




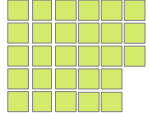
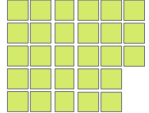

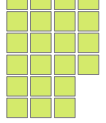
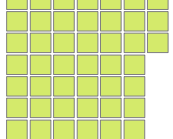
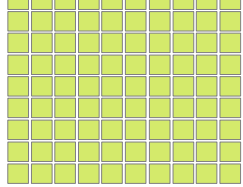
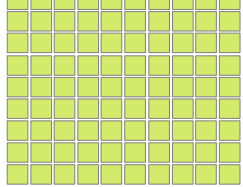

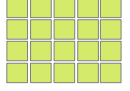
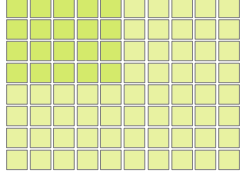
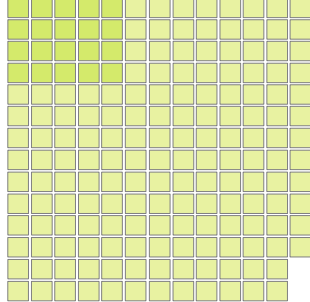
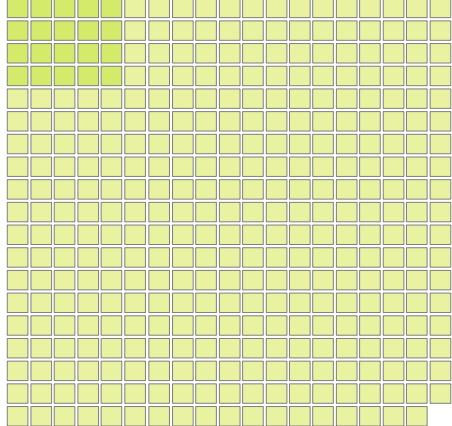
Aglais resource tests

D.Morris – Feb 2021

Resource tests to determine the resources available
on the Cumulus Openstack cloud platform.

Aglais resource tests – Feb 2021

Openstack virtual machines available in 5 flavors

	tiny	small	medium	large	xlarge
cpu cores per VM	2cpu 	6cpu 	14cpu 	28cpu 	28cpu 
memory per VM	6G 	22G 	45G 	90G 	90G 
local disc space per VM	12G 	20G 	20G + 60G 	20G + 160G 	20G + 340G 

The medium, large and xlarge flavors have the same 20G disc for the operating system plus an extra local disc for data.

Aglais resource tests – Feb 2021

Horizon dashboard:

Limit Summary

Compute



Instances

Used 6 of 20



VCPUs

Used 76 of 400



RAM

Used 247GB of 768GB

At first glance, this appears to show : 400 cpu and 768G memory per project
However, this is not consistent with results we see from using the system.
These tests were developed to determine what resources are actually available.

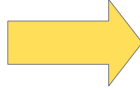
Test procedure:

For each VM flavor :

- Delete everything from all three projects
- Attempt to create the maximum number of VMs in each project
- Count how many were successfully created

Aglais resource tests – Feb 2021

Test #1 - 25 tiny VMs in each Openstack project

Openstack quota limits us to 20 VMs per project.
Create requests rejected once quota is reached.  20 VMs per project



Result : 60 ACTIVE tiny VMs in total

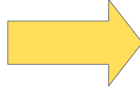
Tiny flavor has 2 vcpu cores, 6G of memory and 12G of disc space

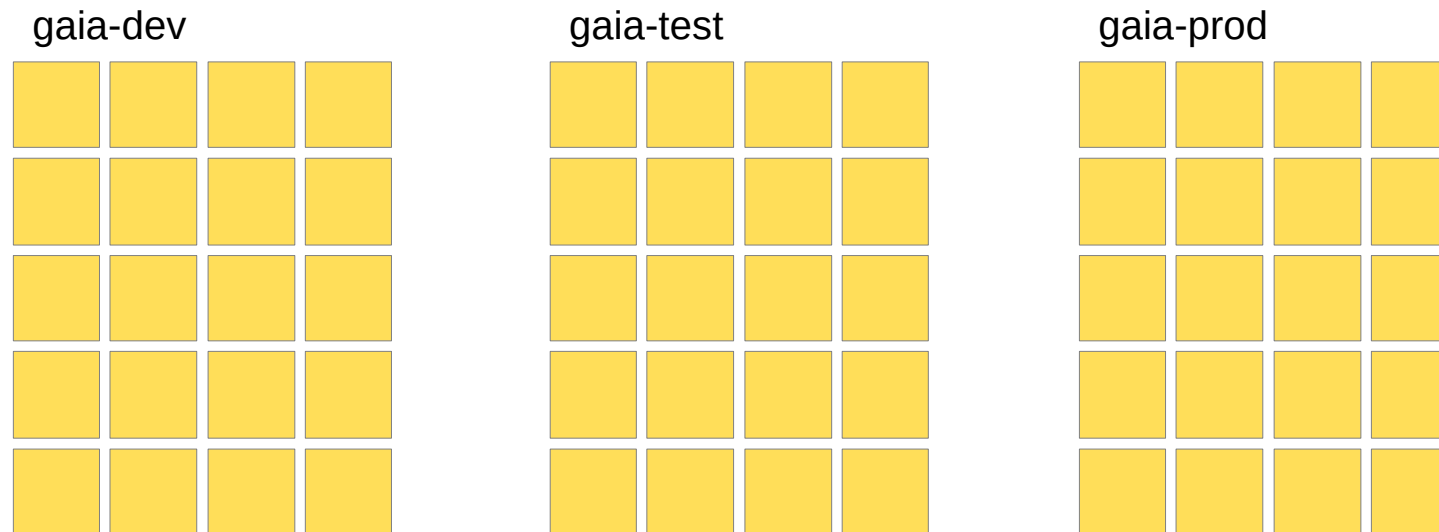
$$\begin{aligned} 60 * 2 &= 120 \text{ cpu cores} \\ 60 * 6 &= 360\text{G memory} \\ 60 * 12 &= 720\text{G local disc} \end{aligned}$$

(*) the 20 virtual machine quota is set by the system administrators for operational reasons.

