SLATE Core Team
Winter 2019 Meeting

@ University of Michigan
Jan, 2019

Logistics:

- Times: Jan 21 - Jan 23
- Indico: https://indico.cern.ch/event/788214/
- Restaurant selected: Grizzly Peak, reservation for 12 Tue night, at 6 PM.
    - https://www.grizzlypeak.net/
- There will be no remote sessions.
- Where: Ann Arbor, MI - 3 sites, 9am-5:30pm each day
    - **West Hall 340** for Monday, January 21
    - **West Hall 348** for Tuesday, January 22
    - **West Hall 348** (ATLAS video-conf room) for Wednesday, January 23rd
- Travel info
    - Michigan Flyer may be an inexpensive option to/from the DTW airport: https://www.michiganflyer.com/
- Parking and map
    - https://ltp.umich.edu/parking/
    - https://drive.google.com/a/umich.edu/open?id=10yJRB7ukJ5WmYNs-w8KoQSPH3tIV7jvx&usp=sharing
- Hotels or Air B-n-B
    - https://campusinfo.umich.edu/article/accommodations
    - Google Hotels
    - https://www.airbnb.com/s/Michigan-Avenue-Ann-Arbor--MI

References:

- Meeting Google Drive folder is**:**

    - https://drive.google.com/drive/folders/1F5lAh4KFQV76-GLXmxs4n8DZpRSgjfNZ?ogsrc=32
- Meeting notes
    - https://docs.google.com/document/d/1M5dPgTH6Vu8bs-tnD_rILba0Tv3zrx9P282cL1Qml2w/edit?usp=sharing
- Meeting Zoom
    - https://utah.zoom.us/j/9651366779
- **SLATE** website: http://slateci.io
- **SLATE** Jira: https://slateci.atlassian.net/
- **SLATE** Architecture: http://bit.ly/slateci-architecture
- **SLATE** People (profiles): Calling Card (please fill out or update)

Key **SLATE** terms: Platform, Edge Cluster, Administrator, Application Administrator, Application Developer

What Success Looks Like

"Building and Supporting a Production Platform"

- Build a functioning SLATE edge cluster (single machine) at Utah supporting centralized DevOps
    - Document (update web docs) a solid build process for a remotely managed k8s edge cluster
    - Document (update web docs) for the required federation steps for a locally managed k8s edge cluster
    - Document (update web docs) for the required federation steps for an existing k8s federated infrastructure such as PRP (single cluster across region), CERN (multiple clusters single site).
    - Document remaining steps before US ATLAS deployment in 4 weeks
- Define clear roadmap and workplan for metrics, logging, display for both applications and systems
    - Demonstrate existing capabilities for both
    - Clear plan for perfSonar mesh of production SLATE sites
    - Comparison with PRP monitoring
- Document (update web docs) for application workflow; settle/resolve all application catalog issues
- Document (update web docs) application deployment tutorial
    - Unify packaged web docs from Gage with slateci.io docs
- Document (update web docs) for cluster admins
    - guidance on storage configuration
- Document (update web docs) Mini-SLATE as a tool assisting development workflow

- Demonstrate DevOps Capabilities
    - Use XCache for ATLAS as primary driver
    - Deploy from scratch
    - Validate
    - Shutdown, cleanup
    - Repeat
    - How quickly can this be done over a three-site federation?
- Review state of existing application catalog
    - Image, chart management
    - How to work with our OSG partners who will be providing images for OSG support software (which soon will include XCache)
    - Focus on services
        - Globus Connect
        - HTCondor on the edge
        - OSG Services (to come, working with OSG)
            - StashCache (a variant of XCache)
            - HTCondor CE
            - Frontier Squid
            - HTCondor submit node (OSG Connect-like)
- Review and collect feedback state of the SLATE user portal, CLI, api-server
    - User workflows - for an app developer
    - User workflows - for an edge cluster operator
- Deployment plan for NMSU and Clemson
- Define R&D and engagement plan with CERN
- Summarize
    - State of the production SLATE infrastructure
    - Readiness to support VOs with containerized applications
        - ATLAS as pathbreaker (UM, UC, OU, UTA, BU, BNL)
            - Need updated hardware recommendation for sites
            - Readiness for ATLAS Sites Jamboree at CERN first week of March
    - Readiness to support new sites
        - A new campus
        - FIONA8 target
    - Readiness to support new federations
        - Integration with PRP, NRP
        - Integration with CERN
    - SLATE project management
        - Github issues versus Jira issues
        - Weekly reporting on activities

