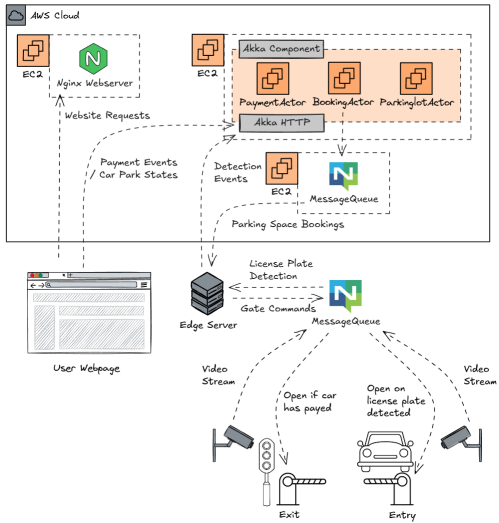
Ticketless Parking

Dominik Schweigl, Nikola Adzic, Daniel Wenger

Universität Innsbruck

System Architecture



Resources (1/3)

- **Dataset: Car License Plate Detection (Kaggle)**
  - 400+ annotated images (front/rear views)
  - Suitable for prototyping and evaluation

Resources (1/3)

- **Dataset: Car License Plate Detection (Kaggle)**
  - 400+ annotated images (front/rear views)
  - Suitable for prototyping and evaluation
- **Computer Vision: YOLOv11 + OCR**
  - Real-time license plate detection on the edge
  - **OCR** extracts the license plate string

Resources (1/3)

- **Dataset: Car License Plate Detection (Kaggle)**
  - 400+ annotated images (front/rear views)
  - Suitable for prototyping and evaluation
- **Computer Vision: YOLOv11 + OCR**
  - Real-time license plate detection on the edge
  - **OCR** extracts the license plate string
- **Messaging: NATS**
  - Pub/Sub for camera frames and control signals (barrier)
  - Asynchronous cloud-to-edge communication

Resources (2/3)

- **Edge Stack: Python + SQLite + FastAPI + Akka**
  - **Python** for ML integration
  - **SQLite** as lightweight local store
  - **FastAPI** to monitor barrier state and camera imput

Resources (2/3)

- **Edge Stack: Python + SQLite + FastAPI + Akka**
  - **Python** for ML integration
  - **SQLite** as lightweight local store
  - **FastAPI** to monitor barrier state and camera imput
- **Cloud Stack: Akka Typed + Akka HTTP + DynamoDB**
  - Actor-based orchestration for *parking lot* functionality
  - **REST API** for edge and web clients
  - Persistent actor state via **DynamoDB**

Resources (3/3)

- **Web Application: React + Tailwind CSS**
  - **React** provides a dynamic UI for availability, reservations, routing and payments
  - **Tailwind** CSS for responsive layout
  - Browser-based **geolocation** supports finding nearby parking facilities

Resources (3/3)

- **Web Application: React + Tailwind CSS**
  - **React** provides a dynamic UI for availability, reservations, routing and payments
  - **Tailwind** CSS for responsive layout
  - Browser-based **geolocation** supports finding nearby parking facilities
- **Hosting & Deployment: EC2 + Nginx + Terraform (+ S3)**
  - EC2 hosts the Akka backend and the web frontend on separate instances
  - **Nginx** serves as webserver for the WebApp
  - **Terraform** defines infrastructure as code; build artifacts are distributed via S3

