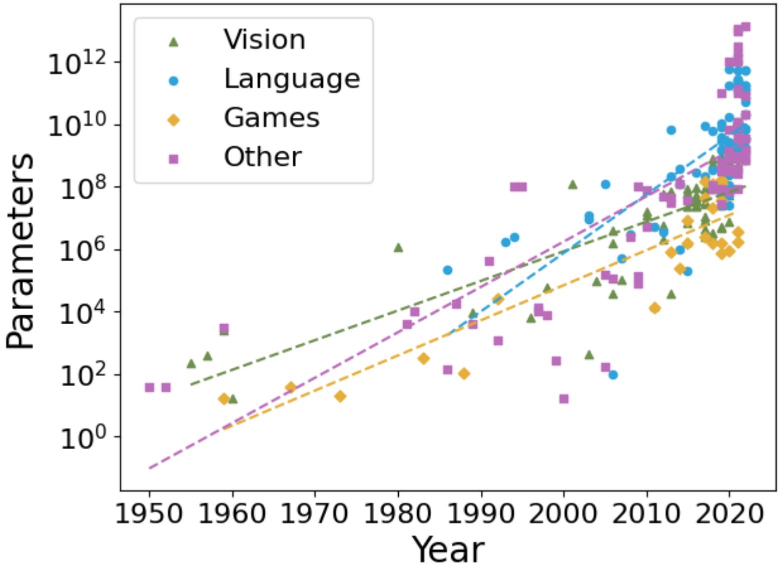
Universität Potsdam

HPI Hasso Plattner Institut
Digital Engineering • Universität Potsdam

# Survey: Data-parallel Architectures for Distributed Machine Learning

Leonard Paeleke
PhD Student
Internet Technology and Softwarization

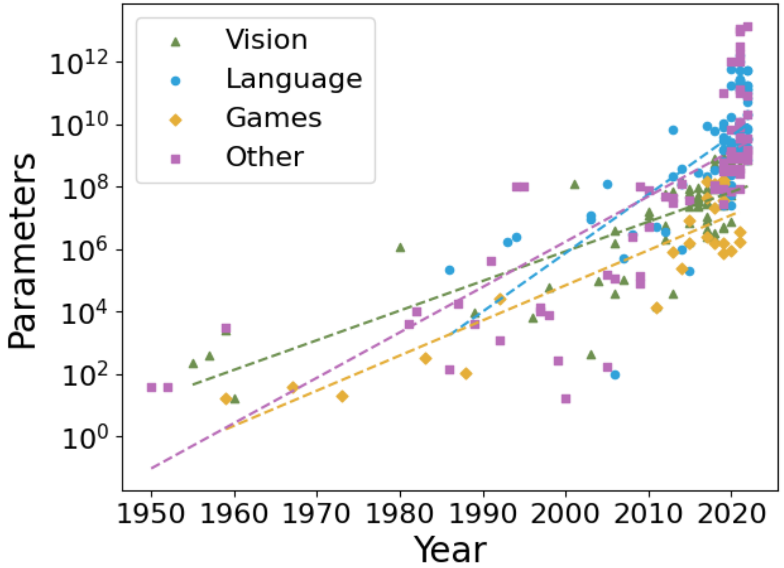
# Relation of Machine Learning and Networking




[1]


[1] Sevilla et al. "*Parameter, Compute and Data Trends in Machine Learning*"

# Relation of Machine Learning and Networking



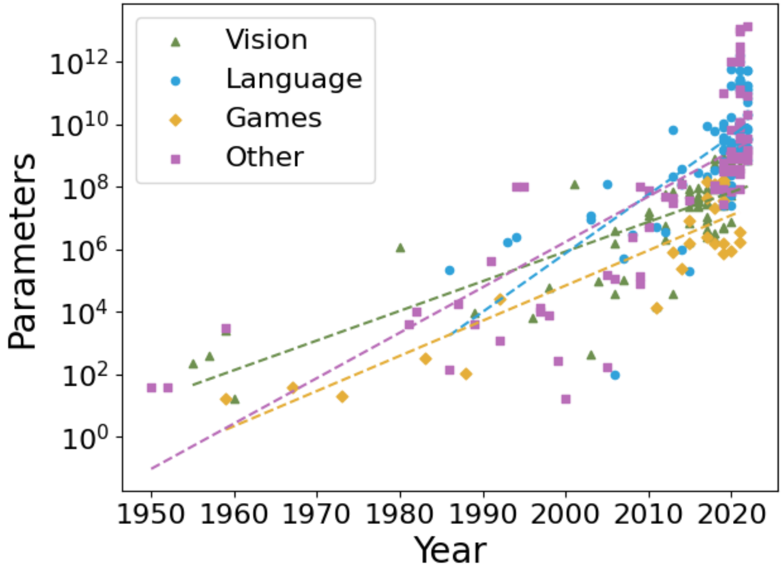

[1]