PREREQUISITES:

- Creation of a Bootable USB image for workshop
- Utah hardware setup in fully remote managed state with Bootable USB image installed

MEETING SCHEDULE

Day 1 - Mon Jan 21

Day 1 9:00am-9:30am Intro and Workshop logistics (and Breakfast)

- Introduction (Rob)
- Workshop logistics (Shawn)
- ATLAS Remote Managed Build Process:
    - https://docs.google.com/document/d/1c7Lmmp_-vsrCKg2Ggie6WWfjcaO-oDw4lOIqFRNXalE/edit#heading=h.x803ifi2xymt

Day 1 9:30am Build Utah machine from raw hardware (Lincoln/Ben facilitate)

- https://docs.google.com/document/d/1c7Lmmp_-vsrCKg2Ggie6WWfjcaO-oDw4lOIqFRNXalE/edit#heading=h.x803ifi2xymt
- Boot machine from USB image
- Install software using Puppet
- Provision network (Calico, metallb)
- Manual registration steps
- Validate basic Kubernetes and SLATE client functionality

[DECISION] << >> BOSS card on every site

<< >> document internal SLATE side BIOS commands

<< >> Turn build process into Google Form with help

<< >> iDRAC settings backed up and restored? Can PXE?

<< >> Put together slides that describe process

Can provide different services potentially

<< >> Put together a SLA for remote managed

- Describe process and platform
- Describe services
- Can provide localized versions if desired
- Ongoing support past grant?

Day 1 12:30pm-1:30pm: Lunch

Day 1 1:30pm: ATLAS Storage (Bob)

- ATLAS project defines disk requirements.
- How do we layout disks on ATLAS nodes?
- How do we provision and advertise the storage consistently for xcache and other SLATE apps?
    - ATLAS XCache is relying on manual partition in current packaging
    - Does this storage packaging allow for other SLATE or ATLAS apps seamlessly?

Day 1 2:30pm: ATLAS Node level Monitoring (Gabriele/Bob)

(Note: Node level Monitoring work will also be happening in parallel.)

- Monitoring doc:
    - https://docs.google.com/document/d/1lMji2dwLPkHPgtgYk5U5f0H500kOLeQswAd9SbBl1RU/edit#heading=h.n9pkgznjiiyp
- Goal for planning session: come up with the general deployment strategy that will be implemented over the next month and record it in JIRA
    - Metrics
    - Logs
- Goal for working session: Prometheus/Grafana installed at least on one cluster standalone
- Items to discuss:
    - Deployment strategy for metrics monitoring (standard helm charts? create our own?)
    - Deployment strategy for logs
    - Outside access (ingress, dns name, …)
    - Location for central services (i.e. Grafana and Prometheus aggregator)
    - Data aggregations from the clusters (Push Gateway, prometheus/thanos/cortex?)
    - Cluster level permission (what requires admin privileges and what doesn't?)

- Metrics (what do we want to monitor from each node?)
- User level permissions (i.e. logins and permissions for Grafana)
- Notification of hardware failures:
  - Notify local SLATE / organization -- request local site to replace

Day 1 3:30pm-3:45pm: Break

Day 1 3:45pm- ATLAS Node level Monitoring (Gabriele/Bob)

Day 1 Network Monitoring

- Relying on existing ATLAS perfSONAR nodes
- Validation and feedback to the ATLAS SLATE nodes?

Day 1 5:15pm - Wrap-up

Day 1 5:30pm - Clean-up for day 1

Day 2 - Tue Jan 22

(NOTE: Room change from 3246 Randall to 348 West Hall)

Day 2 9:00am- ATLAS node building recap

Gaps noted and documented? Do our web docs provide the right information for a new site?

