

# Table of Contents

<b>Starting the Codabix Application</b>	1
<b>Configuration</b>	1
<b>General</b>	2
(1) Codabix Toolbar	2
Licence Dialog	3
(2) Working Area	4
(3) Title Bar	4
(4) Toolbar	4
(5) Menu	5
(6) Node Tree	6
(7) Data Area	7
<b>Nodes</b>	8
What are Nodes?	8
Folder Nodes	8
Datapoint Nodes	8
Link Nodes	8
Directory Nodes	9
<b>Node Properties</b>	9
Common Properties	9
Node Types	9
Datapoint Nodes	10
Value Types	11
Node View	12
Node Structure	13
Create a Node	14
Add a Folder Node	14
Add a Directory Node	15
Add a Datapoint Node	15
Link Nodes	16
Node Access	17
Add / Delete Node Access to a User	17
Add / Remove via the Node View	18
Add / Remove via the User Group	18
<b>User</b>	19
Add Users	19
Edit Users	19
<b>User Group</b>	20
Add User Group	20
Edit User Group	21
Add User to User Group	21
<b>Plugins</b>	22
Device Plugins	22
Exchange Plugins	23
Interface Plugins	23



# Codabix Web Configuration

## Starting the Codabix Application

1. Install Codabix, see [Codabix Installation](#).
2. Make the necessary adjustments [Codabix Setup and First Start](#).



3. Double-click the icon on the Desktop to start Codabix, or run the following command on the command line (e.g. on Windows Server Core):

```
"%ProgramFiles%\Traeger\Codabix\codabix-ui.exe"
```

4. The Codabix Login dialog will show.

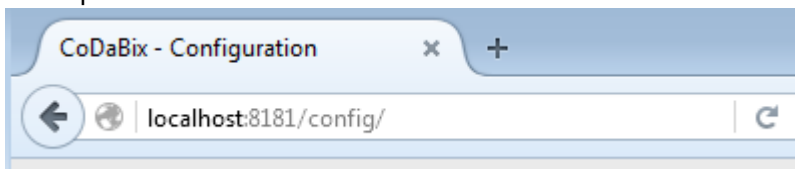
## Configuration

Codabix can be configured using a web-based GUI. On Windows, Codabix uses an embedded web browser to display the Web Configuration.

You can also open the Web Configuration with your own browser (e.g. Edge, Chrome, Firefox, Safari). By default, Codabix uses a local HTTP binding (default port 8181), so you can use the following URL:

```
http://localhost:8181/config/
```

Example:



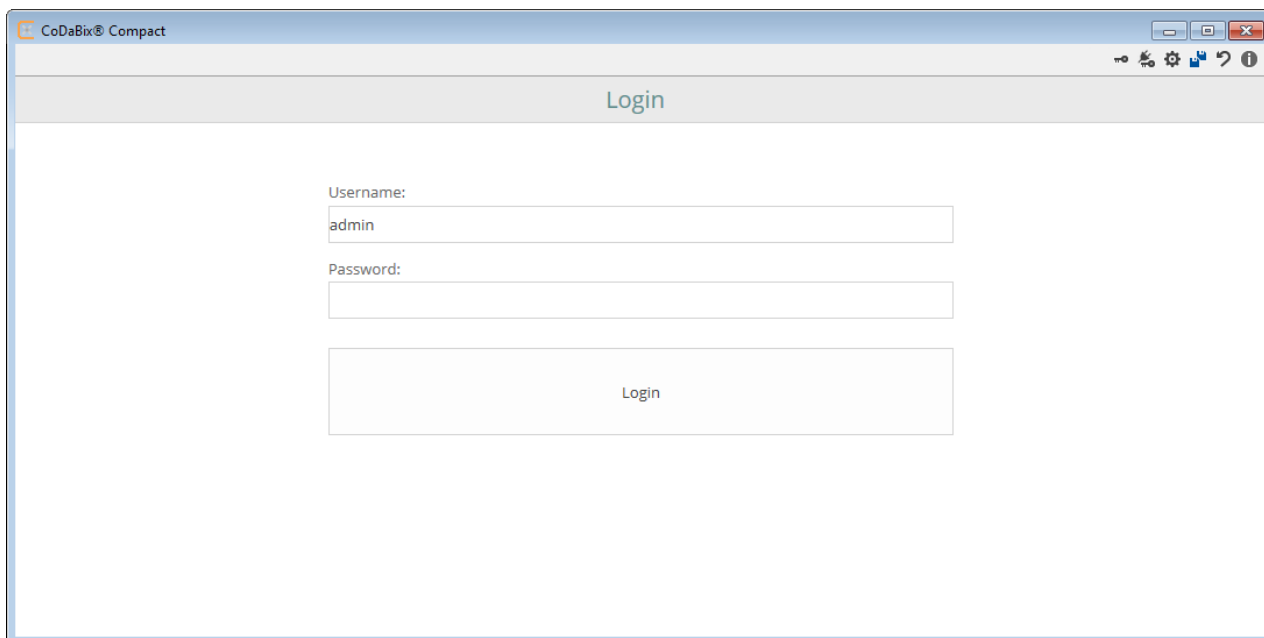
**Please note:** If you want to access the Codabix Web Configuration from a **remote computer**, you must add a remote HTTP binding in the Codabix Settings (for Codabix for Linux this already done by default).

To access Codabix with a browser from a remote computer using HTTP, you can use the following URL:

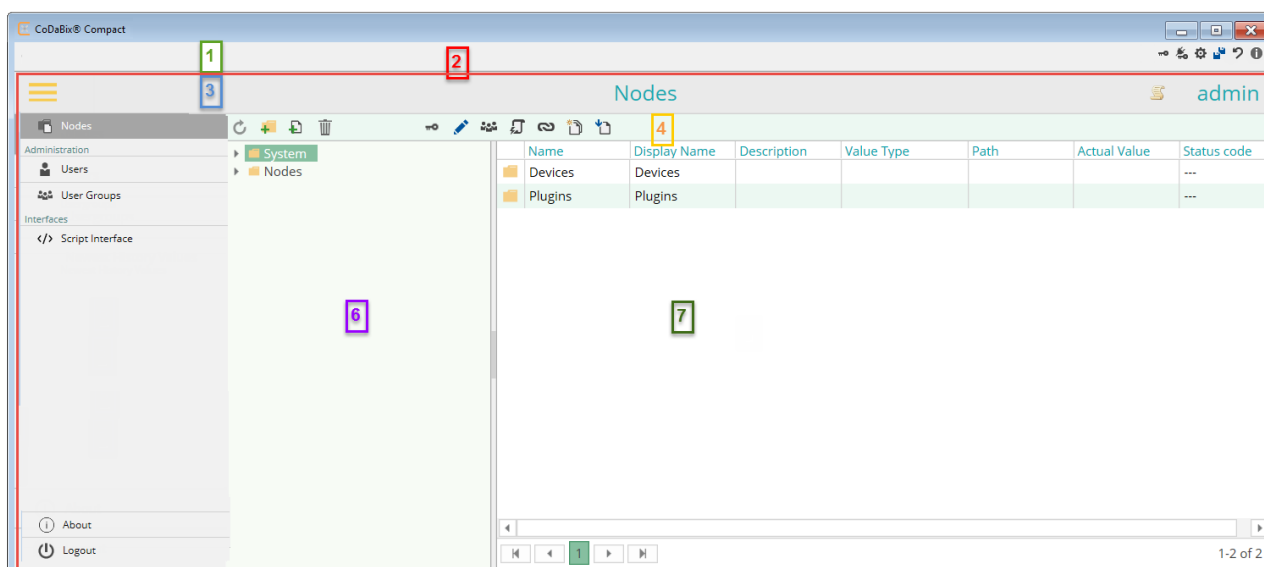
```
http://<hostname>:<port>/config/
```

## General

After starting or calling the URL the login dialog is displayed:






After the login you reach the Codabix main window.



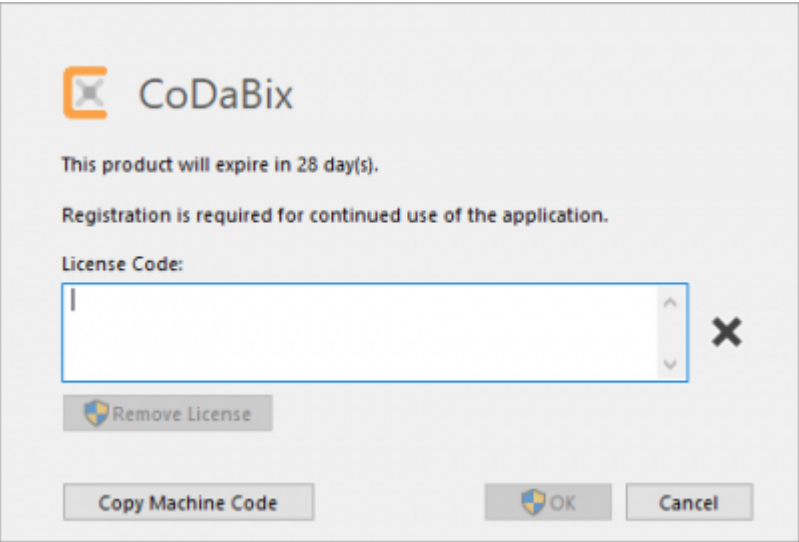
The Codabix Web Configuration is separated in different parts:

### (1) Codabix Toolbar

This toolbar is only visible if you use the Codabix application for the configuration. So this bar is only referring to the Codabix application.

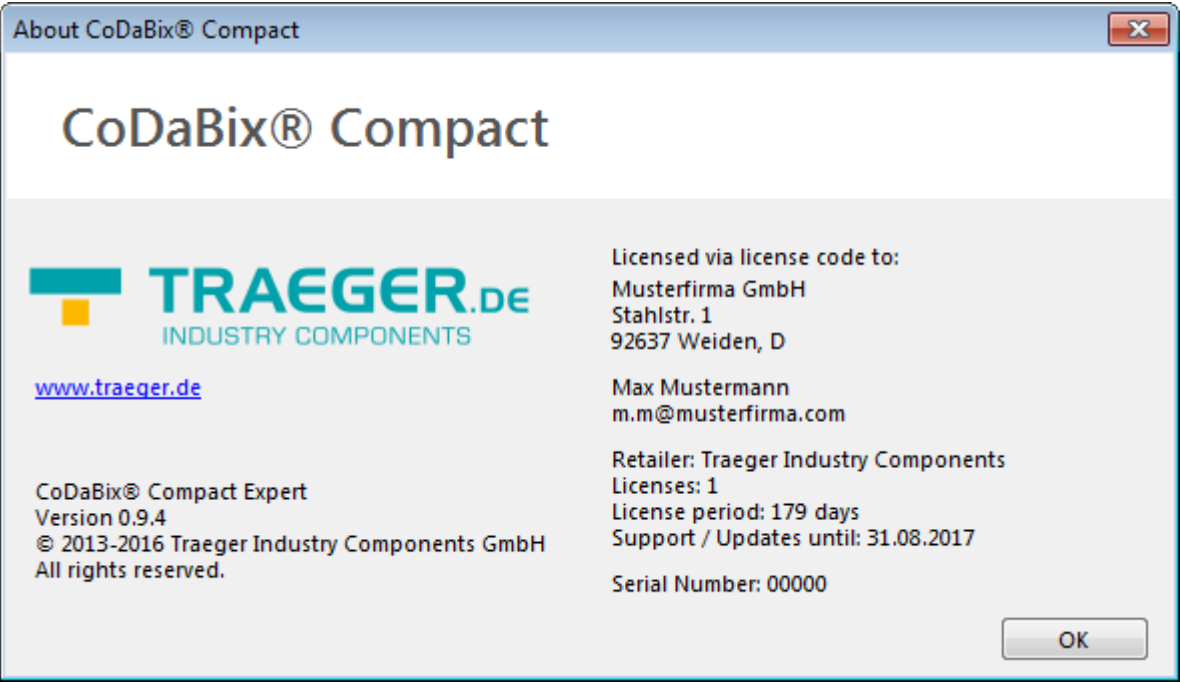
Icon	Description
	Opens the license dialog, in which you can insert, change or delete the license
	See <a href="#">Codabix Settings</a>
	Shows all information on the license and application

Licence Dialog



Field	Description
License Code:	Insert your license key here
Remove License	Remove the license from the computer
Copy Machine Code	Copies the machine code to the clipboard. This is required for the license creation.
OK	The entered license is stored on the computer. If “grayed out” no valid license key was entered.

