



Knative Backstage with the Autoscaler

Paul Morie



September 15, 2020

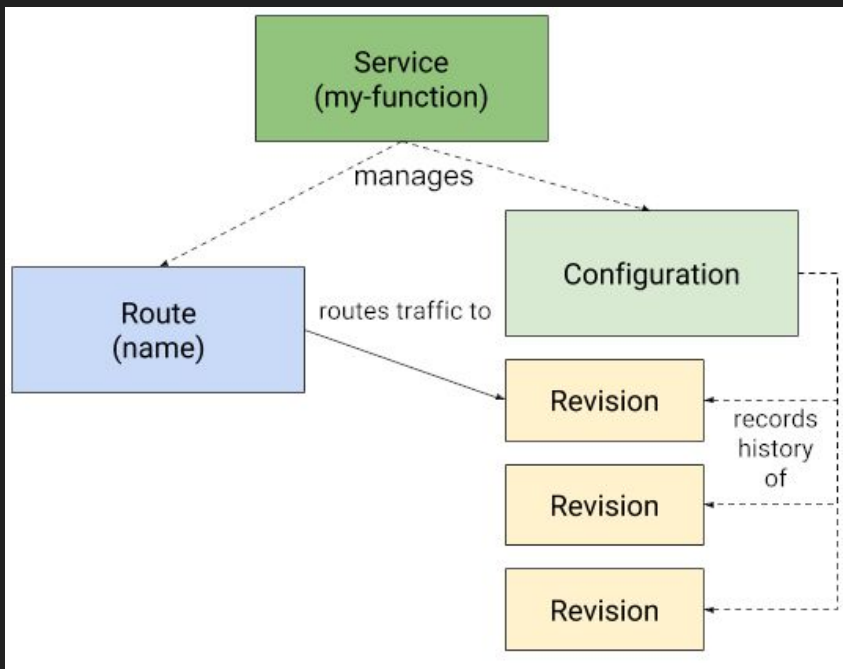
What is Knative?



- Elements of serverless on kubernetes
- Has two major functional areas:
 - Serving
 - Eventing
- Our focus today: serving and deep details of the autoscaling technology

Knative Serving 101: Fundamentals

- Many different ingresses supported (Kourier, Istio, Contour, etc)
- Key API resources
 - Service
 - Route
 - Configuration
 - Revision