- we have a working node at Utah now
- how to generalize the provisioning beyond Dell systems? E.g. HP.
- Ingredients of the **Slate Provisioner** (need a diagram)
  - Registration service (**web app**)
    - collect hardware access information
      - jump box or VPN and credentials
      - IPMI card and credentials
    - information gets stored (**database**)
  - USB - iPXE pre--boot environment with the slate provisioner URL
    - **slate environment script**

            - bash script that executes system utilities to configure the BIOS and collect hardware information such as storage
        - two options - pre-boot info gathering, install
    - **Web app** which prompts system admin for standardized labels, e.g. storage
        - drop downs, standard
    -
    - **Web app** (e.g.) to dynamically create customized Kickstart files, site specific (similar to cobbler)
        - automate labeling of hardware components where possible
    - web server - serves configurations
    -
-

Day 2 9:30am- SLATE client / Web portal (Chris)

- Workflow for new site that deploys Kubernetes and wants to federate with SLATE
- Workflow for raw SLATE deployment
- Anything specific to ATLAS deployment?

Day 2 xcached deployment (Ilija) / Provisioning break out / Monitoring break out

Day 2 ATLAS Security model

- Are there specific ATLAS security concerns or characteristics?

Day 2 12:30pm-1:30pm: Lunch

Day 2 1:30pm- Mini-SLATE dev workflow (Ben/Jason)

Walk through of all in running min-slate

- What mini-slate dev provides?
- How to make use of mini-slate?
- How can we improve mini-slate?
-

Application developer workflow

- How to take an app and see if works using mini-slate?

    - Local helm charts on min-slate? Added last week
    - Need documentation
    - Where to document in order to be fully polished ?
- How to use in a development workflow?
- Command to test local repository?

Day 2 3:30pm-3:45pm: Break

Day 2 How to deal with charts and sources for SLATE for Domain Science Docker images and SLATE helm charts (Chris/Gabriele)

- From our security model - do we want to be responsible for images?
- Trust the major distro
- Make the claims of secure -- possibly automate the build
- Image definitions in SLATE repo and chart in charts directory
- From early - separate
- Connection to Docker hub is broken at the moment
- Many companies have private repos
- Architecture: decision
- OSG has concern on trust chain
- Who builds the app?
- How to make sure the app is legit?
- How to make very supported / standard -- 3rd party Docker repo ?
- Concern of running our own registry or could build or push sources

Day 2 3:45pm -

Day 2 Define R&D and engagement plan with CERN

Day 2 5:15pm - Wrap-up

Day 2 5:30pm - Clean-up for day 2

Day 2 6:00pm- Dinner

Day 3 - Wed Jan 23

SLATE PROJECT TIMELINE

- https://docs.google.com/spreadsheets/d/1xoD4IcA1B9lScUwJeUpuUoU13uvBzkvZPDkz3xJG5-A/edit?usp=sharing

IMPORTANT DATES

- ATLAS Sites Jamboree, March 5-7
    - Functional SLATE platform - indicating resource providers
- SLATE release "beta" Milestone
    - April 30
    - Core
        - Rename VOs to Groups
        - Add accounting properties for Groups (science domain) and Clusters (site, owner)
        - Catalog restructuring
    - Monitoring
        - https://github.com/orgs/slateci/projects/2
          Basic system monitoring infrastructure distributed deployment strategy for Prometheus/Grafana/Thanos.
        - **Installation integration with the SLATE service** install and configure Prometheus during cluster registration (both on full system and mini-SLATE). User perspective: add a cluster and see it represented in the system level monitoring
        - **Monitoring screens for site/platform providers** (better than out of the box): create screens that give platform level and cluster level view. Include map view for cluster, drill down platform/cluster/node/pod.
        - **Monitoring screens for application developers & administrators** create screens that give application developers and administrators an application level view. Include drill down platform/cluster/vo/application/application instance.
    - Provisioner
        - Existing tools cleaned up
            - Old USB image will be modified to serve static "Install SLATE Edge Node" and "Install PerfSONAR" configurations
            - Requires DHCP, untagged, outbound HTTP access
        - Mock-ups of provisioning portal interface complete
        - Alpha specification of provisioner REST API
        - Prototype of provisioning server ready
            - Can serve a dynamically generated iPXE configuration and Kickstart configuration

