

# Autopilot: workload autoscaling at Google

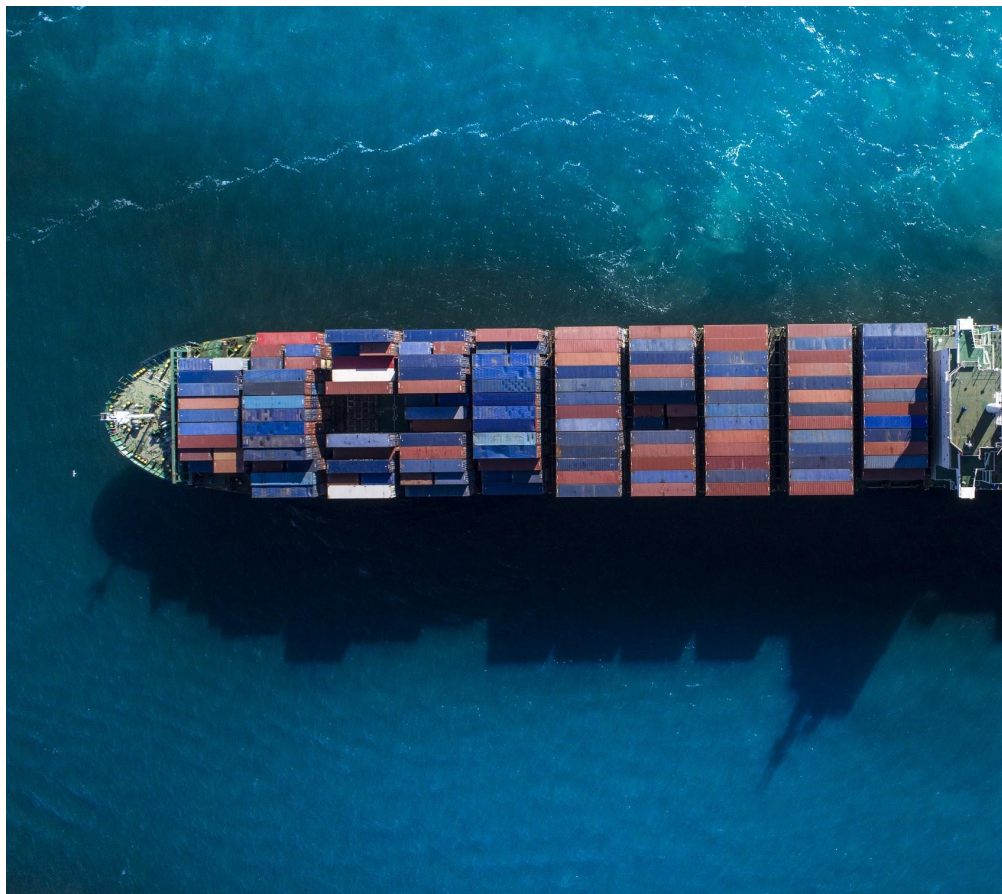
Krzysztof Rządca (Google & University of Warsaw, Poland), Paweł Findeisen, Jacek Świdorski, Przemysław Zych, Przemek Broniek, Jarek Kuśmierek, Paweł Nowak, Beata Strack, Piotr Witusowski, Steven Hand, John Wilkes (Google)

*EuroSys 2020*

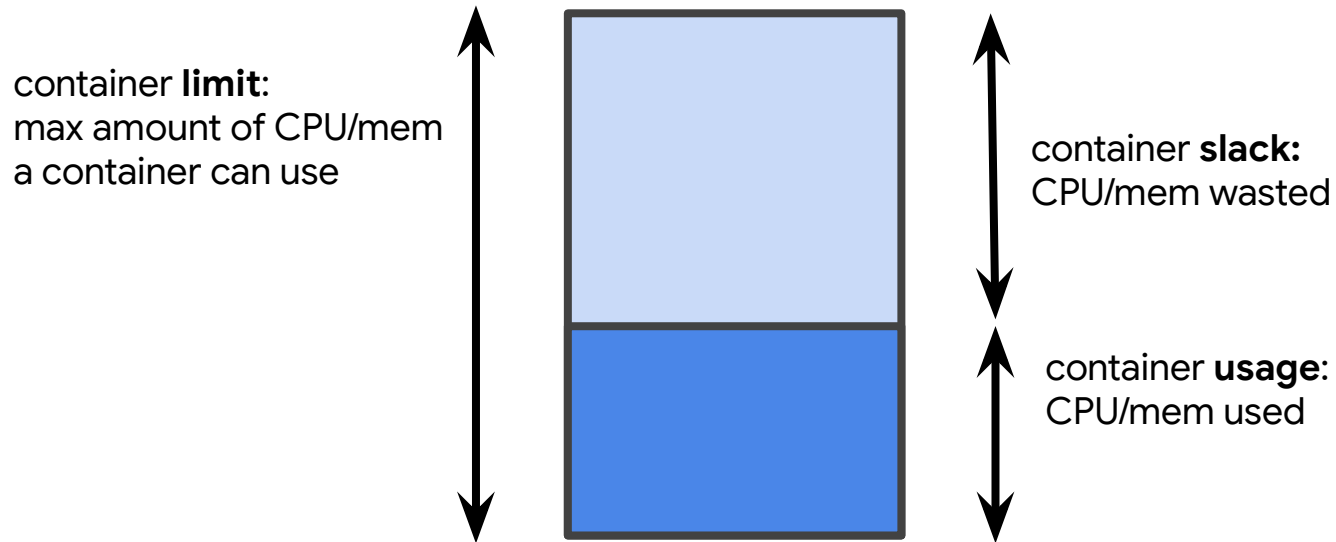
*April 2020*

## Google runs in containers

In any given week, we  
launch over two billion  
containers across  
Google.



## Resource limits are crucial to isolate workloads



Borg, our scheduler,  
packs containers to  
machines by resource  
limits.

