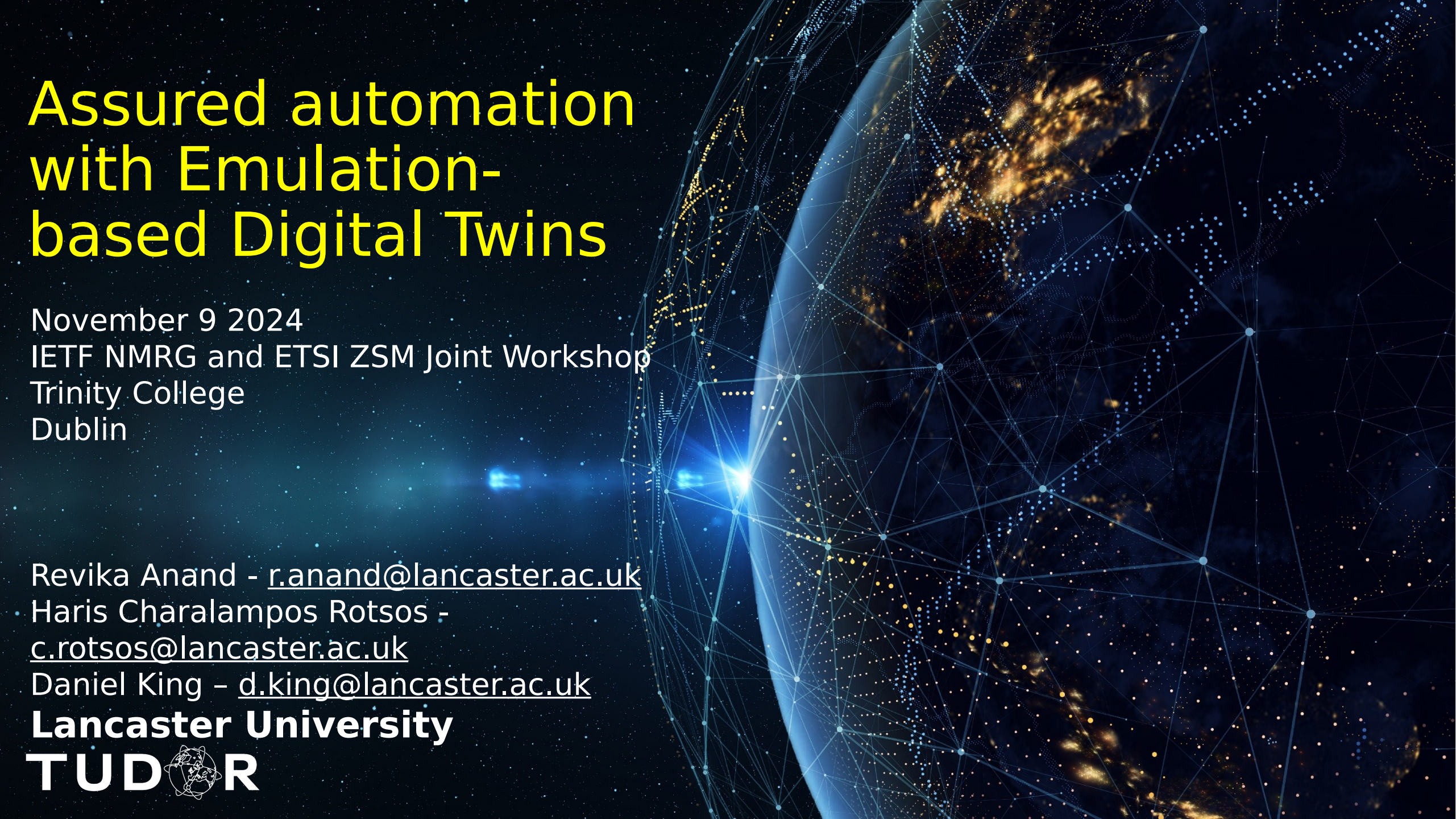
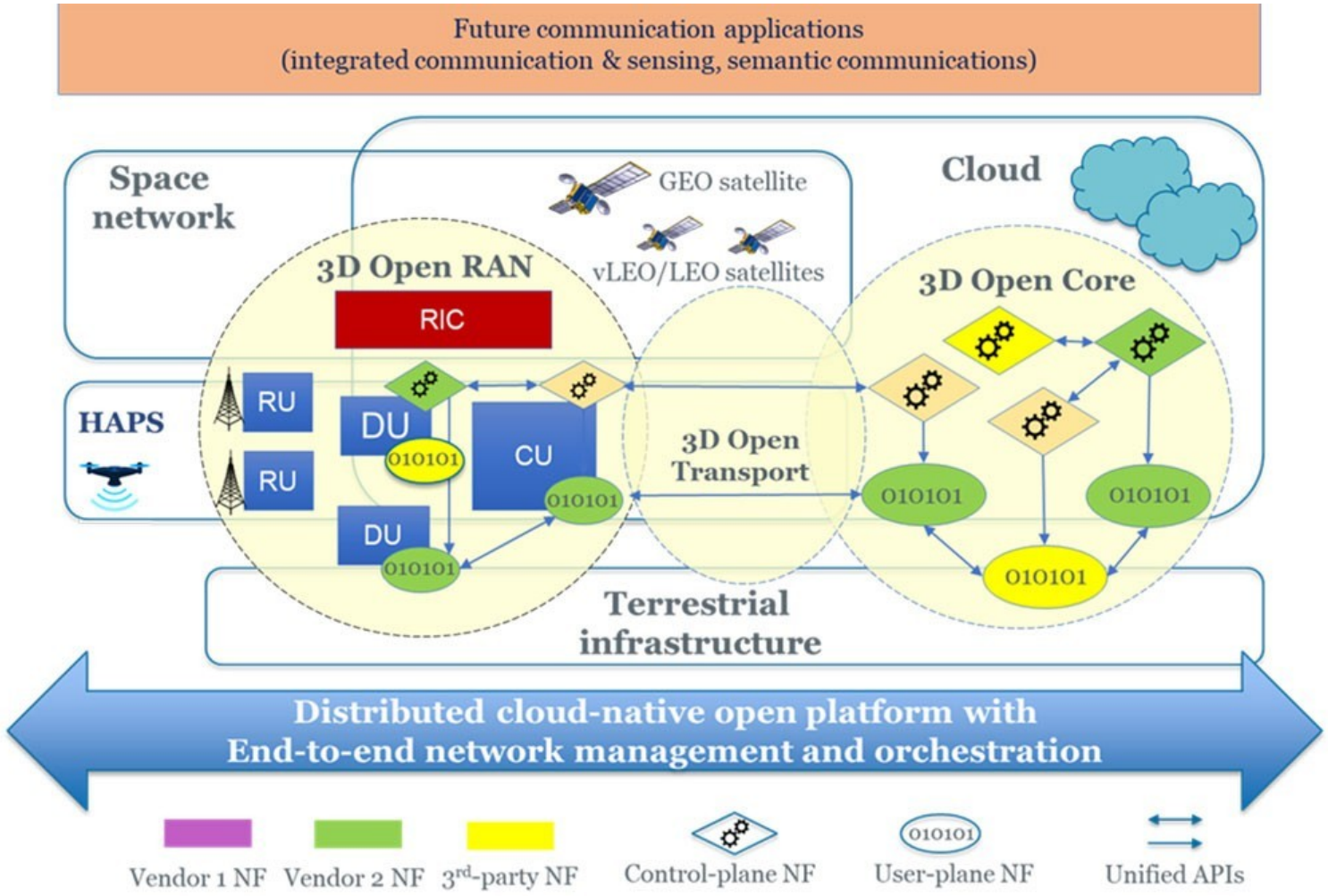
Assured automation with Emulation-based Digital Twins

