EESSI

EUROPEAN ENVIRONMENT FOR
SCIENTIFIC SOFTWARE INSTALLATIONS

EESSI meeting

November 5th 2020

https://github.com/EESSI/meetings/wiki

Agenda

1. Short introduction by new people
2. Testing of 2020.09 version of pilot repository [Alan, Bob, Terje]
3. Progress update + 2020.10 update of pilot repo [Axel, Bob, Kenneth, Peter, Terje]
4. Brainstorm meeting - build script [Caspar]
5. Brainstorm meeting - testing with ReFrame [Kenneth]
6. Meeting with Azure on potential sponsorship [Alan]
7. Meeting with AWS on getting free EC2 credits [Bob]
8. Licenses for EESSI GitHub repositories [Kenneth]
9. Next steps
10. Upcoming events
11. Q & A

Short introduction by new people

New people on the call: feel free to quickly introduce yourself!

- Who are you, where do you work, on what?
- Why are you interested in the EESSI project?
- Are you planning to actively contribute,

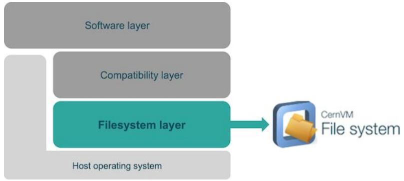
  and if so, to which aspect(s) of the project?

Testing of 2020.09 version of pilot repository

- Alan tested GROMACS on JSC clusters (Skylake and Zen2 a.k.a. AMD Rome)
  - Single node performance seems fine on both
  - Poor performance for multi-node runs due to incorrect OpenMPI/UCX configuration
- Bob & Jaco did some testing with ParaView GUI, ran into problems with fonts
- Terje gave a demo for the new VDI research platform at Univ. of Oslo
  - Very positive response, but even walking through the design, people think it’s magic.
- Bob gave a demo to colleagues running apps on various systems
  - GROMACS, R, ParaView
  - Ubuntu laptop, CentOS cluster node, Raspberry Pi, Windows/WSL (Ubuntu 20.04)
  - Very interested in mounting this on the University’s Linux workspaces

[Alan, Bob, Terje]

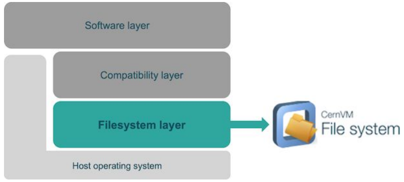
Progress update: filesystem layer



- Second Stratum-1 server @ Univ. of Oslo (by Axel Rosén)
- New release(s) of client packages to include additional Stratum-1
  - See https://github.com/EESSI/filesystem-layer/releases
- Improved CI for testing all playbooks on different Linux distros (see PR #40)
  - Still need to improve the CI for client packages to catch stupid errors
- TODO: set up proper yum/apt repositories for client packages (see issue #49)
- Bob & Kenneth attended monthly CernVM-FS coordination meeting (Oct 13th)
  - upcoming "`cvmfs_publish enter`" feature: very close to our fuse-overlayfs container
  - Discussed our issue with fuse-overlayfs and Singularity, but no solution yet...
  - CMS @ CERN has some automated workflow in CI for deploying software?
  - CernVM Workshop (1-3 Feb 2021) => submit talk on EESSI? (deadline Jan 12th 2021)

[Bob]

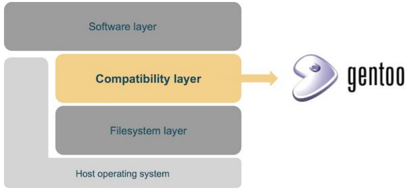
Additional Stratum-1 at Univ. of Oslo



- Why?
  - Curiosity and convincing local site admins of EESSI
- Where?
  - NREC (Norwegian Research and Education Cloud) - OpenStack based
- Resources
  - Centos 7.8 on Intel core (Haswell)
  - 8GB RAM + 2 VCPUs
  - Disk space of software stack: 58GB so far
- How?
  - Used Terraform to create VM
  - On another VM: clone filesystem-layer repo and run ansible-script for stratum1
- Setup time
  - Ansible-script took 88 minutes to run

