AI4EU ROBOTICS PILOT

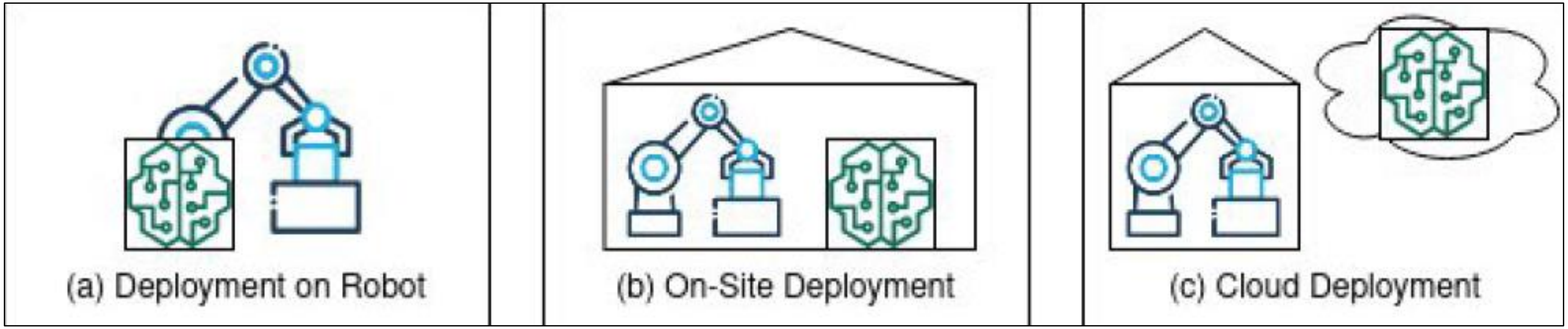
AI4EU model + ROS Interface

Fraunhofer

IPA

AI4EU Platform goals
Deployment of models

- The model is deployed directly on the robot, respectively its controller unit.
- The model is deployed centrally on-site.
- The model is deployed in a cloud environment



Platform architecture
gRPC - Remote Procedure Call (RPC) framework.

- At the core of gRPC, we need to define the messages and services using **protocol buffers**
- The rest of the gRPC code will be generated and we will have to provide an implementation for it.
- One **.proto** file works for over 12 programming languages and allows to scale to millions of RPC per second

Fraunhofer IPA

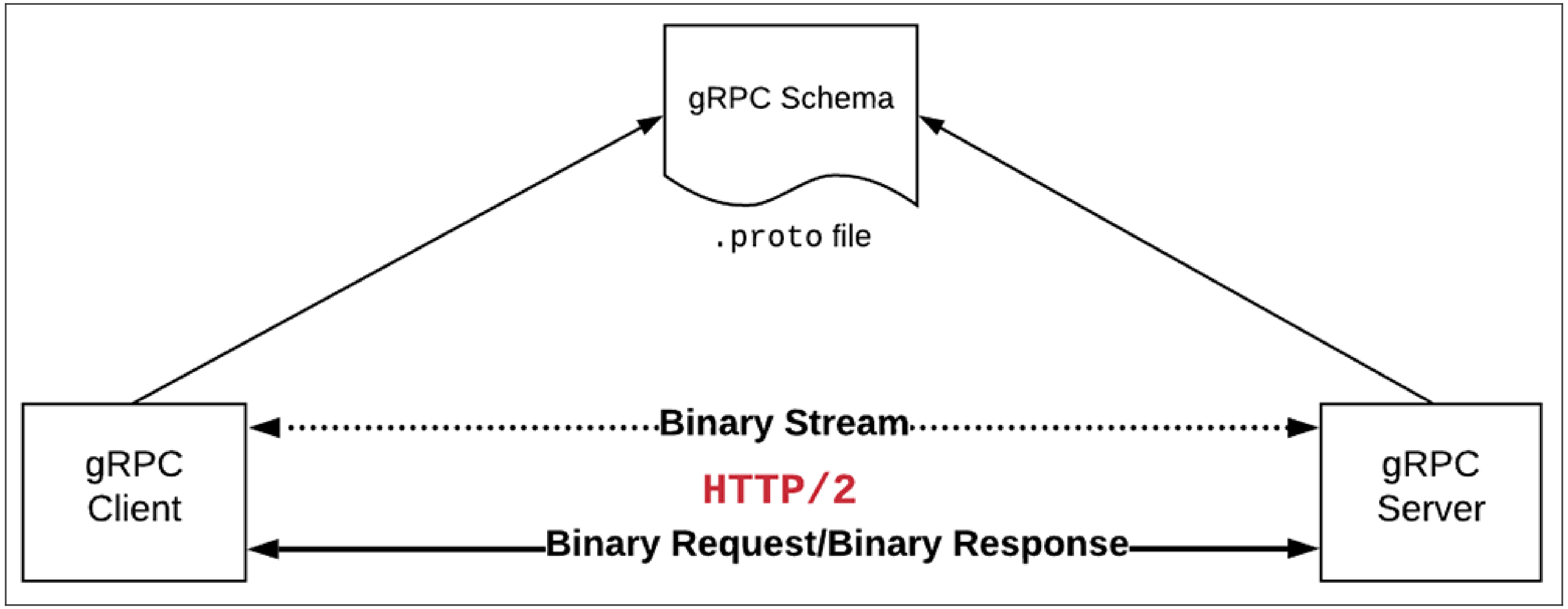
Platform architecture
Protocol Buffers

- Language agnostic
- Easy to write message definition
- Code can be generated for pretty much any language
- payload is binary and efficiently serialized – efficient
- Very convenient for transporting lot of data