November 9 2024
IETF NMRG and ETSI ZSM Joint Workshop
Trinity College
Dublin



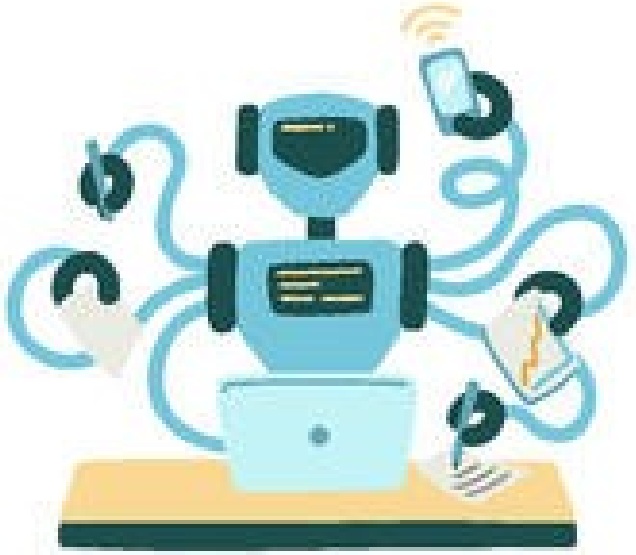
Revika Anand - r.anand@lancaster.ac.uk
Haris Charalampos Rotsos - c.rotsos@lancaster.ac.uk
Daniel King – d.king@lancaster.ac.uk
Lancaster University
TUDOR

TUDOR - towards-ubiquitous-3d-open-resilient-network/our-approach



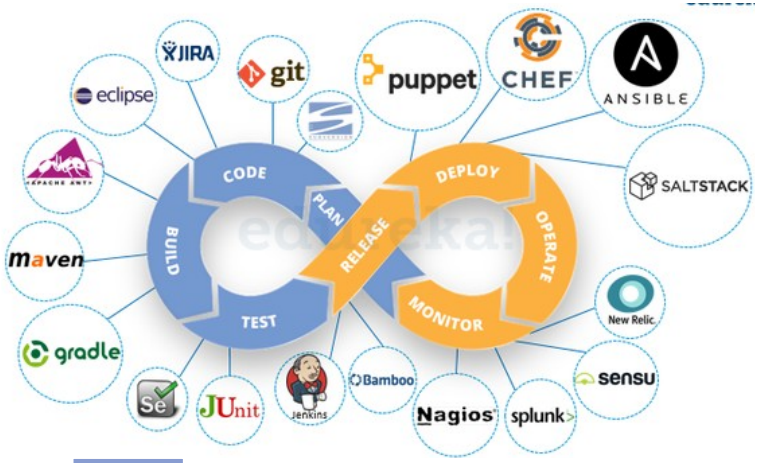
Automation Anxiety

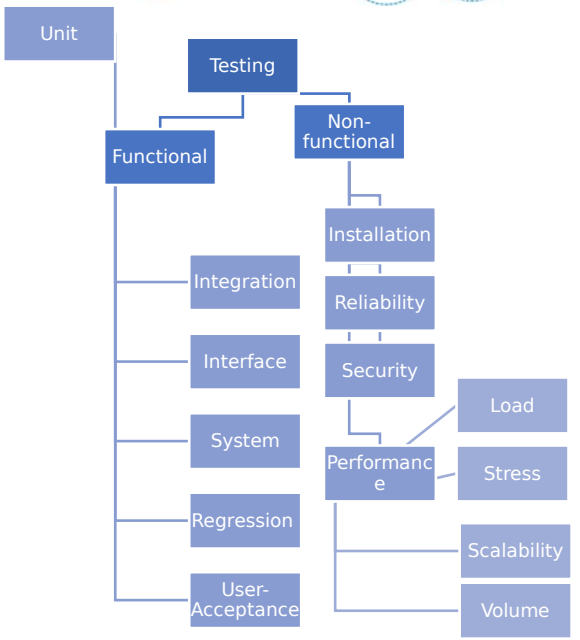
- Automation Correctness.
  - AI/ML isn’t perfect.
  - “Failsafe” mechanisms for unexpected failures.
  - Increased management responsibilities.
- Model complexity.
  - Greater heterogeneity support increase the possibility of modeling error.
  - What is the best value for option Y?
- Telemetry culture and automation visibility
  - Monitor what is available, analyse offline to built intelligence.
  - Are current telemetry approaches appropriate for automated E2E service management?



Assured Automation with Emulation

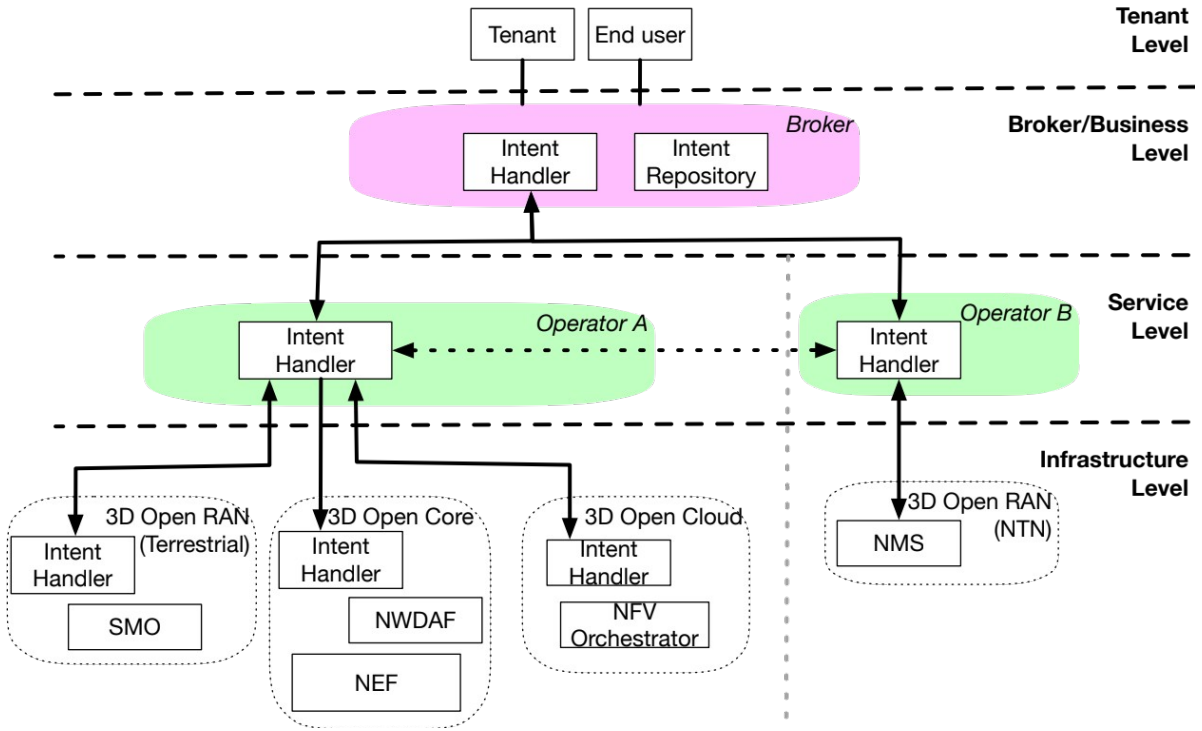
- Cloud systems deploy code changes to production in hours!
- *Testing*: Examine the artifacts and the behaviour of a software, using validation and verification.
- *Monitor*: Collect logging information and performance counter across software layers.
- *Can networks adopt similar testing approaches to validate automation?*
- *Can a NDT accurately recreate the behavior of network devices?*





