Handling Workflows complexity

Workflow Community Summit 2025

Elena Lazovik |

TNO BigDataEurope2023

Self.me

Universitá degli Studi di Trento,

RuG

Distributed Systems and SW architectures

Law & Computing Science

Elena Lazovik

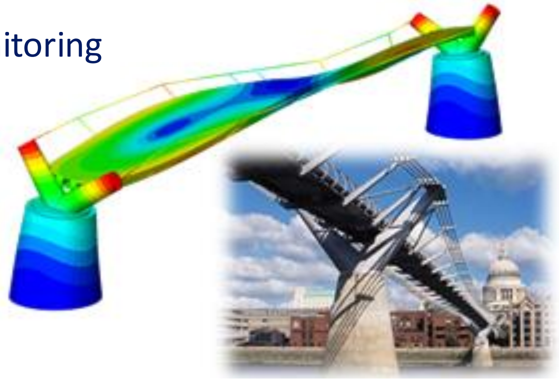
Scientist Specialist on Data-driven complex digital systems at TNO ICT | Big Data | AI on Big Data | Digital Twins | Adaptive IT

https://www.linkedin.com/in/elenalazovik/

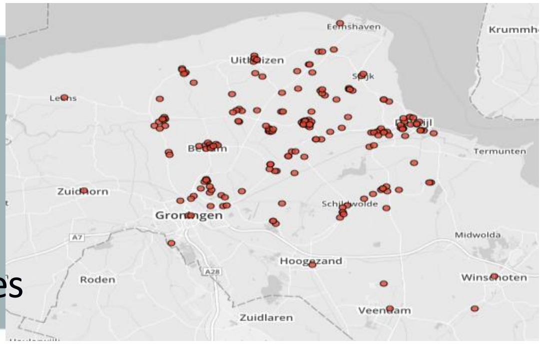
Senior Scientist,

13 years experience with Big Data and workflows

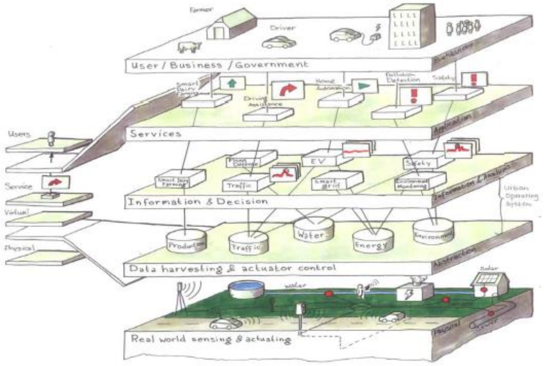
Bridges monitoring



Earthquakes real-time monitoring: 350 buildings in Groningen (NL)

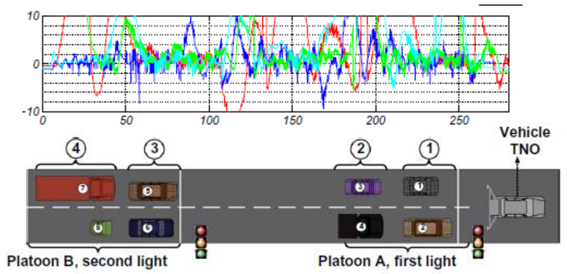


Live monitoring of water and gas pipes for whole NL



BIODT
biodiversitydigitaltwin

Cooperative automated driving



Dikes monitoring

Can sensors warn for dike failure?

Workflow definition

A **workflow** is a repeatable pattern of activity enabled by the systematic organization of resources into processes that transform inputs into outputs.

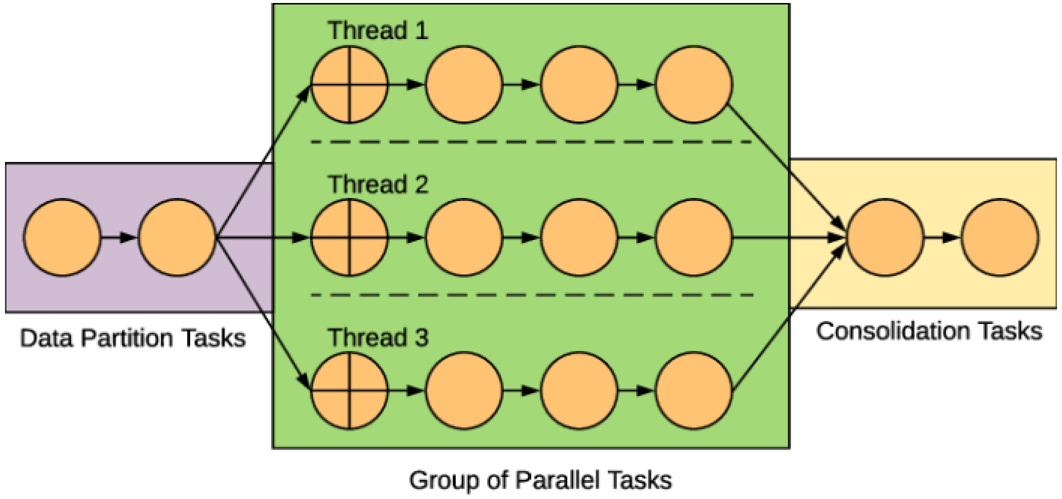
Key Elements of a Workflow:

1. **Tasks/Activities** – Individual steps in the process.
2. **Sequence/Flow** – The order in which tasks are performed.
3. **Roles/Actors** – People, systems, or services responsible for executing the tasks.
4. **Rules/Conditions** – Logic that determines how and when transitions happen between tasks.
5. **Input/Output** – Data or material required or produced by each task.

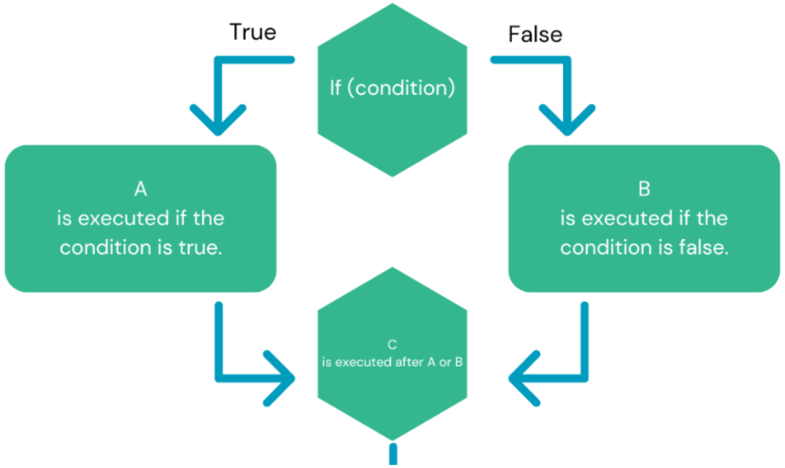
Classical categorization of the workflows

•**Sequential**: Tasks occur one after another.

