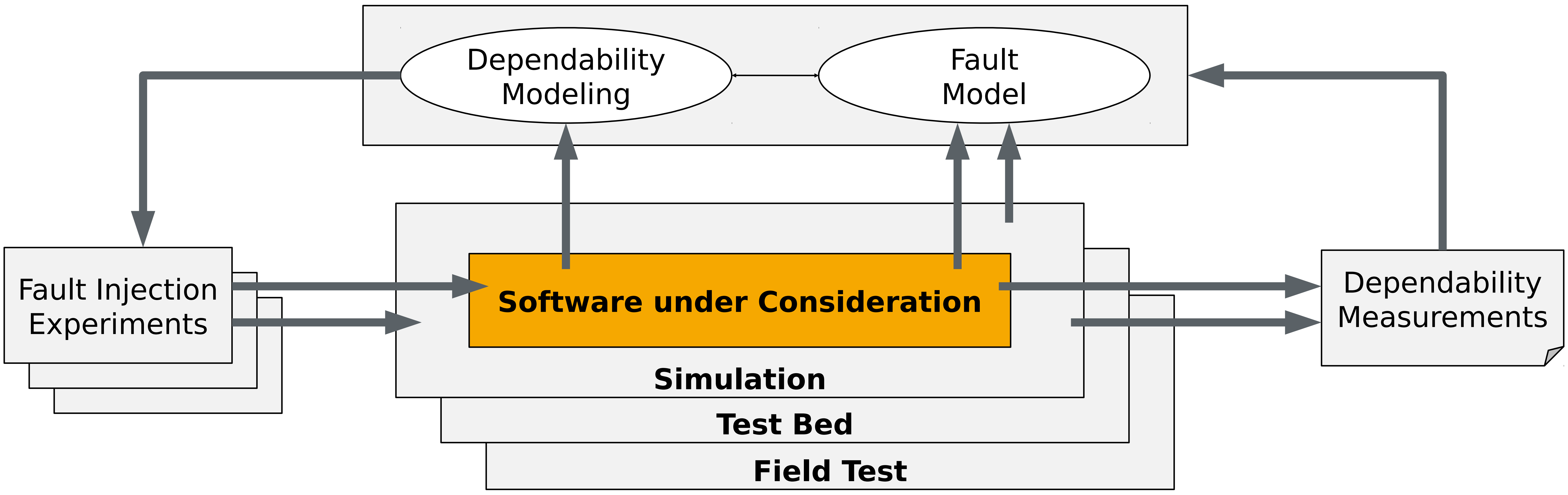
Fault-Injection-Driven Development

Foto: DPA


Dependability Modeling
Fault Model
Fault Injection Experiments
Software under Consideration
Dependability Measurements
Simulation
Test Bed
Field Test

Why?

- ■ Software complexity is increasing drastically.
- ■ Dependability is now an essential non-functional requirement for most distributed software systems.
- ■ Current approaches for increasing software dependability suffer from practical limitations (formal methods) or are subject to developers' biases (testing) and are thus unsuited for complex failure scenarios.

Dependability has to become a first-class concern in software development processes.

How?

1 Modeling assumptions: creating a dependability and a failure cause model.

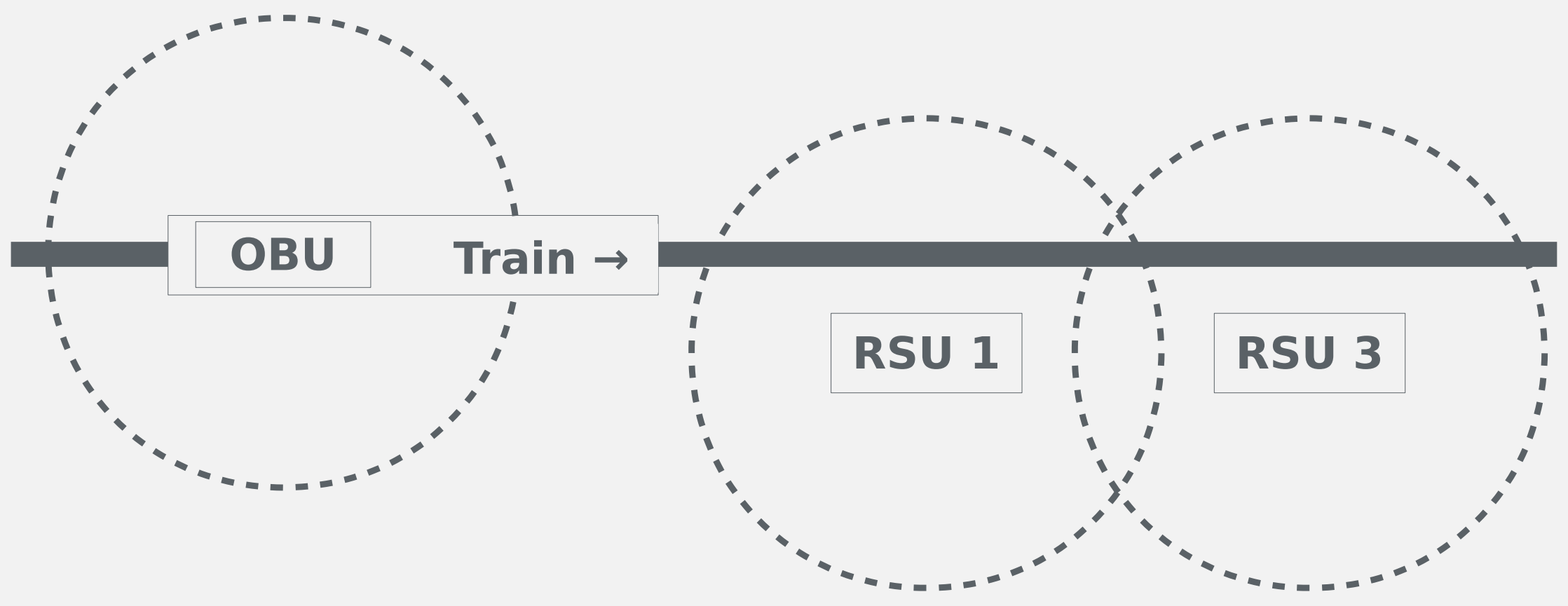
2 Software fault injection: exercising scalable and automated experiments.

