DevOpsCon

Cloud Agnostic Serverless with Fn Project

Todor Todorov | @totollygeek

DevOps Evangelist | Nemetschek BULGARIA | DocuWare

Todor Todorov

@totollygeek

- .NET developer;
- clean code fanatic;
- DevOps evangelist;
- speaker;
- father of 3 boys;
- karaoke enthusiast;

The serverless explained

	DIY	Bare Metal	PaaS	K8s-aaS	Serverless
Hardware					
OS					
Hypervisor					
Virtual OS					
Container Orchestration					
Application Code					

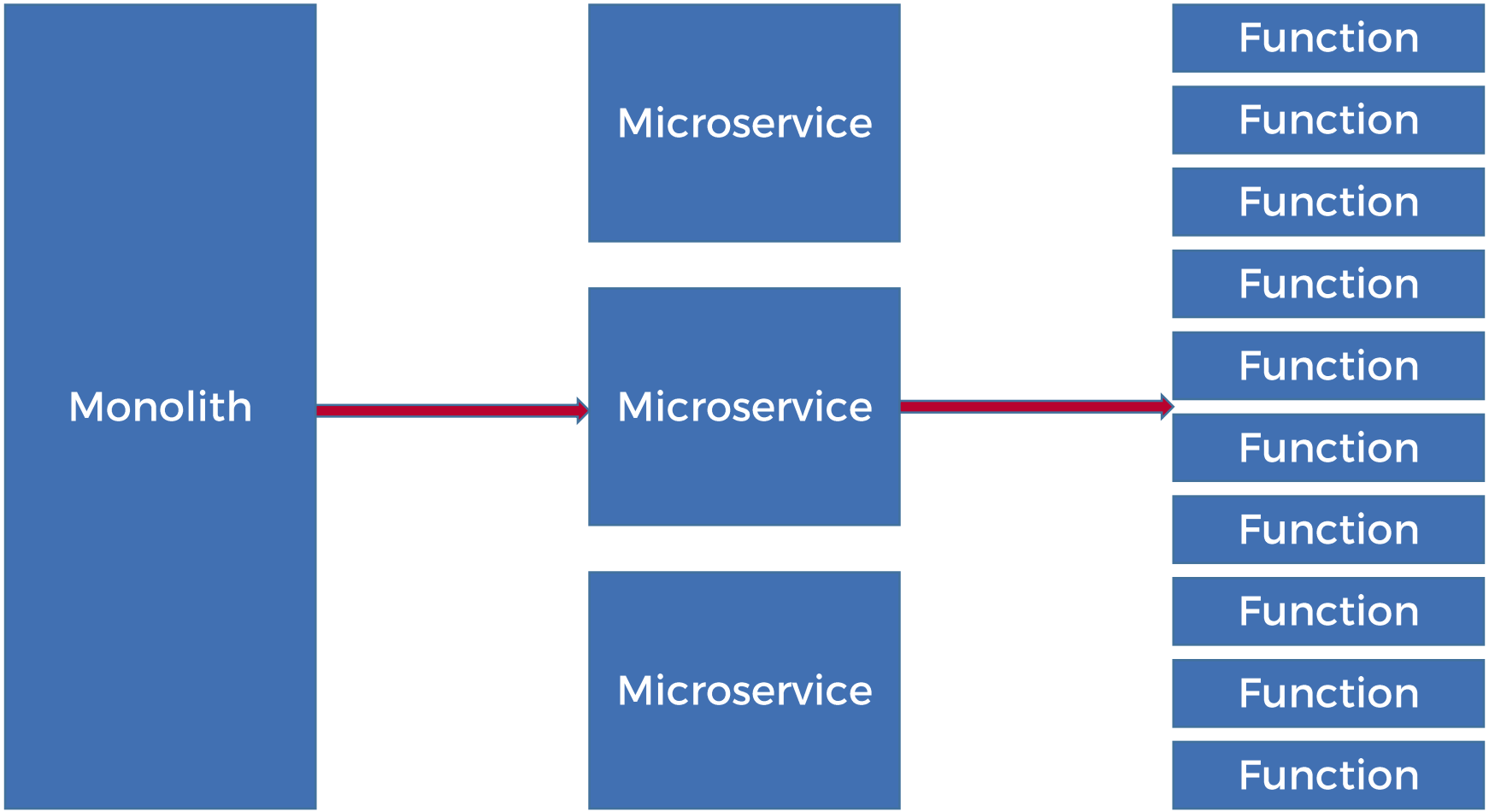
Your problem

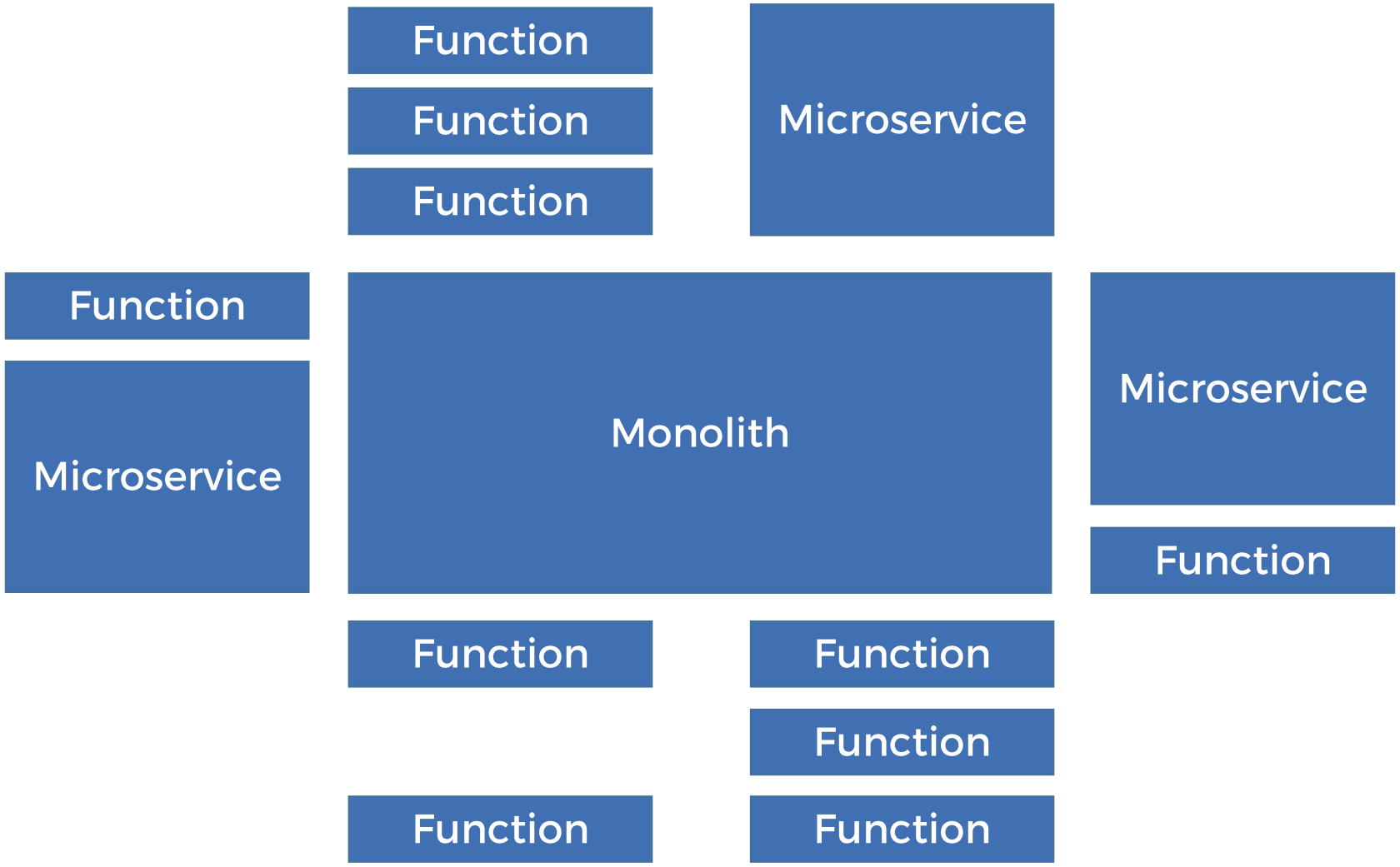
¯_(ツ)_/¯

Somebody else's problem

Credits: Matthew Gilliard


MONOLITH


Monolith
Microservice
Microservice
Microservice
Function
Function
Function
Function
Function
Function
Function
Function
Function