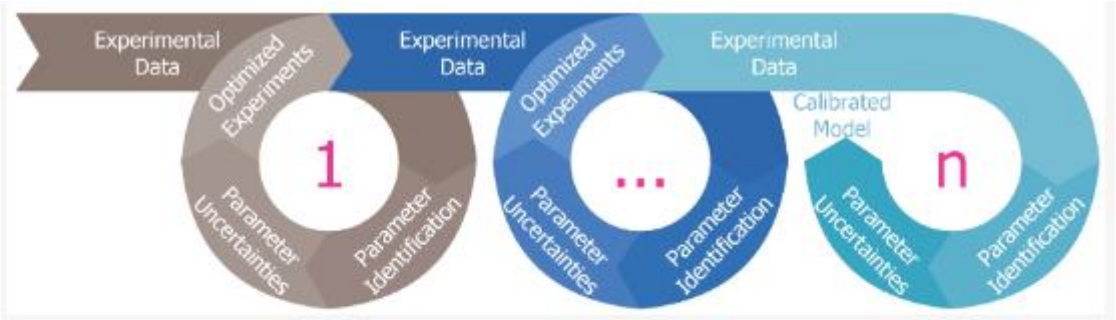
•**Parallel**: Tasks can be performed concurrently.



•**Conditional**: The path of the workflow depends on decisions or conditions.



•**Iterative**: Certain steps repeat based on conditions or until a goal is reached.



•**Special case – (co)simulation**: Certain steps until a goal is reached.

Classification for complexity of handling

The easiest **parallel** workflow, where only the results need to be merged:

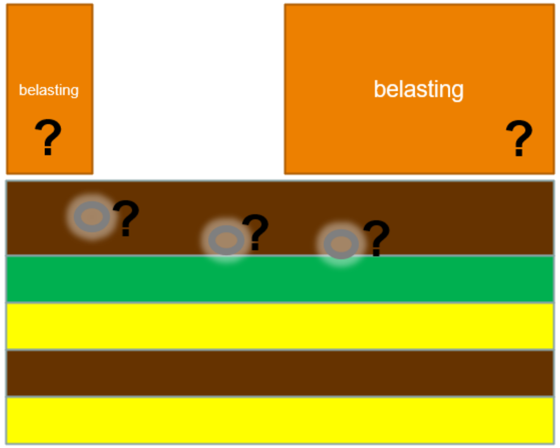
it requests extra handling of distributed data storage and efficient memory and cache mechanisms

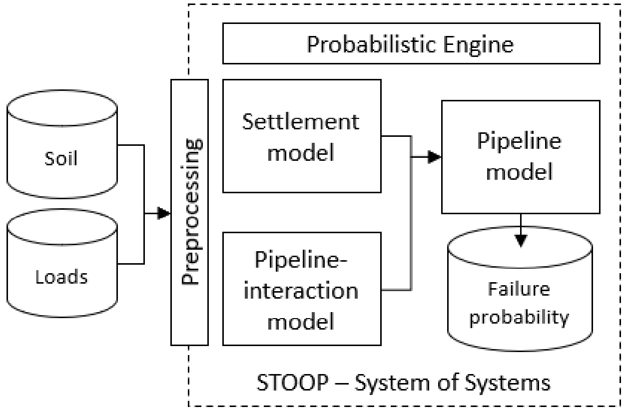
Second place on easiness: simple pipeline – **sequential** workflow:

Requires the same handling as previous, plus the right order of the tasks, and proper organization of the communication with the intermediate results between processing steps

