Exit the Start Center



The *Style* Menu

This menu lists all available styles (look & feel) and allows to change the Start Center style.

Default: the Java default style



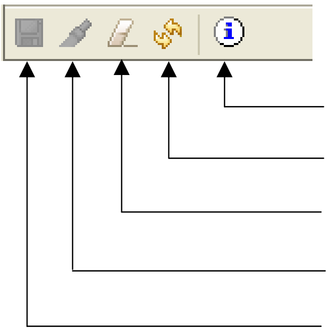
The *Help* Menu

Program Info Display short info about the Start Center

XML Files List the currently loaded XML configuration files

Preconditions List all defined preconditions (not implemented yet)

2.4.2.4 The Tool Bar



Info: Display short info about the Start Center

Refresh: Check preconditions for all currently visible subsystems and processes

Clear: Delete the Output subpage contents, only enabled if subpage is open

Search: Search for text patterns in the Output subpage (not implemented yet)

Save: Store the contents of the Output subpage in a file (not implemented yet)

3. Configuration of the Start Center

3.1 Configuration Concept

The Start Center is configured through one or more XML files, a configuration comprises the following elements:

- the window appearance:
    - the product logo shown in the tree panel header
    - a big product logo shown in the display subwindow
    - the product label shown in the tree panel header
    - the title shown in the window header bar
    - a product icon shown when the Start Center window is iconized
    - width and height of the process tree panel
    - width and height of the display subwindow
- the process tree:
    - the subsystem/process hierarchy
    - the command lines to invoke the processes
    - the preconditions with their command lines
    - the assignment of preconditions to subsystems and processes.

The configuration is based on the configuration concept described in 1.1. The Configuration Editor can be used to set up and edit the configuration.

Please note that a configuration need not be given in one file, but may be split in several files, each containing a part of the overall configuration. Only when merged together, the set of separate files must result in a complete Start Center configuration. All files must be given as parameters to the `start_center` program. Moreover, if a file given in the command line contains attributes that were already given in a preceding file, these attributes overlay and hide the previously given attributes. This allows to build multiple-level configurations, e. g. in the form that a basic configuration on system level may be extended and modified by separate configurations on group and user level. This way, a user may not only add his own processes to the process tree, but also modify e. g. the product logo and the subwindow sizes.

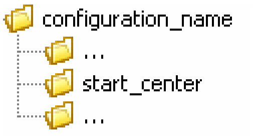
The values of several attributes (like file names and certain text elements) may contain placeholders for environment variables. They have one of the two forms

```
{$variable_name}
{$variable_name=default_value}
```

These placeholders are substituted by the Start Center with the value of the corresponding environment variable. If the variable does not exist and a default value is given, the default value replaces the placeholder. If the variable does not exist and there is no default value, the Start Center will report an error and reject the configuration.

The following describes the configuration, giving both a graphical representation of the elements in Configuration Editor conventions (in terms of groups and attributes) and a pure XML pattern. This should enable a reader to manually set up a configuration in XML form. Within the XML patterns, identifying attributes to be given in this form are shown in bold, attribute values to be supplied by the user are shown in italic and in red text.

3.2 Overall Configuration Structure

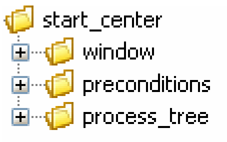

configuration_name
...
start_center
...

```
<?xml version="1.0" encoding="UTF-8"?>

<configuration name="configuration_name">

  ...
  ...  possibly other configuration elements
  ...

  <group name="start_center">
    ...
    ...  all Start Center configuration goes here
    ...
  </group>

  ...
  ...  possibly other configuration elements
  ...

</configuration>
```

The overall configuration must contain a global group named start_center, this group must contain the Start Center configuration. Any configuration elements outside this group are ignored. This allows to keep a Start Center configuration as part of a general configuration.

The <configuration> element may or may not reference a Schema file, In any case, each XML file will be validated, both by the Configuration Editor and by the APIs (and hence by the Start Center), against the basic Schema on which the configuration concept is based.