**Challenges of ML:**

- Number of parameters
- Number of data
- Distributed data


[1] Sevilla et al. "*Parameter, Compute and Data Trends in Machine Learning*"

# Relation of Machine Learning and Networking




[1]

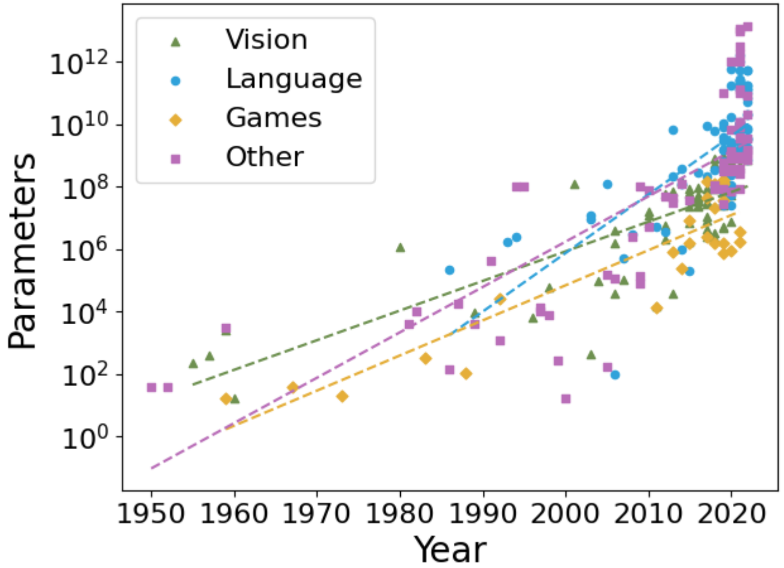
**Challenges of ML:**

- Number of parameters
- Number of data
- Distributed data

Increased training time

[1] Sevilla et al. "*Parameter, Compute and Data Trends in Machine Learning*"

# Relation of Machine Learning and Networking




[1]

**Challenges of ML:**

- Number of parameters
- Number of data
- Distributed data

Increased training time

**How to train such models?**

Parallelize and distribute model training

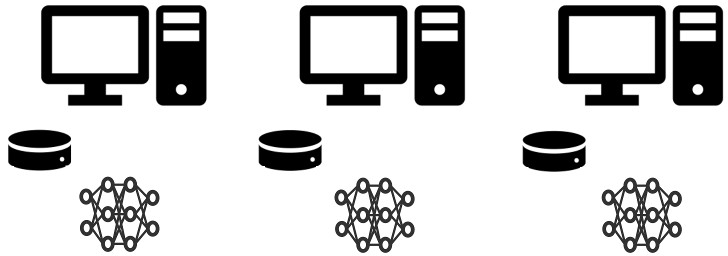
[1] Sevilla et al. "*Parameter, Compute and Data Trends in Machine Learning*"

