



Institute for the Wireless Internet of Things

at Northeastern University

Colosseum Containers



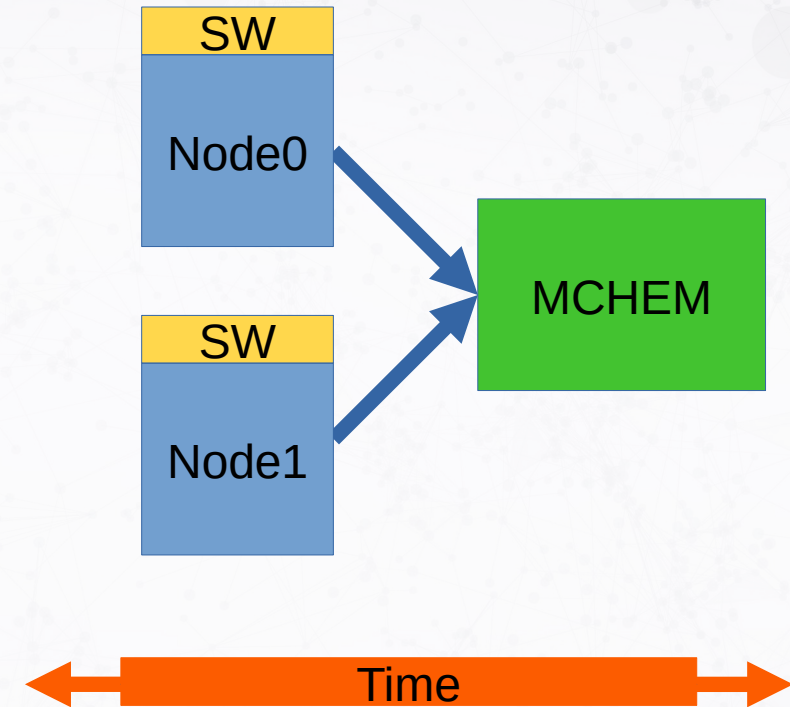
Platforms for Advanced
Wireless Research



N COLOSSEUM
at Northeastern University

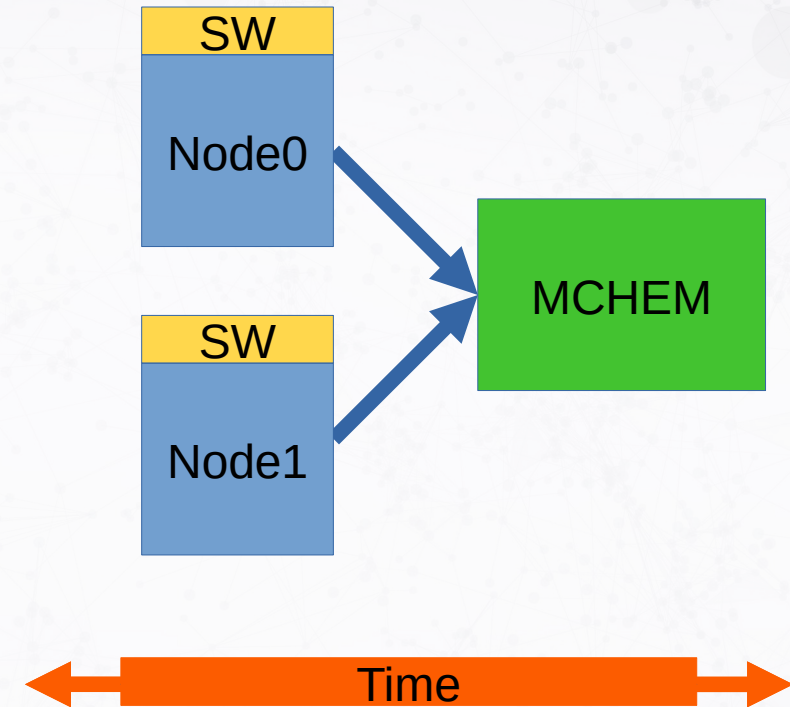
Colosseum Experiment

- Reservation
 - Nodes
 - Time
 - Software



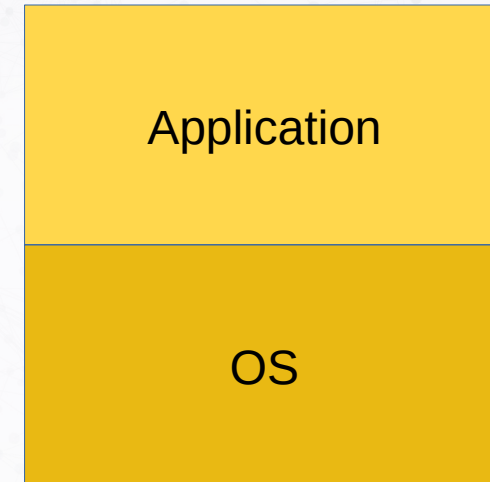
Colosseum Experiment

- Steps:
 - Run software
 - Collect data
 - Save data on the drive
(the proxy file server,
we shall see later)



