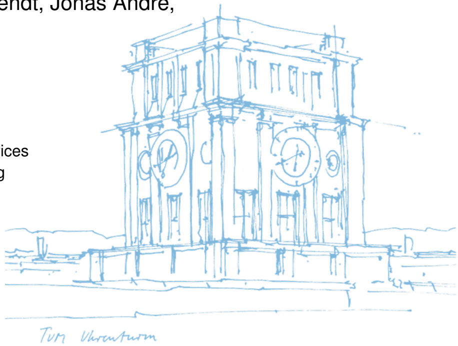
# Modeling Tail-Latencies

**Max Helm**, Florian Wiedner, Alexander Daichendt, Jonas Andre,
Georg Carle

December 1, 2023

Chair of Network Architectures and Services
Department of Computer Engineering
Technical University of Munich

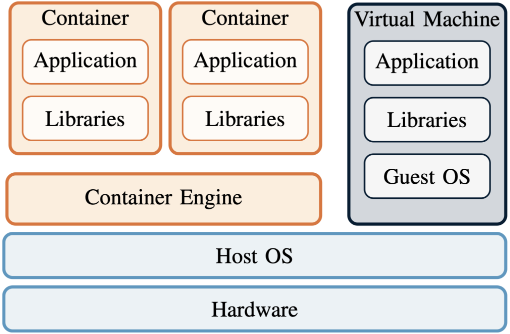


# Agenda

I. Extreme Value Theory Latency Models of Containers

II. Network Calculus as Latency Quantile Predictor Assistant

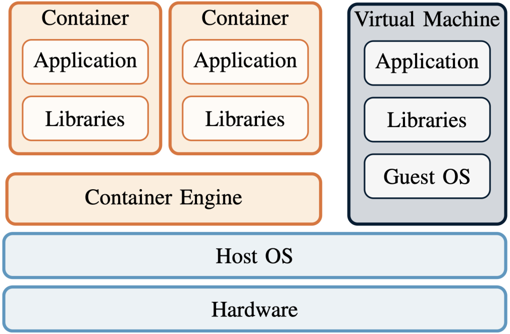
## What? Why?




- Containerized applications are important for sharing hardware resources and providing resources on-demand
- Applications with user interaction are latency-sensitive
- High impact of tail-latencies
- No available forwarding delay benchmark of containers

# I. Extreme Value Theory Delay Models of Containers

## What? Why?

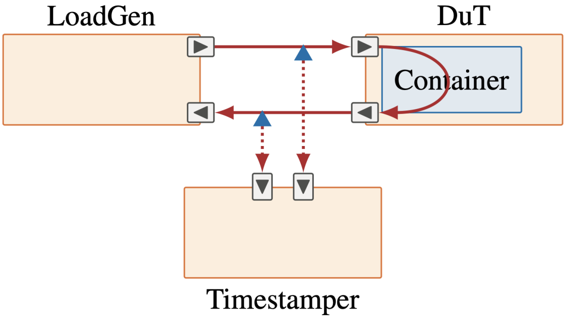



- Containerized applications are important for sharing hardware resources and providing resources on-demand
- Applications with user interaction are latency-sensitive
- High impact of tail-latencies
- No available forwarding delay benchmark of containers

$\Rightarrow$ Can we predict tail-latency behavior of containers?

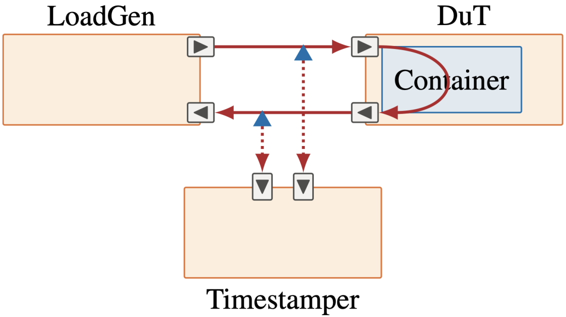
# I. Extreme Value Theory Delay Models of Containers

