CDC-LAB – MEMENTO

This little memo describes the hardware and software of the CDC-Lab and how to deploy, compile and run C and Erlang programs.

1. Hardware, Network and Login

Components

PAI's Concurrent and Distributed Computing Lab (CDC-Lab) is currently composed of 15 machines and possibly your laptop, organized within the unifr intranet, cf. Fig. 1 and Table 1.

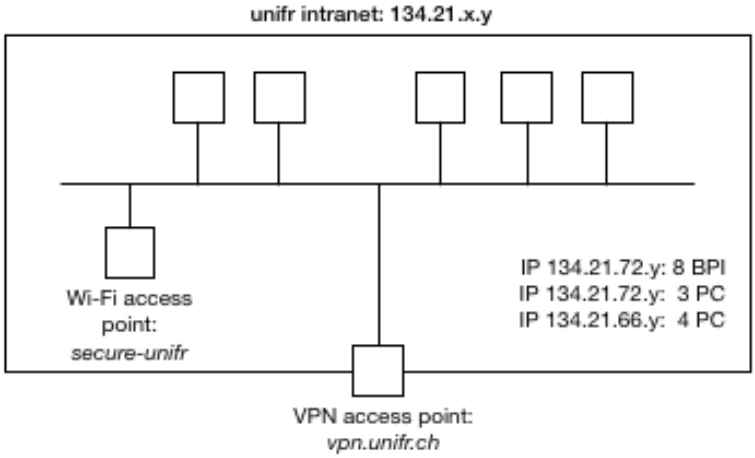

Figure 1. IP network

CDC-Lab A421 – IP Network 134.21.72.y: Banana Pis (Total: 8)

CDC	DIT	DNS	Type	Mark	Hardware	OS	Erlang
1-7	–	diufpc80 *to* diufpc87	SBC	BPI	A20 ARM Cortex - A7 Dual-Core, 32-bit, 1 GHz, RAM 1 GB, SD 32 GB, Ethernet 1 Gb/s, late 2014 5V @ 700mA (=> max 3.5 W),	Rasbian (based on Debian 7 "wheezy")	OTP 17.5.3

CDC-Lab A421 – IP Network 134.21.72.y: Mac minis (total 3)

8 9 10	34750 34755 34756	diufmac31 diufmac53 diufmac54	PC	Mac Mini	Intel 2-Core i5, 64-bit, 1.4 GHz, RAM 4GB, HD 500GB, Ethernet 1 Gb/s, mid 2014	Linux Mint 17.1 (Ubuntu based)	OTP 18.0

Public Room A201 – IP Network 134.21.66.y: PCs (total 4)

11 12 13 14	31390 31391 31392 31393	diufsppc717 diufsppc718 diufsppc719 diufsppc720	PC	HP	Intel i5, 4-core, 64-bit, 2.67 GHz, RAM 4 GB, HD 320 GB, Ethernet 1 Gb/s, mid 2010	Ubuntu 14.0.3	OTP 18.0

Anywhere in the world, but connected to IP Network 134.21.x.y: your laptop

15	–	–	*Student's Laptop*	*any*	*Any not too old*	*Any Unix, supported by Erlang*	*Any not too old*

PC–: Personal Computer without display, keyboard, mouse
PC: Personal Computer with display, keyboard and mouse
SBC: Single Board Computer
BPI: Banana Pi, version MI; for more information cf. https://en.wikipedia.org/wiki/Banana_Pi

Table 1. Hardware, network and software of the CDC-Lab

Entry point

Any of the machines in the unifr intranet may act as an entry point, but most probably it will be your laptop or one of the machines available in the public room A201. Your laptop may be located anywhere in the world but has to be connected to the unifr intranet (134.21.x.y) via ***VPN*** if you are outside the campus, and via ***Wi-Fi 'secure-unifr'*** if you are on campus.

IP Network
The IP network ***134.21.x.y*** is ***unifr's intranet*** which is managed via DHCP (i.e., each machine has a fixed DNS name, but a dynamic IP number allocated at the start of a connection). All machines except for your laptop are physically interconnected with Ethernet cables.

Bandwidth
Typically the bandwidth varies from 10 to 1000 Mb/s for wired Ethernet and wireless Wi-Fi. Note that Ethernet interconnections are much more stable and reliable than Wi-Fi. To measure the actual throughput, you can use the freely available ***iPerf*** software, cf. https://en.wikipedia.org/wiki/Iperf.

Availability
Machines CDC 01–10 are located in the semi-public room A421 and therefore their availabilities are high. Machines CDC 11-14 are located in a public room for use by other students and might accidentally be shut down to save energy in spite of the post-it note stating 'Don't shut down these machines'.

Login
- ***On "134.21.x.y" machines:*** *username* and *password* are the same as for the unifr intranet.

2. System Software and Languages
Each machine runs a Unix variant as well as standard C and Erlang environments, cf. Table 1.

Erlang Installation on your laptop
It is best to install the latest Erlang OTP version from the original site:

https://www.erlang-solutions.com/resources/download.html

by selecting your OS.

Ubuntu: select *standard* and then the latest pre-built package. You may also install Erlang via "Install using repository". Erlang is also available via `apt-get install erlang`.