Function


Function
Function
Function
Microservice
Function
Microservice
Monolith
Microservice
Function
Function
Function
Function
Function
Function

Google Cloud

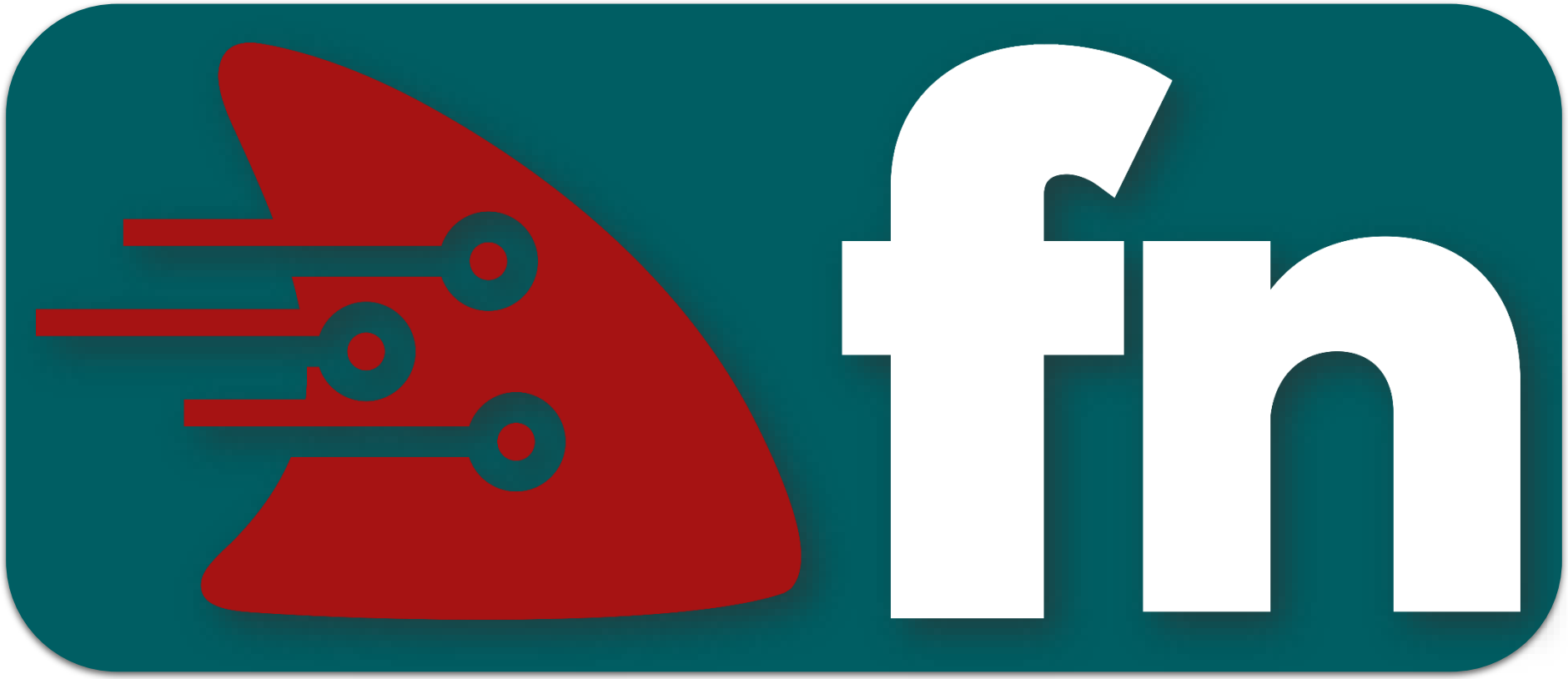
- C#
- F#
- JavaScript
- Java (**ver 2.x only**)
- PowerShell (**ver 2.x only**)
- Python (**ver 2.x only**)
- TypeScript (**ver 2.x only**)
- *some experimental*

