SWITCH

EU H2020-ICT-2014

Information modeling and semantic linking for a software workbench for interactive, time critical and self-adaptive cloud applications

Dr. Zhiming Zhao

Paul Martin, Arie Taal, Francisco Quevedo, David Rogers, Kieran Evans, Andrew Jones, Vlado Stankovski, Salman Taherizadeh, Jernej Trnkoczy, George Suciu, **Zhiming Zhao**

System and network engineering

University of Amsterdam

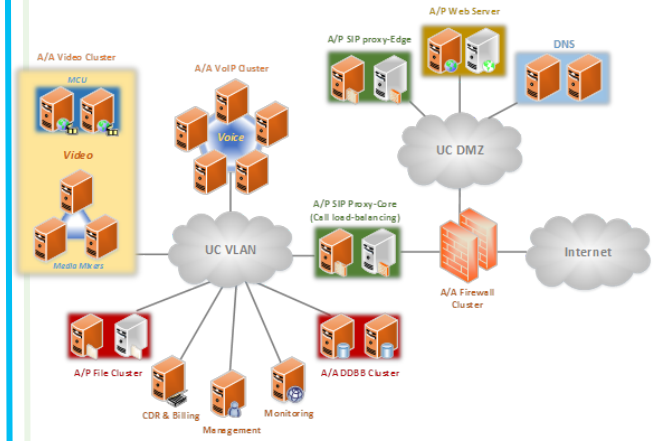
Universiteit van Amsterdam · Wellness Telecom · Cardiff University · Univerza v Ljubljani · Beia Consult International · mog

Outline

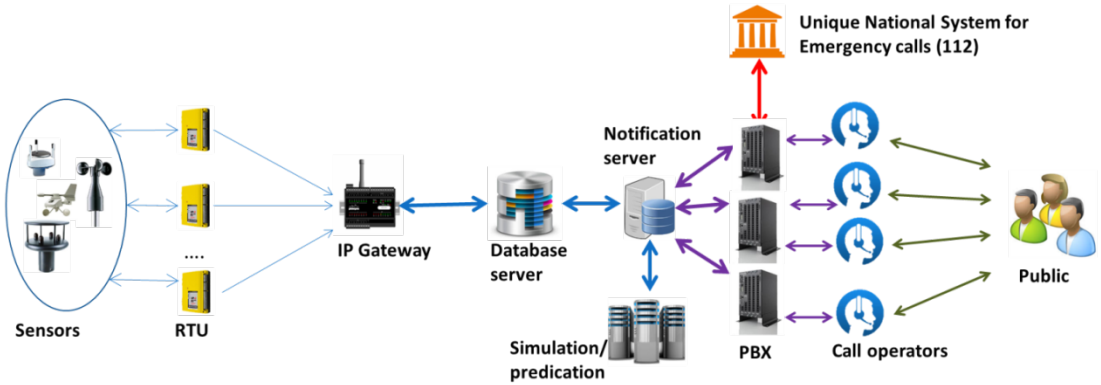
- Problem background
- The SWITCH project
- The semantic linking approach
- Summary

Time critical cloud applications

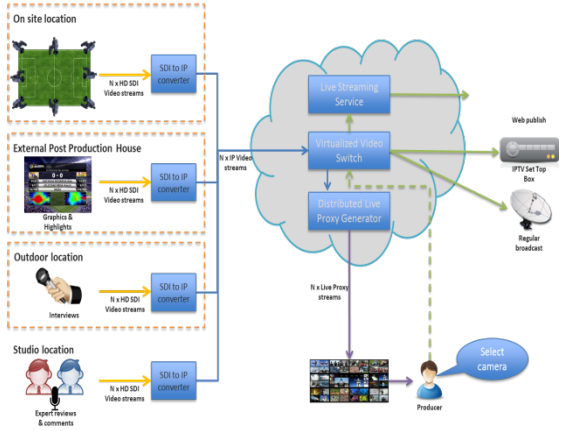
- Have **very high** business potential or social impacts, e.g.,
  - Real-time business collaboration,
  - disaster early warning, and
  - Live event broadcasting;
- Have **very critical** quality requirements for services, e.g.,
  - video quality, system interaction, or data delivery;
- But are **very expensive in** implementation and operation.



1. Real time business communication platform (Wellness telecom, Spain)

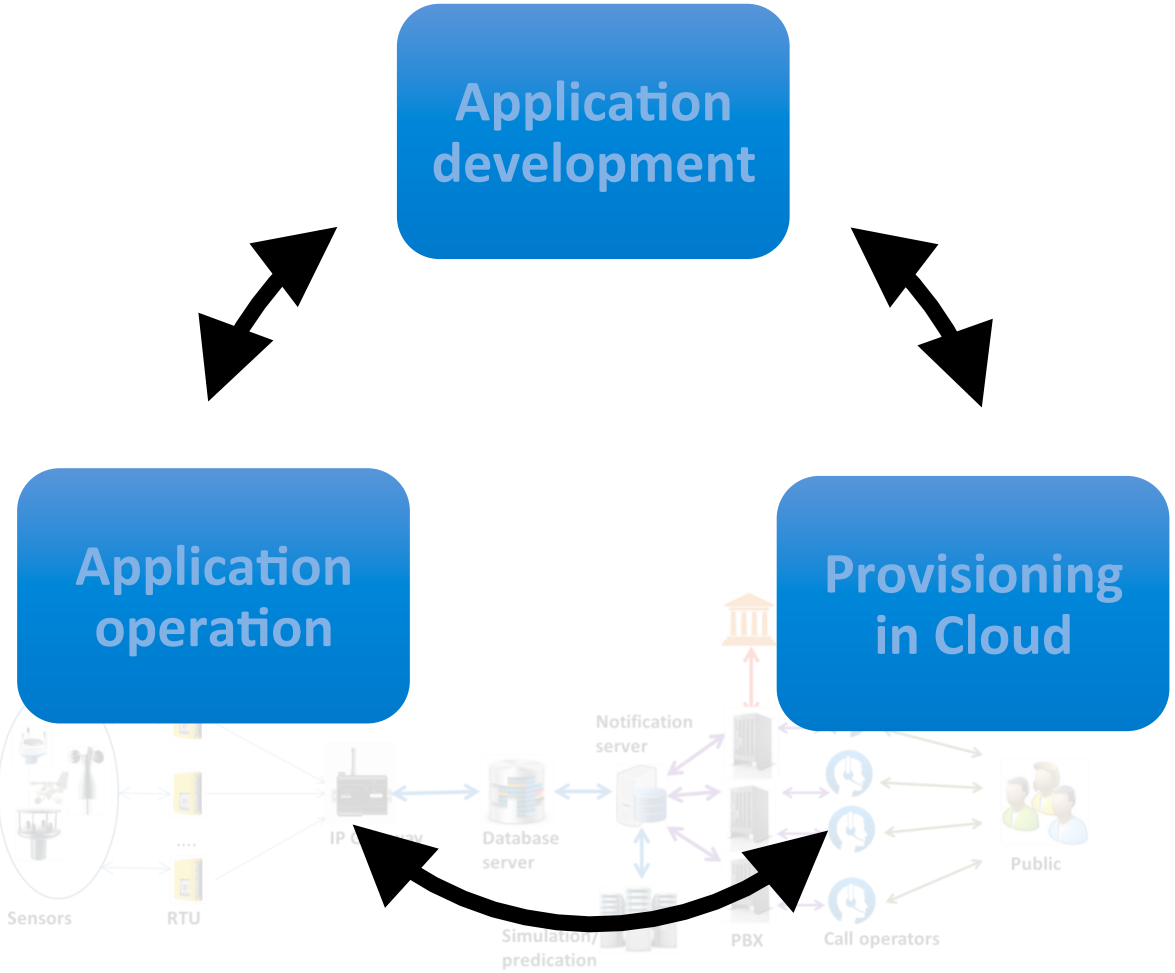


2. The Elastic Disaster Early warning system (BEIA, Romania)



3. Cloud studio for directing and broadcasting live events (MOG, Portugal)

Difficulties and challenges



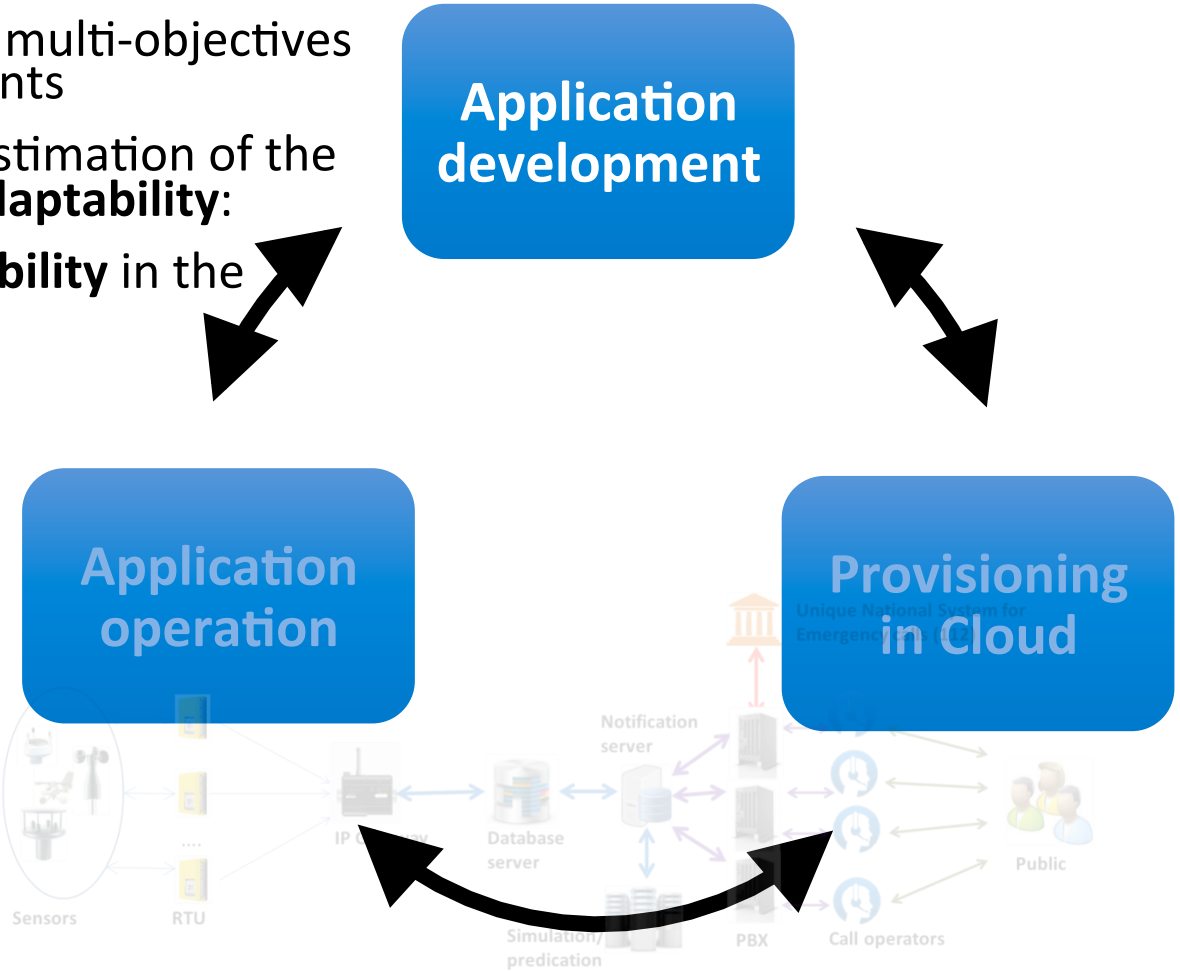
1. Real time business communication platform (Wellness telecom, Spain)

2. The Elastic Disaster Earlier warning system (BEIA, Romania)

3. Cloud studio for directing and broadcasting live events (MOG, Portugal)

Application development challenges

- **System integration** to meet multi-objectives of the performance constraints
- Compose application with estimation of the **system performance and adaptability**:
- Including Cloud **programmability** in the application:



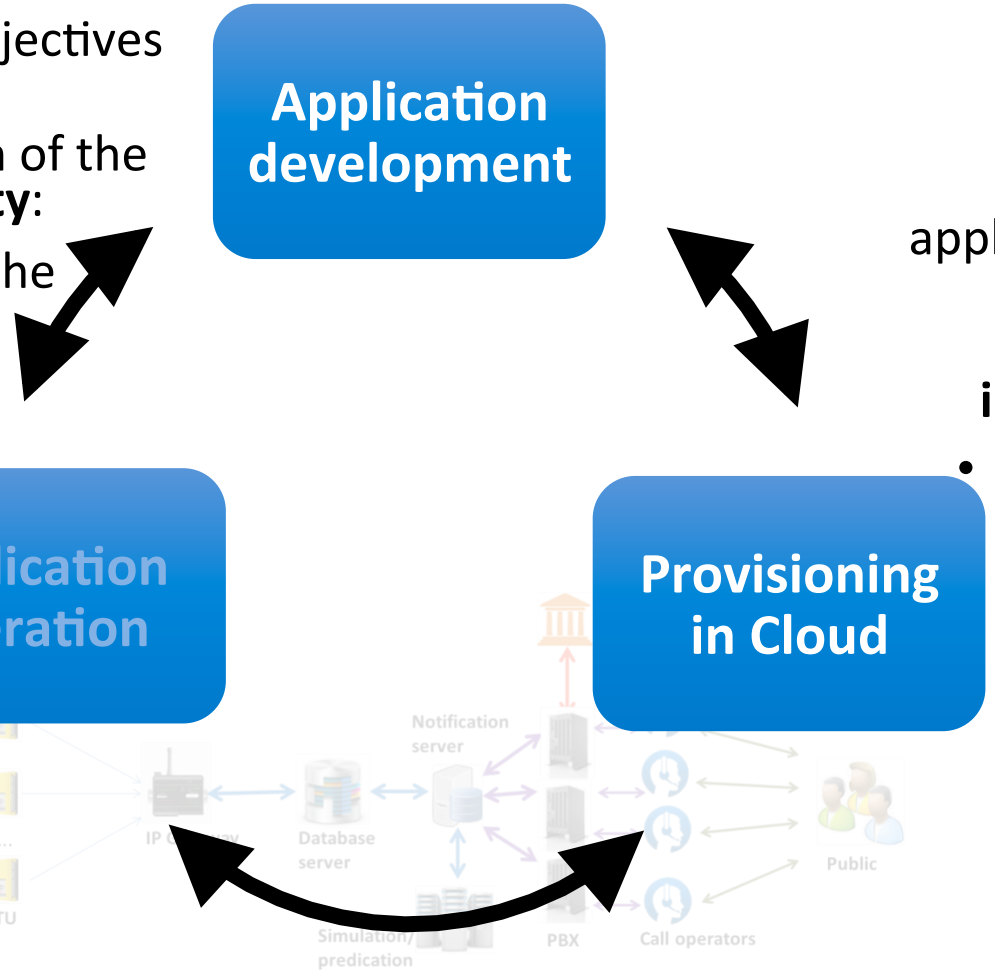
1. Real time business communication platform (Wellness telecom, Spain)

2. The Elastic Disaster Earlier warning system (BEIA, Romania)

3. Cloud studio for directing and broadcasting live events (MOG, Portugal)

Provision/Deployment challenges

- **System integration** to meet multi-objectives of the performance constraints
- Compose application with estimation of the **system performance and adaptability**:
- Including Cloud **programmability** in the application:



- **Customize virtual infrastructure** for an application to meet constraints for different objectives,
- **Provision runtime infrastructure** on Cloud and
- **Deploy the applications** to cloud

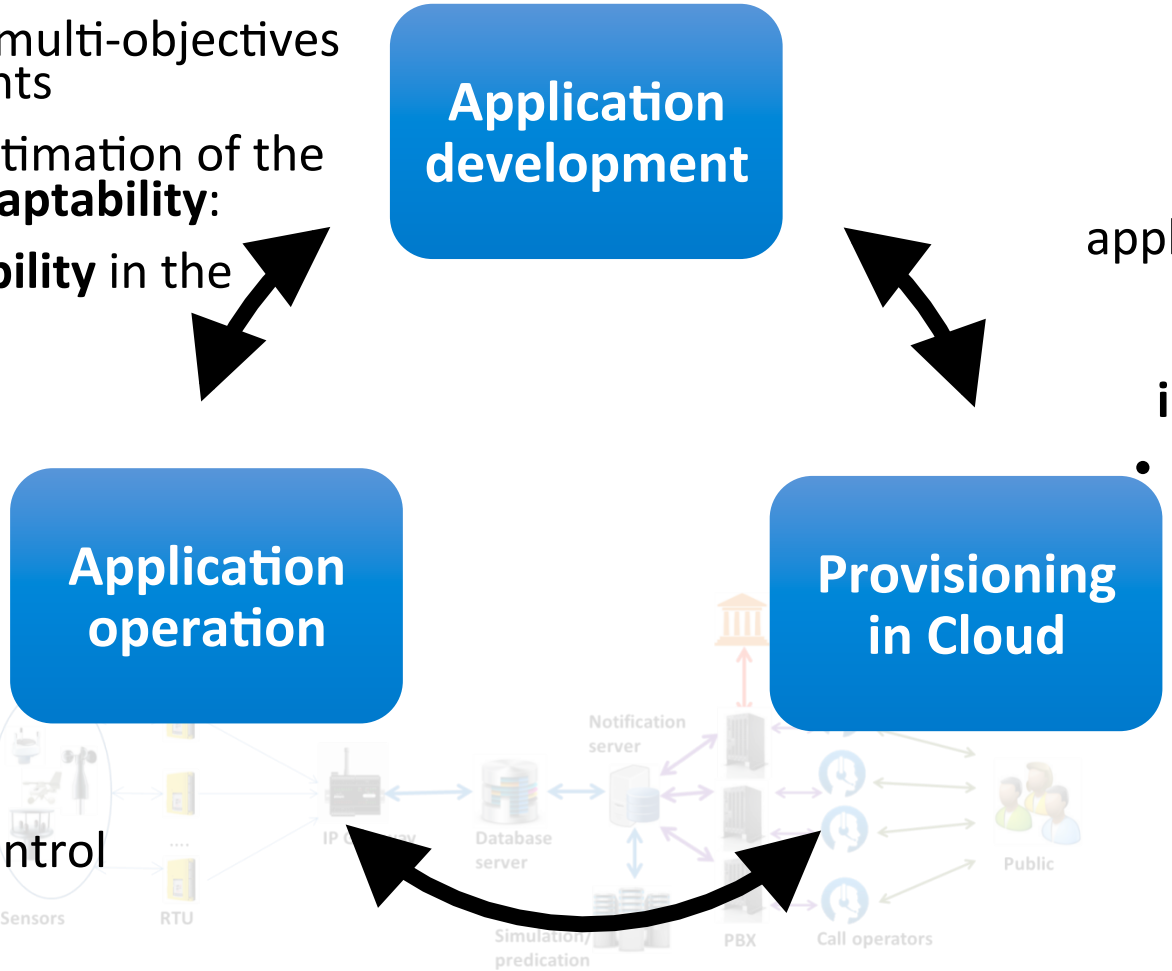
1. Real time business communication platform (Wellness telecom, Spain)

2. The Elastic Disaster Earlier warning system (BEIA, Romania)

3. Cloud studio for directing and broadcasting live events (MOG, Portugal)

Operation challenges

- **System integration** to meet multi-objectives of the performance constraints
- Compose application with estimation of the **system performance and adaptability**:
- Including Cloud **programmability** in the application:



- **Customize virtual infrastructure** for an application to meet constraints for different objectives,
- **Provision runtime infrastructure** on Cloud and
- **Deploy the applications** to cloud

- **Monitor and control** system behavior to meet QoS/QoE constraints
- **Human in the loop** system control

1. Real time business communication platform (Wellness telecom, Spain)

2. The Elastic Disaster Earlier warning system (BEIA, Romania)

3. Cloud studio for directing and broadcasting live events (MOG, Portugal)

Objectives

- **Addressing the entire lifecycle** of time critical cloud applications by new software workbench (middleware and front end tools)
  - **Improve the development and execution model** of time critical applications in programmable, elastic and virtualized environment.
  - **Improve the productivity of application development** by introducing SWITCH Interactive Development Environment
  - **Improve the deployment and provisioning efficiency** by introducing Distributed Runtime Infrastructure Planner
  - **Improve the QoS/QoE control efficiency** by Autonomous Self-Adaptation Platform
  - **Promote and exploit the SWITCH methods and software** by actively combining technology development and the pilot use case providers.

About the SWITCH project

- Between Feb/ 2015 - 2018
- Total budget 2.92 M
- EU H2020 ICT-9
- 6 Partners
- Web: www.switchproject.eu