About

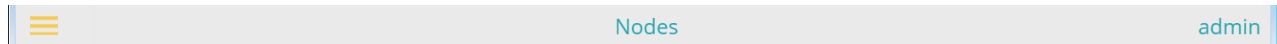


Left area: Information on the application  
Right area: Information on the last installed license

## (2) Working Area

The working area consists of an embedded browser (red border).  
This area is divided into 3 to 4 areas, depending on the choice of menu.

## (3) Title Bar



In the title bar the following actions are possible and the following data is displayed:

Field	Description
	Opens and closes the menu
Nodes	Display of the title of currently selected menu
admin	Display of the name of the user logged in

## (4) Toolbar

Below the title bar is the toolbar. It is automatically adjusted depending on which menu you are in and which item you have selected.

Folder Node:

Name	Display Name	Description	Value Type
JobName	JobName		String
Machine runni...	Machine runni...		Boolean

Datapoint Node:

Name	Display Name	Description	Value Type
JobName	JobName		String
Machine runni...	Machine runni...		Boolean
Temperature	Temperature		Single









User:

First Name	Last Name	Login Email	Phone Number
Demo	User	demo@user.org	+1
Max	Mustermann	m.m@email.com	+49123456789

User Group:

Name	Type
Test Group	A
Rotating Cutter	A
Injection molding	A






















Script Plugins:



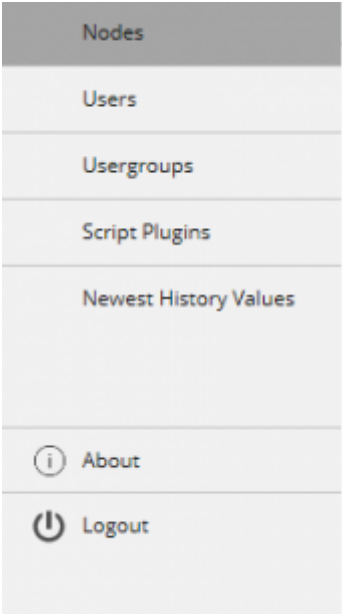
Name	Description	Editor Strictness Level	Script State	Current Script State
test	test	Low	Enabled	NotRunning

**General:**

Grayed out icons are not applicable to the selected item.

Icon	Description	Visible in
	Refresh view	all menu
	Codabix Folder Node	Folder Node, Datapoint Node
	Codabix Datapoint Node	Folder Node, Datapoint Node
	Codabix-linked Folder Node	Folder Node, Datapoint Node
	Codabix-linked Datapoint Node	Folder Node, Datapoint Node
	All icons with this symbol add a new item on the basis of an input dialog	Folder Node, Datapoint Node, User, User Group, Script Plugins
	Delete selected item	Folder Node, Datapoint Node, User, User Group, Script Plugins
	Cancel action	Datapoint Node, User, User Group, Script Plugins
	Saves the edited record or records	Folder Node, Datapoint Node, User, User Group, Script Plugins
	Access to the selected Folder Node or Datapoint Node	Folder Node, Datapoint Node
	Edit selected item	Datapoint Node, User, User Group, Script Plugins
	Add or delete user	User Group
	Add or deselect Folder Node or Datapoint Node	User Group
	Select or deselect the user group for the selected item	Folder Node, Datapoint Node
	create or delete a link	Folder Node, Datapoint Node
	Duplicate the selected item	Folder Node, Datapoint Node
	If you click on a Folder Node, all values of the Datapoint Node are updated. If a Datapoint Node is selected, only this value is updated	Folder Node, Datapoint Node
	Set Datapoint Node value	Datapoint Node
	If available, the historical data of the Datapoint Node is displayed	Datapoint Node
	Edit script	Script Plugin
	Log view of the selected script	Script Plugin

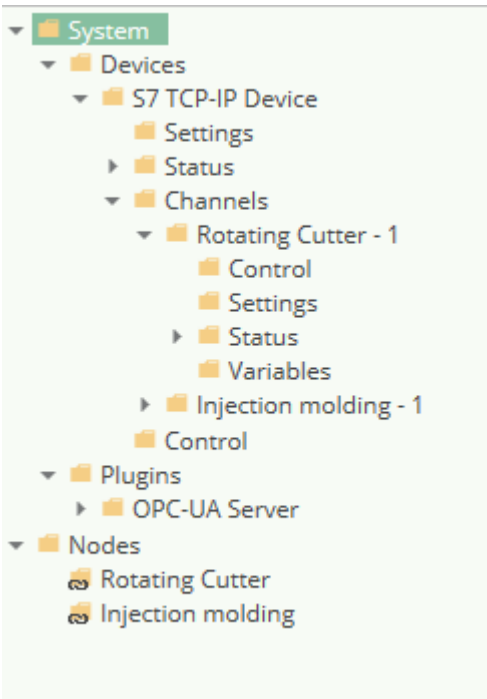
**(5) Menu**



By default, the menu is displayed permanently.

Field	Description
Nodes	Current start page. Edit Folder Nodes and Datapoint Nodes
Users	Edit user
User Groups	Edit user groups
Script Plugins	Lightweight JavaScript (TypeScript) plugin that Codabix extends on custom functions. Executed in a secure environment
Newest History Values	View the of the last registered and stored historical values. Data will only be displayed if a Folder or Datapoint Node has enabled storing historical data
About	Information about the configuration page and, when appropriate, Codabix
Logout	Logout user

(6) Node Tree





The available Folder Nodes are displayed in a tree structure.

