



FireSim

Building/Deploying Simulations

FireSim Intensive

Chisel Community Conference 2018

Speaker: Sagar Karandikar



Berkeley Architecture Research

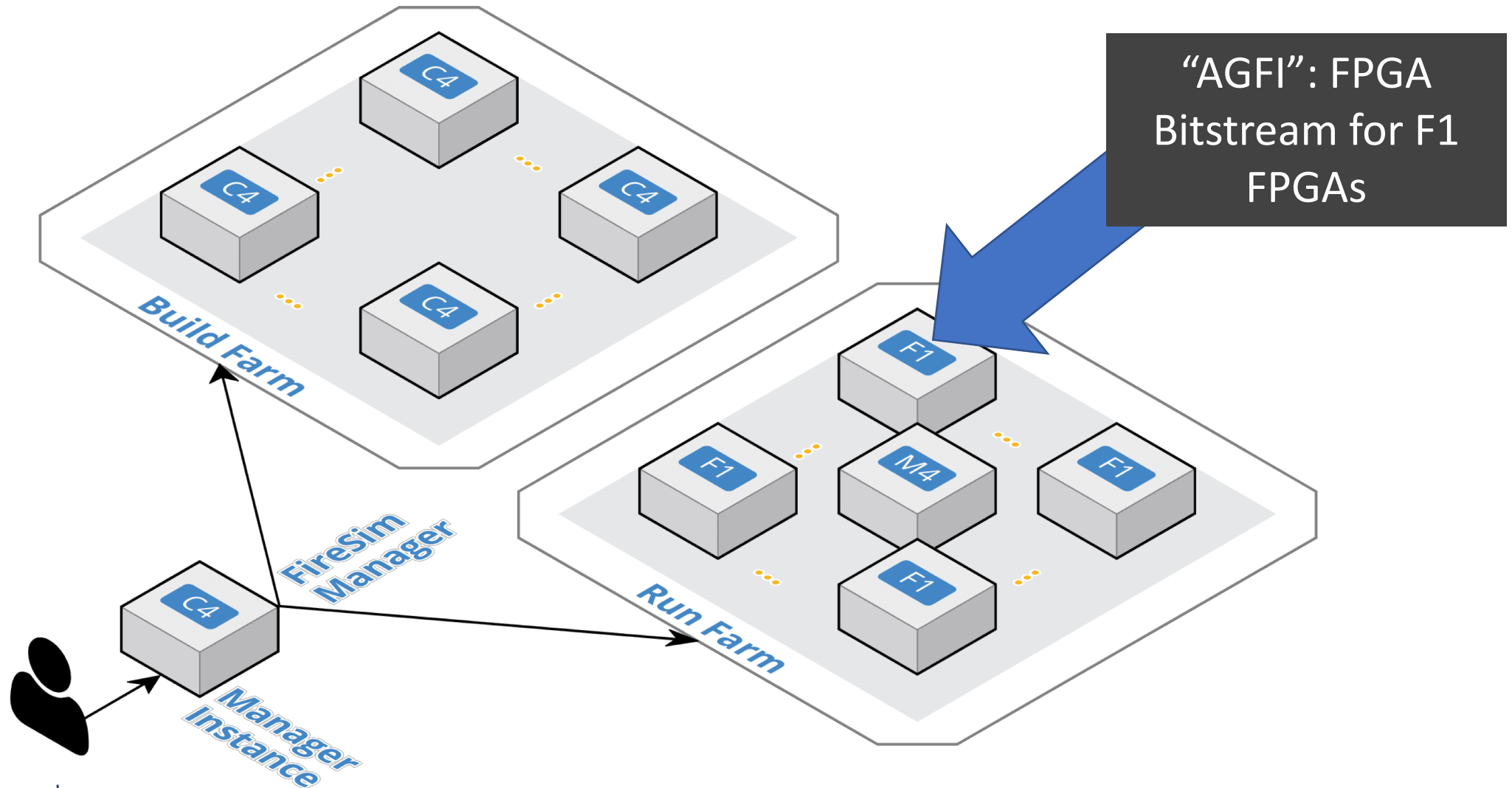


What will we cover?

- Building FireSim FPGA images for a set of targets
- Choosing targets at runtime
- Configuring target software
- Managing EC2 infrastructure for simulations
- Running a simulation
- Customizing workloads: SPEC example



Background Terminology





First-time User Setup

- Won't discuss this today, but we've documented the process of starting with AWS/FireSim from scratch here:

<http://docs.firesim/en/latest/Initial-Setup/index.html>

- For the following walkthrough, we assume that you've setup a Manager instance and cloned firesim into `~/firesim`



Let's simulate the following system:

Target Design:

- One Rocket Chip-based node (RTL)
 - 4 Rocket Cores
 - 16K L1 I\$, D\$
 - Block Device, UART, Serial Adapter
 - No NIC
- 4 MB LLC (RTL Model)
- DDR3 Memory System
- No network
- Boot vanilla buildroot-Linux distro

Host Resources:

- One manager instance (c4.4xlarge)
- One F1 instance (f1.2xlarge, single FPGA)