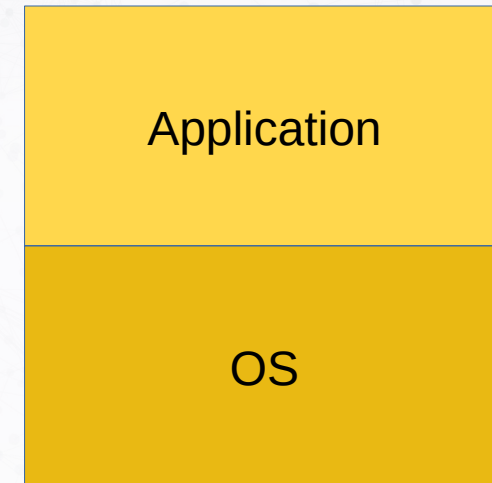
Colosseum Experiment SW

- Software
 - OS
 - Custom/tailored for experiments



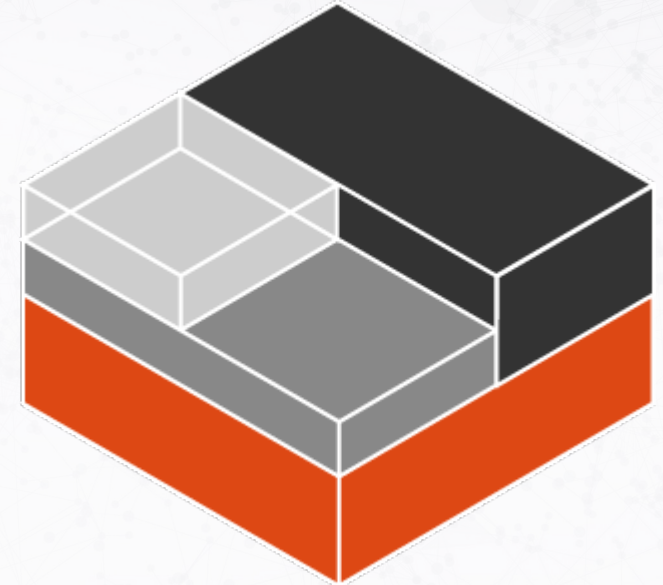
Virtual Machine

- Software is emulated in SRNs
- Linux Containers (LXC)
- Provides nifty clean packaging



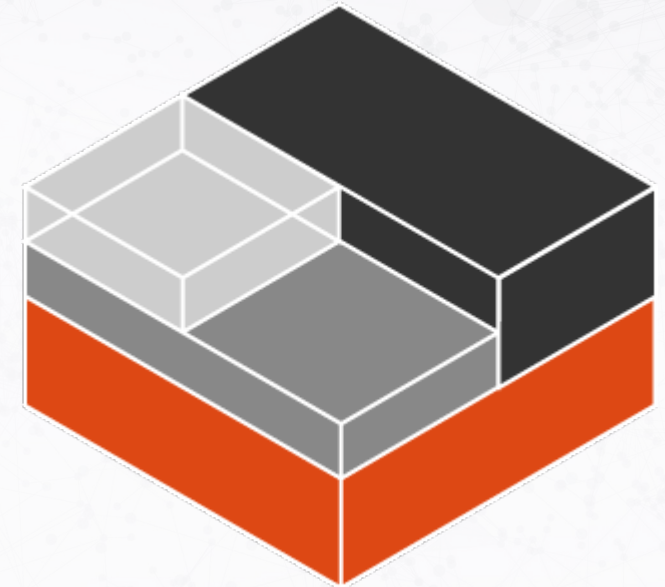
LXC

- Uses Linux kernel namespaces
- Provides hardware controlled access
- Provides process isolation
 - From most hardware
 - From other processes

