



Where's my Instance?

How CloudStack finds a suitable deployment destination

Jithin Raju, ShapeBlue





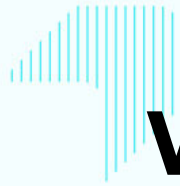
About Me



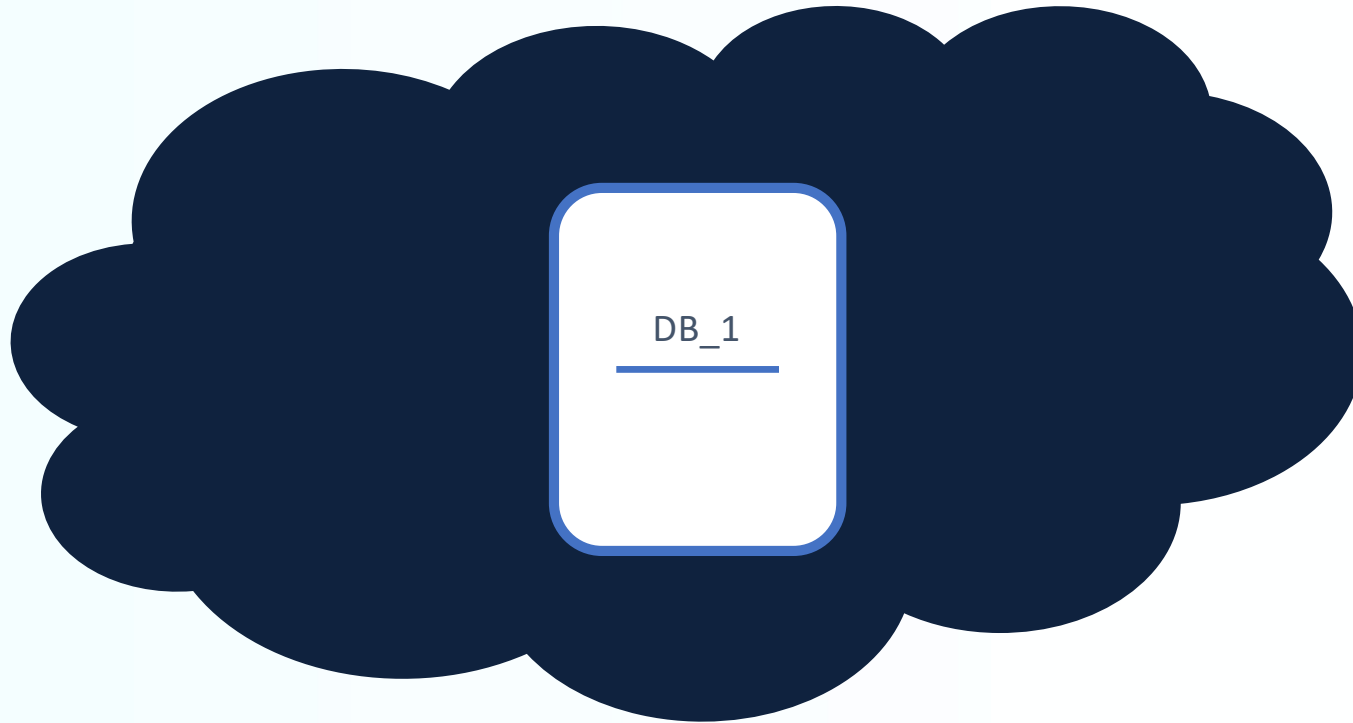
- Apache CloudStack Committer
- Cloud Architect at ShapeBlue
- Hands-on experience in designing, deploying, and supporting CloudStack

<https://www.linkedin.com/in/rajujith/>



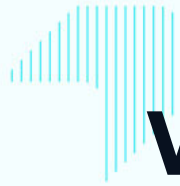


Where's my Instance

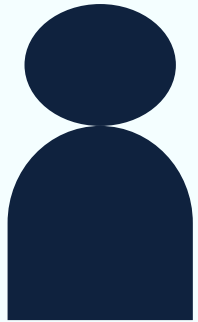




**IT'S JUST
SOMEONE ELSE'S COMPUTER**



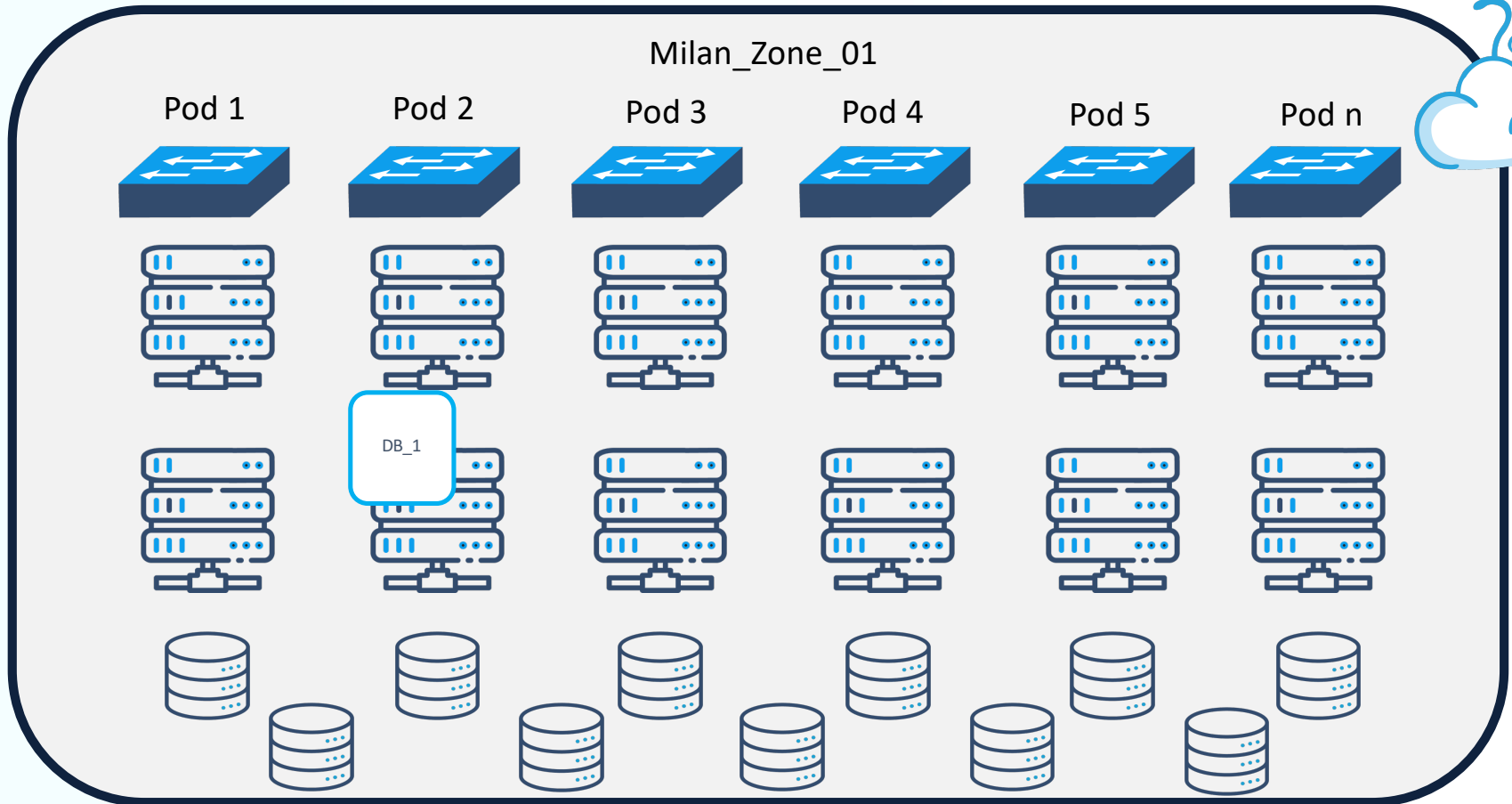
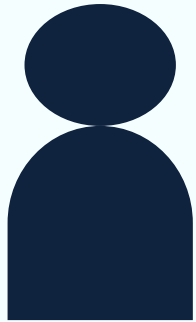
Where's my Instance