Structure of the Node tree:

Node	Description	Purpose
System	System Nodes	Contains all Nodes that are necessary for the operation of Codabix. Most Nodes and variables are not editable here. Fixed Folder Nodes are: -System -Device -Plugins -Settings -Status -Channels -Variables -Control
Device	Device Nodes	Contains all Nodes of the devices registered in Codabix
S7 TCP-IP Device	S7 Device Plugin Container	Contains all dataa of the plugins required for Codabix and the user, for example status, channels, etc.
Settings	Settings of the plugin	Possible Settings of the device
Status	Status of the plugin	General information on the plugin, for example error, plugin started
Channels	Node for all channels	Contains all defined channels of the plugin
Channel e.g. Rotating Cutter - 1	Defined channel in the plugin	Contains - general settings of the channel - Status of the channel - all defined Datapoint Nodes
Plugin	Plugin Nodes	Root node for all plug-ins, similar to Devices
Nodes	Root node for user defined nodes	Contains all user defined nodes and variables

## (7) Data Area

Name	Display Name	Description	Value Type	Path	Actual Value	Status code
Devices	Devices					---
Plugins	Plugins					---

1

◀
▶

◀
1
▶
2
3

1-2 of 2

Number	Description
1	Data of the selected menu or Folder Node is displayed here

Number	Description
2	Navigation bar, here you switch views of the respective data. Possible navigation: top of page previous page next page bottom of page
3	Number of all records

## Nodes

### What are Nodes?

*"The OPC information model is a so-called Full Mesh Network based on nodes. These Nodes can include any kind of meta information, and are similar to the objects of object-oriented programming (OOP). A Node can have attributes for read access (DA, HDA) [...]. Nodes hold process data as well all other types of metadata. The OPC namespace contains the type model."*

Source: wikipedia.org

Codabix is similarly structured as specified in the OPC UA. Depending on the Node type, a Node has different functions and properties.

A distinction is made between:

- Folder Node
- Datapoint Node
- Link Node
- Directory Node

### Folder Nodes

A Folder Node can also be referred to as a root node, since it may provide additional Folder or Datapoint Nodes but it itself may may not provide any value.

Set properties are automatically applied to the Child Nodes, except when they define a separate condition.

### Datapoint Nodes

A Datapoint Node in Codabix is like a variable with additional attributes (properties).

A Datapoint Node must not contain Child Nodes.

### Link Nodes

Link Nodes have only the properties they have in common.

A Link Node points to another Node that has already been created. This way it is possible to be assemble an individual record from several devices, Folder and Datapoint Nodes.

## Directory Nodes

The Directory Node is a Folder Node that directly points to a physical directory and automatically represents the substructure.

# Node Properties

## Common Properties

Field	Description	Data type
Lokal ID	Unique ID in the Codabix System e.g. used in access through the REST interface	Long
Global ID	Unique ID across Codabix	GUID
Name	Unique name within the Parent Node, for example for the OPC UA addressing, JSON Interface e.g. job number	String
Display Name	Display name for the user e.g. job number	String
Path	e.g.: PLC address (according to S7 Syntax, DB1000.DBB 500, Word) OPC UA Node (3:AirConditioner_1.State) File path (R:\\MachineData) <b>NOTE:</b> The path is not dealt with at the <i>Folder</i> type	String
Max Value Age (ms)	Relevant for a synchronous read. When a synchronous read operation doesn't specify a <b>time to live</b> , this value (in milliseconds) is used instead. This means that if the currently set node value is <b>not</b> older than this age, it will be returned directly; otherwise, the value will be read from the device.	Integer

## Node Types

When creating a Folder Node you can decide between the types:

- Folder
- Directory

### Type: Folder

The Node is a regular Folder Node with no further special properties.

### Type: Directory

The Node is a Folder Node and represents a physical registry in the data system. The Path property of the Node states the registry path (it can contain environmental variables). When browsing Nodes (e.g. via the web configuration) subdirectories are automatically created and aligned in the path property as corresponding Folder Nodes (type **directory**) and Datapoint Nodes (type **file**) with the related file path.

**Note:** The specified path is subject to the **Access Security** restrictions that have been

defined in the [Codabix Project Settings](#).

**Note:** On Windows 10 Version 1511 and older (and Windows Server 2012 R2 and older), e.g. on Windows 7, the maximum file path length is limited to 260 characters (MAX\_PATH). On Windows 10 Version 1607 and higher (as well as Windows Server 2016 and higher) longer path names can be used. However, for this you will need to enable the setting “Enable Win32 long paths” in the Windows Group Policy, see [Enabling Win32 Long Path Support](#).

## Datapoint Nodes

Additionally to the previously mentioned [properties](#) the Datapoint Node has the following properties:

Name	Description	Data Type
Description	Description of the datapoint	String
Location	Used during evaluation of conditions. Used to display e.g. of the origin of the message. Arbitrarily assignable	String
Value Types	Data type of the node, further information see table <a href="#">Value Types</a>	enum
Min Value	Minimum value, is required for the evaluation of conditions	Double
Max Value	Maximum value, is required for the evaluation of Conditions	Double
Hysteresis	Threshold value used in the evaluation of conditions when e.g. minor temperature fluctuations are to be compensated.	Double
Scaling Factor	Factor that is added to the currently read value	Double
Scaling Offset	The value is added to the currently read value	Double
Unit	Unit of the value e.g. °C	String
Precision	Number of decimal digits	Number
History Options	Specifies how history values are to be captured and written into the database for this node. <b>No:</b> No history values are being captured for this node. <b>Yes; only on Value Change:</b> When a <b>History Interval</b> is set for the node, history values are being captured by the <i>History Timer</i> for this node from the actual value at the specified history interval; otherwise, history values are being captured when writing a value to the node. A captured value is inserted into the database only if it differs from the last written history value for that node. Additionally, in case of a device variable, a subscription is being created for this node. <b>Yes:</b> When a <b>History Interval</b> is set for the node, history values are being captured by the <i>History Timer</i> for this node from the actual value at the specified history interval; otherwise, history values are being captured when writing a value to the node. Captured history values are inserted into the database. Additionally, in case of a device variable, a subscription is being created for this node.	Enum
History Interval	The interval at which history values shall be captured.	Enum
History Resolution	Specifies the resolution to which numeric historic values are rounded when being captured.	Double

**Note:**

- To determine if a captured history value differs from the last history value (for the setting **Yes; only on Value Change**) only properties Value and Status are considered, but not property Timestamp (CreationTimestamp).
- When the **History Options** of a node are changed from **No** to **Yes; only on Value Change** while Codabix is running, the first captured history value after this change will be written to the database in every case, even when it is not different from the last written history value for that node.
- When the **History Options** of a node are set to **Yes; only on Value Change** and Codabix is shut down and later restarted, the first captured history value of the node after the restart will be written into the database **in every case**, even when it is not different from the last written history value for that node.
- When the **History Options** of a node are set to **Yes; only on Value Change** (or no **History Interval** is specified), the *Creation Timestamp* of the actual value will be kept when capturing the history value; otherwise, the *Creation Timestamp* will be set to the current time.
- In a **Script**, the setting **Yes; only on Value Change** corresponds to the value `codabix.NodeHistoryOptions.Subscription | codabix.NodeHistoryOptions.ValueChange` (contains `codabix.NodeHistoryOptions.Active`), and the setting **Yes** corresponds to the value `codabix.NodeHistoryOptions.Subscription` (contains `codabix.NodeHistoryOptions.Active`)
- History Values are inserted into the database **asynchronously** in the background by the History Value Dispatcher. This means that when you write a value to a node (where no *History Interval* is set), even though the write operation finished, the captured history value might not yet be in the database. This is especially true if you set the *DB Update Mode* to **Restricted** in the Codabix Settings, because captured history values are then inserted into the database only every 5 seconds.