3.3 Basic Start Center Configuration Layout


start_center
⊞ window
⊞ preconditions
⊞ process_tree

```
<group name="start_center">
  <group name="window"> ... </group>
  <group name="preconditions"> ... </group>
  <group name="process_tree"> ... </group>
</group>
```

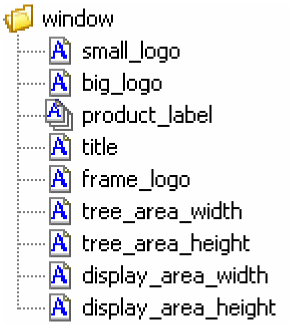
The start_center group must contain three subgroups:

window contains all attributes that determine the appearance of the Start Center window

preconditions contains the definition of all preconditions used in the configuration

process_tree defines the process tree, including the assignment of precontions to subsystems and processes

3.4 The window Group



window
- small_logo
- big_logo
- product_label
- title
- frame_logo
- tree_area_width
- tree_area_height
- display_area_width
- display_area_height

```
<group name="window">
  <attribute name="small_logo"><value>file_name</value></attribute>
  <attribute name="big_logo"><value>file_name</value></attribute>

  <attribute name="product_label">
    <list>
      <value>product_label_line</value>
      <value>product_label_line</value>
      ...
    </list>
  </attribute>

  <attribute name="title"><value>title_text</value></attribute>
  <attribute name="frame_logo"><value>file_name</value></attribute>
  <attribute name="tree_area_width"><value>integer</value></attribute>
  <attribute name="tree_area_height"><value>integer</value></attribute>
  <attribute name="display_area_width"><value>integer</value></attribute>
  <attribute name="display_area_height"><value>integer</value></attribute>
</group>
```

The `window` group comprises a number of attributes that determine the appearance of the Start Center window:

`small_logo` file name of the small product logo displayed in the tree view header
(may contain environment variable placeholders)

`big_logo` file name of the big product logo displayed in the display area of the detail view
(may contain environment variable placeholders)

`product_label` the product label text (a list attribute)
may consist of several lines and contain environment variable placeholders,
each value in the list corresponds to a line of text,
lines may contain basic HTML formatting tags, such as <b> or <i>,
lines may contain environment variable placeholders

`title` the title shown in the window header
(may contain environment variable placeholders)

`frame_logo` file name of the icon displayd for the iconized Start Center window
(may contain environment variable placeholders)

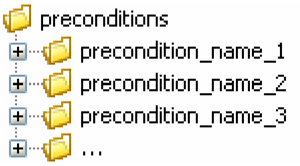
`tree_area_widht` an integer defining the width of the product tree panel (in pixels),
if omitted, a default width is assumed

`tree_area_height` an integer defining the height of the product tree panel (in pixels) ,
if omitted, a default height is assumed

`display_area_width` an integer defining the width of the display subwindow (in pixels) ,
if omitted, a default width is assumed

`display_area_height` an integer defining the height of the display subwindow (in pixels) ,
if omitted, a default height is assumed

3.5 The preconditions Group

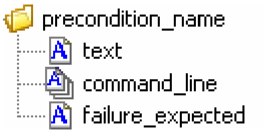


preconditions
- precondition_name_1
- precondition_name_2
- precondition_name_3
- ...

```
<group name="preconditions">
  <group name="precondition_name_1", class="precondition">...</group>
  <group name="precondition_name_2", class="precondition">...</group>
  <group name="precondition_name_3", class="precondition">...</group>
  ...
</group>
```

The preconditions group defines all preconditions used in the configuration.

3.6 Precondition Definition



precondition_name
- text
- command_line
- failure_expected

```
<group name="precondition_name", class="precondition">
  <attribute name="text"><value>precondition_text</value></attribute>

  <attribute name="command_line">
    <list>
       <value>command_name</value>
       <value>parameter_or_option</value>
       <value>parameter_or_option</value>
       ...
    </list>
  </attribute>

  <attribute name="failure_expected"><value>true/false</value></attribute>
</group>
```

For each precondition there is a dedicated process (a script or program) that checks the precondition and reports its value via its exit status. Any output of the process is discarded. The exit status may be interpreted in two ways:

positive: 0 = precondition fulfilled, otherwise precondition not fulfilled (the default)

negative: 0 = precondition not fulfilled, otherwise precondition fulfilled

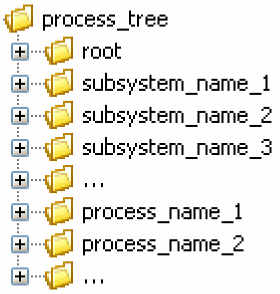
Each precondition is defined by a group whose name is an identifier serving as the internal name of the precondition and whose class is precondition. It contains the following attributes:

text — a one-line text describing the precondition, shown in the precondition table
(may contain environment variable placeholders)

command_line — the command line to invoke the process (a list attribute)
The first value in the list is the command name to start the process,
each parameterand option is given as a separate value in the list.
Lines may contain environment variable placeholders.