Management and Orchestration for 3D Open Networks

- New management approaches required for 3D Open Networks.
  - Multi-technology, multi-domain, dynamic infrastructure, adaptation during operation
- Intent-based networking.
  - Decouple network configuration from requirements
  - Easy addition of intelligence
  - Improve automation
- Challenges:
  - Heterogeneity
  - Cover both deployment and operation
  - Built-in correctness validation

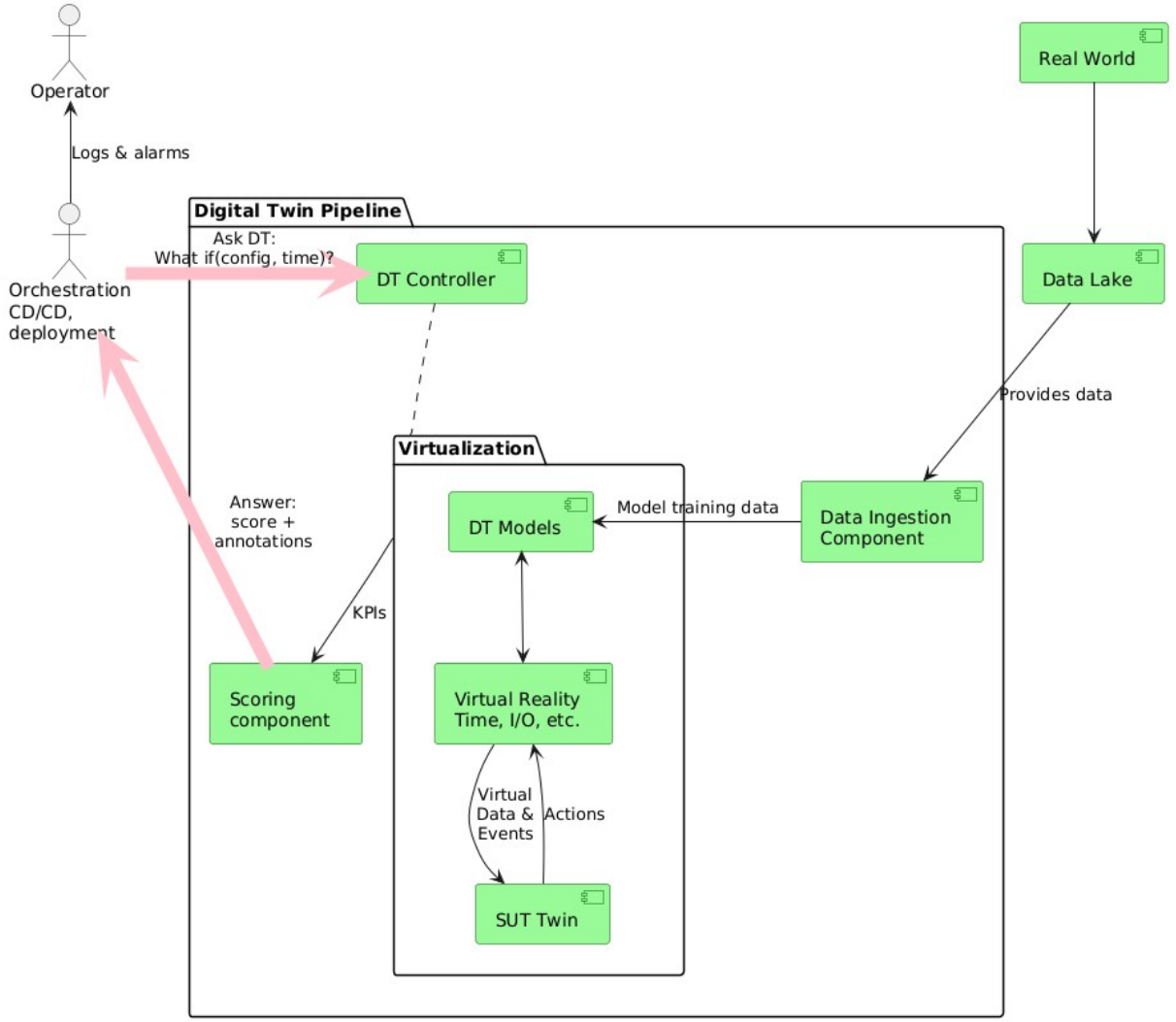


Key Technical Achievements
Closed-control automation for 3D Open Networks

- Life-cycle automation for intent-based management in 3D Open Networks.
  - Use-case: Automated slice provision with bandwidth (BR 2.2a) and priority (BR 2.2b) management.
- Multi-domain monitoring.
  - Multi-domain monitoring architecture.

3D Open Digital Twin

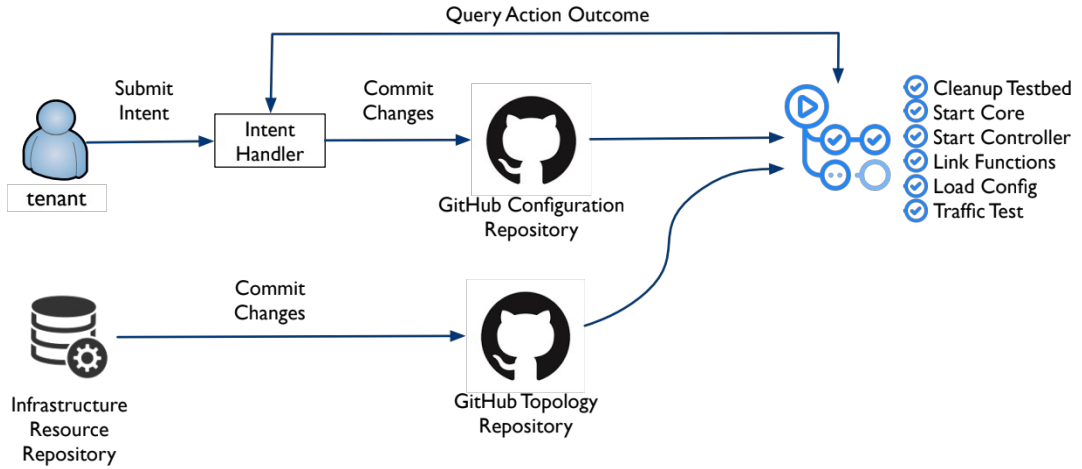
- A digital model of a physical system that replicates realistically behaviors.
- TUDOR implements two DT systems to scope modeling: RAN and Core.
  - Use-cases: Validation, Prediction, ML training
- Challenges:
  - Monitoring & Telemetry
  - Implementation maturity
  - Integration with management mechanisms



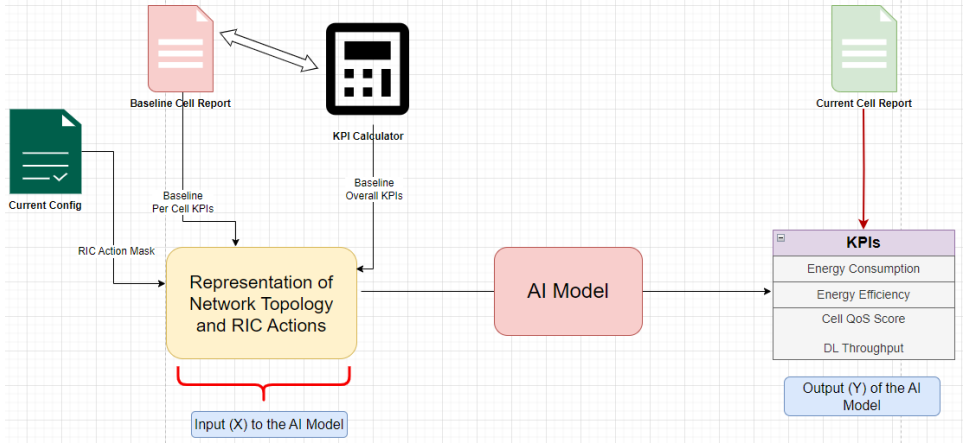
TUDOR Digital Twin system architecture.