[Axel]

Progress update: compatibility layer



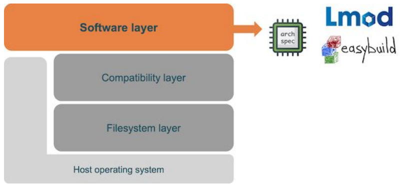
- Security updates applied in Gentoo Prefix
- Updated snapshot + bootstrap script for Gentoo Prefix (see PR #54)
- Docker container to bootstrap Gentoo Prefix (see PR #50)
  - Bootstrapping Prefix is now basically as easy as:

    ```
    singularity run docker://eessi/bootstrap-prefix:centos8-$(uname -m)
    ```

  - Snapshot to use can be controlled via environment variable
- Additional packages in 2020.10 set (see PRs #21 + #23 in `gentoo-overlay` repo)
  - Support libraries for Open MPI: `sys-fabric/rdma-core`, `sys-fabric/opa-psm2`
  - Fonts (for GUI apps like ParaView): `media-fonts/dejavu`, `media-fonts/liberation-fonts`
  - Useful tools: `app-editors/emacs`, `app-editors/vim`, `dev-util/strace`

[Peter]

Progress update: software layer



- Build script was updated (see PRs #28 + #32)
  - Use EasyBuild v4.3.1 (and clean up stuff pulled from PRs)
  - Also install TensorFlow 2.3.1 (works on `x86_64` and `aarch64`)
- Custom easyconfig file for fontconfig (see PR #31)
  - To make sure fonts are picked up from compatibility layer
  - Need to figure out a clean way of pushing this into EasyBuild?
- Enhanced init script to allow specifying which software subdir to use (see PR #27)
  - Via `$EESSI_SOFTWARE_SUBDIR_OVERRIDE`
  - Required for Graviton2 (not recognized yet in latest archspec release)
  - Useful for benchmarking impact of using installations for older CPU arch
- Build script is still an ugly/hackisk bash script, for now…
  - Working on new EasyBuild feature to clean this up (more info coming up)

[Kenneth]