Using the `firesim` manager command line

From inside `~/firesim`, **source** `sourceme-fl-manager.sh`

This properly sets up your environment and puts `firesim` on your `$PATH`

This means that we can now call `firesim` from anywhere on the instance. It will always run from the directory:

```
your_firesim_clone/deploy/
```

in this case:

```
~/firesim/deploy/
```



Configuring the Manager. 4 files in firesim/deploy/

config_build_recipes.ini

```

1 # See docs/Advanced-Usage/Manager/Manager-Configuration-Files.rst for documentation of all of
2 # these params.
3 # This file contains sections that describe hardware designs that /can/ be built.
4 # edit config_build.ini to actually "turn on" a config to be built when you run
5 # builddefi.
6
7 [[firesim-singlecore-no-nic-lbp]]
8 DESIGN=FireSimNoNIC
9 TARGET_CONFIG=FireSimRocketChipSingleCoreConfig
10 PLATFORM_CONFIG=FireSimConfig
11 instancetype=C4_4xlarge
12 deploytriple=None
13
14 [[firesim-quadcore-nic-ddr3-11c4mb]]
15 DESIGN=FireSim
16 TARGET_CONFIG=FireSimRocketChipQuadCoreConfig
17 PLATFORM_CONFIG=FireSimDDR3FRFCPLL4MConfig
18 instancetype=C4_4xlarge
19 deploytriple=None
20
21 [[firesim-quadcore-no-nic-ddr3-11c4mb]]
22 DESIGN=FireSimNoNIC
23 TARGET_CONFIG=FireSimRocketChipQuadCoreConfig
24 PLATFORM_CONFIG=FireSimDDR3FRFCPLL4MConfig
25 instancetype=C4_4xlarge
26 deploytriple=None
27
28 # BOOM-based targets
29 [[fireboom-singlecore-no-nic-ddr3-11c4mb]]
30 DESIGN=FireSimNoNIC
31 TARGET_CONFIG=FireSimBooMConfig
32 PLATFORM_CONFIG=FireSimDDR3FRFCPLL4MConfig
33 instancetype=C4_4xlarge
34 deploytriple=None
35
36 [[fireboom-singlecore-nic-ddr3-11c4mb]]
37 DESIGN=FireBooM
38 TARGET_CONFIG=FireSimBooMConfig
39 PLATFORM_CONFIG=FireSimDDR3FRFCPLL4MConfig
40 instancetype=C4_4xlarge
41 deploytriple=None

```

config_build.ini

```

1 # See docs/Advanced-Usage/Manager/Manager-Configuration-Files.rst for documentation of all of
2 # these params.
3
4 [[build]]
5 bucketname=firesim-721179603761
6 buildinstancemarket=ondemand
7 spotinterruptionbehavior=terminate
8 spotmaxprice=ondemand
9
10 [[builds]]
11 # this section references builds defined in config_build_recipes.ini
12 # if you add a build here, it will be built when you run builddefi
13 firesim-singlecore-no-nic-lbp
14 firesim-quadcore-no-nic-ddr3-11c4mb
15 fireboom-singlecore-no-nic-ddr3-11c4mb
16 fireboom-singlecore-nic-ddr3-11c4mb
17
18 [[agfistoshare]]
19 fireboom-singlecore-no-nic-lbp
20 fireboom-singlecore-no-nic-ddr3-11c4mb
21 fireboom-quadcore-no-nic-ddr3-11c4mb
22 fireboom-quadcore-nic-ddr3-11c4mb
23
24 [[sharewithaccounts]]
25 accountids=123456789012

```

config_hwdb.ini

```

1 # See docs/Advanced-Usage/Manager/Manager-Configuration-Files.rst for documentation of all of
2 # these params.
3
4 # Hardware configs represent a combination of an agfi, a deploytriple override
5 # (if needed), and a custom runtime config (if needed)
6
7 # The AGFIs provided below are public and available to all users.
8 # Only AGFIs for the latest release of FireSim are guaranteed to be available.
9 # If you are using an older version of FireSim, you will need to generate your
10 # own images.
11
12 [[firesim-singlecore-no-nic-lbp]]
13 agfi=agfi-0584a1a71df6a005a
14 deploytripleoverride=None
15 customruntimeconfig=None
16
17 [[firesim-quadcore-no-nic-ddr3-11c4mb]]
18 agfi=agfi-06b9b705ab9af1238
19 deploytripleoverride=None
20 customruntimeconfig=None
21
22 [[firesim-quadcore-nic-ddr3-11c4mb]]
23 agfi=agfi-030b49bce9bd5ef96
24 deploytripleoverride=None
25 customruntimeconfig=None
26
27 [[fireboom-singlecore-no-nic-ddr3-11c4mb]]
28 agfi=agfi-090491454199fb160
29 deploytripleoverride=None
30 customruntimeconfig=None
31
32 [[fireboom-singlecore-nic-ddr3-11c4mb]]
33 agfi=agfi-0df9101df7b7ff708
34 deploytripleoverride=None
35 customruntimeconfig=None

```

config_runtime.ini

```

1 # See docs/Advanced-Usage/Manager/Manager-Configuration-Files.rst for documentation of all of
2 # these params.
3
4 [[runtime]]
5
6 [[runfpga]]
7 uniformtag=mainrunfpga
8
9
10 [[instancemarket=ondemand]]
11 spotinterruptionbehavior=terminate
12 spotmaxprice=ondemand
13
14 [[targetconfig]]
15 topology=to_net_config
16 no_net_num_nodes=1
17 linklatency=5405
18 switchinglatency=10
19 networkwidth=200
20 profileinterval=1
21
22 # This references a section from config_hwconfigs.ini
23 # In homogeneous configurations, use this to set the hardware config deployed
24 # for all simulators
25 defaulthwconfig=fireboom-quadcore-no-nic-ddr3-11c4mb
26
27 [[tracing]]
28 enable=no
29 startcycle=0
30 endcycle=1
31
32 [[workload]]
33 workloadname=linux-uniform.json
34 terminateoncompletion=no