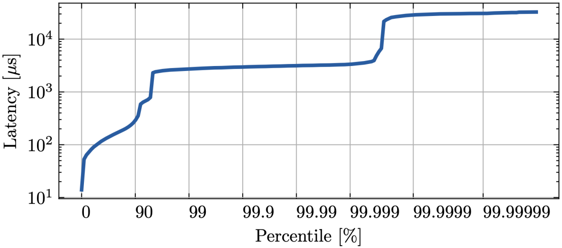
## Measurements



# I. Extreme Value Theory Delay Models of Containers

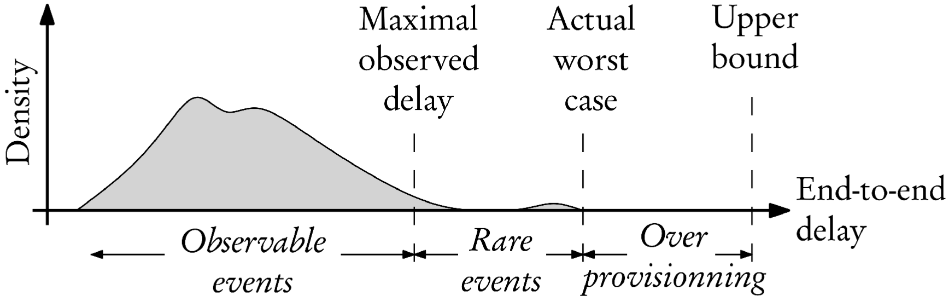
## Measurements



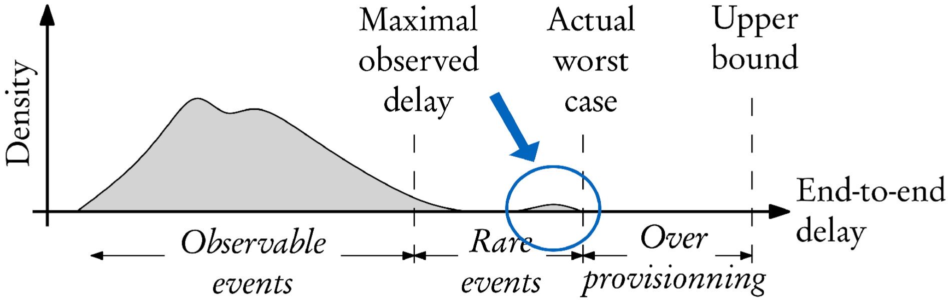



# I. Extreme Value Theory Delay Models of Containers

## Tail Latencies and Rare Events

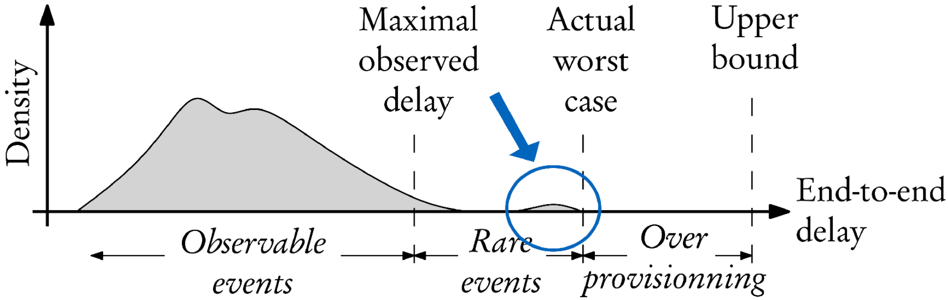


## Tail Latencies and Rare Events



# I. Extreme Value Theory Delay Models of Containers

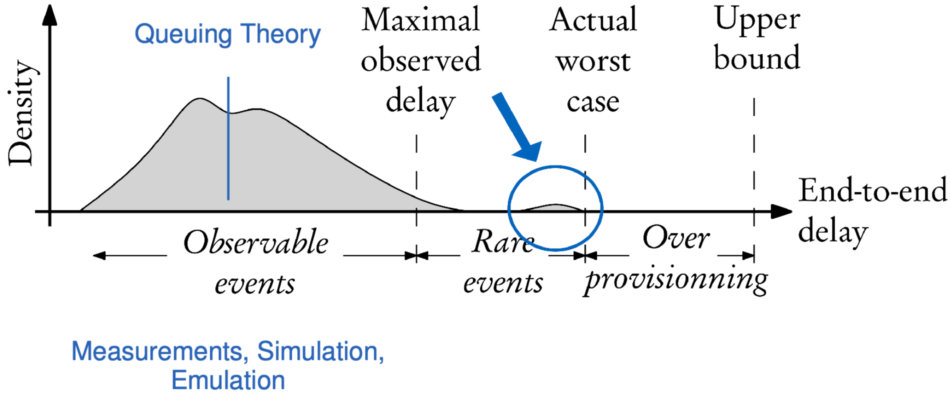
## Tail Latencies and Rare Events




Measurements, Simulation, Emulation

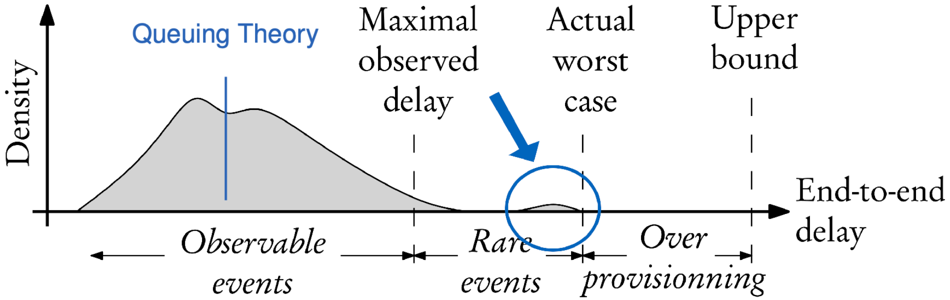
# I. Extreme Value Theory Delay Models of Containers

## Tail Latencies and Rare Events



# I. Extreme Value Theory Delay Models of Containers

## Tail Latencies and Rare Events

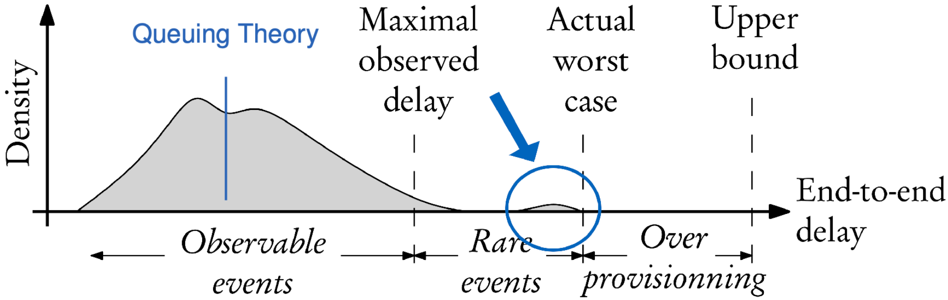