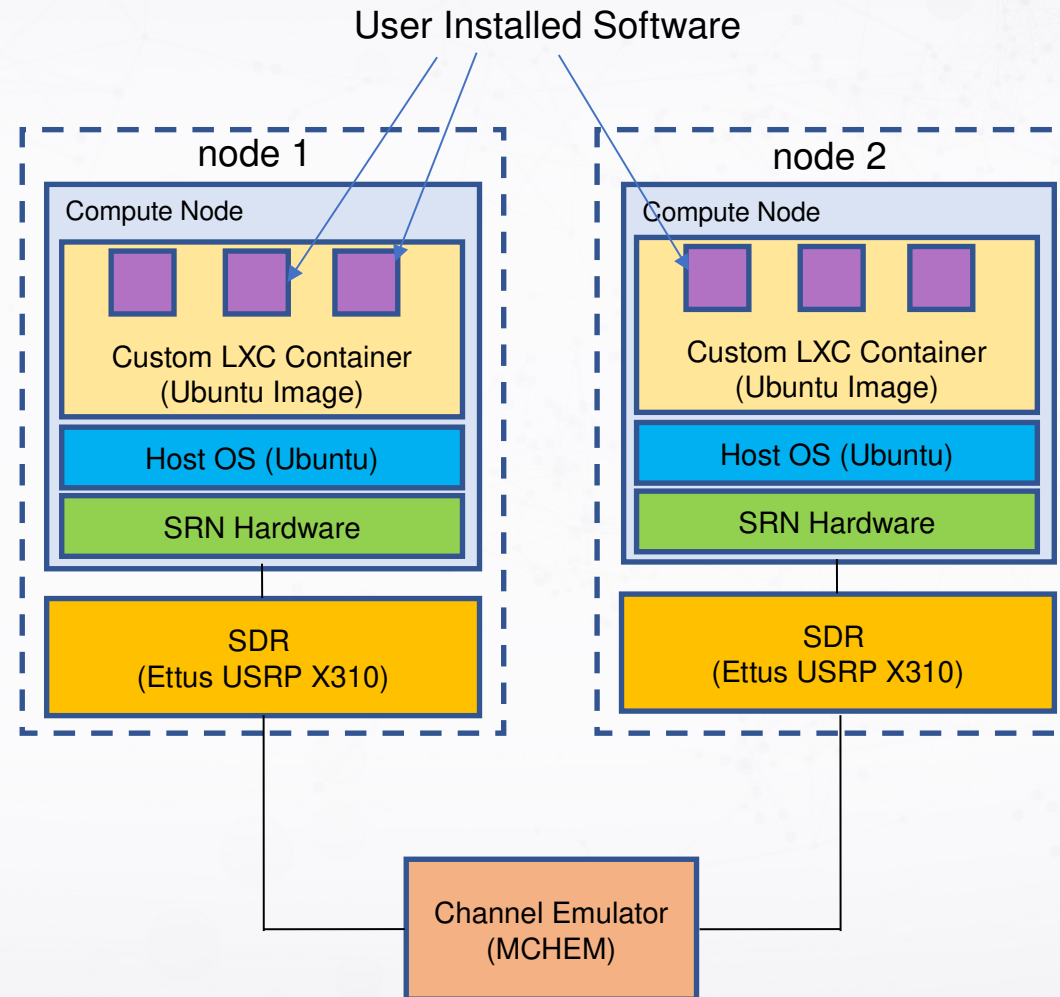


Colosseum containers, recap

- We use containers because:
 -) a way to package experiment software
 -) comprehensive of OS and stacks
 -) provide process isolation and hardware controlled access

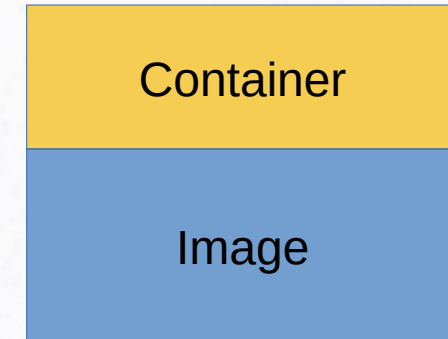


Colosseum node architecture



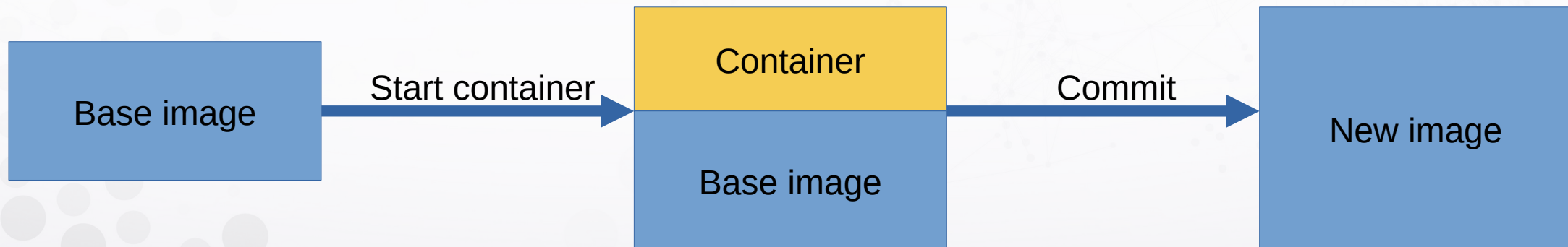
LXC – image and containers

- An **LXC image** is a filesystem comprising the OS and any software
 - ▷ Immutable, static
- An **LXC container** is a running instance of an image
 - ▷ Can be changed, possibly saved as images