- ❑ JavaScript
- ❑ Python
- ❑ Java (**8, 11**)
- ❑ C# (**dotnetcore2.1**)
- ❑ Go (**1.x**)
- ❑ Ruby (**2.5**)

Google Cloud

- ❑ JavaScript
- ❑ Go
- ❑ Python

That's it?!?


fn


MIND THE GAP

What is Fn Project?

- Independent **open-source** serverless compute platform

- Not tied to any cloud vendor
- Can be run on premises
- Supported by **Oracle**
- Containers are primitives
- Strong enterprise in mind

What is a function container?

- Sandboxed process
- Short running
- Event-driven
- Stateless (-ish)

Anatomy of an Fn function

- Small chunk of code wrapped in a container
- Gets input from http-stream and environment
- Sends output to http-stream
- Logs to STDERR / syslog

FDK (Function Development Kit) support

Officially supported

Community supported

Go

Java

Node.js

Python

Ruby

C#

The Fn Server

- Runs in a container also
- Handles as an API gateway
- Exposes REST interface
- Storing logs

The Fn CLI

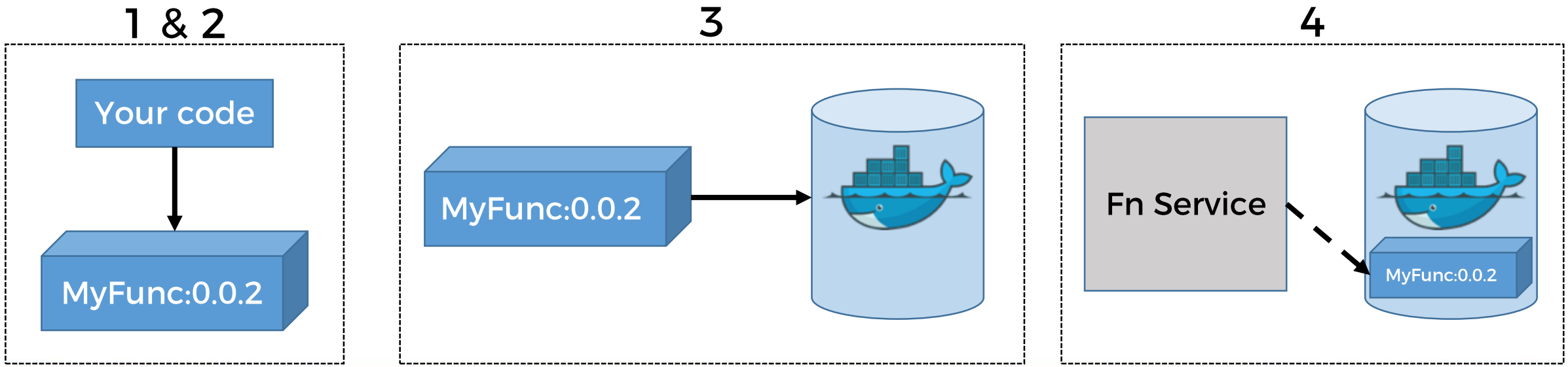
- Of course we have a CLI!
- Used to interact with Fn Server
- Initialization
- Deployment
- Invokation