# Distributed Machine Learning

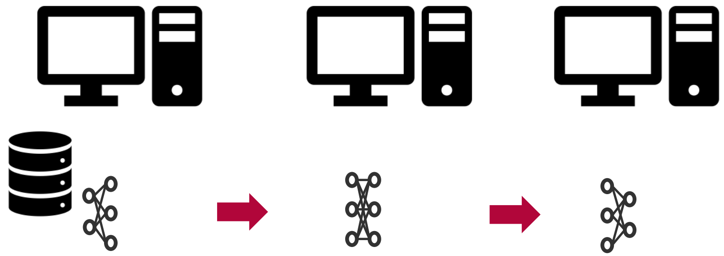
Parallelism of training

1. Model
2. Data
3. Worker

Data-Parallel



Model-Parallel

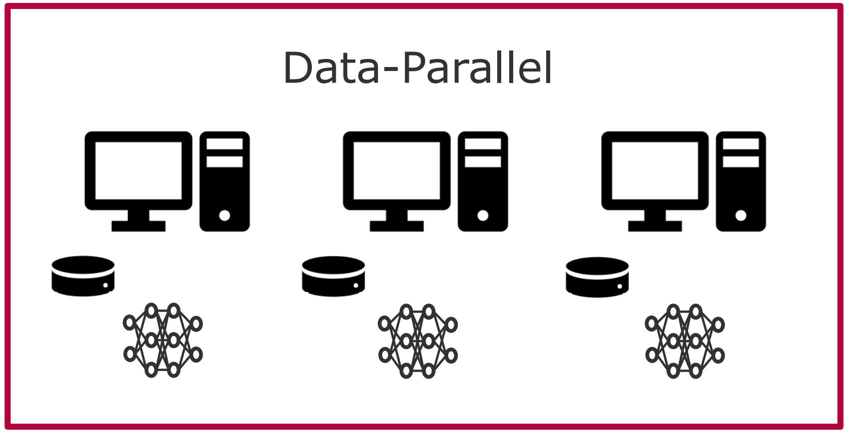


Leonard Paeleke,
PhD Student

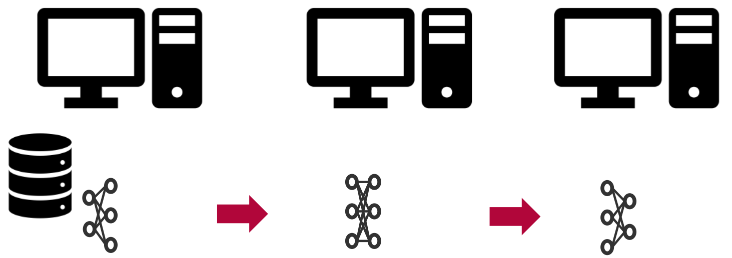
# Distributed Machine Learning

Parallelism of training

1. Model
2. Data
3. Worker



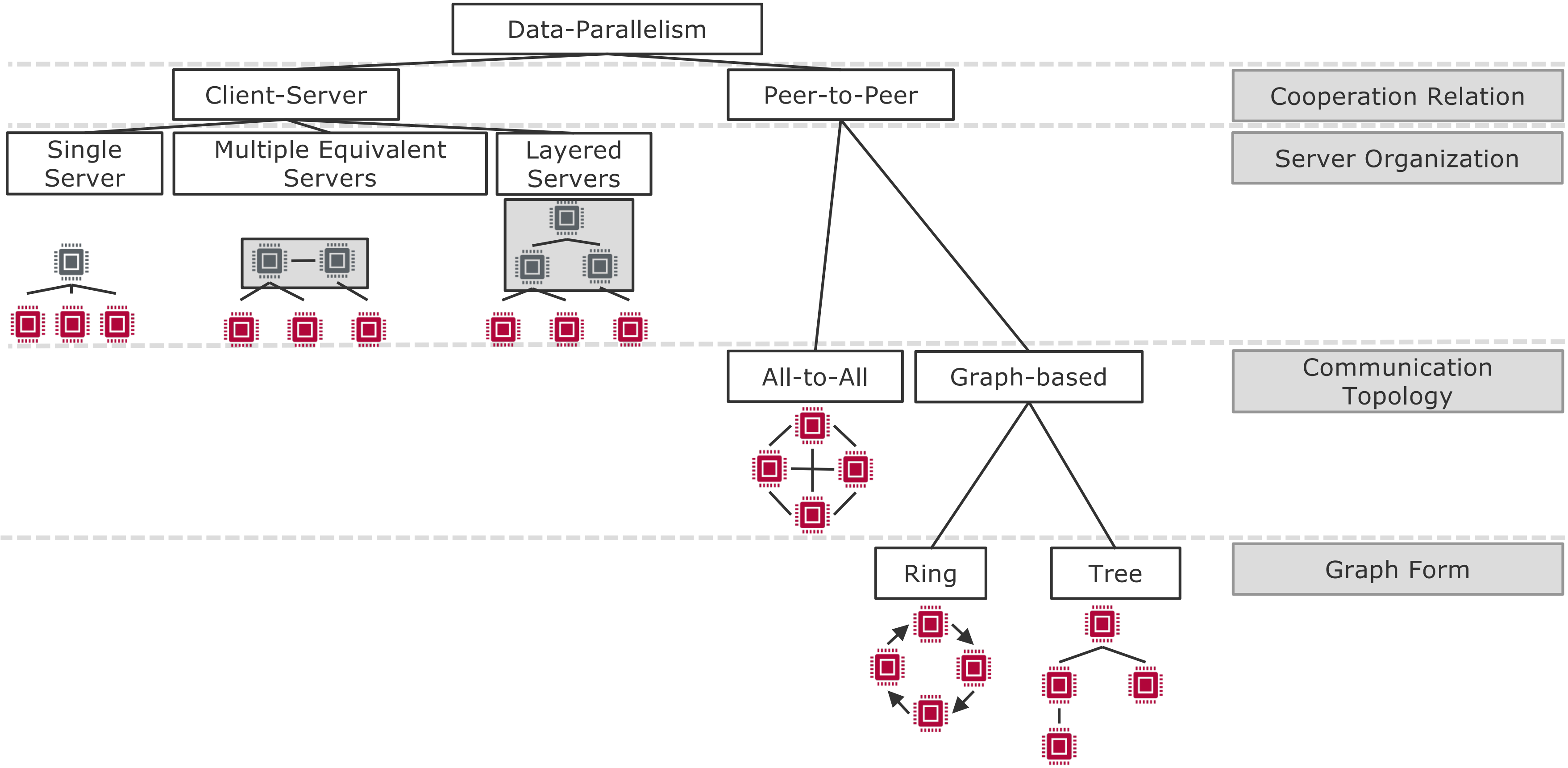

Model-Parallel



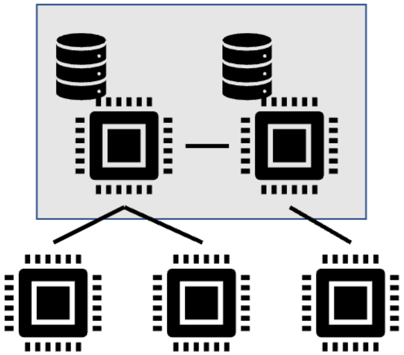
Leonard Paeleke,
PhD Student

# Data-parallel training architectures

From perspective of communication relationship:



# Data-parallel training architectures

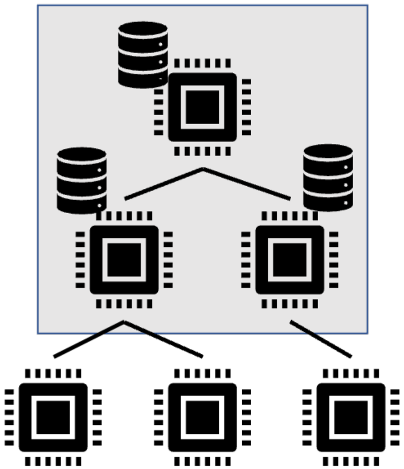


1. Parameter Server
   - Client-Server architecture: *Workers* synchronize via centralized server (*parameter server*)
   - *Parameter server*: maintains and distributes model and data, and aggregates updates
   - *Worker*: processes assigned data, computes updates and sends updates to the *parameter server*

