Computação Móvel e Ubíqua

Instituto de Informática - UFG

Prof. Fábio M. Costa — 2023/1

Desenvolvimento de aplicações

Middleware para computação móvel, IoT e nuvem

Roteiro

- Modelo geral de aplicações
- Visão geral do ambiente computacional
- Middleware de mensagens/eventos: Kafka
- Middleware para *Web Services*: gRPC e *Protocol Buffers*
- Alternativas:
  - MQTT
  - RESTful
  - serverless computing / function-as-a-service)

Modelo geral de aplicações

Mobilidade, ubiquidade

Modelo abstrato

- Recursos físicos (ex.: dispositivos IoT) acessíveis como serviços

Modelo concreto 0

- Acesso direto aos dispositivos (geralmente local apenas)

Modelo concreto 1

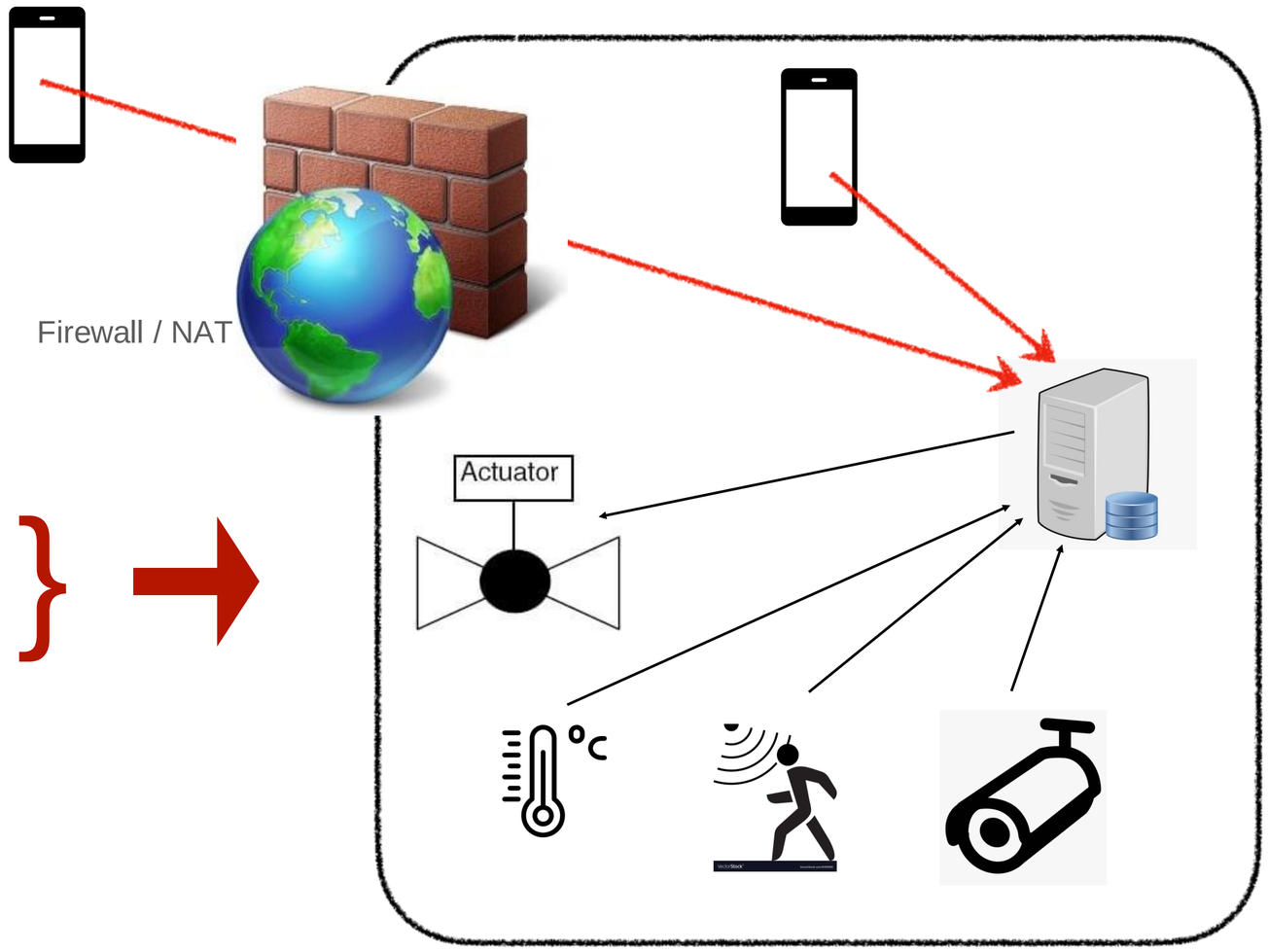
- Acesso via serviços hospedados em um ou mais servidores na rede local