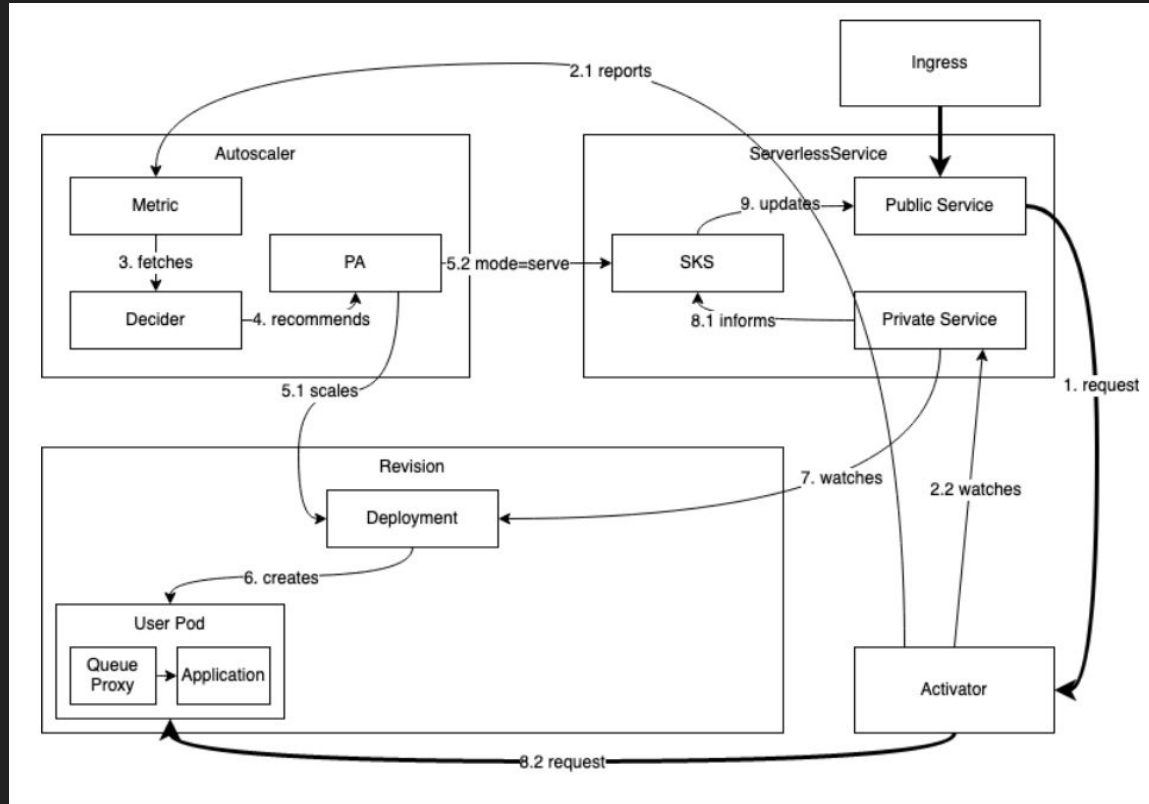
Knative Serving 201: Autoscaling

- Autoscaler
 - Collects and receives metrics from all relevant components
 - Makes scaling decisions
 - Programs the Kube API server to change replica counts
- SKS (Serverless Services)
 - An abstraction on top of Kubernetes Services
 - Controls data flow via SERVE and PROXY modes
- Activator
 - Data path component involved in scaling to/from zero
 - Also performs capacity aware load balancing
- Queue Proxy
 - Sidecar to all user pods
 - Collects metrics (scraped by autoscaler)
 - Queues requests if too many reach a pod at once

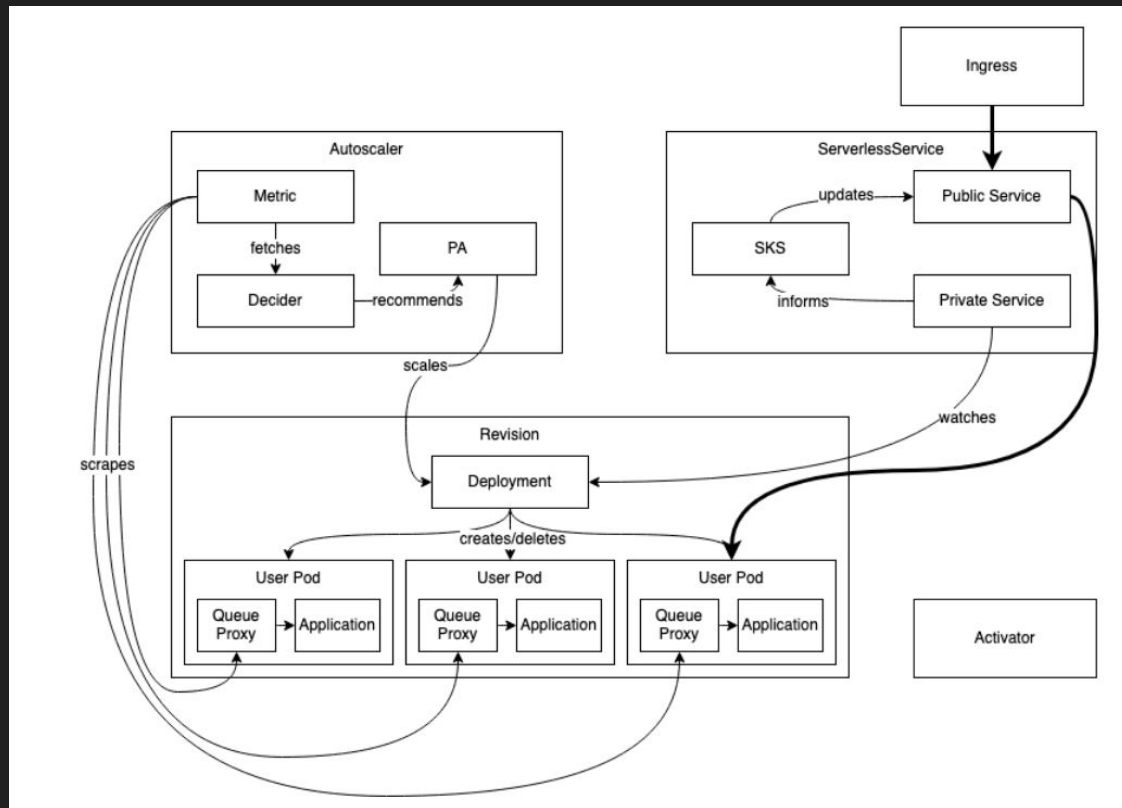
Autoscaling: Relation to HPA

- We don't use HPA currently
- HPA does not (currently) support scaling to/from zero at GA level
 - There is some [initial treatment](#) but that's only one piece of the puzzle!
- HPA is designed to scale based on CPU/memory metrics and requires a custom metrics server to scale based on requests
- Community felt KPA was easier to follow and maintain than a flow using HPA in the steady state
- Performance is also a factor; we'll see a critical use case where being able to poke the autoscaler is super important