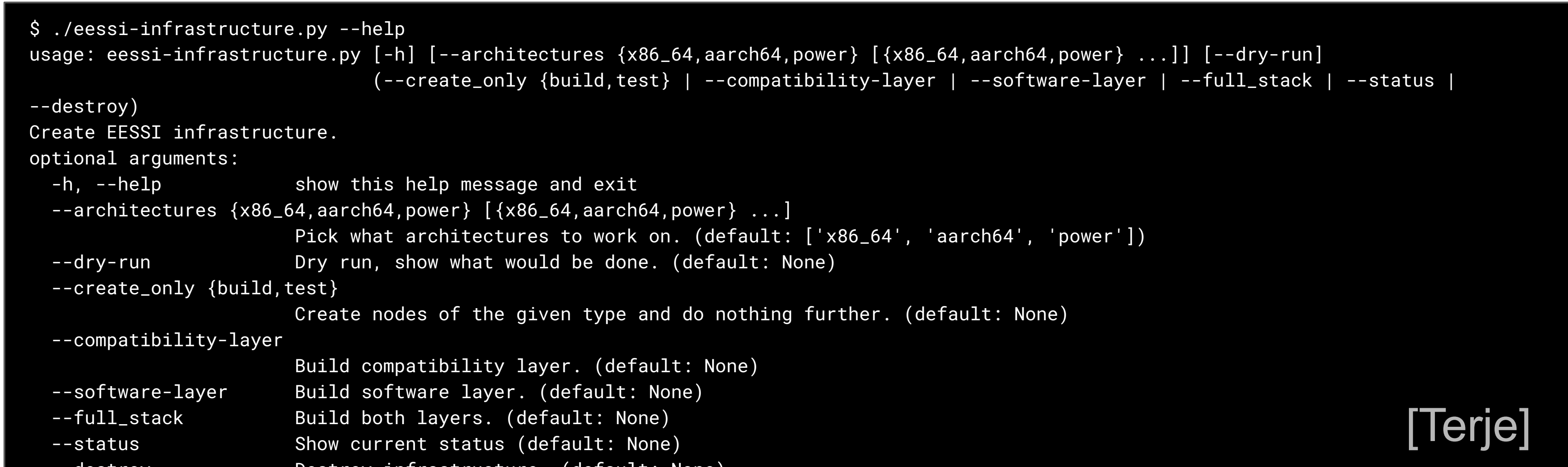
Progress update: automating deployment

- See https://github.com/terjekv/eessi-compat-builder
- Current focus is on automating build compatibility layer
- Goal: Automate building of both compat layer and generic software layer
- Combining Terraform + Ansible
- Also allow for generation of standardized testing and build nodes
- Hosts run on AWS, but Terraform supports “anything” with an API
  - We can also use multiple providers in the same setup, increasing access to hardware
  - Easily portable
  - Easily modified on-the-fly
- One Python command line interface for all the tooling!
- No need to manually use or learn Terraform or Ansible!

[Terje]

Automating deployment - workflow

- Terraform creates hosts
- Ansible pushes jobs to hosts (in parallel across architectures)
- Results are fetched and tarballs created
- Terraform destroys hosts



```
$ ./eessi-infrastructure.py --help
usage: eessi-infrastructure.py [-h] [--architectures {x86_64,aarch64,power} [{x86_64,aarch64,power} ...]] [--dry-run]
                               (--create_only {build,test} | --compatibility-layer | --software-layer | --full_stack | --status |
--destroy)
Create EESSI infrastructure.
optional arguments:
  -h, --help            show this help message and exit
  --architectures {x86_64,aarch64,power} [{x86_64,aarch64,power} ...]
                        Pick what architectures to work on. (default: ['x86_64', 'aarch64', 'power'])
  --dry-run             Dry run, show what would be done. (default: None)
  --create_only {build,test}
                        Create nodes of the given type and do nothing further. (default: None)
  --compatibility-layer
                        Build compatibility layer. (default: None)
  --software-layer      Build software layer. (default: None)
  --full_stack          Build both layers. (default: None)
  --status              Show current status (default: None)
  --destroy             Destroy infrastructure. (default: None)
```

[Terje]

EESSI pilot repository

NOT FOR PRODUCTION USE!

2020.10 version of pilot software stack

- Software:
  - `foss/2020a` toolchain (GCC 9.3, OpenMPI 4.0.3, OpenBLAS 0.3.9, FFTW 3.3.8)
  - GROMACS 2020.1
  - OpenFOAM (two variants: version 8 and v2006)
  - R 4.0.0 (incl. ~800 R packages as extensions)
  - Bioconductor 3.11 (bundle of 262 R packages)
  - **TensorFlow 2.3.1** *(new!)*
- Targets:
  - `aarch64/generic` *(note: only TensorFlow for now...)*
  - `x86_64/generic`
  - `x86_64/intel/{haswell,skylake_avx512}`
  - `x86_64/amd/zen2`