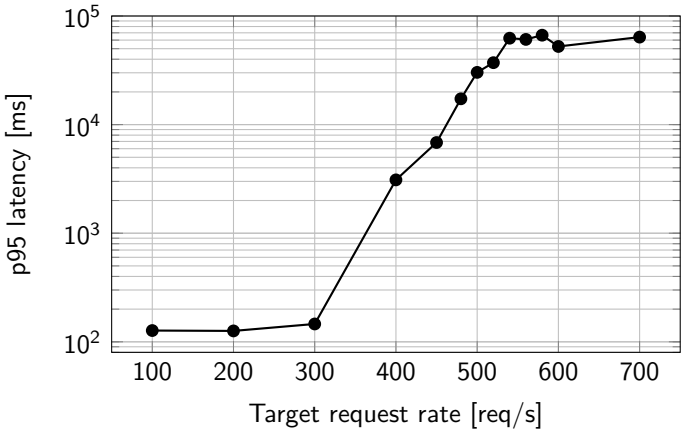
Live Demo

Video

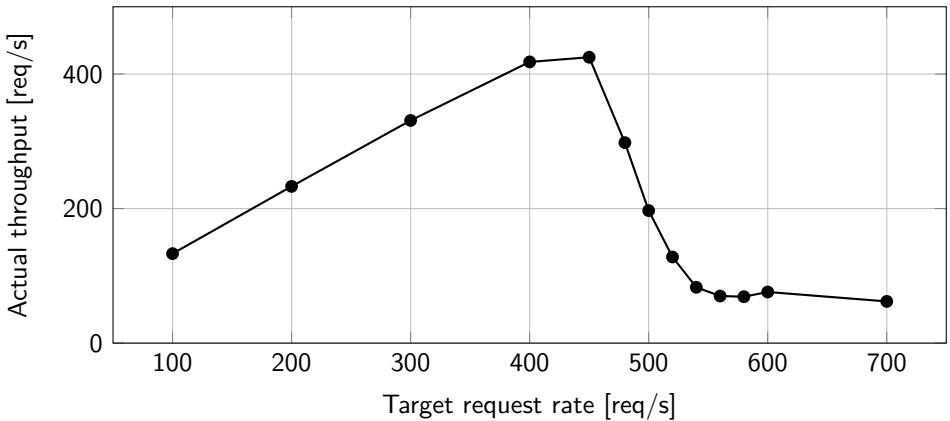
Evaluation Result (1/3)

Cloud Backend (Load Test)

- Stable up to ≈**300 req/s**
- Saturation at ≈**400 req/s**
- Overload causes:
  - high tail latency
  - request timeouts

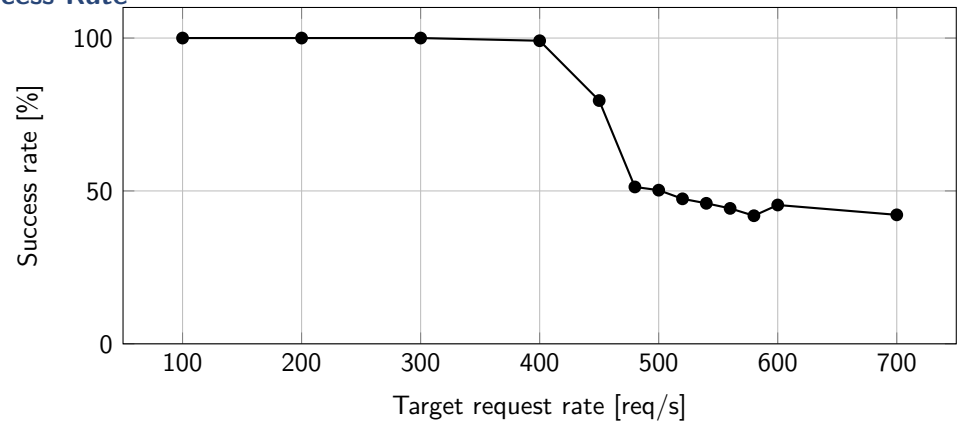


Evaluation Result (1/3)