Complexity explodes when the workflows become conditional and/or iterative

Example of conditional workflow

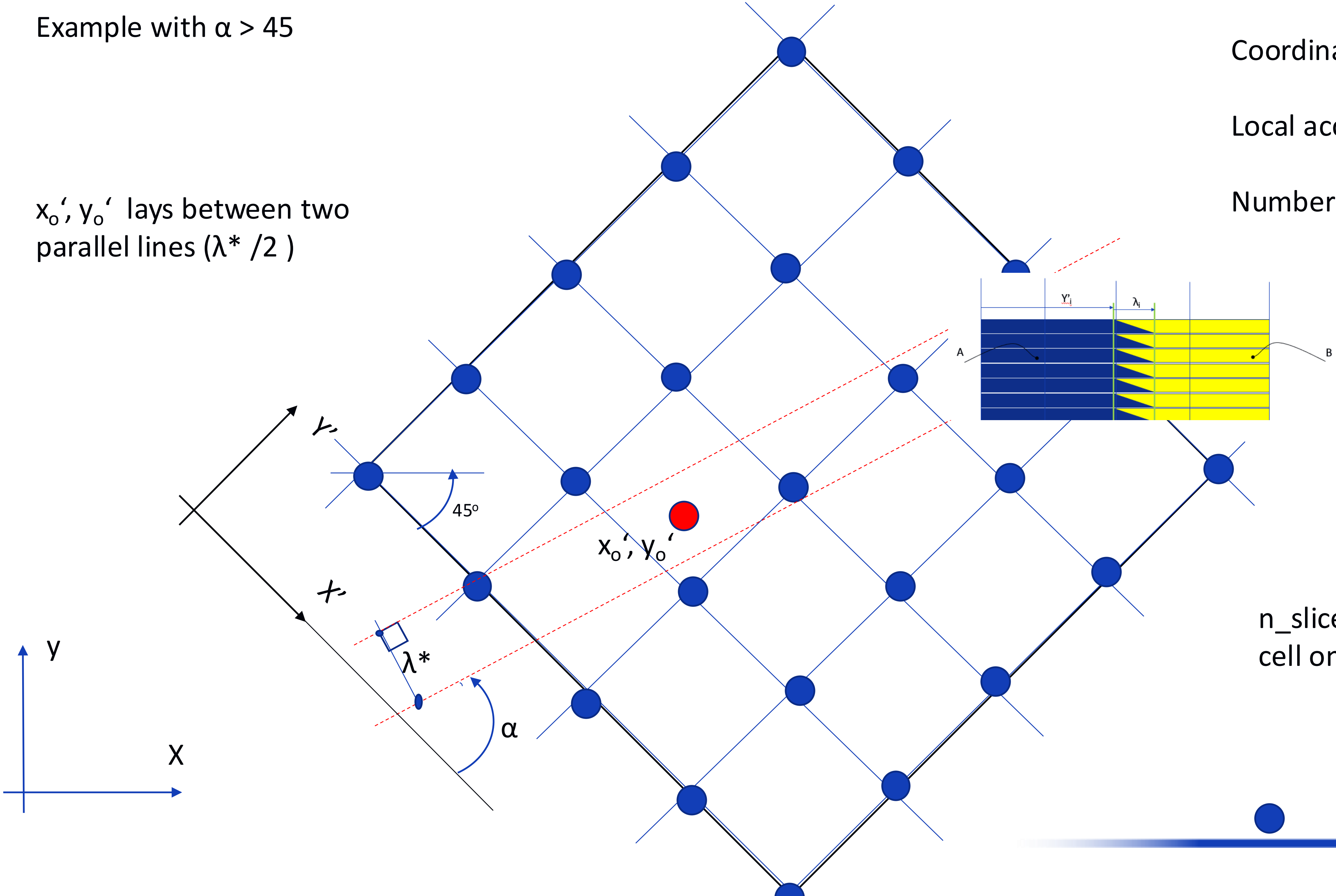




- Predicting chance of failure of all water and gas pipelines in the Netherlands for next 30 years
- Sensitivity analysis and Monte Carlo scenario
- Combination 10 models from different organizations
- 400 combinations of stochastic parameters

Example with $\alpha > 45$

x_o', y_o' lays between two parallel lines (λ^* /2)



Coordinates x, y

Local accent x' y'

Number of slices n_slices

n_slices – grid for 70x70 m cell on the ground

Autoflow – sensitivity analysis

Automated DSettlement input (XML)

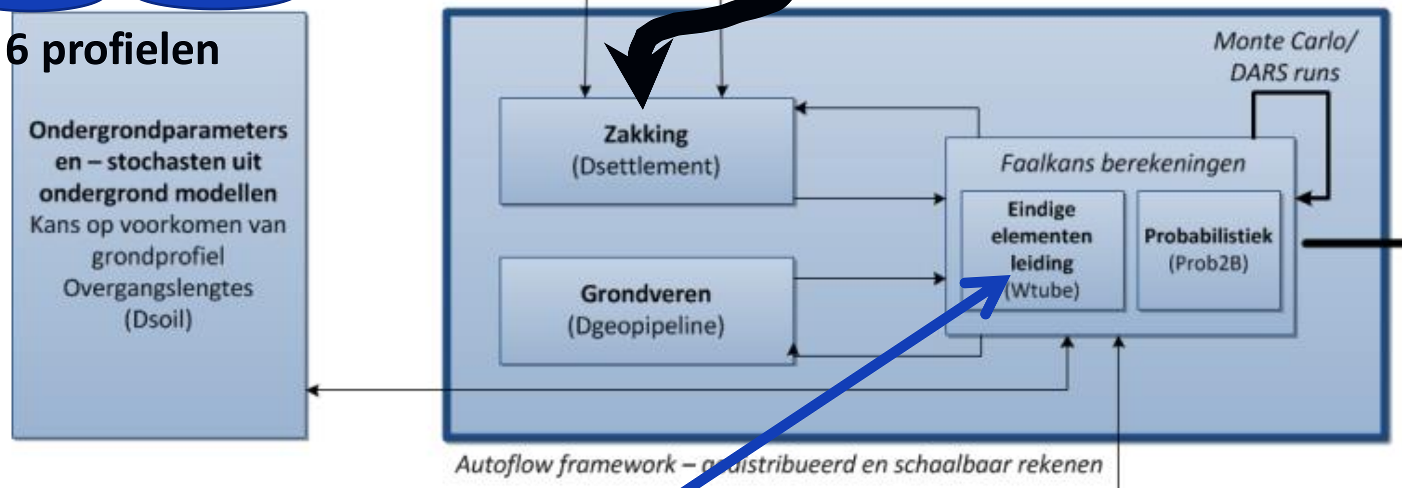
-1m above layer

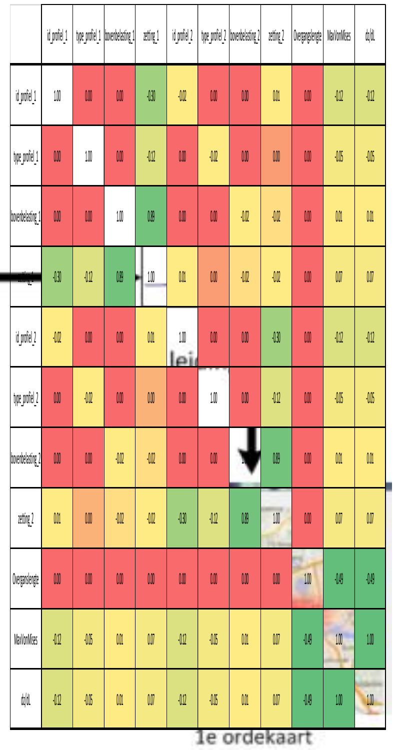
[0.1,2m]