Measurements, Simulation, Emulation

Formal methods, e.g., Network Calculus

# I. Extreme Value Theory Delay Models of Containers

## Tail Latencies and Rare Events




Measurements, Simulation, Emulation

?

Formal methods, e.g., Network Calculus

## Modeling Approach: Extreme Value Theory

### Extreme Value Theory (EVT):

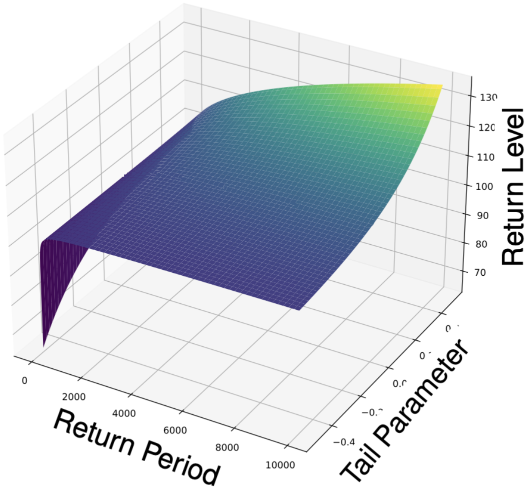
- Predict future extreme events based on historical data
- Previously used for natural disaster prediction
- High latencies are a type of extreme event in networks

### Modeling Approach:

- Select a threshold (what are tail latencies?)
- Fit a Generalized Pareto Distribution (GPD) to values above threshold using, e.g., a Maximum Likelihood Estimator (MLE)
- Obtained model can be used to extrapolate to future events, assess ”expected worst-case behavior”