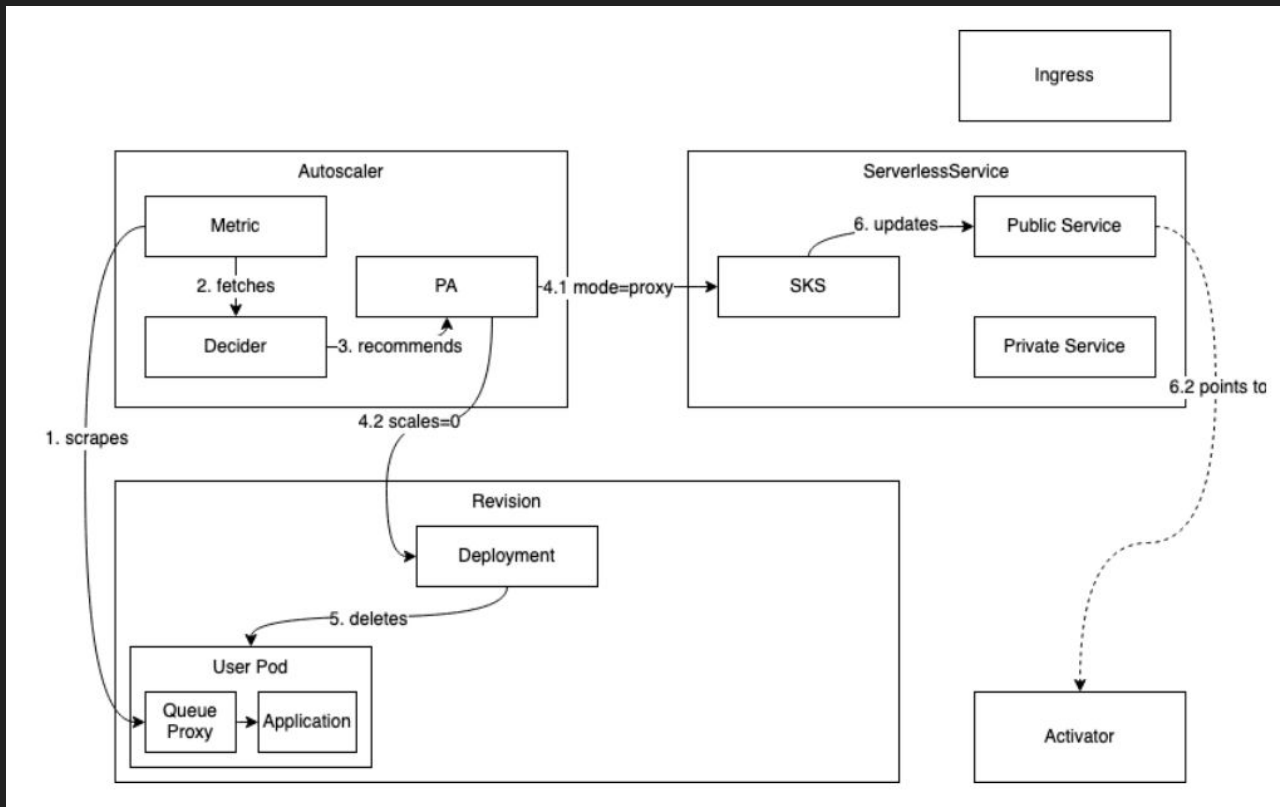
Autoscaling: Scaling from Zero



Autoscaling: Steady State



Autoscaling: Scaling Down



Thanks!

- Thank YOU for watching this talk
- Thanks to my teammates and all my open source colleagues for developing such cool tech!
- Thanks to Markus Thömmes, who wrote the awesome doc I learned a lot about this topic from!
 - <https://github.com/knative/serving/blob/master/docs/scaling/SYSTEM.md>
- Markus Thömmes and Evan Anderson, who created the diagrams used in this presentation