LXC – packaging lifecycle

- 1 Download a standard image
- 2 Start a container for that image
- 3 Customize the container
- 4 Commit the container to a new image
- 5 Upload the new image



Colosseum image repo

- A special server, proxy-file
- Folder for common images
 - ▷ /share/nas/common
- Folder for custom images
 - ▷ /share/nas/<your_group>/images

Proxy-file
server

An example, packaging for mobile experiment

- Goal:
- Package software in an image with GNURadio executables
- Make it ready to use in a Colosseum experiment

GNURadio?



- Software development toolkit
- Compose signal processing blocks
- Usable with USRP x310 radios (Colosseum main RF hardware)

GNURadio companion

transmitter.grc - /root (on neu-gladiator-1-gladiator-test-container-srn19)

File Edit View Run Tools Help

Options
Output Language: Python
Generate Options: QT GUI

Variable
Id: samp_rate
Value: 1M

Signal Source
Sample Rate: 1M
Waveform: Triangle
Frequency: 100k
Amplitude: 1
Offset: 0
Initial Phase (Radians): 0

Throttle
Sample Rate: 1M

UHD: USRP Sink
Sync: Unknown PPS
Samp rate (Sps): 1M
Ch0: Center Freq (Hz): 1G
Ch0: Gain Value: 100
Ch0: Gain Type: Absolute (dB)
Ch0: Antenna: TX/RX

Core

- Audio
- Boolean Operators
- Byte Operators
- Channelizers
- Channel Models
- Coding
- Control Port
- Debug Tools
- Deprecated
- Digital Television
- Equalizers
- Error Coding
- File Operators
- Filters
- Fourier Analysis
- GUI Widgets
- Impairment Mod
- Instrumentation
- Level Controller
- Math Operators
- Measurement Tools
- Message Tools
- Misc
- Modulators
- Networking Tools

<<< Welcome to GNU Radio Companion 3.8.2.0 >>>

Block paths:
/usr/share/gnuradio/grc/-
blocks

Id	Value
Imports	
Variables	
samp_rate	1000000

GNURadio features

- Send and receive arbitrary wireless signals
- Develop and test new wireless modulation solutions
- Take advantage from the extensive block library, e.g.,
 - Use existing blocks for 802.11 communication

Creating an image for Colosseum

- Pre-requisites:
- A Linux environment
- LXC/LXD installed
- A Colosseum account
- SSH proxy setup (details on the website wiki)

Step 1 – Base image retrieval

- Some starting points are in the Proxy-file server:
 - base-1604-cuda.tar.gz
 - base-1604-nocuda.tar.gz
 - base-2104.tar.gz
- Folders:
 - share/nas/common/
 - share/nas/gladiators/
images/

Step 1 – Base image retrieval

- Some starting points are in the Proxy-file server:
 - base-1604-cuda.tar.gz
 - base-1604-nocuda.tar.gz
 - base-2104.tar.gz
- Folders:
 - share/nas/common/
 - share/nas/gladiators/images/

```
$> rsync -vP -e ssh file-proxy:/share/nas/gladiators/images/base-2104.tar.gz ./
```

Step 2 – Configuration check

- LXD requires some configuration for running unprivileged containers (safe option)
- Check */etc/subuid* and */etc/subgid* files

```
lxd:100000:65536  
root:100000:65536
```

- Restart LXD

```
$> systemctl restart lxd
```

Step 3 – Base image import in LXC

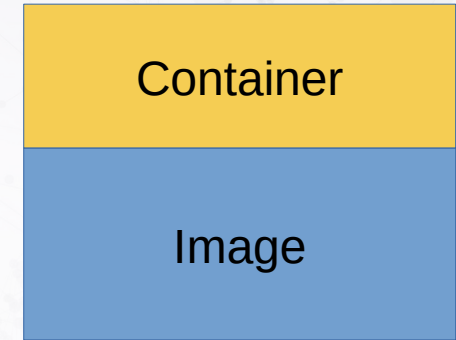
- Need to import the compressed file in LXC

Image

```
$> lxc image import base-2104.tar.gz --alias base  
$> lxc image list
```


Step 4 – Start a container

- Initialize a container
- Start the container
- Start a terminal interface for the container



```
$> lxc init local:base my-cont  
$> lxc start my-cont  
$> lxc exec my-cont /bin/bash
```