* Universiteit van Amsterdam

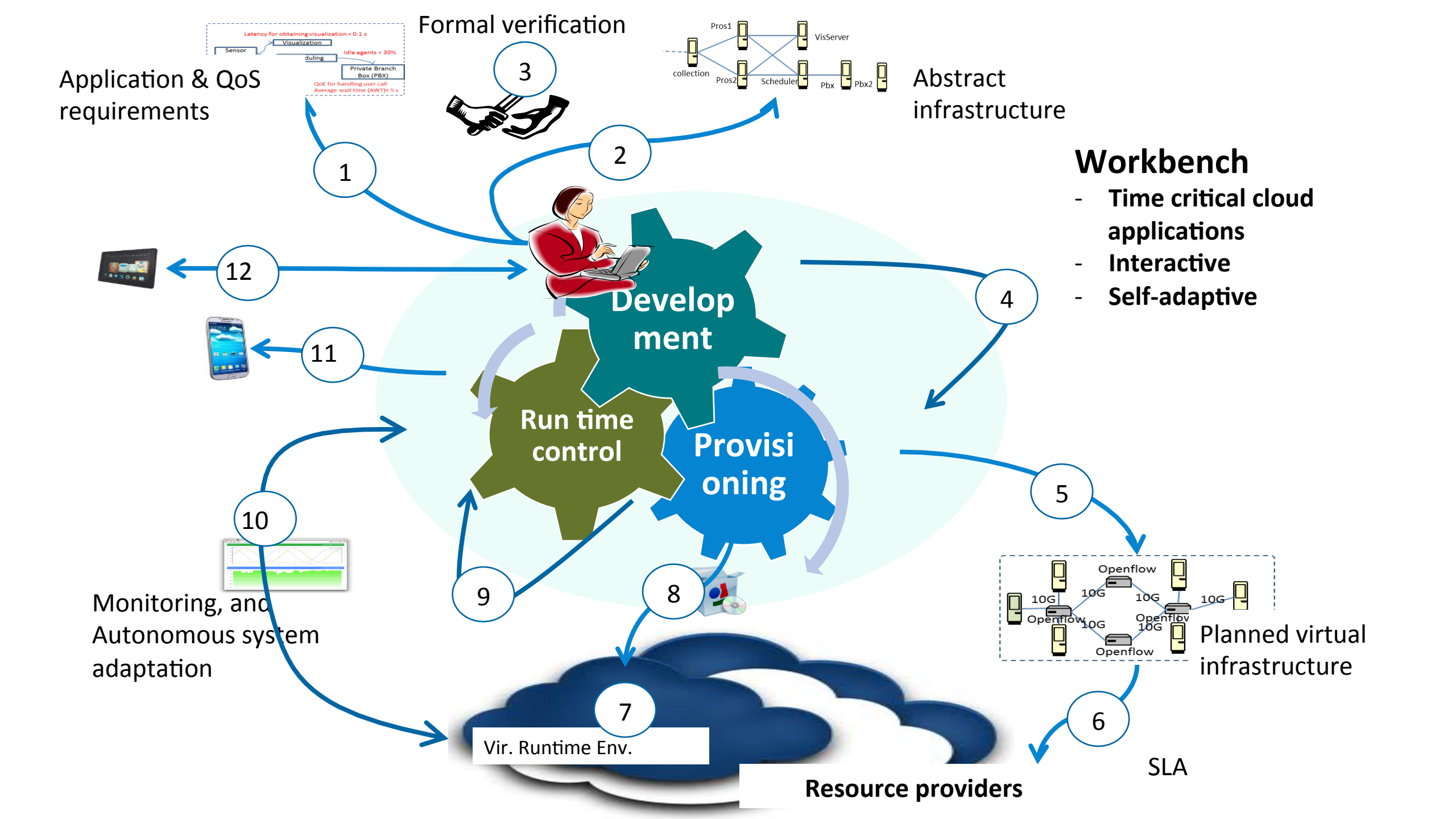
Wellness Telecom

Cardiff University

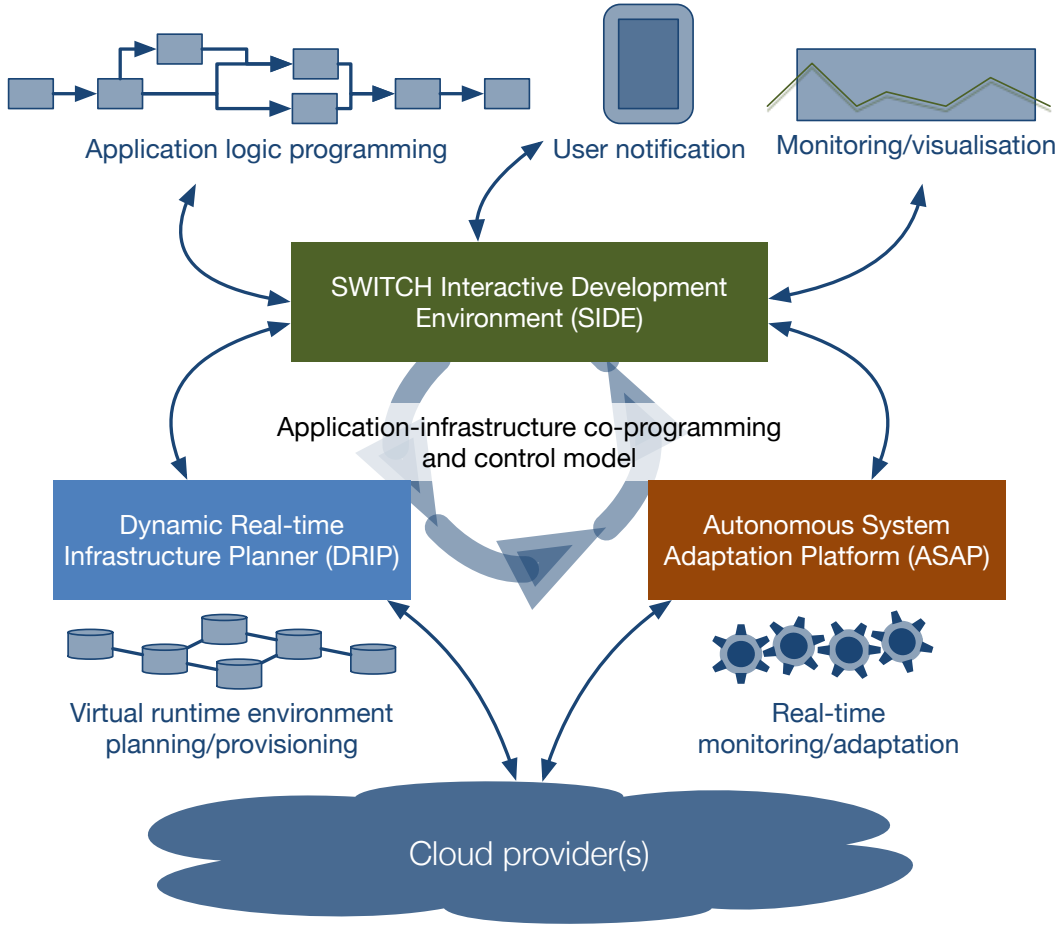
Beia Consult International

Univerza v Ljubljani

mog exceeding media challenges


Application & QoS requirements
Formal verification
Abstract infrastructure
Workbench
- Time critical cloud applications
- Interactive
- Self-adaptive
Develop ment
Run time control
Provisi oning
Monitoring, and Autonomous system adaptation
Planned virtual infrastructure
SLA
Vir. Runtime Env.
Resource providers
Pros1
Pros2
VisServer
collection
Scheduler
Pbx
Pbx2
Openflow
10G
1
2
3
4
5
6
7
8
9
10
11
12

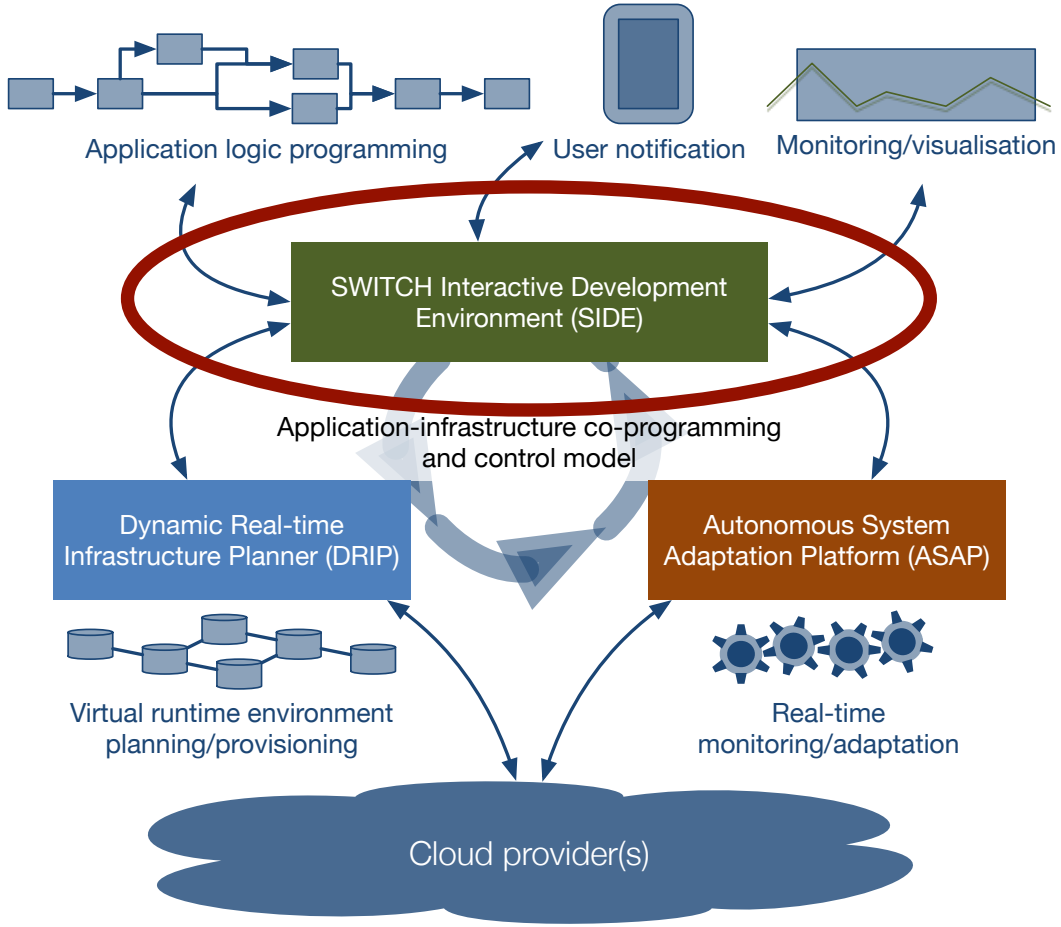
The SWITCH approach



The three subsystems are coordinated according to an **application-infrastructure co-programming and control model**:

- Each stage of the application lifecycle is supported by intelligent decision support provided by the SWITCH **Knowledge Base**.
- The developer is kept in the loop at every point in the lifecycle, to intercede or approve, or simply to observe.

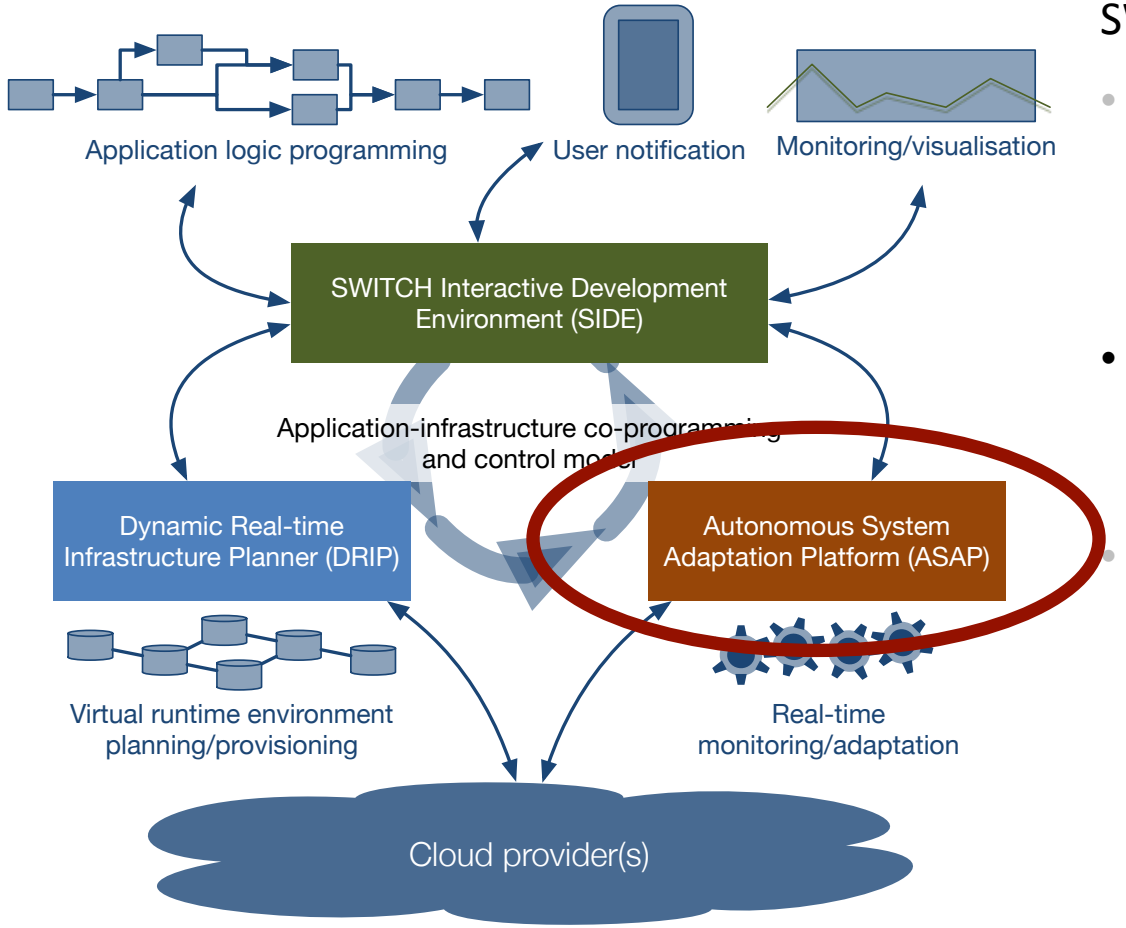
The SWITCH approach



SWITCH has three core subsystems:

- The **SWITCH interactive development environment (SIDE)** provides the user interface for composing applications and visualising the outputs of the other subsystems.
- The **Dynamic Real-time Infrastructure Planner (DRIP)** constructs infrastructures compatible with applications using available Cloud resources.
- The **Autonomous System Adaptation Platform (ASAP)** monitors applications at runtime and adapts application or infrastructure in response to threats.