Milan_Zone_01

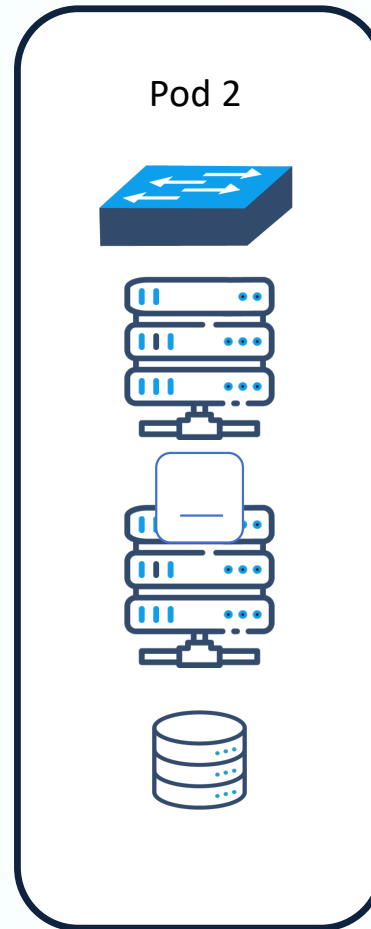


Where's my Instance

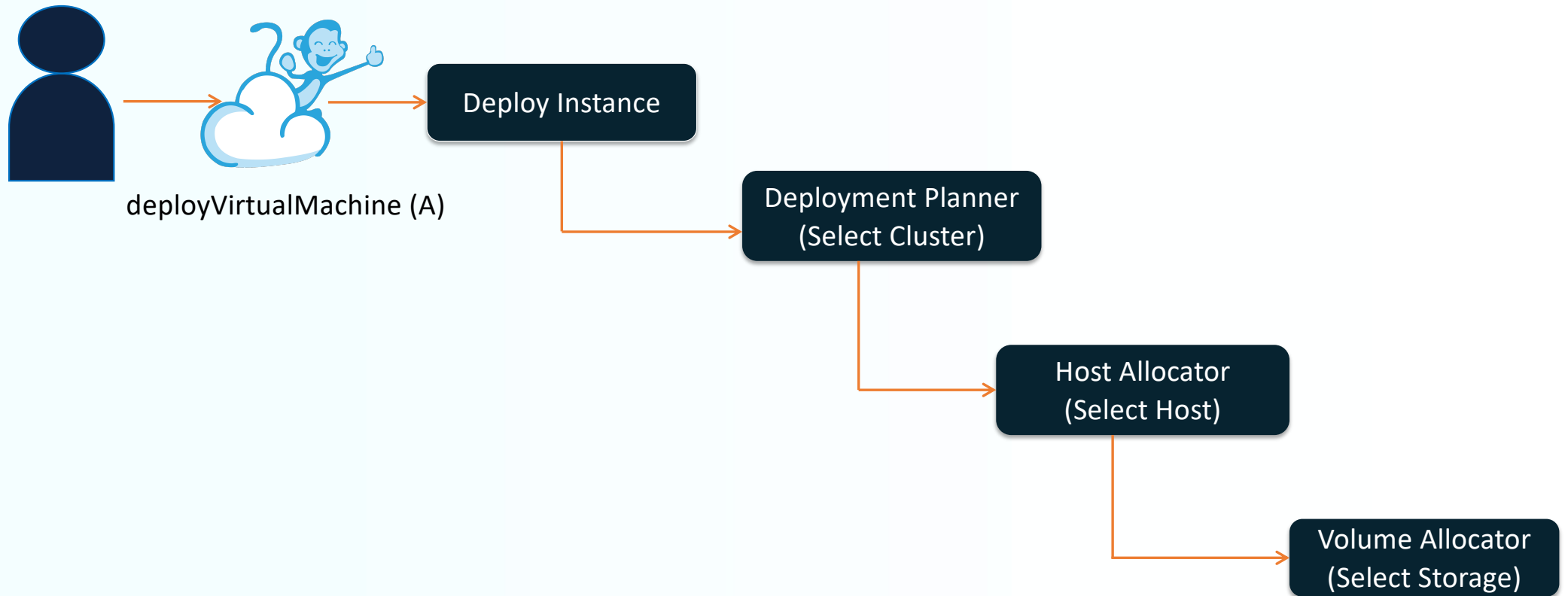


Finding a suitable deployment destination

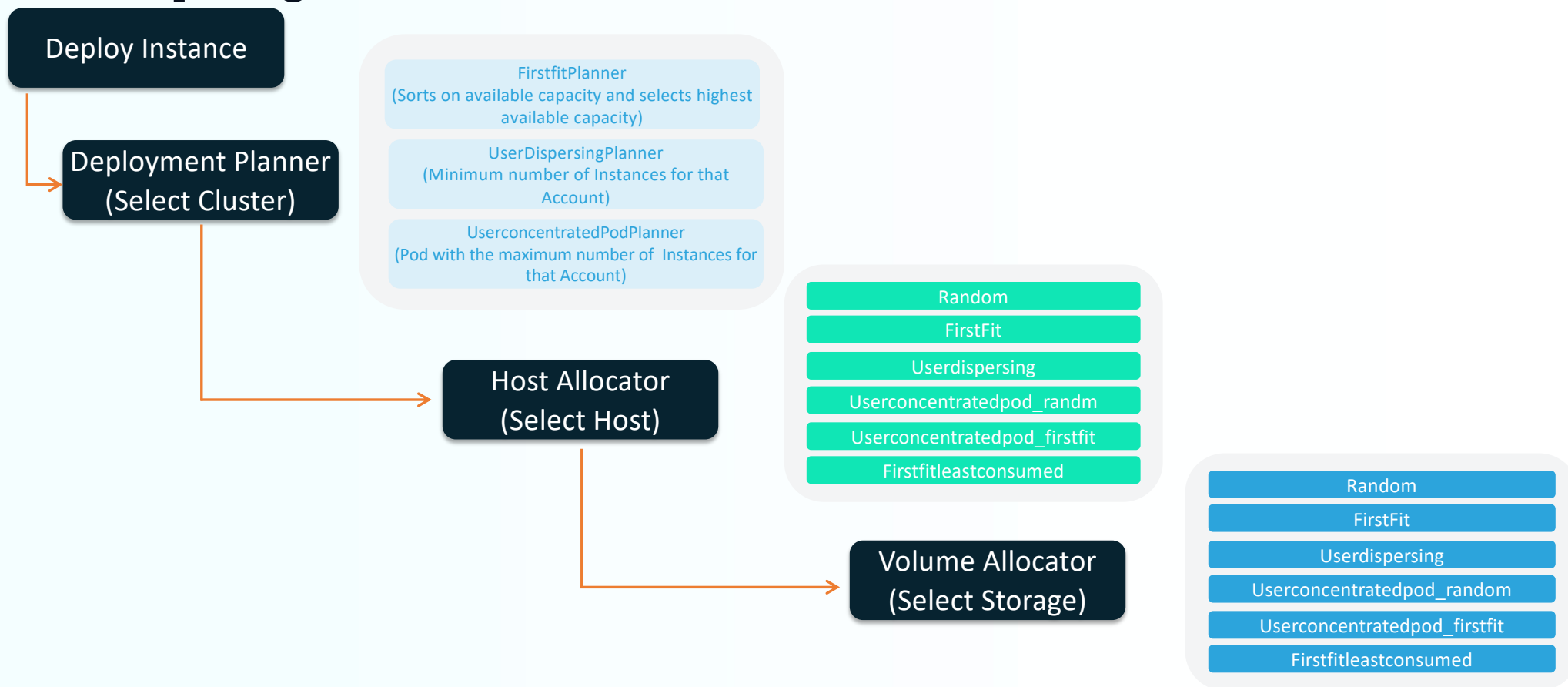
Found Pod1,
Cluster CL2,
Host H2,
Primary
Storage PS3



How CloudStack finds a suitable deployment destination



How CloudStack finds a suitable deployment destination

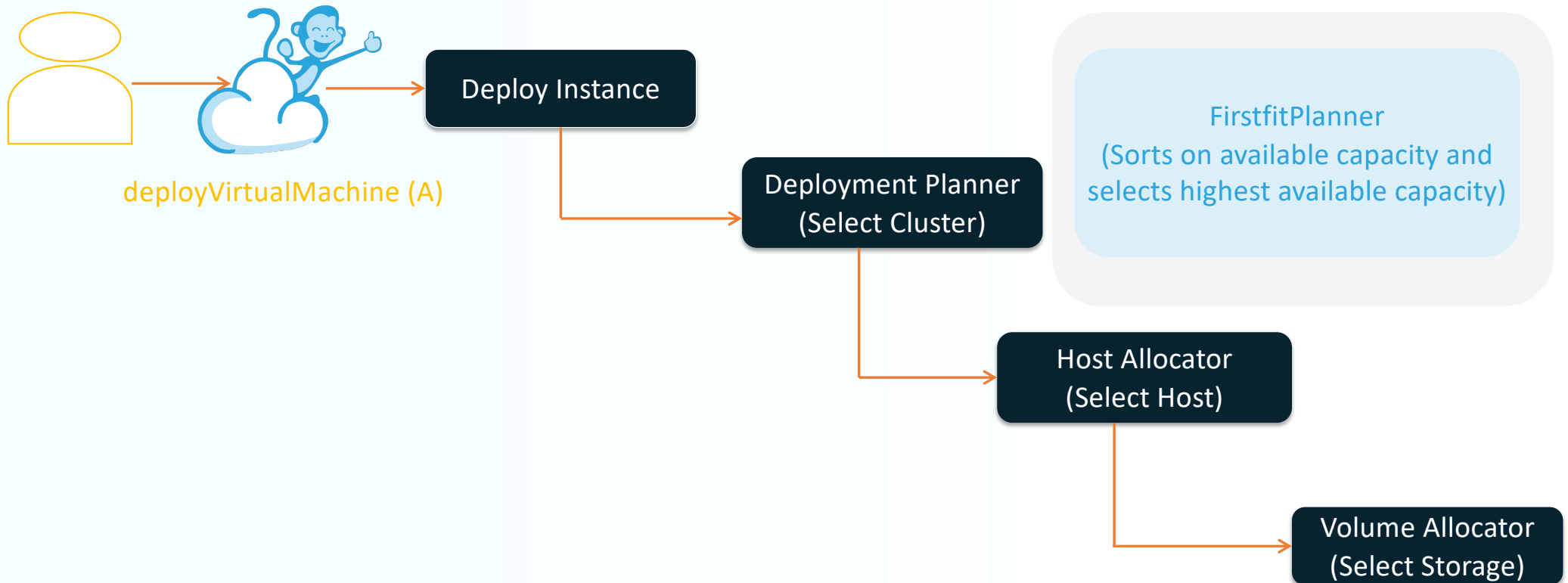




Deployment Planner

- Global configuration '[vm.deployment.planner](#)' enables root admins to influence the VM deployment process by selecting a Cluster. Also configurable per Compute Offering.
 - FirstFitPlanner
 - UserDispersingPlanner
 - UserconcentratedPodPlanner

How CloudStack finds a suitable deployment destination





Deployment Planner (FirstFitPlanner)



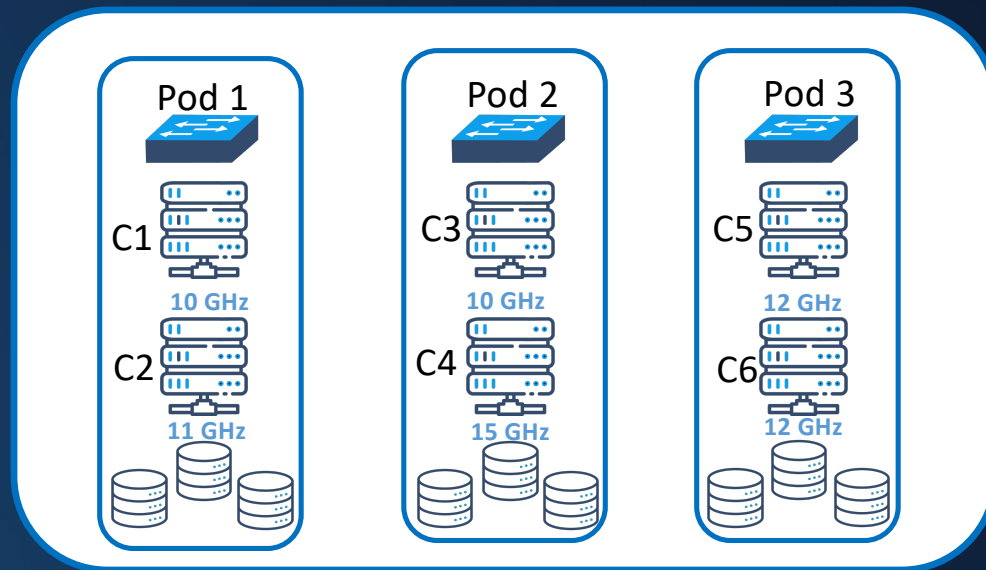


Deployment Planner (FirstFitPlanner)

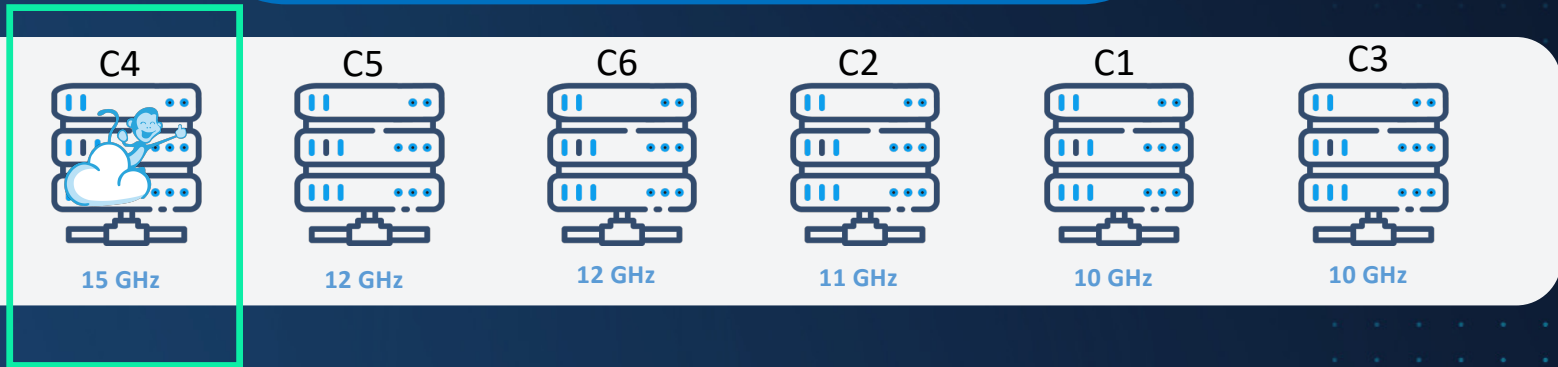
General-purpose, balanced placement — when you want CloudStack to fill clusters based on available capacity rather than user affinity or dispersion.

- Default
- Simplest, capacity-oriented
- Prioritises Clusters with the most available (Free) capacity
- Creates a sorted list of Clusters in a zone, ordered by total available CPU/Memory
- Order: Descending i.e clusters with more free resources are checked first

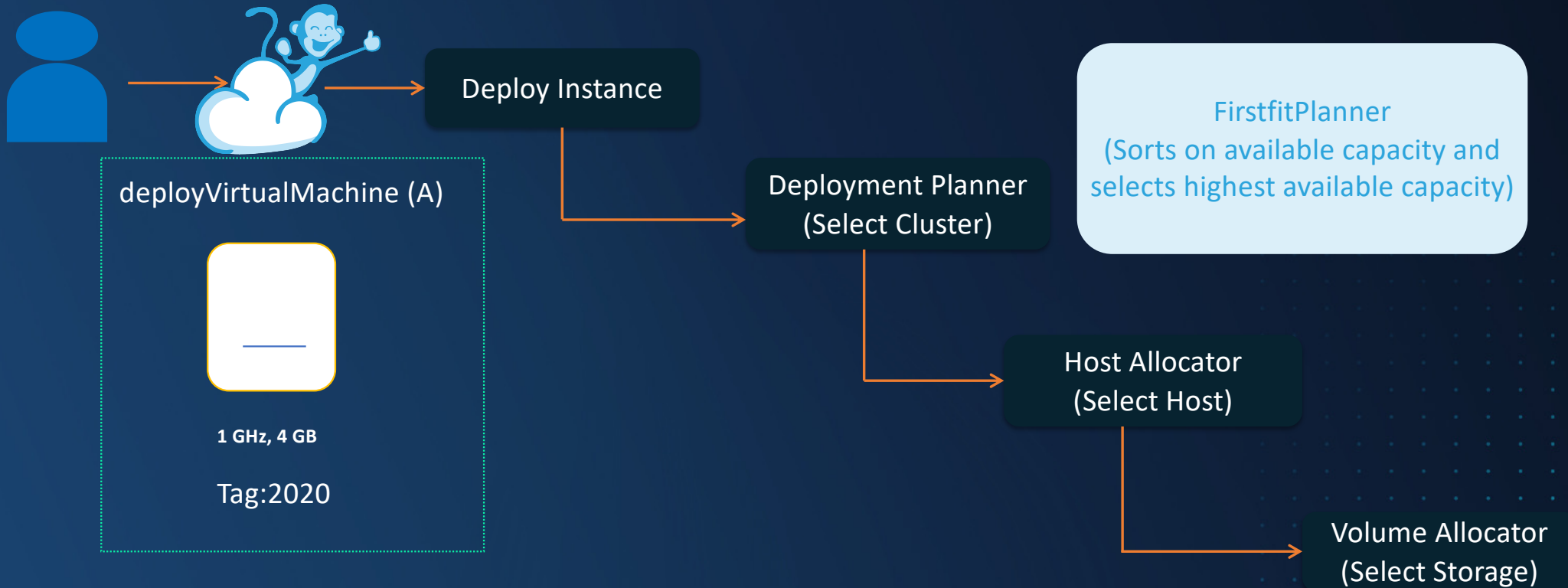
Deployment Planner (FirstFitPlanner)