## Value Types

Codabix provides the following value types:

Name	According Data Type	Length in Bits
Blob	binary, optional statement of file name and MIME type possible	arbitrary
String	String	arbitrary
Null	without value	0
Boolean	Bool	1
SByte	signed Byte	8
Byte	unsigend Byte	8
Int16	signed Integer	16
UInt16	unsigned Integer	16
Int32	signed Integer	32
UInt32	unsinged Integer	32
Int64	signed Integer	64
UInt64	unsigned Integer	64
Single	single Floating Point	32
Double	double Floating Point	64

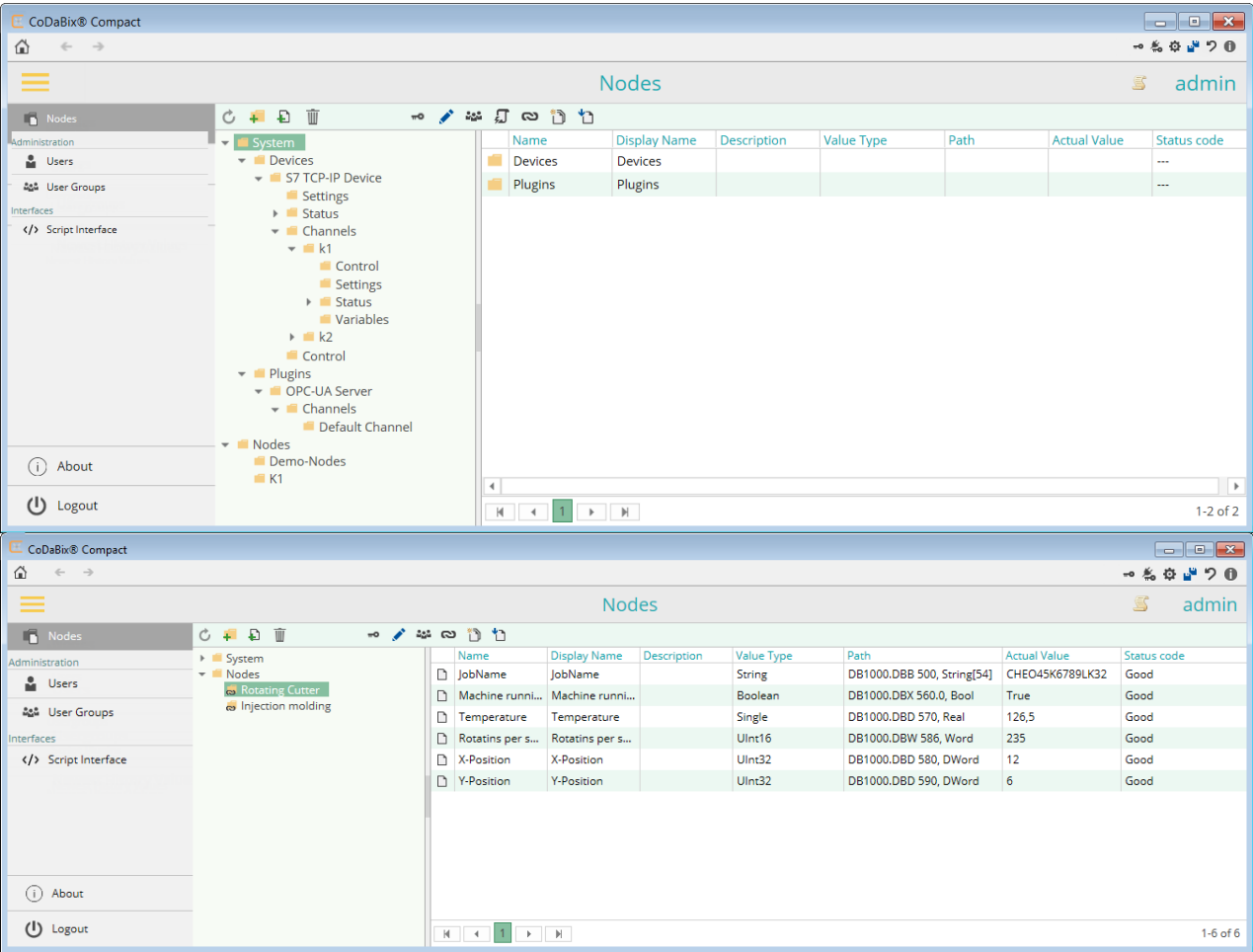
Name	According Data Type	Length in Bits
File	The Node represents a physical file that can be accessed via OPC UA Client or via HTTP Access URL. The "Path" property of the Node contains a file path. <b>Note:</b> The specified path is subject to the <b>Access Security</b> restrictions that have been defined in the <a href="#">Codabix Project Settings</a> .	

An array can also be created from each data type. Exceptions are the data types:


- Null
- File

The length of the array is decided during the writing of the Datapoint Node.