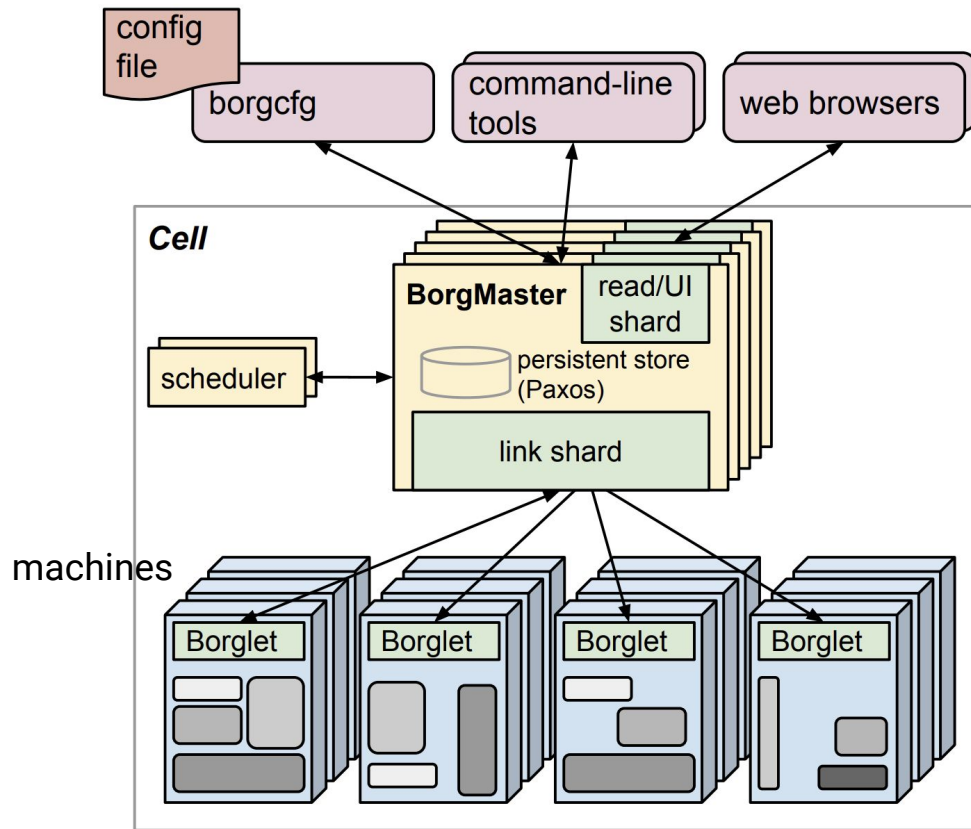
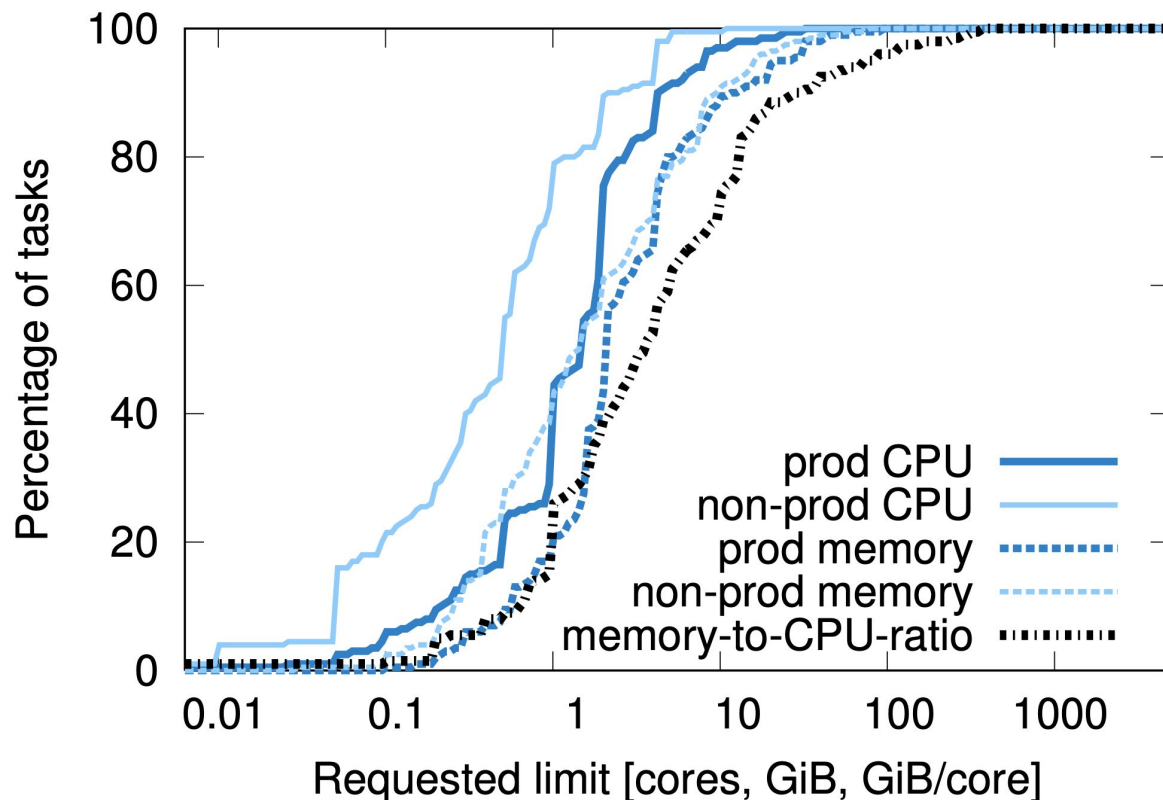


image source: <http://dx.doi.org/10.1145/2741948.2741964> [Verma et al., EuroSys'15]

Limits are fine-grained:  
CPU in milli-cores  
memory in bytes

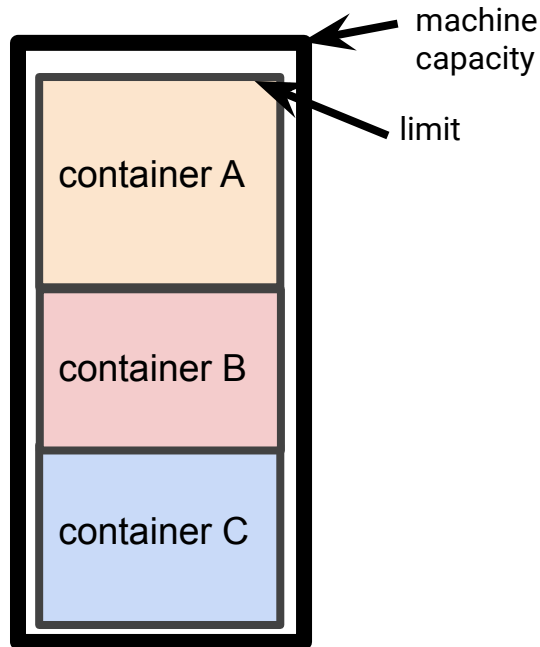


Source: <http://dx.doi.org/10.1145/2741948.2741964> [Verma et al., EuroSys'15]

# We pack containers to machines by limits.

Proprietary + Confidential

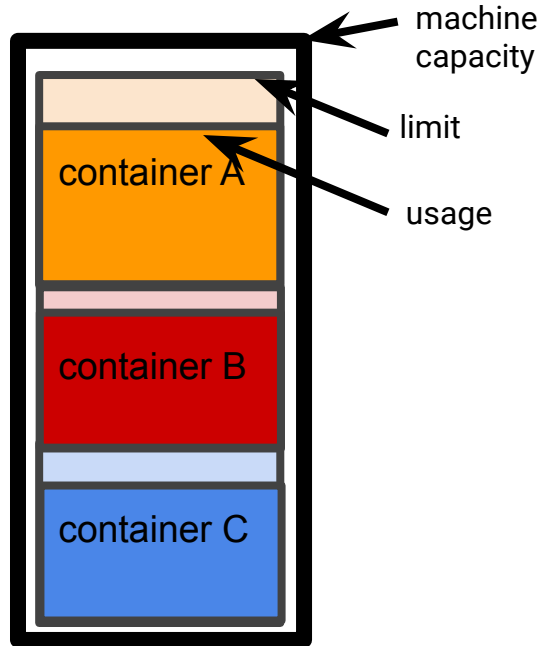
## So, precise limits are crucial for efficiency and reliability.