```



Setting up the manager to build FPGA images

```
0 / mosh-client (tmux) #1
config_build_recipes.ini buffers
1 # Build-time design configuration for the FireSim Simulation Manager
1 # See docs/Advanced-Usage/Manager/Manager-Configuration-Files.rst for documentation of all of
  these params.
2
3 # this file contains sections that describe hardware designs that /can/ be built.
4 # edit config_build.ini to actually "turn on" a config to be built when you run
5 # buildafi
6
7 [firesim-singlecore-no-nic-lbp]
8 DESIGN=FireSimNoNIC
9 TARGET_CONFIG=FireSimRocketChipSingleCoreConfig
10 PLATFORM_CONFIG=FireSimConfig
11 instancetype=c4.4xlarge
12 deploytriplet=None
13
14 [firesim-quadcore-nic-ddr3-1lc4mb]
15 DESIGN=FireSim
16 TARGET_CONFIG=FireSimRocketChipQuadCoreConfig
17 PLATFORM_CONFIG=FireSimDDR3FRFCFSLLC4MBConfig
18 instancetype=c4.4xlarge
19 deploytriplet=None
20
21 [firesim-quadcore-no-nic-ddr3-1lc4mb]
22 DESIGN=FireSimNoNIC
23 TARGET_CONFIG=FireSimRocketChipQuadCoreConfig
24 PLATFORM_CONFIG=FireSimDDR3FRFCFSLLC4MBConfig
25 instancetype=c4.4xlarge
26 deploytriplet=None
27
28 # BOOM-based targets
29 [fireboom-singlecore-no-nic-ddr3-1lc4mb]
30 DESIGN=FireBoomNoNIC
31 TARGET_CONFIG=FireSimBoomConfig
32 PLATFORM_CONFIG=FireSimDDR3FRFCFSLLC4MBConfig
33 instancetype=c4.4xlarge
34 deploytriplet=None
35
36 [fireboom-singlecore-nic-ddr3-1lc4mb]
37 DESIGN=FireBoom
38 TARGET_CONFIG=FireSimBoomConfig
```

```
0 / mosh-client (tmux) #1
config_build.ini buffers
2 # BUILDTIME/AGFI management configuration for the FireSim Simulation Manager
1 # See docs/Advanced-Usage/Manager/Manager-Configuration-Files.rst for documentation of all of
  these params.
3
4 [afibuild]
5 s3bucketname=firesim-721179603761
6 buildinstancemarket=ondemand
7 spotinterruptionbehavior=terminate
8 spotmaxprice=ondemand
9
10 [builds]
11 # this section references builds defined in config_build_recipes.ini
12 # if you add a build here, it will be built when you run buildafi
13 firesim-singlecore-no-nic-lbp
14 firesim-quadcore-no-nic-ddr3-1lc4mb
15 firesim-quadcore-nic-ddr3-1lc4mb
16 fireboom-singlecore-no-nic-ddr3-1lc4mb
17 fireboom-singlecore-nic-ddr3-1lc4mb
18
19 [agfistoshare]
20 firesim-singlecore-no-nic-lbp
21 firesim-quadcore-no-nic-ddr3-1lc4mb
22 firesim-quadcore-nic-ddr3-1lc4mb
23 fireboom-singlecore-no-nic-ddr3-1lc4mb
24 fireboom-singlecore-nic-ddr3-1lc4mb
25
26 [sharewithaccounts]
27 somebodyname=123456789012
```



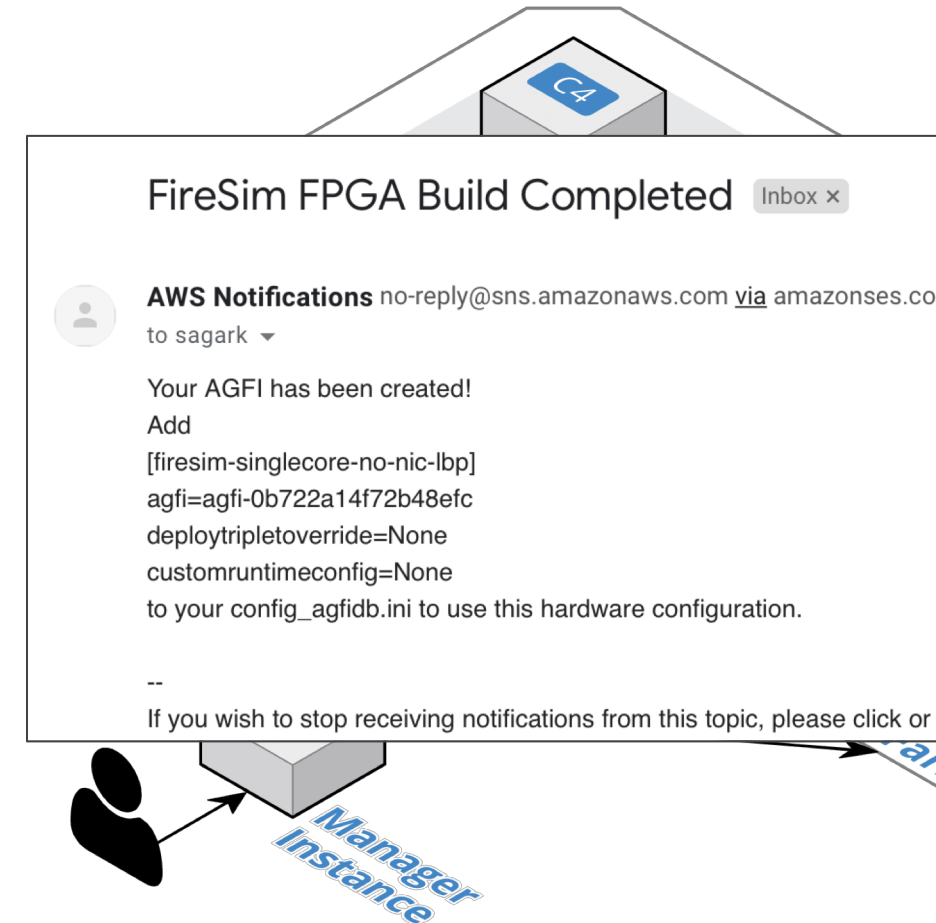


Running builds

- Once we've configured **what** we want to build, let's build it

```
$ firesim buildafi
```