# I. Extreme Value Theory Delay Models of Containers

## EVT Model




### Performance metrics:

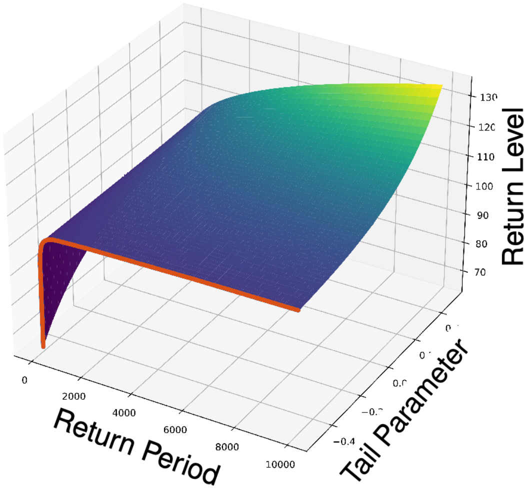
- Return level: Expected worst-case latency
- Return period: Within this timeframe
- E.g., within 10 minutes the expected worst-case latency is 30µs, within 20 minutes it is 35µs

### Model convergence:

- Expected worst-case latency converges or diverges based on sign of tail parameter
- Return period $\to \infty$

# I. Extreme Value Theory Delay Models of Containers

## EVT Model




### Performance metrics:

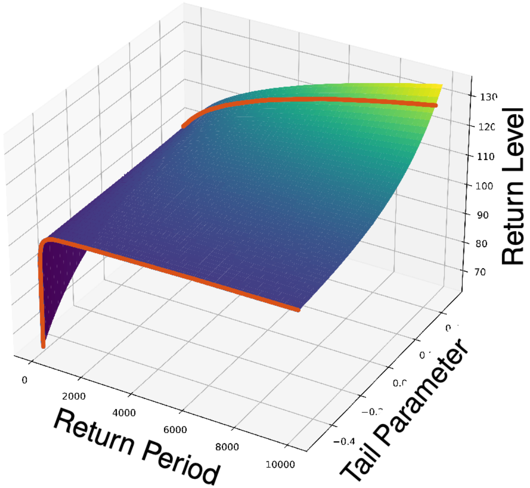
- Return level: Expected worst-case latency
- Return period: Within this timeframe
- E.g., within 10 minutes the expected worst-case latency is 30µs, within 20 minutes it is 35µs

### Model convergence:

- Expected worst-case latency converges or diverges based on sign of tail parameter
- Return period $\rightarrow \infty$

# I. Extreme Value Theory Delay Models of Containers

## EVT Model




### Performance metrics:

- Return level: Expected worst-case latency
- Return period: Within this timeframe
- E.g., within 10 minutes the expected worst-case latency is 30µs, within 20 minutes it is 35µs

### Model convergence:

- Expected worst-case latency converges or diverges based on sign of tail parameter
- Return period $\rightarrow \infty$

~~Return level is the expected worst-case latency for a given timespan~~

# I. Extreme Value Theory Delay Models of Containers

~~Return level is the expected worst-case latency for a given timespan~~

Return level is the latency that is expected to be exceeded exactly once during a given timespan

~~Return level is the expected worst-case latency for a given timespan~~

Return level is the latency that is expected to be exceeded exactly once during a given timespan

Experiment:

- Divide container latency measurements into 20% training, 80% evaluation
- Fit an EVT model to the 20%
- Make predictions for the remaining 80%

| **Platform** | **Exceedances of return level** |
|---|---|
| Optimal Model | 1.00 |
| Container | 1.50 |

~~Return level is the expected worst-case latency for a given timespan~~

Return level is the latency that is expected to be exceeded exactly once during a given timespan

Experiment:

- Divide container latency measurements into 20% training, 80% evaluation
- Fit an EVT model to the 20%
- Make predictions for the remaining 80%

| **Platform** | **Exceedances of return level** |
|---|---|
| Optimal Model | 1.00 |
| Container | 1.50 |
| Virtual Machine | 2.58 |