We pack containers to machines by limits.

Proprietary + Confidential

So, precise limits are crucial for efficiency and reliability.

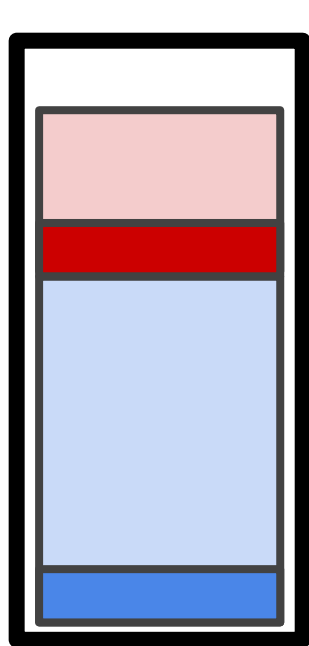


precise  
limits

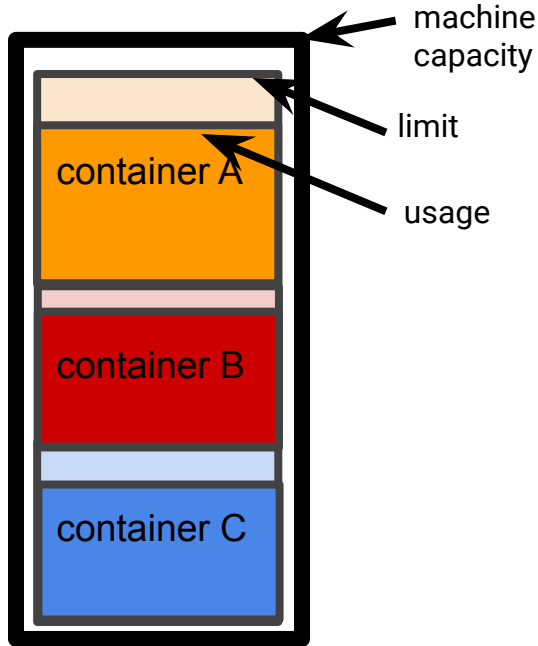
**good!**

We pack containers to machines by limits.

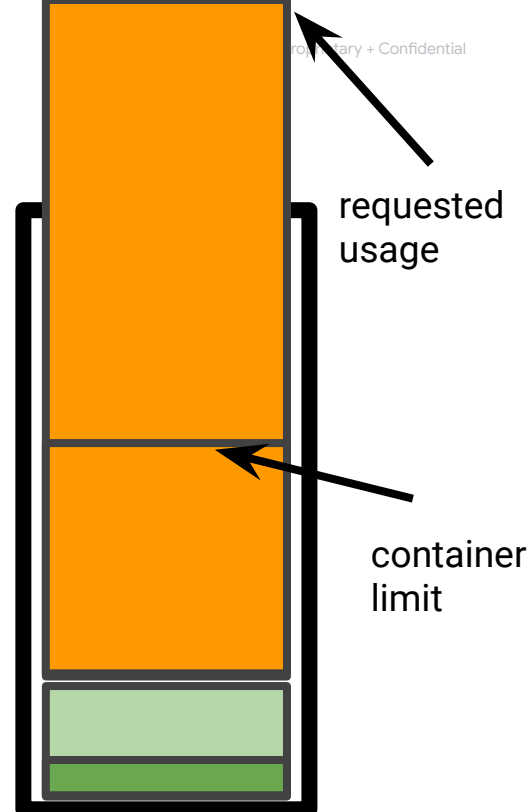
So, precise limits are crucial for efficiency and reliability.