- This completely automates the process. For each design, in-parallel:
 - Launch a build instance (c4.4xlarge)
 - Run FireSim generator (Chisel/FIRRTL/MIDAS)
 - Ship infrastructure to build instances, run Vivado FPGA builds in parallel
 - Collect results back onto manager instance
 - ~/firesim/deploy/results-workload/TIMESTAMP-config/ contains final results + QoR information + dcps to open in Vivado for further manual tuning
 - Email you the entry to put into config_hwdb.ini
 - Terminate the build instance



Status - We have FPGA images for the simulator of our design



Target Design:

- ✓ One Rocket Chip-based node (RTL)
 - 4 Rocket Cores
 - 16K L1 I\$, D\$
 - Block Device, UART, Serial Adapter
 - No NIC
- ✓ 4 MB LLC (RTL Model)
- ✓ DDR3 Memory System
- ✓ No network
 - Boot vanilla buildroot-Linux distro

Host Resources:

- ✓ One manager instance (c4.4xlarge)
 - One F1 instance (f1.2xlarge, single FPGA)



Now, let's build our target-software

- We need:
 - bbl + vmlinux image: Berkeley BootLoader + Linux kernel image as payload
 - Root Filesystem disk image
- FireSim provides two levels of workload automation:
 - `firesim-software` repo for automatically building base-images for various distros (e.g. buildroot or Fedora) that are compatible with Rocket / BOOM

```
cd firesim/sw/firesim-software
./sw-manager.py -c br-disk.json build # simple buildroot distro
```

This lets us use the `linux-uniform.json` workload, which we will see later
 - `firesim` manager / workload generation tool can bake custom workloads into rootfses
 - We'll get to this later



Status

Target Design:

- ✓ One Rocket Chip-based node (RTL)
 - 4 Rocket Cores
 - 16K L1 I\$, D\$
 - Block Device, UART, Serial Adapter
 - No NIC
- ✓ 4 MB LLC (RTL Model)
- ✓ DDR3 Memory System
- ✓ No network
- ✓ Boot vanilla buildroot-Linux distro

Host Resources:

- ✓ One manager instance (c4.4xlarge)
 - One F1 instance (f1.2xlarge, single FPGA)



Setting up the manager's runtime configuration

config_build_recipes.ini

```

1 # See docs/Advanced-Usage/Manager/Manager-Configuration-Files.rst for documentation of all of
2 # these params.
3 # This file contains sections that describe hardware designs that /can/ be built.
4 # edit config_build.ini to actually "turn on" a config to be built when you run
5 # buildafi.
6
7 [[firesim-singlecore-no-nic-lbp]]
8 DESIGN=FireSimNoNIC
9 TARGET_CONFIG=FireSimRocketChipSingleCoreConfig
10 PLATFORM_CONFIG=FireSimConfig
11 instancetype=C4_4xlarge
12 deploytriple=None
13
14 [[firesim-quadcore-nic-ddr3-11c4mb]]
15 DESIGN=FireSim
16 TARGET_CONFIG=FireSimRocketChipQuadCoreConfig
17 PLATFORM_CONFIG=FireSimDDR3FRFCPLL4MConfig
18 instancetype=C4_4xlarge
19 deploytriple=None
20
21 [[firesim-quadcore-no-nic-ddr3-11c4mb]]
22 DESIGN=FireSimNoNIC
23 TARGET_CONFIG=FireSimRocketChipQuadCoreConfig
24 PLATFORM_CONFIG=FireSimDDR3FRFCPLL4MConfig
25 instancetype=C4_4xlarge
26 deploytriple=None
27
28 # BOOM-based targets
29 [[fireboom-singlecore-no-nic-ddr3-11c4mb]]
30 DESIGN=FireSimNoNIC
31 TARGET_CONFIG=FireSimBoomConfig
32 PLATFORM_CONFIG=FireSimDDR3FRFCPLL4MConfig
33 instancetype=C4_4xlarge
34 deploytriple=None
35
36 [[fireboom-singlecore-nic-ddr3-11c4mb]]
37 DESIGN=FireBoom
38 TARGET_CONFIG=FireSimBoomConfig
39 PLATFORM_CONFIG=FireSimDDR3FRFCPLL4MConfig
40 instancetype=C4_4xlarge
41 deploytriple=None

```

config_build.ini

```

1 # See docs/Advanced-Usage/Manager/Manager-Configuration-Files.rst for documentation of all of
2 # these params.
3
4 [[builds]]
5 # this section references builds defined in config_build_recipes.ini
6 # if you add a build here, it will be built when you run buildafi
7
8 [[firesim-singlecore-no-nic-lbp]]
9 firesim-singlecore-no-nic-ddr3-11c4mb
10
11 [[firesim-quadcore-nic-ddr3-11c4mb]]
12 firesim-quadcore-nic-ddr3-11c4mb
13
14 [[fireboom-singlecore-no-nic-ddr3-11c4mb]]
15 fireboom-singlecore-nic-ddr3-11c4mb
16
17 [[agfistoshare]]
18
19 [[sharewithaccounts]]
20
21 [[somebodysname]]
22