Ordered List



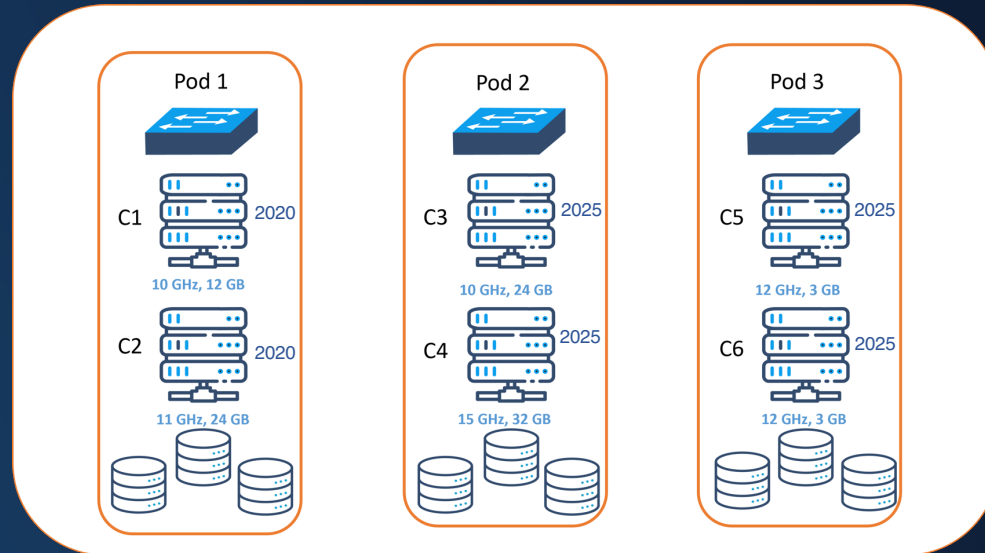
How CloudStack finds a suitable deployment destination



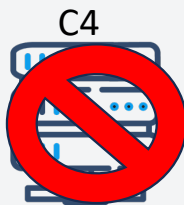
Deployment Planner (FirstFitPlanner)



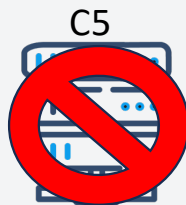
1 GHz, 4 GB
Tag:2020



Ordered
List



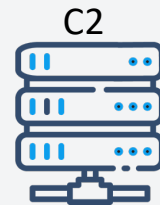
15 GHz, 32 GB



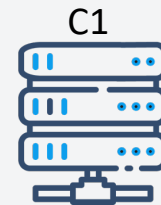
12 GHz, 3 GB



12 GHz, 3 GB



11 GHz, 24GB



10 GHz, 12 GB



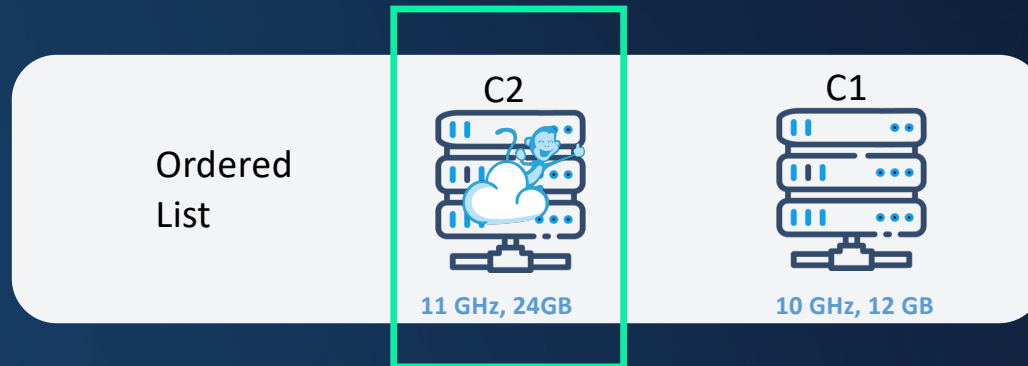
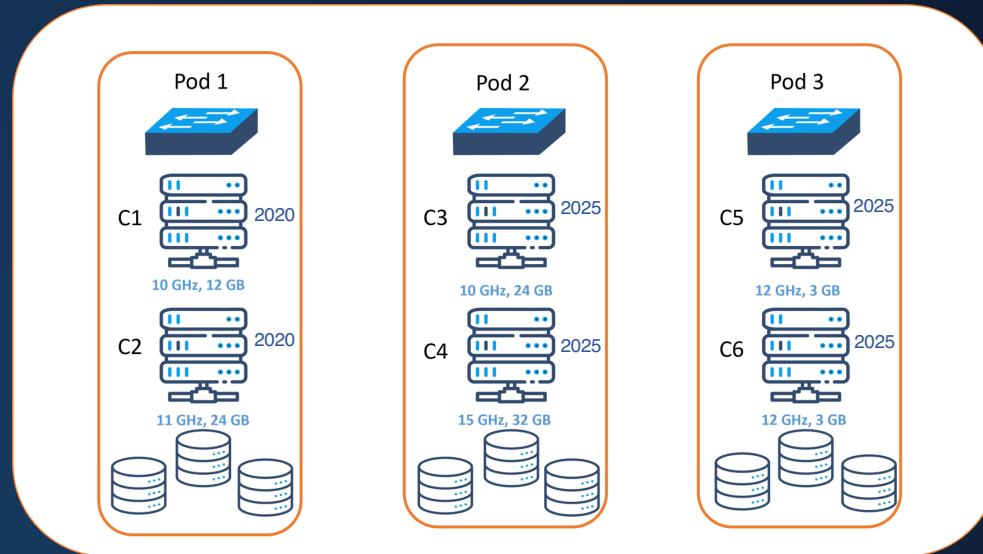
10 GHz, 24 GB

Deployment Planner (FirstFitPlanner)

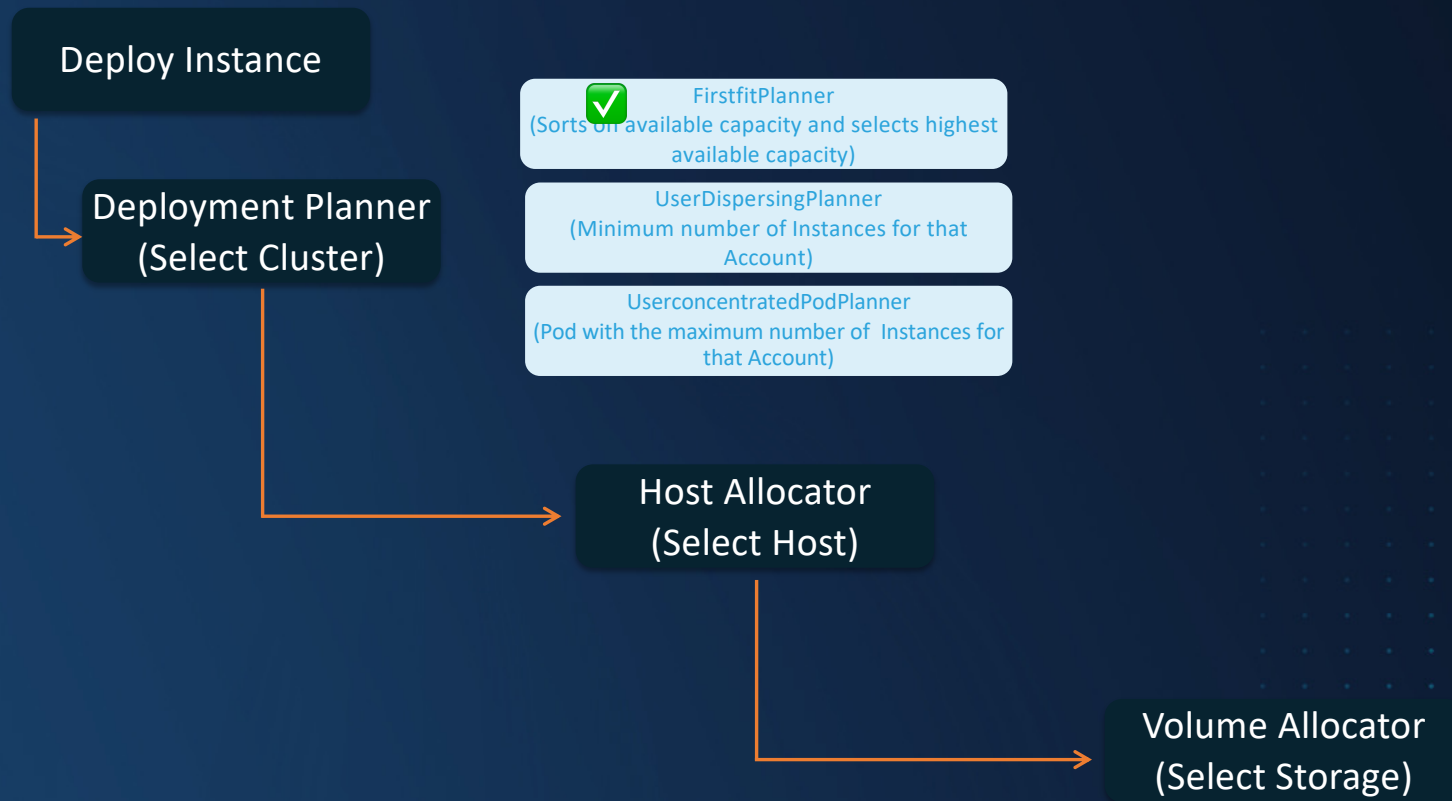


1 GHz, 4 GB

Tag:2020



How CloudStack finds a suitable deployment destination





Deployment Planner

(UserDispersingPlanner)



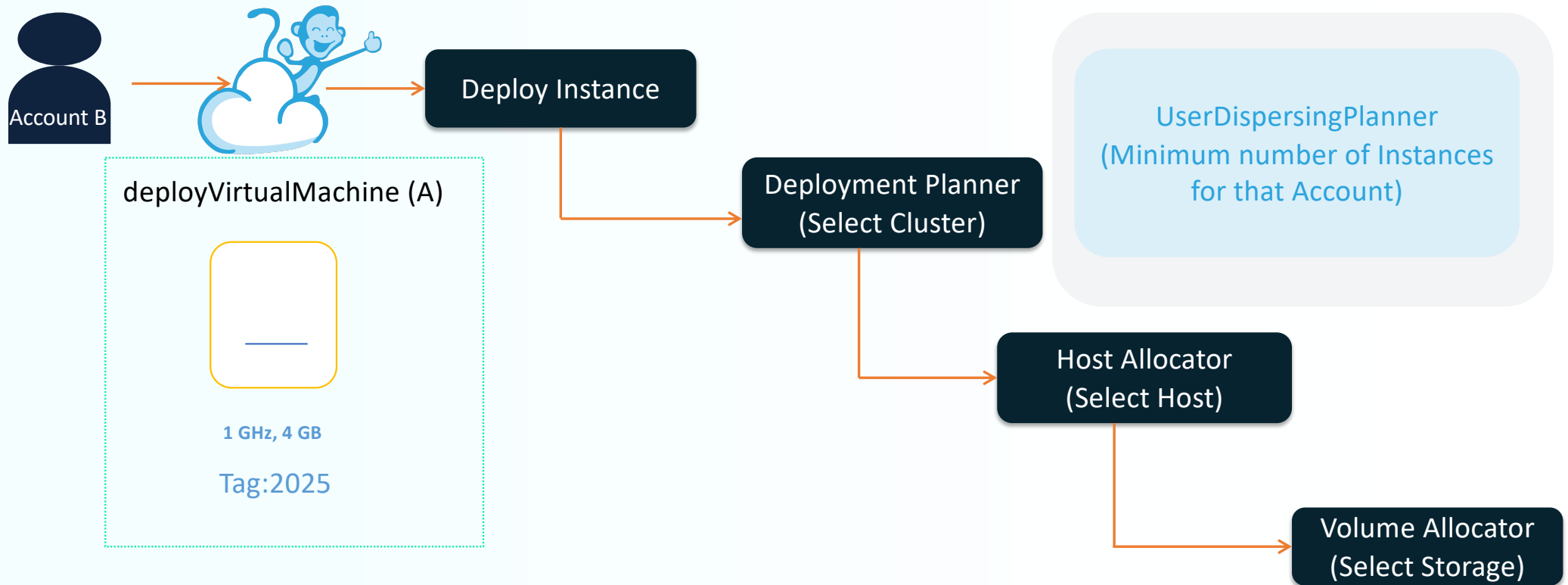


Deployment Planner (UserDispensingPlanner)