Conceptual Architecture
Motivation

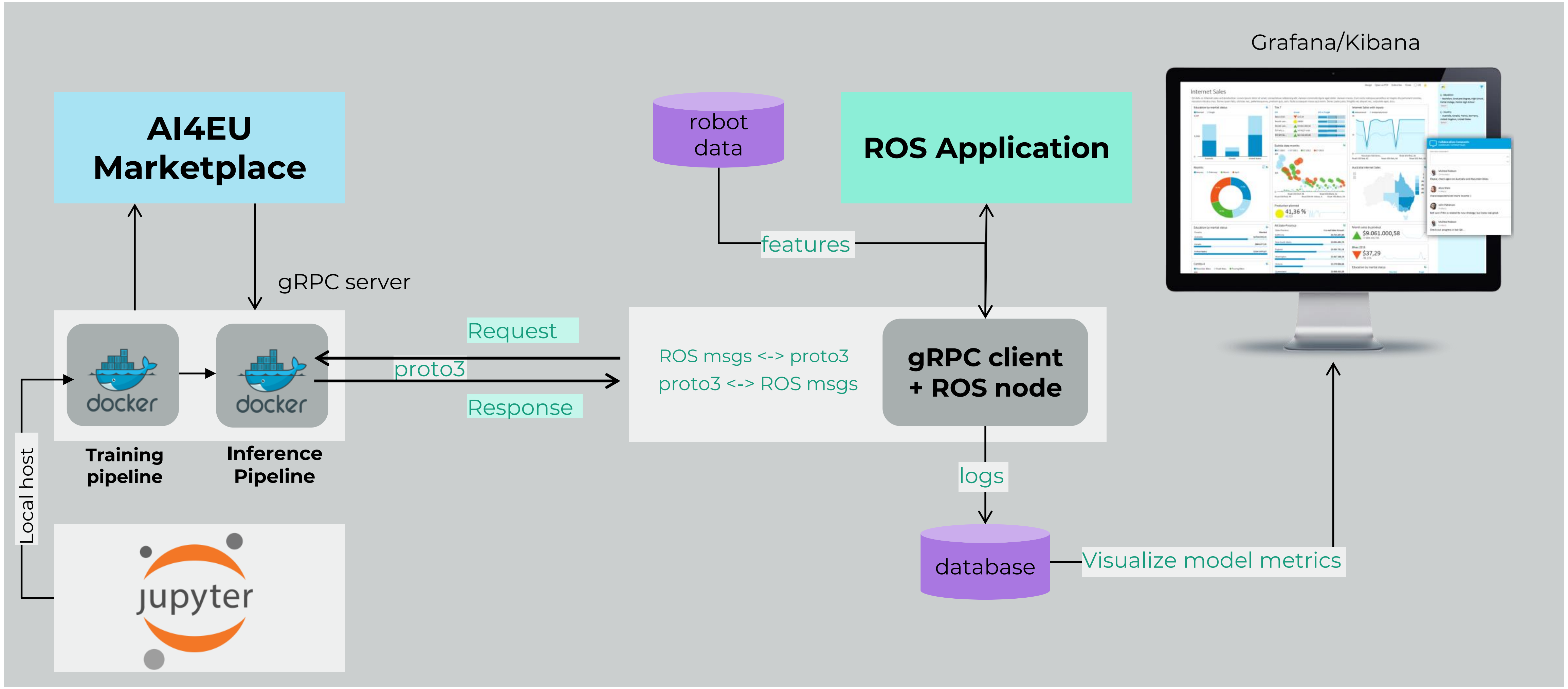
- AI4EU platform should be made accessible for robotics community
    - ROS is a de-facto standard for robot application developement
- **ROS does not have a nice way to use ML models**
    - Trained model and ingestion pipelines are hardcoded
        - Model update needs rebuild of ROS pkgs
    - No model version control
    - No means to monitor model performance
- **gRPC model deployment as microservice**
    - Independent from ROS system
        - ROS pkgs does not need build after each model update
    - Model version control with docker
    - Model monitoring can be made possible
    - Access to open source ML models from marketplace*

Platform architecture
gRPC - Remote Procedure Call (RPC) framework.



Conceptual Architecture

ROS + gRPC server



End of Presentation

Thank You

Fraunhofer IPA