Leonard Paeleke,
PhD Student

# Data-parallel training architectures

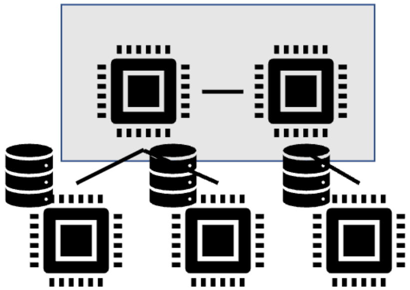
1. Parameter Server
2. In-Network Aggregation
   - Client-Server architecture
   - Network infrastructure act as centralized server and synchronizes workers



Leonard Paeleke,
PhD Student

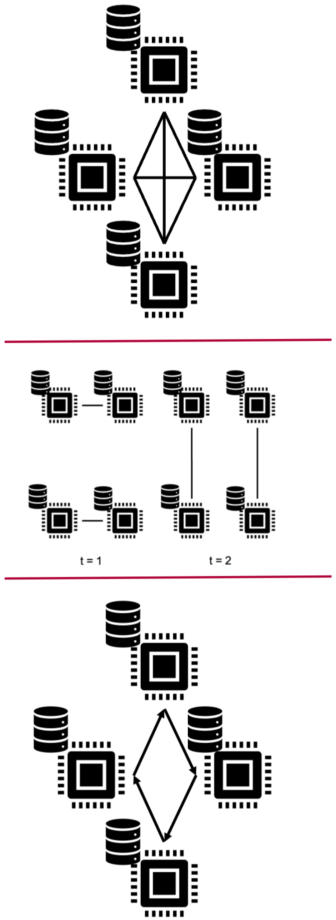
# Data-parallel training architectures

1. Parameter Server
2. In-Network Aggregation
3. Federated Learning
   - Client-Server architecture
   - Data is stored locally and not exchanged
   - Designed for edge-servers and smartphones



Leonard Paeleke,
PhD Student

# Data-parallel training architectures




1. Parameter Server
2. In-Network Aggregation
3. Federated Learning
4. All-Reduce
   - Peer-to-Peer architecture: *Workers* synchronize with peers through direct communication
   - Several communication topologies possible: E.g., All-to-All, Butterfly or Ring.