Resilient, user-aware placement — when you want to avoid placing all VMs of a single user or tenant in the same cluster.
Reduces the impact on tenants in case of infrastructure failures

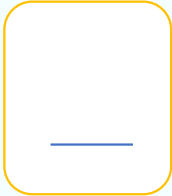
- Spreads a user's Instances evenly across Clusters in Zone
- "Starting" Instances as well from 4.20.2 onwards
- Clusters with fewer Instances of the Account are preferred by default
- Global setting `vm.user.dispersion.weight`, which defaults to 1
 - `Dispersion_weight < 1` → capacity gets more priority

How CloudStack finds a suitable deployment destination



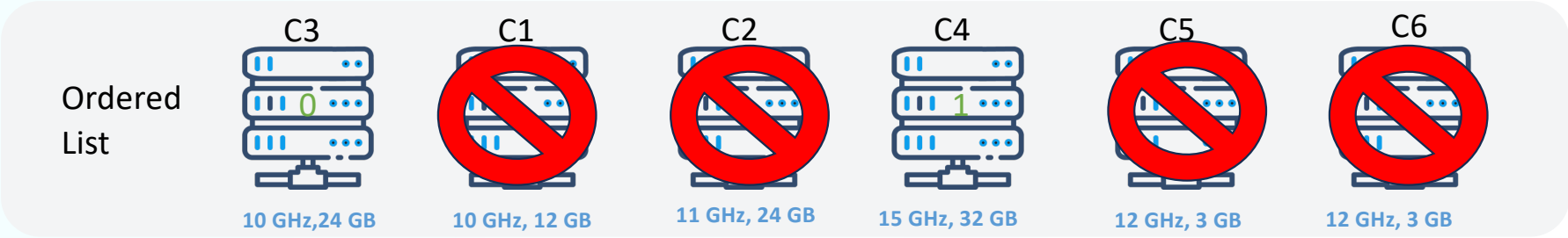
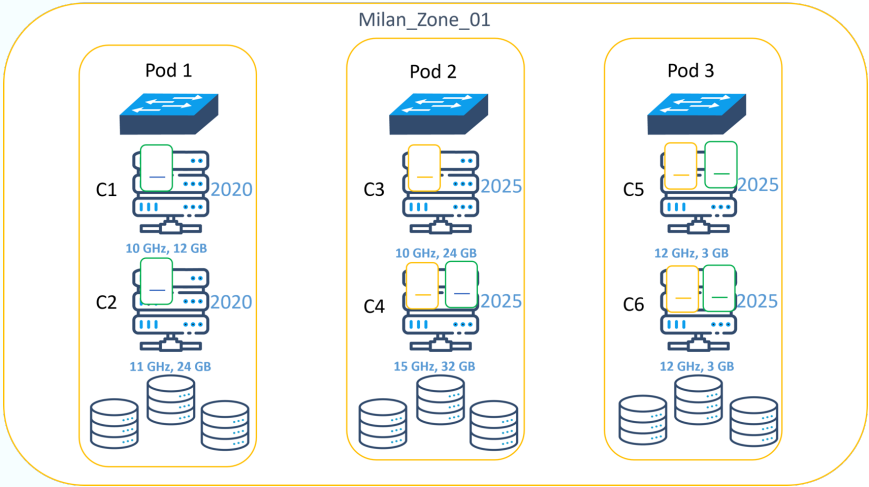
Deployment Planner (UserDispersingPlanner)

Account B



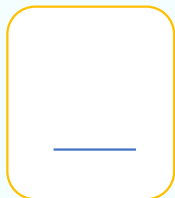
1 GHz, 4 GB

Tag:2025



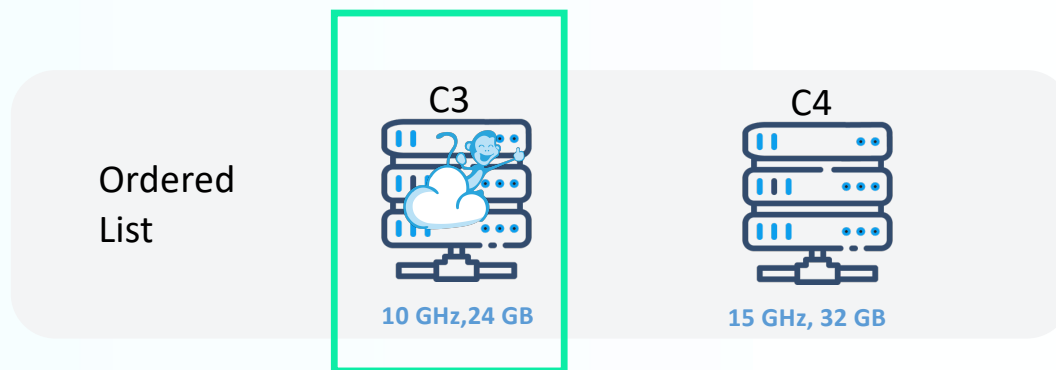
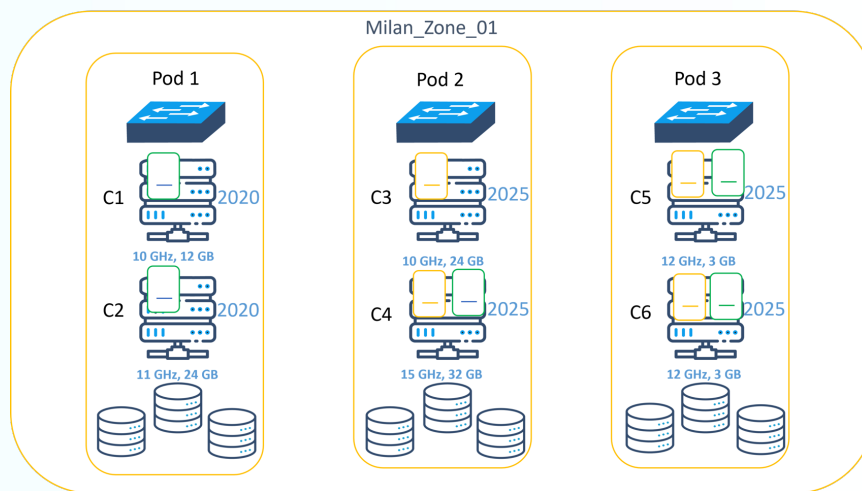
Deployment Planner (UserDispersingPlanner)

Account B



1 GHz, 4 GB

Tag:2025





Deployment Planner

(UserconcentratedPodPlanner)



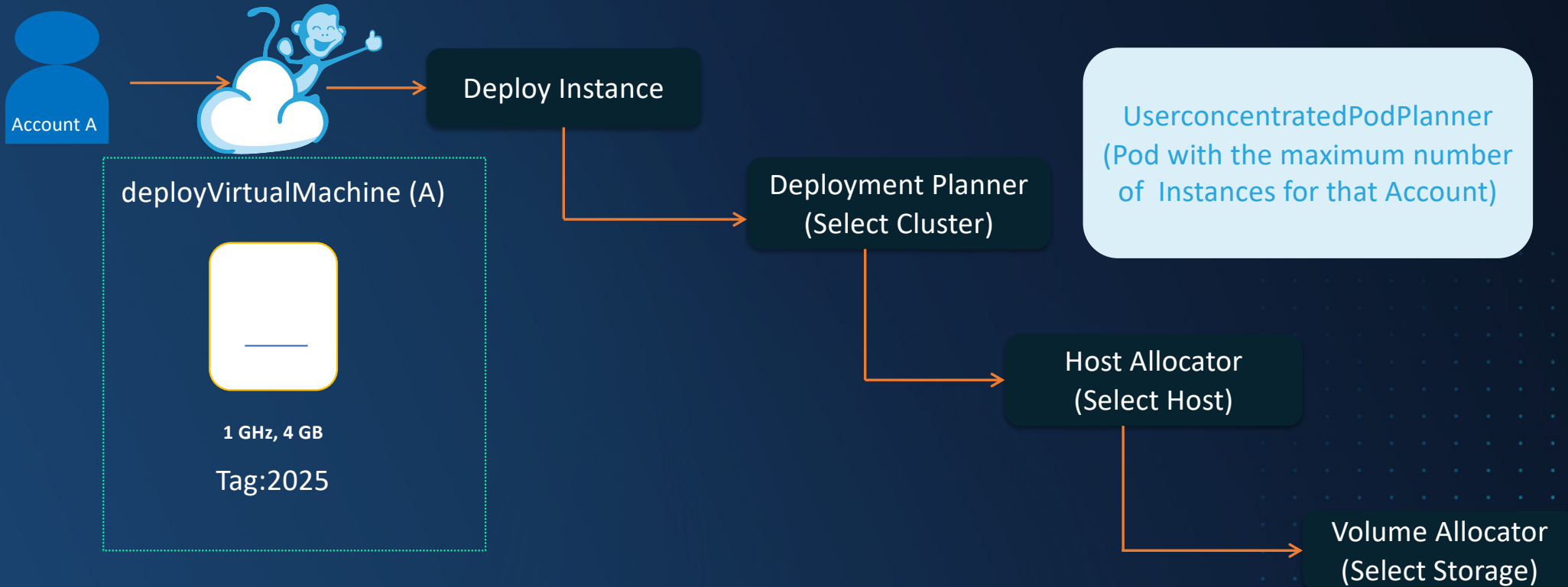
A decorative graphic consisting of a series of vertical lines of varying heights, creating a stylized arrow or wave effect pointing to the right.

Deployment Planner (UserconcentratedPodPlanner)

Performance and locality optimization — when you want a user's workloads to stay within the same pod for faster communication and better data locality.

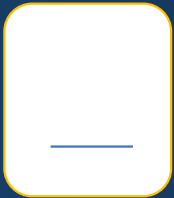
- Pod with the maximum number of Instances for that Account
- Sorting at the Pod level
- Clusters in Pods where the user is already active are prioritized.
- Not sorted based on capacity
- Starting Instances as well, soon

How CloudStack finds a suitable deployment destination



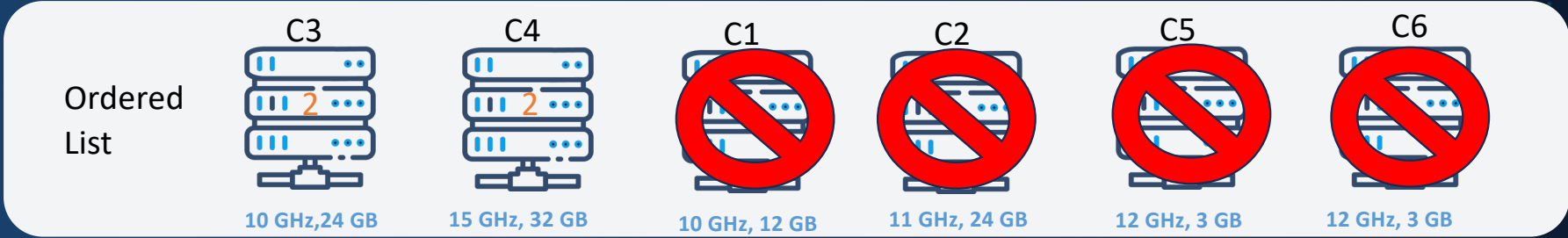
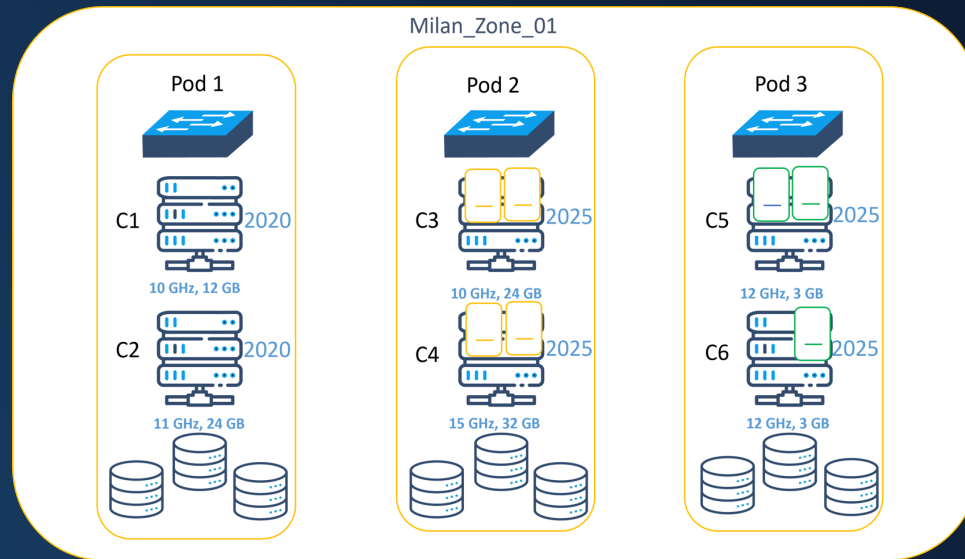
Deployment Planner (UserconcentratedPodPlanner)