> The injection locations are derived from the dependability model.
>
> The fault load is derived from the failure cause model.

3 Feedback loop: enhance the software under consideration itself and the previously created models.

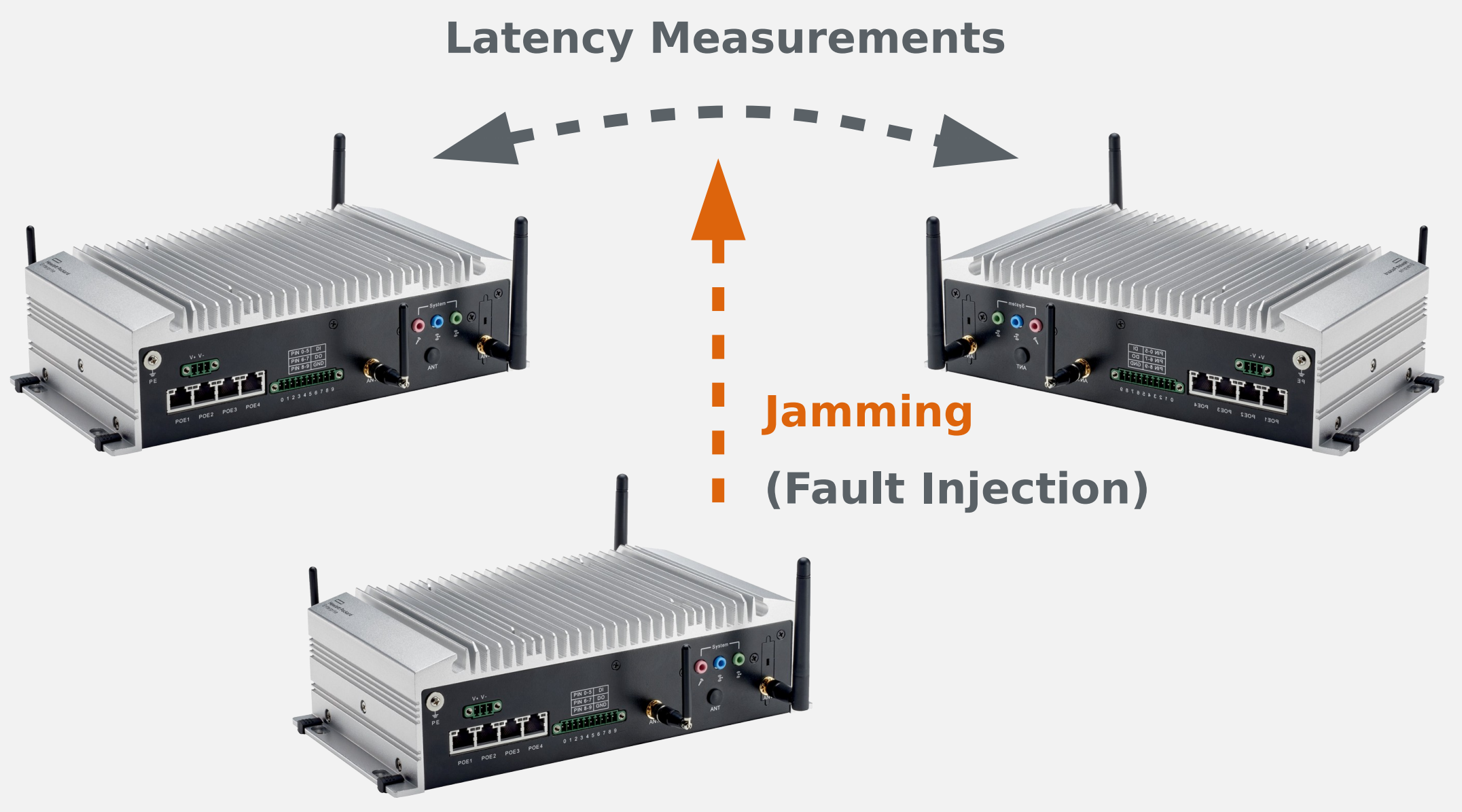
Simulation

using the network simulator ns3



Test Bed

in the IoT Lab at HPI



Field Test

at the Erzgebirgsbahn (BSg)

RAIL2X

Project Partners:

DB SIEMENS DLR Dralle Systementwicklungen

Operating Systems and Middleware Group
Prof. Dr. Andreas Polze
Lena Feinbube
Jossekin Beilharz
Lukas Pirl
Rail2X@hpi.de

Hasso Plattner Institute for Digital Engineering gGmbH
Prof.-Dr.-Helmert-Str. 2-3
14482 Potsdam

HPI Hasso Plattner Institut
Digital Engineering · Universität Potsdam