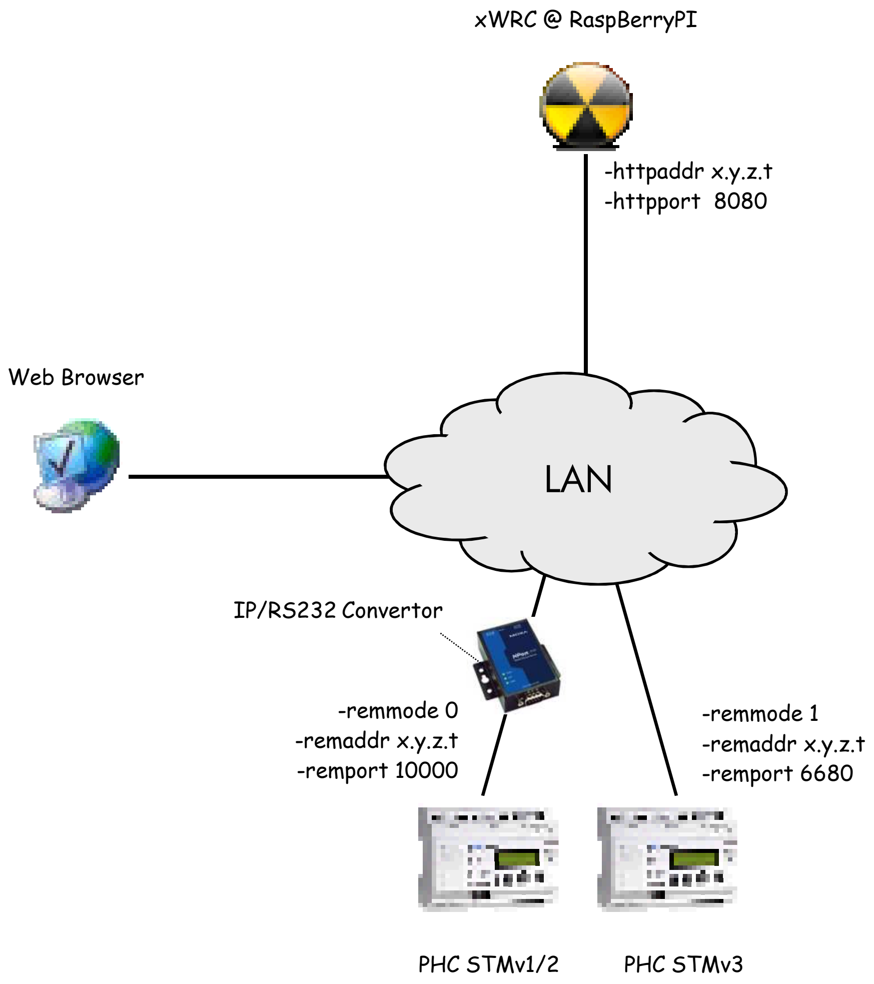
xWRC Overview

Jo Simons (icontrol@telenet.be)

xWRC Overview (1/1)



xWRC Installation

Jo Simons (icontrol@telenet.be)

xWRC Installation (1/2)

- Download latest packages:
  - From http://phc-forum.de/index.php/forum/visualisierung/37-raspi4phc:
  - xwrc.<version>.data.zip
  - xwrc.<version>.core.zip

- Install data package:
  - Contains files and directories that are not directly related to a specific version of xWRC
    - main webserver pages
    - iControl pages and samples
  - Named xwrc.<version>.data.zip, i.e. xwrc.3.3.0.9.data.zip
  - Unzip zip-file while maintaining directory structure
    - cd xwrc
    - unzip xwrc.3.3.0.9.data.zip

- Install core package:
  - Contains files and directories that are directly related to a specific version of xWRC
    - executables, ini-file(s), help file
  - Named xwrc.<version>.core.zip, i.e. xwrc.3.3.0.9.core.zip
  - Unzip zip-file while maintaining directory structure and give execute rights to Raspberry executable
    - cd xwrc
    - unzip xwrc.3.3.0.9.core.zip
    - cd bin
    - chmod +x xwrc.raspi
    - copy your xwrc.license.bin file to xwrc/bin

xWRC Installation (2/2)

- Getting help:
  - cd xwrc/bin
  - ./xwrc.raspi -?
  - or read xwrc.help.pdf

- Connecting your RS232-to-IP convertor:
  - Locate RS232 interface on your STM, then connect it to convertor as follows:
    - STM.0V -> convertor.Gnd
    - STM.TxD -> convertor.RxD
    - STM.RxD -> convertor.TxD

- Configure your RS232-to-IP convertor:
  - Setup the RS232 side to 19200 baud, 8 databits, 1 stopbit, no parity, no flowcontrol, disable any FIFO
  - Setup the IP side:
    - TCP server mode
    - either let it send packets to xWRC when character 0xC1 is received on the RS232 side (more performant)
    - or let it send each byte to xWRC when above setting is not available (less performant)

xWRC Usage

Jo Simons (icontrol@telenet.be)