Initialization of functions

- Calling **fn init** to create a boilerplate in a folder
- Does not do anything on the server
- Boilerplate includes: **Dockerfile**, **func.yaml** & **code**

How deployment works

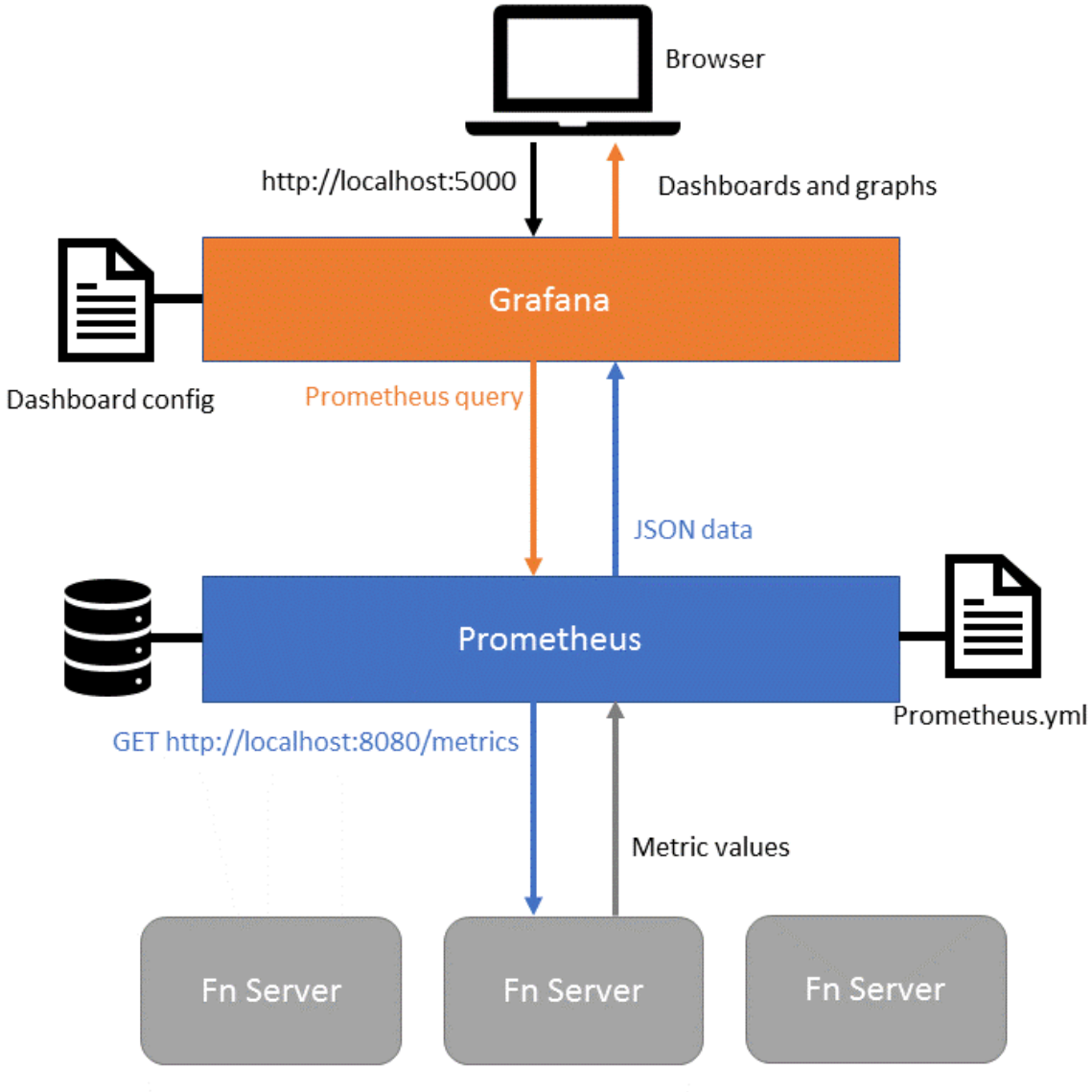
1. Bumps function version
2. Builds container
3. Pushes it to registry
4. Creates or update function & trigger in server