Key Technical Achievements
Digital Twin Platform

- 3D Open RAN DT
  - RAN energy optimization
  - Predict power consumption of different xAPPs.
- 3D Open Core DT
  - Validate configuration correctness.
  - Use CI/CD pipeline to automated testing.

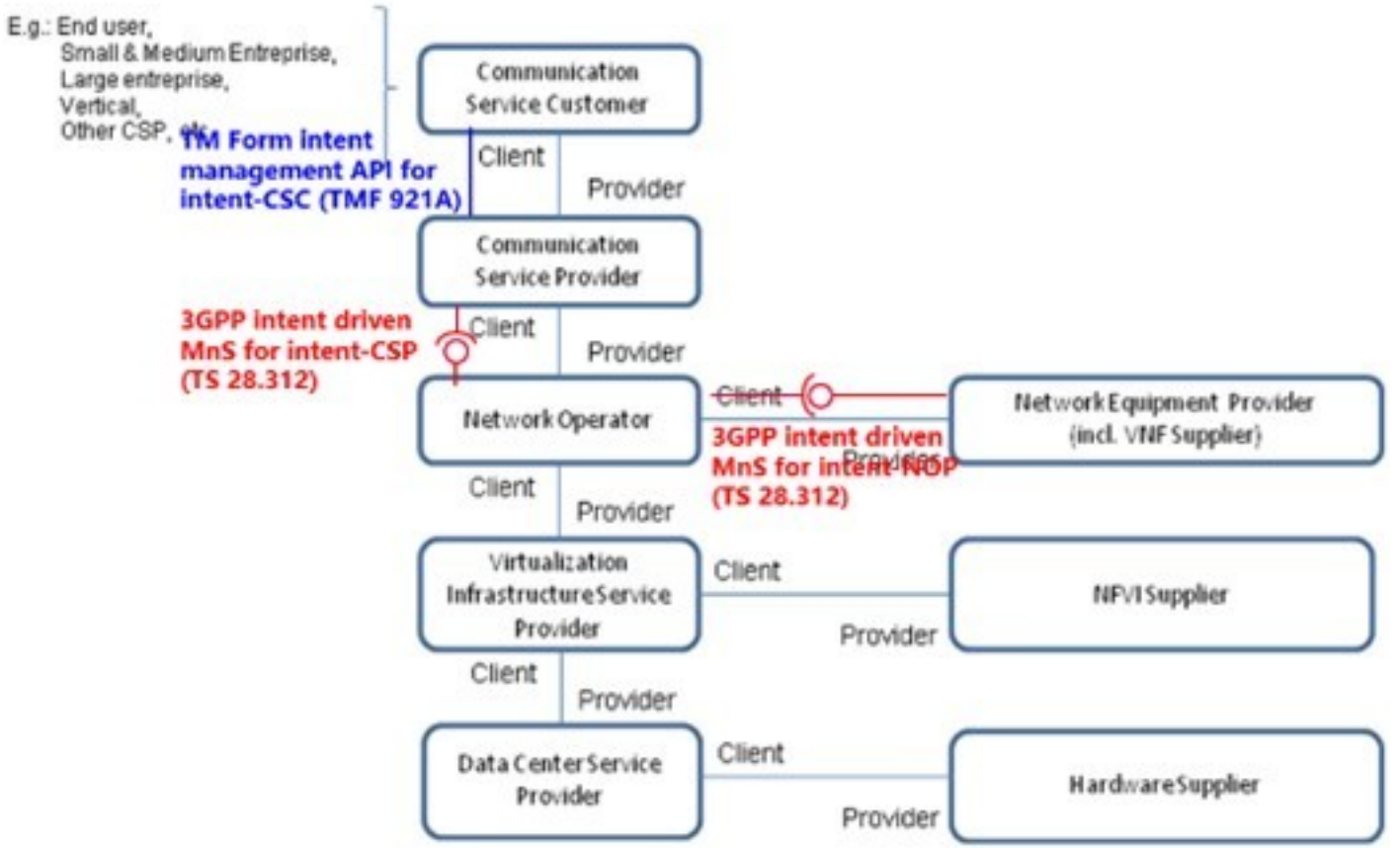


3D Open Core DT: CI/CD configuration validation (BR 2



3D Open RAN DT: CI/CD configuration validation (BR2

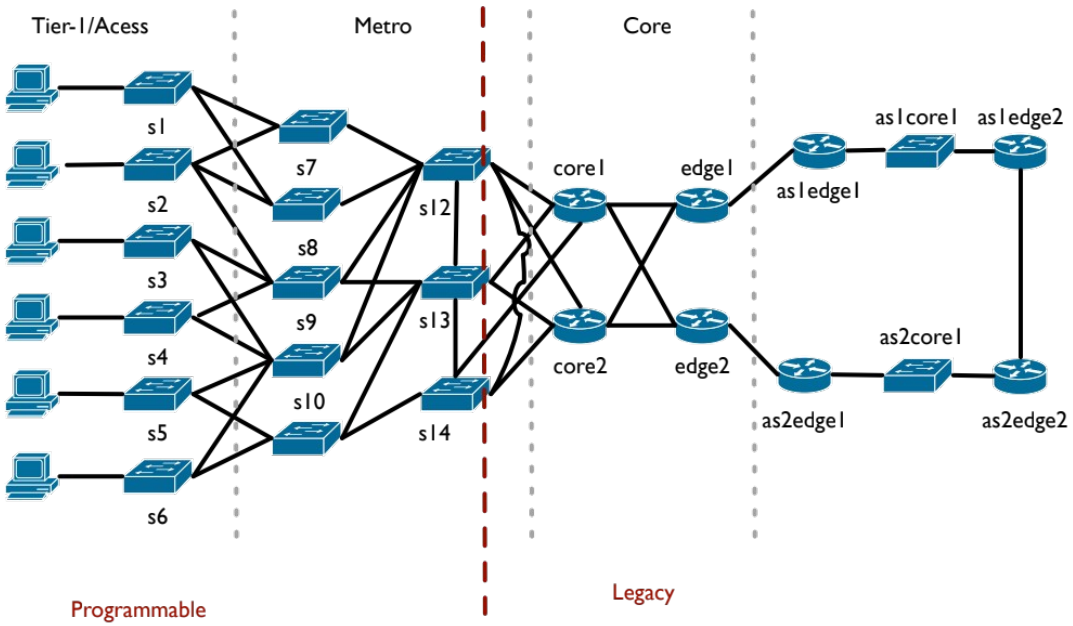
3GPP Compatibility



TUDOR IBN Architecture

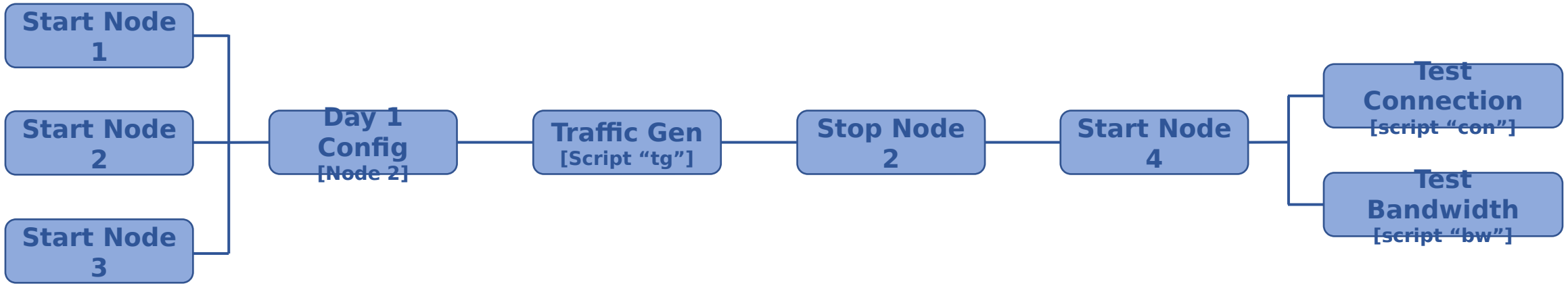
TS 28.312: "Management and orchestration; Intent driven management services for mobile networks".