```

config_hwdb.ini

```

1 # See docs/Advanced-Usage/Manager/Manager-Configuration-Files.rst for documentation of all of
2 # these params.
3
4 # Hardware configs represent a combination of an agfi, a deploytriple override
5 # (if needed), and a custom runtime config (if needed)
6
7 # The AGFIs provided below are public and available to all users.
8 # Only AGFIs for the latest release of FireSim are guaranteed to be available.
9 # If you are using an older version of FireSim, you will need to generate your
10 # own images.
11
12 [[firesim-singlecore-no-nic-lbp]]
13 agfi=agfi-9584a1a71df6a005a
14 deploytripleoverride=None
15 customruntimeconfig=None
16
17 [[firesim-quadcore-no-nic-ddr3-11c4mb]]
18 agfi=agfi-96b9b705ab9af1238
19 deploytripleoverride=None
20 customruntimeconfig=None
21
22 [[firesim-quadcore-nic-ddr3-11c4mb]]
23 agfi=agfi-93b049bce9bd5ef96
24 deploytripleoverride=None
25 customruntimeconfig=None
26
27 [[fireboom-singlecore-nic-ddr3-11c4mb]]
28 agfi=agfi-990491454199fb160
29 deploytripleoverride=None
30 customruntimeconfig=None
31
32 [[fireboom-singlecore-no-nic-ddr3-11c4mb]]
33 agfi=agfi-9df9101df7b7ff708
34 deploytripleoverride=None
35 customruntimeconfig=None

```

config_runtime.ini

```

1 # See docs/Advanced-Usage/Manager/Manager-Configuration-Files.rst for documentation of all of
2 # these params.
3
4 [[runtime]]
5
6 [[runfarm]]
7
8 [[uniformtopmainrunfarm]]
9
10 [[fire_16xlarge-0]]
11 [[fire_16xlarge-1]]
12
13 [[fire_2xlarge-1]]
14
15 [[runinstancemarket-ondemand]]
16 [[spotinterruptionbehavior-terminate]]
17 [[spotmaxprice-ondemand]]
18
19 [[targetconfig]]
20 [[topology-to-net-config]]
21 [[no_net_num_nodes=1]]
22 [[linklatency=5405]]
23 [[switchinglatency=10]]
24 [[netbandwidth=200]]
25 [[profileinterval=1]]
26
27 # This references a section from config_hwconfigs.ini
28 # In homogeneous configurations, use this to set the hardware config deployed
29 # for all simulators
30 [[defaulthwconfig=fireboom-singlecore-no-nic-ddr3-11c4mb]]
31
32 [[tracing]]
33 enable=no
34 startcycle=0
35 endcycle=1
36
37 [[workload]]
38 workloadname=linux-uniform.json
39 terminateoncompletion=no

```





Setting up the manager's runtime configuration

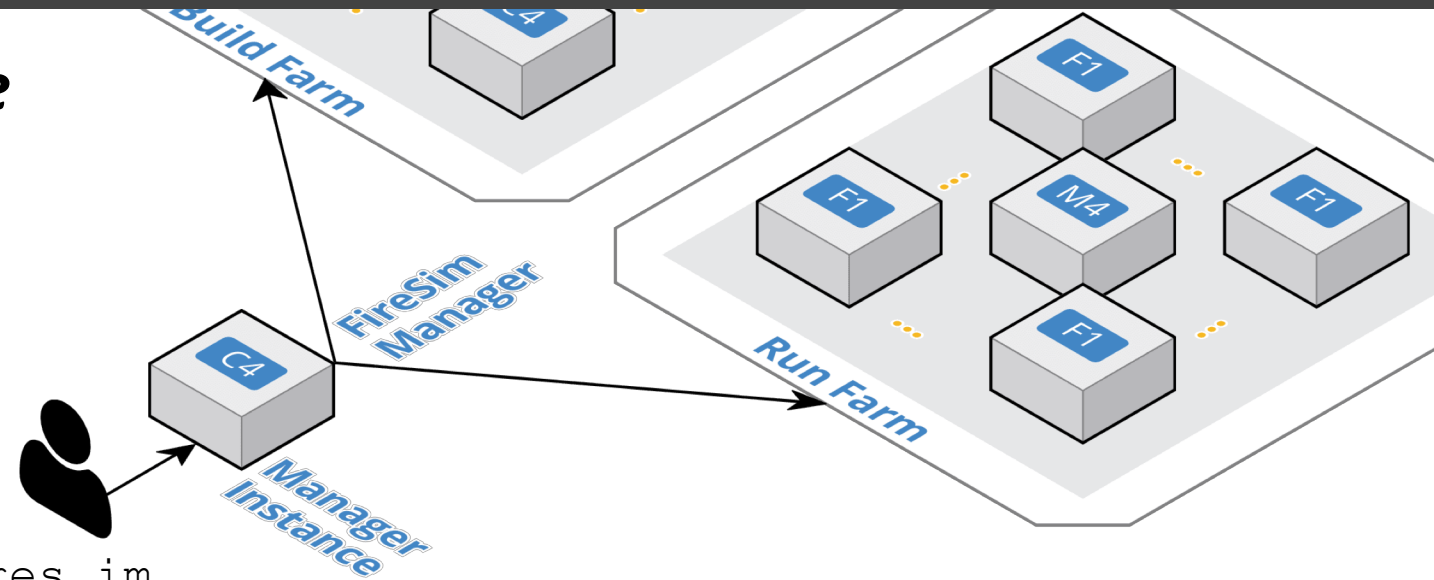
```
0 / mosh-client (tmux) #1
config_hwdb.ini buffers
1 # Hardware config database for FireSim Simulation Manager
2 # See docs/Advanced-Usage/Manager/Manager-Configuration-Files.rst for documentation of all of
3 # these params.
4 # Hardware configs represent a combination of an agfi, a deploytriplet override
5 # (if needed), and a custom runtime config (if needed)
6 # The AGFIs provided below are public and available to all users.
7 # Only AGFIs for the latest release of FireSim are guaranteed to be available.
8 # If you are using an older version of FireSim, you will need to generate your
9 # own images.
10
11 [firesim-singlecore-no-nic-lbp]
12 agfi=agfi-0584a1a71df6a005a
13 deploytripletoverride=None
14 customruntimeconfig=None
15
16 [firesim-quadcore-no-nic-ddr3-llc4mb]
17 agfi=agfi-06b9b705ab9af1238
18 deploytripletoverride=None
19 customruntimeconfig=None
20
21 [firesim-quadcore-nic-ddr3-llc4mb]
22 agfi=agfi-030b49bce9bd5ef96
23 deploytripletoverride=None
24 customruntimeconfig=None
25
26 [fireboom-singlecore-nic-ddr3-llc4mb]
27 agfi=agfi-090491454199fb160
28 deploytripletoverride=None
29 customruntimeconfig=None
30
31 [fireboom-singlecore-no-nic-ddr3-llc4mb]
32 agfi=agfi-0df9101df7b7ff708
33 deploytripletoverride=None
34 customruntimeconfig=None
35
```

```
0 / mosh-client (tmux) #1
config_runtime.ini buffers
1 # RUNTIME configuration for the FireSim Simulation Manager
2 # See docs/Advanced-Usage/Manager/Manager-Configuration-Files.rst for documentation of all of
3 # these params.
4
5 [runfarm]
6 runfarmtag=mainrunfarm
7
8 f1_16xlarges=0
9 m4_16xlarges=0
10 f1_2xlarges=1
11
12 runinstancemarket=ondemand
13 spotinterruptionbehavior=terminate
14 spotmaxprice=ondemand
15
16 [targetconfig]
17 topology=no_net_config
18 no_net_num_nodes=1
19 linklatency=6405
20 switchinglatency=10
21 netbandwidth=200
22 profileinterval=-1
23
24 # This references a section from config_hwconfigs.ini
25 # In homogeneous configurations, use this to set the hardware config deployed
26 # for all simulators
27 defaulthwconfig=firesim-quadcore-no-nic-ddr3-llc4mb
28
29 [tracing]
30 enable=no
31 startcycle=0
32 endcycle=-1
33
34 [workload]
35 workloadname=linux-uniform.json
36 terminateoncompletion=no
```