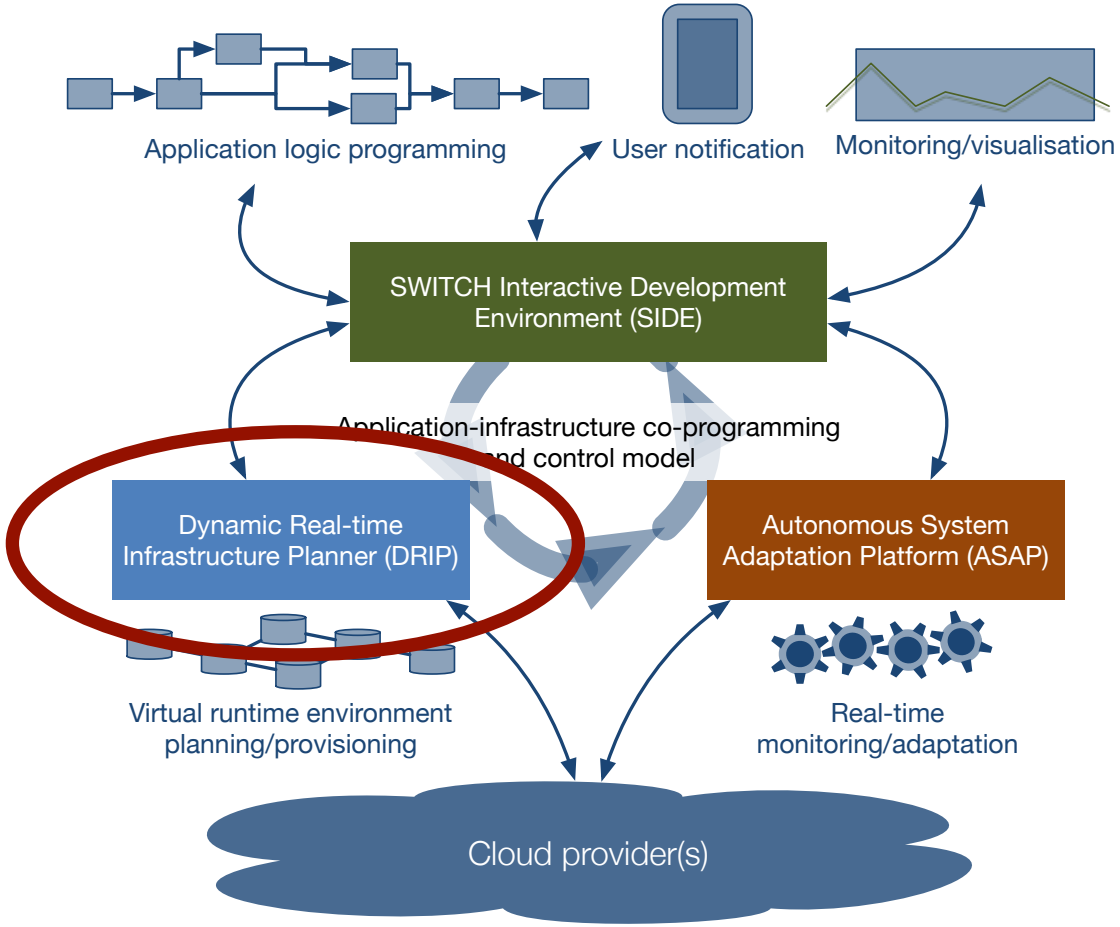
The SWITCH approach



SWITCH has three core subsystems:

- The **SWITCH interactive development environment (SIDE)** provides the user interface for composing applications and visualising the outputs of the other subsystems.
- The **Dynamic Real-time Infrastructure Planner (DRIP)** constructs infrastructures compatible with applications using available Cloud resources.
- The **Autonomous System Adaptation Platform (ASAP)** monitors applications at runtime and adapts application or infrastructure in response to threats.

The SWITCH approach



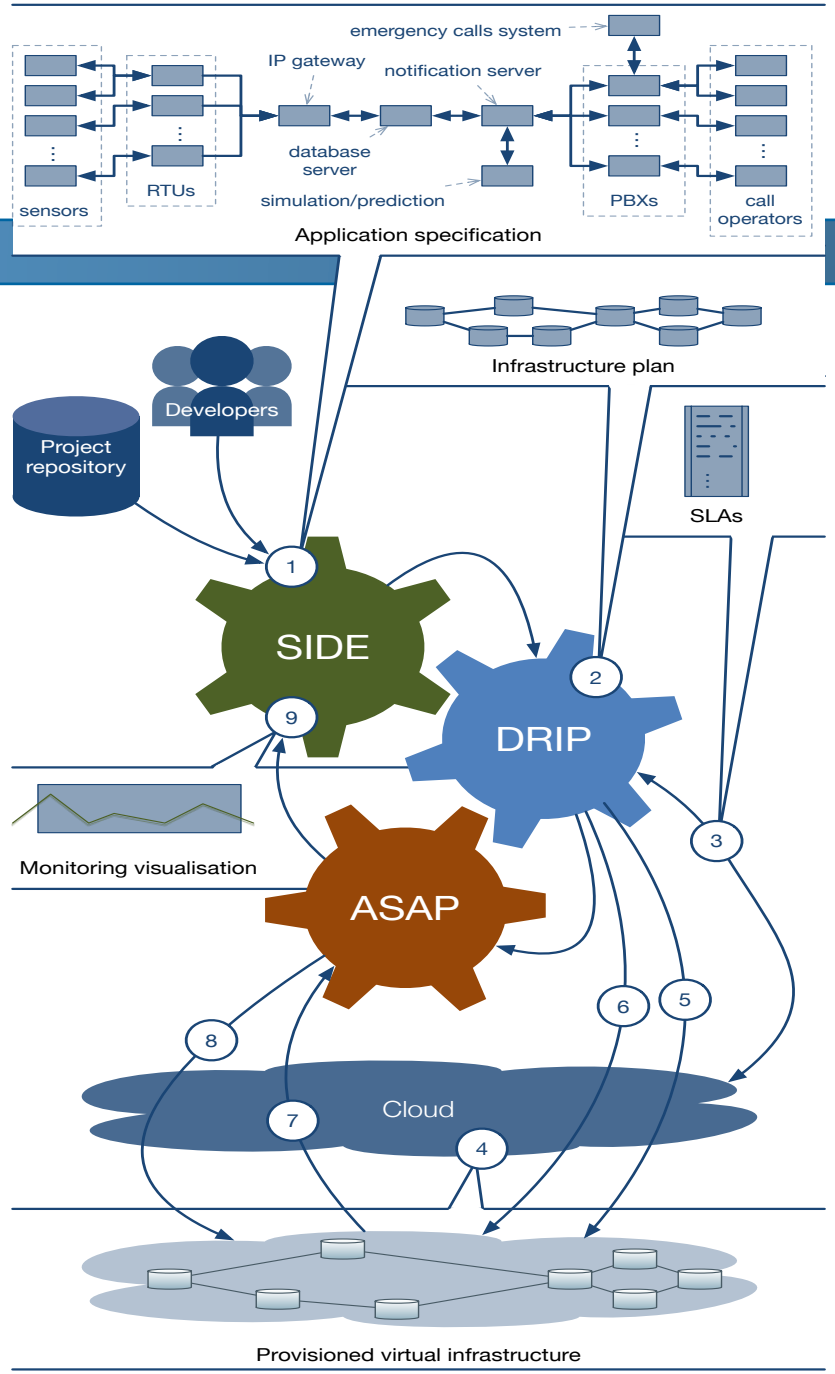
SWITCH has three core subsystems:

- The **SWITCH interactive development environment (SIDE)** provides the user interface for composing applications and visualising the outputs of the other subsystems.
- The **Dynamic Real-time Infrastructure Planner (DRIP)** constructs infrastructures compatible with applications using available Cloud resources.
- The **Autonomous System Adaptation Platform (ASAP)** monitors applications at runtime and adapts application or infrastructure in response to threats.

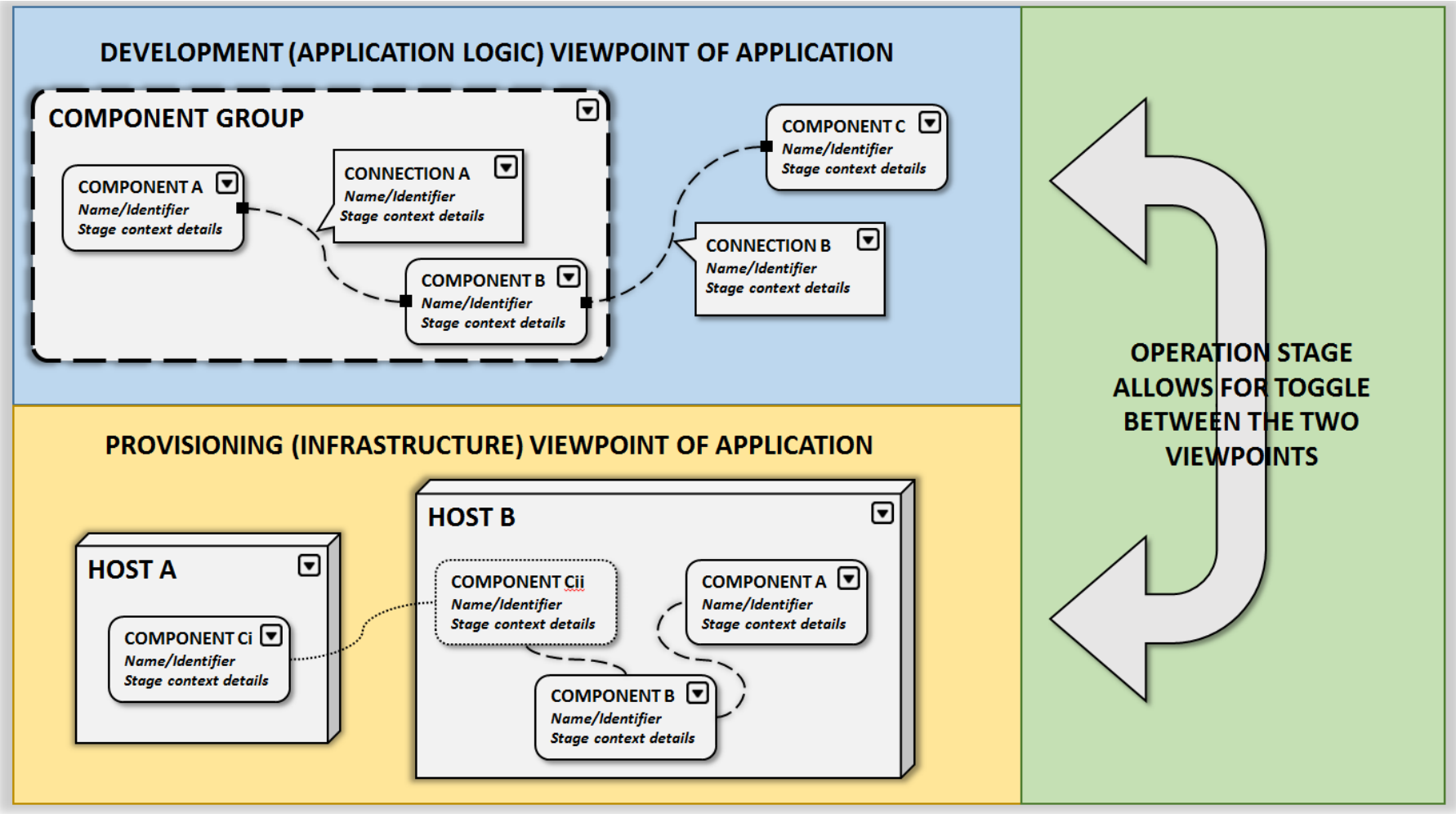
Semantic linking tasks in SWITCH

The SWITCH application lifecycle is split into a number of interlinked phases:

1. **Application composition and verification.**
2. Resource selection and infrastructure planning.
3. SLA negotiation.
4. Infrastructure provisioning.
5. Application deployment.
6. Application execution and runtime management.
7. Runtime monitoring and diagnosis.
8. Runtime adaptation.
9. Runtime visualisation and feedback.



