Written Assignment

Harpinder Jot Singh

2017A7PS0057P

Twine - FB Tupperware

- A unified cluster management system for shared infrastructure managing all services (web, ads, cache, search, news feed, etc) and machines
- Design Decisions
    - dynamic machine partitioning **over** static clusters
        - improves efficiency and ensures fleet wide optimization
    - Customization in shared infrastructure **over** private pools
        - ubiquotous shared infrastructure is goal
    - small machines **over** global machines
- Twine uses abstraction called entitlement = a logical grouping of machines that represent a quota of resources granted to a buisness unit
- Machines can be added or removed from an entitlement
- Jobs are bounds to entitlement instead of cluster and within entitlement, tasks of different jobs could stack together
- Any free machine can be ued by any entitlement in the region
- The **twine scheduler** is the control plane responsible for job and container life cycle management. It's deployed at regional and global scopes, where regional scheduler serves the same region and global scheduler manages servers from multiple regions.
- It is sharded across region level, where each shard manages a subset of entitlements in that region
- Schedulers connect to ZooKeeper via service discovery and operate in accordance with configuration from Configerator.
- The containers they schedule are composed of fbpkgs, the processes inside them consume configuration, and their endpoints are advertised by service discovery.
- Scheduler can evict a service from a machine at any time
- During a machine failure, it creates an unavailability event in Resource Broker, which notifies the allocator and scheduler.
- The scheduler disables the affected tasks in the service discovery system so that clients stop sending traffic to these tasks.
- A job is impacted by the machine failure if it has tasks running on the machine.

- If an impacted job has a TaskController, the scheduler informs the TaskController of the affected tasks.
- After the TaskController acknowledges that these tasks can be moved, the scheduler requests the allocator to deallocate the tasks and allocate new instances of the tasks on other machines.
- The scheduler instructs agents to start the new tasks accordingly.
- Finally, the scheduler enables the tasks in the service discovery system so that clients can send traffic to the newly started tasks.
- For **Job isolation**, twine packages an application's code and dependencies into an image and deploys it onto servers as containers, further cgroups, namespaces are used to provide isolation
- Containers provide isolation between multiple applications running on the same server, allowing developers to focus on application logic without worrying about how to acquire servers or orchestrate upgrades of their applications
- A daemon named **twine agent** runs inside everys erver, which is responsible for setting up and tearing down containers.
- Application-level schedulers are built atop Twine to better support **vertical workloads** such as stateful, batch, machine learning, stream processing, and video processing.
- For workloads that require better locality for compute and storage, Twine allows an entitlement to override the default spread policy and pin its machines and jobs to a specific DC. These workloads are in the minority.

Google Borg

- It is a cluster manager that runs millions of jobs from many applications across number of clusters
- Users submit the work to Borg in terms of jobs each consisting one or more tasks that all run the same binary/program.Each job runs in one Borg cell, a set of machines that are managed as a unit.
- The cells run a **heterogeneous workload** with two parts
    - Production jobs - such as Gmail, Calendar, etc, i.e. the ones which the clients access
        - these should never go down
        - and suffer very less latency
    - Batch jobs
        - takes few seconds to days to complete
- Every borg cell consists of set of machines, logivally centeralized controller called the Borgmaster, an agent process called Borglet that runs on each machine in a cell.
- A program is to be run on cluster, the request is given to borgmaster
- Borgmaster writes to persistent storage
- Scheduler sometimes later (asynchronously) picks the job **priority wise**, checks how many taks to be run and finds what machines should be used for the purpose

- The scheduling algorithm is appended with a round robin scheme to ensure non=blocking and fairness for each user
- The production jobs are given higher priority and can preempt non-production jobs whenever they want
- In borg, tasks are run inside cgroup-based resource container and settings of the container are ensured by Borglet.
- Tasks are permitted to consume resources up to their limits. Most of them are allowed to go beyond that for compressible resources like CPU, to take advaltage of unused resources

References

- Twine: A Unified Cluster Management System for Shared Infrastructure
  - youtube video
- Large-scale cluster management at Google with Borg
  - youtube video