Modelo concreto 2

- Acesso via serviços hospedados na nuvem

Modelo concreto 3

- Acesso via serviços hospedados na névoa e/ou na nuvem



Modelo geral de aplicações

Mobilidade, ubiquidade

Modelo abstrato

- Recursos físicos (ex.: dispositivos IoT) acessíveis como serviços

Modelo concreto 1

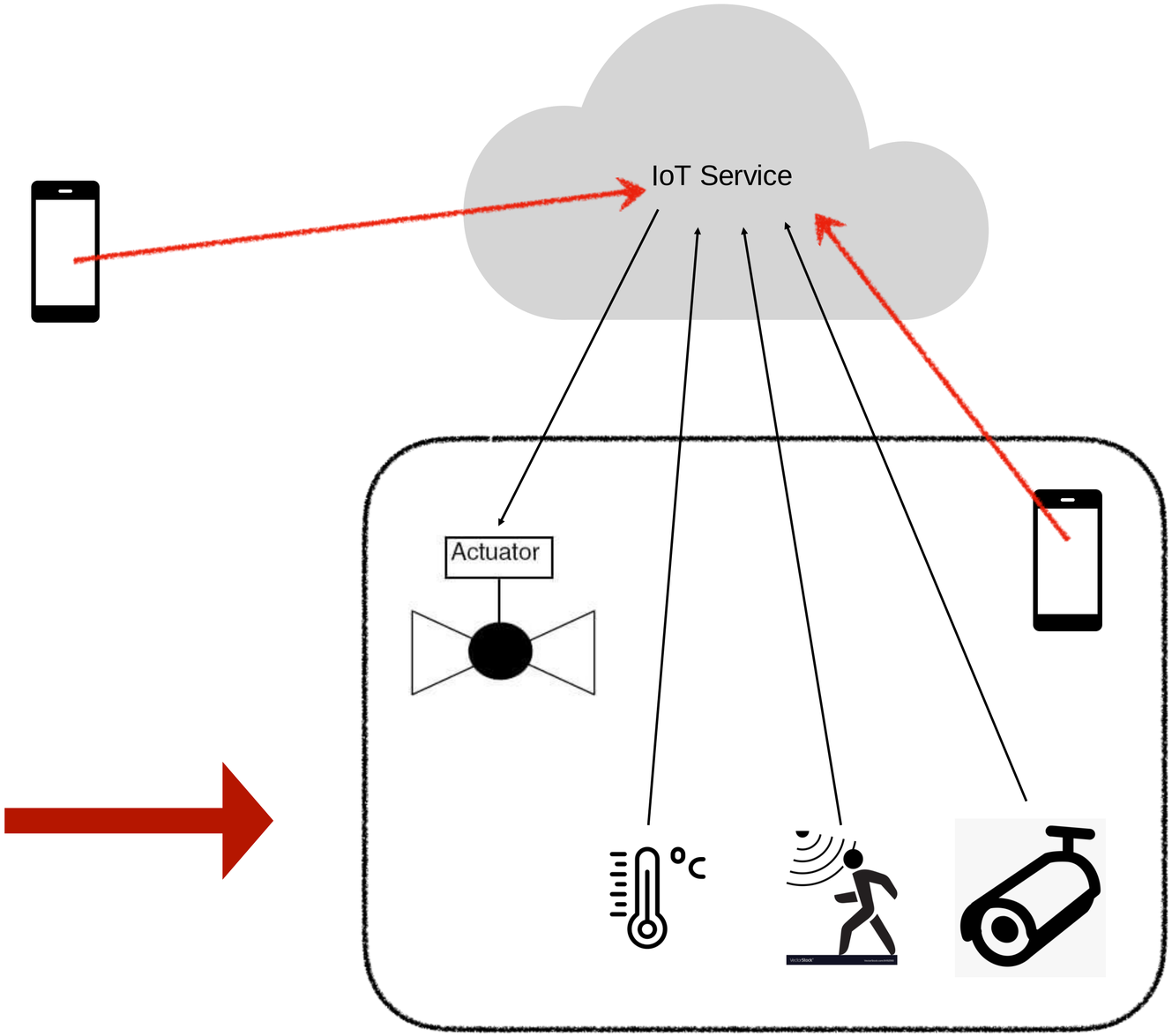
- Acesso via serviços hospedados em um ou mais servidores na rede local

