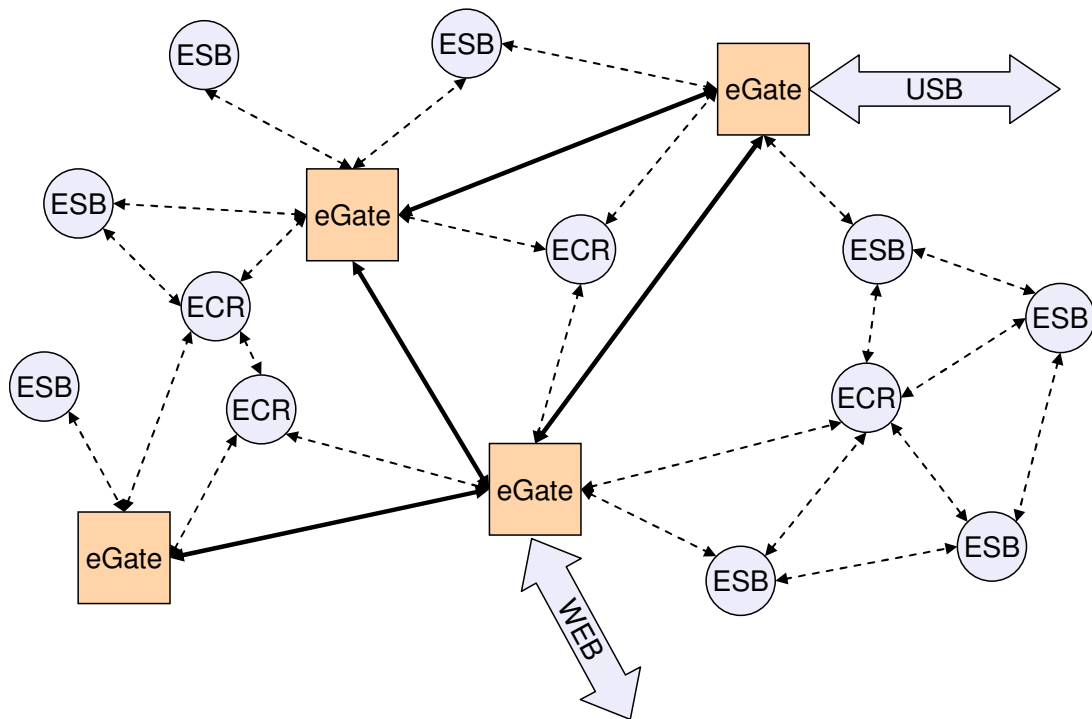


# User Guide (for ScatterViewer)

A platform for teaching & prototyping  
wireless sensor networks



[scatterweb.mi.fu-berlin.de](http://scatterweb.mi.fu-berlin.de)

Computer Systems & Telematics  
Freie Universität Berlin, Germany



# Content

<b>1</b>	<b>Introduction .....</b>	<b>5</b>
<b>2</b>	<b>Connection Dialog.....</b>	<b>6</b>
	2.1 IP Connection .....	6
	2.2 Serial Port Connection.....	7
<b>3</b>	<b>Node .....</b>	<b>8</b>
	3.1 Node Tables .....	8
	3.2 Nodes Tree.....	9
	3.3 Identification .....	9
	3.4 LEDs.....	10
	3.5 Reset Node.....	10
	3.6 Beeper.....	11
	3.7 Sensors .....	11
	3.8 Announce & Firmware Flags .....	12
	3.9 Transceiver.....	12
	3.10 Power Supply .....	12

<b>4</b>	<b>Status .....</b>	<b>13</b>
	4.1 ID and reset.....	13
	4.2 Scattergate info .....	14
	4.3 Loaded to EEPROM .....	15
	4.4 Additional Information.....	16
	4.5 Firmware Flags.....	16
<b>5</b>	<b>Console .....</b>	<b>17</b>
<b>6</b>	<b>Log.....</b>	<b>18</b>
	6.1 LogFile.....	18
	6.2 LogPrint .....	18
	6.3 Tabs of the single sensor node even.....	19
<b>7</b>	<b>Graph.....</b>	<b>20</b>
	7.1 Tabs of the single sensor nodes.....	20
	7.2 The Graph .....	21

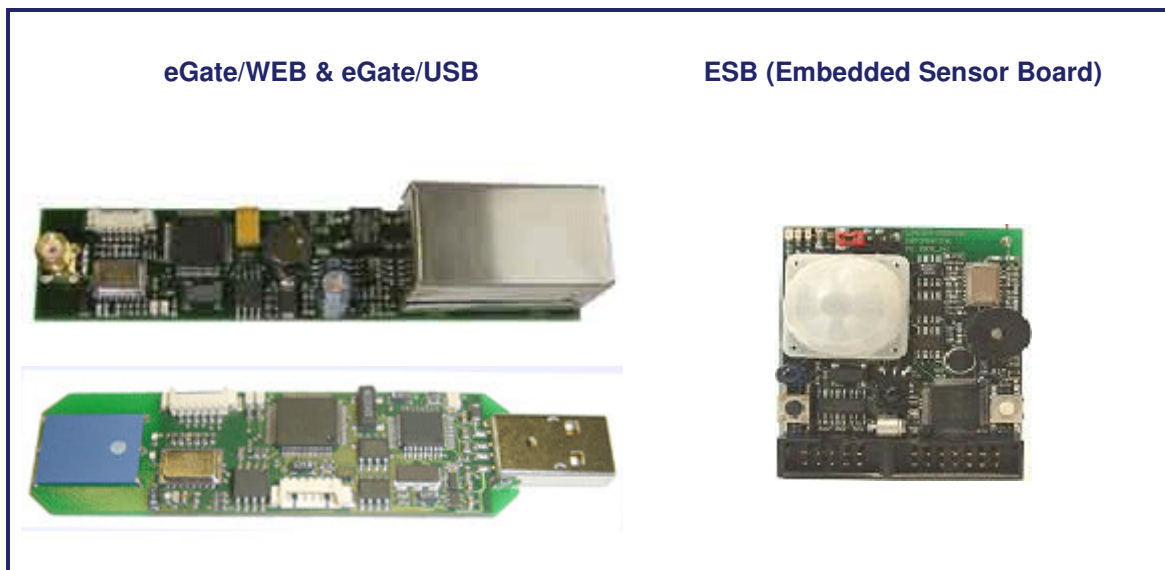
# 1 Introduction

The ScatterViewer permits:

- Data Logging
- Node Managing
- OTA Flashing
- Net-Scanning
- ScatterRouting

The ScatterViewer supports three ways of using:

- As embedded applet: upload the ScatterViewer on the eGate/WEB and use it browser-based
- As executable program: run the ScatterViewer local to gain access to eGate/USB (via COM) or eGate/WEB (via IP)
- As hosted applet: host the ScatterViewer on the internet without uploading on the eGate/WEB

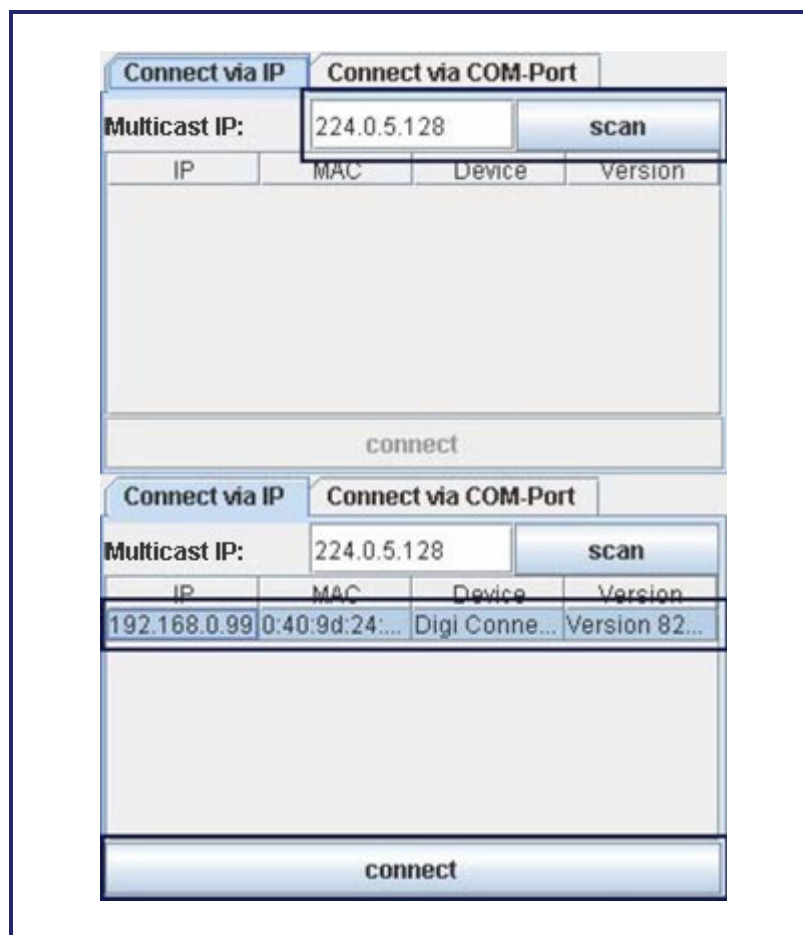


## 2 Connection Dialog

If you don't use the ScatterViewer as an Embedded Applet, you must choose a connection. We provide an opportunity to run a search for the discovery of the eGate/WEB. You can also connect to an eGate/USB over a serial port (WinXP/2000).

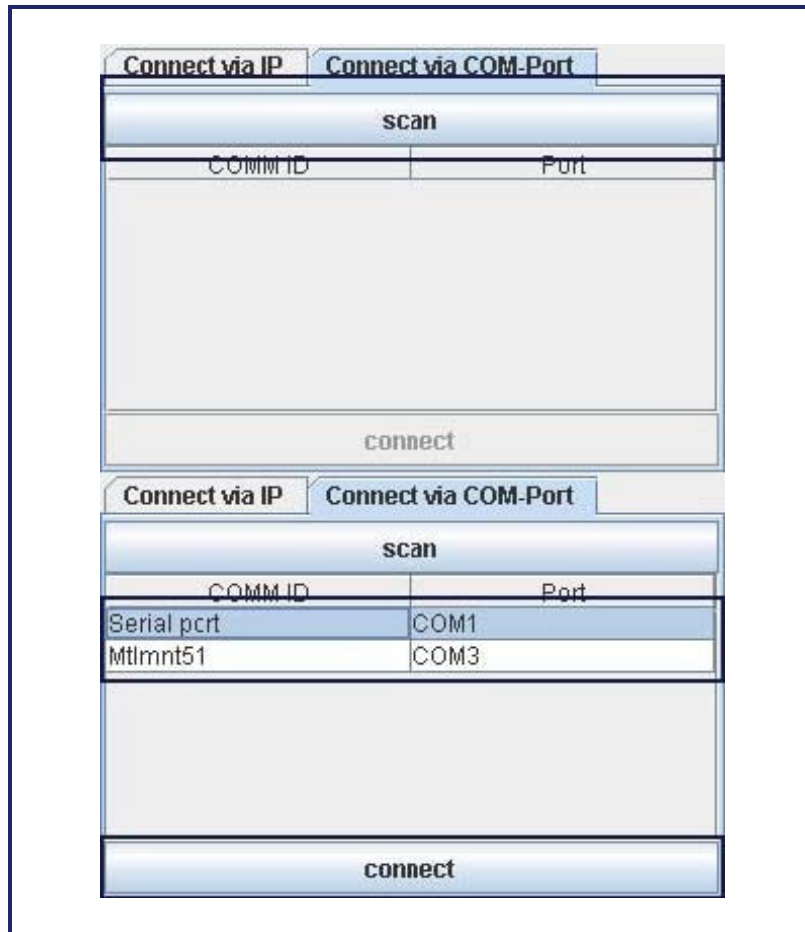
### 2.1 IP Connection

You should use the given IP address. Press the scan button to search for devices. After the search all located devices will appear in the table and the connect button will be activated. Choose your device and click on the connect button to start the ScatterViewer.



## 2.2 Serial Port Connection

This connection mode is only applicable on Windows XP and 2000 System because the software uses the Windows registry and loads a java com-port driver. Press the scan button to search for devices. After the search all located devices will appear in the table and the connect button will be activated. Choose your device and click on the connect button to start the ScatterViewer.