Node View



Field	Description
Name	internal unique name of the Folder Node
Display Name	Display name of the Node
Description	Description of the Node
Value Type	Data type or kind of data of the Node. See <a href="#">Properties</a>

Field	Description
Path	e.g. PLC Address, OPC Node, data path to the node. See <a href="#">Properties</a>
Actual Value	Value after click on  . Live data is not displayed automatically. See <a href="#">(4) Toolbar</a>
Status	Current status of the Node. Possible status: Good Bad

Status at the device:

	Name	Display Name	Description	Value Type	Path	Actual Value	Status code
System	Report	Report	Provides report data about the...				---
Devices	Code	Code	The code of the status.	Int32		0	Good
S7 TCP-IP Device	Severity	Severity	The severity of the status.	String		Moderate	Good
Settings	Text	Text	The text of the status.	String		OK	Good
Channels							
Rotating Cutter - 1							
Control							
Settings							
Status							
Variables							
Injection molding - 1							

At System / Devices / Device Plugin / Channels / Channel / Status you will find the status for the selected channel with the associated message.

When the code is  $\geq 0$  the channel is running ok.

## Node Structure

There are System Nodes and User Nodes.

System Nodes are fixed, predetermined structures that are necessary for the operation of Codabix and its plugins.

User Nodes are created and managed by the user according to his needs.

Structure of System Nodes:

- System
  - Devices
    - Device Plugin
      - Settings (*settings of plugin*)
      - Status (*status of plugin*)
      - Channels (*defined channels*)
        - Channel 1 (*defined channel*)
          - Control (*Datapoint Nodes to control the channel*)
          - Settings (*settings of the channel*)
          - Status (*status informations of the channel*)
          - Variables (*defined variables*)
        - Channel 2
          - Control ...
      - Control (*Datapoint Nodes to control the plugin*)
  - Plugins
    - Plugin
      - Settings
      - Status
      - Channels

- Channel 1
  - Control
  - Settings
  - Status
  - Variables
- Channel 2
  - Control ...
- Control

Structure of User Nodes:



- Nodes
  - User Folder Node
    - User Datapoint Node
    - User Folder Node
    - User Datapoint Node
    - User Linked Node
  - User Folder Node...

See also [\(6\) Node Tree](#).

## Create a Node

You can create your own Nodes in the *Nodes* view under Nodes.

In order to be able to create a Node, you always have to select the superior Node first. You can find the following items for adding Nodes in the toolbar:

	Add Folder or Directory Nodes
	Add Datapoint Nodes

### Add a Folder Node

Add new Folder Node
✕

Name:

Display Name:

Node Type:

Folder ▼

Path:

Refresh Interval:

<Inherit> ▼

Max Value Age (ms):

✕

History Value Interval:

▼

✓ ✕

- Enter name e.g. Press
- Enter display name e.g. Press



- Node type: select “Folder”
- when appropriate set intervals
- create a Node with ✓

### Add a Directory Node

Add new Folder Node
✕

Name:

Display Name:

Node Type:

Path:

Refresh Interval:

Max Value Age (ms):

History Value Interval:

- enter name e.g. DirectoryR\_MData
- enter display name e.g. machine data
- path: enter directory path e.g. R:\\MachineData
- when appropriate set intervals
- create a Node with ✓



In order for you to be shown folders and files you have to refresh the view, e.g. by clicking “User” and then clicking “Nodes”. Depending on the directory size it is possible that you will not see all files and subdirectories at once.

### Add a Datapoint Node

Add new Datapoint Node
✕

Name:

Display Name:

Description:

Location:

Value Type:

Path:

Min Value:

Max Value:

Hysteresis:

Scaling Factor:

Scaling Offset:

Unit:

Precision:

Publishing Level:

Refresh Interval:

Max Value Age (ms):

History Value Interval:

Has History Values:

String

String

Blob

Null

Boolean

SByte

Byte

Int16

UInt16

- enter name e.g. Pressure (bar)
- enter display name e.g. Pressure (bar)
- select value type e.g. Single
- when appropriate enter boundaries, scales, historical data (as e.g. Min Value = 20,5, Scaling Factor = 2,398 ...)
- create a Node with ✓

The Node could look like this:

Edit Variable

Name:

Display Name:

Description:

Location:

Node Type:

Path:

Min Value:

Max Value:

Hysteresis:

Scaling Factor:

Scaling Offset:

Unit:

Precision:

Publishing Level:

Refresh Interval:

Max Value Age (ms):

History Value Interval:

History Options:

✓

✕

## Link Nodes

For the selected item, the following dialog will be displayed:

Virtual Link

System

Devices

S7 TCP-IP Device

Control

Settings

Status

Channels

Rotating Cutter - 1

Control

Settings

Status

Variables

Injection molding - 1



Plugins

Nodes

✓

✕

The selected item will be linked to the selected Node in the Node tree. The following icons are possible:

	linked Folder Node
	linked Datapoint Node

If a Folder Node is linked onto a Datapoint Node, it is automatically converted to a Folder node.

Node Access

For the selected item the following dialog will be displayed:

Edit Node

Local ID:

160

Global ID:

e6c5e15e-33be-4b30-b313-38a497f28b1d

Absolute Node Path:

/Nodes/Rotating Cutter

Token:

160:BAuNH3rQeqZCdTM57zxjKnSWW5I4wiKG

Access URL:

http://localhost:8181/api/get?token=160%3aBAuNH3rQeqZCdTM57zxjKnSWW5I4wiKG

X

Field	Description
Local ID	Unique Codabix internal ID of the selected item. See <a href="#">Abbreviations / Glossary</a>
Global ID	The GUID unambiguously identifies the Node beyond system, e.g. Use of Codabix at several locations that are to form a record together. <a href="#">Abbreviations / Glossary</a>
Absolute Node Path	Path for accessing the Node. In linked datapoints the path display to the origin of the datapoint is displayed.
Token	Act like a password for JSON to gain access to the Node.
Access URL	Read the actual value via HTTP directly and display it in a browser window.