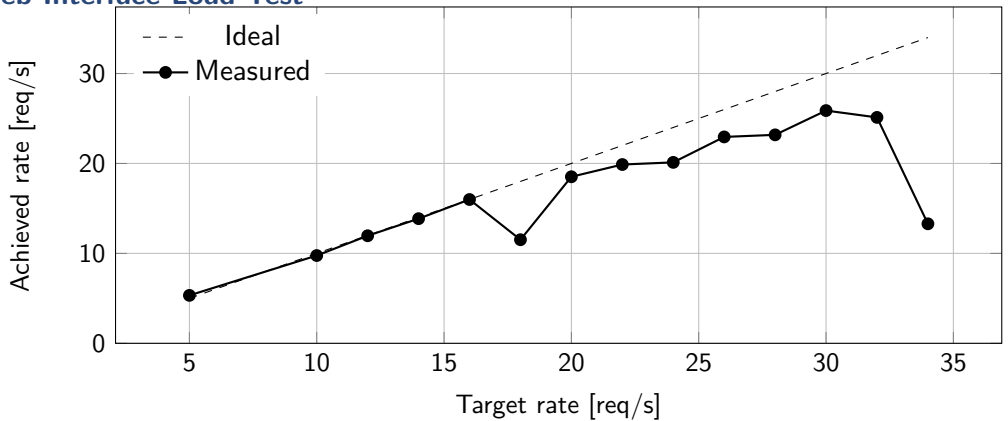
Evaluation Result (1/3)

Success Rate



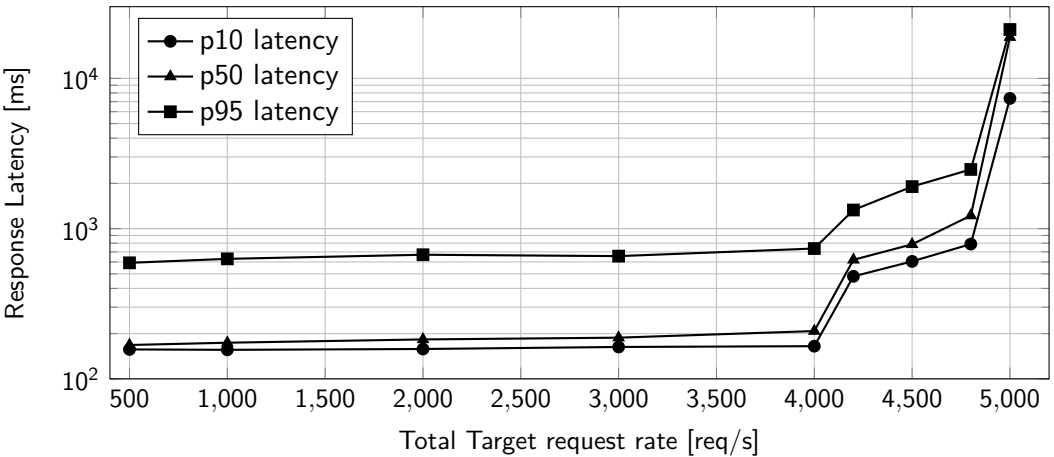
Evaluation Result (2/3)

Web Interface Load Test



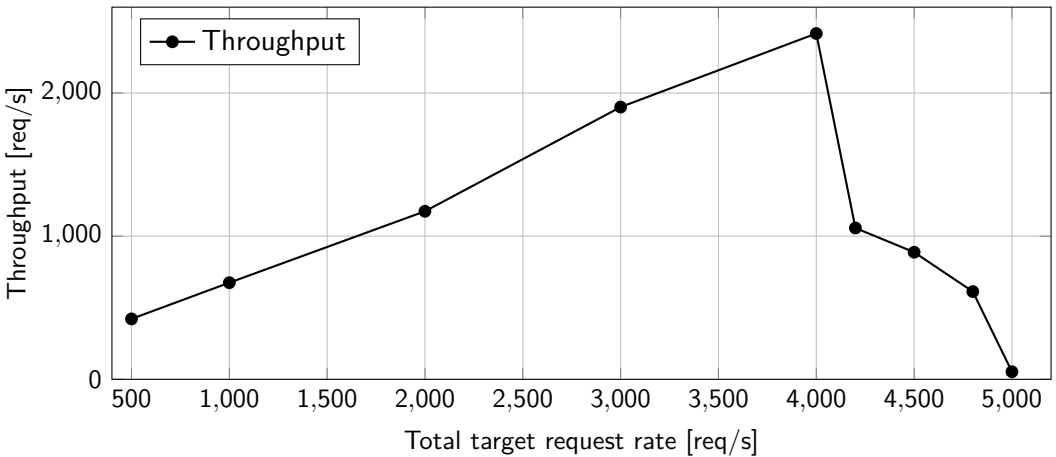
Evaluation Result (2/3)

