



**The Abdus Salam
International Centre for Theoretical Physics**



2273-4

**Second Workshop on Open Source and the Internet for Building Global
Scientific Communities with Emphasis on Environmental Monitoring and
Distributed Instrumentation**

28 November - 16 December, 2011

Software for the Geode

U. Raich
*CERN, Geneva
Switzerland*

Software for the Geode



Ulrich (Uli) Raich
CERN BE department
Beam Instrumentation Group

Shopping list



- We need something that loads a system kernel into the RAM memory used by the processor (boot loader, Wrap uses BIOS)
- We need a Linux kernel
- We need a Linux root file system with startup files and a command interpreter

PCEngines information



- **Voyage Linux WEB site**
- Buy a CF card
- Connect it to your PC (you need a CF card reader)
- Partition the CF card into 2 partitions
- Download voyage Linux to your PC
- Write it to the newly created CF partitions
- Put the CF card into the Wrap
- Boot it

Let's try!



- I do not have a CF card reader on the Laptop but I do have one at home
- System has already been prepared
- How do we see it the system does something?
- Connect the Wrap's console port to a terminal emulator!

minicom



- Type *minicom*
- Answer: *command not found*
- What's wrong?

Install minicom



- apt-get install minicom installs minicom into the system
- How does minicom work?
- *man minicom*
- As super user: *minicom -s*
- Defined the base configuration
- Define baud rates (Wrap uses 38400 8N1)
- Save it in the defaults file
- Install minicom in the toolbar

Start it!



- Connect the RS-232 connection to the Wrap board
- Connect power and watch
- You don't see anything?
- Check the cable (null modem?), check the power
- You only have strange characters?
- Check the baud rate.
- It works? We are in business!