Modelo concreto 2

- Acesso via serviços hospedados na nuvem

Modelo concreto 3

- Acesso via serviços hospedados na névoa e/ou na nuvem



Ambiente computacional

(usado no curso)

- Nuvem pública — AWS EC2
- Dispositivos — Raspberry Pi com sensores
- Apache Kafka — comunicação com dispositivos
- gRPC e Protocol Buffers — serviços Web, acesso dos clientes

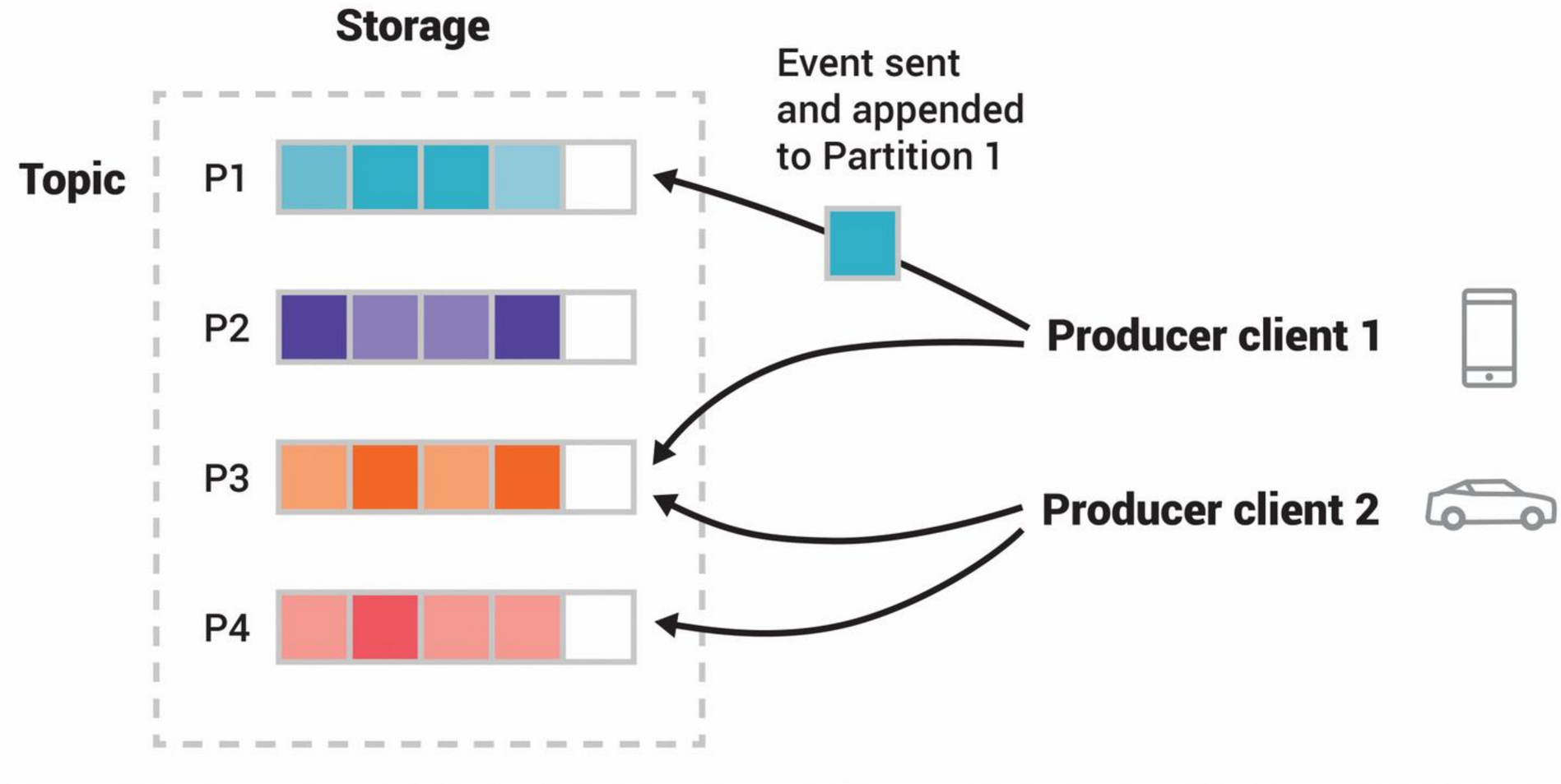
Apache Kafka

Middleware para *streaming* de eventos

- Broker (ou cluster de brokers) — recebe, armazena e repassa mensagens relativas a eventos
- Clientes
  - Publisher: produz eventos (ex.: sensor que capta e transmite dados)
  - Subscriber (ou consumer): recebe eventos de interesse
- Tópicos: usados para organizar os eventos no broker e repassá-los para os consumidores interessados