### 3 Node

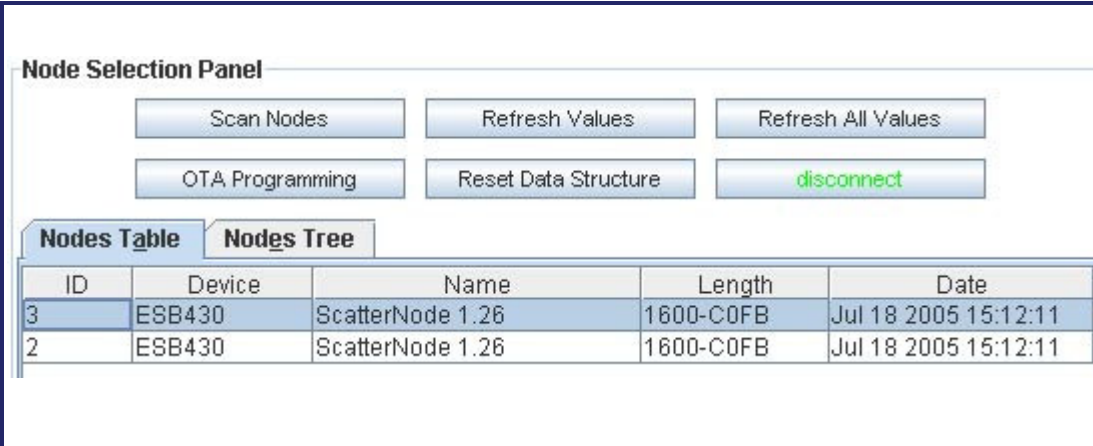
The node tab offers an overview of all reachable nodes. All located nodes will appear in the table. The software removes automatically all non-reachable nodes after a timeout and a non-answering ping.

Using:

- Press the scan button to rescan the network.
- The Refresh Values button loads the values of the selected node again and the Refresh All Values button loads the values of all nodes.
- Press the OTA Programming button to flash the selected node over the air with the image of the eGate.
- The Reset Data Structure button resets the internal data structure and scans the network again.
- The disconnect button and connect button respectively enables you to disconnect and reconnect respectively to the eGate.

There are two ways available to display the nodes. Both views enable you to read the sensor values and to change all node properties. The node tab consists of the following segments:

#### 3.1 Node Tables

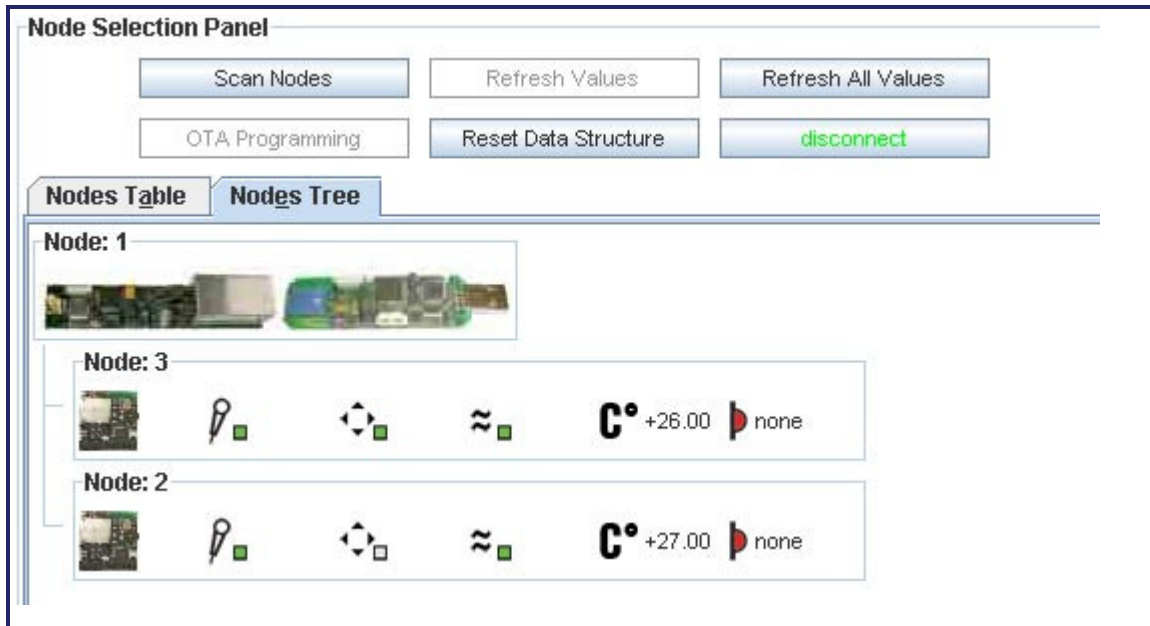


The screenshot shows a software interface titled "Node Selection Panel". It contains several buttons: "Scan Nodes", "Refresh Values", "Refresh All Values", "OTA Programming", "Reset Data Structure", and "disconnect" (highlighted in green). Below the buttons are two tabs: "Nodes Table" and "Nodes Tree". The "Nodes Table" tab is active and displays a table with the following data:

ID	Device	Name	Length	Date
3	ESB430	ScatterNode 1.26	1600-C0FB	Jul 18 2005 15:12:11
2	ESB430	ScatterNode 1.26	1600-C0FB	Jul 18 2005 15:12:11