Invocation of functions

- From the CLI
- With HTTP request
- From the UI


DEMO



Source: https://fnproject.io/tutorials/grafana/

See metrics with Grafana & Prometheus

Source: https://fnproject.io/tutorials/grafana/

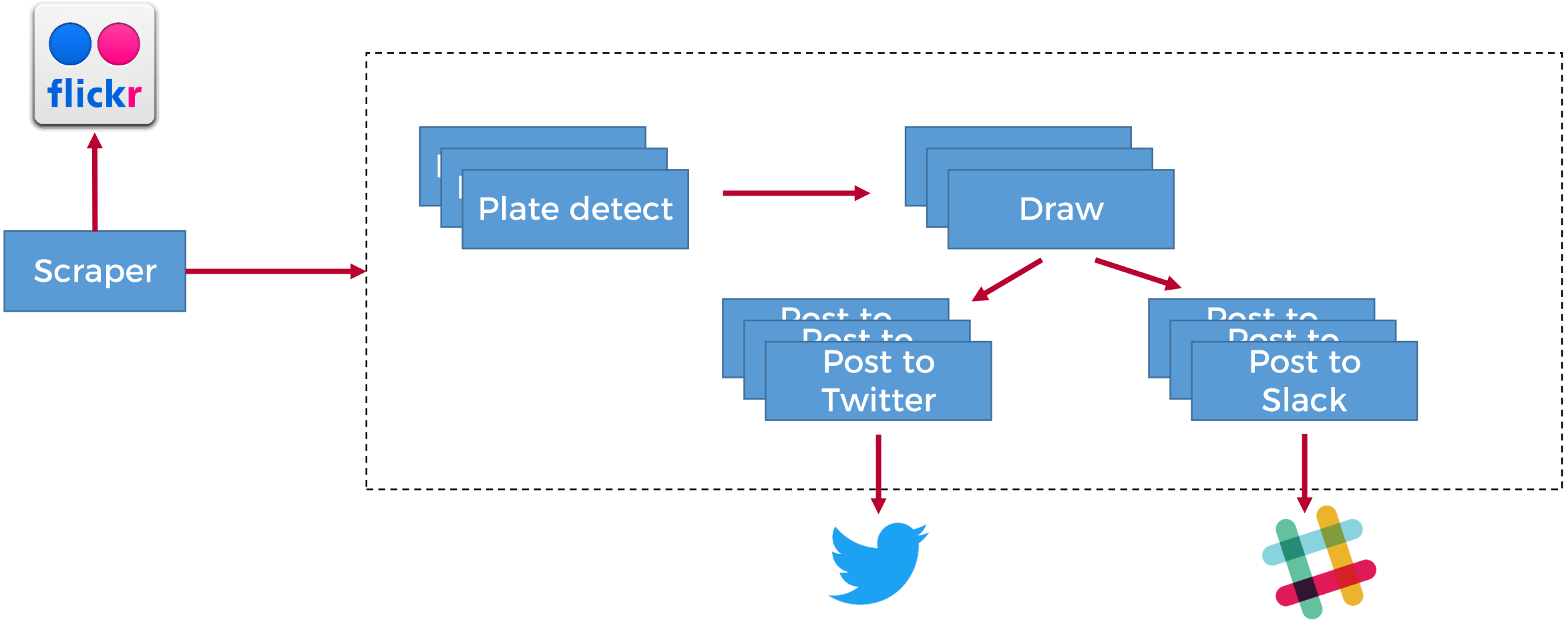
Using it in Kubernetes

- Recommended way of deployment for production