Account A



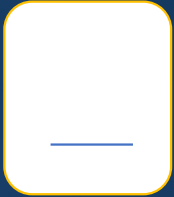
1 GHz, 4 GB

Tag:2025



Deployment Planner (UserconcentratedPodPlanner)

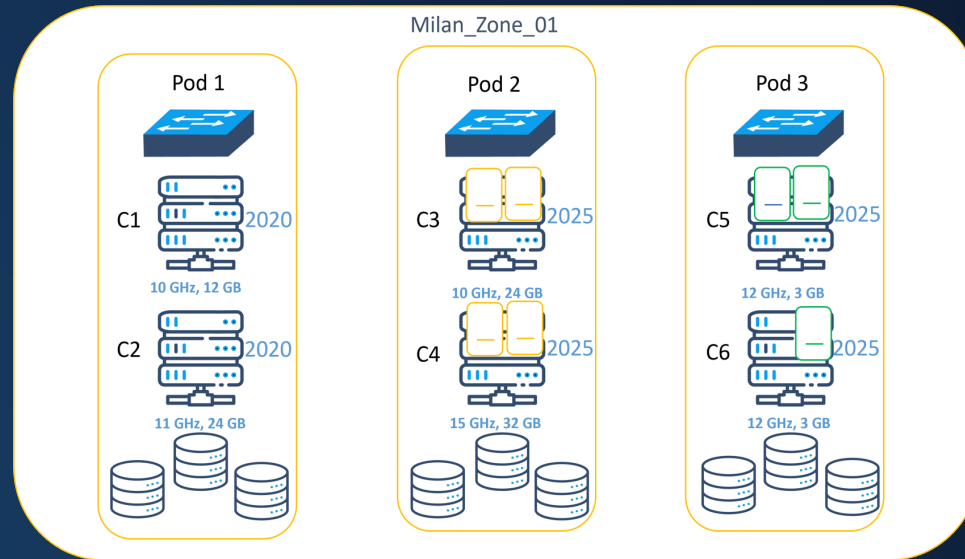
Account A



1 GHz, 4 GB

Tag:2025

Milan_Zone_01

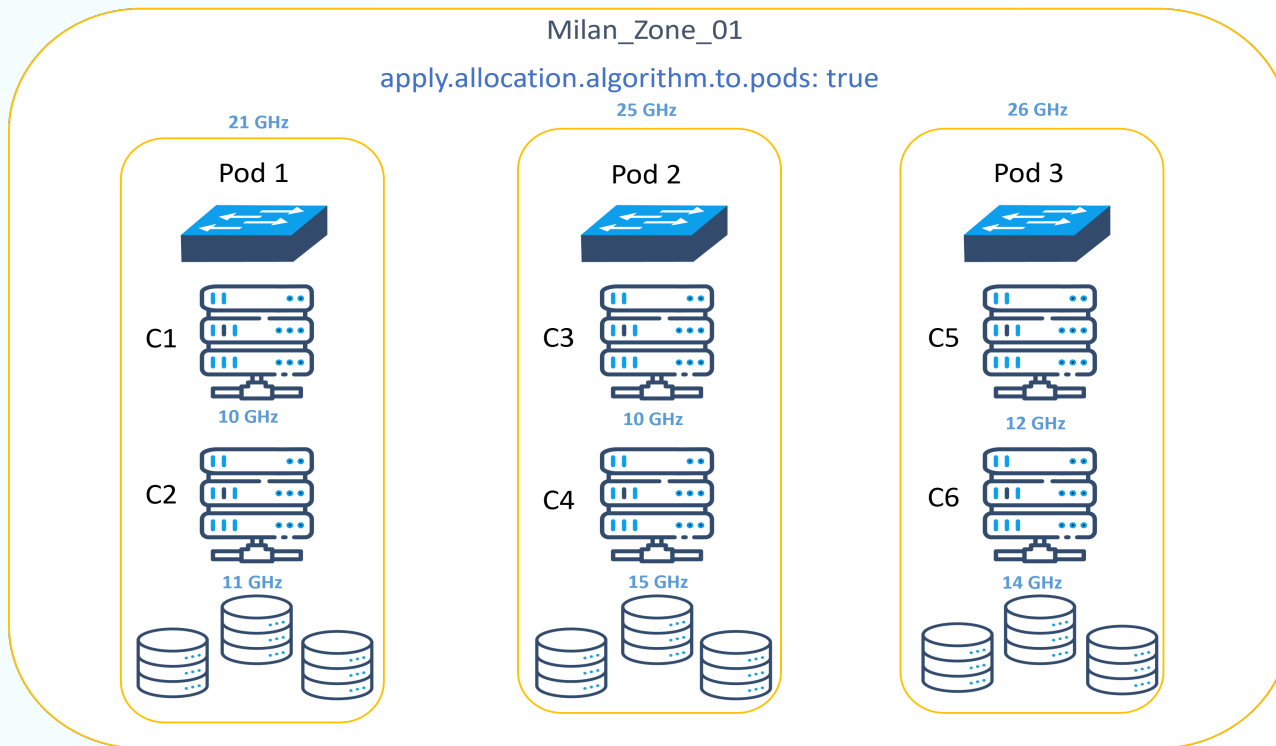


Ordered
List

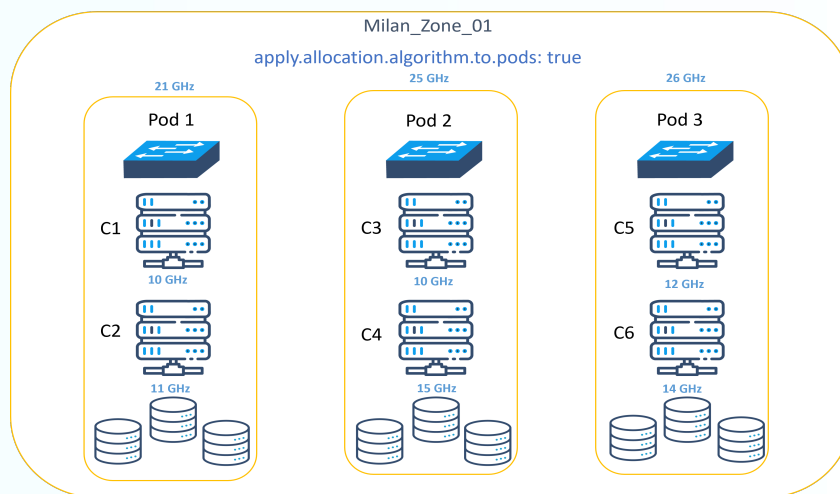




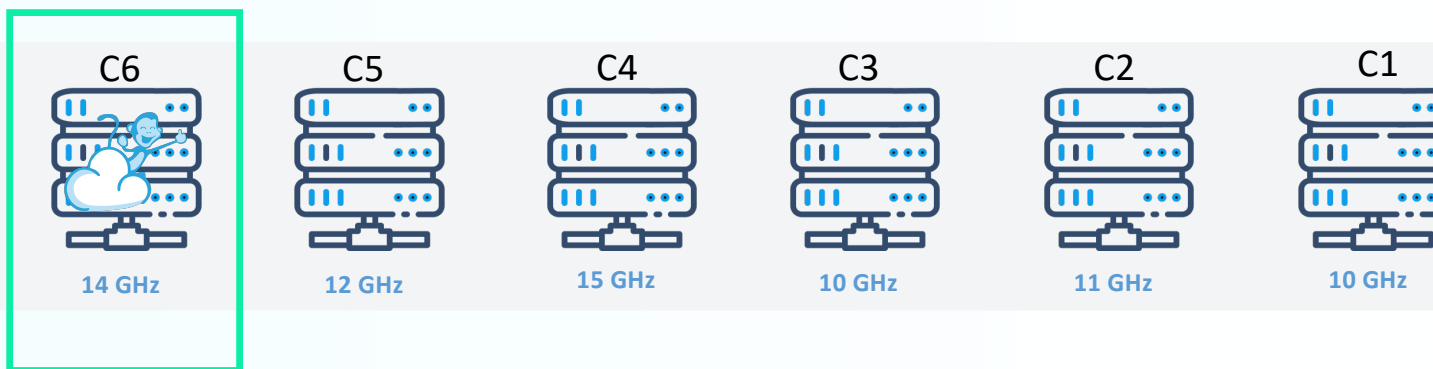
Deployment Planner (Pod ordering)

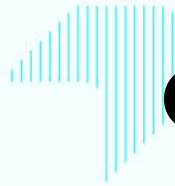


Deployment Planner (FirstFitPlanner)



Ordered List



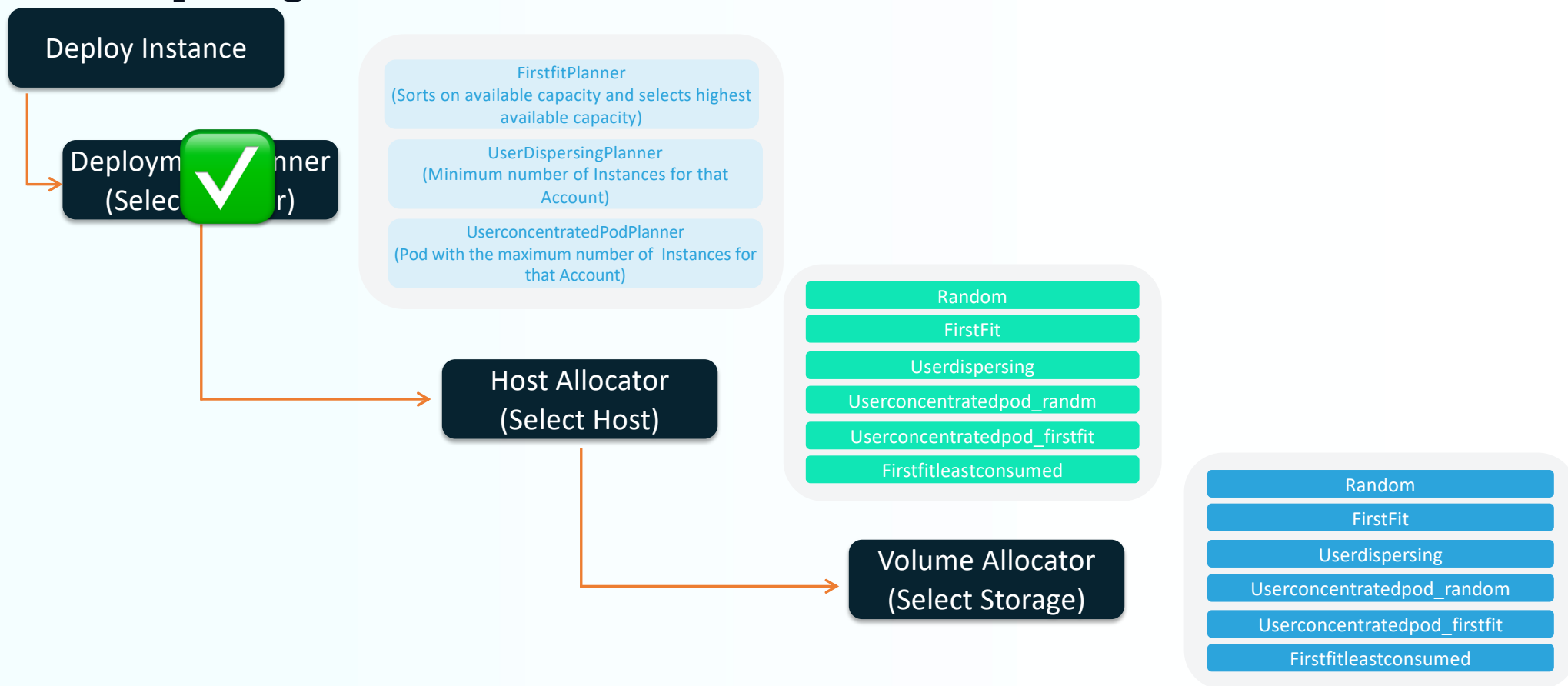


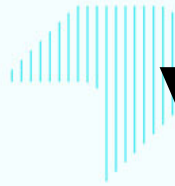
Other Influencing Factors



- Hypervisor Type
- Cluster State
- CPU Architecture
- GPU
- Dedication
- Affinity Groups
- Hardware specific resource limits
- OS preference