Use-case: Intent Validation

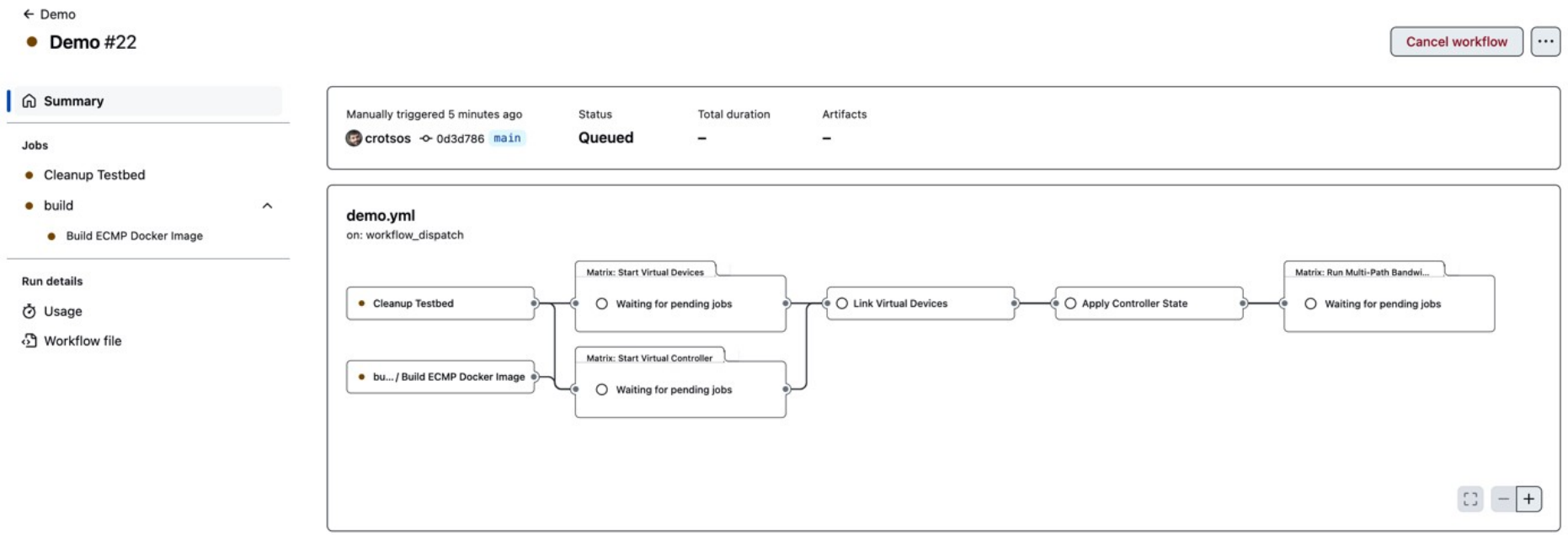


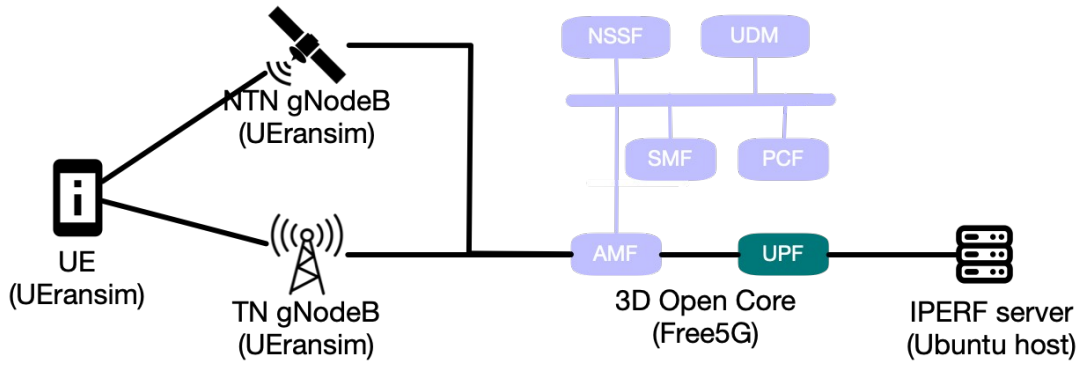
NDT Life-cycle model

- Node, network and script actions get associated event tags
  - e.g. , “node created”
- Test life-cycle architecture: model node interactions using event tags
- Global event ordering in a test scenario
- Easy identification of parallelizable interactions



3D Open Core DT Use-case





Monitoring in action

Next Steps

- Complete intent use-case
  - Run-time slice adaptation
  - Use Core DT to verify configuration
  - Intent-driven uniprobe monitoring services
- Intent Handler Evaluation
  - Scalability, incl. monitoring
  - Responsiveness
  - Generality
- DT Evaluation
  - Scalability
  - Accuracy
  - Energy Efficiency

SDO Opportunity?

- Emulation technologies offer a great opportunity to develop high-precision DT models.
  - Use cases: integration testing, automation plan validation, CI for networks.
- NES is a flexible Emulation-based DT
  - Full test lifecycle support (setup and test)
  - Model-driver test specification and test scripting language
  - Integrates diverse emulation technologies in a single topology
- Standardization opportunity: Can we define a generic model for test automation
  - Allow network community to share best practices via regression tests for configuration.