Nginx Server Response Latency



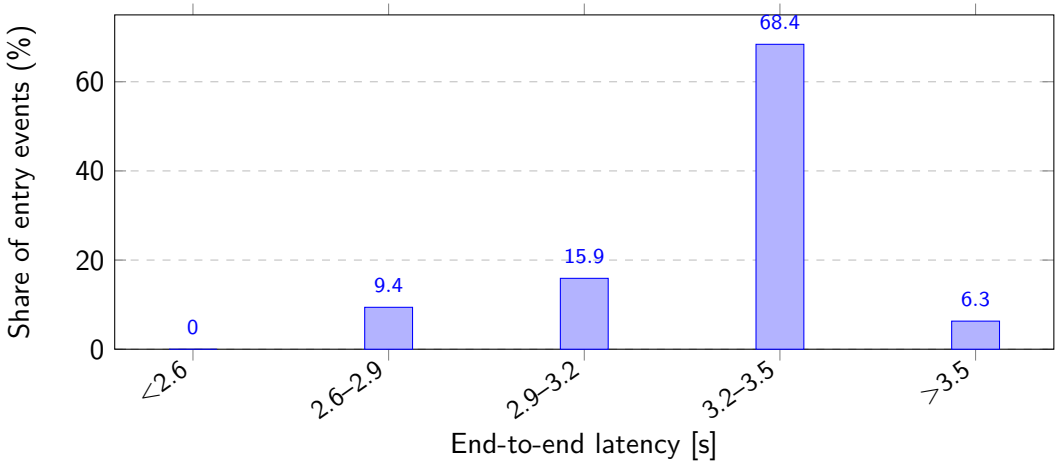
Evaluation Result (2/3)

Nginx Throughput



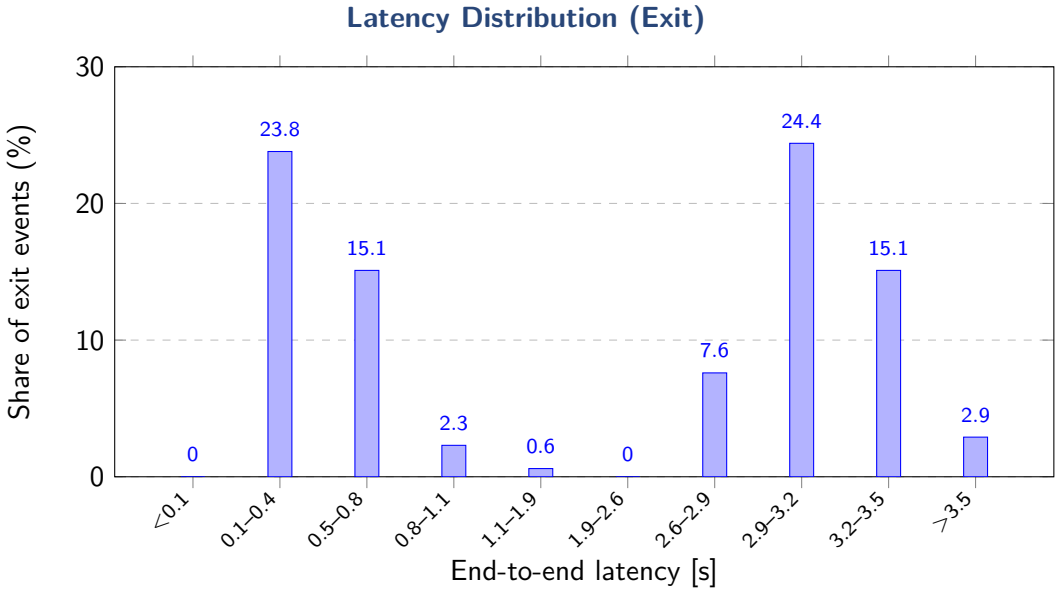
Evaluation Result (3/3)

Latency Distribution (Entry)



Total exit samples: 414

Evaluation Result (3/3)



Thank you for your attention!

Questions?