Launching Simulation Instances

- Once we've configured **what we want to simulate** and **what infrastructure** we require, let's launch our simulation (F1) instances



```
$ firesim launchrunfarm
FireSim Manager. Docs: http://docs.firesim.com
Running: launchrunfarm
```

```
Waiting for instance boots: f1.16xlarges
Waiting for instance boots: m4.16xlarges
Waiting for instance boots: f1.2xlarges
i-0d6c29ac507139163 booted!
```

```
The full log of this run is: /home/centos/firesim-new/deploy/logs/2018-05-19--00-19-43-1
```



Status

Target Design:

- ✓ One Rocket Chip-based node (RTL)
 - 4 Rocket Cores
 - 16K L1 I\$, D\$
 - Block Device, UART, Serial Adapter
 - No NIC
- ✓ 4 MB LLC (RTL Model)
- ✓ DDR3 Memory System
- ✓ No network
- ✓ Boot vanilla buildroot-Linux distro

Host Resources:

- ✓ One manager instance (c4.4xlarge)
- ✓ One F1 instance (f1.2xlarge, single FPGA)

Distributing Simulation Infrastructure to the Run Farm



- Next, we need to setup our run farm instance with simulation infrastructure

```
$ firesim infrasetup
```

```
FireSim Manager. Docs: http://docs.firesim.com
```

```
Running: infrasetup
```

```
Building FPGA software driver for FireSimNoNIC-FireSimRocketChipQuadCoreConfig-FireSimDDR3FRFCFSLLC4MBConfig
```

```
[172.30.2.254] Executing task 'instance_liveness'
```

```
[172.30.2.254] Checking if host instance is up...
```

```
[172.30.2.254] Executing task 'infrasetup_node_wrapper'
```

```
[172.30.2.254] Copying FPGA simulation infrastructure for slot: 0.
```

```
[172.30.2.254] Installing AWS FPGA SDK on remote nodes. Upstream hash: 2fdf23ffad944cb94f98d09eed0f34c220c522fe
```

```
[172.30.2.254] Unloading EDMA Driver Kernel Module.
```

```
[172.30.2.254] Copying AWS FPGA EDMA driver to remote node.
```

```
[172.30.2.254] Clearing FPGA Slot 0.
```

```
[172.30.2.254] Checking for Cleared FPGA Slot 0.
```

```
[172.30.2.254] Flashing FPGA Slot: 0 with agfi: agfi-06b9b705ab9af1238.
```

```
[172.30.2.254] Checking for Flashed FPGA Slot: 0 with agfi: agfi-06b9b705ab9af1238.
```

```
[172.30.2.254] Loading EDMA Driver Kernel Module.
```

```
[172.30.2.254] Starting Vivado hw_server.
```

```
[172.30.2.254] Starting Vivado virtual JTAG.
```

```
The full log of this run is:
```

```
/home/centos/firesim/deploy/logs/2018-11-14--16-42-35-infrasetup-CHLESMNB83XFPZUR.log
```





Let's run a simulation!

```
$ firesim runworkload
FireSim Manager. Docs: http://docs.firesim.com
Running: runworkload

Creating the directory:
/home/centos/firesim/deploy/results-workload/2018-11-14-
-16-47-17-linux-uniform/
[172.30.2.254] Executing task 'instance_liveness'
[172.30.2.254] Checking if host instance is up...
[172.30.2.254] Executing task 'boot_switch_wrapper'
[172.30.2.254] Executing task 'boot_simulation_wrapper'
[172.30.2.254] Starting FPGA simulation for slot: 0.
```