        - Prototype of pre-boot environment ready
        - https://github.com/orgs/slateci/projects/4
    - Applications
        - Application development tutorial (full workflow) featuring MiniSLATE
        - Unified documentation
        - PerfSONAR
        - Globus Connect
        - HTCondor-k8s (for science application backfill)
        - OSG services:
            - XCache
            - StashCache
            - HTCondorCE
            - frontier-squid
            - submit host
    - Web portal/Interfaces
        - Meets Provisioner-beta requirements
        - Site registration
        - Supports SLATE-Lite and production SLATE
        - Integration with monitoring
    - Logging
        - nginx sidecar
        - elasticsearch sidecar
    - Community Outreach
        - Announce to slateci-announce@
        - project community update
        - invite participation (Slack)
- NMSU and Clemson deployment
    - requires SLATE beta
    - requires hardware purchase
        - need to know which network device to plug into
        - need to spec the hardware
    - physical installation w/ site admins
    - who are the contacts?
        - Matt at NMSU
        - ?? at Clemson
    - Which applications?
        - JupyterLab
        - containerized CE
        - OSG submit node (slate certificate)
        - stashcache (registered)
    - Small node?
- GPN
    - Great Plains Network community (May 21-23)
        - Note - here is a link to the CENIC PRP tutorial: link
- NSF DIBBS Meeting (TBD)
- SLATE release 1.0 "for PEARC19 release"
    - July 1

    - Monitoring, provisioning, application deployment, logging service
- PEARC19 Tutorial proposal deadline, Feb 20
    - Resource providers
    - Application developers
- PEARC19
    - July 28 - August 1 (Chicago)
    - https://www.pearc19.pearc.org/
- NRP
    - TBD
- SLATE release 1.1 "for SC19 release"
    - Oct 1
    - Features: TBD
- SC19
    - https://sc19.supercomputing.org/important-deadlines/

OUTREACH
- Community Outreach Communication
    - Reach out to our AZ/Cyverse, SGCI
    - Campus Champions announcement
    - CaRCC - systems track

Day 3 9:00am -- ATLAS node building recap

- What are the gaps for deployment in 8 weeks?
- **The ATLAS Sites Jamboree is March 5-7**
    - Need recommendation to UTA, BU, OU, BNL for slate node purchases
-

Day 3 Shuffling SLATE application catalog in github (Chris)

- Location of standard helm repository: https://github.com/helm/charts
- Automation of some of the packaging?
    - Jenkins would be able to notice and do some of the repackaging automatically
- Maintaining different versions in the repository
- https://github.com/helm/monocular Helm chart discovery UI
-

Day 3 Organizing github repos (Chris)

- How to accomplish cleanly?

How to deal with charts and sources for SLATE for Domain Science Docker images and SLATE helm charts (Chris/Gabriele)

- From our security model - do we want to be responsible for images?
- Trust the major distro
- Make the claims of secure -- possibly automate the build
- Image definitions in SLATE repo and chart in charts directory
- From early - separate
- Connection to Docker hub is broken at the moment
- Many companies have private repos
- Architecture: decision
- OSG has concern on trust chain
- Who builds the app?
- How to make sure the app is legit?
- How to make very supported / standard -- 3rd party Docker repo ?
- Concern of running our own registry or could build or push sources

Day 3 Define R&D and engagement plan with CERN (Bob, Rob, Shawn, …)

Day 3 12:30pm-1:30pm: Lunch

Day 3 SLATE Deployment Standards (Lincoln/Ben)

- Normalize naming scheme of SLATE and document in SLATE Site inventory
    - https://docs.google.com/spreadsheets/d/1hptxa-7Qow0j-xhNohlG0j6RbAD1kD1qaWvYllxF-zs/edit#gid=388455458
    - Necessary for optimum Puppet
- Puppet manifest on various infrastructures
    - How to be consistent across different infrastructure at different sites?
    - How to deliver consistent Puppet manifest to different entities
        - Central deployment?
        - Download?
- Drive formatting
- Software updates (Chris)
    - K8s, Helm in particular
    - How/when to update (schedule)?
    - Central vs what is done on the clusters (esp those we don't control)
    - Compatibility of versions