# Impact of data-parallel architecture on training

Differences of architectures:

- Data distribution
- Straggler handling
- Communication effort
- Computational effort

# Impact of data-parallel architecture on training

Differences of architectures:

- Data distribution
- Straggler handling
- Communication effort
- Computational effort

Impacts on training:

- Order of updates
- Age of information
- Time to convergence
- Energy consumption

Model quality

# Impact of data-parallel architecture on training

Differences of architectures:

- Data distribution
- Straggler handling
- Communication effort
- Computational effort

Impacts on training:

- Order of updates
- Age of information
- Time to convergence
- Energy consumption

Model quality

How to select an architecture for a learning task?

# Impact of data-parallel architecture on training

Differences of architectures:

- Data distribution
- Straggler handling
- Communication effort
- Computational effort

Impacts on training:

- Order of updates
- Age of information
- Time to convergence
- Energy consumption

Model quality

How to select an architecture for a learning task?

Common metrics:

- <u>Model quality</u>
- <u>Training time</u>
- Resource consumption
- Energy consumption
- Green house emission

# Survey - Approach

- Studies evaluate training of ML models for several learning tasks and data-parallel architectures
- Metrics: Model quality or training time

TABLE I: Model quality of distributed ML architectures on typical ML applications

| | Parameter Server | In-network Aggregation | Federated Learning | All-reduce approach |
|---|---|---|---|---|
| img trans | | | | |
| img class | o [27] + [6] o [23] - [24] | o [36] + [19] | + [13] o [6] + [5] | o [13] o [27] o [23] + [24] |
| obj det | | - [36] | + [10] | |
| lang mod | + [27] + [39] o [23] | + [36] | + [5] | - [27] + [39] o [23] |
| quest answ | + [11] | + [36] | | |
| time-series pred | | | | |
| click-through pred / Recommendation | + [44] | + [36] | | |
| RL | o [22] | o [22] | | |
| channel decision | + [10] (Beam-selection) | | | |
| signature class | | | | |
| anomaly class | | | | |
| code gen | | | | |
| lang class (SQL intrusion) | | + [29] | | |

TABLE II: Training time of distributed ML architectures on typical ML applications

| | Parameter Server | In-network Aggregation | Federated Learning | All-reduce approach |
|---|---|---|---|---|
| img trans | - [16] | o [36] | | o [36] o [16] |
| img class | - [27] + [6] o [16] - [23] | o [36] + [19] | + [13] o [6] + [5] | o [36] o [13] + [27] + [16] + [23] |
| obj det | | o [36] | | o [36] |
| lang mod | - [27] + [39] o [16] | + [36] | + [5] | - [36] + [27] - [39] + [16] |
| quest answ | + [11] | + [36] | | - [36] |
| time-series pred | | | | |
| click-through pred/ Recommendation | + [44] | + [36] | | - [36] |
| RL | - [22] | + [22] | | o [22] |
| channel decision | | | | |
| signature class | | + [13] | | o [13] |
| anomaly class | | | | |
| code gen | | | | |
| lang class (SQL intrusion) | | | | |

Leonard Paeleke,
PhD Student

# Survey - Results

Model quality:

- In many studies Client-Server architectures outperform Peer-to-Peer
- Federated Learning rarely compared to other architectures

Training time:

- Different measures used, e.g., wall-clock time and CPU time
- Improvements by reducing:
  - Communication overhead
  - Waiting time for stragglers
  - Time for parameter aggregation

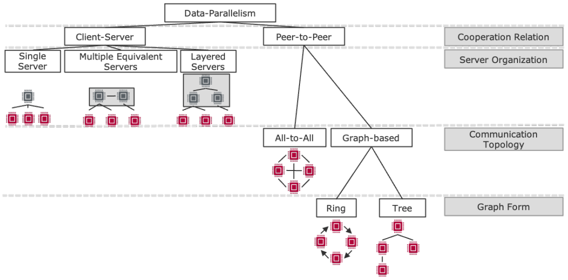
# Survey - Results

Challenges:

- Comparability between studies
- Lack of definitions, e.g., completion of training

Suggestion:

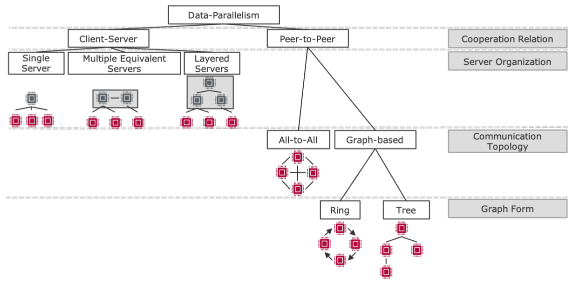
- Reference learning task with predefined model, hyperparameters, data set, fixed train-test split, and measurement for model quality
- Definition of training time
- Normalizing testbed by speed of CPUs/ GPUs





Leonard Paeleke

PhD Student

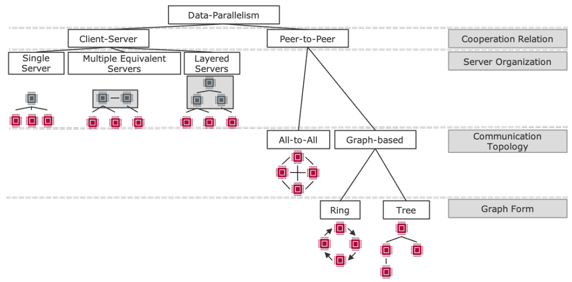



Architecture impact training in

- Order of updates
- Age of information
- Time to convergence
- Energy consumption

Leonard Paeleke

PhD Student



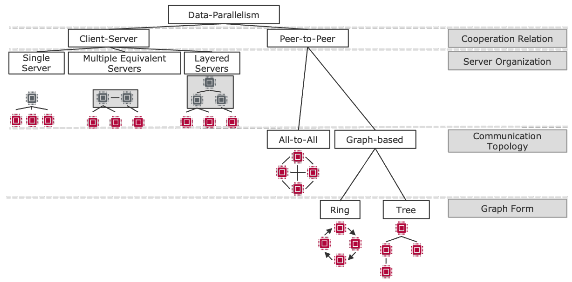

Architecture impact training in

- Order of updates
- Age of information
- Time to convergence
- Energy consumption

Open question:

- Which data-parallel architecture to choose for a learning task?

Leonard Paeleke
PhD Student




Architecture impact training in

- Order of updates
- Age of information
- Time to convergence
- Energy consumption

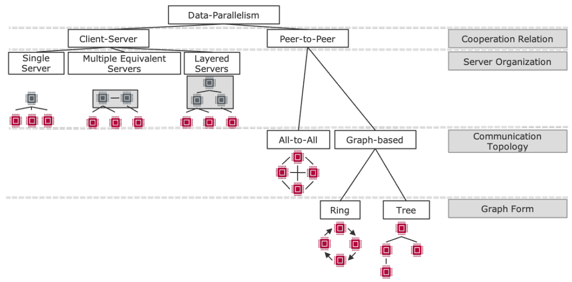
Open question:

- Which data-parallel architecture to choose for a learning task?

Identified problem:

- Comparability of studies


Leonard Paeleke
PhD Student




Architecture impact training in

- Order of updates
- Age of information
- Time to convergence
- Energy consumption

Open question:

- Which data-parallel architecture to choose for a learning task?

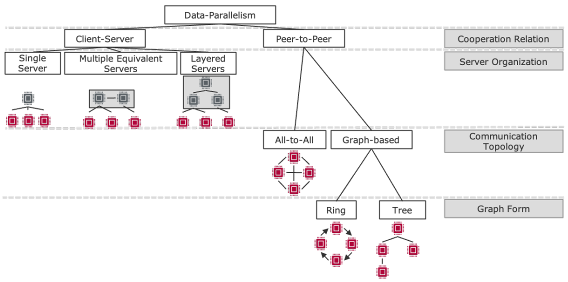
Identified problem:

- Comparability of studies

Suggestion:

- Reference testbed
- Normalization of training

Leonard Paeleke
PhD Student




Architecture impact training in

- Order of updates
- Age of information
- Time to convergence
- Energy consumption

Open question:

- Which data-parallel architecture to choose for a learning task?

Identified problem:

- Comparability of studies

Suggestion:

- Reference testbed
- Normalization of training

Thank you for your attention!

Leonard Paeleke
PhD Student