- Helm chart available on GitHub

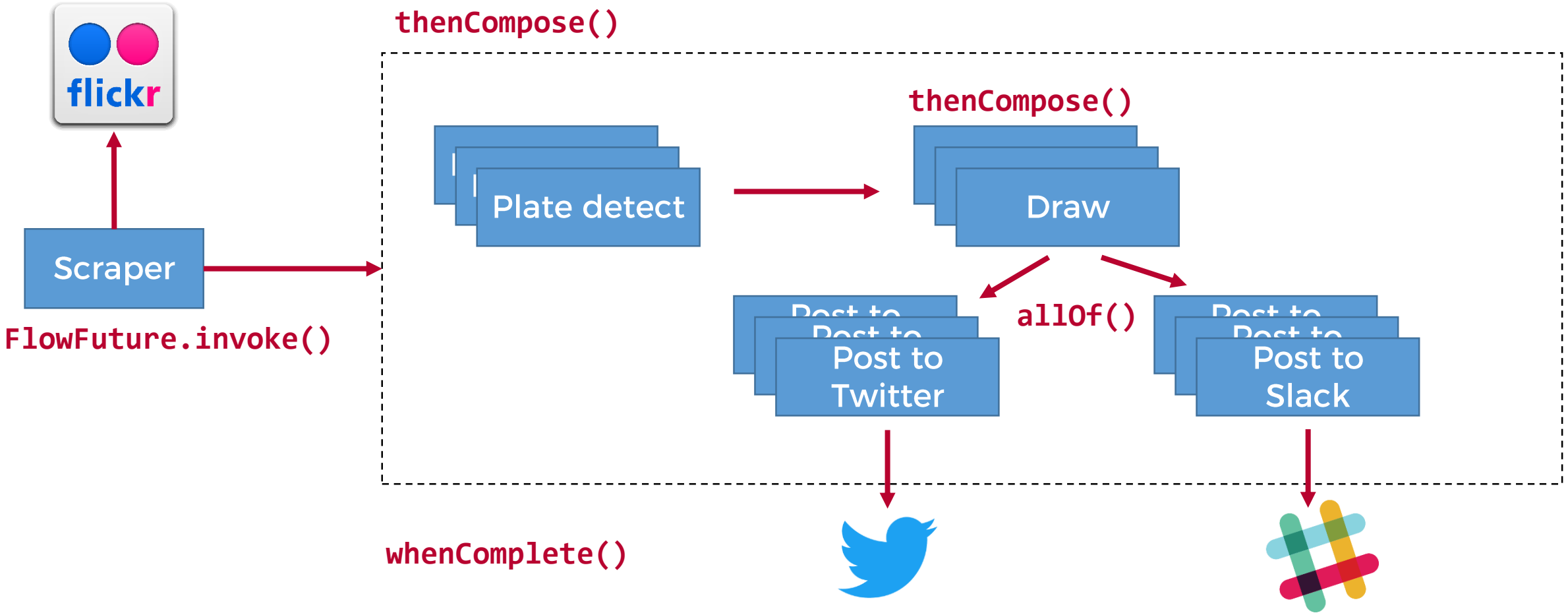
Fn Flow

- Building scalable distributed applications out of functions
- Flows are functions also
- Support complex parallel processes with error handling, which is testable
- Flow functions scale as normal functions
- Currently supports Java, hopefully more to come