Step 5 – Customize

- You have full-root access to the system
- The current administrative password is “toor”

```
#> passwd
```

- In Ubuntu systems, software can be conveniently installed through the apt interface:

```
#> apt update  
#> apt install gnuradio grr1.2-gtk-3.0
```

Step 6 – Create custom image

- Ready to “commit” our work, and save it in a new LXC image



```
#> exit
$> lxc stop my-cont
$> lxc publish my-cont --alias my_image
$> lxc image list
```


Step 7 – Upload the custom image

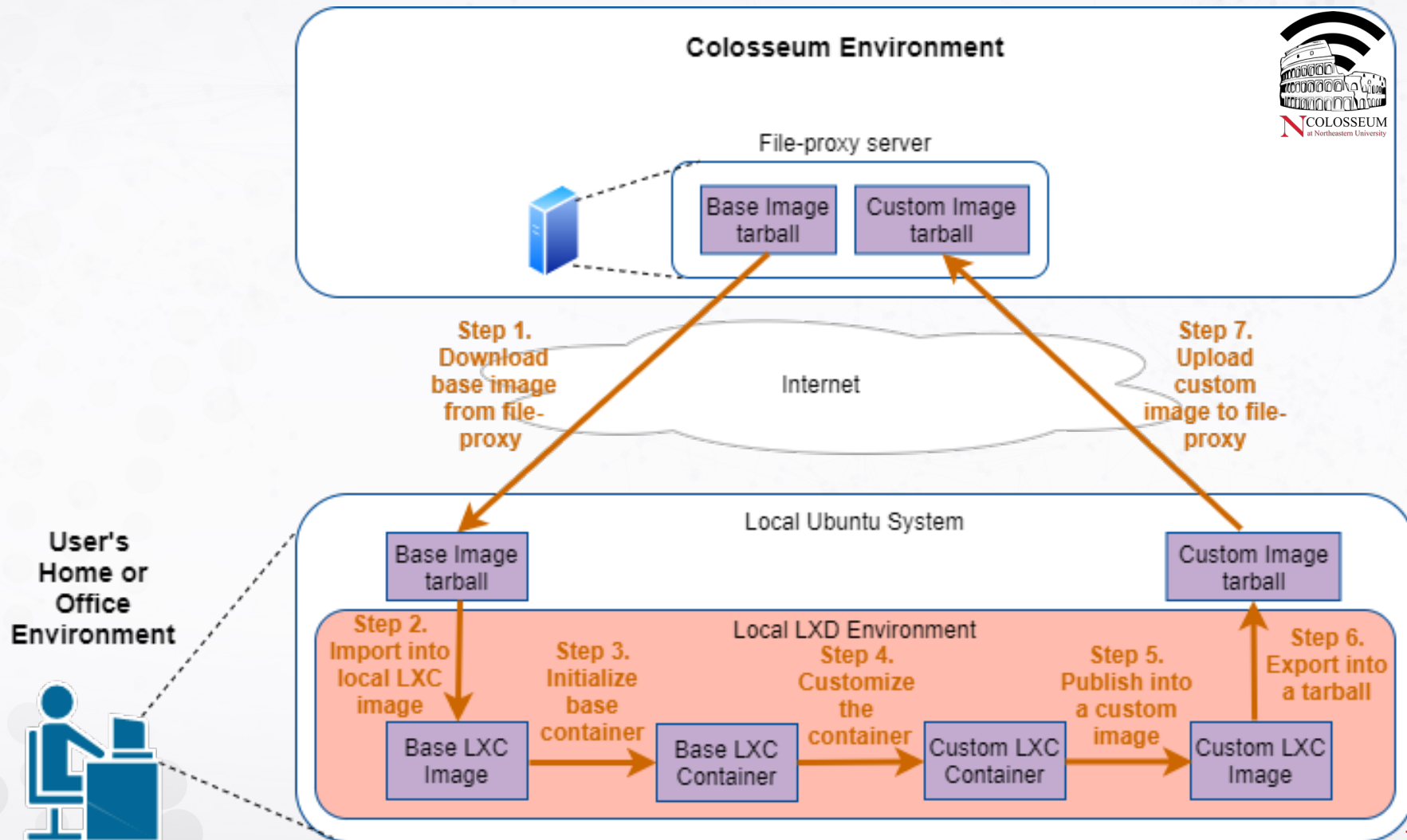
- Export the image as a compressed archive
- Upload it on Colosseum for experiments
- The image identifier **must** be unique for the team

```
$> lxc image export my_image <unique_id>  
$> rsync -vP -e ssh <unique_id>.tar.gz file-proxy:/share/nas/gladiators/images/
```