Aglais resource tests – Feb 2021

Test #2 - 25 small VMs in each Openstack project

Openstack quota limits us to 20 VMs per project.
Create requests rejected once quota is reached.  20 VMs per project



Target : 60 small VMs in total

Small flavor has 6 vcpu cores, 22G of memory and 20G of disc space

$$\begin{aligned}
 60 * 6 &= 360 \text{ cpu cores} \\
 60 * 22 &= 1320\text{G memory} \\
 60 * 20 &= 1200\text{G local disc}
 \end{aligned}$$

Aglais resource tests – Feb 2021

Test #2 - 25 small VMs in each Openstack project

29 of the 60 VMs failed with '*No valid host found*'



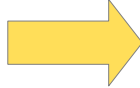
Result : 31 ACTIVE and 29 FAILED small VMs

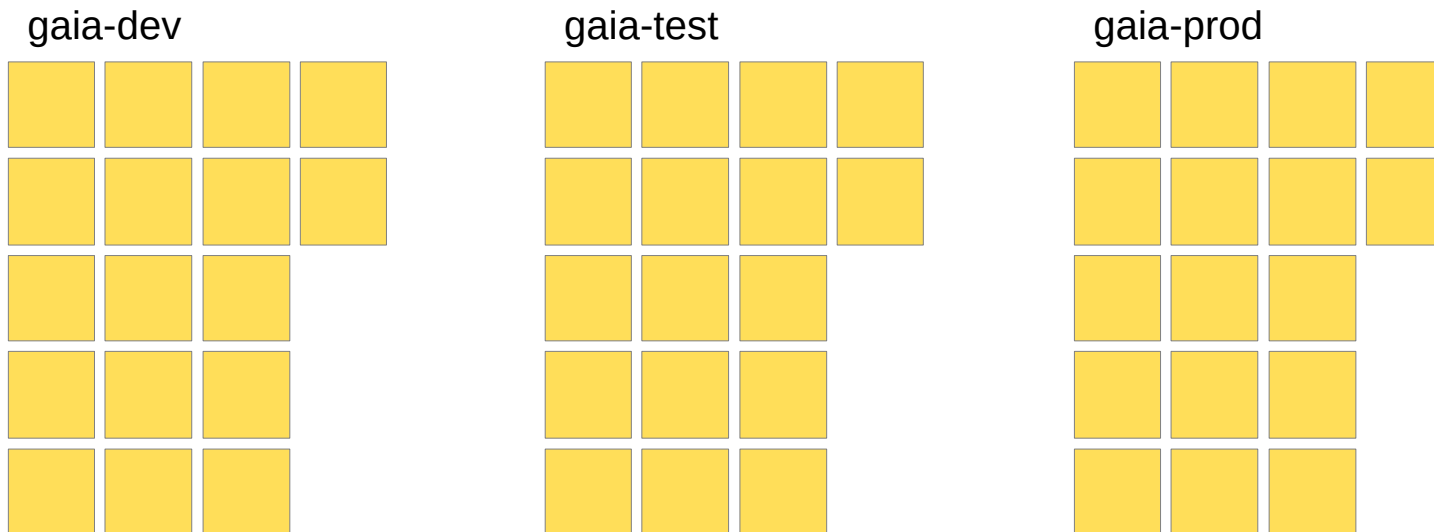
Small flavor has 6 vcpu cores, 22G of memory and 20G of disc space

$31 * 6 = 186$ cpu cores
 $31 * 22 = 682\text{G}$ memory
 $31 * 20 = 620\text{G}$ local disc

Aglais resource tests – Feb 2021

Test #3 - 25 medium VMs in each Openstack project

Quota limits us to 768G of memory per project.
Create requests rejected once quota is reached.  17 VMs per project



Target : 51 medium VMs in total

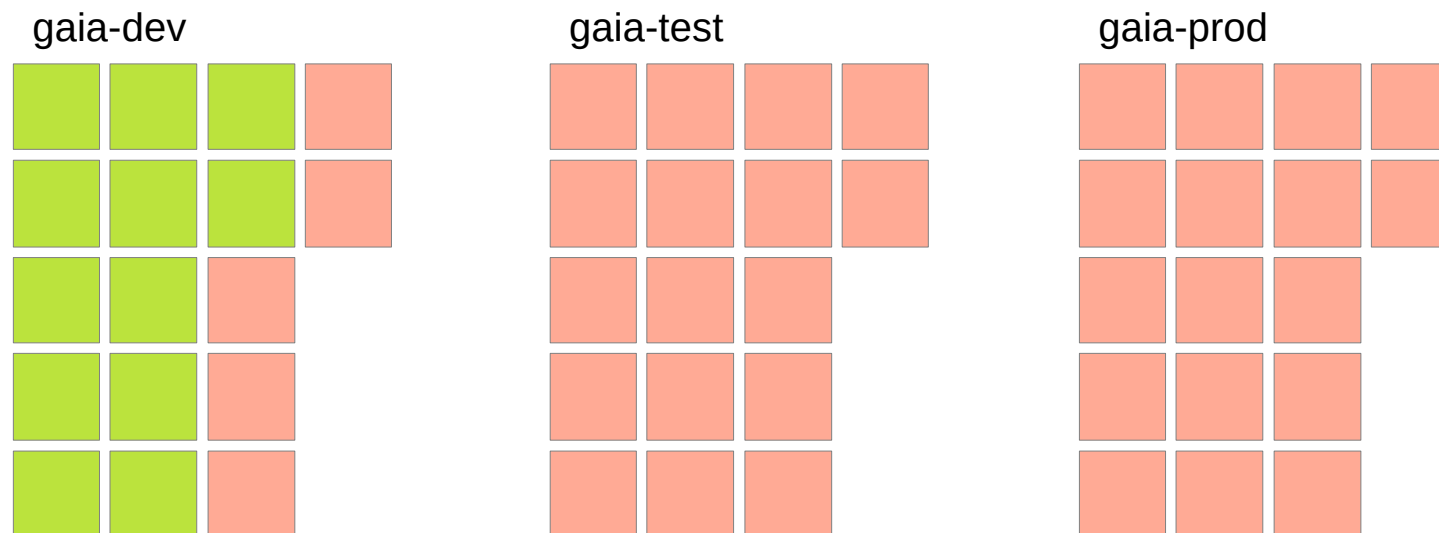
Medium flavor has 14 vcpu cores, 45G of memory and 80G of disc space

$$\begin{aligned}
 51 * 14 &= 714 \text{ cpu cores} \\
 51 * 45 &= 2295\text{G memory} \\
 51 * 80 &= 4080\text{G local disc}
 \end{aligned}$$