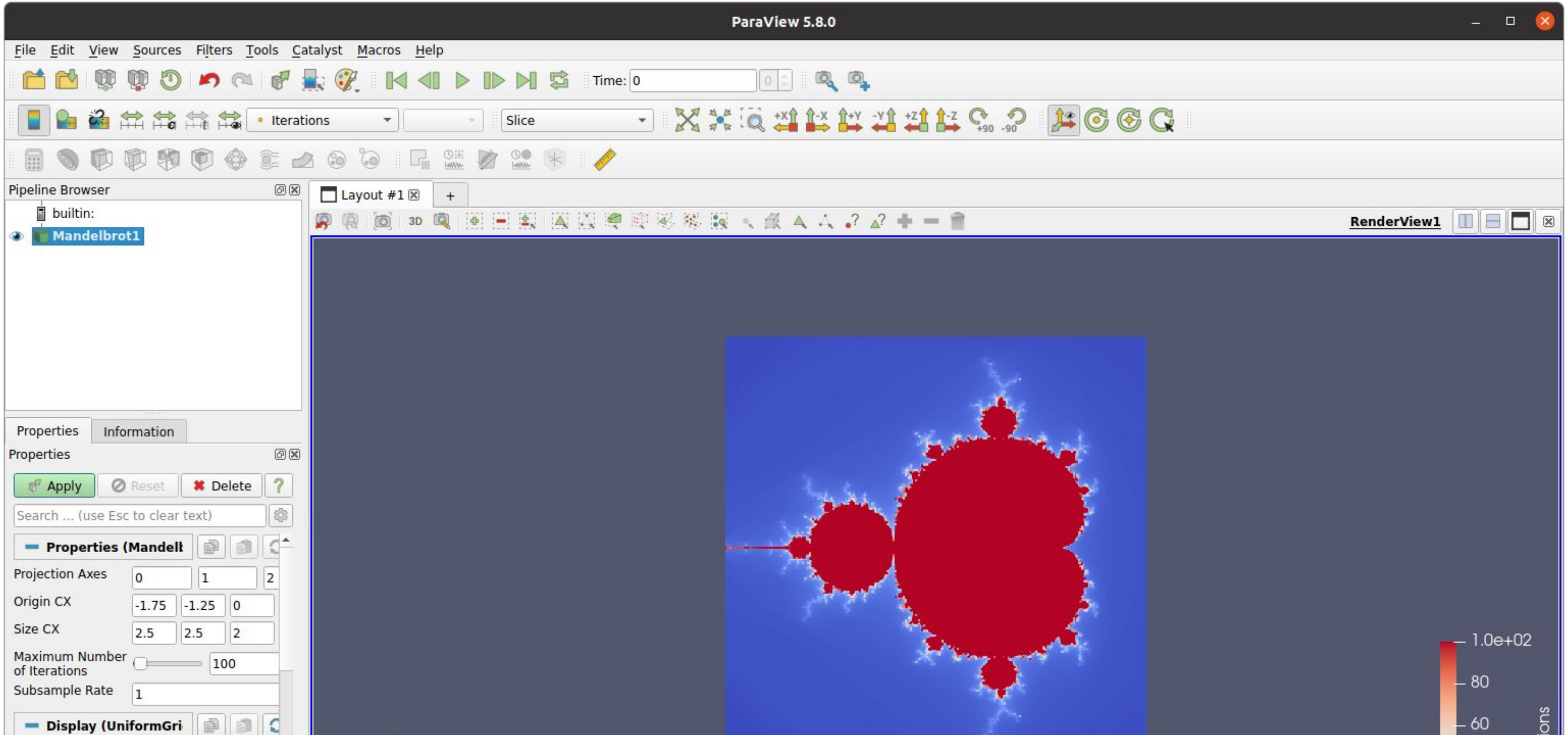
Updated documentation at https://eessi.github.io/docs/pilot

[Kenneth]

EESSI pilot repository

NOT FOR PRODUCTION USE!

- Font issues with ParaView fixed in 2020.10 version of pilot repo



Time: 0

Projection Axes	0	1	2
Origin CX	-1.75	-1.25	0
Size CX	2.5	2.5	2
Maximum Number of Iterations	100		
Subsample Rate	1		