Apache Kafka

Arquitetura de mensagens básica: tópicos e partições



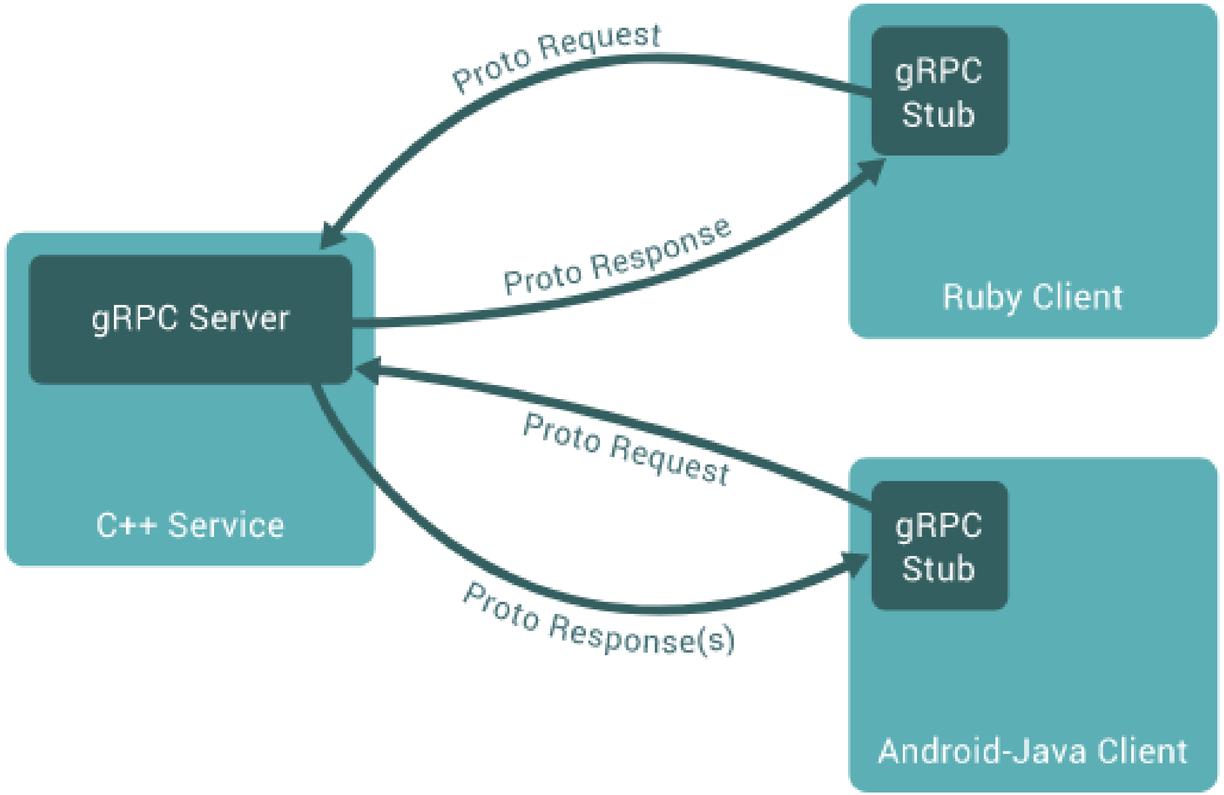
Apache Kafka

Usaremos para comunicação nuvem <—> dispositivo

- ***In a nutshell: streaming-based persistent pub-sub middleware***
- Instale o servidor Kafka na máquina que atuará como *broker*
  - https://www.apache.org/dyn/closer.cgi?path=/kafka/3.3.1/kafka_2.13-3.3.1.tgz
  - Quickstart: https://kafka.apache.org/quickstart (instruções gerais para instalação e uso do *broker*)
  - Obs.: edite o arquivo **`config/server.properties`** para colocar o endereço IP da máquina na qual você vai rodar o broker — altere a variável **`advertised.listeners`**, que deverá ficar assim:
    - **`advertised.listeners=PLAINTEXT://<endereçoIP>:9092`**
- Kafka-Python Client: https://github.com/dpkp/kafka-python
  - instale nas máquinas onde haverá clientes (*consumers* e/ou *producers*)

