IBM Cloud **Code Engine**

Jeremias Werner | Lead Architect Code Engine
jerewern@de.ibm.com

About me...

Jeremias Werner

STSM, Serverless, IBM Cloud
jerewern@de.ibm.com

- **2007-2015** – Software Development
- **2016** – Initial committer of Apache OpenWhisk
- **2016-2020** – Architect of IBM Cloud Functions
- **2020-2025** – Architect of IBM Cloud Code Engine
- **2024-2025** – Architect of Serverless Fleets, a feature in Code Engine to support large scale compute intensive workloads

I’am a proud father of two children, enjoy activities in the nature and interested in financial markets...

IBM Cloud Code Engine

Run any code. Easily. At scale

IBM Cloud Code Engine is a fully-managed runtime where developers can:

Go live in seconds

Easily scale any code even down to zero

Pay only for what you use

Developers

Is Code Engine right for you?

Always included:

- Pay as you go
- Built-in security
- Private networking between your apps and jobs

Containers/ Kubernetes developer

Code Engine easily runs your containers. It is built on Kubernetes with all its features, without the learning curve. Keep your existing Kubernetes experience or Code Engine's simple user interface - the choice is yours.

PaaS/ Cloud Foundry developer

With Code Engine's simplified user experience, you can focus on writing & pushing code to the cloud. All infrastructure, networking, security and versioning of your code is managed for you.

Batch Jobs developers

Simply submit your "run to completion" batch jobs and scale them out. Never worry about provisioning, managing or removing infrastructure

Serverless/ Functions developer

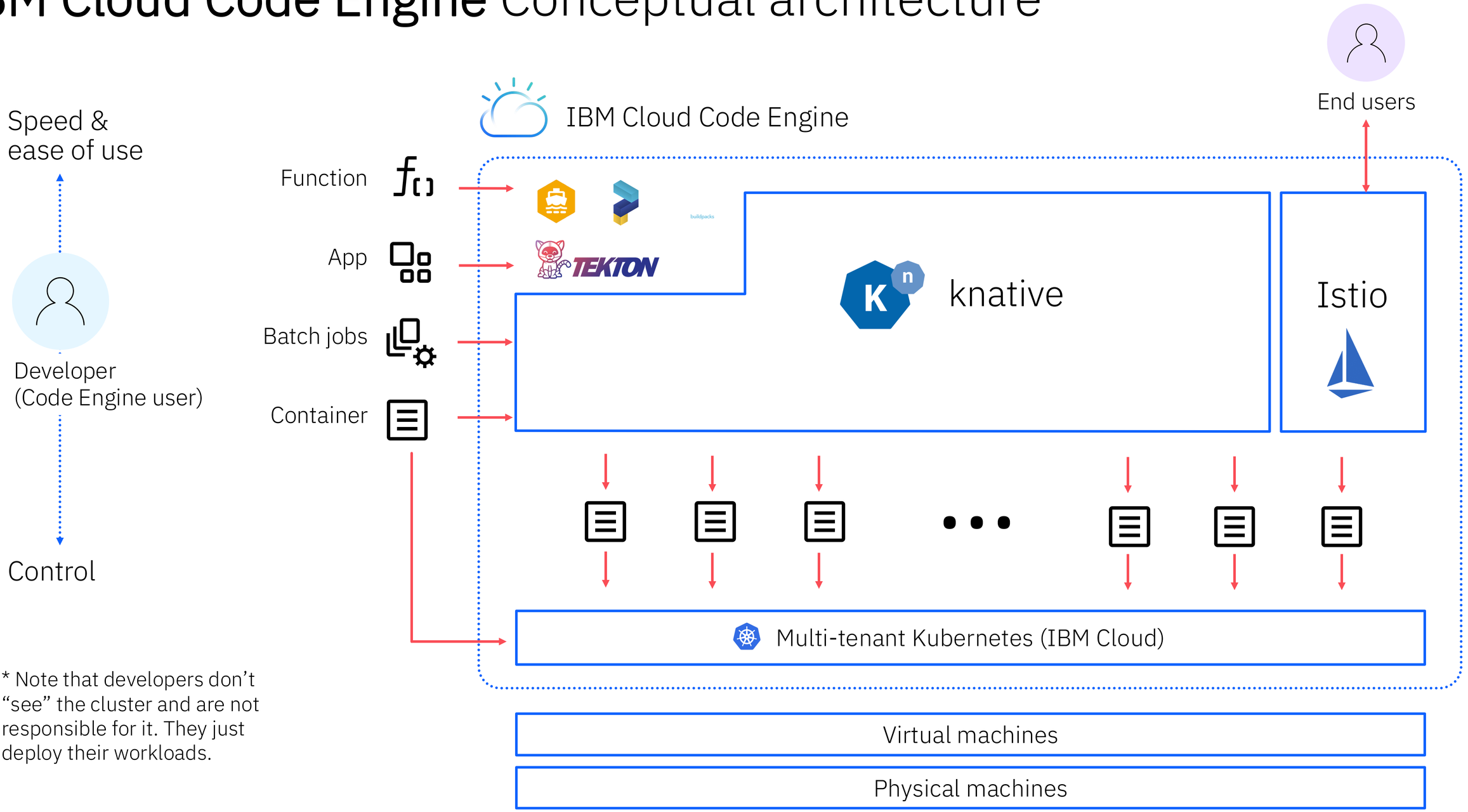
Code Engine supports running your event-driven workloads without needing to manually setup, manage or scale your code. You get the same "Functions-as-a-Service" experience that you expect from a Serverless platform.