Reminders

- LXC images are immutable
- Every experiment will be consistent and repeatable
- You cannot save data in the Colosseum container

Recap





Institute for the Wireless Internet of Things

at Northeastern University

Proxy File Server



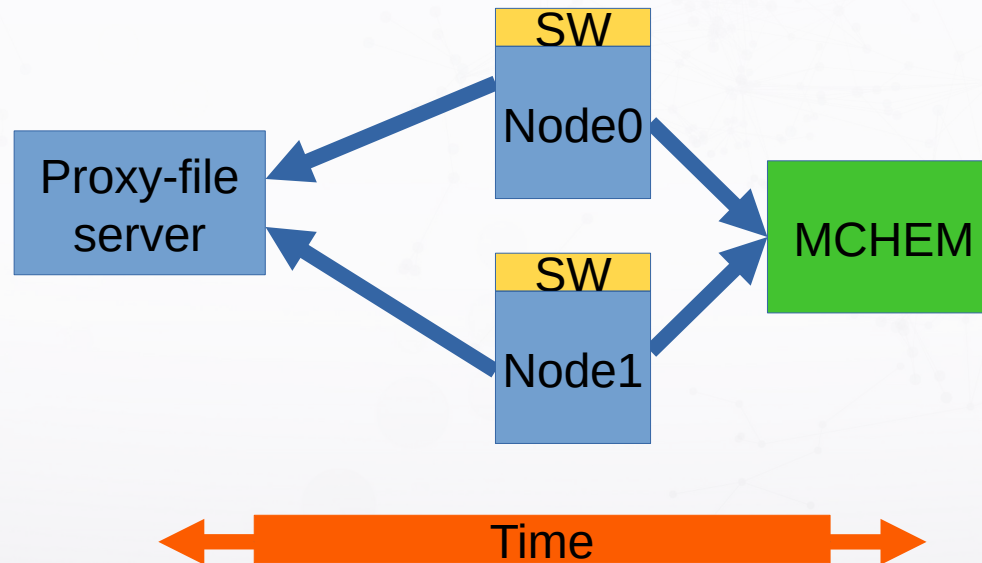
Platforms for Advanced
Wireless Research



NCOLOSSEUM
at Northeastern University

Save experiment data

- The LXC container of your image is run with the folder ***share*** pointing to the remote folder */share/<your_team>/reservation/<reservation_id>*



Save experiment data

- Data must be written as an unprivileged users

```
#> su srn-user  
$> cp log.data /share/
```


Load Experiment configuration

- The same applies for fetching data from the server and store it in the containers

```
#> cp /share/my.conf .
```

Snapshotting the container

- Another way of saving configurations and setup of a container is by performing a snapshot
- Within the running container, it results in a “publish & export”, producing a archived image, saved on file-proxy server

```
#> colosseumcli snapshot my_second_unique_id
```



Institute for the Wireless Internet of Things

at Northeastern University

Experimenting



Platforms for Advanced
Wireless Research



NCOLOSSEUM
at Northeastern University

Step 1 – Setup of the reservation

- Upload the image prepared with GNUMRadio
- Make and start a reservation for two nodes, using the image
- Check the SRNs identifier

The screenshot shows the 'Request New Reservation' page on the experiments.colosseum.net website. The page has a navigation bar with 'Home', 'Reservations', and 'Batch Jobs'. The 'Reservations' section is active.

The reservation form includes the following fields:

- Name:** A dropdown menu is open, showing a list of images. 'my-colosseum-image' is selected and highlighted in blue. Other options include cellos-bernardo, cellos-leo, frank-wifi-basic, michele-wifi-basic, shweta-oal-harqbr, shweta-oal-new, shweta-oaldevbr-w26-ulfix, shweta-srsite, srsite-basic-20-04, webinar-1604, common, base-1604-cuda, base-1604-nocuda, incumbent-active-v1-4, incumbent-active-v1-5, and incumbent-dsrc-v2-2.
- Start date:** A text input field.
- Start time:** A text input field.
- Duration:** A text input field with a note: 'Note: 5 minutes'.
- Number of SRNs:** A text input field.
- Node 1:** A dropdown menu with 'shweta-srsite' selected.

Below the form, there is a table titled 'Request New Reservation' showing resource availability for 'Tue 14 July'. The table has columns for time slots: 11:00 pm, 11:10 pm, 11:20 pm, 11:30 pm, and 11:40 pm. The rows are numbered 1 to 32. Some rows (6, 14, 16) are highlighted in red, indicating they are not available.