failure_expected — true if the exit status is to be negated (i. e. 0 = precondition not fulfilled, otherwise fulfilled),
if omitted, false is assumed

3.7 The `process_tree` Group

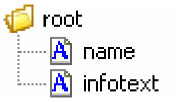


process_tree
root
subsystem_name_1
subsystem_name_2
subsystem_name_3
...
process_name_1
process_name_2
...

```
<group name="process_tree">
  <group name="root", class="root">...</group>
  <group name="subsystem_name_1", class="subsystem">...</group>
  <group name="subsystem_name_2", class="subsystem">...</group>
  <group name="subsystem_name_3", class="subsystem">...</group>
  ...
  <group name="process_name_1", class="process">...</group>
  <group name="process_name_2", class="process">...</group>
  ...
</group>
```

The `process_tree` group defines the hierarchical subsystem/process tree. The first subgroup defines the root node of the tree. It is represented by a group with class `root` that contains the root attributes. Then the first level of the tree is given as a list of subsystem and/or process definitions, each given as a group containing the respective attributes. Groups with class `subsystem` define subsystems, groups with class `process` define processes. Subsystem groups contain the deeper levels of the tree. There may be only subsystem groups, only process groups or an arbitrary mixture of both.

3.8 Tree Root Definition



root
name
infotext

```
<group name="root", class="root">...</group>
  <attribute name="name"><value>root_display_name</value></attribute>
  <attribute name="infotext"><value>info_text</value></attribute>
</group>
```

The group with class `root` defines the root node of the process tree. It contains the following attributes:

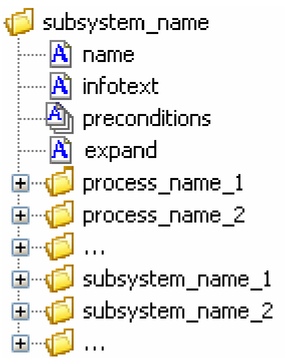
`name` the name of the root node to be displayed in the tree panel

`infotext` the informal text to be displayed in the *Info* subpage of the detailed view
The attribute value may have one of two forms:

`@`*`file_name`* Info text given as HTML text contained in file *`file_name`*.
The file name may contain environment variable placeholders.

`text` Text given as a one-line short text

3.9 Subsystem Definition



subsystem_name
name
infotext
preconditions
expand
process_name_1
process_name_2
...
subsystem_name_1
subsystem_name_2
...

```
<group name="subsystem_name", class="subsystem">
  <attribute name="name"><value>root_display_name</value></attribute>
  <attribute name="infotext"><value>info_text</value></attribute>

  <attribute name="preconditions">
    <list>
      <value>precondition_name_1</value>
      <value>precondition_name_2</value>
      ...
    </list>
  </attribute>

  <attribute name="expand"><value>true/false</value></attribute>

  <group name="process_name_1", class="process">...</group>
  <group name="process_name_2", class="process">...</group>
  <group name="process_name_3", class="process">...</group>
  ...
  <group name="subsystem_name_1", class="subsystem">...</group>
  <group name="subsystem_name_2", class="subsystem">...</group>
  ...
</group>
```

A group with class `subsystem` defines a subsystem. A subsystem may contain a mixture of processes and sub-subsystems. The following attributes define the properties of the subsystem:

`name` the name of the root node to be displayed in the tree panel

`infotext` the informal text to be displayed in the *Info* subpage of the detailed view
The attribute value may have one of two forms:

`@file_name` Info text given as HTML text contained in file `file_name`.
The file name may contain environment variable placeholders.

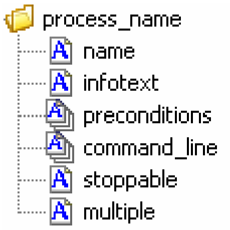
`text` Text given as a one-line short text

`preconditions` list of preconditions assigned to this subsystem
This attribute is a list attribute, each value in the list is a precondition name from the `preconditions` group.

`expand` a boolean value indicating whether the subsystem branch is initially expanded:
`true` = subsystem branch initially expanded
`false` = subsystem branch initially collapsed (default)

Then there may be a list of process and subsystem definitions, each given as a group with class `subsystem` or `process`, respectively.