GeoTop

Polygone

6 profielen

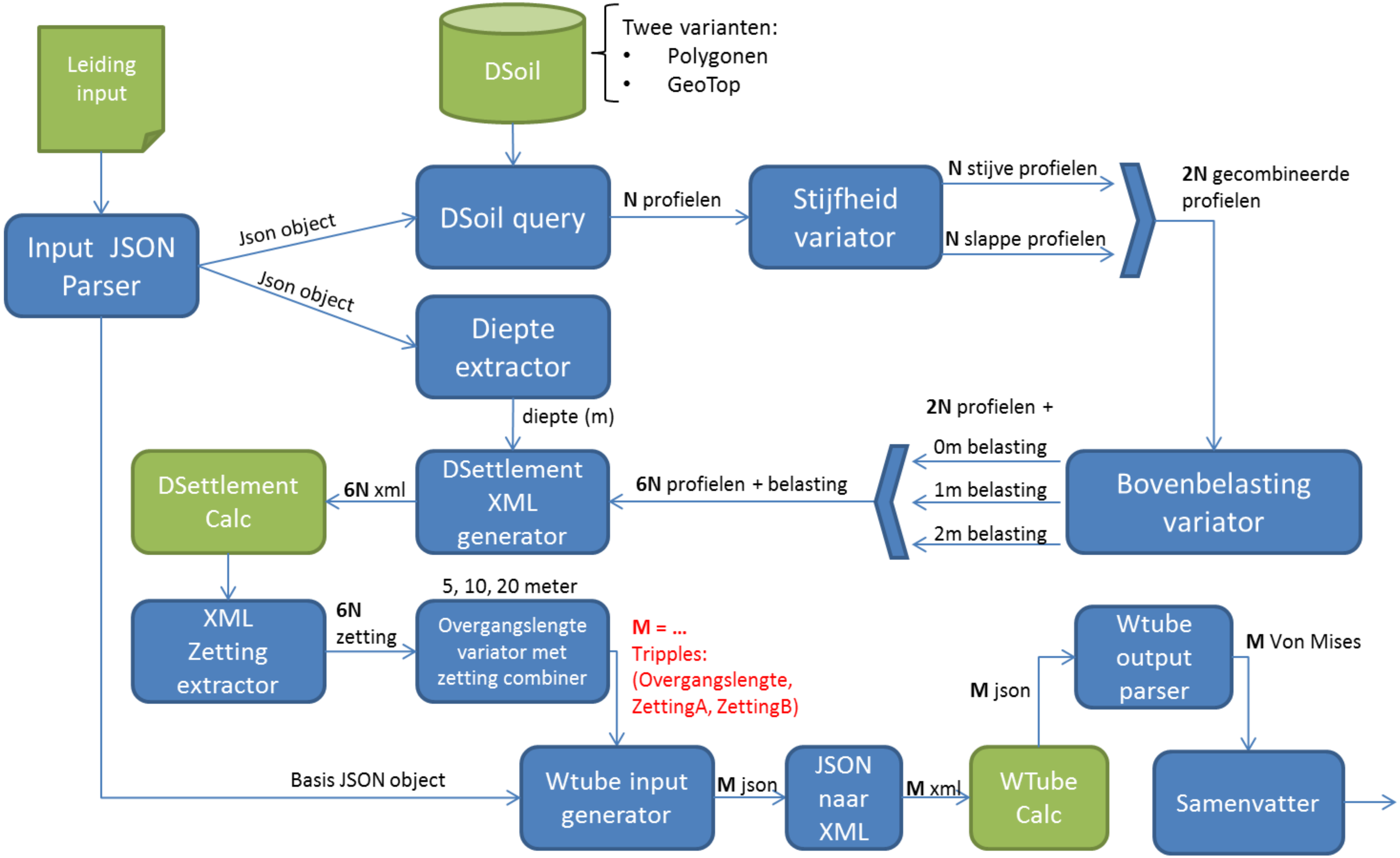




Automated input voor WTube
Overgangs-lengte [5,10,20m]

HDPE, D=110 mm, t=6.3 mm
Staal, D100 mm, t=3.6 mm

Logical workflow – Sensitivity analysis



2N x M profiles

400 parallel branches

Addition of iterative workflow

- From previous workflow to Monte Carlo probabilistic calculations for thousands of grid cells with 400 probabilistic parameters

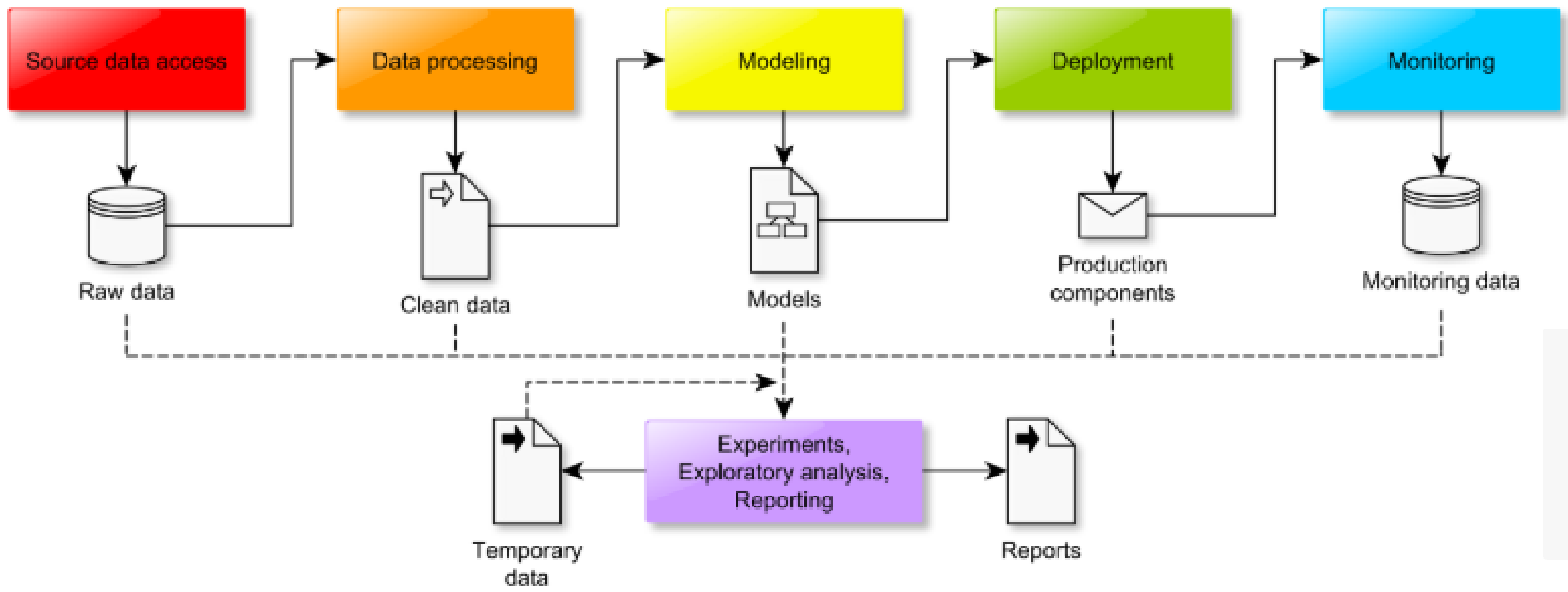
- Used Monte Carlo simulations to accelerate physics based model computations