SDO Opportunity Work

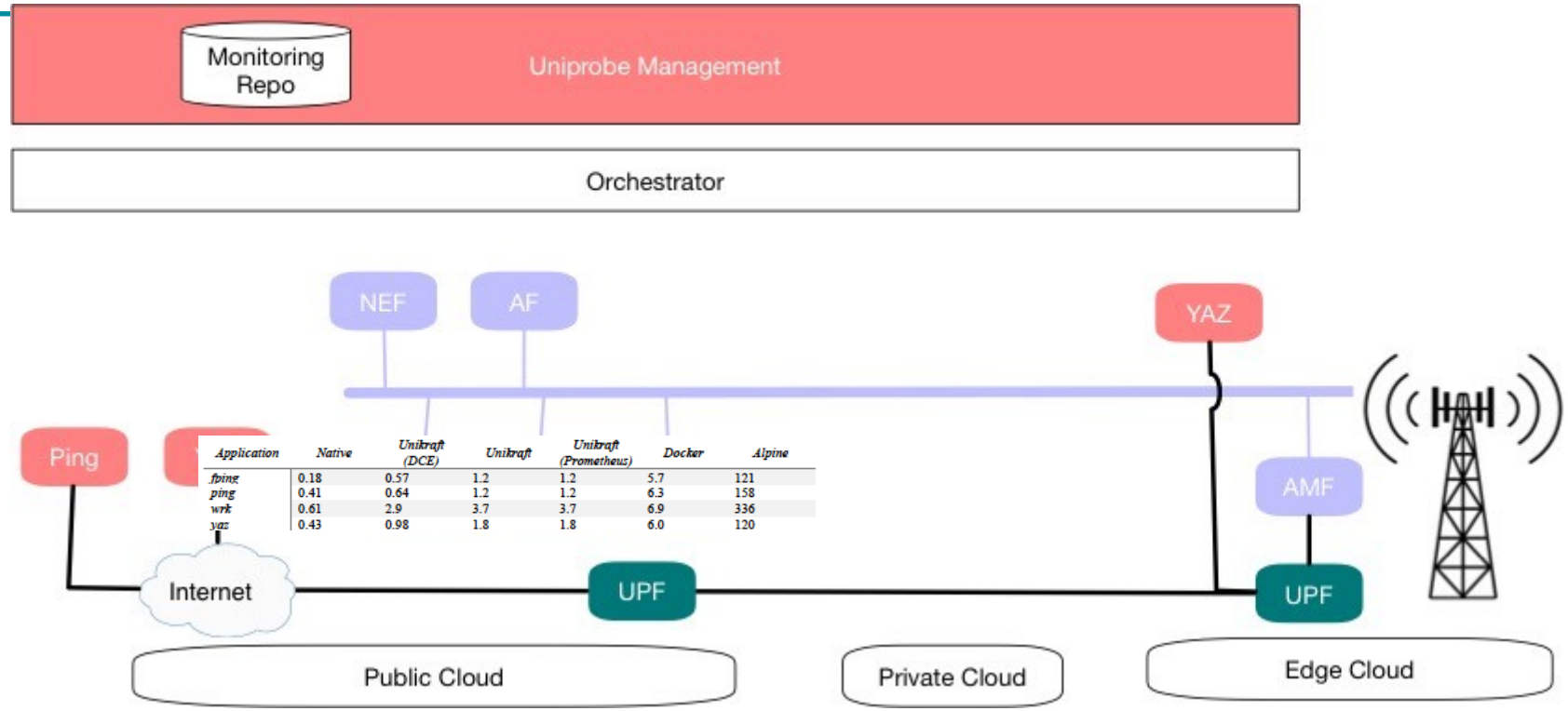
- Define a standard model for test descriptions
  - Capture both topology and test activities
  - Allow easy integration with different processes, e.g., validate human or AI/ML configuration, security policy conformance.
- Lots of topology models
  - Mininet, Docker compose, NFV-MAN, SIMAP?
- Test modeling is more complex
  - Ansible cookbooks, Juju charms, TMForum ODA
- Opportunity: Community network test suites
  - Allow users to exchange test suites for configuration
  - A repeatable test suite is used for consistent validation configuration updates.
  - Exchange best practices between vendors and operators.

Backup Slides

- Path Observation
- Intent Handler
- Intent Reasoner
- TUDOR Ontology
- TUDOR Slice Example

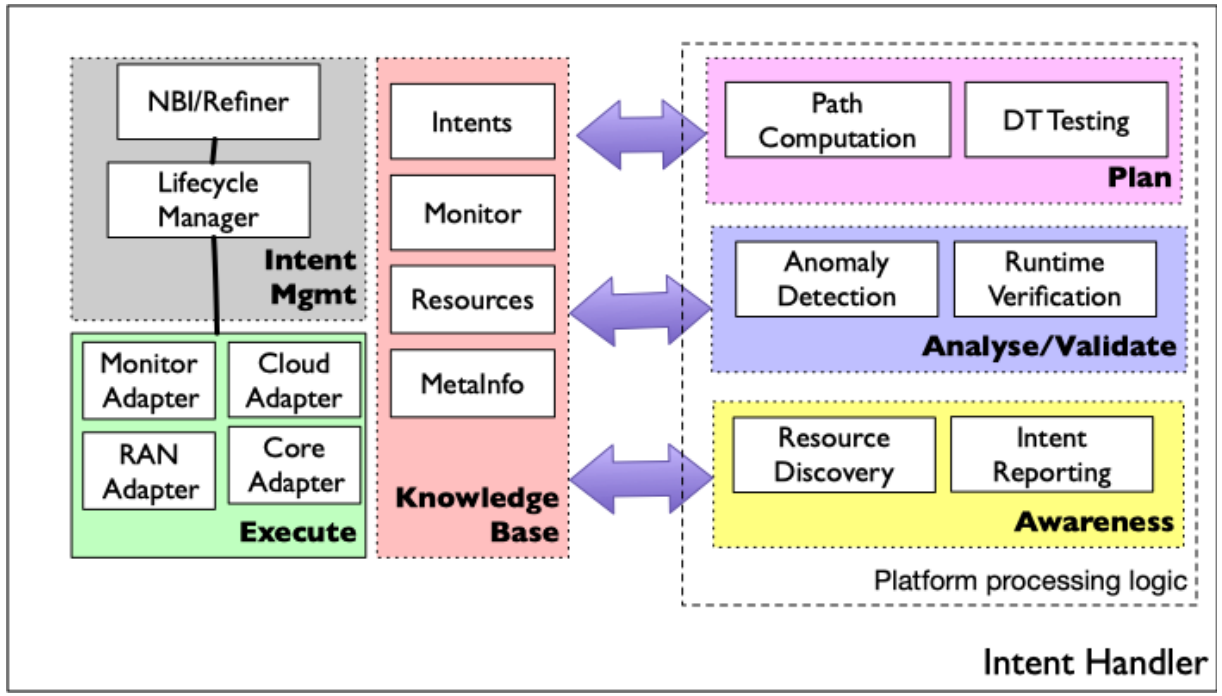
Uniprobe in action

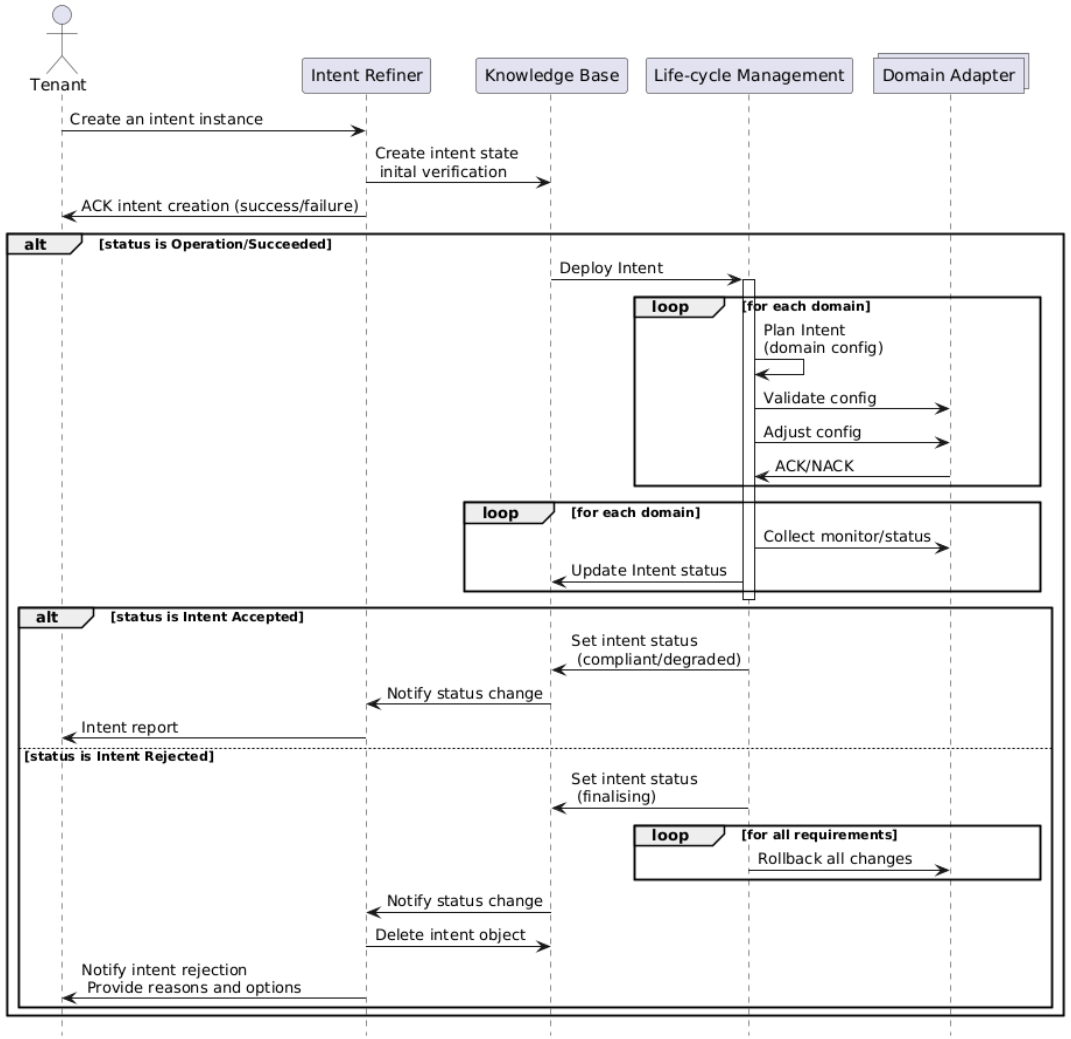
- Operate using OSM and OpenStack.
- Embedded monitoring adaptor for the TUDOR monitoring subsystem.
- Supported apps
  - YAZ, ping, iperf, trace route, wrk
- Can operate with 64 Mb and 1 vCPU.