resource are  
wasted  
(underallocated  
machine)  
**bad!**



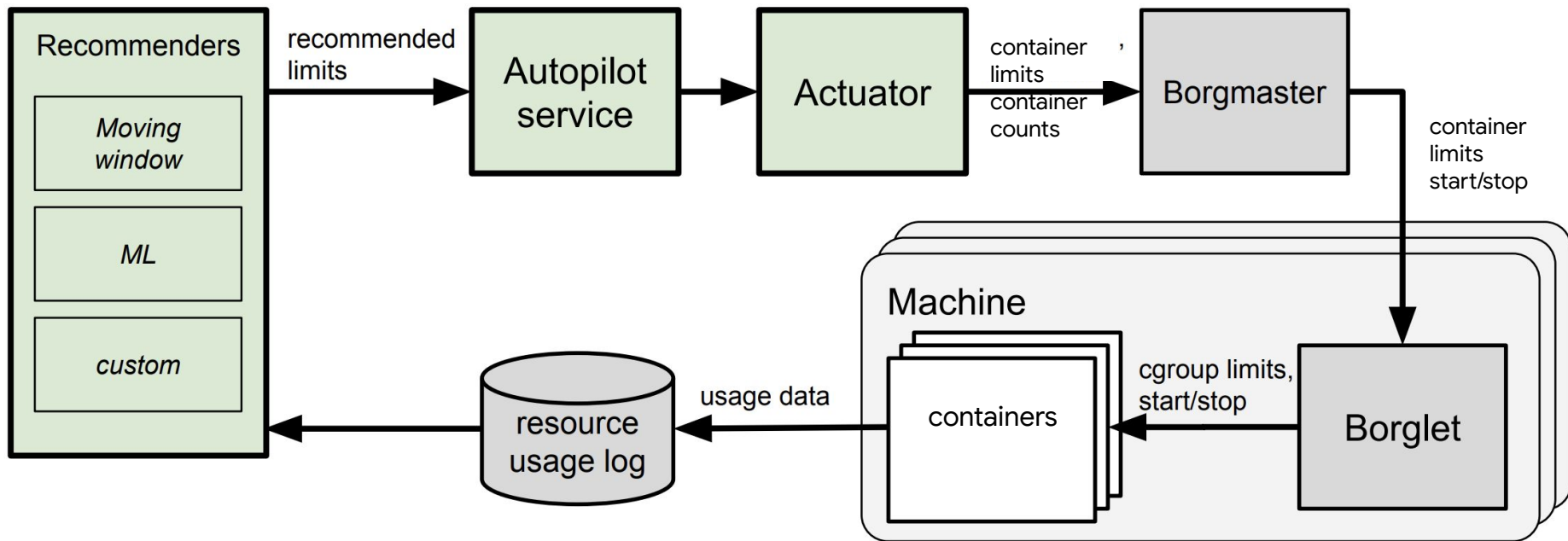
precise  
limits  
**good!**



out-of-resources  
crash  
**bad!**



## Autopilot acts as a controller for Borg limits.

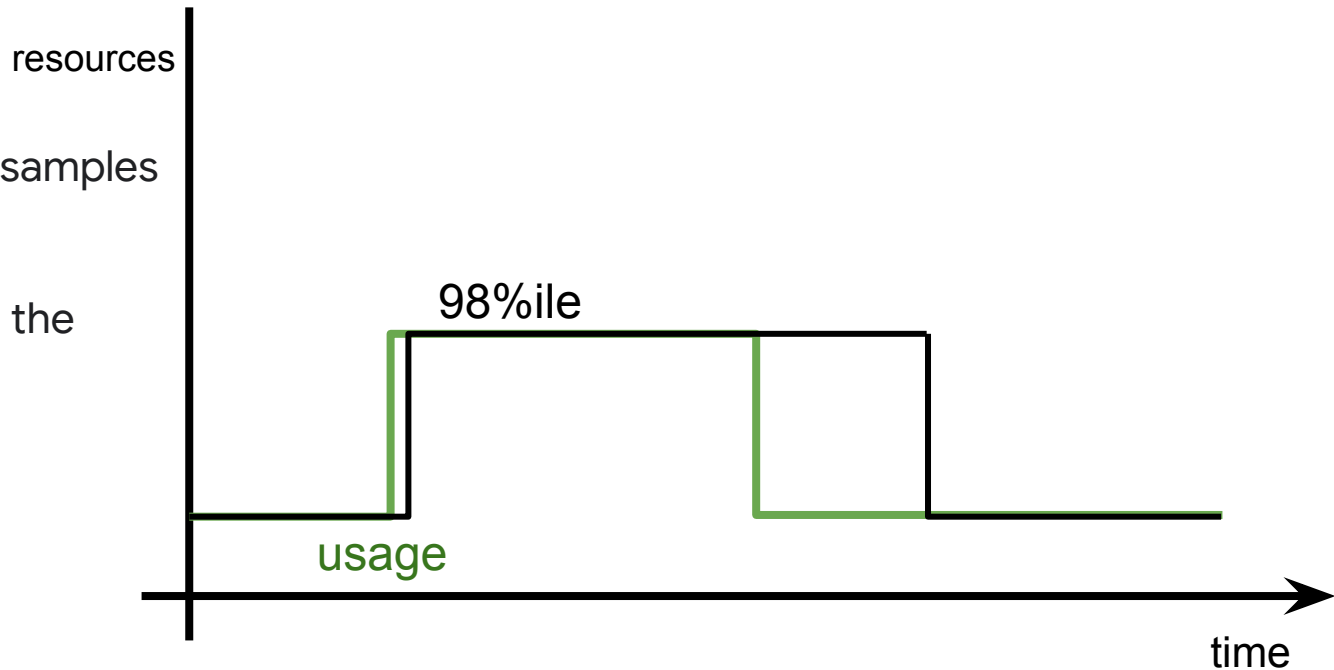


Autopilot continuously adjusts resource limits:  
CPU/Mem limits for containers (vertical scaling),  
number of replicas (horizontal scaling).

# Autopilot Recommenders

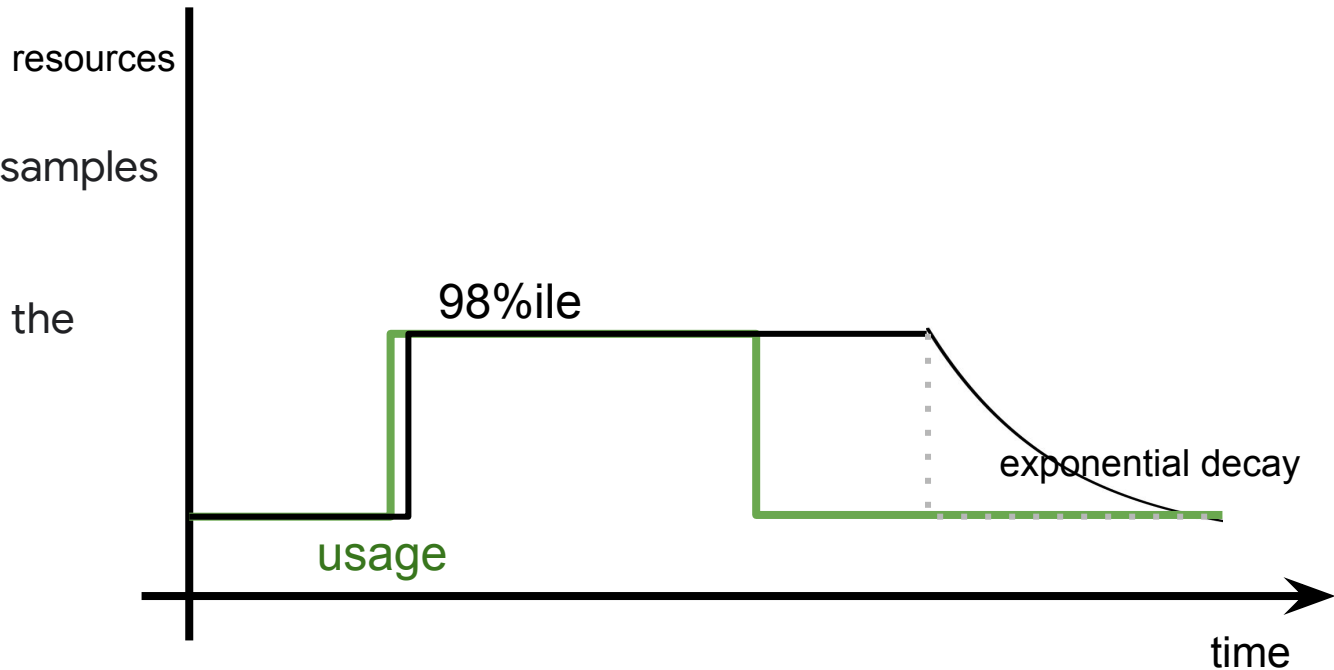
## Moving window recommenders

- Exponentially-decaying samples (half-life of 48 hours)
- Compute statistics over the samples, e.g. 95%ile
- add a safety margin