- Explored 400 combinations of stochastic parameters in parallel

- Enabled efficient exploration of different system behaviors despire brute-force nature

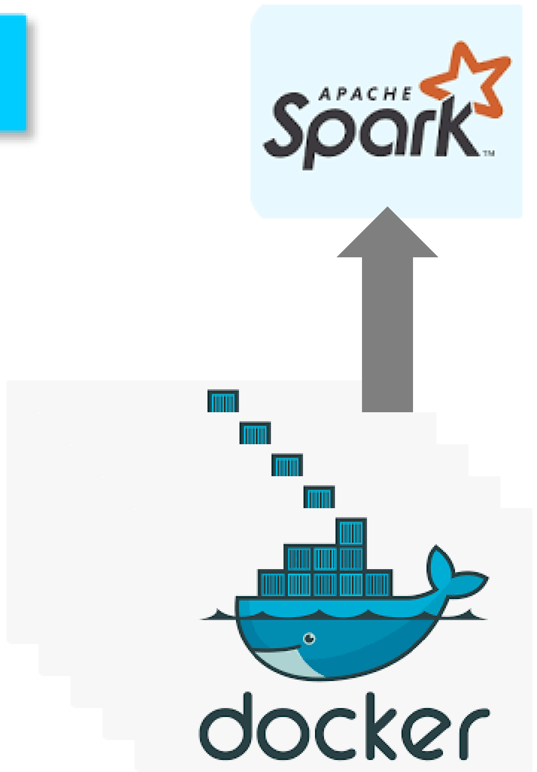
- Achieved massive performance gains through parallel processing
  - Reduced runtime from 1/2 year to 4 days