At the bottom right, there is a summary table:

	Current	Cost	Remaining
Tokens	2514	12	2502

Step 2 – Log in and start RF

- Log-in

```
$> ssh -Y <srn-hostname1>  
$> ssh -Y <srn-hostname2>
```

Forward of the
graphical front-
end through ssh

- Start RF (scenario 1009, with CF 1GHz and 0dB of path loss)

```
#> colosseumcli rf start 1009 -c
```

Step 3 – Start software

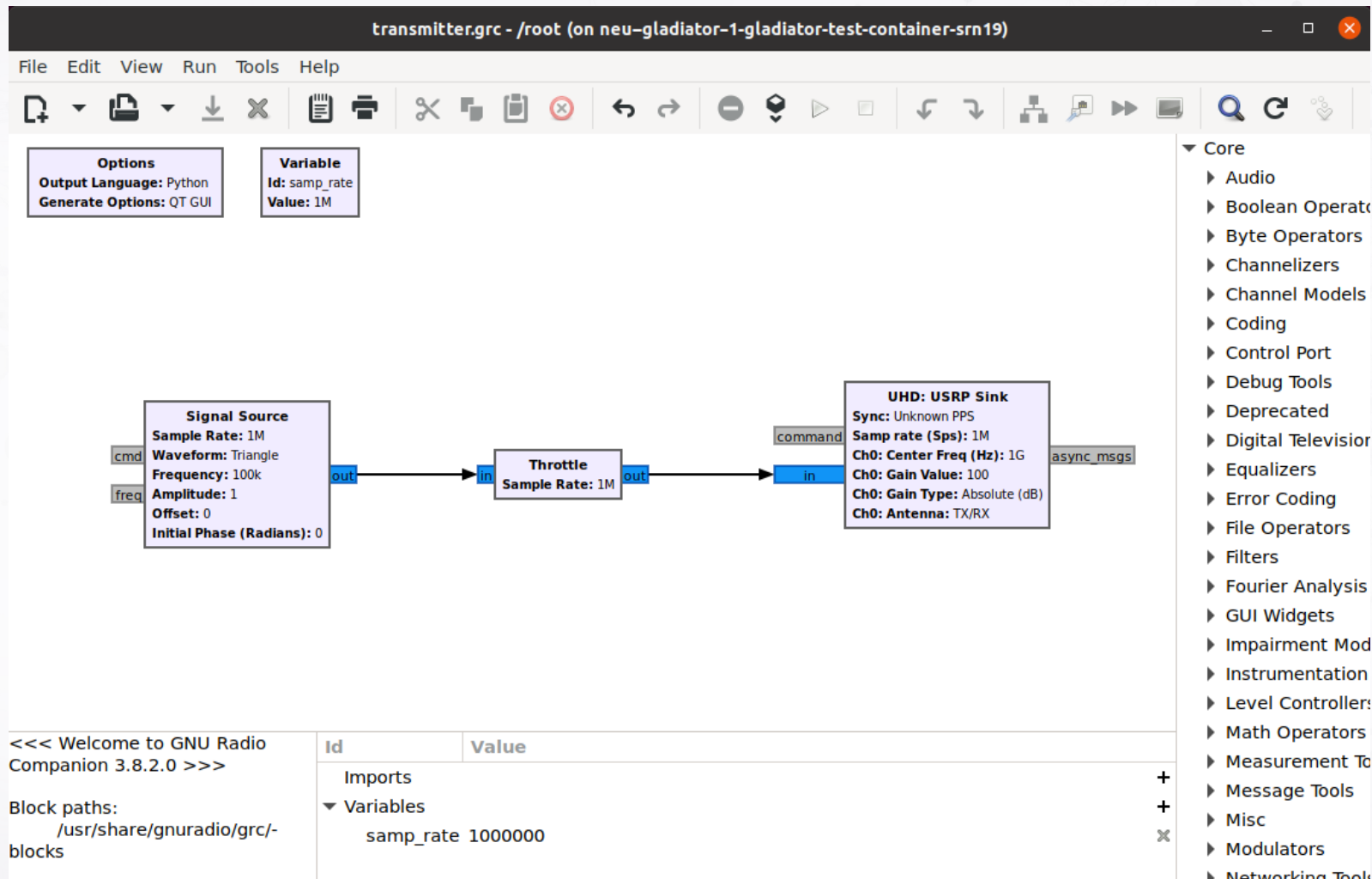
- In both terminals:
- update the FPGA image:

```
#> ./flash_fpga_x310.sh
```

- start GNURadio companion:

```
#> gnuradio-companion
```


Step 4 – Design the transmitter



Step 4 – Design the receiver

receiver.grc - /root (on neu-gladiator-1-gladiator-test-container-srn20)