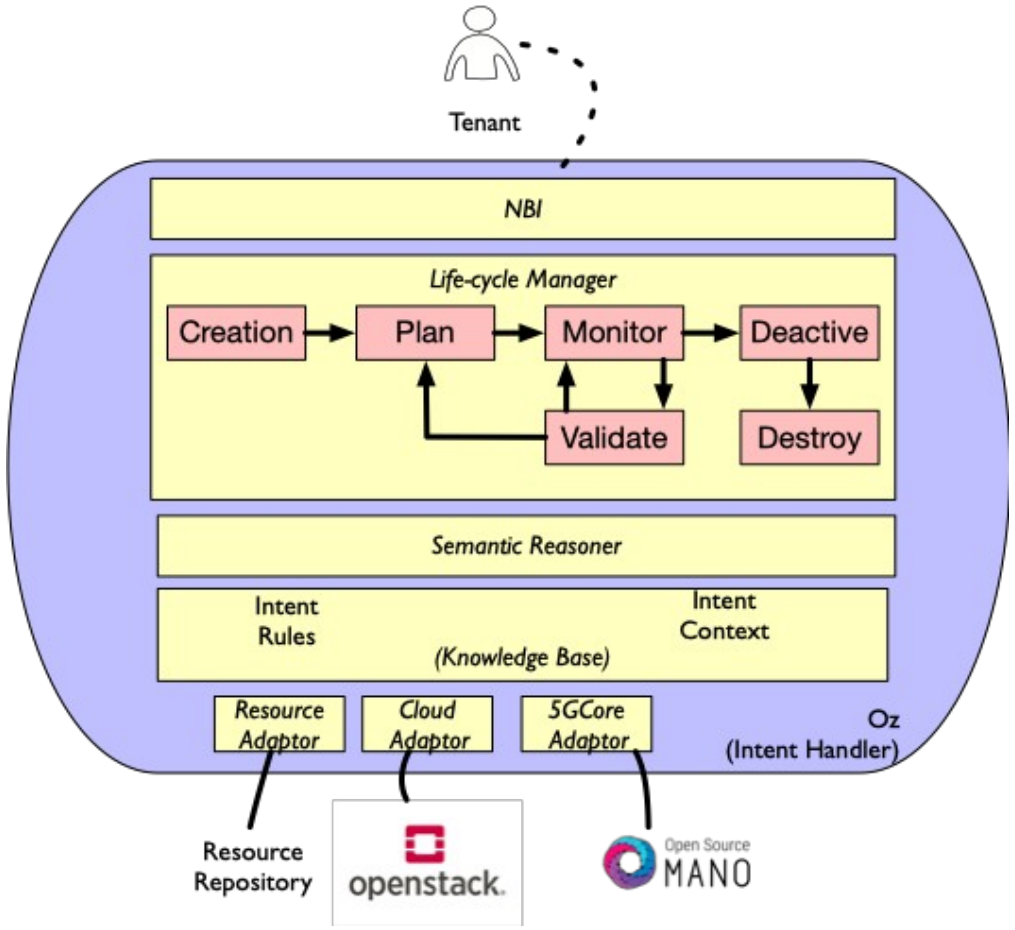
Application	Native	Unikraft (DCE)	Unikraft	Unikraft (Prometheus)	Docker	Alpine
fping	0.18	0.57	1.2	1.2	5.7	121
ping	0.41	0.64	1.2	1.2	6.3	158
wrk	0.61	2.9	3.7	3.7	6.9	336
yaz	0.43	0.98	1.8	1.8	6.0	120