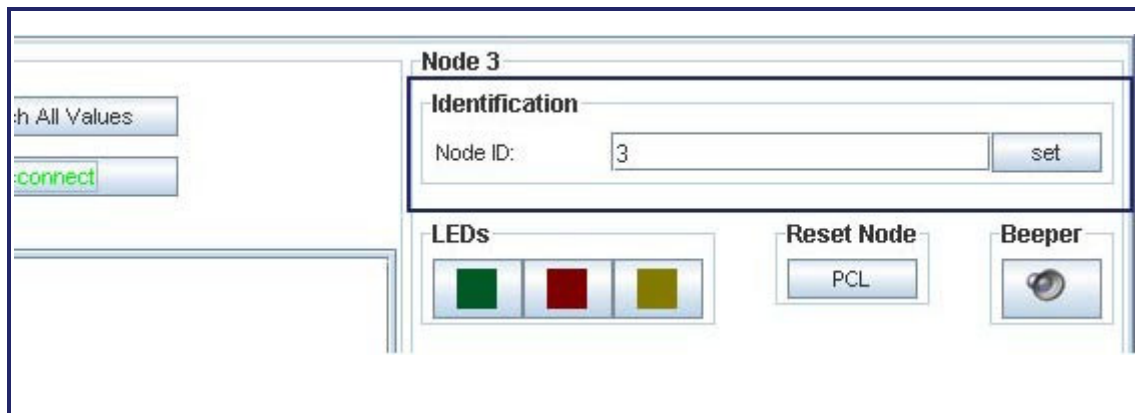


### 3.2 Nodes Tree



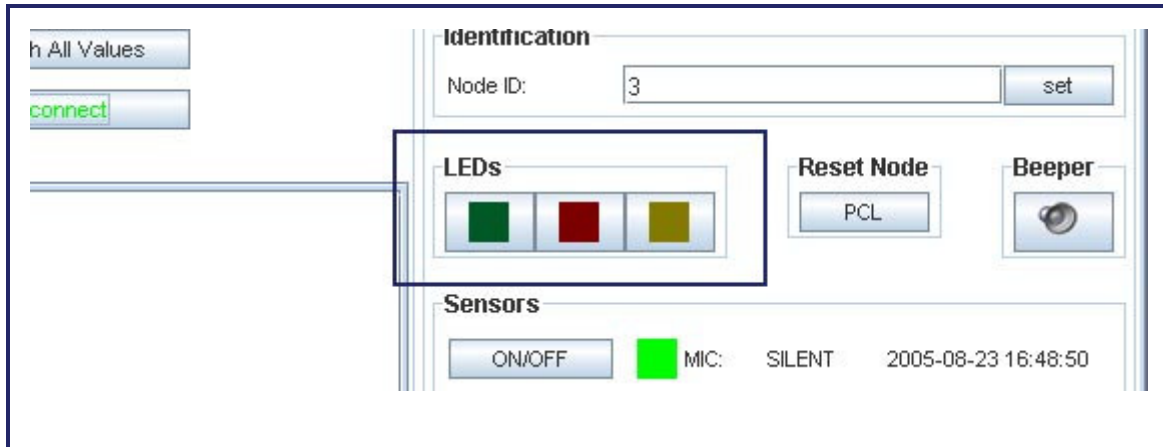
### 3.3 Identification

You can change the ID of the selected node and set it with a click on the [set](#) button.



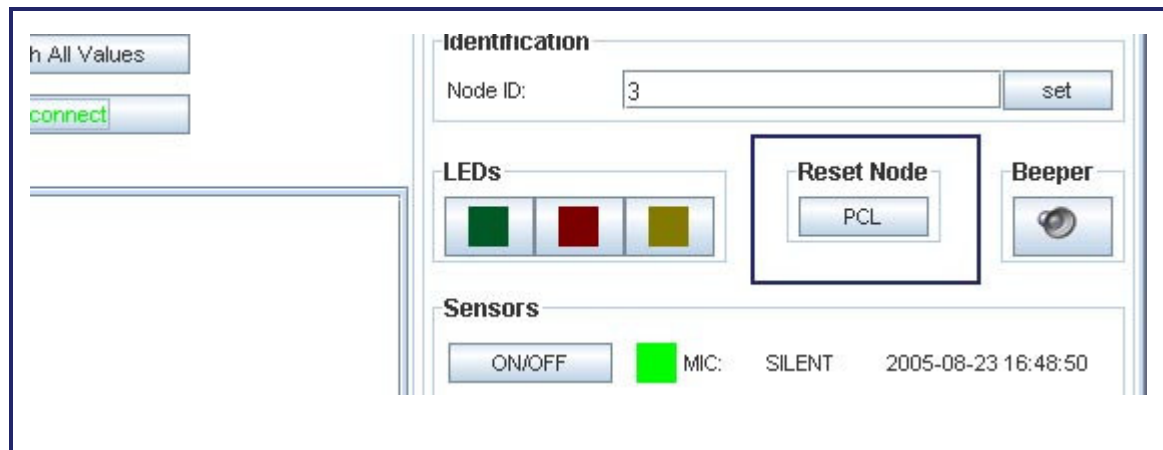
### 3.4 LEDs

You can switch the LEDs of the selected node with a click on the corresponding picture.



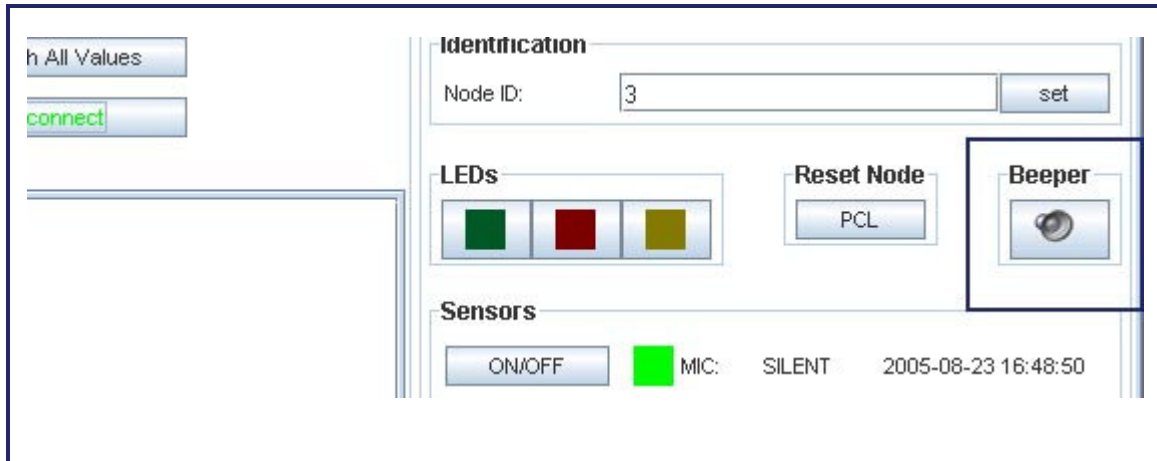
### 3.5 Reset Node

You can reset the selected node with a click on the PCL button.



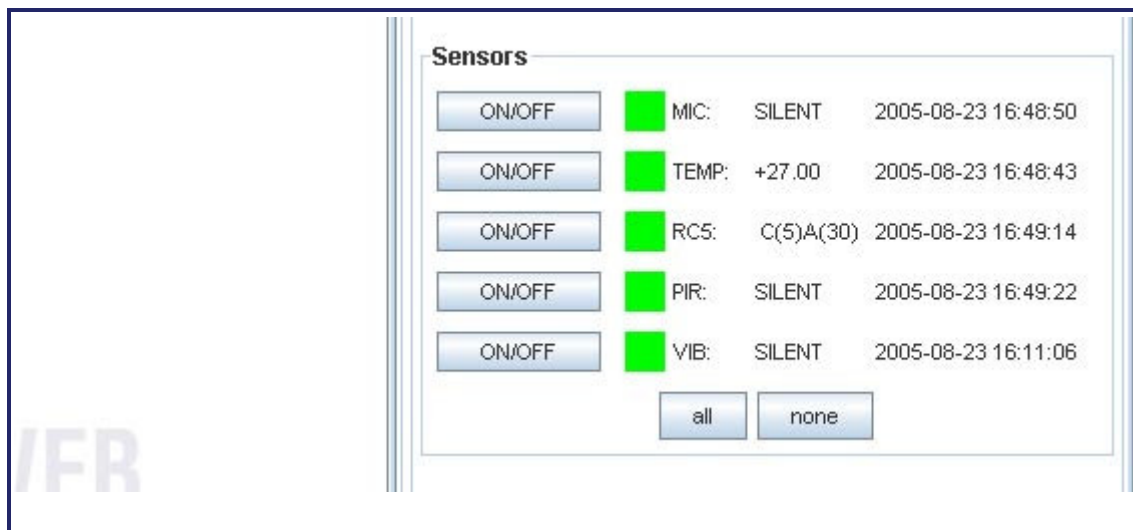
### 3.6 Beeper

You can switch the beeper of the selected node with a click on the beeper button.