Aglais resource tests – Feb 2021

Test #3 - 25 medium VMs in each Openstack project

39 of the 51 VMs failed with '*No valid host found*'



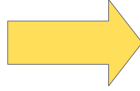
Result : 12 ACTIVE and 39 failed medium VMs

Medium flavor has 14 vcpu cores, 45G of memory and 80G of disc space

$12 * 14 = 168$ cpu cores
 $12 * 45 = 540$ G memory
 $12 * 80 = 960$ G local disc

Aglais resource tests – Feb 2021

Test #4 - 10 large VMs in each Openstack project

Quota limits us to 768G of memory per project.
Create requests rejected once quota is reached.  8 VMs per project



Target : 24 large VMs in total

Large flavor has 28 vcpu cores, 90G of memory and 180G of disc space

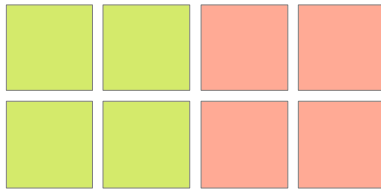
$$\begin{aligned} 24 * 28 &= 672 \text{ cpu cores} \\ 24 * 90 &= 2160\text{G memory} \\ 24 * 180 &= 4320\text{G local disc} \end{aligned}$$

Aglais resource tests – Feb 2021

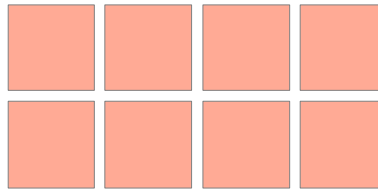
Test #4 - 10 large VMs in each Openstack project

20 of the 24 VMs failed with '*No valid host found*'

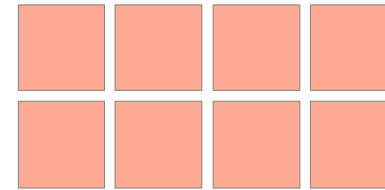
gaia-dev



gaia-test



gaia-prod



Result : 4 large VMs in total

Large flavor has 28 vcpu cores, 90G of memory and 180G of disc space

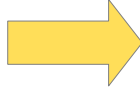
$$4 * 28 = 112 \text{ cpu cores}$$

$$4 * 90 = 360\text{G memory}$$

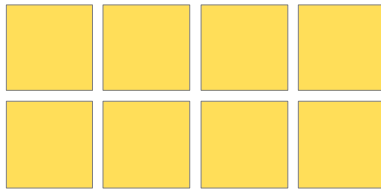
$$4 * 180 = 720\text{G local disc}$$

Aglais resource tests – Feb 2021

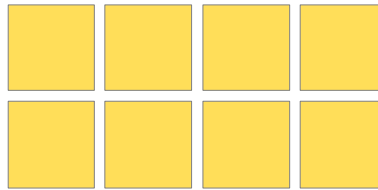
Test #5 - 5 eXtra-large VMs in each Openstack project

Quota limits us to 768G of memory per project.
Create requests rejected once quota is reached.  8 VMs per project

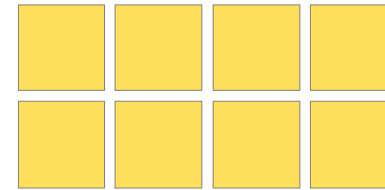
gaia-dev



gaia-test



gaia-prod



Target : 24 XLarge VMs in total

XLarge flavor has 28 vcpu cores, 90G of memory and 360G of disc space

$24 * 28 = 672$ cpu cores

$24 * 90 = 2160$ G memory

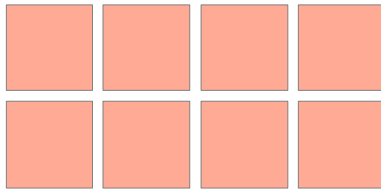
$24 * 360 = 8640$ G local disc

Aglais resource tests – Feb 2021

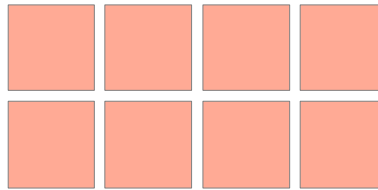
Test #5 - 5 eXtra-large VMs in each Openstack project

All 24 VMs failed with *'No valid host found'*

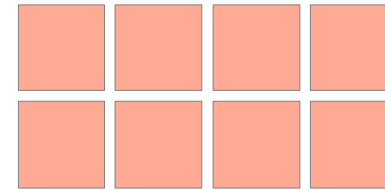
gaia-dev



gaia-test



gaia-prod



None of the physical hosts was able to accommodate the 360G local disc needed for a XL machine.

Conclusion : available disc space < 360

Aglais resource tests – Feb 2021

In order to reserve resources for our project, our allocation has been '*pinned*' to four physical machines.

These resources are reserved for our use.

When resources are in high demand released resources may be allocated to other projects.

If we delete and re-create a 10 VM cluster, some of those resources can get allocated to another project in the gap between the delete and create commands.