Intent Handler Architecture





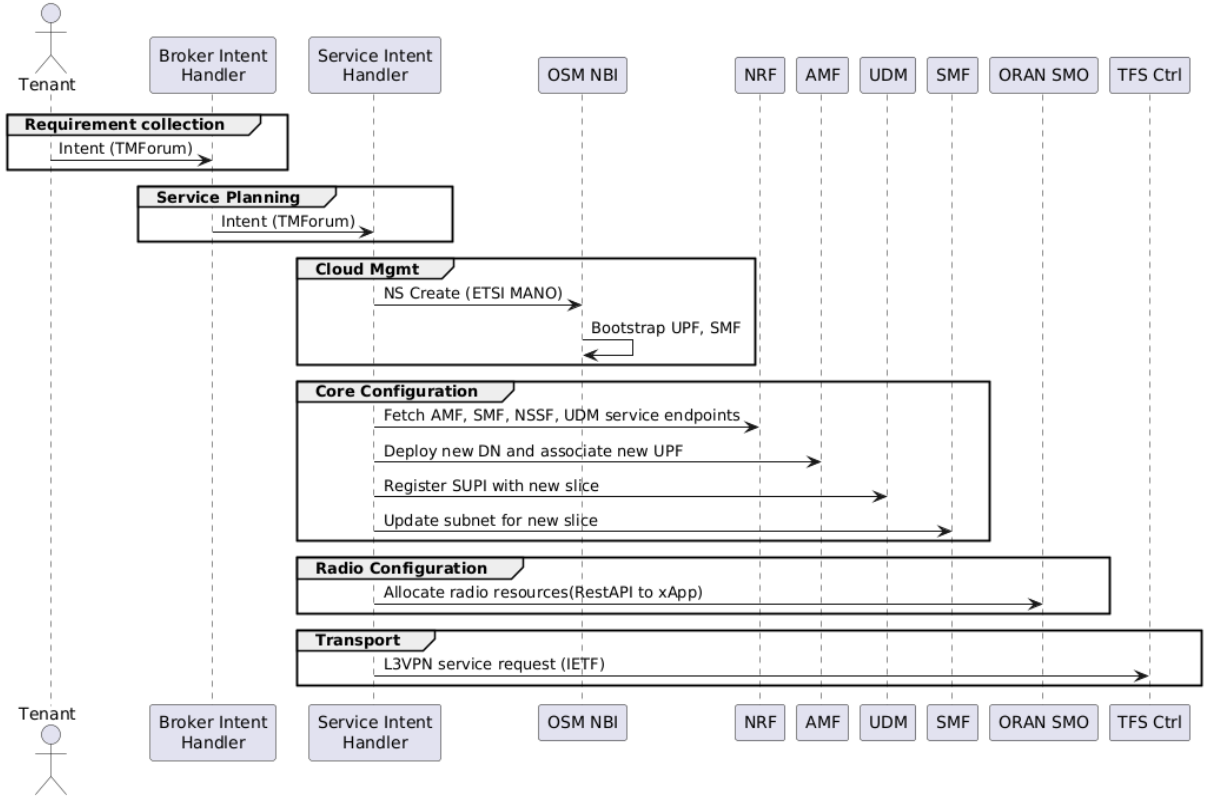
Oz semantic reasoner

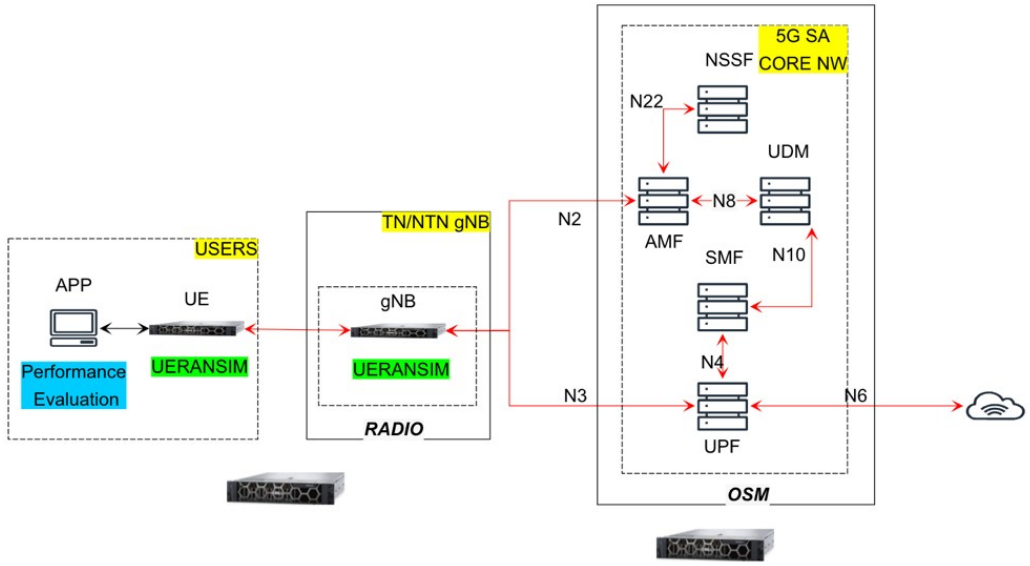



```
requiresAvailability(I) :-
  rdfs_individual_of(I, icm:'PropertyParameter'),
  rdf_has(I, kpi:availability, _),
  add_constraint(validHost, [Host, DU],
  (
    constraint(checkAvailability, false);
    \+ (
          isRunningOn(ODU, Host),
          hasDeploymentUnit(F, DU),
          hasDeploymentUnit(OF, ODU),
          hasFunction(FC, F),
          hasFunction(OFC, OF),
          FC \= OFC
        ))).
```

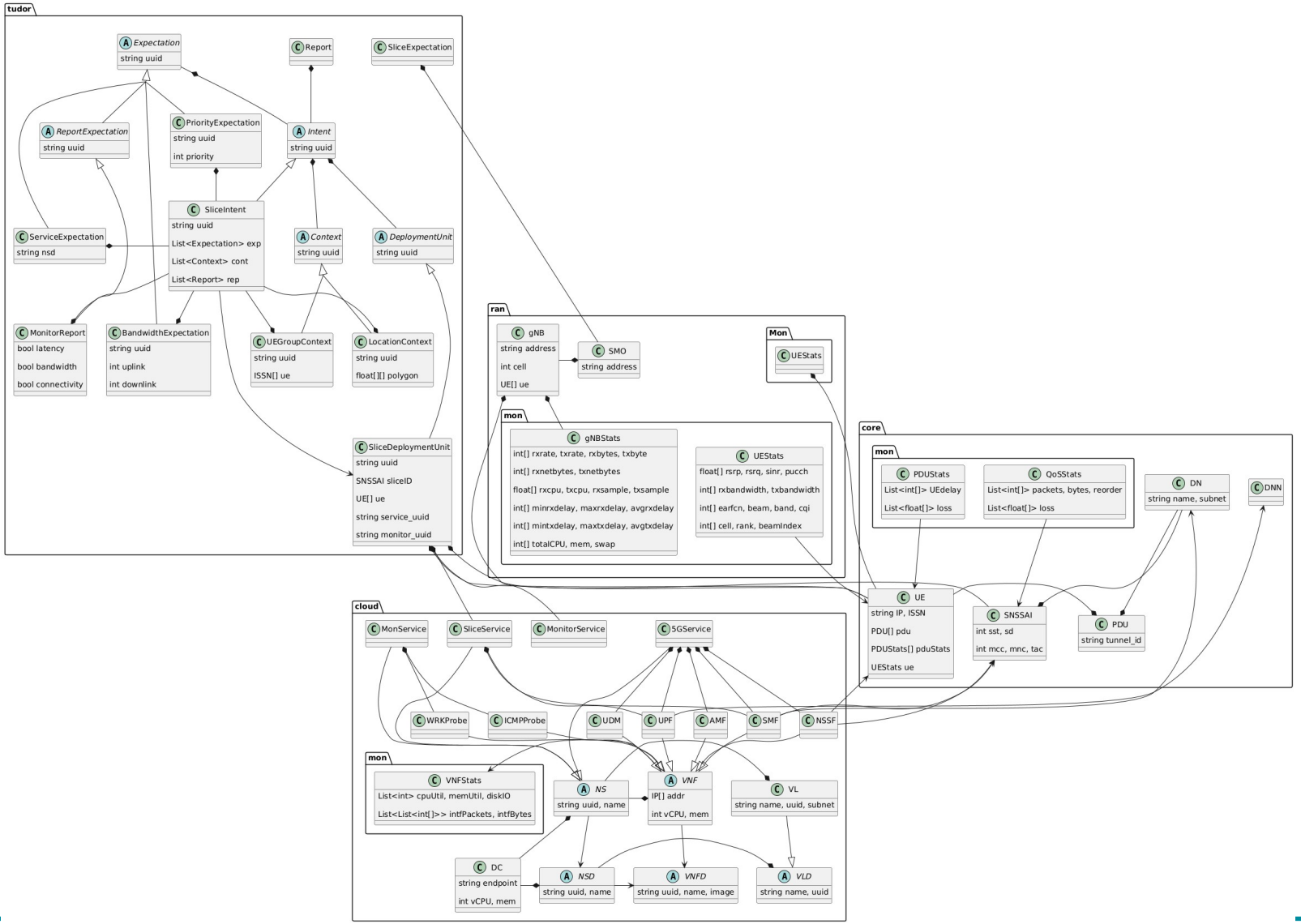
```
capableDC(Host, DepU) :-
        DC(Host),
        hasVCPUCapacity(Host, DepU),
        hasMemoryCapacity(Host, DepU),
        hasStorageCapacity(Host, DepU).
        hasMemoryResource(Host, 100GB),
        connected(Host, Switch),
        switch(Switch).
```

Use-case: Slice deployment





TUDOR Ontology



3GPP Intent model