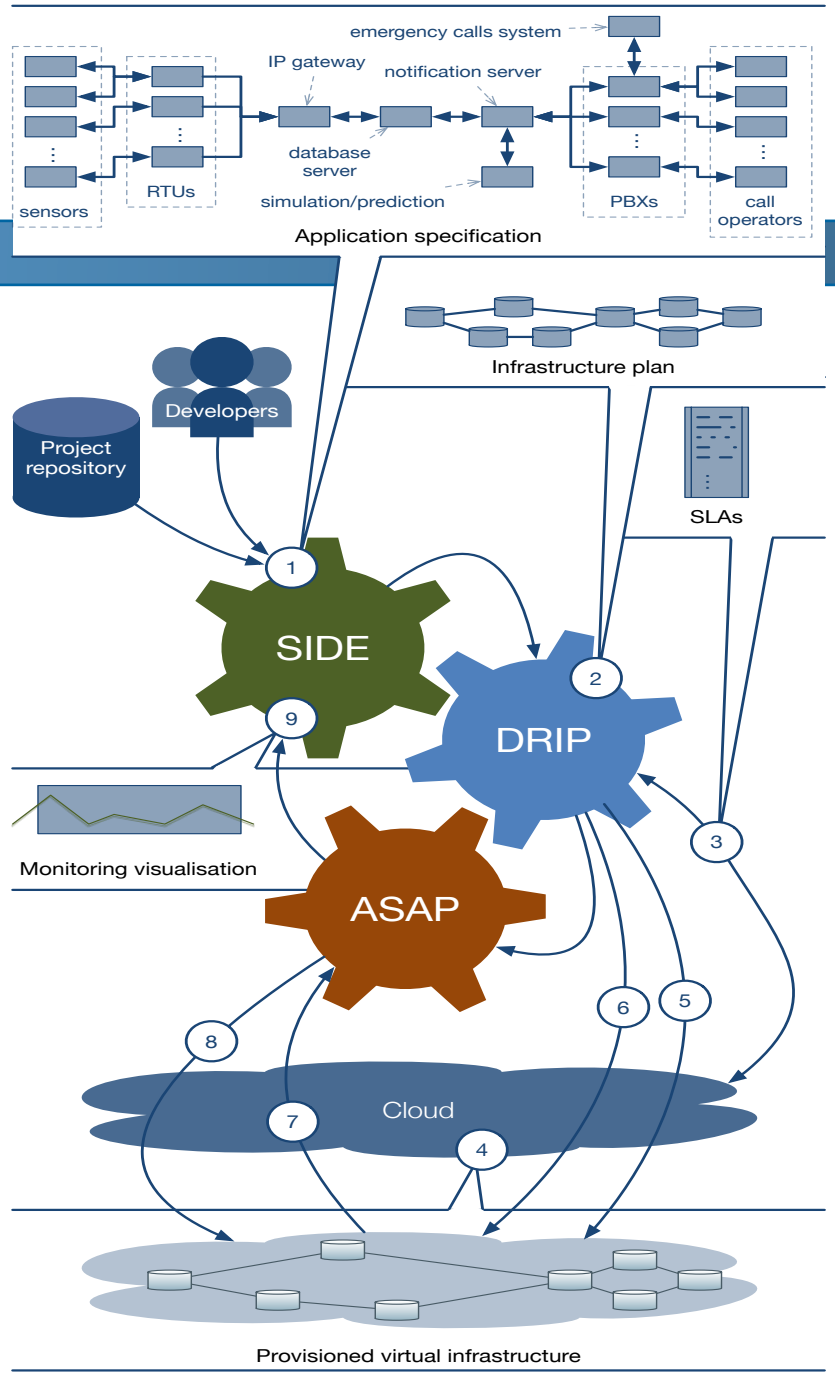
GUI presentation based on different development phases



Resource selection and infrastructure planning

The SWITCH application lifecycle is split into a number of interlinked phases:

1. Application composition and verification.
2. **Resource selection and infrastructure planning.**
3. SLA negotiation.
4. Infrastructure provisioning.
5. Application deployment.
6. Application execution and runtime management.
7. Runtime monitoring and diagnosis.
8. Runtime adaptation.
9. Runtime visualisation and feedback.