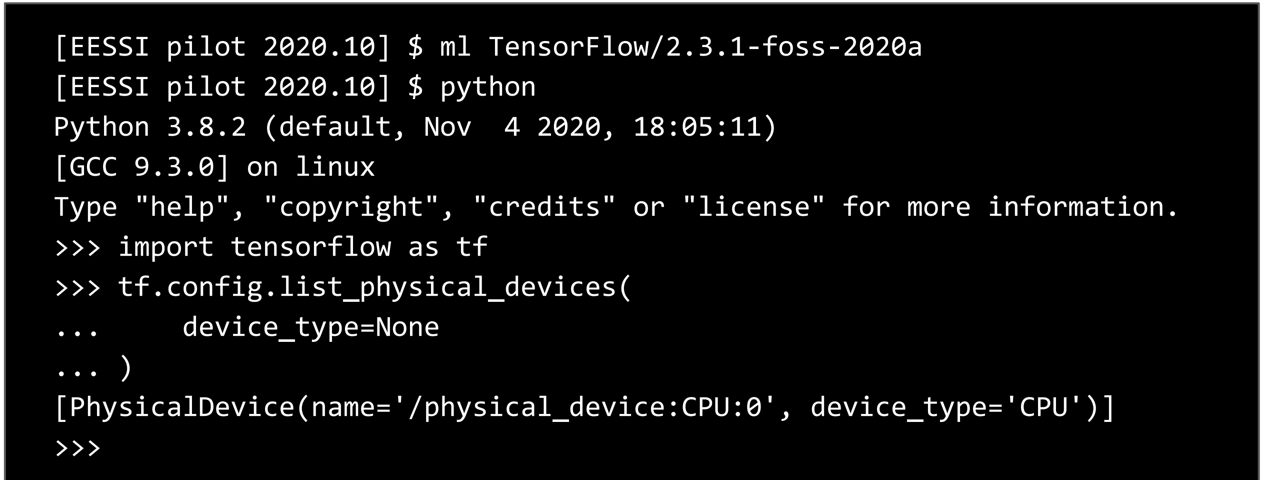
[Bob]

EESSI pilot repository

NOT FOR PRODUCTION USE!

- TensorFlow works on Raspberry Pi via `aarch64/generic` installation!
- TensorFlow 2.3.1 built on Graviton2 instance in AWS
- Using build script in “generic” mode:

```
./EESSI-pilot-install-software.sh --generic
```



```
[EESSI pilot 2020.10] $ ml TensorFlow/2.3.1-foss-2020a
[EESSI pilot 2020.10] $ python
Python 3.8.2 (default, Nov  4 2020, 18:05:11)
[GCC 9.3.0] on linux
Type "help", "copyright", "credits" or "license" for more information.
>>> import tensorflow as tf
>>> tf.config.list_physical_devices(
...     device_type=None
... )
[PhysicalDevice(name='/physical_device:CPU:0', device_type='CPU')]
>>>
```

[Bob]

Brainstorm meeting - Build script

- Previous: hardcoded build script used for 2020.09
- generic build script in software-layer/pull/5 but all outside EasyBuild
- New goal: add EasyBuild functionality to build from a *build specification* file
- Discussed desired format and capabilities
- See:
  - Issue describing feature: https://github.com/easybuilders/easybuild-framework/issues/3468
  - WIP pull request (work done by INUITS consultancy company, funded by HPC-UGent) https://github.com/easybuilders/easybuild-framework/pull/3479
  - Meeting notes: https://github.com/EESSI/software-layer/wiki/Brainstorm-meeting-(Oct-9th-2020)

[Caspar]

Brainstorm meeting - Build script

Example:

eb eessi-2020.10.yml --labels='gpu'

With *eessi-2020.10.yml*:

```
easybuild_version: 4.3.0
robot: True
software:
  EasyBuild:
    toolchains:
      SYSTEM:
        versions: [4.3.1]
  OpenFOAM:
    toolchains:
      foss-2020a:
        versions: [8, v2006]
  ...
```

```
...
GROMACS:
    toolchains:
      foss-2020a:
        exclude-labels: system:gpu
        versions:
          2020.1:
          2020.3:
            from_pr: 1234
      fosscuda-2020a:
        include-labels: system:gpu
        versions: [2020.1]
  R:
    toolchains:
      foss-2020a:
          versions: [4.0.0]

  R-bundle-Bioconductor:
    toolchains:
      foss-2020a:
        versions:
          3.11:
            versionsuffix: '-R-4.0.0'
            exclude-labels: arch:aarch64
```

[Caspar]