How CloudStack finds a suitable deployment destination



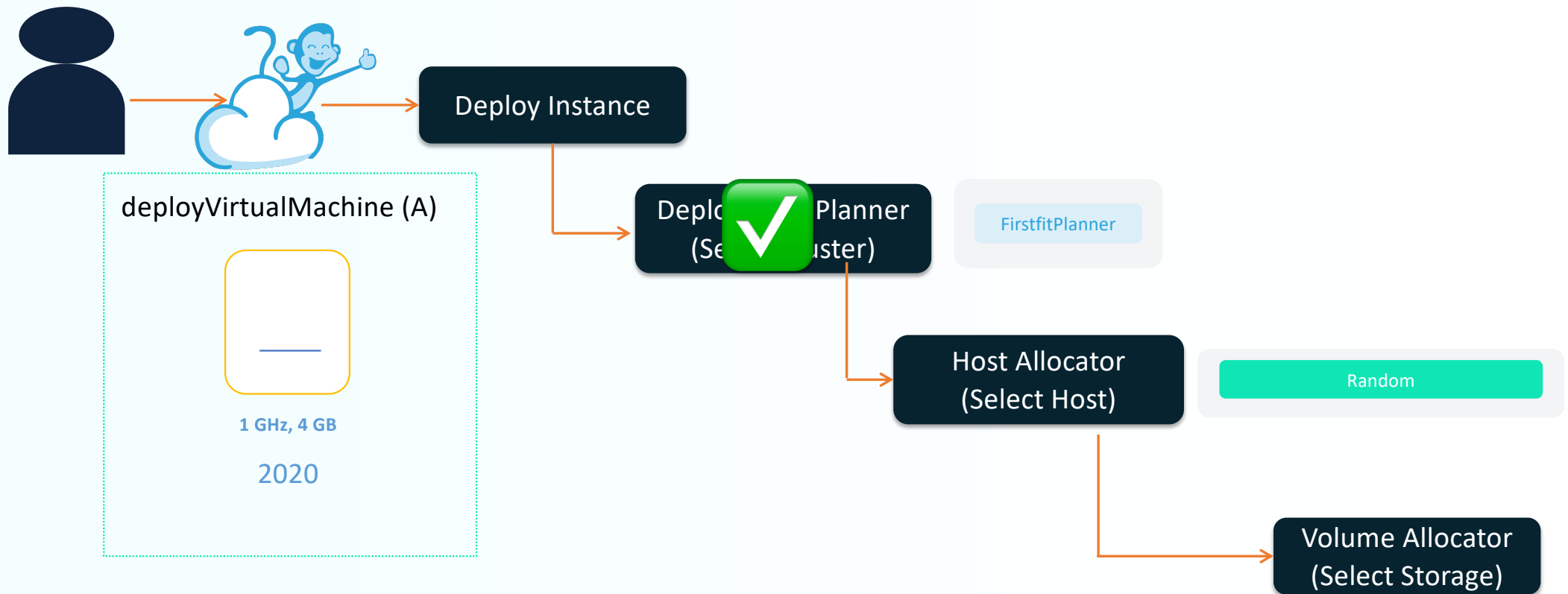


VM Allocation Algorithm

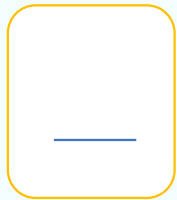


- Global : `vm.allocation.algorithm`
 - Random (Default)
 - FirstFit
 - Userdispersing
 - Firstfileastconsumed
 - UserconcentratedPod_Random
 - UserconcentratedPod_FirstFit

How CloudStack finds a suitable deployment destination



VM Allocation Algorithm(Random)

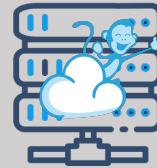


1 GHz, 4 GB

Tag:2020

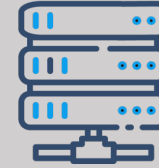
Ordered
List

C2



11 GHz, 24GB

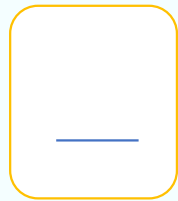
C1



10 GHz, 12 GB

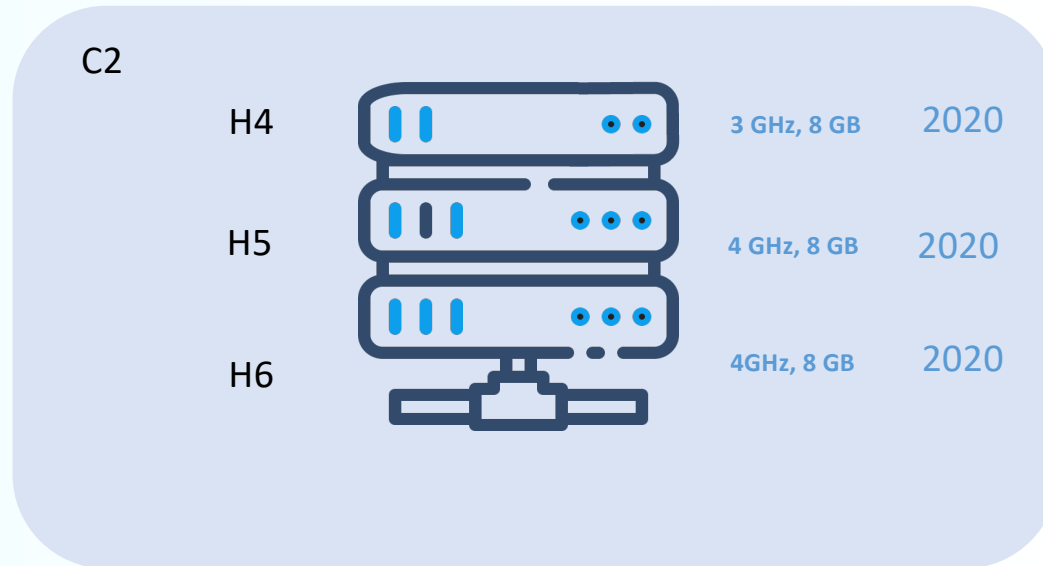


VM Allocation Algorithm(Random)

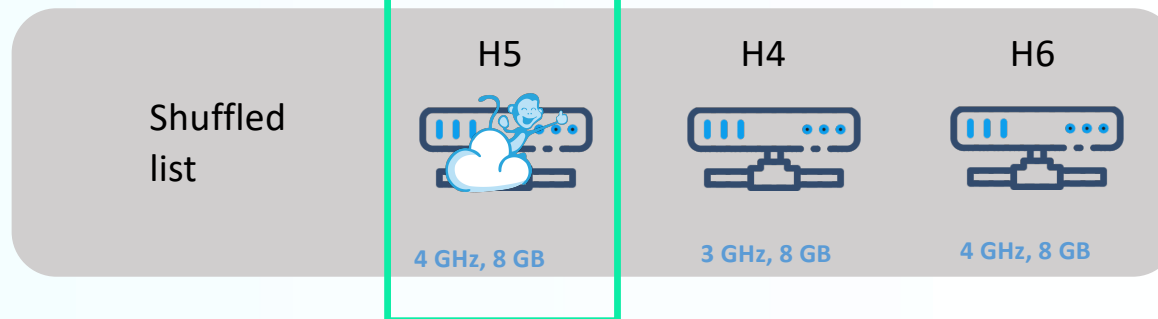


1 GHz, 4 GB

Tag:2020



Simple and even distribution , no sorting or ordering



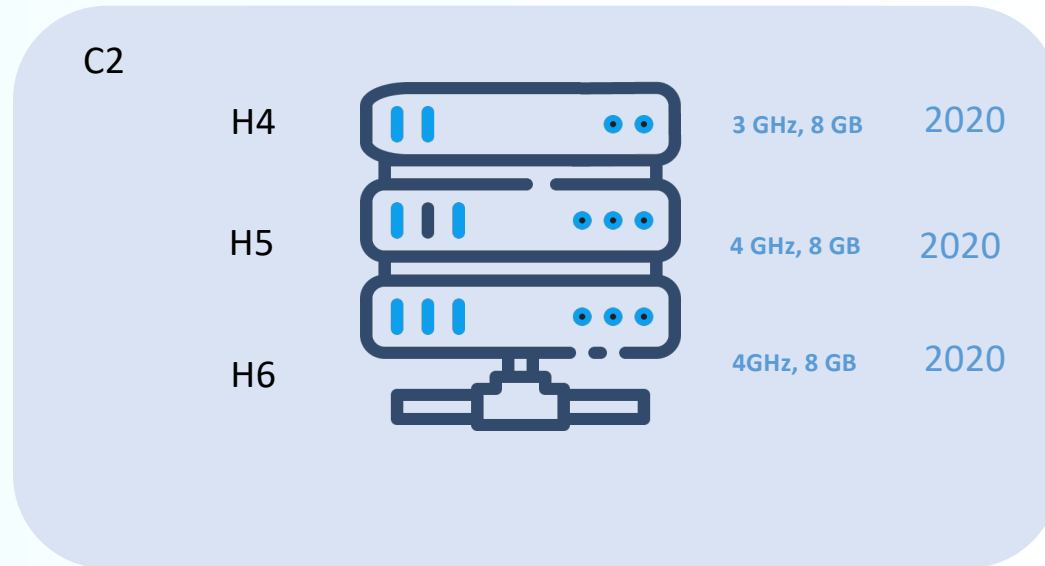


VM Allocation Algorithm(Firstfit)

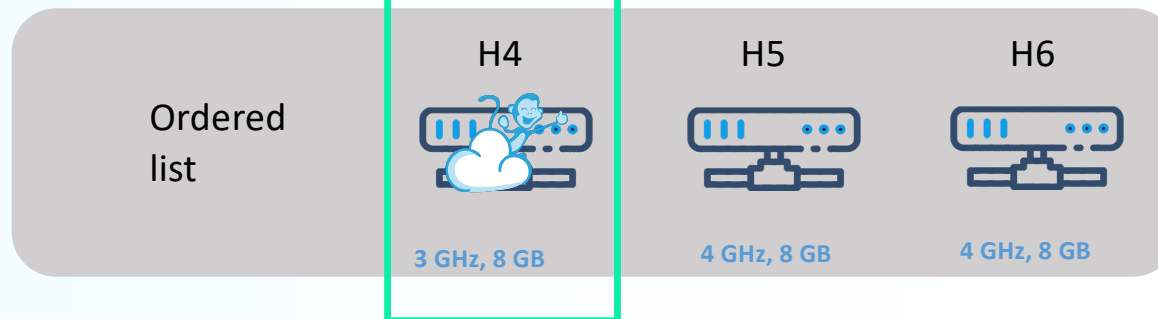


1 GHz, 4 GB

Tag:2020



Simple, straightforward,
and predictable



VM Allocation Algorithm (Userdispersing)