Fn Flow | Licence plate example



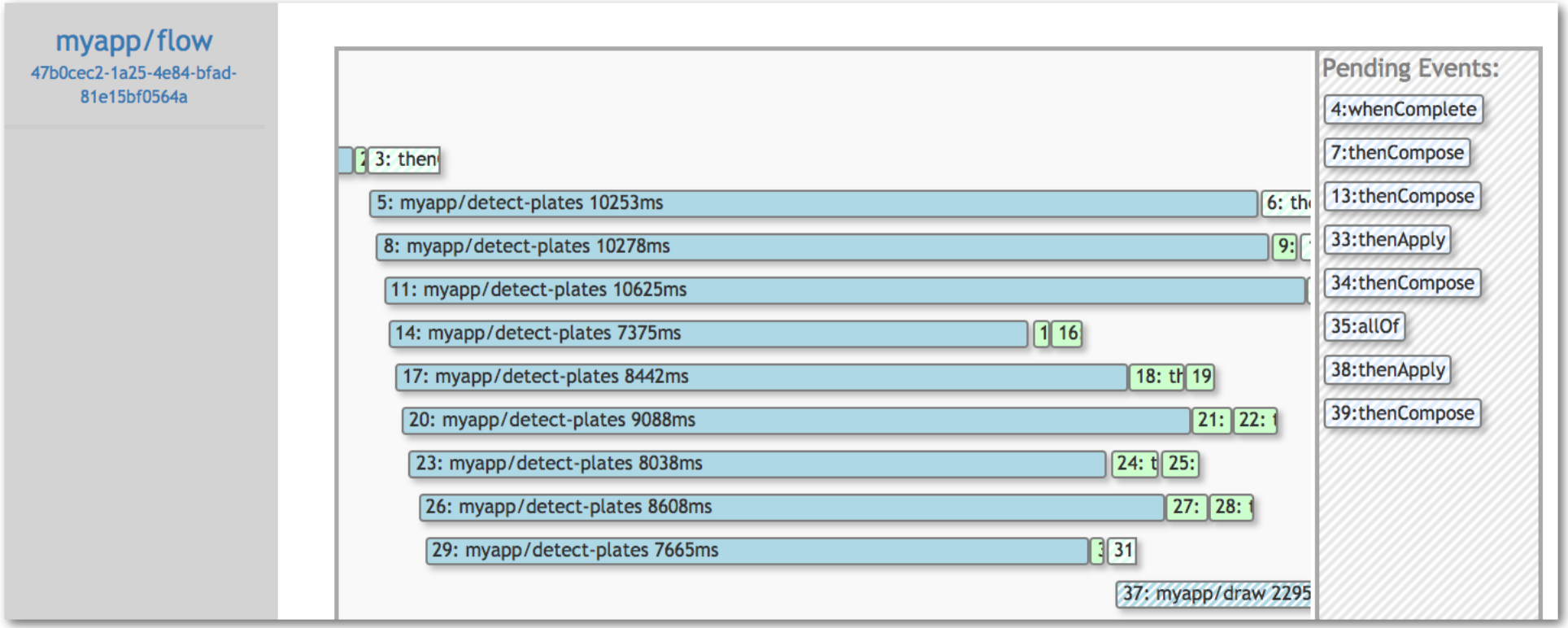
Fn Flow | Licence plate example



```
1   public class VistaFlow {
2     public void handleRequest(ScrapeReq input) throws Exception {
3
4     FlowFuture<ScrapeResp> scrapes = currentFlow().invokeFunction("scraper");
5
6     scrapes.thenCompose(resp -> {
7       List<FlowFuture<?>> pendingTasks = results.stream().map(scrapeResult -> {
8
9         String id = scrapeResult.id;
10
11        return currentFlow().invokeFunction("detect-plates").thenCompose((plateResp) -> {
12
13          return currentFlow().invokeFunction("draw").thenCompose((drawResp) -> {
14
15            return currentFlow().allOf(
16              currentFlow().invokeFunction("twitter"),
17              currentFlow().invokeFunction("slack"));
18          });
19        });
20      });
21
22    }).whenComplete((v, throwable) -> {
23        if (throwable != null) {
24          postMessageToSlack("An Error Occurred.");
25        } else {
26          postMessageToSlack("Finished Scraping.");
27        }
28      });
29    }
30  }
```

Fn Flow | Licence plate example

Fn Flow UI



Source: https://github.com/fnproject/flow


DEMO

Where to find me:

Todor Todorov

e: todor@todorov.bg

b: www.todorov.bg

t: www.twitter.com/totollygeek

l: www.linkedin.com/in/totollygeek

g: www.github.com/totollygeek