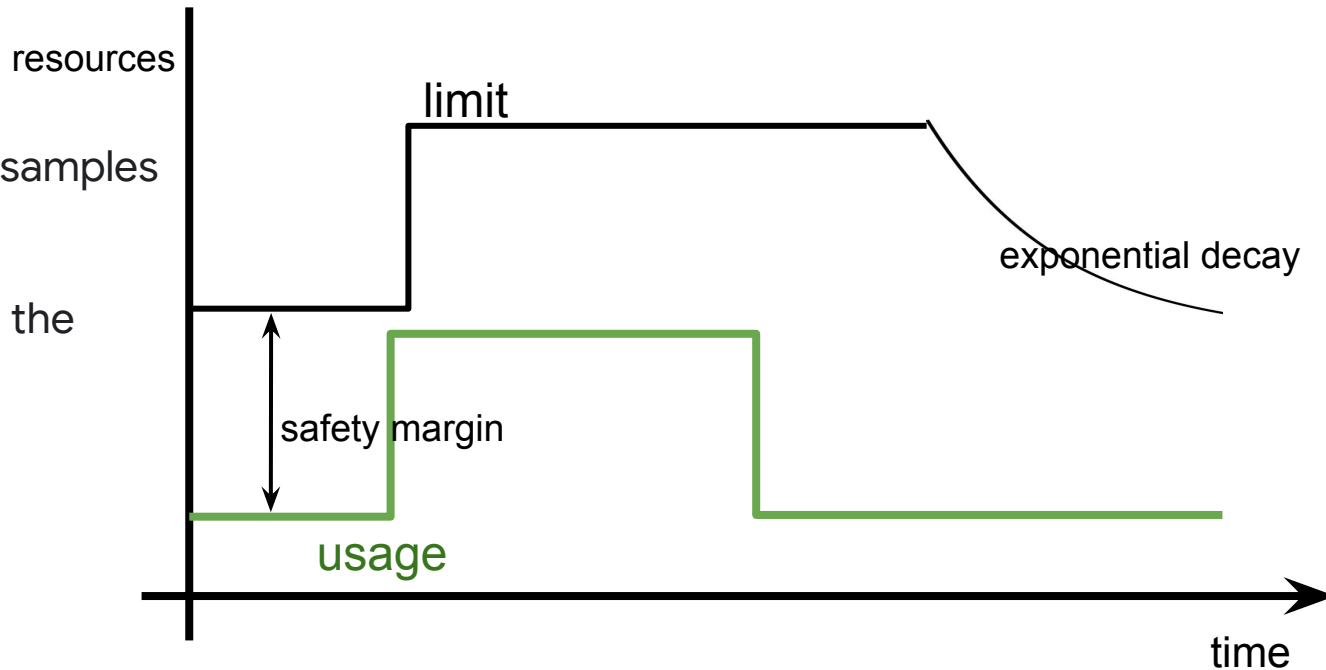
## Moving window recommenders

- Exponentially-decaying samples (half-life of 48 hours)
- Compute statistics over the samples, e.g. 95%ile
- add a safety margin



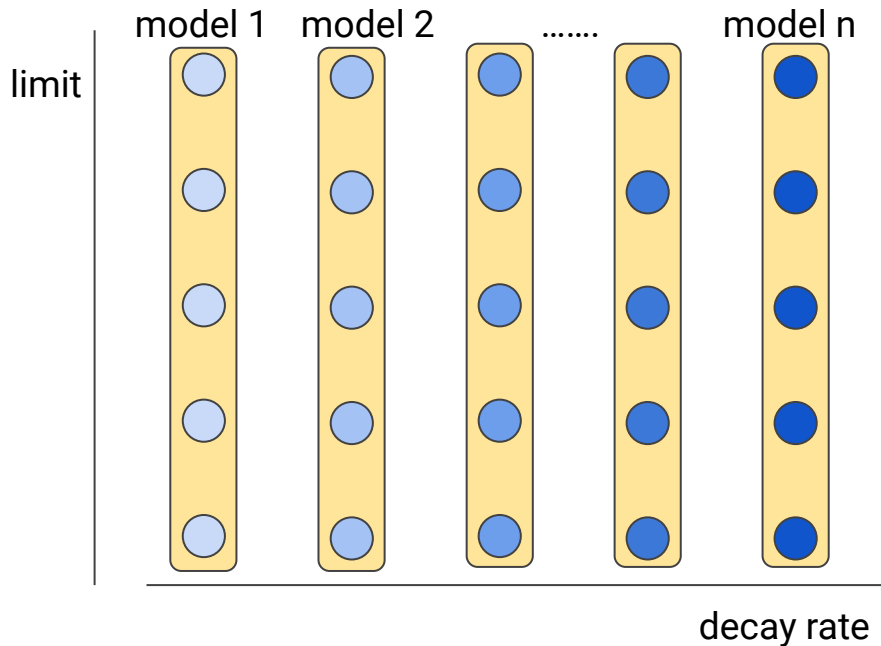
## Moving window recommenders

- Exponentially-decaying samples (half-life of 48 hours)
- Compute statistics over the samples, e.g. 95%ile
- add a safety margin



# Machine learning recommenders

- Each model is an arg-max algorithm picking a limit value
- Each model is parametrized by the decay rate and the safety margin.
- The recommender picks the model performing the best over a longer time period.



Evaluation:

Observational study of production jobs

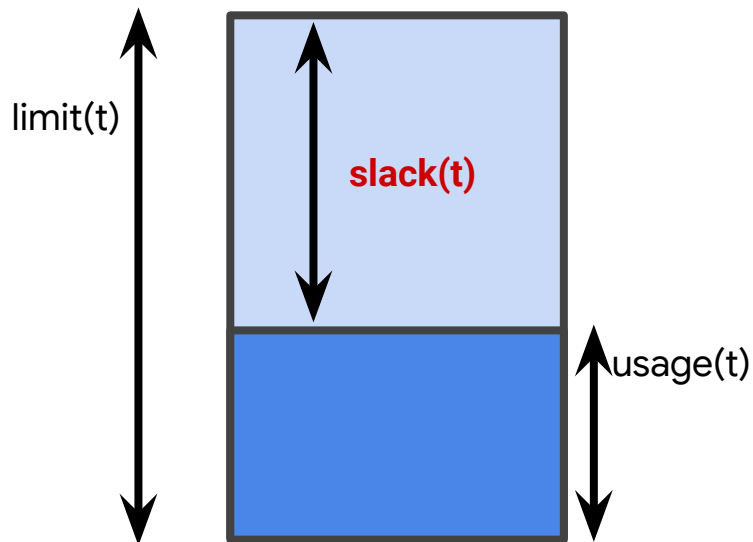
Focus on memory

## Autopilot efficiency - reduction of slack

**absolute slack:**

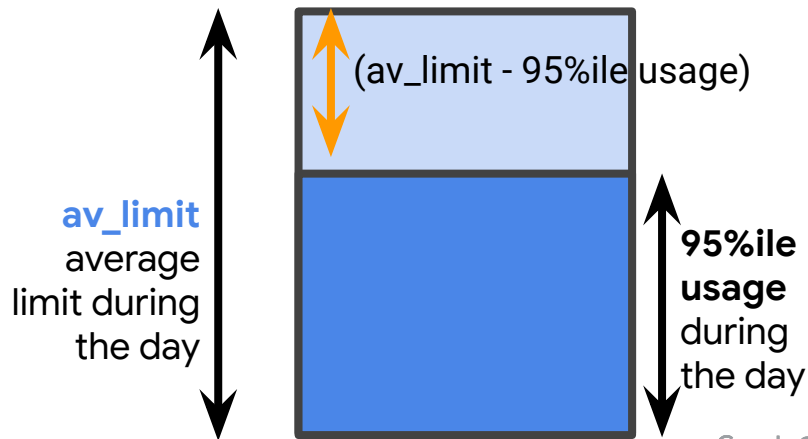
$$\int \text{slack}(t) dt = \int \text{limit}(t) dt - \int \text{usage}(t) dt$$