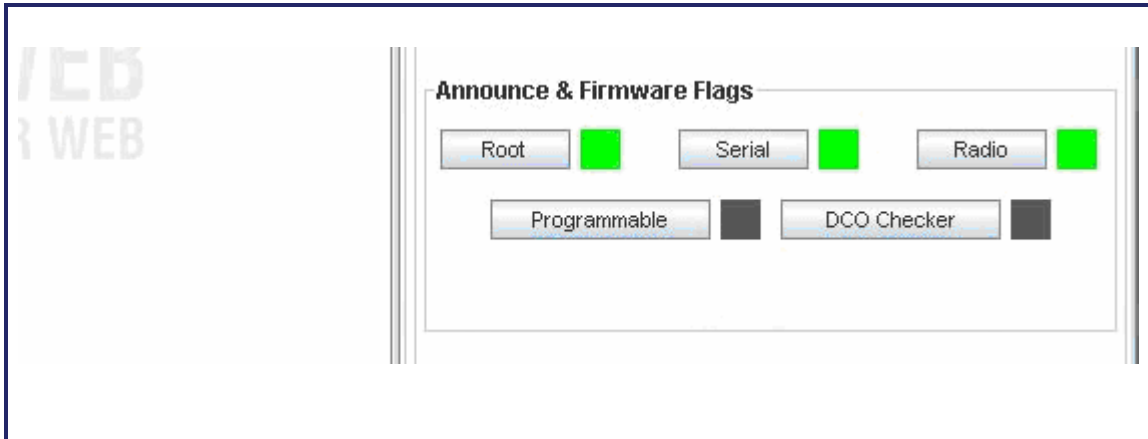
### 3.7 Sensors

You can switch the sensors of the selected node with a click on the corresponding button. The LEDs show you which sensors are switched on.



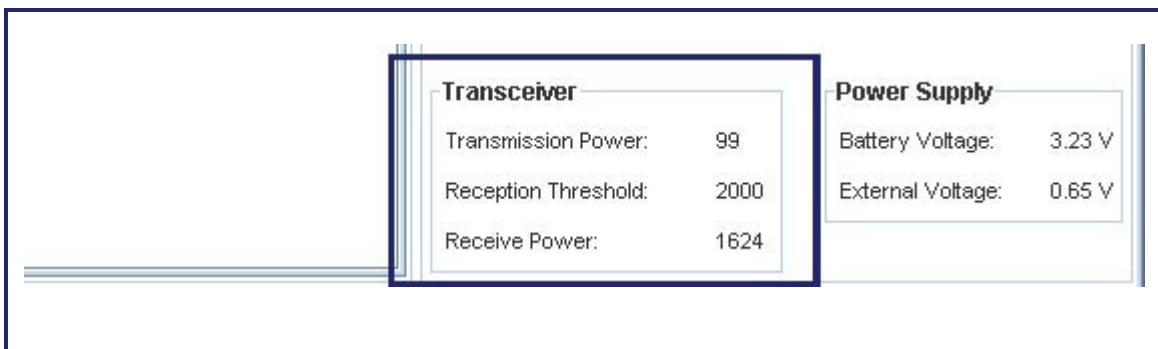
### 3.8 Announce & Firmware Flags

The Root, Serial, and Radio buttons enable you to set the announce flags. The Programmable and DCO Checker buttons enable you to set the firmware flags. The LEDs show you which flags are set.



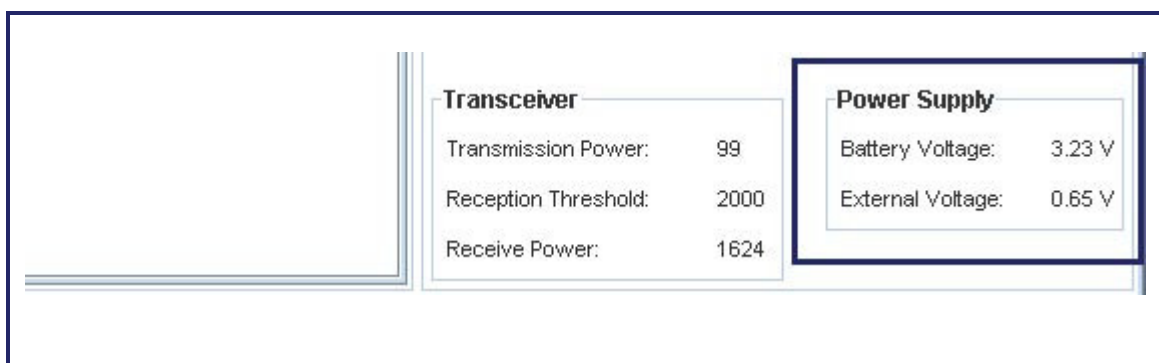
### 3.9 Transceiver

This panel shows you all needed information about the transceiver like the transmission power (value range between 00 and 99), the reception threshold (value range between 0000 and 4096), and the receive power.



### 3.10 Power Supply

This panel shows you all needed information about the power supply like the battery voltage and the external voltage.



## 4 Status

The status tab combines all required information about the eGate. The status tab consists of the following segments:

### 4.1 ID and reset

You can change the ID of the eGate and set it with a click on the **set** button. You can also reset the eGate with a click on the **PCL** button.

The screenshot displays the 'Status' tab interface for an eGate. It is divided into several sections:

- ID and reset:** A section with a text input for 'Node ID' containing the value '1', a 'set' button, and a 'PCL' button.
- Scattergate info:** A table with the following data:

ID	Device	Name	Length
1	USB430	ScatterGate 1.26	1600-B88E
- Loaded to EEPROM:** A table with the following data:

Device	Name	Length	Date
ESB430	ScatterNode 1.26	1600-C10D	Feb 15 2005 12:06:52

Below the table are two buttons: 'Load to EEPROM' and 'Clear EEPROM'.
- additional information:** A list of parameters:
  - Firmware Flags: 07
  - Transmit Power: 255
  - rx Receive Limit: 65535
- Firmware Flags:** Two status indicators: 'Programmable' with a green square and 'DCO' with a green square.