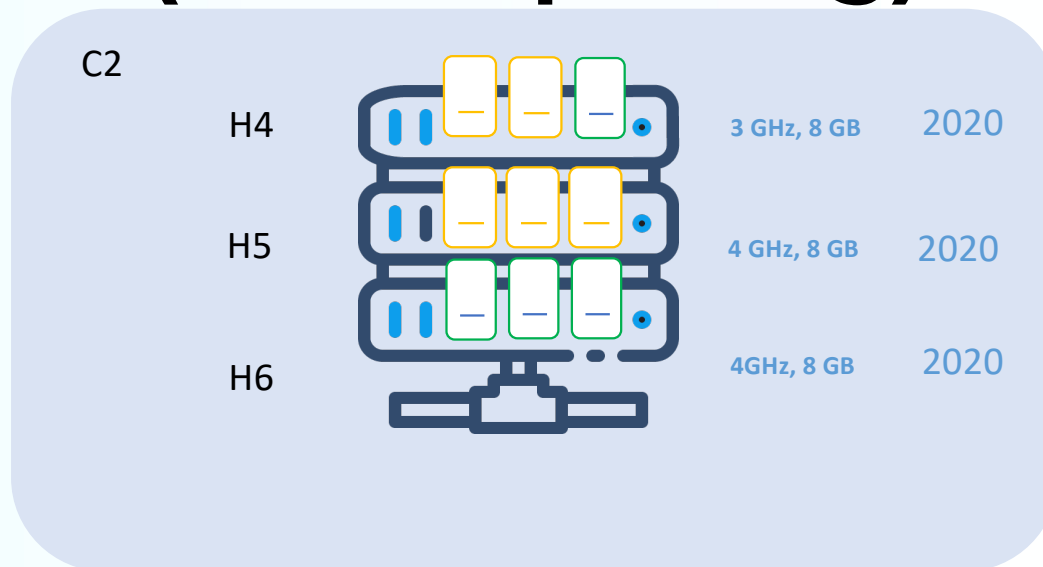


Account A

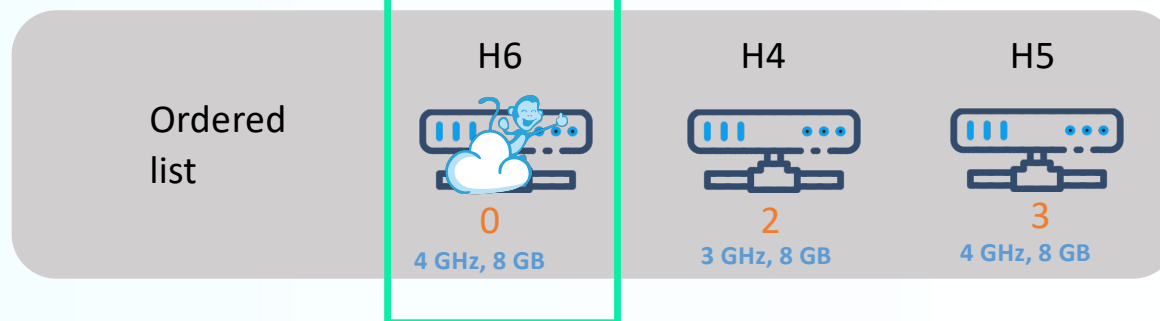


1 GHz, 4 GB

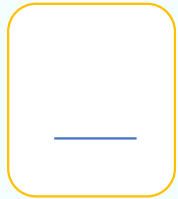
Tag:2020



Reduces the impact on tenants in case of infrastructure failures

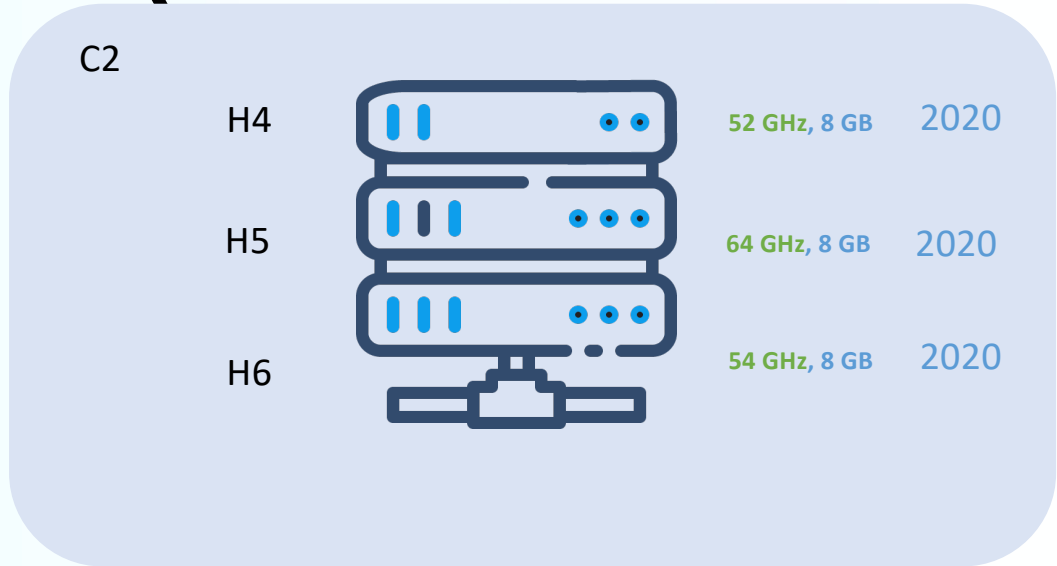


VM Allocation Algorithm (FirstFitLeastConsumed)

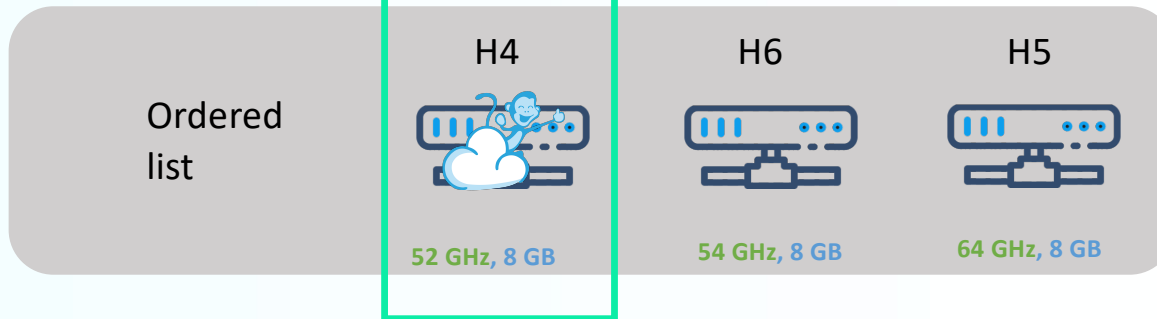


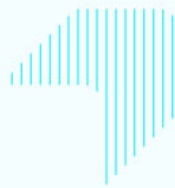
1 GHz, 4 GB

Tag:2020

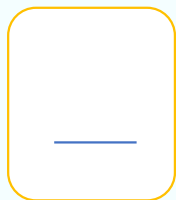


Load-balanced , even utilisation



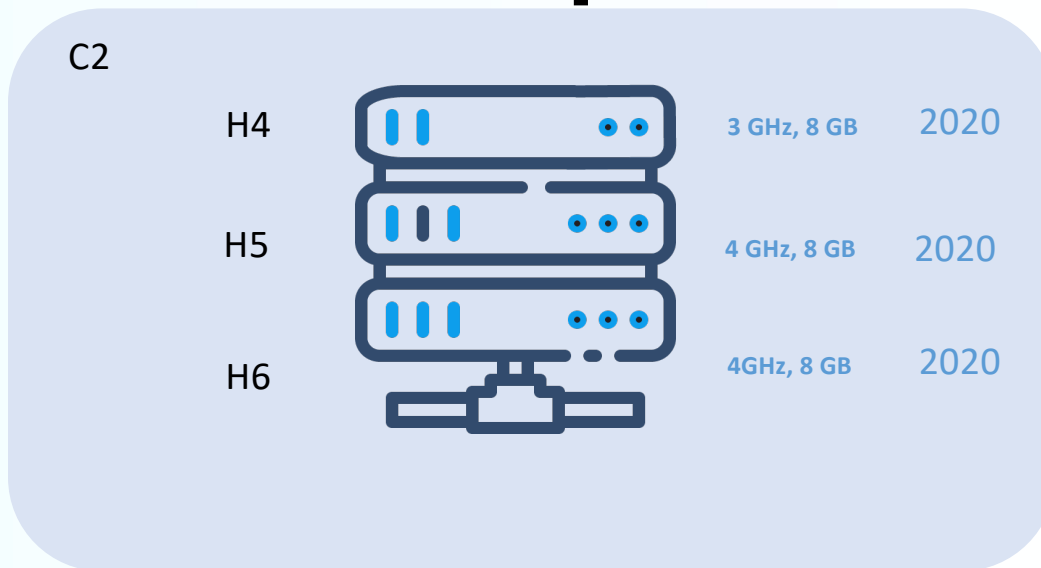


VM Allocation Algorithm (Userconcentratedpod_random)

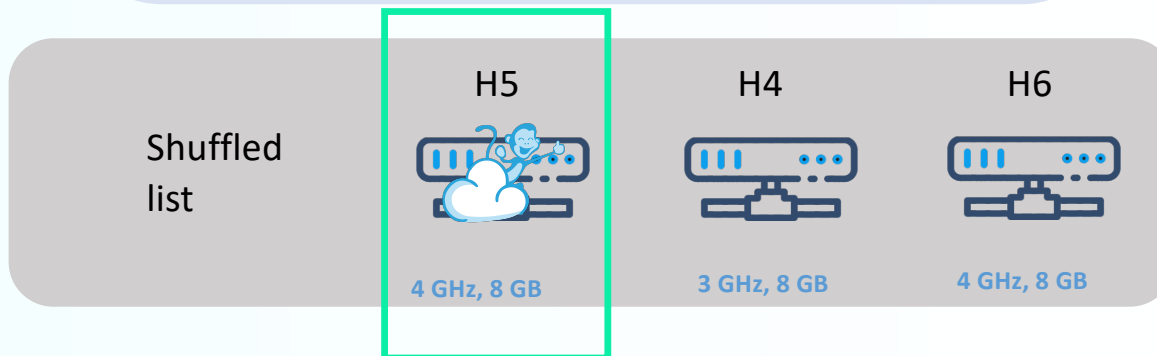


1 GHz, 4 GB

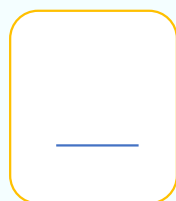
Tag:2020



Same as Random

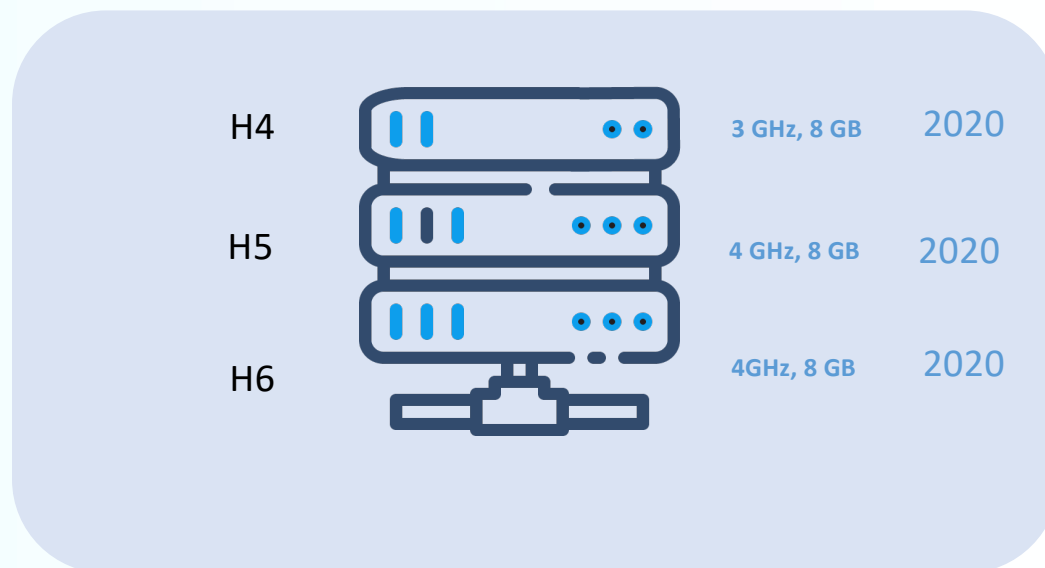


VM Allocation Algorithm (Userconcentratedpod_firstfit)