Composable workflows and monitoring

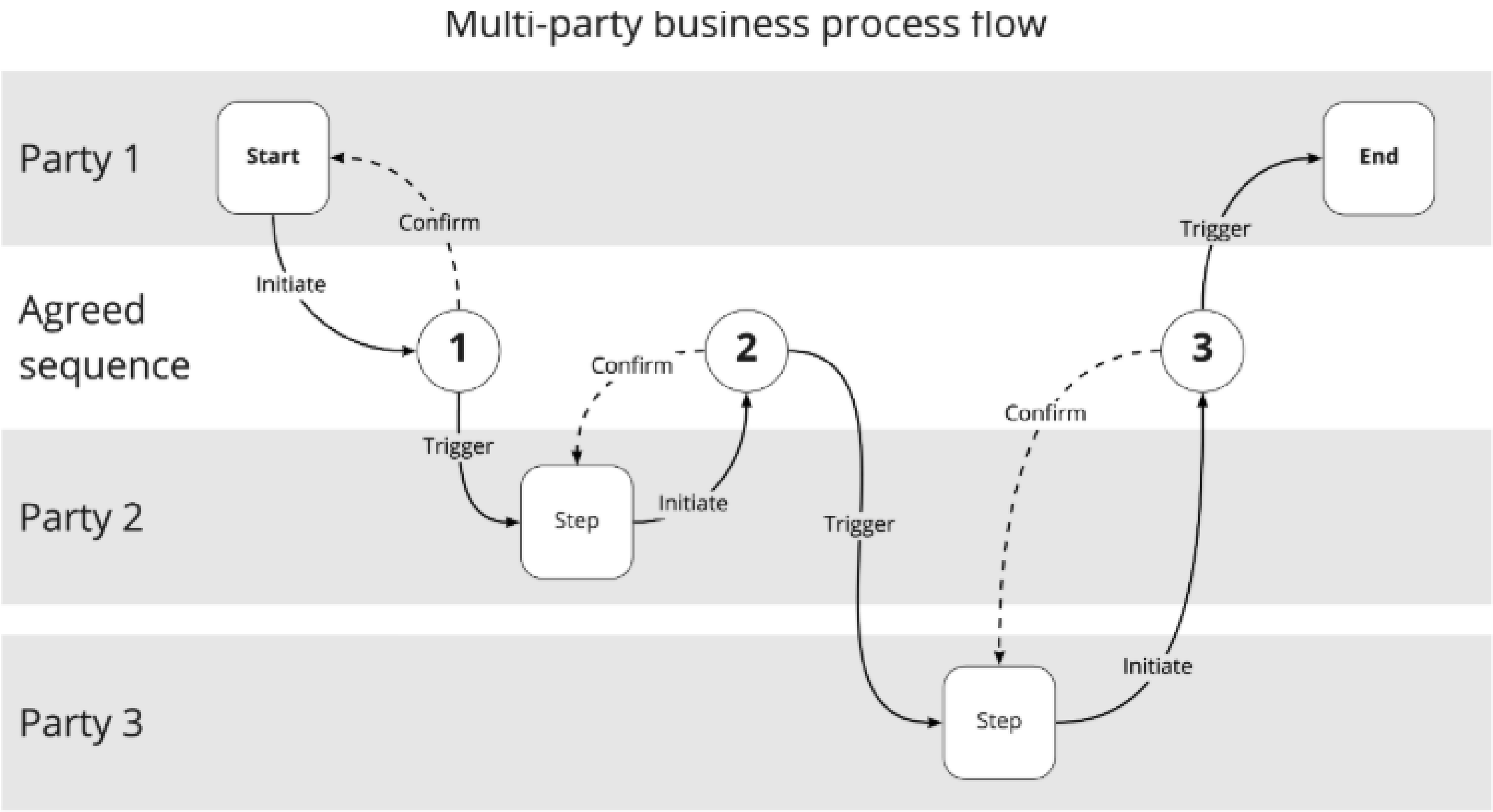


Convergence

controller



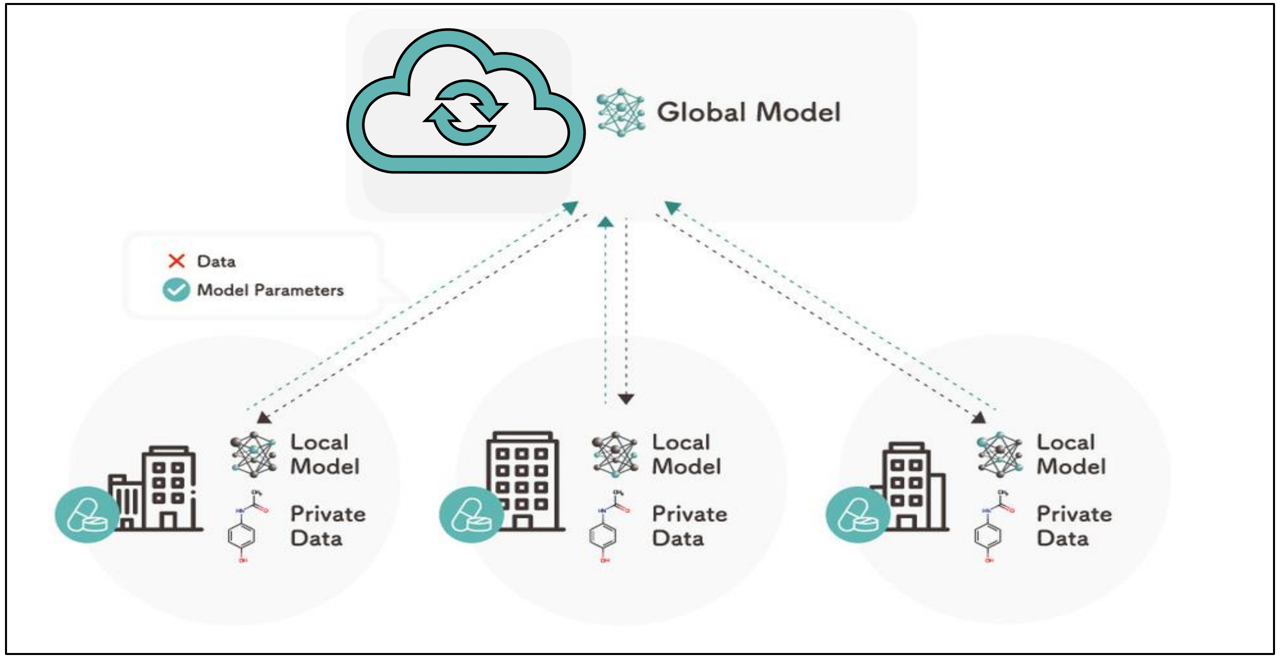
Extra complexity: multi-party workflows



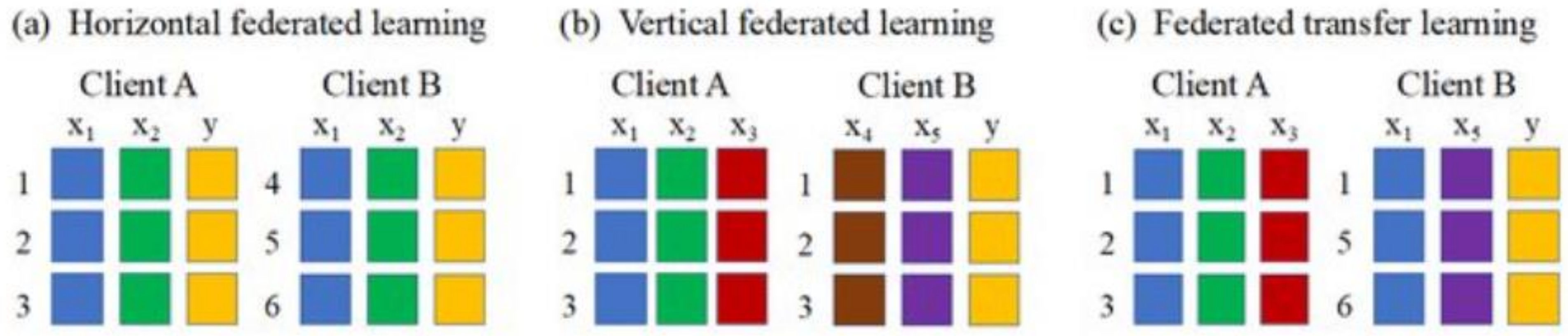
Workflows with IoT devices: Federated Learning in short

A decentralized Machine Learning (ML) technique

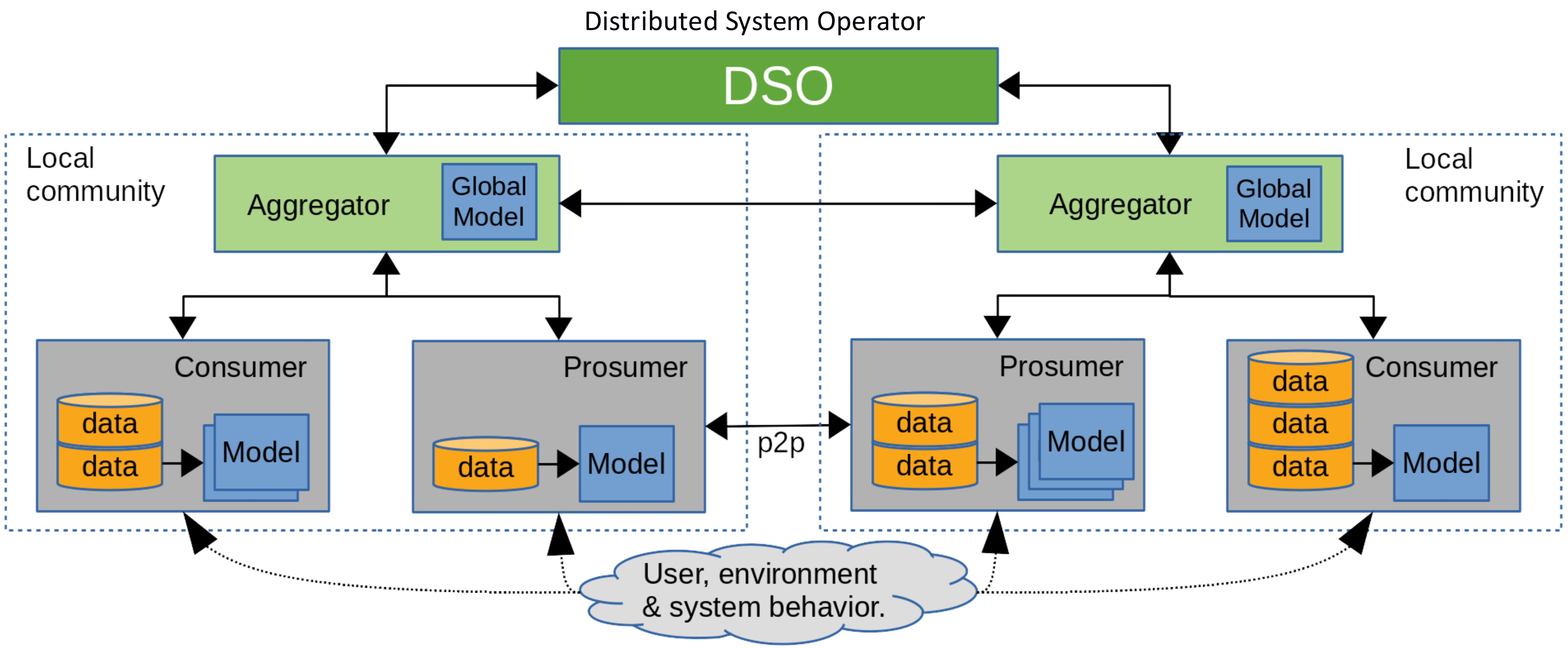
- Uses local information to build global knowledge, without disclosing sensitive data
- Enables learning a single model across multiple distributed devices using local data samples
- Allows privacy-preserving data aggregation from decentralized nodes