3.10 Process Definition



```
<group name="process_name", class="process">
  <attribute name="name"><value>root_display_name</value></attribute>
  <attribute name="infotext"><value>info_text</value></attribute>

  <attribute name="preconditions">
    <list>
      <value>precondition_name_1</value>
      <value>precondition_name_2</value>
      ...
    </list>
  </attribute>

  <attribute name="command_line">
    <list>
       <value>command_name</value>
       <value>parameter_or_option</value>
       <value>parameter_or_option</value>
       ...
    </list>
  </attribute>

  <attribute name="stoppable"><value>true/false</value></attribute>
  <attribute name="multiple"><value>true/false</value></attribute>
</group>
```

A group with class `subsystem` defines a subsystem. A subsystem may contain a mixture of processes and sub-subsystems. The following attributes define the properties of the subsystem:

`name` the name of the root node to be displayed in the tree panel

`infotext` the informal text to be displayed in the *Info* subpage of the detailed view
The attribute value may have one of two forms:

`@file_name` Info text given as HTML text contained in file `file_name`.
The file name may contain environment variable placeholders.

`text` Text given as a one-line short text

`preconditions` list of preconditions assigned to this subsystem
This attribute is a list attribute, each value in the list is a precondition name from the `preconditions` group.

`command_line` the command line to invoke the process (a list attribute)
The first value in the list is the command name to start the process,
each parameter and option is given as a separate value in the list (see note below).
Lines may contain environment variable placeholders.

`stoppable` a boolean value indicating whether the processed may be stopped via the Start Center when running:
`true` = process may be stopped, *Stop* button enabled
`false` = process cannot be stopped, *Stop* button disabled

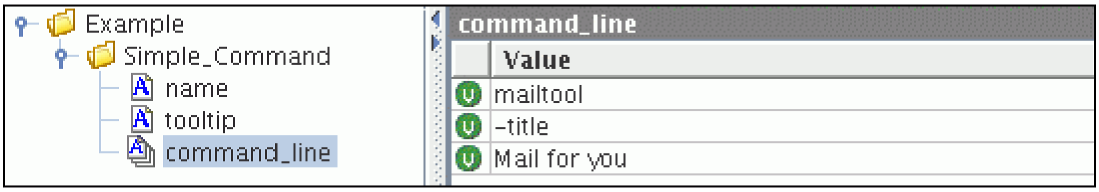
`multiple` a boolean value indicating whether there may be parallel instances of the process:
`true` = several parallel instances of the process may be started
`false` = only one instance of the process can be active at a time

This attribute is not implemented yet.

A note on command lines:

There is no shell involved in evaluating the command line, therefore only simple command calls without any shell functionality can be used here. The placeholders are evalutated by the Configuration Editor itself. Here is an example for some command line

```
mailtool -title "Mail for you"
```

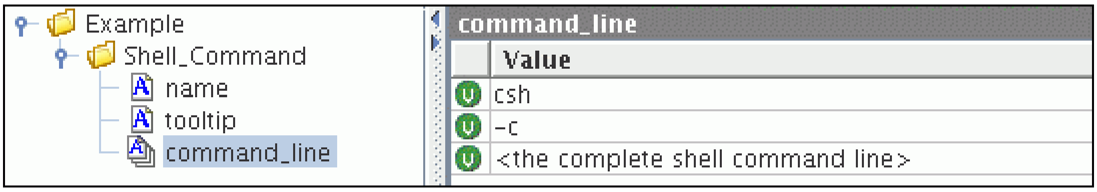


command_line

	Value
V	mailtool
V	-title
V	Mail for you

If shell functionality is needed (e. g. grouped commands), use a shell as the process with the actual command line in the `−c` option. Then the command line must be given as three values:

```
csh -c "the complete shell command line"
```

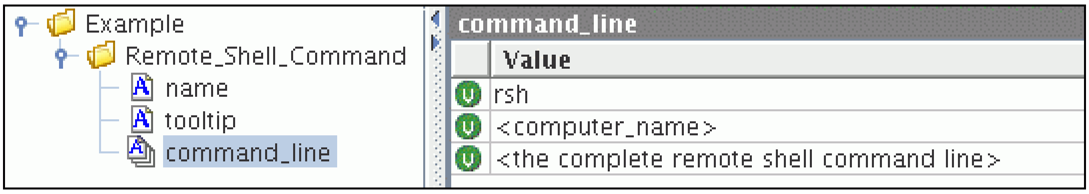


command_line

	Value
V	csh
V	-c
V	<the complete shell command line>

Analogously, a remote shell can be used. Again, the command line has to be given as three values:

```
rsh computer_name "the complete remote shell command line"
```



command_line

	Value
V	rsh
V	<computer_name>
V	<the complete remote shell command line>