gRPC e Protocol Buffers

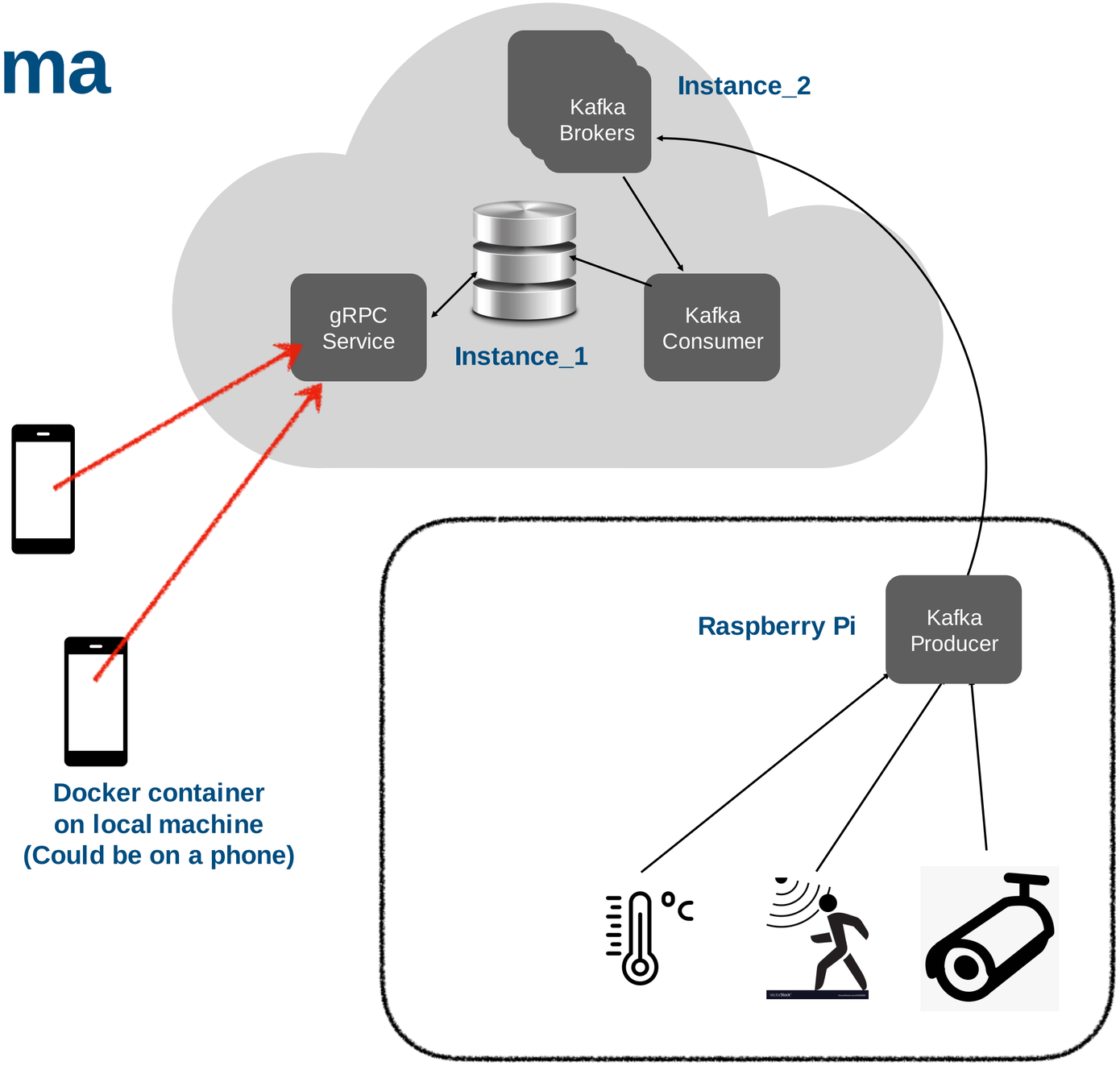
- ***Service-oriented RPC Middleware, with support for a variety of interaction patterns***
- *Quickstart in Python:*
  - https://grpc.io/docs/languages/python/quickstart/