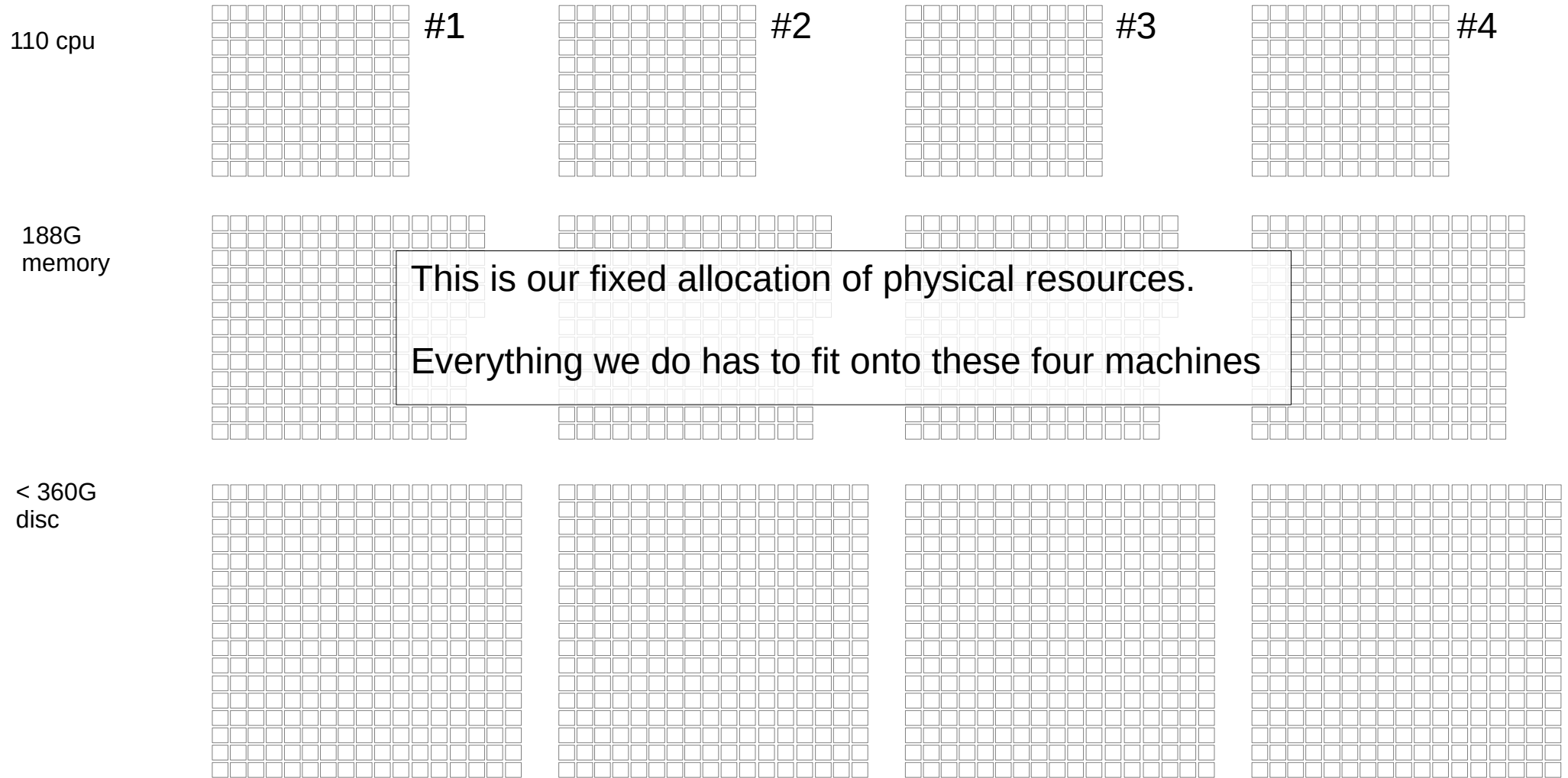
Pinning prevents that from happening.

It also means we can't expand to use other physical machines.

Pinning works both ways – other projects can't use our resources, but we can't use any other resources either.

Aglais resource tests – Feb 2021

Gaia allocation pinned to 4 physical machines

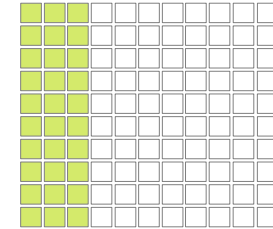
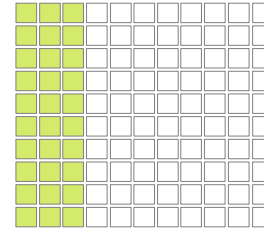
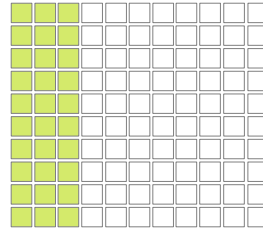
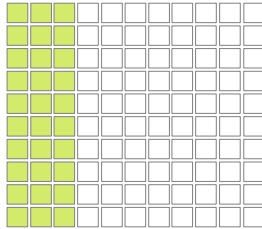




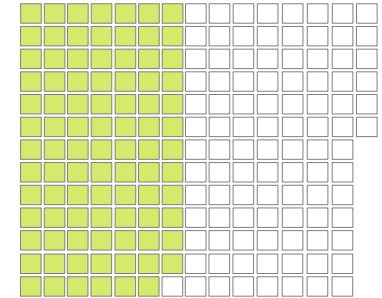
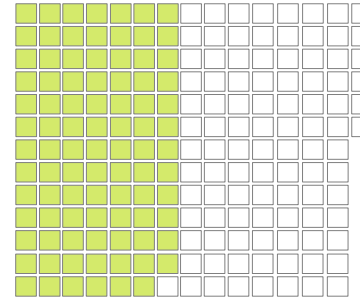
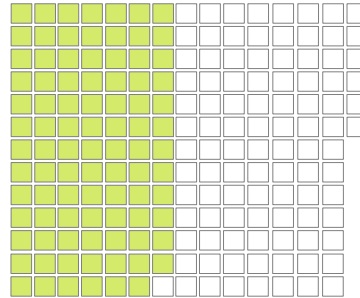
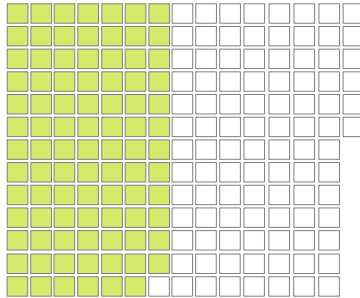
Aglais resource tests – Feb 2021