Day 3 3:30pm-3:45pm: Break

Day 3 Scaling / HA SLATE (etcd and general discussion) (Ben)

- Deployment plan for NMSU and Clemson (see above)
- Single master vs two for redundancy
- In future, multiple etcd for failover
- Recovery from hard crash
- Chris' development to bring a cluster back w/ apps, etc.
- Post 1.0 release

Day 3 5:00pm - Wrap-up -- Did we hit our success targets for the workshop?
YES!

Day 3 5:30pm - Clean-up for day 3

MEETING Topics (Brainstorming list - Not prioritized as of 12/21)

- End to end guide - acquisition of hardware to fully functioning SLATE site
  - https://docs.google.com/document/d/1c7Lmmp_-vsrCKg2Ggie6WWfjcaO-oDw4lOIqFRNXalE/edit#heading=h.x803ifi2xymt
  - Description of hardware - probably in ATLAS scope (Lincoln, Rob and Shawn discussing)

            - Lots of spindles important. SSD/NVMe use-case needs testing
        - Remote Installation on Utah cluster
        - Registration of new cluster
        - Create accomplish management --
            - Puppet
                - Kubeadm or Kubernetes module
                - Image
        - System level Monitoring/logging
            - Logs
            - Basic metrics
            - perfSONAR -
                - already have 2 instances at each Tier 2 and Tier 1
                    - Will utilize these instances
                        - Integrate our scheduled tests
                    - Will need some end to end tests as we bring up basic box
                - NOT application logging
        - Storage (what is our target, what do we do near-term to advertise, configure and utilize storage)
            - Near term - manual partition
                - What does xcache need - size and performance
                    - Ilija has package ready
                    - ATLAS version of xcache
            - Long term
        - Network
            - Load balancing - metallb
            - Network - calico
            - Ingress controller -- build document of ingress for example
                -
        - Supporting xcache in ATLAS
            -
    - MiniSLATE
        - What it does
        - How to make use of it
        - How can we improve it?
        - Do same build process on mini-slate with group
        -
    - Monitoring

        - Metric (prometheus/graphana) - Gabriele, Bob
            - Goal for planning session: come up with the general deployment strategy that will be implemented over the next few months and record it in JIRA
            - Goal for working session: prometheus/graphana installed at least on one cluster standalone
            - Items to discuss:
                - Deployment strategy (standard helm charts? create our own?)
                - Outside access (ingress, dns name, …)
                - Location for central services (i.e. graphana and prometheus aggregator)
                - Data aggregations from the clusters (push gateway, prometheus/thanos/cortex?)
                - Cluster level permission (what requires admin privileges and what doesn't?)
                - Metrics (what do we want to monitor?)
                - User level permissions (i.e. logins and permissions for graphana)
    - Log file collection
    - Application workflow
    - Scaling SLATE (etcd and general discussion)
    - Onboarding New Mexico, Clemson, ATLAS, OSG,
        - What do we expect of them?
            - Hardware
            - Personnel time
        - What do we expect of SLATE team?
    -
    - Supporting stashcache and hosted CE in OSG
    -
    - Security
    - Software versioning - how to be consistent
    - Networking
        - Network cleanliness between sites
        - load balancers - metallb, IP, host port
        - Getting switches in place and functioning
        - Do we investigate Cillium instead of Calico due to potential performance? → DECISION: stick with Calico for now

            - Benchmark: https://itnext.io/benchmark-results-of-kubernetes-network-plugins-cni-over-10gbit-s-network-36475925a560
            - BK: Calico is more performant when under heavy load. Cilium does have better network rule capabilities and more integrations. Cilium also **requires** a recent kernel to be effective.
    - Discussion on portal interaction with users
        - Workflow for new site that deploys Kubernetes and wants to federate with SLATE
        - Workflow for raw SLATE deployment
    -