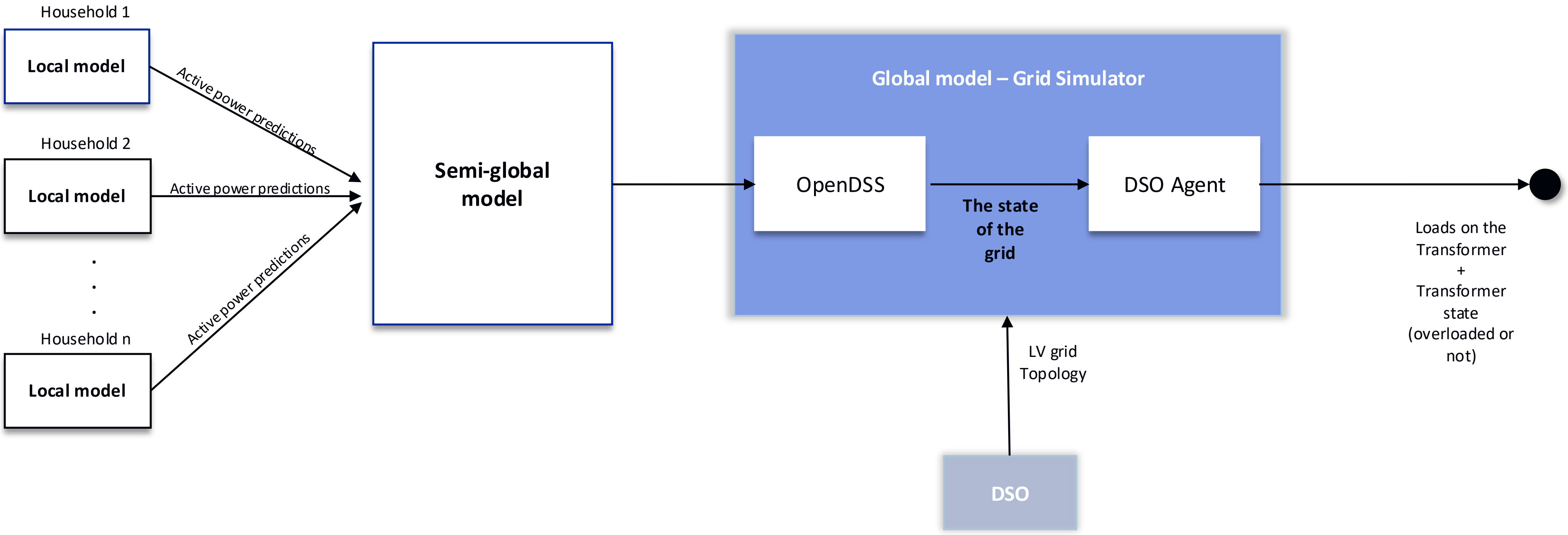
Types of the Federated Learning



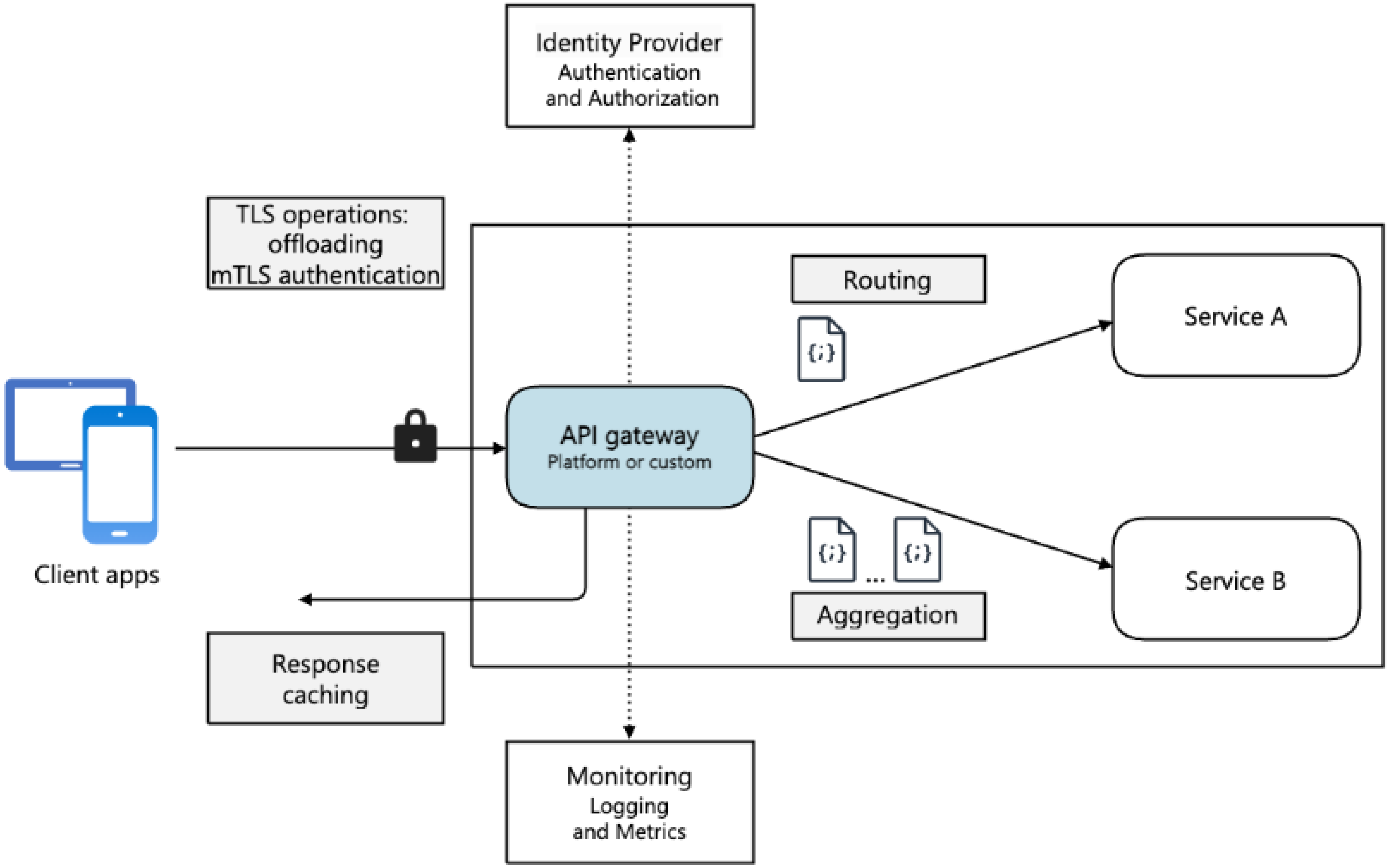
Federated Learning for LV grid state estimation