60 tiny VMs, 15 per physical host – limited by 20 VM quota

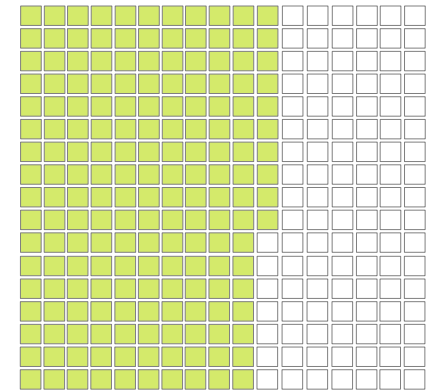
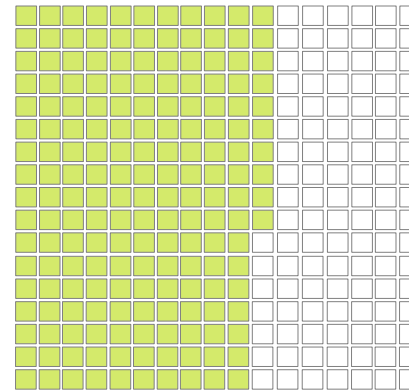
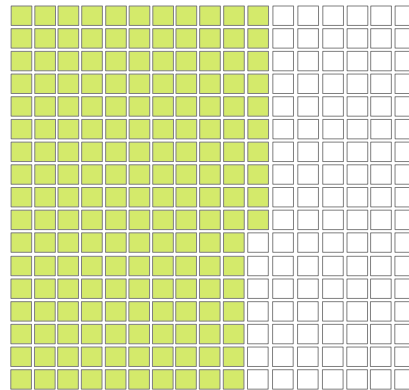
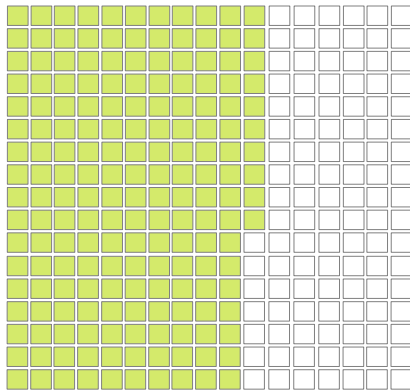
$15 * 2 = 30/110$
cpu



$15 * 6 = 90/188$
memory



$15 * 12 = 180/360$
disc

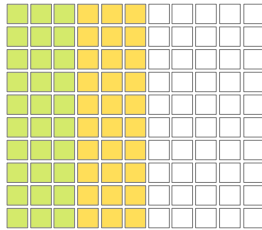




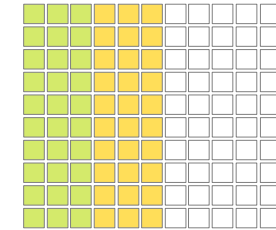
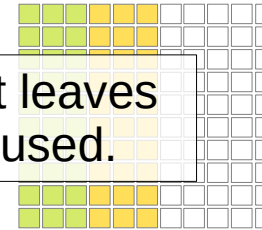
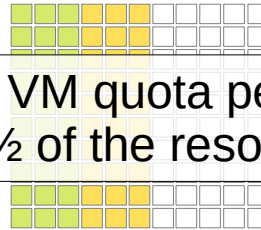
Aglaia resource tests – Feb 2021

120 tiny VMs, 30 per physical host (double the VM quota)

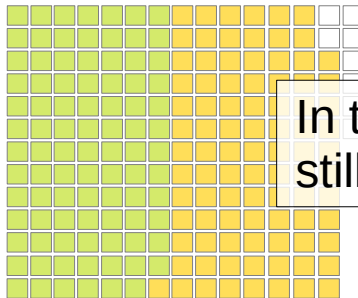
$30 * 2 = 60/110$
cpu



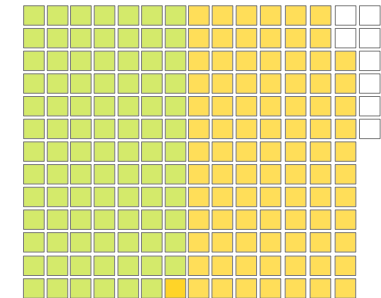
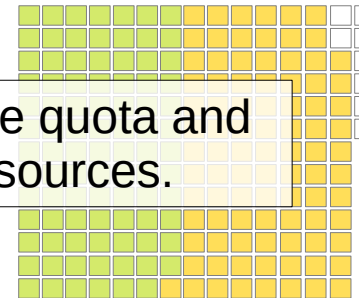
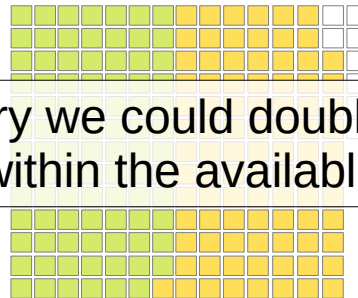
20 VM quota per project leaves
> ½ of the resources unused.



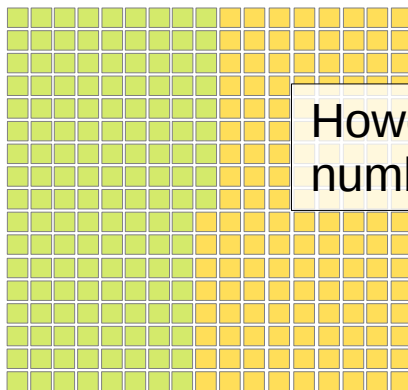
$30 * 6 = 180/188$
memory



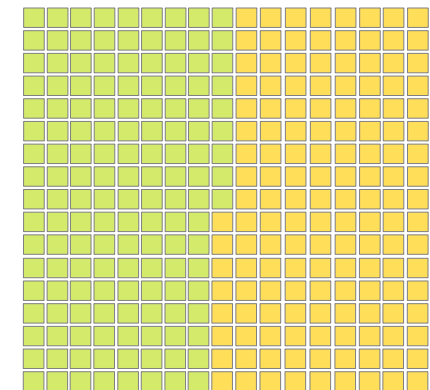
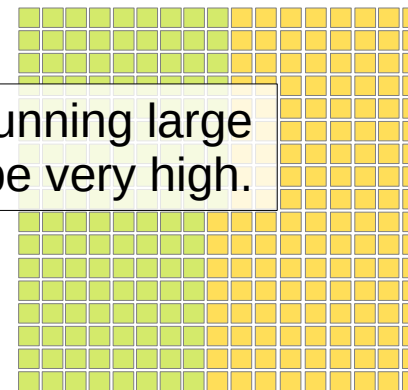
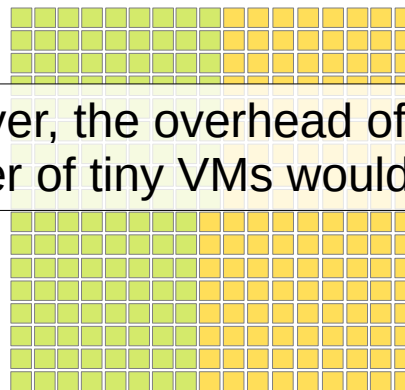
In theory we could double the quota and
still fit within the available resources.