Monitoring

```
FireSim Simulation Status @ 2018-11-14 16:48:28.240439
-----
This workload's output is located in:
/home/centos/firesim/deploy/results-workload/2018-11-14--16-48-24-linux-uniform/
This run's log is located in:
/home/centos/firesim/deploy/logs/2018-11-14--16-48-24-runworkload-GJ1IUXMSBGWPNGR3.log
This status will update every 10s.
-----
Instances
-----
Instance IP: 172.30.2.254 | Terminated: False
-----
Simulated Switches
-----
-----
Simulated Nodes/Jobs
-----
Instance IP: 172.30.2.254 | Job: linux-uniform0 | Sim running: True
-----
Summary
-----
1/1 instances are still running.
1/1 simulations are still running.
-----
[0] 0:mosh-client* "[mosh] centos@ip-172-" 08:48 14-Nov-18
```





Manually Interacting with simulations

```
$ ssh 172.30.2.254  
$ screen -r fsim0
```

```
Script started, file is uartlog  
AFI PCI Vendor ID: 0x1d0f, Device ID 0xf000  
Using edma queue: /dev/edma0_queue_0  
UART0 is here (stdin/stdout).  
random min: 0x0, random max: 0xffffffffffffffff  
Zeroing out FPGA DRAM. This will take a few seconds...  
Commencing simulation.  
[ 0.000000] OF: fdt: Ignoring memory range 0x80000000 - 0x80200000  
[ 0.000000] Linux version 4.15.0-rc6-31587-gcae6324ee357 (centos@ip-172-30-2-134.us-west-2.com  
pute.internal) (gcc version 7.2.0 (GCC)) #1 SMP Sat Nov 10 20:17:09 UTC 2018  
[ 0.000000] bootconsole [early0] enabled  
[ 0.000000] Zone ranges:  
[ 0.000000] DMA [mem 0x0000000080200000-0x00000000ffffffff]  
[ 0.000000] Normal [mem 0x0000000010000000-0x0000000047ffffffff]  
[ 0.000000] Movable zone start for each node  
[ 0.000000] Early memory node ranges  
[ 0.000000] node 0: [mem 0x0000000080200000-0x0000000047ffffffff]  
[ 0.000000] Initmem setup node 0 [mem 0x0000000080200000-0x0000000047ffffffff]  
[ 0.000000] elf_hwcap is 0x112d  
[ 0.000000] percpu: Embedded 14 pages/cpu @ (ptrval) s24664 r0 d32680 u57344  
[ 0.000000] Built 1 zonelists, mobility grouping on. Total pages: 4136455  
[ 0.000000] Kernel command line: root=/dev/generic-blkdev rw  
[ 0.000000] Dentry cache hash table entries: 2097152 (order: 12, 16777216 bytes)  
[ 0.000000] Inode-cache hash table entries: 1048576 (order: 11, 8388608 bytes)  
[ 0.000000] Sorting __ex_table...
```

```
FireSim Simulation Status @ 2018-11-14 16:48:28.240439
```

```
-----  
This workload's output is located in:  
/home/centos/firesim/deploy/results-workload/2018-11-14--16-48-24-linux-uniform/  
This run's log is located in:  
/home/centos/firesim/deploy/logs/2018-11-14--16-48-24-runworkload-GJ1IUXMSBGWPNGR3.log  
This status will update every 10s.  
-----
```

```
Instances
```

```
-----  
Instance IP: 172.30.2.254 | Terminated: False  
-----
```

```
Simulated Switches
```

```
-----  
Simulated Nodes/Jobs
```

```
-----  
Instance IP: 172.30.2.254 | Job: linux-uniform0 | Sim running: True  
-----
```

```
Summary
```

```
-----  
1/1 instances are still running.  
1/1 simulations are still running.  
-----
```

```
[0] 0:mosh-client*
```

```
"[mosh] centos@ip-172-" 08:48
```



```
Be [0] 0:mosh-client* "[mosh] [screen 0: bas" 08:54 14-Nov-18
```



Shutting down simulations/Capturing results

- Frequently, we want to capture results from simulations automatically
- FireSim's workload system supports this!
 - More on how to specify what to capture in a few mins
- But let's continue with our example – it's configured just to capture each simulated system's console output, once the simulation is powered off. If we do so, the manager will produce:

...

```
FireSim Simulation Exited Successfully. See results in:  
/home/centos/firesim/deploy/results-workload/2018-11-14--16-52-49-linux-  
uniform/
```

```
The full log of this run is:
```

```
/home/centos/firesim/deploy/logs/2018-11-14--16-52-49-runworkload-  
HSMYYMD4JBA6BHT5.log
```



Finally, get rid of our run farm EC2 instances

- Easy!

```
$ firesim terminaterunfarm
FireSim Manager. Docs: http://docs.fires.im
Running: terminaterunfarm
```

```
IMPORTANT!: This will terminate the following instances:
```

```
f1.16xlarges
```

```
[]
```

```
m4.16xlarges
```

```
[]
```

```
f1.2xlarges
```

```
['i-033959e7513fcf928']
```

```
Type yes, then press enter, to continue. Otherwise, the operation will be cancelled.
```

```
yes
```

```
Instances terminated. Please confirm in your AWS Management Console.
```