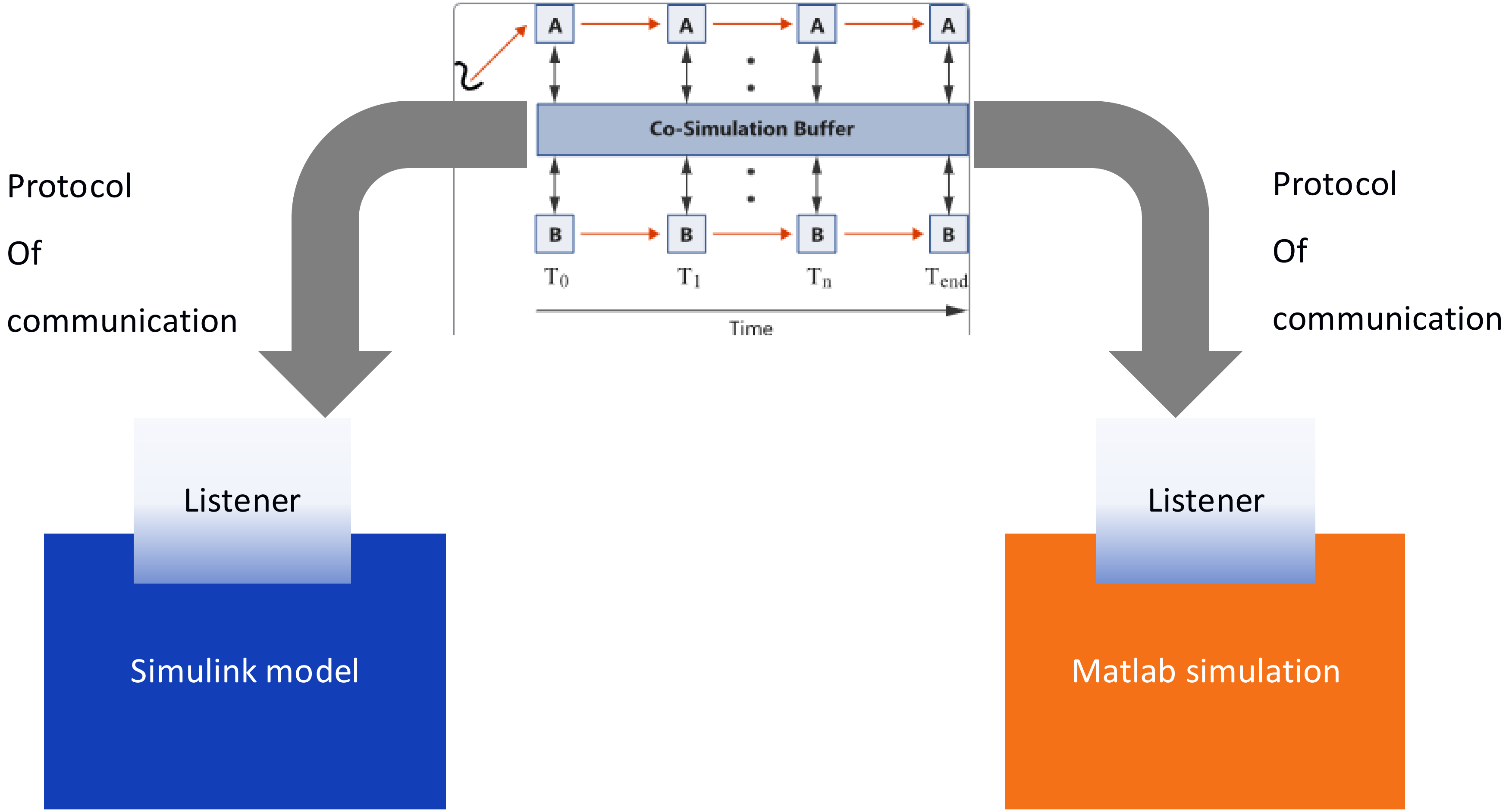
FL platform for LV grid state estimation



Multi-party: extra processes to handle



Co-simulations



Conclusion

Workflows are complex

Complexity explodes for conditional and iterative workflows

Multi-party workflows lead to handling of external data exchange

Complexity will grow even more in future because of the interconnectedness and servitization

All of that requires **flexible** architectures and implementation of workflow engines for future-proof computing

Classification from digital infrastructure perspective

Workflow Type	Characteristics	Key Infrastructure Needs
AI/ML	Compute-intensive, real-time or batch	GPU/TPU, orchestration, edge/cloud coordination
Data Processing	Batch/stream hybrid, high-volume	Distributed storage, high I/O, stream support
HPC/Scientific	Tight coupling, massive parallelism	HPC schedulers, fast interconnects, large memory nodes
CI/CD & DevOps	Fast iteration, automation	Containers, pipelines, observability
Web/API	Stateless, scalable, service-oriented	Load balancing, API management, scaling policies
Edge/IoT	Decentralized, low latency, disconnected	Lightweight runtimes, local storage, OTA updates
Secure/Regulated	High trust, restricted data	Isolation, encryption, auditing
Enterprise/Business	Integration-heavy, stable, governed	Legacy support, metadata management, compliance tools

Thank you for the attention!

elena.lazovik@tno.nl

https://www.linkedin.com/in/elenalazovik/