$\Rightarrow$ The predicted worst-case latency is exceeded 1.5 times instead of the expected one time on average

# I. Extreme Value Theory Delay Models of Containers

~~Return level is the expected worst-case latency for a given timespan~~

Return level is the latency that is expected to be exceeded exactly once during a given timespan

Experiment:

- Divide container latency measurements into 20% training, 80% evaluation
- Fit an EVT model to the 20%
- Make predictions for the remaining 80%

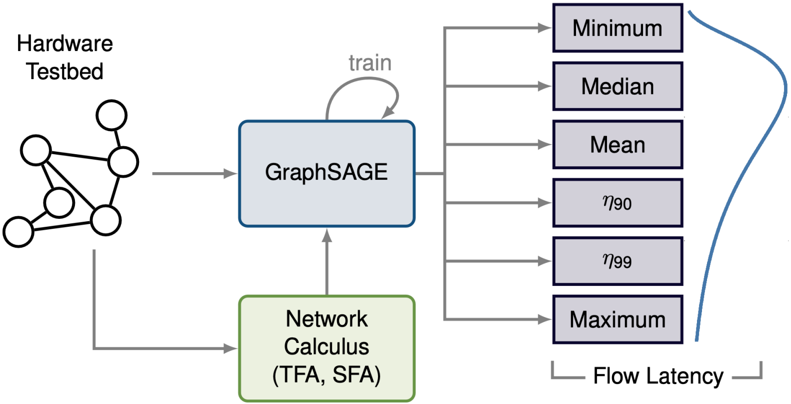
| Platform | Exceedances of return level |
|---|---|
| Optimal Model | 1.00 |
| Container | 1.50 |
| Virtual Machine | 2.58 |

$\Rightarrow$ The predicted worst-case latency is exceeded 1.5 times instead of the expected one time on average

(this type of verification of an EVT model is typically not done in literature due to scarcity of evaulation data)

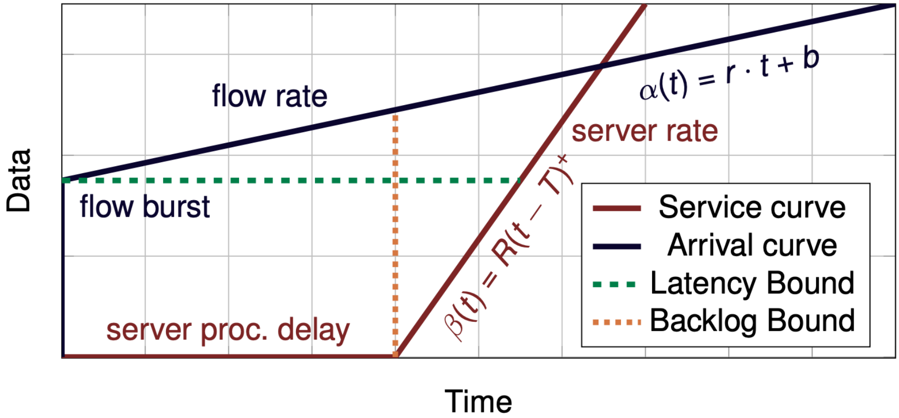
# II. Network Calculus as Latency Quantile Predictor Assistant

## What? Why?



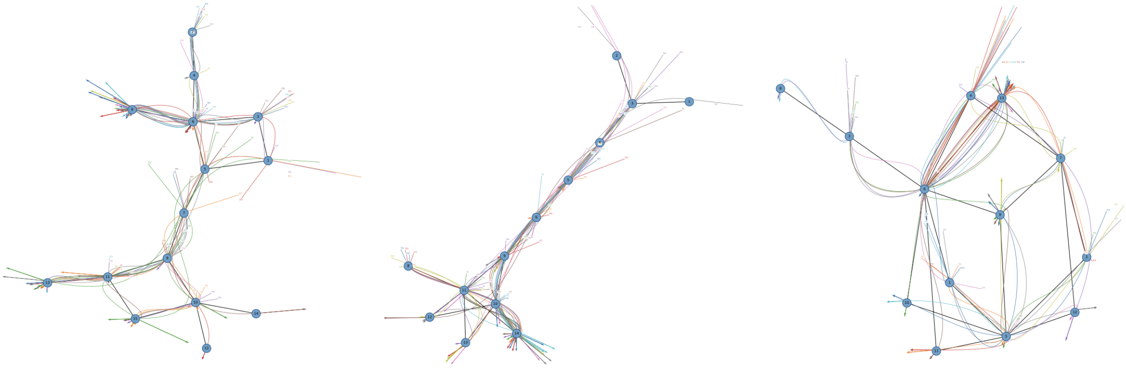
# II. Network Calculus as Latency Quantile Predictor Assistant