Add / Delete Node Access to a User

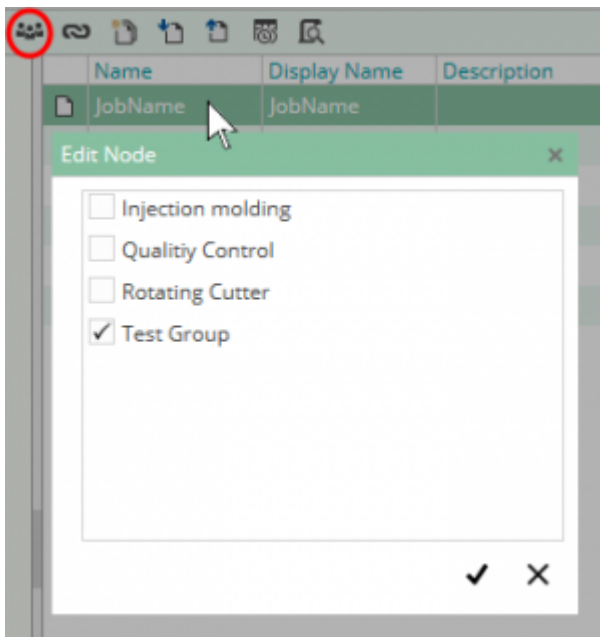
**A node access can only be added or removed to / from a user group.**

There are two ways to add a Node to the user via the user group:


- via the Node view

- via the user group directly

### Add / Remove via the Node View

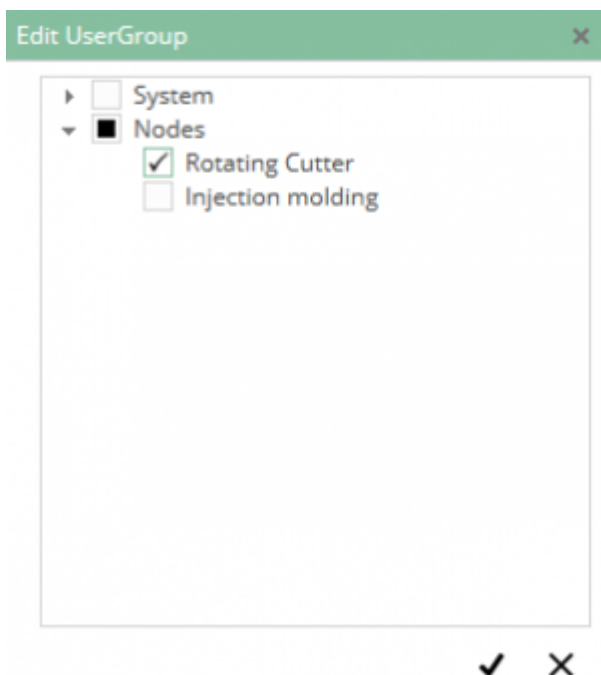


If a Folder Node is selected, automatically all the Nodes lying underneath inherit the selected user group.

The user group is not displayed in the underlying nodes, if you click .

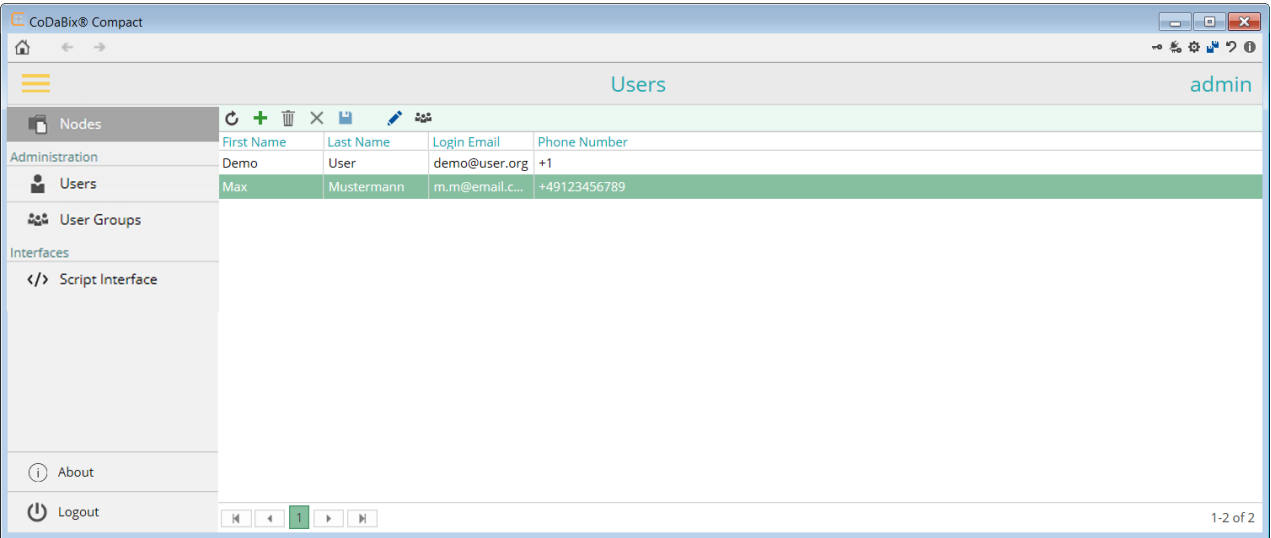
Thus, you have the option to explicitly set the user group to a specific Folder or Datapoint Node, if for example, the Parent Node loses access.

### Add / Remove via the User Group



All previously selected Folder and Datapoint Nodes are displayed by hooks. When you select a Folder Node, all nodes below will be automatically selected or deselected.

# User



Here you create the users who shall have access to the dataa provided by Codabix. Only the administrator has the right to access the configuration page.

Functions:

- add / edit / delete users
- asign user group(s)

## Add Users

Add new User

First Name:

Max

Last Name:

Mustermann

Login Email:

m.m@email.com

Phone Number:

+49123456789

Login Password:

.....

.....

✓

✗

Field	Description
First Name	First name
Last Name	Last name
Login Email	Email address: Can occur only once
Phone Number	Phone number: Can occur only once
Login Password	Password with repetition

## Edit Users

Edit User

First Name:

Max

Last Name:

Mustermann

Login Email:

m.m@email.com

Phone Number:

+49123456789

Login Password:

\*\*\*\*\*

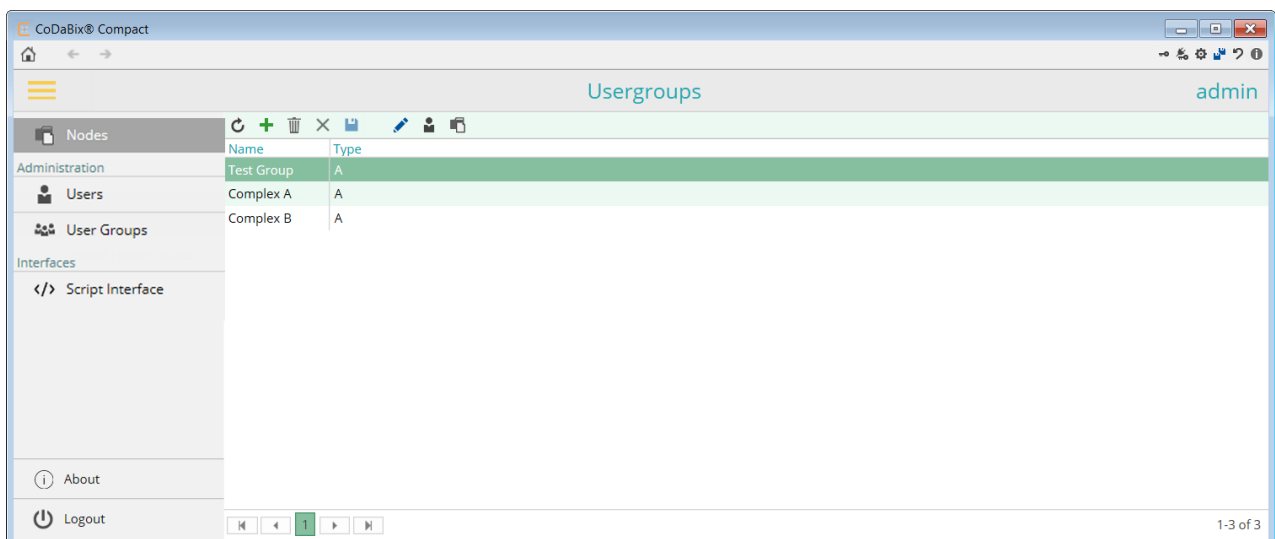
\*\*\*\*\*

✓

✕

The same settings as in [Add Users](#) apply.

## User Group



A Node cannot be directly assigned to a user. You always need a user group. This simplifies the administration of the Node access.

Functions:

- Add / edit/ delete user group
- Add user to user group
- Add / delete Node

### Add User Group

Add new UserGroup

Name:

Type:

A

A

B

Field	Description
Name	Display name of the group
Type	A or B, currently not used

At the moment the usertype is not used.

Edit User Group

Edit UserGroup

Name:

Test Group

Type:

A

✓

✕

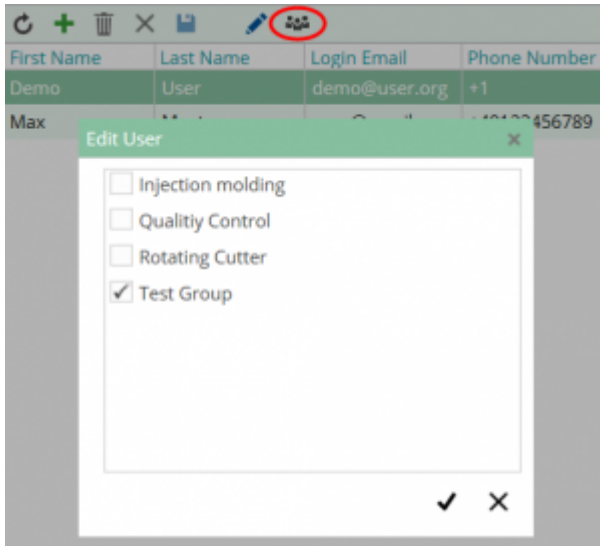
The same settings as in [Add User Group](#) apply.

Add User to User Group

There are two ways to add a user to user group:

Via the user menu:

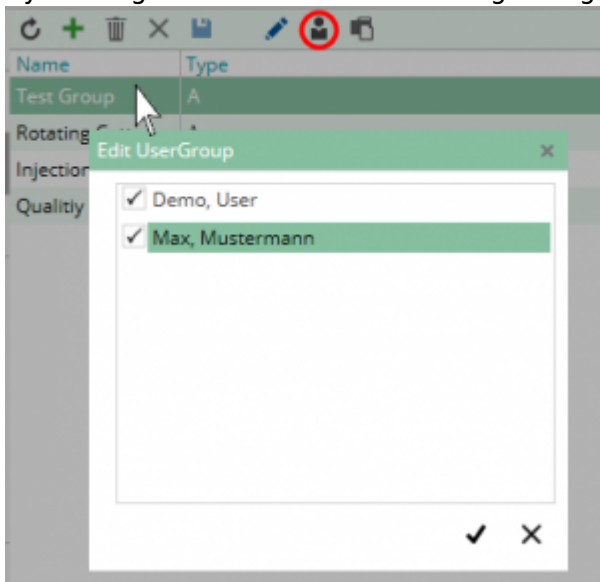
For the selected item the following dialog will be displayed:



Here you can add or deselect the user groups for the selected item.

### Via the user group:

By clicking the user icon the following dialog is displayed for the selected user group:



All available users can be selected or deselected here.

## Plugins

Codabix can be extended to add new functions by plugins. These plugins are grouped by their classification. Depending on the category, the plugin offers a special range of services, organizations and Nodes. Some plugins can provide configuration files and additionally, as required by the circumstances, include a configuration application.

### Device Plugins

All device plugins are using the Codabix Device Model. Each device that is provided is registered and managed by the Codabix Device Manager.



See [Device Plugins](#).

## **S7Device**

Connection to Siemens SIMATIC S7.

See [S7 Device Plugin](#).

## **Exchange Plugins**

All Exchange Plugins are using the Codabix Storage Model. Each Exchange Plugin that is provided is registered and managed by the Codabix Device Manager.

See [Exchange Plugins](#).

## **Database**

Codabix supports connections to databases.

More information can be found at [Database Plugin](#).

## **CSV**

For further information about the handling with CSV files see [CSV Exchange Plugin](#).

## **Interface Plugins**

Codabix contains an API interface.

For more informations see [Interface Plugins](#).

## **REST**

The REST interface allows access to the Node Codabix via HTTP request formatted as a JSON object.

For more informations see [REST Interface Plugin](#)

## **Script Plugins**

Script plugins extend Codabix with additional functionalities via JavaScript editor. They are executed in a secure environment.

See [Script Interface Plugin](#).

### **OPC UA Server Interface**

Codabix contains an OPC UA Server for the exchange of data.

More information under [OPC UA Server Interface Plugin](#).

From:

<https://codabix.de/> - **CoDaBix®**

Permanent link:

<https://codabix.de/en/config>

Last update: **2023/11/03 13:10**