File Edit View Run Tools Help

Options
Output Language: Python
Generate Options: QT GUI

Variable
Id: samp_rate
Value: 1M

UHD: USRP Source
Sync: Unknown PPS
Samp rate (Sps): 1M
Ch0: Center Freq (Hz): 1G
Ch0: AGC: Default
Ch0: Gain Value: 100
Ch0: Gain Type: Absolute (dB)
Ch0: Antenna: RX2

QT GUI Time Sink
Number of Points: 1.024k
Sample Rate: 1M
Autoscale: No

<<< Welcome to GNU Radio Companion 3.8.2.0 >>>

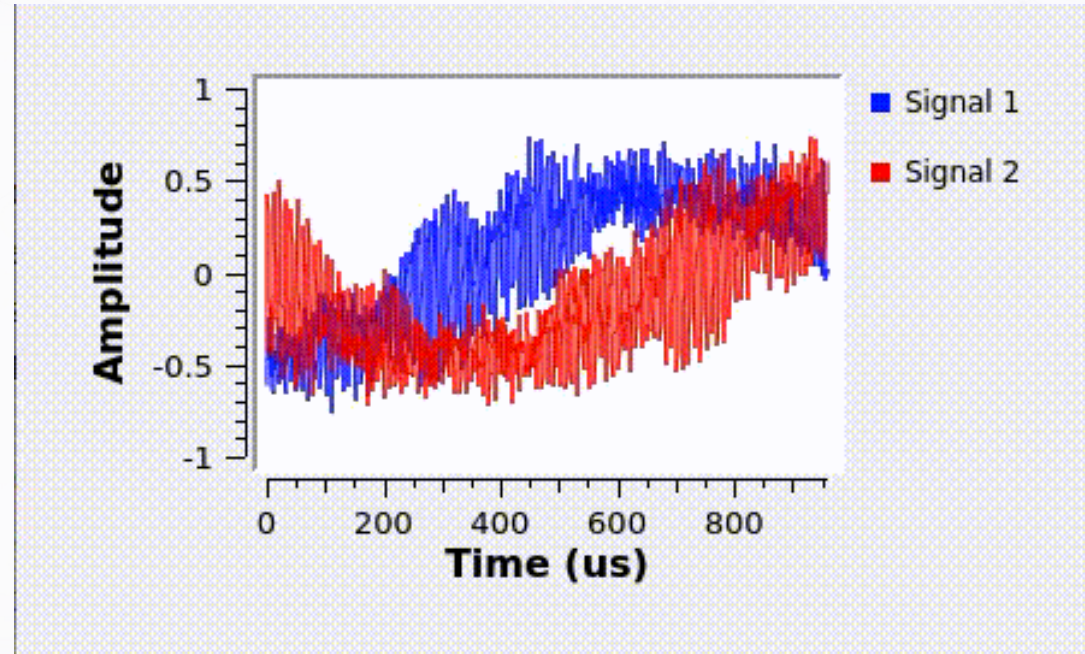
Block paths:
/usr/share/gnuradio/grc/-
blocks

Id	Value
Imports	
Variables	
samp_rate	1000000

Core

- Audio
- Boolean Operators
- Byte Operators
- Channelizers
- Channel Models
- Coding
- Control Port
- Debug Tools
- Deprecated
- Digital Television
- Equalizers
- Error Coding
- File Operators
- Filters
- Fourier Analysis
- GUI Widgets
- Impairment Models
- Instrumentation
- Level Controller
- Math Operators
- Measurement Tools
- Message Tools
- Misc

Step 5 – Observe the spectrum



Colosseum Freshdesk Wiki Links

- Accessing Colosseum Servers:
 - <https://colosseumneu.freshdesk.com/support/solutions/articles/61000253362-accessing-colosseum-resources>
- SSH Proxy setup
 - <https://colosseumneu.freshdesk.com/support/solutions/articles/61000253369-ssh-proxy-setup>
- File transfer using scp and rsync:
 - <https://colosseumneu.freshdesk.com/a/solutions/articles/61000253365>
- Transferring base image from NAS to local machine:
 - <https://colosseumneu.freshdesk.com/support/solutions/articles/61000253371-transferring-the-base-lxc-image-from-the-nas>
- Details on LXD commands and configuration:
 - <https://colosseumneu.freshdesk.com/a/solutions/articles/61000253368>
- Working on a container locally and preparing it for upload:
 - <https://colosseumneu.freshdesk.com/a/solutions/articles/61000253428>
- Uploading a customized container image to File Proxy server:
 - <https://colosseumneu.freshdesk.com/support/solutions/articles/61000253372-upload-an-lxc-container>
- Installing Colosseum CLI and taking a snapshot of your container
 - <https://colosseumneu.freshdesk.com/support/solutions/articles/61000253397-colosseum-cli>

Colosseum containers

- Q & A