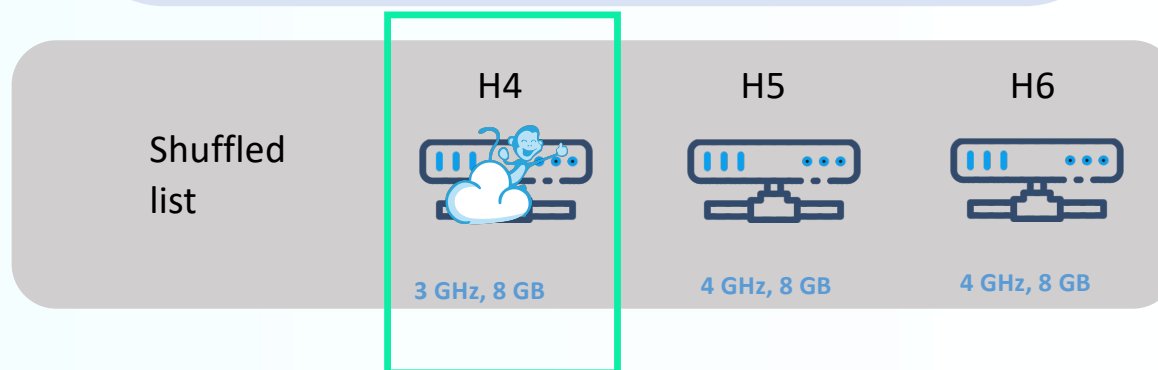


1 GHz, 4 GB

Tag:2020



Same as FirstFit

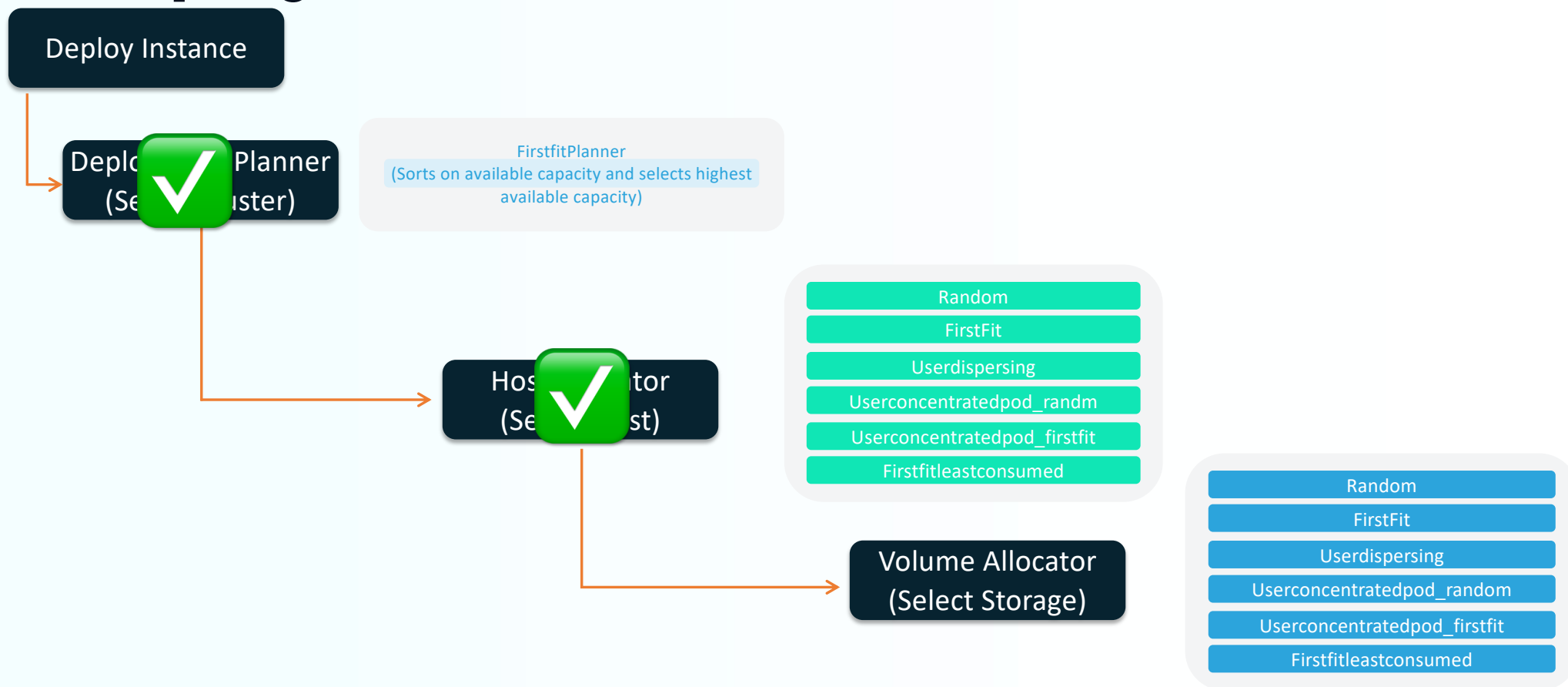


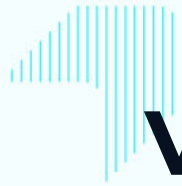
A decorative graphic consisting of a series of vertical lines of varying heights, resembling a bar chart or a stylized 'S' shape, located in the top left corner.

Capacity Type to order

- `host.capacityType.to.order.clusters`
- Default CPU
- Options: CPU, RAM
- New in 4.21 : COMBINED
 - Default : 0.5
 - `host.capacityType.to.order.clusters.cputomemoryweight`
 - Metric = CPU * weight + RAM * (1- weight)

How CloudStack finds a suitable deployment destination

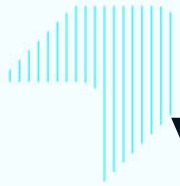




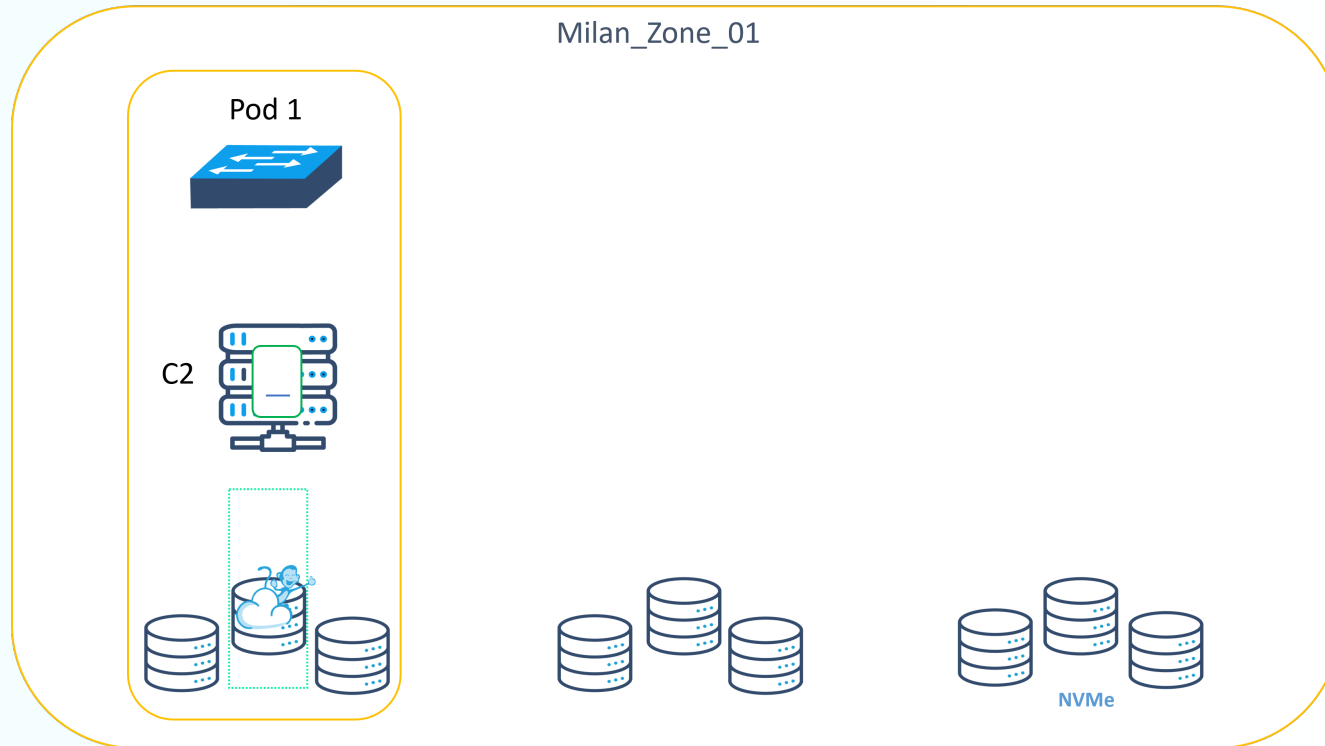
Volume Allocator



- `volume.allocation.algorithm (4.21)`:
 - Random (Default)
 - FirstFit
 - Userdispersing
 - Firstfileastconsumed
 - UserconcentratedPod_Random
 - UserconcentratedPod_FirstFit

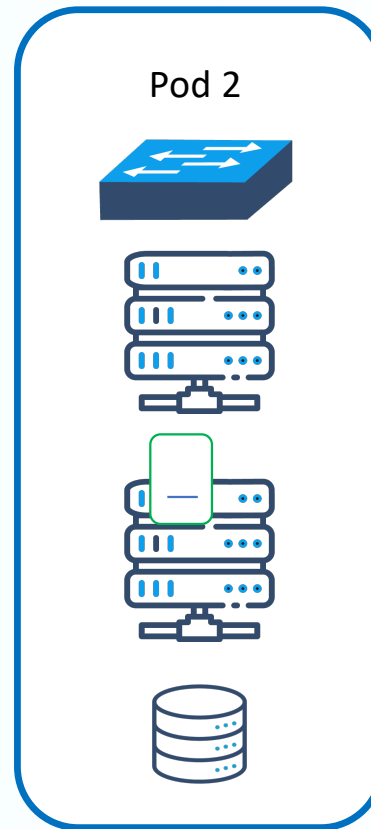


Volume Allocation



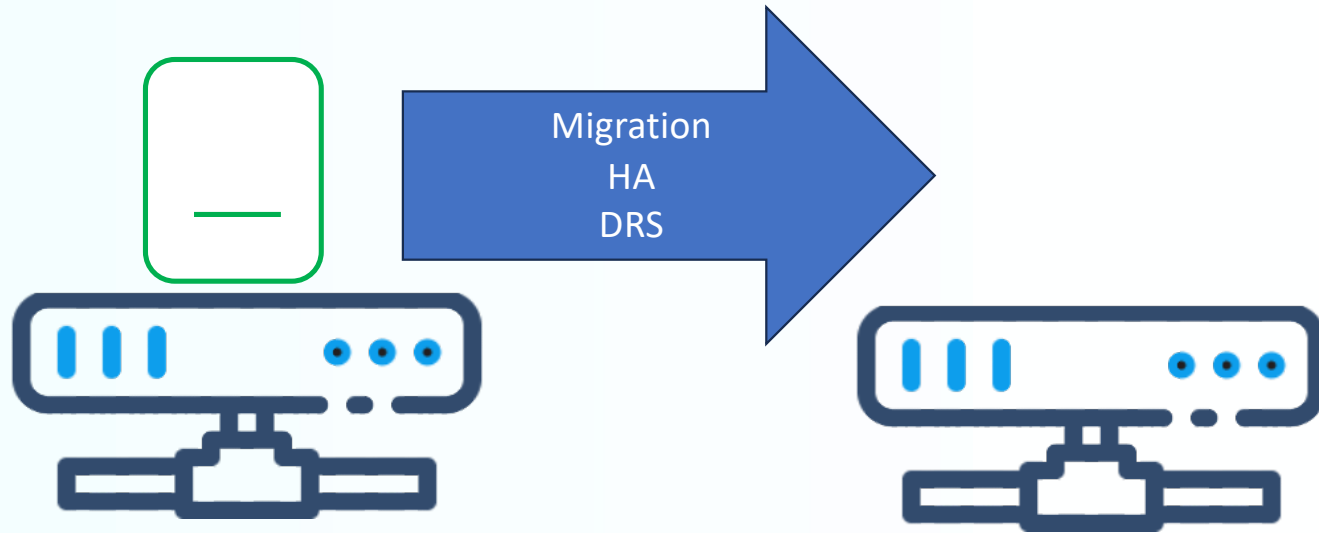
Finding a suitable deployment destination

Found Pod1,
Cluster CL2,
Host H2,
Primary
Storage PS3





This doesn't stop here!

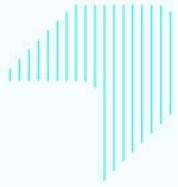




References



- https://docs.cloudstack.apache.org/en/latest/adminguide/vm_volume_allocators.html
- Introduce volume allocation algorithm global configuration #10696 (4.21)
- CPU to Memory weight based algorithm to order cluster #10997 (4.21)
- Consider Instance in Starting state as well for allocation algorithm #11751 (4.20.2)
- Consider Instance in Starting state for UserConcentratedPod #11845 (4.20.3)



Thank you!
Now it is time for
Questions 😊

Fill in the CloudStack User Survey

Help us understand the CloudStack Ecosystem