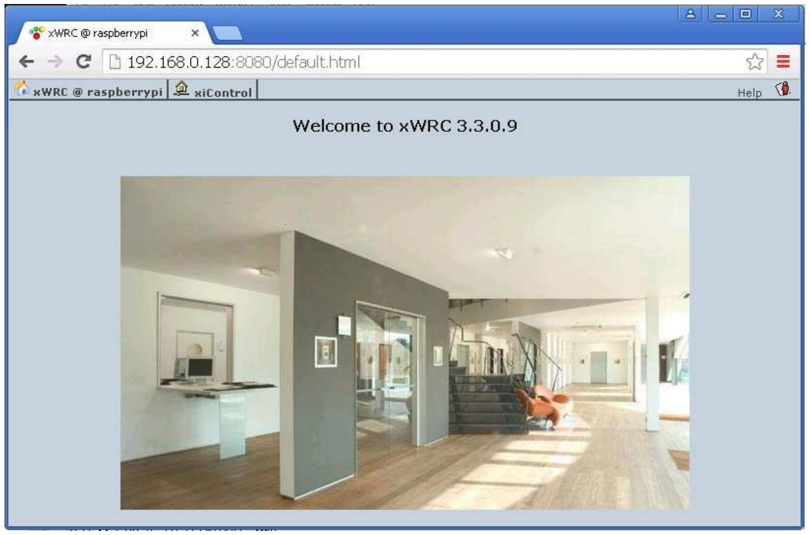
xWRC Usage (1/4)

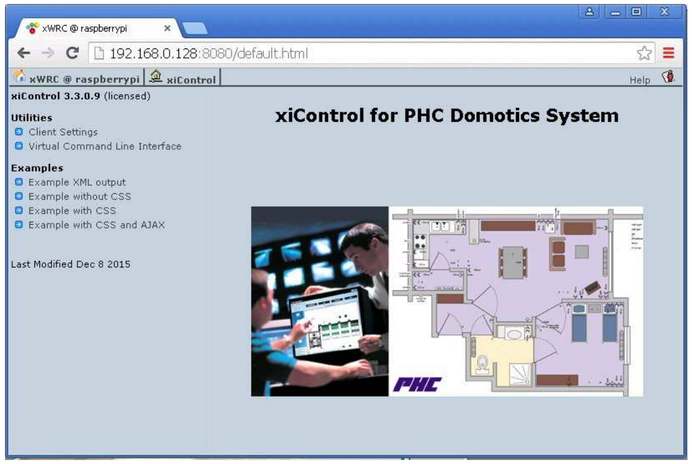
- Start xWRC:
  - cd xwrc/bin
  - ./xwrc.raspi –httpport 8080 –remaddr <ip-address-of-convertor>

- Use browser to access xWRC:
  - http://<ip-address-of-raspberry>:8080/
  - Click ‘Help’ in the topside menu to open the help file in pdf format

- Using the URL bar to send commands to your PHC:
  - Refer to section 2.4 ‘Structured command URL’ in the help file
  - In the URL bar of your browser enter the structured URL containing commands
    - http://<ip-address-of-raspberry>:8080/icontrol.dll?ccmd=omd.0.out0.toggle&verbose
    - the ‘&verbose’ lets xWRC send back a more textual result format of the commands
    - use ‘&terse’ to return the outcome of the commands only
    - try it out !!!

- Using the Virtual Command Line Interface:
  - Click ‘xiControl’ in the topside menu
  - Then click on ‘Virtual Command Line Interface’ in the leftside menu
  - On the right side, enter commands in the ‘Ccmd’ box and press ‘Execute’
    - i.e. omd.0.out0.toggle
    - the verbose, terse and xml results will be shown in the boxes on the page





xWRC Usage (2/4)

- Structured naming of the images in xwrc/icontrol/img directory:
  - Command images: c.<type>.<cmd>.gif
    - Contain only 1 image, about 16x16 pixels
    - Typically used to represent an event that triggers a programmed function in a STM
      - i.e. c.in.ingt0.gif, c.im.start.gif, c.jrm.down.gif
    - Others have more general use
      - i.e. c.onoff.gif, c.reload.gif, c.stop.gif, …
  - Status images, basic: s.<type>.<state>.gif
    - Contain only 1 image, about 16x16 pixels
    - Images for visualising on/off type of outputs
      - i.e. s.light.0.gif, s.light.1.gif, s.led.0.gif, s.led.1.gif, …
    - Images for visualising JRM type of outputs (i.e. using omd to simulate jrm, or using feedback programming)
      - i.e. s.jrm.0.gif, s.jrm.1.gif, s.jrm.2.gif (0=stopped, 1=moving down, 2=moving up respectively)
    - Images for visualising dimmer channels have 9 images
      - States: 0=off,1=12%,2=25%,3=37%,4=50%,5=62%,6=75%,7=87%,8=100% on
      - Small vertical layout, 16x16 pixels: s.dimv.<state>.gif
      - Large horizontal layout, 80x16 pixels: s.dimh.<state>.gif
      - Convert dimmer level to state as follows: state = (level + 31) / 32
      - i.e. s.dimv.8.gif, s.dimh.3.gif
  - Status images, advanced: s.<type>.gif
    - Contain multiple images used by ohcAjaxSendCmd() function, see later on page 4/5
    - About 16x(N x 16) pixels, where N is the number of states that are present
    - i.e. s.4state.gif, s.jrm.gif, s.light.gif, s.dimv.gif