unit: capacity of a single (largish)  
machine



**relative slack:**

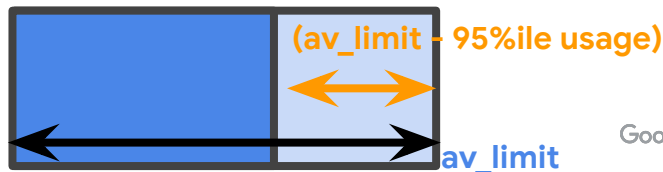
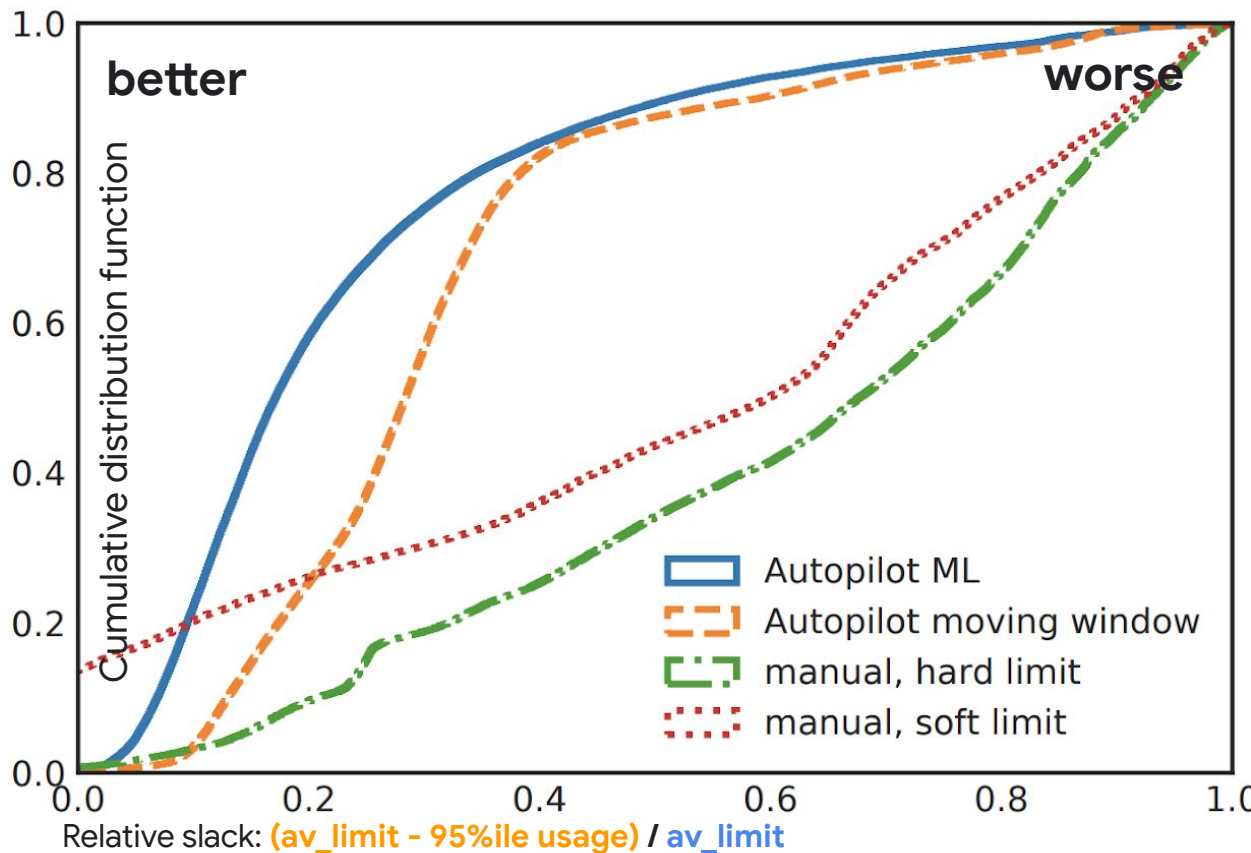
$$(\text{av\_limit} - 95\%ile \text{ usage}) / (\text{av\_limit})$$





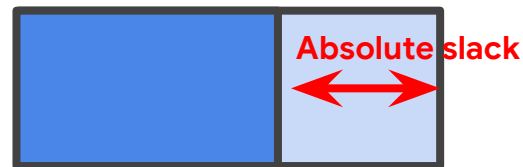
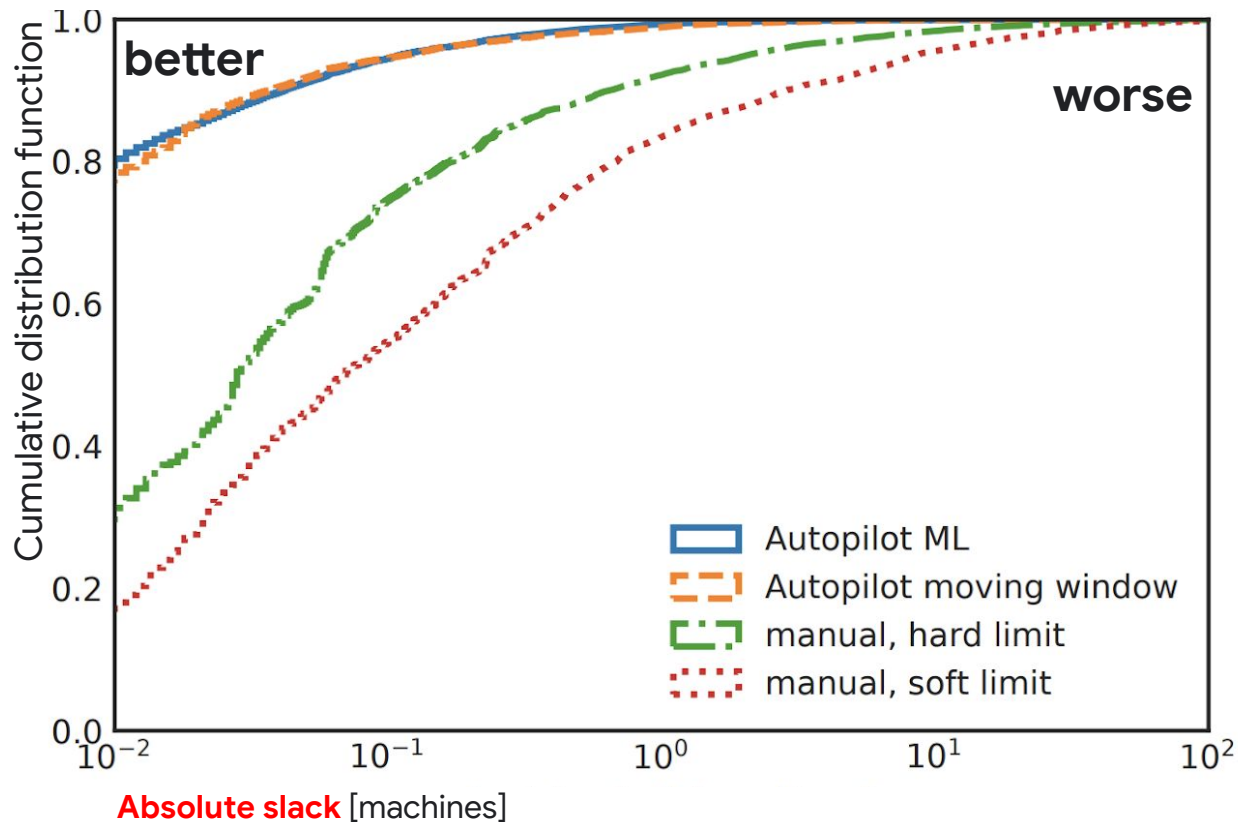
Autopiloted jobs have significantly smaller relative slack.