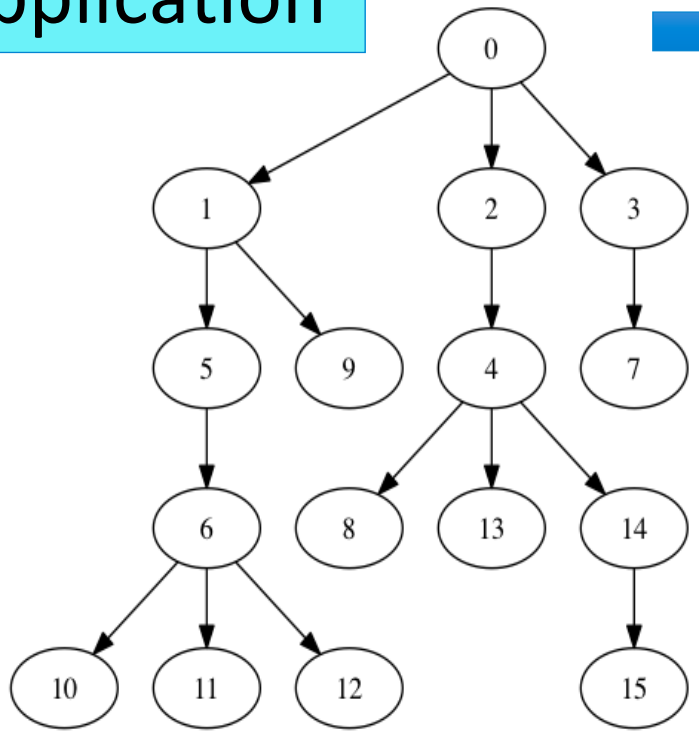


Infrastructure planning

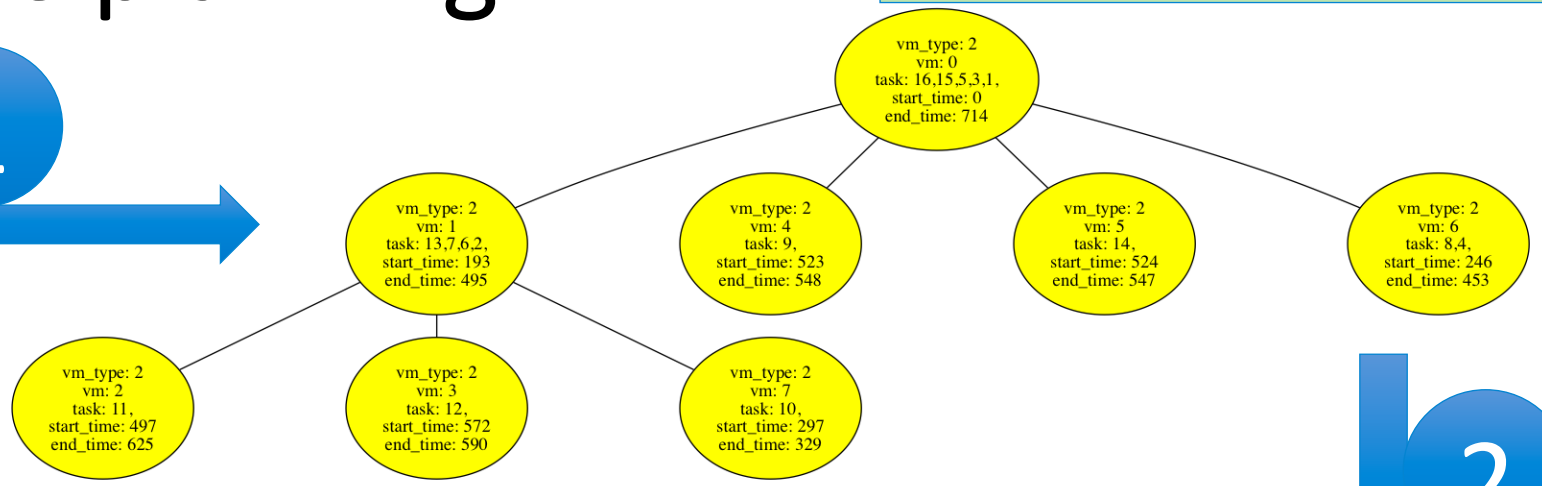
SWITCH

Application

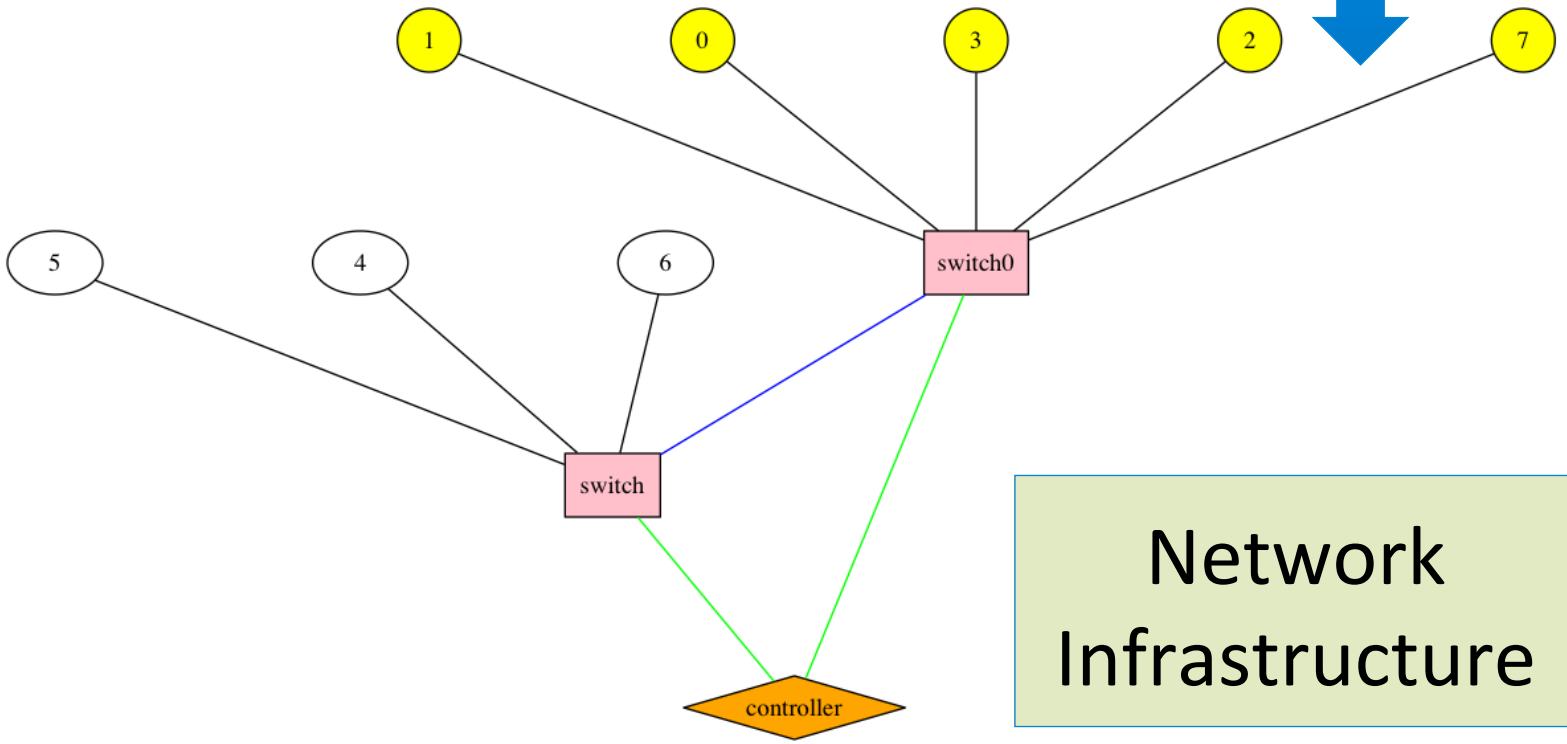
1



VM Infrastructure



2



Network Infrastructure

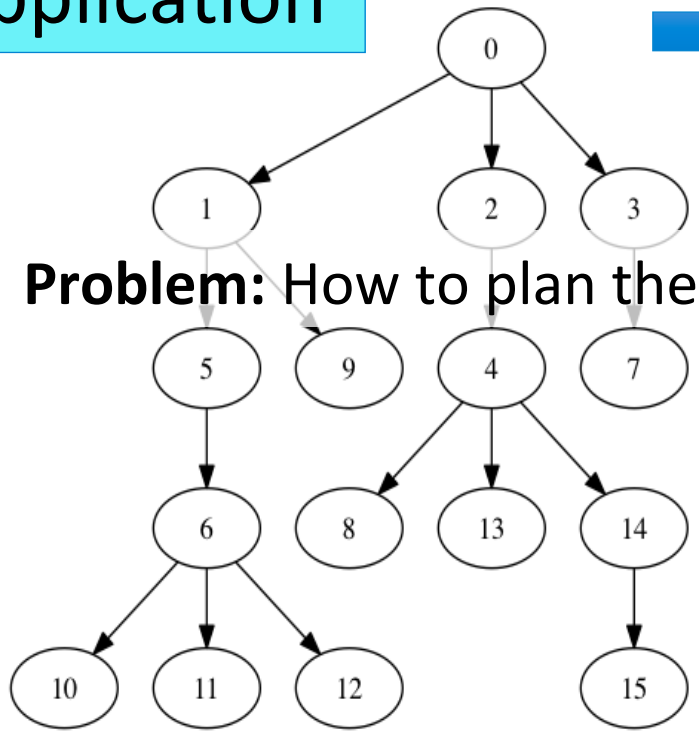
Infrastructure planning

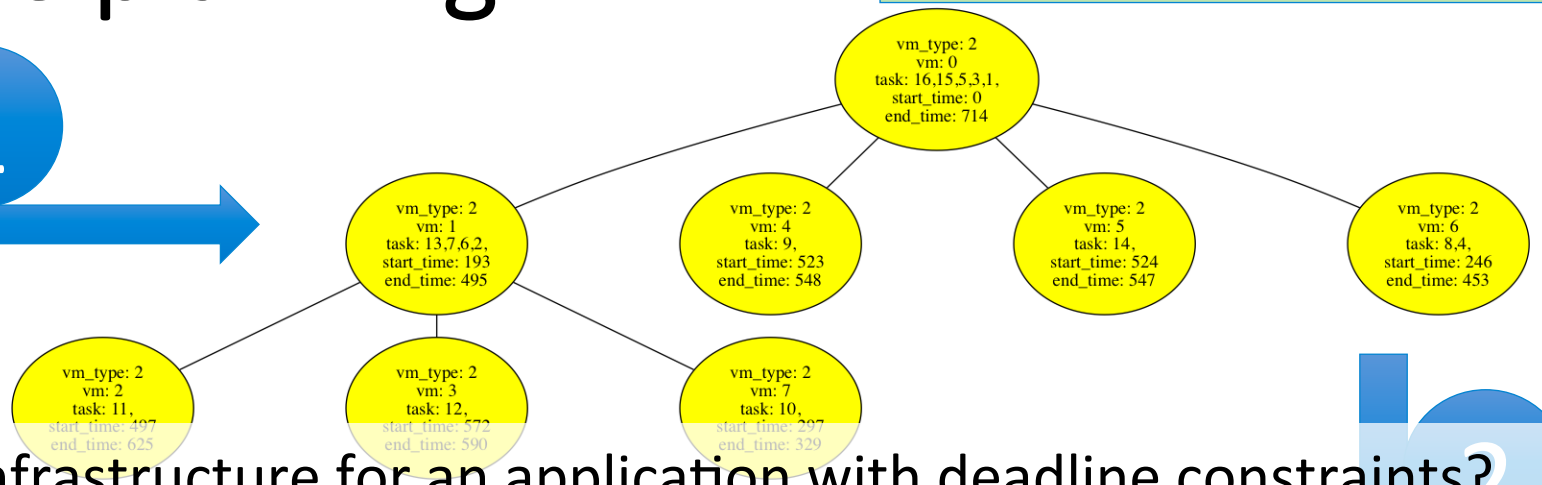
SWITCH

Application

1

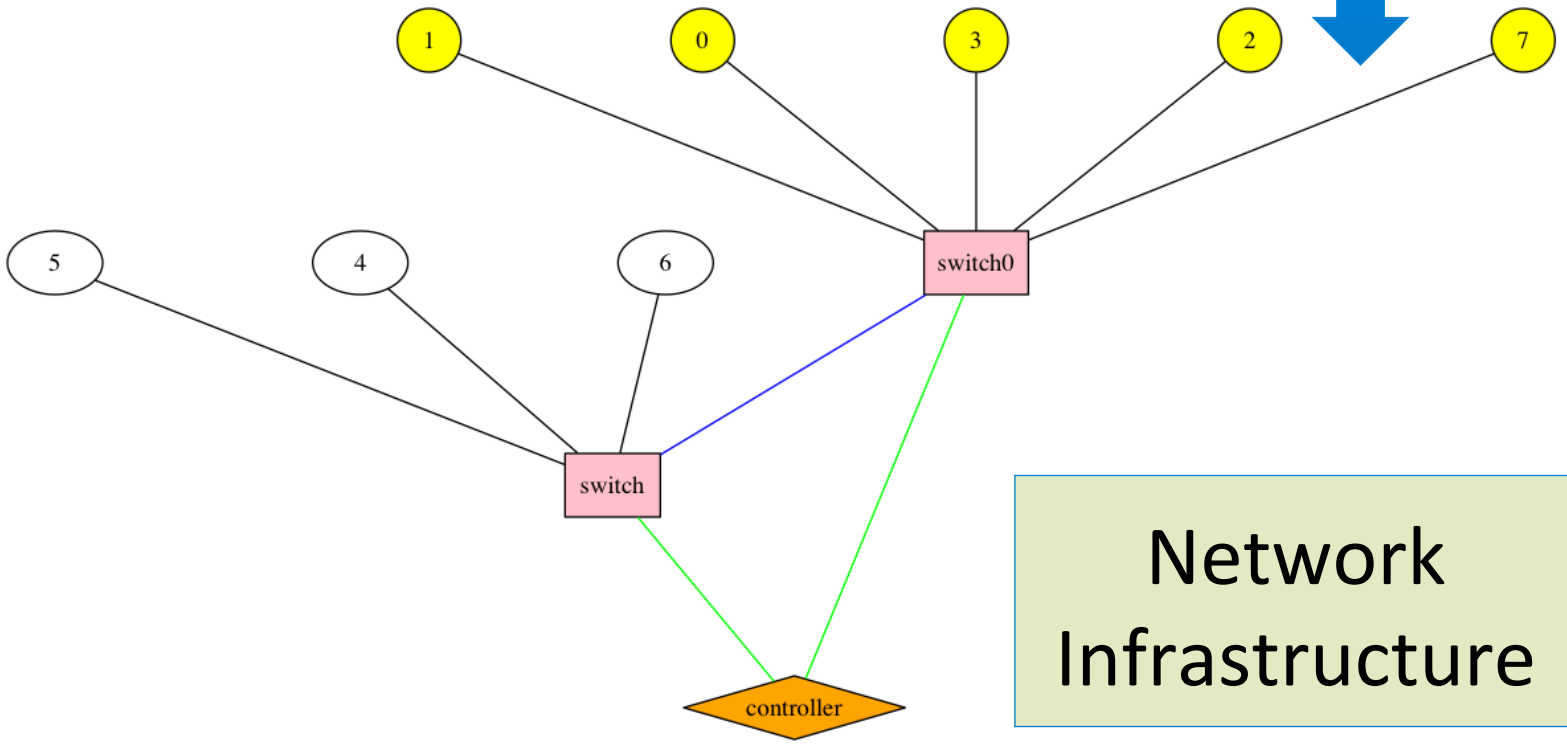
VM Infrastructure





Problem: How to plan the VM infrastructure for an application with deadline constraints?

2

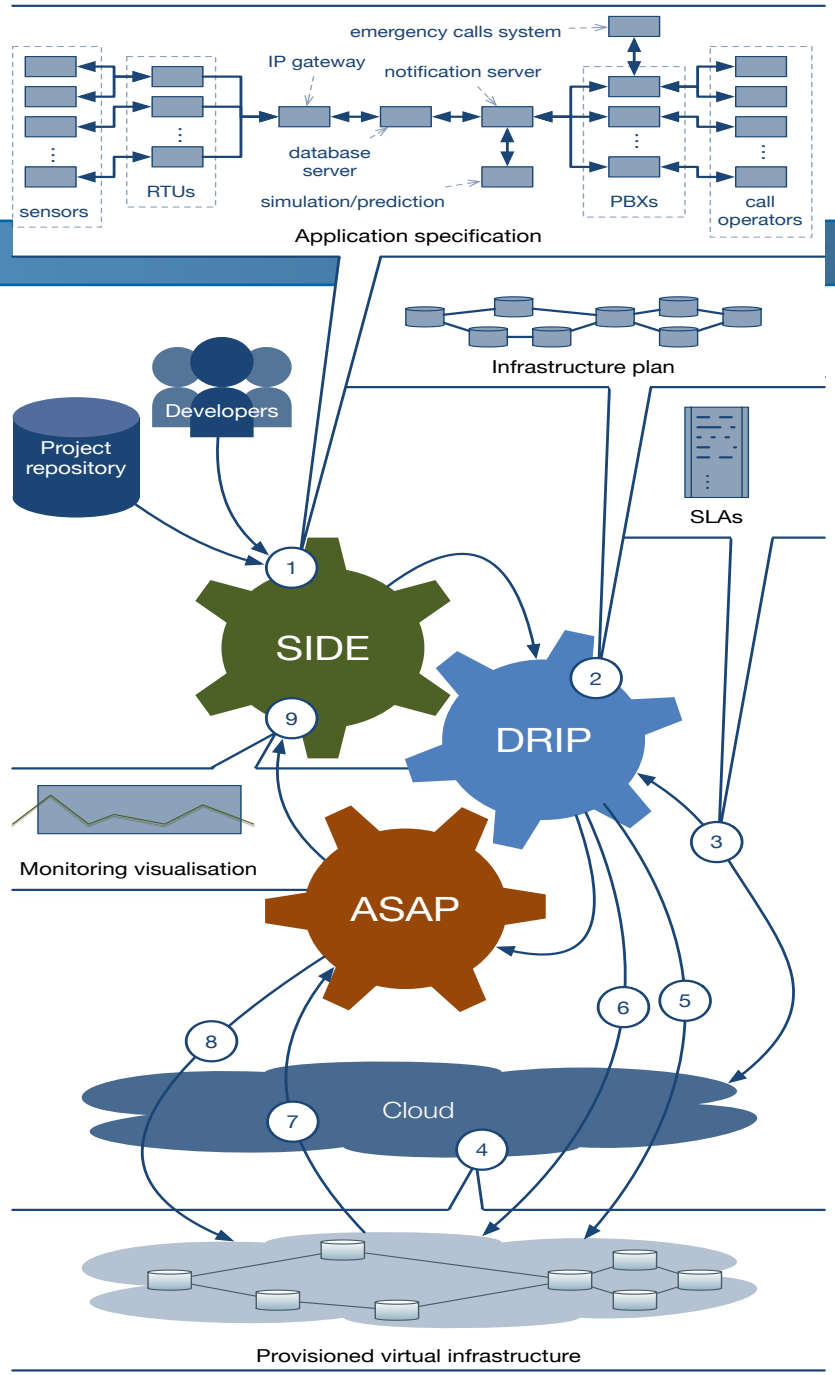


Network Infrastructure

SLA negotiation and infrastructure provisioning

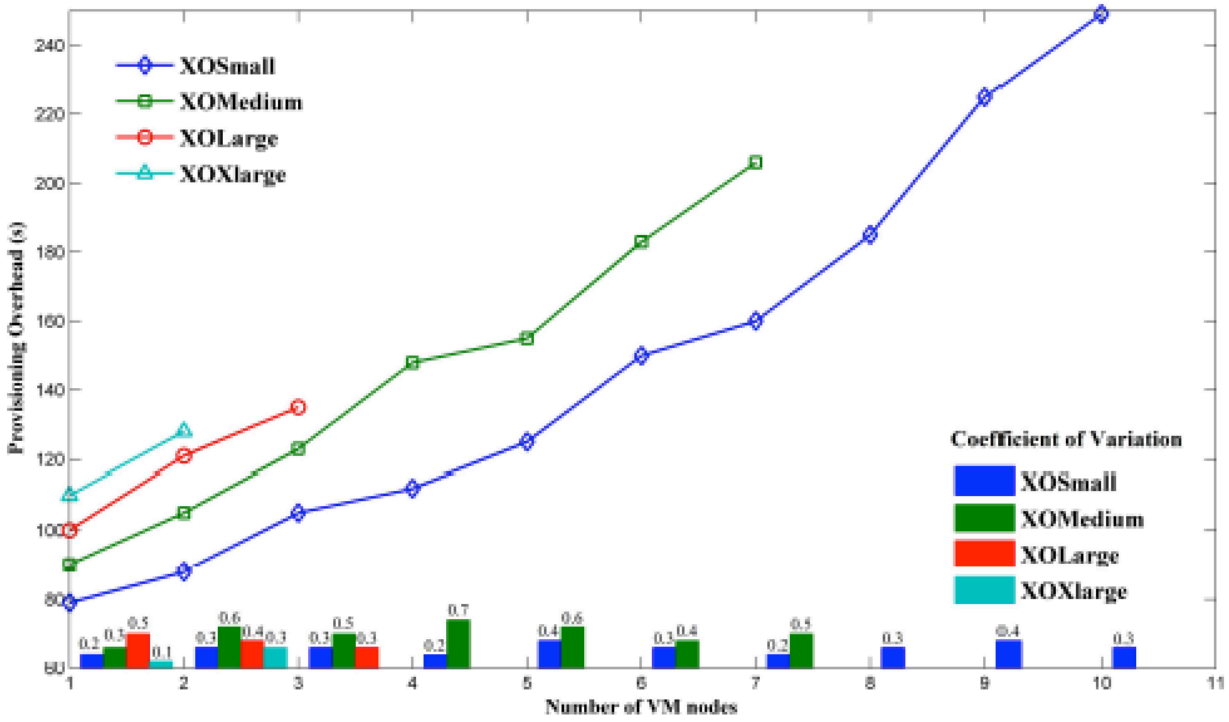
The SWITCH application lifecycle is split into a number of interlinked phases:

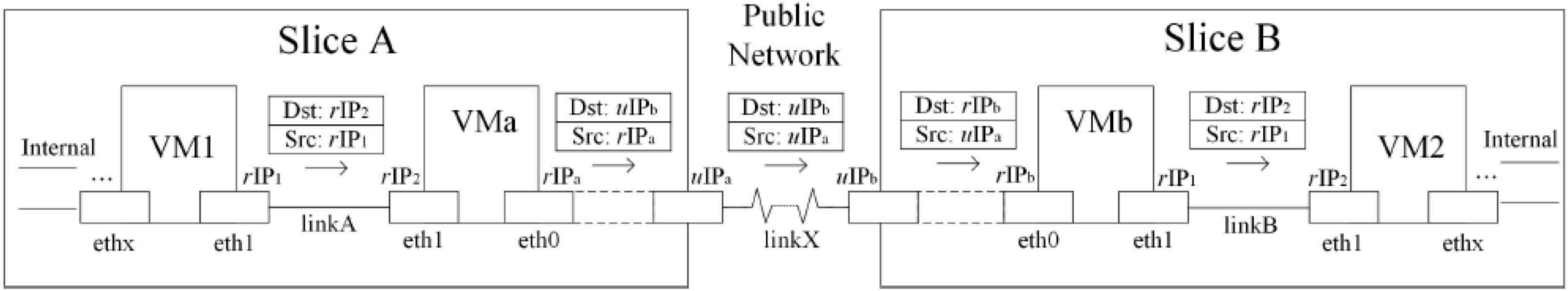
1. Application composition and verification.
2. Resource selection and infrastructure planning.
3. **SLA negotiation.**
4. **Infrastructure provisioning.**
5. Application deployment.
6. Application execution and runtime management.
7. Runtime monitoring and diagnosis.
8. Runtime adaptation.
9. Runtime visualisation and feedback.



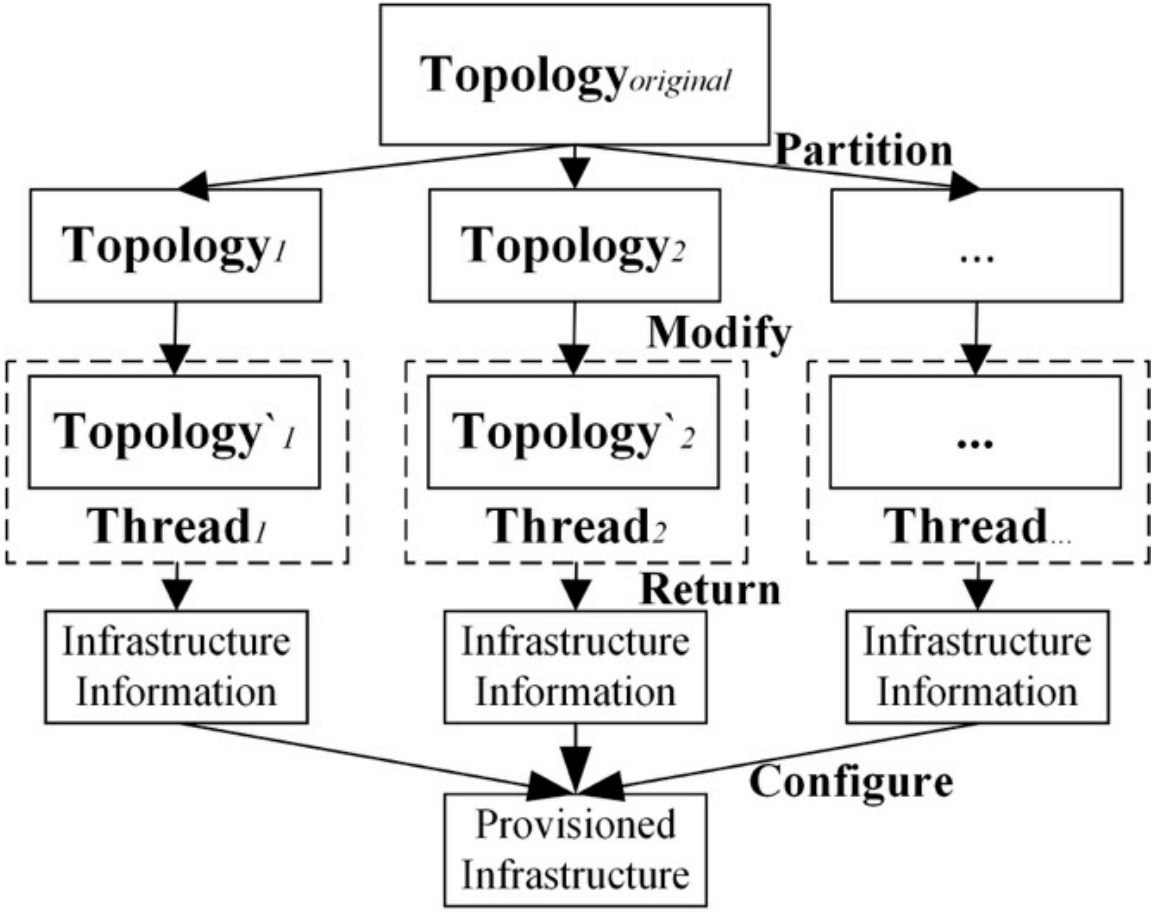
Experiments on ExoGENI

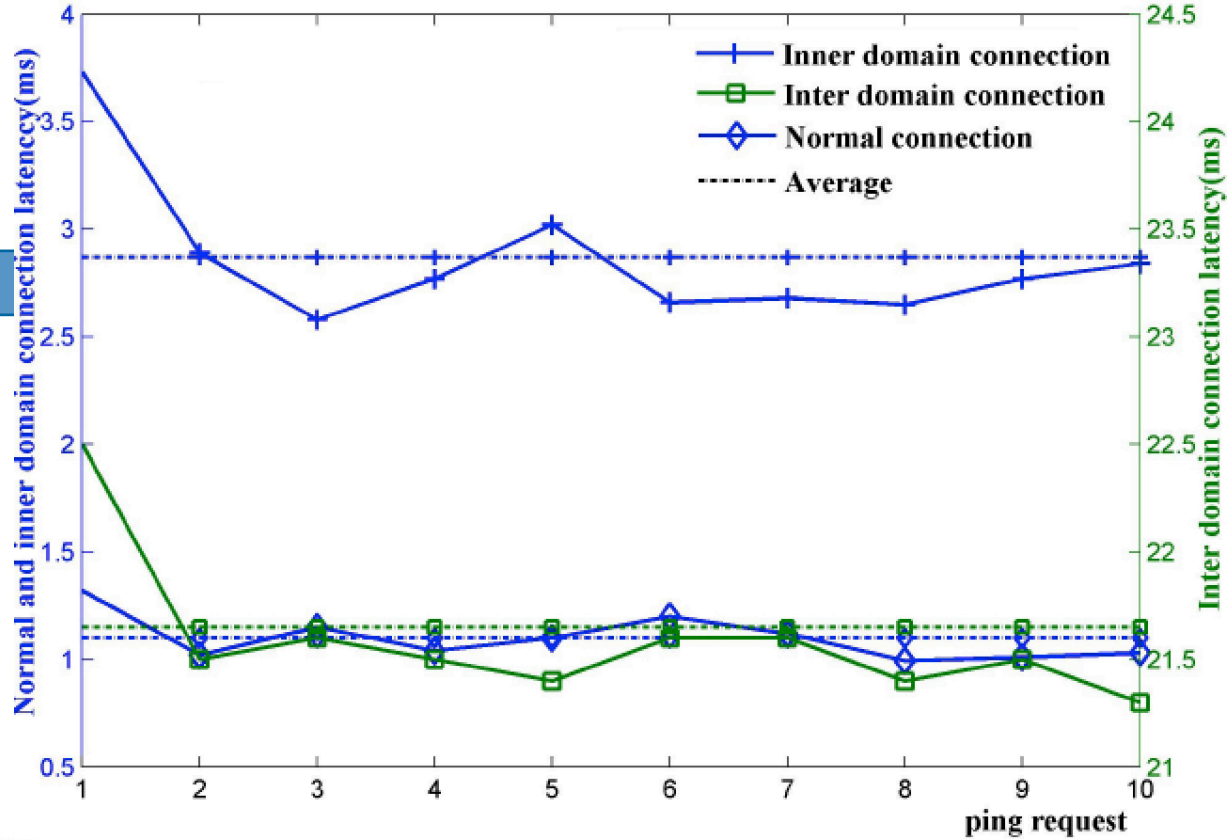
- Sequential provisioning results in high time cost
- Difficult to automatically provision multi RACK slide (virtual infrastructure)
- Network configuration between RACKs difficult