## Network Calculus Basics



# II. Network Calculus as Latency Quantile Predictor Assistant

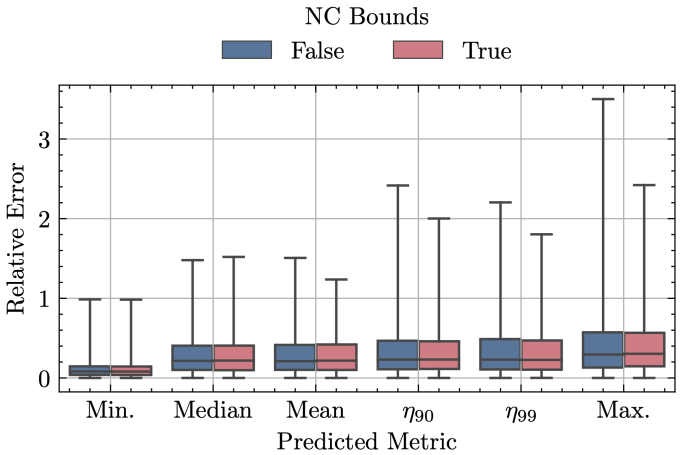
## Network Topologies



(a) Network I (b) Network II (c) Network III

# II. Network Calculus as Latency Quantile Predictor Assistant

## Latency Quantile Point Predictions



## Importance of Network Calculus Results

Analysis methods:

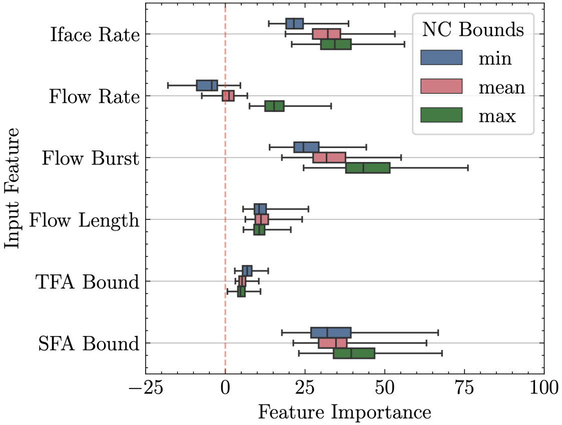
- Total Flow Analysis (**TFA**): Bounds on flow aggregates on per-hop basis
- Separate Flow Analysis (**SFA**): Bounds per flow using left-over service curves and service curve convolutions
- SFA bounds tighter or equally tight as TFA bounds
- (other analytical and linear programming-based approaches exist)

# II. Network Calculus as Latency Quantile Predictor Assistant

## Importance of Network Calculus Results

Analysis methods:

- Total Flow Analysis (**TFA**): Bounds on flow aggregates on per-hop basis
- Separate Flow Analysis (**SFA**): Bounds per flow using left-over service curves and service curve convolutions
- SFA bounds tighter or equally tight as TFA bounds
- (other analytical and linear programming-based approaches exist)



# Conclusion

## I. Extreme Value Theory Latency Models of Containers

- EVT suitable to model tail latencies

*Wiedner, F., Helm, M., Daichendt, A., Andre, J., & Carle, G. (2023). Containing Low Tail-Latencies in Packet Processing Using Lightweight Virtualization. In 35rd International Teletraffic Congress (ITC-35).*

## II. Network Calculus as Latency Quantile Predictor Assistant

- Network Calculus bounds helpful for other modeling approaches

*Helm, M., & Carle, G.. (2023). Predicting Latency Quantiles using Network Calculus-assisted GNNs. In Proceedings of the 2nd Graph Neural Networking Workshop 2023 (GNNet '23).*