A random sample of 5000 jobs in each category.



## Autopiloted jobs save significant capacity.

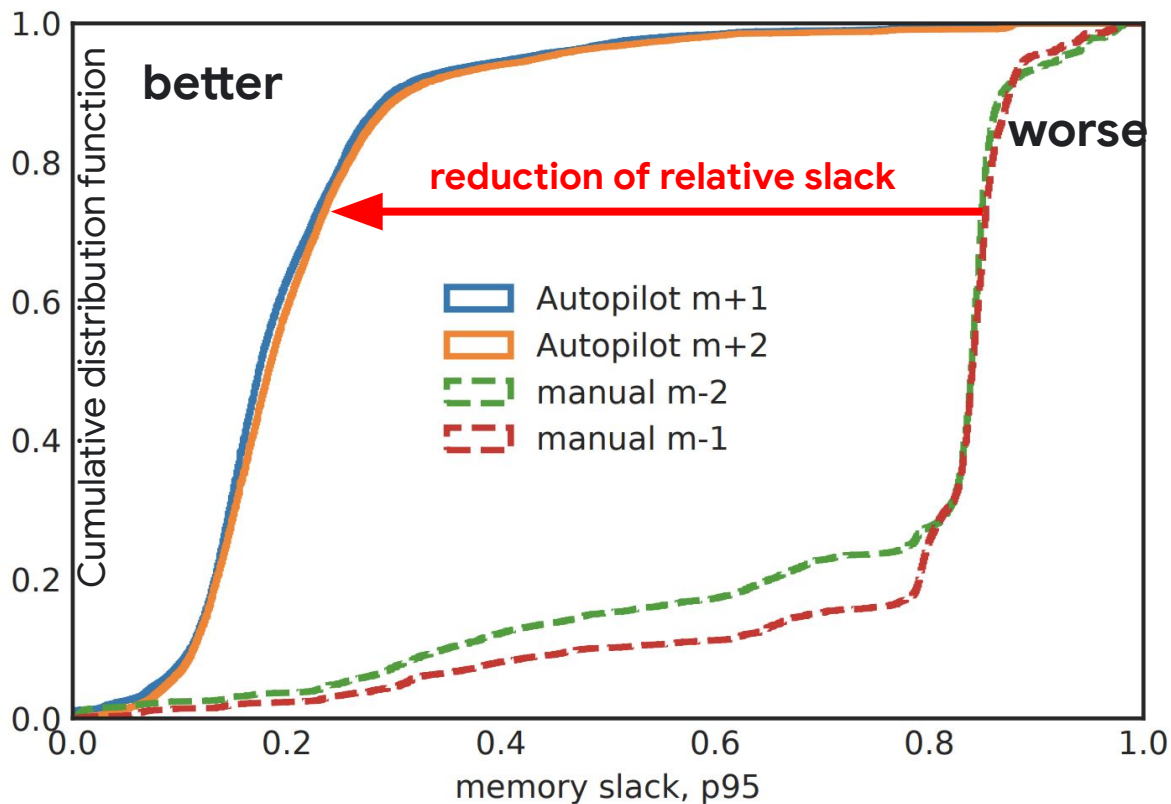
A random sample of 5000 jobs in each category.



When jobs migrate to Autopilot, their slack is significantly reduced.

A random sample of 500 jobs that migrated to autopilot in a certain month,  $m_0$ .

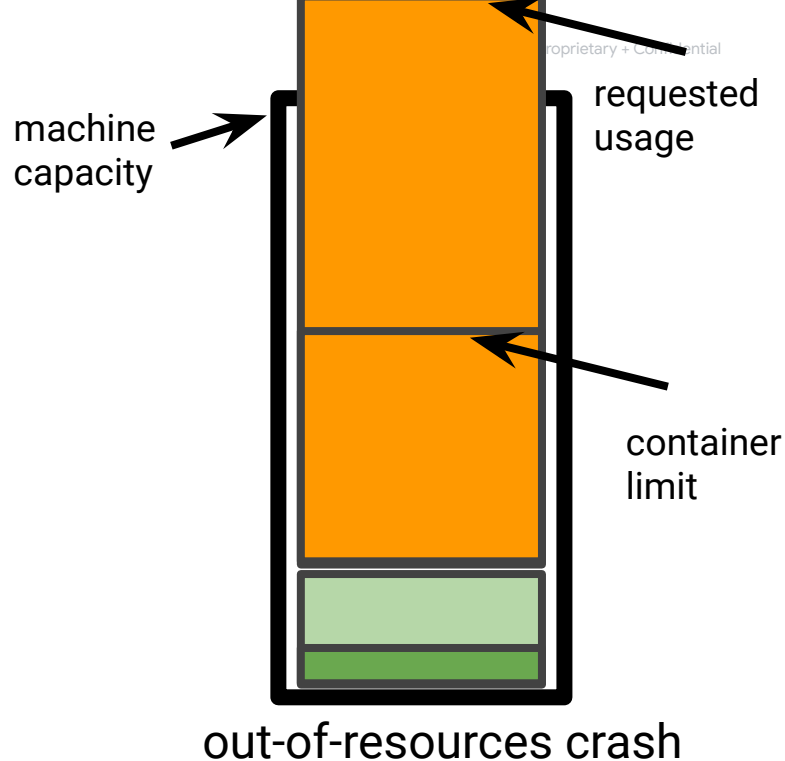
CDFs for slack for 2 months before and after migration



Autopilot Reliability:  
how frequent are  
out-of-memory errors.