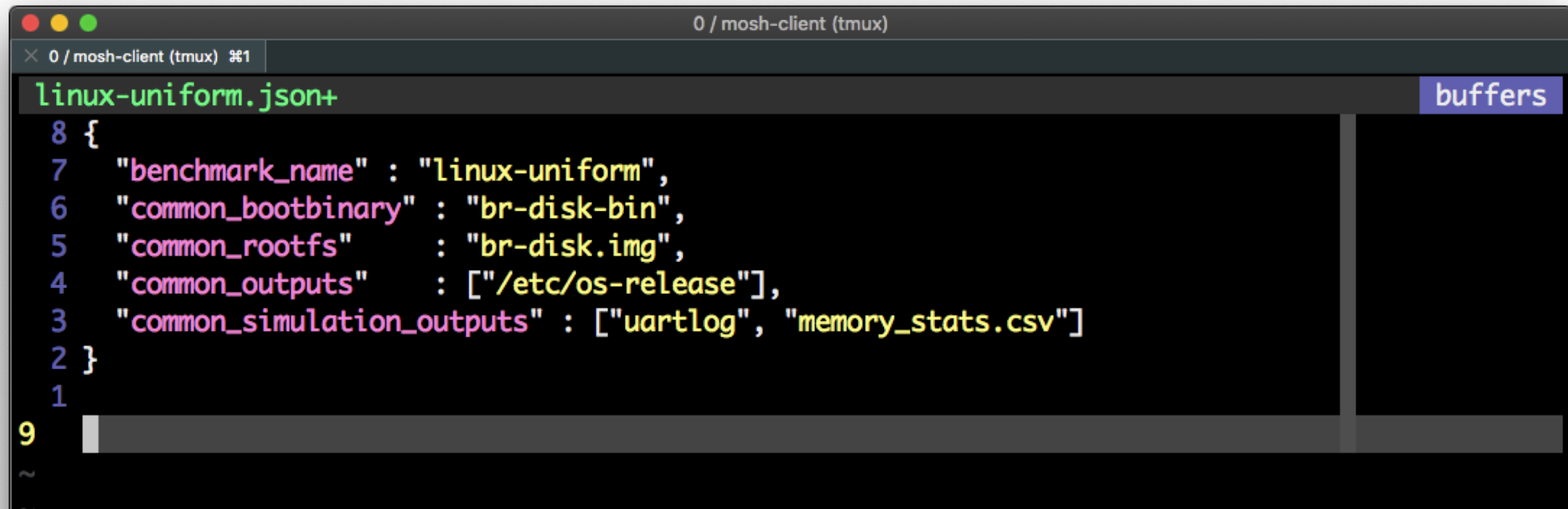
```
The full log of this run is:
```

```
/home/centos/firesim/deploy/logs/2018-11-14--17-00-01-terminaterunfarm-RGWY68L5ICAYQTA3.log
```



Custom Workloads: linux-uniform workload

- Previously, we relied on `linux-uniform.json` as our workload
- These jsons live in `firesim/deploy/workloads/`



```
0 / mosh-client (tmux)
0 / mosh-client (tmux) #1
linux-uniform.json+ buffers
8 {
7   "benchmark_name" : "linux-uniform",
6   "common_bootbinary" : "br-disk-bin",
5   "common_rootfs"      : "br-disk.img",
4   "common_outputs"    : ["/etc/os-release"],
3   "common_simulation_outputs" : ["uartlog", "memory_stats.csv"]
2 }
1
9
```



Let's look at a more interesting example: SPEC

- JSONs are used for two things:
 - Building rootfs/binary combos automatically, with benchmark infrastructure built-in
 - Deploying simulations with the manager (like we saw previously) and knowing which results to collect at the end

```
0 / mosh-client (tmux) #1
spec17-intrate.json buffers
1 [
2   "common_bootbinary" : "bbl-vmlinux",
3   "benchmark_name" : "spec17-intrate",
4   "deliver_dir" : "spec17-intrate",
5   "common_args" : ["--copies 4"],
6   "common_files" : ["intrate.sh"],
7   "common_outputs" : ["/output"],
8   "common_simulation_outputs" : ["uartlog"],
9   "workloads" : [
10    {
11      "name": "500.perlbench_r",
12      "files": ["500.perlbench_r"],
13      "command": "cd /spec17-intrate && ./intrate.sh 500.perlbench_r",
14      "simulation_outputs": [],
15      "outputs": []
16    },
17    {
18      "name": "502.gcc_r",
19      "files": ["502.gcc_r"],
20      "command": "cd /spec17-intrate && ./intrate.sh 502.gcc_r",
21      "simulation_outputs": [],
22      "outputs": []
23    },
24    {
25      "name": "505.mcf_r",
26      "files": ["505.mcf_r"],
27      "command": "cd /spec17-intrate && ./intrate.sh 505.mcf_r",
28      "simulation_outputs": [],
29      "outputs": []
30    }
31  ]
32 ]
```




Building/Deploying SPEC on Rocket/BOOM

- **Building:** We don't have enough time to go into detail here. Look at the `spec17-%` target in `firesim/deploy/workloads/Makefile`
 - At a high-level, you can just run `make spec17-intrate`, and you'll get 10 rootfs/linux image combos that will automatically run the spec benchmarks in parallel
- **Deploying:** Set your workload to `spec17-intrate.json` in `config_runtime.ini`, set the `# f1.2xlarges` to 10, select the hardware config you want to benchmark, then `firesim launchrunfarm/infrasetup/runworkload` as usual
- At the end, all your performance results live in one directory on the manager:
`firesim/deploy/results-workload/TIMESTAMP-spec17-intrate-HASH/`
- Instances get automatically terminated one-by-one as benchmarks complete – essentially zero cost to running in parallel on EC2, since you pay by the machine-second anyway



Summary

- Don't fret if you didn't catch everything, everything we showed you today is documented in excruciating detail at <http://docs.fires.im>
- We learned how to:
 - Build FireSim FPGA images for a set of targets
 - <http://docs.fires.im/en/latest/Building-a-FireSim-AFI.html>
 - Setup/Launch a simulation, including choosing targets, configuring target software, and managing EC2 infrastructure
 - <http://docs.fires.im/en/latest/Running-Simulations-Tutorial/Running-a-Single-Node-Simulation.html>
 - Customize workloads: SPEC example
 - <http://docs.fires.im/en/latest/Advanced-Usage/Workloads/Defining-Custom-Workloads.html>
 - <http://docs.fires.im/en/latest/Advanced-Usage/Workloads/SPEC-2017.html>