Competitive Overview

	Multi-tenant Container service	“Push” source code	Batch jobs	Functions
aws	AWS Fargate AWS App Runner	AWS Elastic Beanstalk AWS App Runner	AWS Bach	AWS Lambda
Azure	Azure Container Instances Azure Container Apps	Azure app service	Azure Batch	Azure Functions Azure Containers Apps
Google Cloud	Google Cloud Run	Google App Engine Google Cloud Run	Google Batch Google Cloud Run	Google Cloud Functions
IBM **Cloud**	 Code Engine			

* Services in red are the closest to Code Engine in terms of functionality

IBM Cloud Code Engine Conceptual architecture



* Note that developers don't "see" the cluster and are not responsible for it. They just deploy their workloads.

Demo

https://www.ibm.com/cloud/code-engine

Apps

Web apps, microservices, REST APIs, HTTP servers, Single-Page-Apps

Jobs

Run-to-completion, bash scripts, ongoing tasks

Functions

Chat bot integrations, “Glue code”, Webhook targets

Apps

Web apps, microservices, REST APIs, HTTP servers, Single-Page-Apps

Jobs

Run-to-completion, bash scripts, ongoing tasks

Functions

Chat bot integrations, “Glue code”, Webhook targets

Fleets *

Large scale, **compute-intensive workloads** like Monte Carlo simulations, financial risk modelling, chemical molecule calculations, AI training & fine-tuning (serverless GPUs)

* Currently being rolled out in stages

Batch jobs vs. Fleets

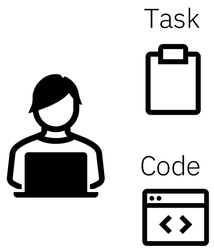
Batch Jobs	Serverless Fleets
Multi tenant isolation	Single tenant isolation
Small-medium size tasks	Large size tasks
Static array	Dynamic task queuing
vCPU only	vCPU & GPU
No control over machine profile	Full control over machine profile
Require private path to connect to VPC	Natively connect to the users VPC
vCPU-hour price: $ 0.123516	vCPU-hour price: $ 0.0480
GB-hour price: $ 0.012816	GB-hour price: $ 0.0042

The concept

Serverless Fleets run your tasks to completion

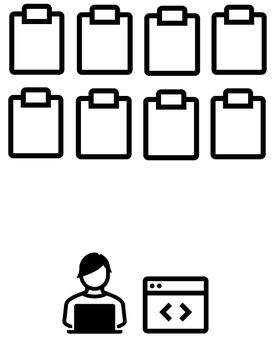
Code Engine

1. User has tasks



My local resources are not enough to run all my tasks

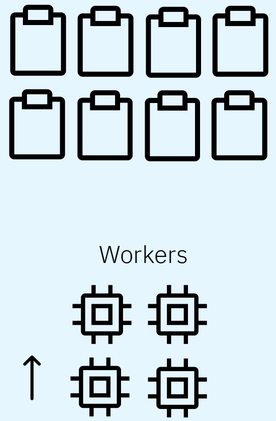
2. User starts fleet on cloud



Starting a fleet in the cloud via UI, CLI, API. Specifying:

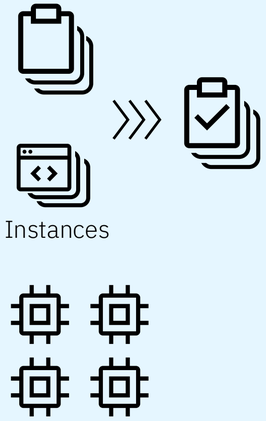
- Code
- Instance resources & scaling
- Worker machine type (optionally)

3.1 Instantiate tasks & workers



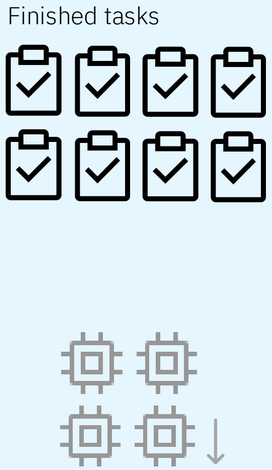
Automatically instantiating infrastructure resources (workers) based on number of tasks & instance resources and scaling specs

3.2 Instances processing tasks

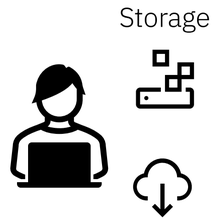


Automatically starting instances of user code to process tasks

3.3 Decommission workers



Automaticlly scaling down all infrastructure resources



Download results from storage and iterate

Demo

https://www.ibm.com/cloud/code-engine

Pricing

- Completely based on usage (allocated vCPU & memory)
- Pricing on 100ms basis
- Charging stops when apps/jobs idle and automatically scale to zero
- Free tier resets every month

Pricing

Always free	CPU 100,000 vCPU seconds per month	Memory 200,000 GB seconds per month	Incoming requests 100,000 HTTP requests per month
Beyond free tier	CPU $0.00003431 per vCPU second	Memory $0.00000356 per GB second	Incoming requests $0.538 per 1 million HTTP requests

Interested?

Visit us https://www.ibm.com/products/code-engine