$30 * 12 = 360/360$
disc



However, the overhead of running large
number of tiny VMs would be very high.





Aglais resource tests – Feb 2021

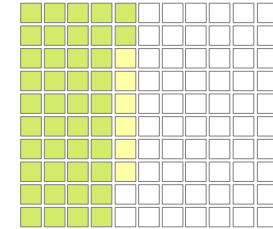
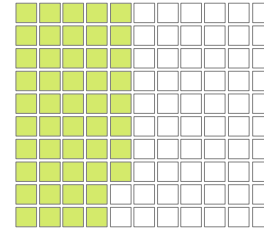
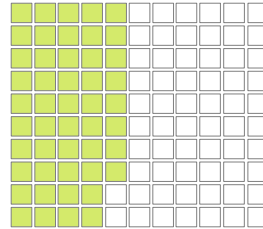
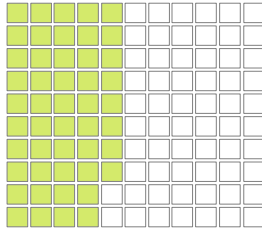
31 small VMs, 8 per physical host – limited by memory

(*) 1 host only has 7

$$8 * 6 = 48/110$$

cpu

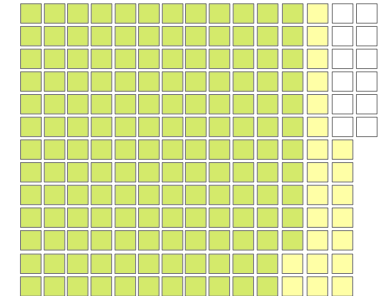
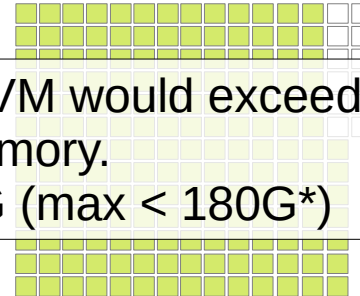
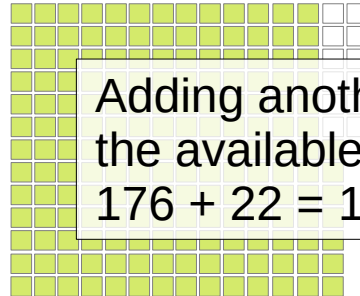
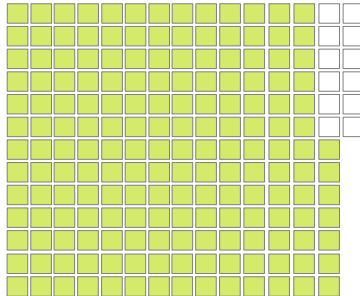
$$+1 = 54/110$$



$$8 * 22 = 176/188$$

memory

$$+1 = 198/188$$

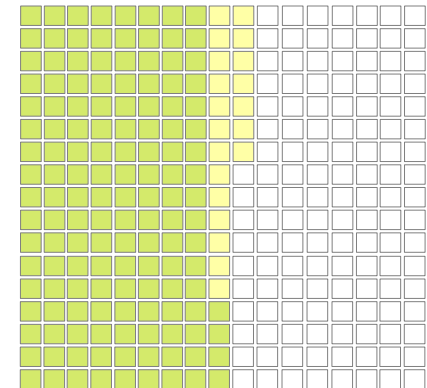
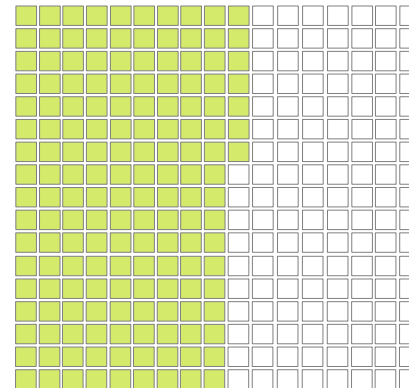
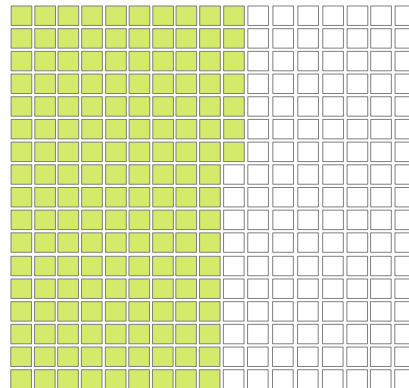
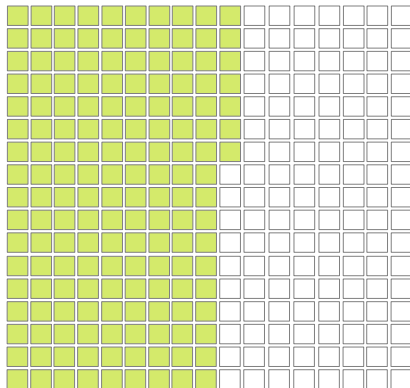


Adding another VM would exceed
the available memory.
 $176 + 22 = 198\text{G}$ (max < 180G*)

$$8 * 20 = 160/360$$

disc

$$+1 = 180/360$$





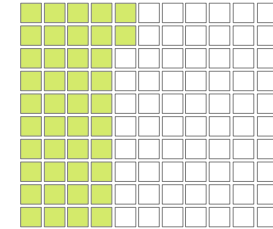
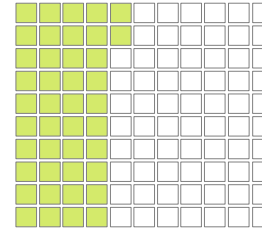
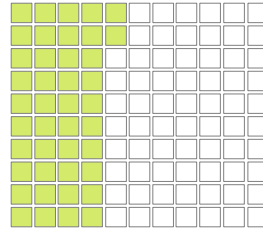
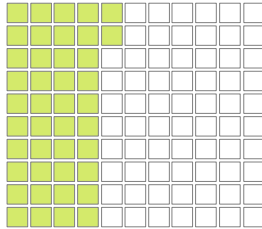
Aglais resource tests – Feb 2021