xWRC Usage (3/4)

- Sending commands from a HTML page
  - Using a plain hyperlink
    - <a href="icontrol.dll?ccmd=omd.0.out2.toggle&file=<?request file?>">Toggle</a>
  - Using a plain hyperlink and displaying the result of the command inline
    - <a href="icontrol.dll?ccmd=omd.0.out2.toggle&file=<?request file?>">
    - <img border="0" src="/<?plugin alias?>/img/s.light.<?phc omd.0.out2?>.gif" title="Toggle">
    - </a>
  - Using a plain hyperlink and displaying the result of the command at a CSS specified location
    - Refer to 'Examples 1->Sample page with CSS but no AJAX'
    - <a href="icontrol.dll?ccmd=omd.0.out2.toggle&file=<?request file?>">
    - <img border="0"style="position: absolute; top:62px; left:55px"
    - src="/<?plugin alias?>/img/s.light.<?phc omd.0.out2?>.gif" title="Toggle">
    - </a>
  - Using an AJAX hyperlink and displaying the result of the command at a CSS specified location
    - Refer to 'Examples 1->Sample page with CSS and AJAX'
    - <a href="javascript:ohcAjaxSendCmd('!omd.00.out7.toggle', 0, 'omd.00.out7', 16)">
    - <img border="0"
    - id="omd.00.out7"
    - src="/<?plugin alias?>/img/pipe.gif"
    - style="position: absolute; top:40px; left:306px;
    - background-image: url(/<?plugin alias?>/img/s.light.gif); width: 16px; height: 16px;"
    - title="Toggle">
    - </a>

xWRC Usage (4/4)

- Explaining ohcAjaxSendCmd(<strCcmd>,<nDelay>,<strStatus>,<dyImage>)
  - This function will call xWRC over a background connection to:
    - Use terse response mode
    - Execute strCcmd, this can be one or more events (i.e. imd.0.in0.ingt0) or module actions (i.e. omd.0.out0.toggle)
      Refer to section 3.4.1 ‘Compound-cmd syntax’ in the help file
      Please add an exclamation mark ‘!’ in front of each cmd to suppress the results from the xWRC response
    - Insert a pause of nDelay milliseconds to let your PHC system handle strCcmd, 0 means no delay
      This is typically useful when setting a dimmer level, triggering a STM function that takes some time to complete…
    - Fetch the new status of affected outputs by executing strStatus, this is again a compound-cmd
      Refer to section 3.4.1 ‘Compound-cmd syntax’ in the help file
      The terse result of each cmd will be applied to all visual objects with a same ‘id’ as the cmd
      So ohcAjaxSendCmd('!', 0, 'omd.02.out4', 16) will apply new status to all visual objects with id=‘omd.02.out4’
    - Applying new status to a visual object means shifting the view area on the objects image, the shift size is dyImage
      The image is organised as multiple images stacked in vertical direction with each image being dyImage pixels high and representing a specific state, state 0 starts at yOffset=0, state 1 at yOffset=16, …
      For dimmer modules the image state is calculated as follows: state=level+31/32
  - Applying new status for a single output
    - <a href="javascript:ohcAjaxSendCmd('!imd.00.in12.outlt1', 0, 'omd.02.out3', 16)”>Toggle</a>
    - <a href="javascript:ohcAjaxSendCmd('!omd.02.out4.toggle', 0, 'omd.02.out4', 16)”>Toggle</a>
  - Applying new status for multiple outputs (like status of jrm simulation with omd)
    - <a href="javascript:ohcAjaxSendCmd('', 50, 'omd.01.out0:omd.01.out1', 16)">Refresh</a>
    - <a href="javascript:ohcAjaxSendCmd('!imd.00.in5.ingt0', 50, 'omd.01.out0:omd.01.out1', 16)">Stop</a>
  - Applying new status for a dimmer output
    - <a href="javascript:ohcAjaxSendCmd('' , 500, 'dim.00.out0;dim.00.out1', 16)">Refresh</a>
    - <a href="javascript:ohcAjaxSendCmd('!dim.00.out0.set.127.1', 500, 'dim.00.out0;dim.00.out1', 16)">50%</a>