OS X: select the latest pre-built package corresponding to your OS X version. Erlang is also available through Homebrew (`brew install erlang`) or MacPorts (`port install erlang`) if you have either of these package managers installed.

Windows: Erlang also runs under Windows, but not our scripts. We recommend partitioning your disk or at least virtualizing a Linux, e.g. Ubuntu, via the freely available virtualization platform VirtualBox.

32-bit or 64-bit: be sure to install the 32-bit or 64-bit version corresponding to your machine's architecture (if you don't know, type `uname -m` in the command line or consult the manual of your machine).

In the following, we will give a manual example for deploying a C resp. an Erlang program on one of the Banana Pi machines. If you do this often, we recommend writing a script or two.

For more complex situations where an Erlang program is deployed on a set of machines, the scripts are more complicated and will be provided to you.

3. Running a C Program on a CDC-Lab Machine

Assumptions about the environment

The following environment is assumed on all machines:

- A family of C programs, say **hw1.c** and **hw2.c** have a common **Makefile** and are regrouped in a folder, e.g. **~/mpe/c/hw**. The acronym **mpe** stands for "my programming environment". If the directories **~/mpe/c/** do not exist, they have to be created first, e.g. with the command `mkdir`.
- All programs are running in a Unix shell such as bash, with or without an explicitly opened window.
- You are connected to a machine in the subnet "134.21.x.y", e.g. your laptop is connected to this subnet via the Wi-Fi **secure-unifr** channel available on the unifr campus or via **VPN**.

Example of a deployment on a Banana Pi

In this example, we will use the program **~/mpe/c/hw/hw1.c**, existing on the local machine:

```
#include <stdio.h>
int main() { printf("Hello World\n"); return 0; }
```

and we will copy and execute it onto a Banana Pi.

Very first connection. Create the folders **~/mpe/c/** in your directory on a remote machine, in this example on the BPI diufpc84:

```
(L1) local ~$ ssh  <user_name_on_diufpc84>@diufpc84

(R1) diufpc84 ~$ mkdir  ~/mpe  ~/mpe/c

(R2) diufpc84 ~$ exit
(L2)
```

Follow-up connections. Compress your C program and copy it to the remote machine at **~/mpe/c/**:

```
(L1) local ~$ cd ~/mpe/c/
(L2) local ~/mpe/c$ zip -rq hw.zip  hw          # Compress folder hw
(L3) local ~/mpe/c$ scp hw.zip  \
                    <user_name_on_diufpc84>@diufpc84:~/mpe/c
(L4) local ~/mpe/c$ ssh <user_name_on_diufpc84>@diufpc84
```

unzip, compile and execute hw1:

```
(R1) diufpc84 ~$ cd ~/mpe/c
(R2) diufpc84 ~/mpe/c$ unzip hw.zip > /dev/null
(R3) diufpc84 ~/mpe/c$ cd hw
(R4) diufpc84 ~/mpe/c/hw$ gcc hw1.c -o hw1      # compile hw1.c
(R5) diufpc84 ~/mpe/c/hw$ ./hw1                 # run program hw1
      Hello World                               # result produced by hw1
```

And finally remove all created files and exit all remote shells:

```
(R6) diufpc84 ~/mpe/c/hw$ rm -rf ../hw*; exit
(L5)
```

There exist multiple variants to obtain the same results, some using more advanced line commands, opening multiple Unix shell windows, e.g., one per machine, or by using scripts. It is up to you to find out your best way of working.

4. Running an Erlang Program on CDC-Lab's Machines

Assumptions about the environment

- For Erlang programs running on a single machine, the assumptions are quite similar to the ones for C programs. The main difference is that the Erlang programs are located at **~/mpe/erl/**.
- Erlang programs which are intended to be deployed on a couple of machines are best deployed and executed with scripts. The main script is provided at **~/mpe/erl/teda/scripts**, with the assumption that the source codes are located at **~/mpe/erl/teda/apps**. For more information, consult the *Erlang teDA Memento*.

Example of a deployment on one the of Banana PI

Quite similar to C programs, but Erlang programs are running in a special dedicated **Erlang shell**, which runs on top of a Unix shell. For more information, consult the *Erlang teDA Memento*.

Example of a deployment on all CDC-Lab machines

Repeat n times the deployment on 'one CDC-Lab machine'. Moreover you will have to indicate in some way the set of Erlang shells that have to work together. For more information, consult the *Erlang teDA Memento*.

Florian Evéquoz, Laura Rettig, Béat Hirsbrunner, Nicolas Juillerat, Christian Göttel, University of Fribourg,

History:
Version 2.0, 18 September 2016:

- The original text has been written by BH and proofread by NJ and CG
- The miscellaneous software have been installed by:
  - NJ: CDC 01 (Mac mini, router), CDC 10-17 (BPI) and CDC 20-23 (HP)
  - Simon Studer and CG: CDC 31-33 (Mac mini), also used for the robotics lab
  - Birgit Ducarroz: CDC 40-43 (HP)
- The figures, text, and instructions have been adapted to the all-intranet setup by Laura Rettig