12 medium VMs, 3 per physical host – limited by memory or disc

$$3 * 14 = 42/110$$

cpu

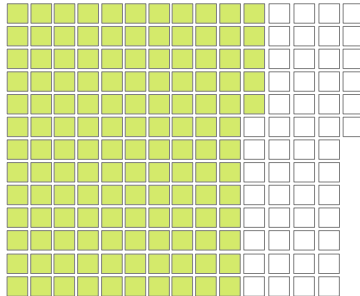
$$+1 = 56/110$$



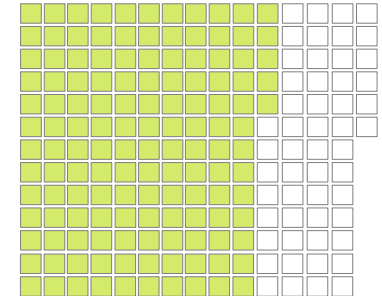
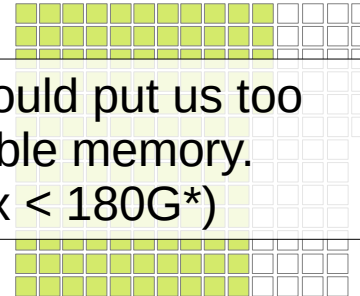
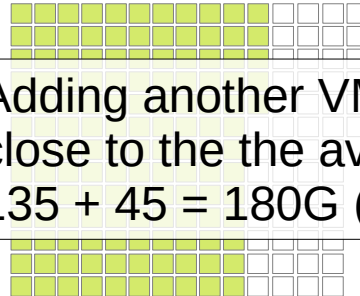
$$3 * 45 = 135/188$$

memory

$$+1 = 180/188$$



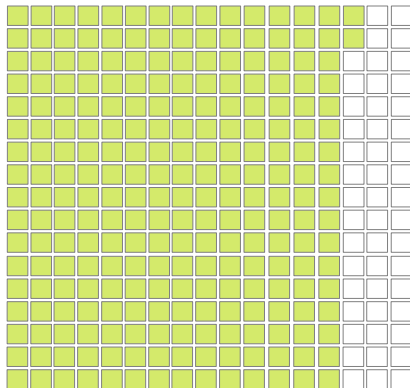
Adding another VM would put us too close to the the available memory.
 $135 + 45 = 180\text{G}$ (max < 180G*)



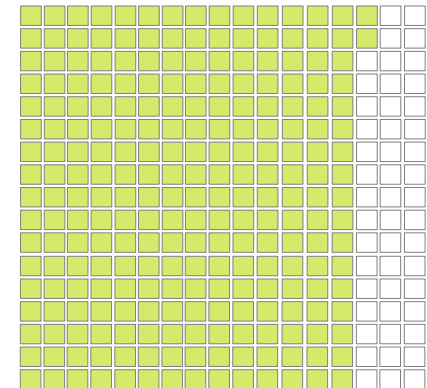
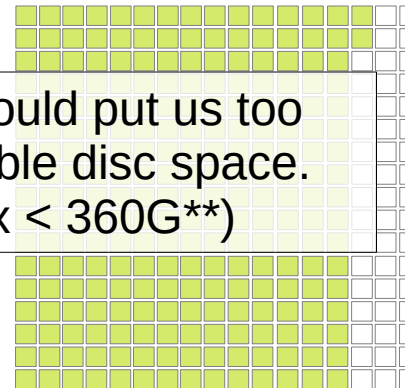
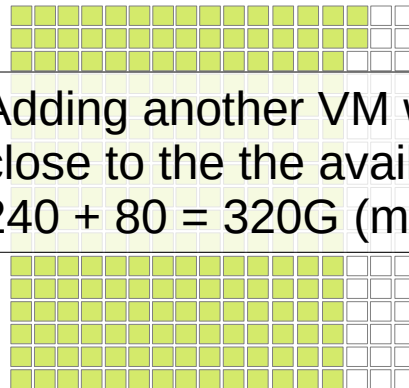
$$3 * 80 = 240/360$$

disc

$$+1 = 320/360$$



Adding another VM would put us too close to the the available disc space.
 $240 + 80 = 320\text{G}$ (max < 360G**)



(*) each physical host only has 188G of memory and Openstack reserves ~8G? for the system

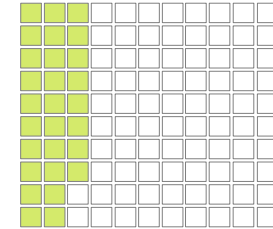
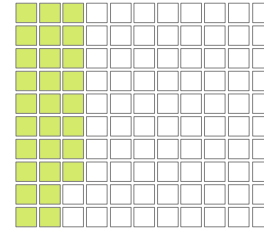
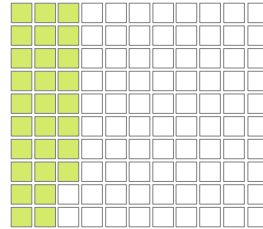
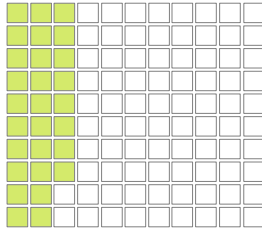
(**) the 360G limit on local disc is inferred from the xlarge test



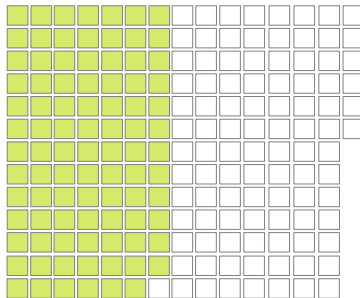
Aglais resource tests – Feb 2021

4 large VMs, 1 per physical host – limited by memory and disc

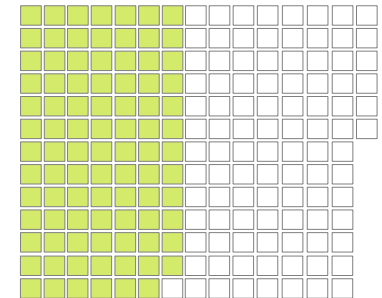
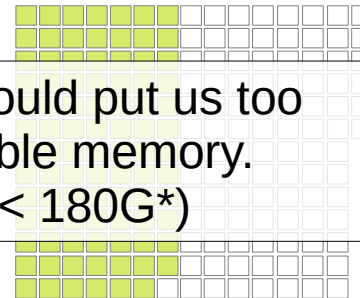
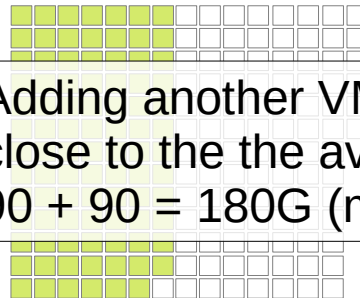
28/110
cpu
+1 = 56/110