Brainstorm meeting - testing with ReFrame

- Meeting on Fri Oct 23th 2020 to discuss testing in software layer (& beyond)
- Joined by Kenneth, Caspar, Terje, Alan, Victor (CSCS)
- 'tests' subdirectories in compatibility layer + software layer
- ReFrame tests implemented as Python classes
- Try to avoid hardcoding site-specific things in the tests
    - Some developments being done in ReFrame to make that easier
- Look into easy tests to kickstart things
    - Check output of ucx_info and ompi_info for support for fast interconnects
    - MPI hello world
- Later also actual application tests
    - GROMACS, OpenFOAM, TensorFlow, …
- Reference numbers for performance vs differences between sites?

[Kenneth]

Meeting with Azure on potential sponsorship

- 2-hour meeting on Oct 8th 2020 [Alan, Kenneth, Davide]
  - Plenty of tech participants from Microsoft Azure
- Introductory presentation on EESSI given by Kenneth
  - Was recorded and shared internally by Microsoft
- Significant potential interest
  - Options: Azure hours, Microsoft staff helping out, Microsoft ECIF funding, EU funding calls…
  - Looks like a connection to an ISV or customer will get us the most return

    (e.g. HPC Now!, working on a joint customer project) as that would mean person effort
- Internal project proposal in MS Azure to motivate sponsorship
- Another meeting near the end of the month

[Alan]

Meeting with AWS on sponsored EC2 credits

- Quick informal meeting on Wed Nov 4th (Bob, Kenneth, Terje)
- Discussion of more structured approach to providing EC2 credits for EESSI
  - Until now limited credits were provided to Kenneth + Terje to play with Arm Graviton2 instances
  - Plan is to set up an “EESSI” account in AWS that holds sponsored credits
  - Who gets access to (and for what) is under our control
  - Account would get created via Univ. of Groningen (Bob)
- Goals:
  - Building software for Arm64 (`aarch64/graviton2` + `aarch64/generic`)
  - Testing on Arm64 instances (manual, CI, etc.)
  - Work on automating the workflow via Terraform + Ansible (see Terje’s work)

[Bob]

Licenses for EESSI GitHub repositories

- We should put the necessary licenses in place in each EESSI repository
  - LICENSE file + update of README file with “License” section
- General Public License version 2 (GPLv2) makes most sense
  - Both EasyBuild and Gentoo (Prefix) are licensed under GPLv2
    - So stuff we implement on top should be compatible with GPLv2
  - Other projects have more permissive licenses
    - `galaxyproject/ansible-cvmfs` + Lmod => MIT
    - CernVM-FS + ReFrame => BSD-3
    - `archspec` => Apache v2 or MIT (dual license)
- Objections?

[Kenneth]

Next steps

- Build script leveraging support for “easystacks” in EasyBuild
- Testing & continuous integration
  - More & better CI checks for EESSI layers
  - Tests for software layer via ReFrame (smoke tests, apps, benchmarks, …)
- Documentation
  - How to “set up” a build nodes
  - Benchmark results?
- Automatic deployment of software to CernVM-FS Stratum-0
  - Triggered by pull requests in GitHub?
  - Leveraging both cloud resources in Azure/AWS + resources at HPC sites
- EESSI pilot stack on POWER?
  - Resources can be requested at OSU Open Source Lab (https://osuosl.org)

Upcoming events

- Terje’s talk on EasyBuild on ARM (+ EESSI pitch at the end)
  - ARM HPC Users Group @ Supercomputing’20
  - Video is available already on AHUG YouTube channel
- Kenneth’s “software demo” talk at SORCE
  - International Series of Online Research Software Events
  - Planned for Wed Nov 25th 2020 at 5pm CET
- EasyBuild User Meeting (virtual): most likely week of 25-29 Jan 2021
  - Perhaps including a half day session on EESSI?
- CernVM-FS Workshop: 1-3 Feb 2021
- FOSDEM’21 (virtual): 6-7 Feb 2021
  - Hopefully including an HPC devroom… (to be confirmed soon)