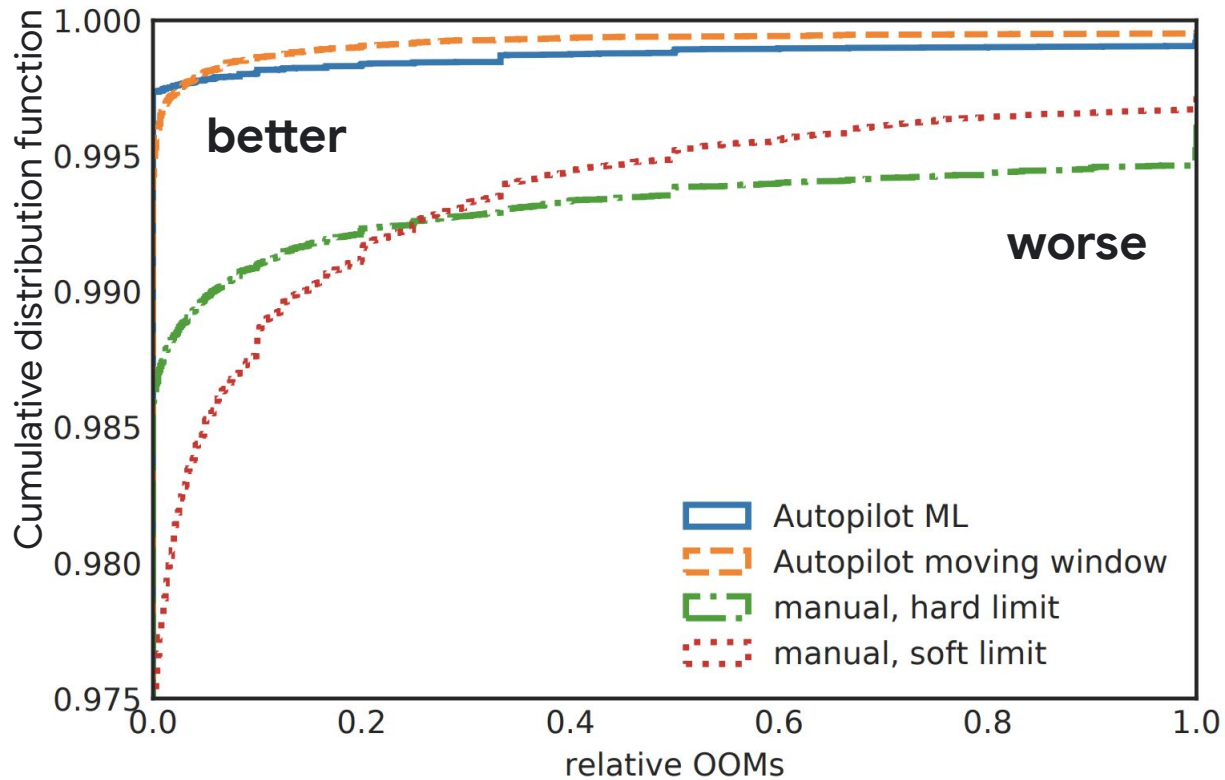
We count terminations of  
containers.

We weight the number of  
terminations by the average  
number of containers of a job.



Autopilot reduces the frequency of out-of-memory events.

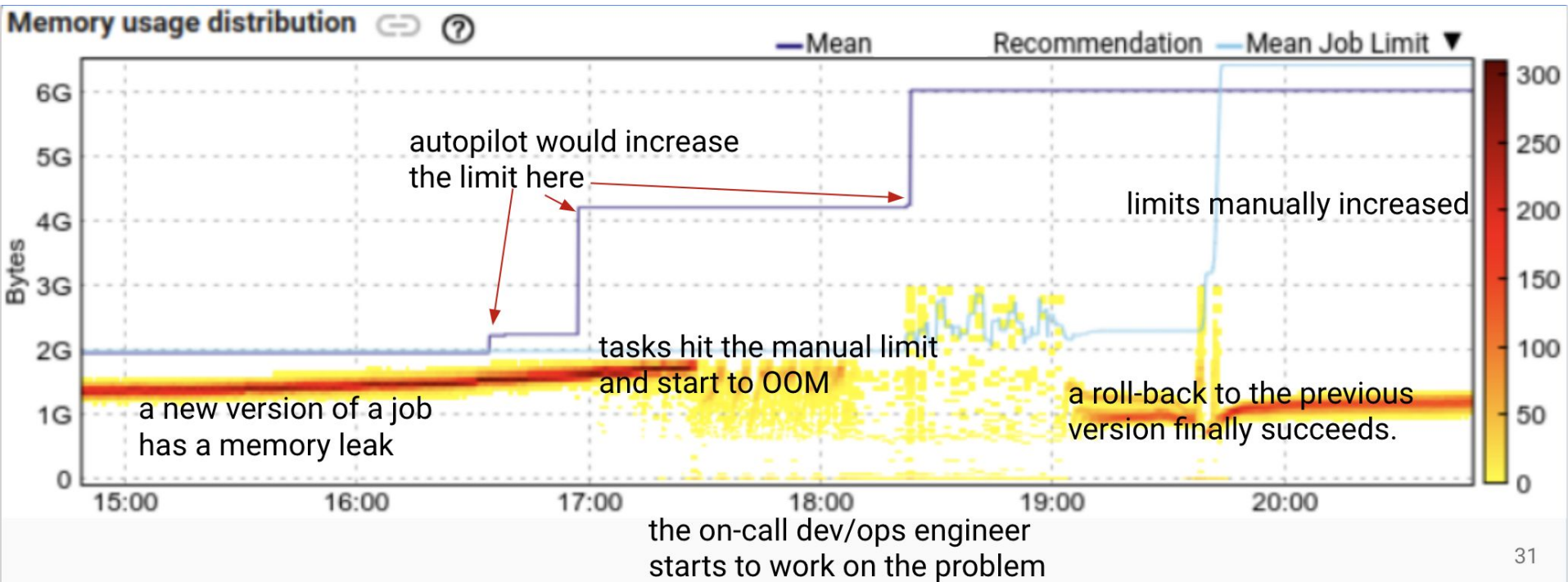
OOMs are rare: 99.5% of autopiloted jobs have no OOMs.



DevOps:

Autopiloted jobs account for over  
48% of Google's fleet-wide resource  
usage.

# Autopilot's dynamic limits could help to keep the job running despite bugs.



## Autopilot: workload autoscaling at Google

1. Efficient scheduling requires fine-grained control of jobs' limits
2. Humans are bad at setting the limits precisely.
3. Autopilot uses past usage to drive future limits
4. Autopilot reduces relative slack by 2x  
...and it reduces the number of jobs severely impacted by OOMs 10x