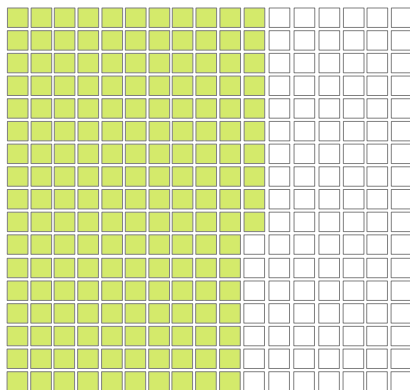
90/188
memory
+1 = 180/188



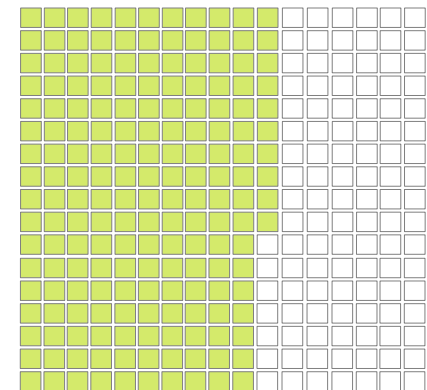
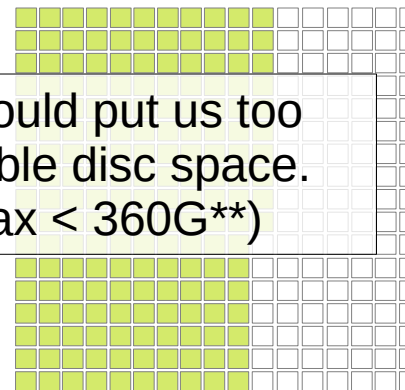
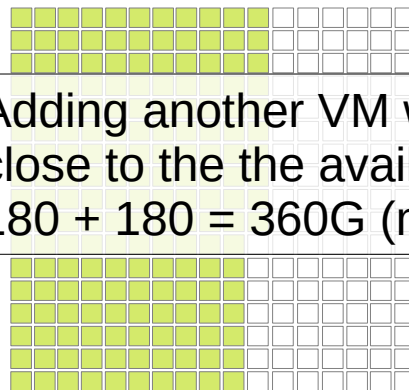
Adding another VM would put us too close to the the available memory.
 $90 + 90 = 180\text{G}$ (max < 180G*)



180/360
disc
+1 = 360/360



Adding another VM would put us too close to the the available disc space.
 $180 + 180 = 360\text{G}$ (max < 360G**)



Aglais resource tests – Feb 2021

Horizon dashboard:

Limit Summary

Compute



Instances

Used 6 of 20



VCPUs

Used 76 of 400



RAM

Used 247GB of 768GB

Horizon appears to show : 400 cpu, 768G memory **per project**

What we physically have : 440 cpu, 752G memory and 1440 local disc **in total**

Maximum we can actually use: 186 cpu, 682G memory and 960 local disc

Resource request for 2021 was 18500 CPU-months

$18500/12 \approx 1540$ cpu cores

Assuming 3G memory per core ≈ 4620 G memory

(*) section 5.1 quotes minimum deployment as 6 nodes with 16 cores and 64G memory

(*) nothing in the resource request about dev, test and prod projects

Aglais resource tests – Feb 2021

For comparison :

DPAC Tech Note on *“Efficient cross-matching in Spark”* by Enrique Utrilla
GAIA-C9-TN-ESAC-EUM-100

- *“A dedicated Apache Spark 2.4 cluster over 30 physical nodes in the Gaia cluster at ESAC, with NETApp storage.”*
- *“By default each user session is assigned a maximum of 80 CPU cores.”*

Our current live deployment

- Spark 2.4 cluster running in 4 medium virtual machines.
- Maximum of 56 cpu cores for the whole cluster.

- NetApp is a commercial cloud storage provider offering specialized hardware for cloud and on-premises storage systems (most which are SSD based).

Aglais resource tests – Feb 2021

Options to explore :

Work with StackHPC & Cambridge to increase the available resources

- Cambridge are limited by the resources they have.
- New resources in December were a welcome increase. It solved the immediate problem, but isn't enough to meet our 2021 allocation.
- What can we do / who can we contact to help get more resources from IRIS allocated to the Cambridge system ?
- Propose a series of gradual increments to get from where we are to our full 2021 and 2022 allocations.
- If we know what equipment is arriving when, then both ourselves and Cambridge can plan ahead.

Aglais resource tests – Feb 2021

Options to explore :

Work with StackHPC & Cambridge to optimize the available resources

- Work with StackHPC & Cambridge to increase the available disc space.
- Can we change the way the local discs are partitioned ?
- How much difference would an extra SSD per machine make ?
- Work with StackHPC & Cambridge to negotiate access to monitoring data from the physical platform. If resources are scarce we need to know how much impact our design choices have.
- Openstack is designed for a commercial setting, with a strong barrier between what users and administrators can see.
- That doesn't work so well on a system that is low on resources runs into contention issues between concurrent jobs.
- Reality is we are struggling to figure out where the bottlenecks are by making guesses based on screen shots of performance metrics

Aglais resource tests – Feb 2021

Options to explore :

Continue to work on making our deployments portable

- Both the Ansible and Kubernetes deployments could be moved to another platform
 - RAL or Somerville Openstack platforms may have more resources
 - Both would need Rancher deployment for Kubernetes (#386)
 - Both would need Echo S3 storage for data (#246)
 - We have physical machines available at ROE suitable for a Spark platform
 - The two Gaia machines have 96 cores and 250G memory each
 - The four LSST machines have xx cores and yyy memory each
 - Combined they would create a reasonable bare metal deployment
 - Our existing Ansible deployment could be adapted to run on these machines.
 - Our Rancher K8s deployment could be adapted to run on these machines.
- Commercial cloud platforms have more resources
 - On-deploy deployment on commercial platform, create, analyse, delete
 - Commercial cloud would need S3 storage for data (#246)