## 4.2 Scattergate info

The Scattergate info panel shows you all important information about the running software.

The screenshot shows a web interface for Scattergate info. At the top left is a 'PCL' button. A central table displays the following data:

ID	Device	Name	Length	Date
1	USB430	ScatterGate 1.26	1600-B88E	Jan 3 2005 01:53:59

Below this table, there are two sections. On the left, a 'Length' field shows '600-C10D' and a 'Date' field shows 'Feb 15 2005 12:06:52', with a 'Clear EEPROM' button below. On the right, an 'additional information' section lists:

- Firmware Flags: 07
- DCO: D0
- Transmit Power: 255
- BCS: B7
- rx Receive Limit: 65535
- Panic Flags: FF

At the bottom, there are two status indicators: 'Programmable' with a green square and 'DCO' with a green square.

### 4.3 Loaded to EEPROM

You can upload a new image on the Gate with a click on the [Load to EEPROM](#) button. You can also clear the complete EEPROM with a click on the [Clear EEPROM](#) button. If an image is detected on the eGate, the software shows you any information about that.

<b>ID and reset</b>		<b>Scattergate info</b>	
Node ID:	<input type="text" value="1"/> <input type="button" value="set"/> <input type="button" value="PCL"/>	<b>ID</b>	<input type="text" value="1"/>
		<b>Device</b>	<input type="text" value="USB430"/>
		<b>Name</b>	<input type="text" value="ScatterGate 1.26"/>
		<b>Length</b>	<input type="text" value="1600-B88E"/>
<b>Loaded to EEPROM</b>			
<b>Device</b>	<b>Name</b>	<b>Length</b>	<b>Date</b>
<input type="text" value="ESB430"/>	<input type="text" value="ScatterNode 1.26"/>	<input type="text" value="1600-C10D"/>	<input type="text" value="Feb 15 2005 12:06:52"/>
<input type="button" value="Load to EEPROM"/>		<input type="button" value="Clear EEPROM"/>	
<b>additional information</b>			
Firmware Flags:	<input type="text" value="07"/>		
Transmit Power:	<input type="text" value="255"/>		
rx Receive Limit:	<input type="text" value="65535"/>		
<b>Firmware Flags</b>			
<input type="button" value="Programmable"/>	<input checked="" type="checkbox"/>	<input type="button" value="DCO"/>	<input checked="" type="checkbox"/>

#### 4.4 Additional Information

That panel contains additional information about the eGate like firmware flags, transmit power, RX receive limit, DCO, BCS, and panic flags.



The screenshot shows the 'Scattergate info' panel. It includes a 'PCL' button, a table with columns 'ID', 'Device', 'Name', 'Length', and 'Date'. The table contains one entry: ID: 1, Device: USB430, Name: ScatterGate 1.26, Length: 1600-B88E, Date: Jan 3 2005 01:53:59. Below this is a section for 'additional information' with fields for Firmware Flags (07), DCO (D0), Transmit Power (255), BCS (B7), rx Receive Limit (65535), and Panic Flags (FF). There is also a 'Clear EEPROM' button and a status bar with 'Programmable' and 'DCO' buttons, each with a green LED indicator.

ID	Device	Name	Length	Date
1	USB430	ScatterGate 1.26	1600-B88E	Jan 3 2005 01:53:59

**additional information**

Firmware Flags:	07	DCO:	D0
Transmit Power:	255	BCS:	B7
rx Receive Limit:	65535	Panic Flags:	FF

Clear EEPROM

Programmable  DCO 

#### 4.5 Firmware Flags

The Programmable and DCO buttons enable you to set that flags. The LEDs show you which flags are set.

The screenshot shows the 'ID and reset' panel with 'Node ID' set to 1 and 'set' and 'PCL' buttons. The 'Scattergate info' panel is also visible. Below is the 'Loaded to EEPROM' section with a table for 'Device', 'Name', 'Length', and 'Date'. The table contains one entry: Device: ESB430, Name: ScatterNode 1.26, Length: 1600-C10D, Date: Feb 15 2005 12:06:52. There are 'Load to EEPROM' and 'Clear EEPROM' buttons. To the right is the 'additional information' section with fields for Firmware Flags (07), Transmit Power (255), and rx Receive Limit (65535). At the bottom is the 'Firmware Flags' section with 'Programmable' and 'DCO' buttons, each with a green LED indicator.