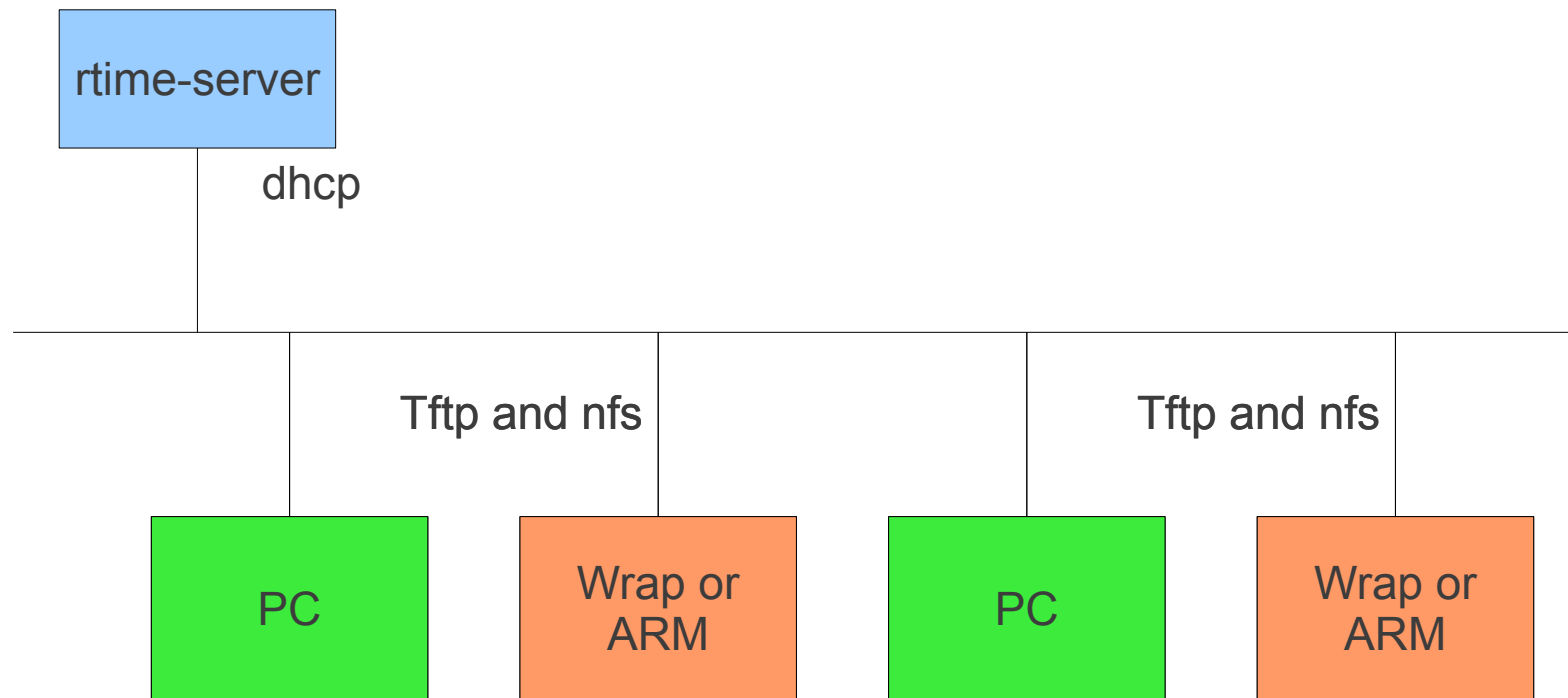
At ICTP



- We do not have CF card readers!
- We do not have CF cards for each Wrap board!
- How do we boot the kernel?

Let's take out the CF card and see what it does without it.

Network layout



Booting over the network



- Wrap BIOS consists of 2 parts:
 - Settings of system resource parameters
 - Etherboot

[Etherboot Website](#)

Boot sequence



- Get an Ethernet connection going between the PC and the Wrap board
- Wrap needs an IP address which it gets via dhcp from the PC
 - -> we must set up a DHCP server on the PC
- The boot loader is downloaded into the Wrap via tftp
 - -> need the boot loader (pxelinux.0) from the **syslinux project**
 - -> we must setup a tftp server on the PC

Setup DHCP



In the lab a single DHCP server is set up for you serving all embedded systems (Wrap and ARM)

- We must provide the configuration file `/etc/dhcp/dhcpd.conf` and we must run the dhcp daemon

DCHP config file



```
#
# DHCP Server Configuration file.
# see /usr/share/doc/dhcp*/dhcpd.conf.sample
# see 'man 5 dhcpd.conf'
#
subnet 10.41.25.0 netmask 255.255.255.0 {
    range 10.41.25.200 10.41.25.220;
    default-lease-time 86400;
    max-lease-time 86400;
    option domain-name-servers 10.41.25.254;
    authoritative;
    option broadcast-address 10.41.25.255;
    option routers 10.41.25.254;
}
host wrap {
    hardware ethernet 00:0D:B9:04:4B:2C;
    option host-name "wrap";
    option root-path "/opt/ICTP/micros/pcengines/voyage-0.8.0";
    filename "pxelinux.0";
    fixed-address 10.41.25.201;
    next-server 10.41.25.12;
}
```

Setup tftpd



- The tftp daemon is started by xinet
- The xinetd must be running!
- We need to have a look at `/etc/xinetd.conf` and the startup file for tftp in `/etc/xinetd.d`

/var/lib/tftpboot



- As seen in the `/etc/xinetd.d/tftp` the tftp server uses `/var/lib/tftpboot` as directory for the files it distributes
- Here is `/var/lib/tftpboot`
- There are 2 files of interest:
 - `pxelinux.0`, which is the boot loader. This is a binary file
 - `pxelinuc.cfg` which contains configuration files for each Wrap system served

pxeconfig file



- For each Wrap board there is a file with the Wrap's MAC address as file name or a default file for all
- **pxe config file**
- Let's check again the files in **/var/lib/tftpboot** where we should find:
 - vmlinuz
 - display.msg

What is still missing?



- We have the boot loader
- We have the kernel
- We do not have...?

The root file system



- The root file system must be mounted through nfs
 - The network driver and nfs must be included into the kernel image (not as a kernel module!)
 - The root file system must be exported by the nfs server
exports are defined in `/etc/exports`
 - An nfs server must be running on the PC hosting the root file system for the Wrap

Summary



- Wrap gets its IP address through DHCP
- It asks its tftp server for pxelinux.0 (bootloader)
- The bootloader is transferred to the Wrap and started
- The boot loader is started and gets its configuration for pxelinux.cfg
- It loads the kernel and starts it
- It displays the welcome message display.msg
- The kernel mounts its root file system on nfs and starts the initialisation program (init)
- Control is passed to the command interpreter