Arquitetura de sistema

Com apenas sensores

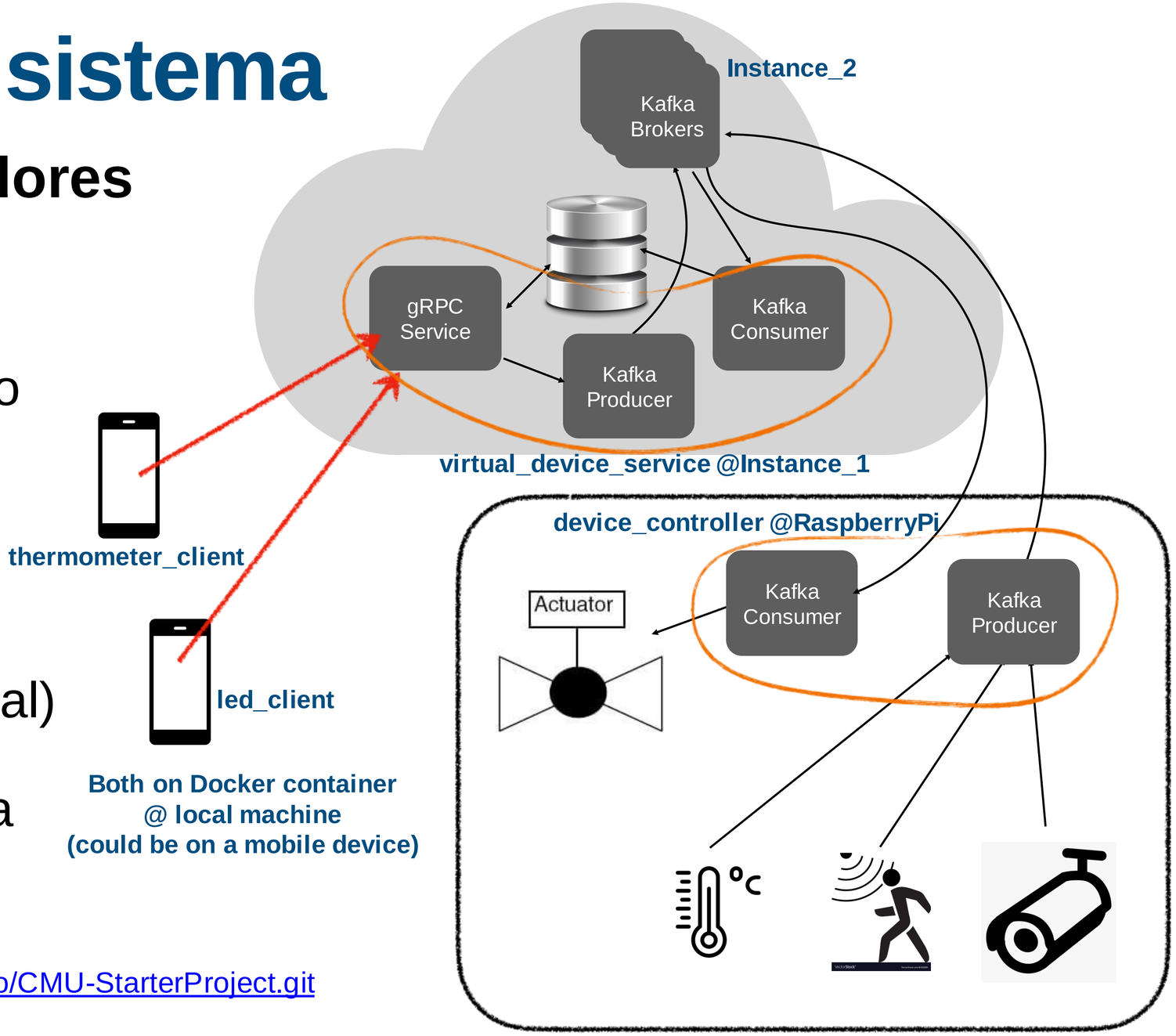
- Produtor na rede local lê sensores e publica no broker (na nuvem)
- Consumidor recebe eventos e escreve em um BD compartilhado
- Serviço Web responde requisições para acesso ao sensor (virtual)
- Obs.: assume que sensores estão fisicamente ligados ao dispositivo onde roda o Produtor (Raspberry Pi)
  - Do contrário usar um protocolo leve (MQTT, CoAP) para comunicação entre sensores e Produtor.



Arquitetura de sistema

Com sensores e atuadores

- Cliente envia comandos aos atuadores via serviço Web
- Produtor (na nuvem) publica os comandos
- Consumidor (na rede local) recebe eventos (comandos) e os executa nos atuadores.



Source code: https://github.com/professorfabio/CMU-StarterProject.git

Projeto

Ideias

- Construir um “digital twin” de um ambiente físico
- Construir uma aplicação ubíqua — continuidade transparente entre dispositivos
- Controle autônomo de dispositivos do ambiente com base em sensores e regras
- Aplicação móvel que se adapta à localização do usuário
- Localização *indoors* como contexto para uma aplicação adaptativa
- Aplicação móvel adaptativa baseada em contexto
- Virtualização de dispositivos IoT na nuvem e/ou na fog
- Offloading de funções de uma aplicação móvel