Node ID: 1 set PCL

ID	Device	Name	Length
1	USB430	ScatterGate 1.26	1600-B88E

**Loaded to EEPROM**



Device	Name	Length	Date
ESB430	ScatterNode 1.26	1600-C10D	Feb 15 2005 12:06:52

Load to EEPROM Clear EEPROM

**additional information**

Firmware Flags:	07
Transmit Power:	255
rx Receive Limit:	65535

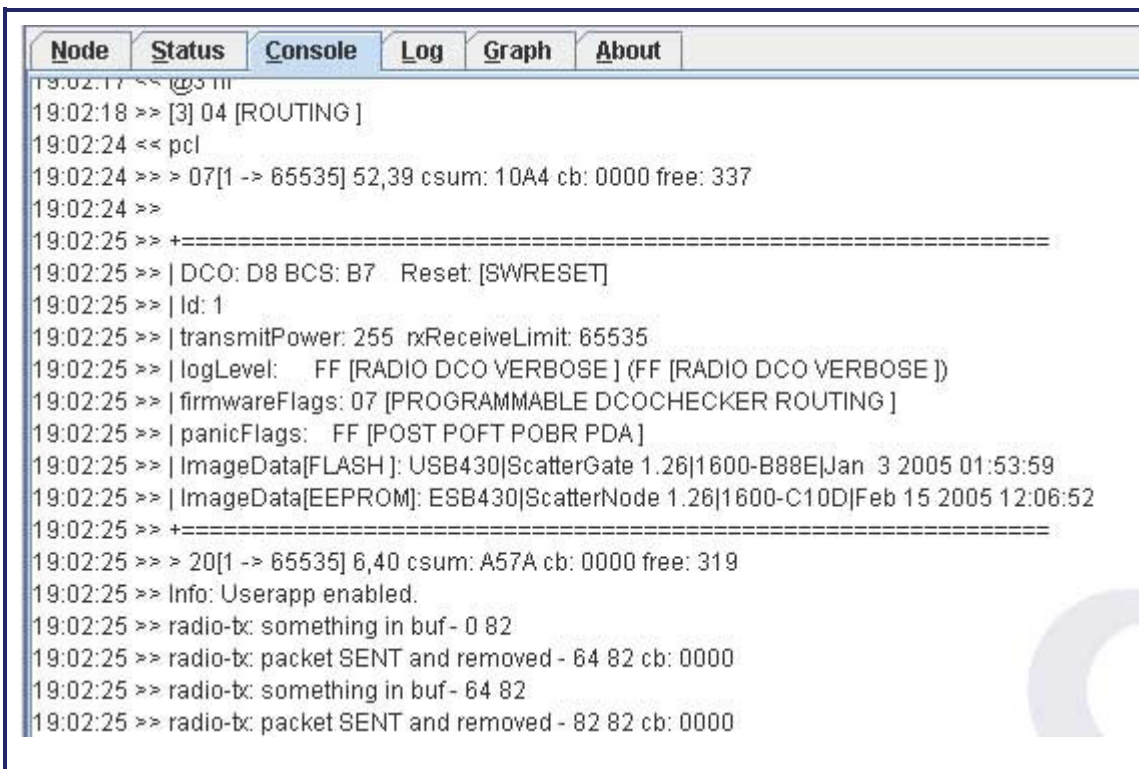
**Firmware Flags**

Programmable  DCO 



## 5 Console

The console is comparable to a terminal application which communicates with the eGate. It handles not only the data traffic but also the communication because it enables you to send commands over the console to the eGate or to nodes which are in the range of sending. The symbols << and >> identify the direction of communication. The << symbol means outgoing messages and the >> symbol means incoming messages.



```
Node Status Console Log Graph About
19:02:17 << @5 III
19:02:18 >> [3] 04 [ROUTING ]
19:02:24 << pcl
19:02:24 >> > 07[1 -> 65535] 52,39 csum: 10A4 cb: 0000 free: 337
19:02:24 >>
19:02:25 >> +=====
19:02:25 >> | DCO: D8 BCS: B7  Reset: [SWRESET]
19:02:25 >> | Id: 1
19:02:25 >> | transmitPower: 255  rxReceiveLimit: 65535
19:02:25 >> | logLevel:  FF [RADIO DCO VERBOSE ] (FF [RADIO DCO VERBOSE ])
19:02:25 >> | firmwareFlags: 07 [PROGRAMMABLE DCOCHECKER ROUTING ]
19:02:25 >> | panicFlags:  FF [POST POFT POBR PDA ]
19:02:25 >> | ImageData[FLASH]: USB430|ScatterGate 1.26|1600-B88E|Jan  3 2005 01:53:59
19:02:25 >> | ImageData[EEPROM]: ESB430|ScatterNode 1.26|1600-C10D|Feb 15 2005 12:06:52
19:02:25 >> +=====
19:02:25 >> > 20[1 -> 65535] 6,40 csum: A57A cb: 0000 free: 319
19:02:25 >> Info: Userapp enabled.
19:02:25 >> radio-tx: something in buf- 0 82
19:02:25 >> radio-tx: packet SENT and removed - 64 82 cb: 0000
19:02:25 >> radio-tx: something in buf- 64 82
19:02:25 >> radio-tx: packet SENT and removed - 82 82 cb: 0000
```

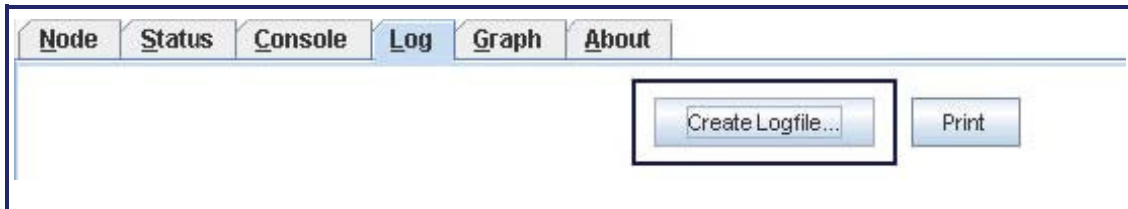
The sensor data are an extensive part of the total data traffic. So you are able to filter these data with the [show sensor data](#) button.



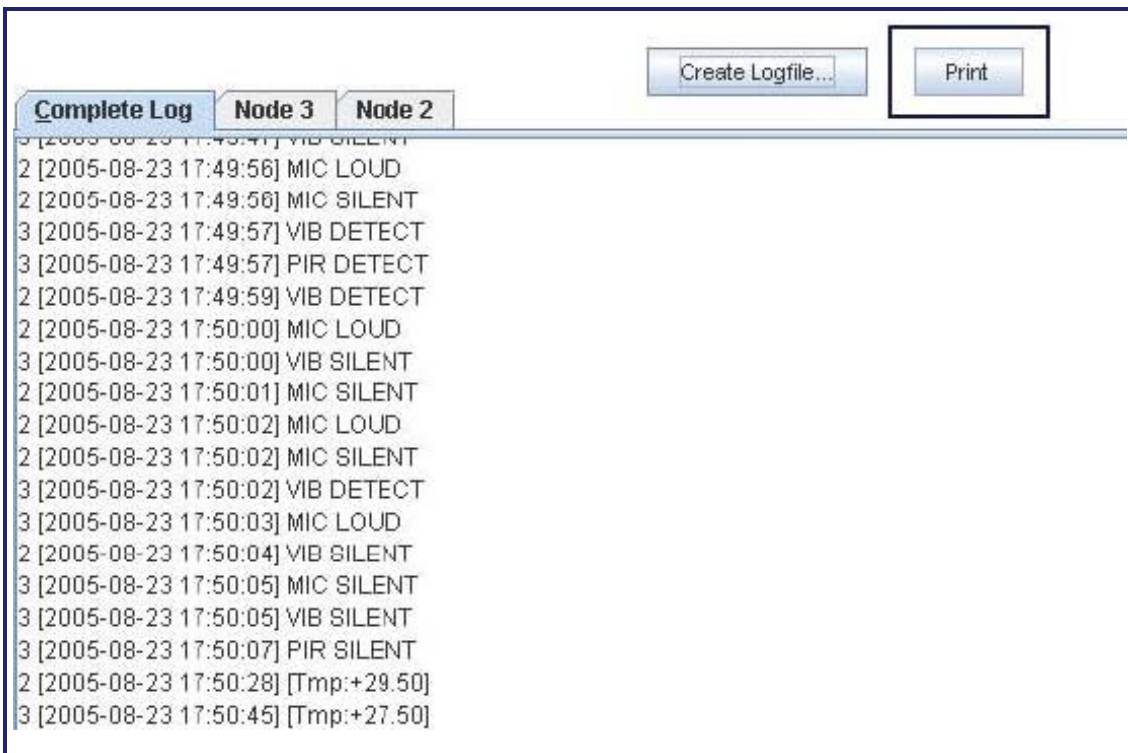
## 6 Log

The log tab includes all sensor events, which are obtained since the ScatterViewer is started. There is also the possibility to file all sensor events or to print them:

### 6.1 LogFile

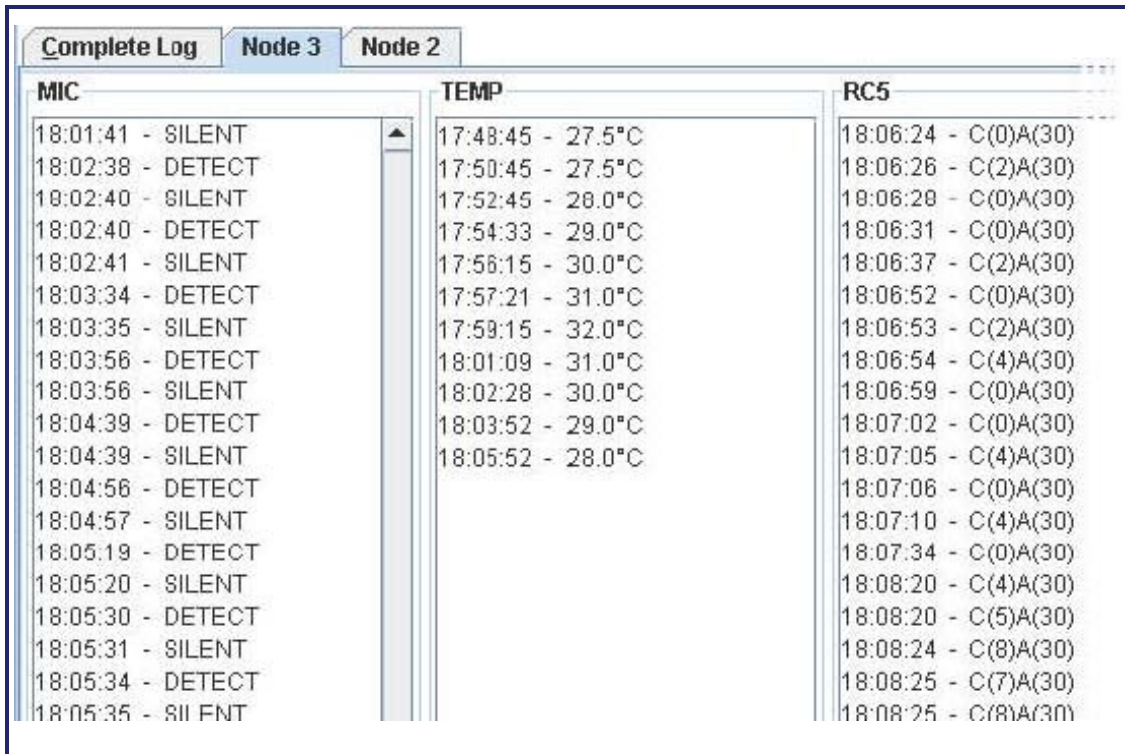


### 6.2 LogPrint



### 6.3 Tabs of the single sensor node even

The software constructs a tab for every node which is sending sensor events like microphone, temperature, RC5, movement, and vibration. If there is no node which is sending sensor events, only the empty complete log tab is visible.



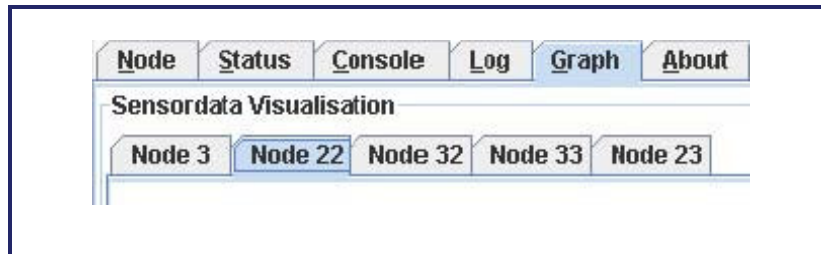
MIC	TEMP	RC5
18:01:41 - SILENT	17:48:45 - 27.5°C	18:06:24 - C(0)A(30)
18:02:38 - DETECT	17:50:45 - 27.5°C	18:06:26 - C(2)A(30)
18:02:40 - SILENT	17:52:45 - 28.0°C	18:06:28 - C(0)A(30)
18:02:40 - DETECT	17:54:33 - 29.0°C	18:06:31 - C(0)A(30)
18:02:41 - SILENT	17:56:15 - 30.0°C	18:06:37 - C(2)A(30)
18:03:34 - DETECT	17:57:21 - 31.0°C	18:06:52 - C(0)A(30)
18:03:35 - SILENT	17:59:15 - 32.0°C	18:06:53 - C(2)A(30)
18:03:56 - DETECT	18:01:09 - 31.0°C	18:06:54 - C(4)A(30)
18:03:56 - SILENT	18:02:28 - 30.0°C	18:06:59 - C(0)A(30)
18:04:39 - DETECT	18:03:52 - 29.0°C	18:07:02 - C(0)A(30)
18:04:39 - SILENT	18:05:52 - 28.0°C	18:07:05 - C(4)A(30)
18:04:56 - DETECT		18:07:06 - C(0)A(30)
18:04:57 - SILENT		18:07:10 - C(4)A(30)
18:05:19 - DETECT		18:07:34 - C(0)A(30)
18:05:20 - SILENT		18:08:20 - C(4)A(30)
18:05:30 - DETECT		18:08:20 - C(5)A(30)
18:05:31 - SILENT		18:08:24 - C(8)A(30)
18:05:34 - DETECT		18:08:25 - C(7)A(30)
18:05:35 - SILENT		18:08:25 - C(8)A(30)

## 7 Graph

The graph tab is a visual realization of the received sensor events. It illustrates temperature, movement, microphone, and vibration. Thought has not been given to the RC5 sensor because it should seem obvious to you that this is unsuitable for this kind of visualization.

### 7.1 *Tabs of the single sensor nodes*

The software constructs a tab for every node which is sending sensor events like microphone, temperature, movement, and vibration. If there is no node which is sending sensor events, also no tab is visible.



## 7.2 The Graph

The most of the sensors like vibration, microphone, and movement are event-based. This means that we differentiate between detect and silent (sensor detecting) in contrast to the temperature sensor, which sends repeatedly the temperature (sensor repeating).