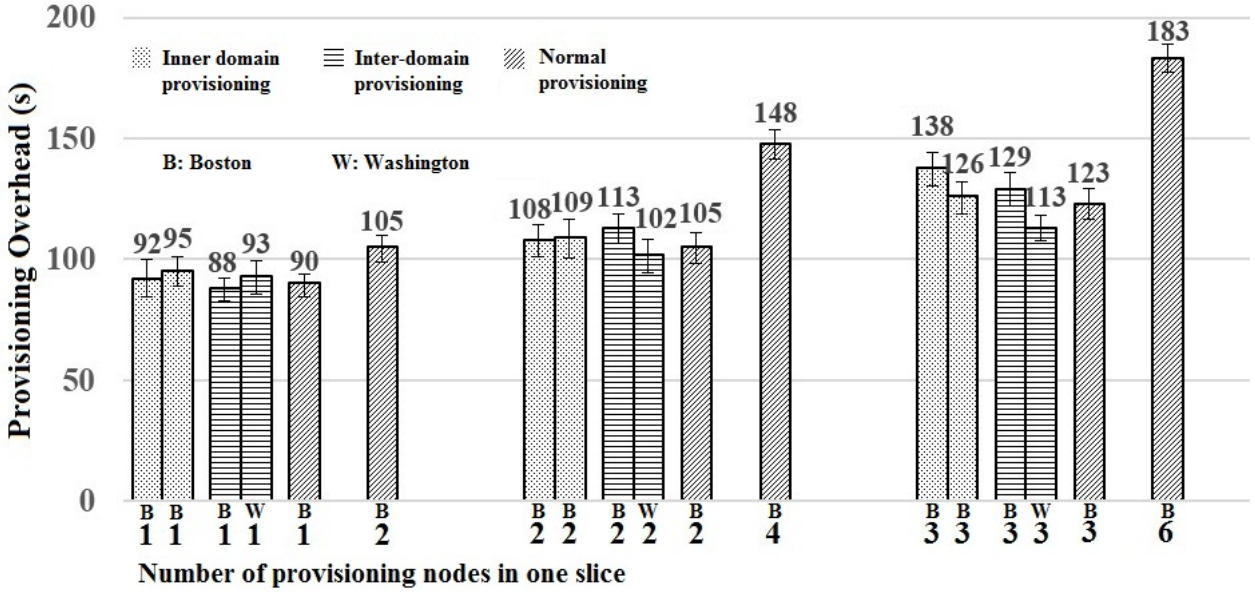




Experiments on ExoGENI



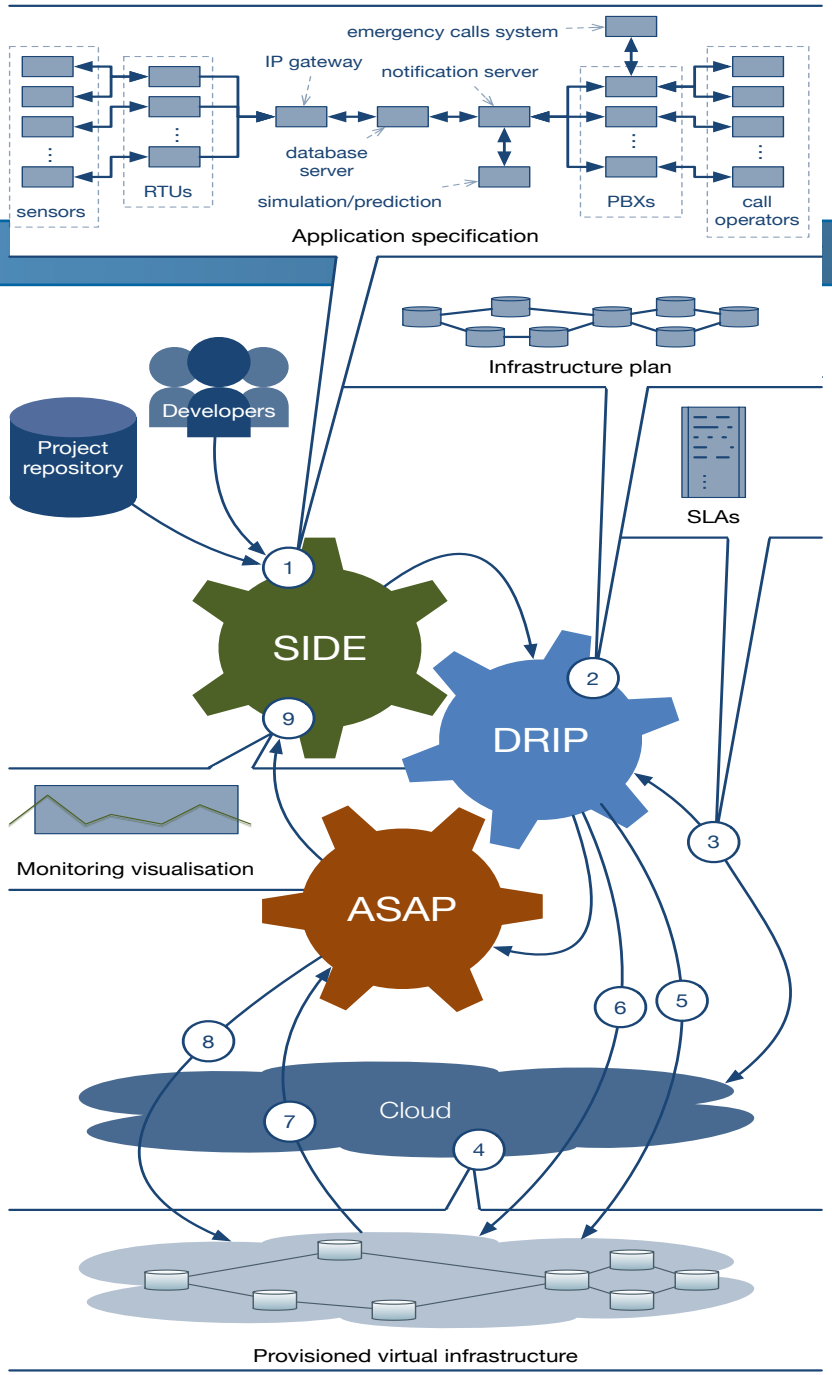




Deployment

The SWITCH application lifecycle is split into a number of interlinked phases:

1. Application composition and verification.
2. Resource selection and infrastructure planning.
3. SLA negotiation.
4. Infrastructure provisioning.
5. **Application deployment.**
6. **Application execution and runtime management.**
7. Runtime monitoring and diagnosis.
8. Runtime adaptation.
9. Runtime visualisation and feedback.



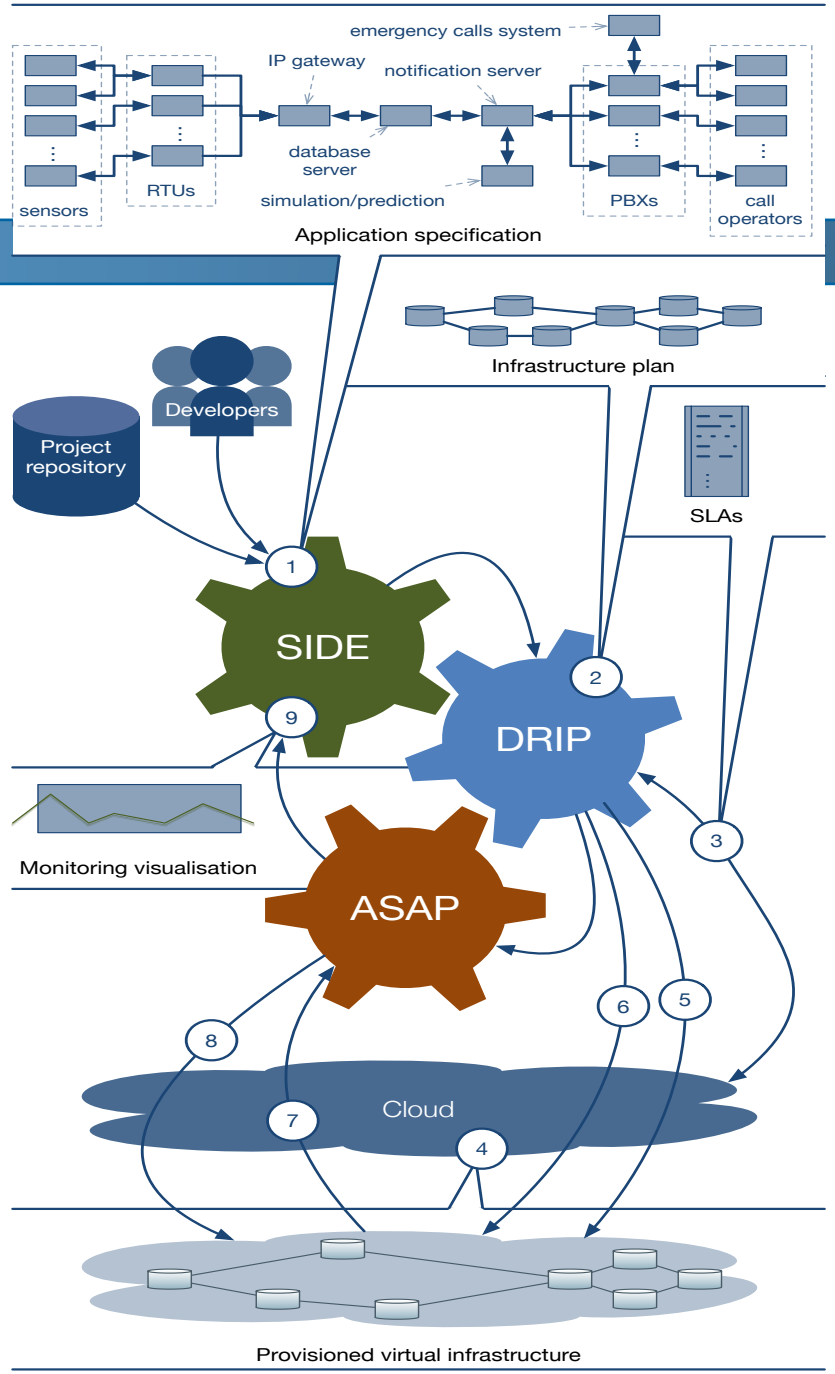
Control interface at different level

- Problem
  - How to provide a common and reusable interface for controlling the computing resources of the infrastructure?
- Approach
  - Investigate the suitability of a lightweight tool for orchestrating the deployment and linking of containers on the infrastructure that can be easily extended.

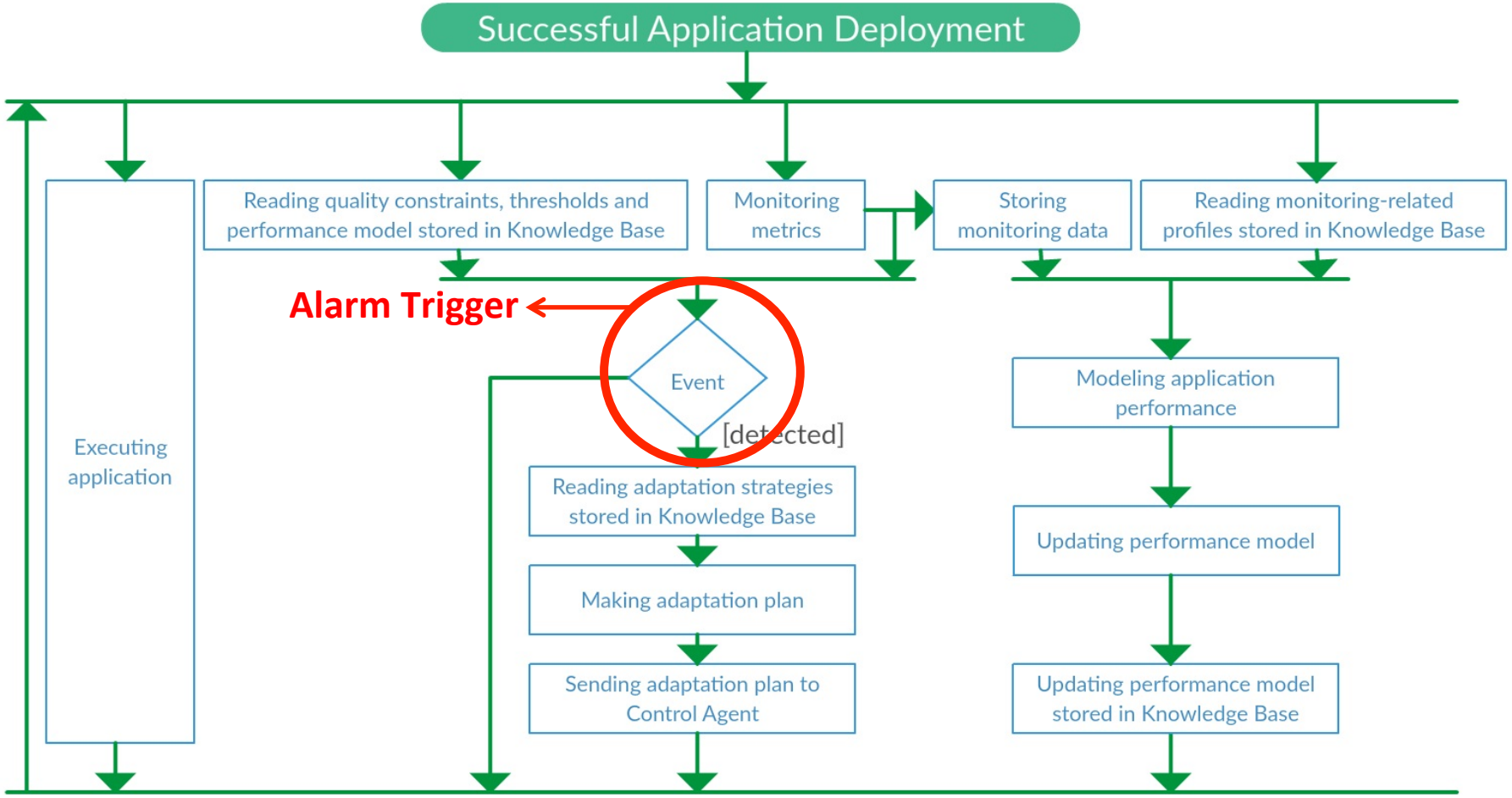
Runtime monitoring and adaptation

The SWITCH application lifecycle is split into a number of interlinked phases:

1. Application composition and verification.
2. Resource selection and infrastructure planning.
3. SLA negotiation.
4. Infrastructure provisioning.
5. Application deployment.
6. Application execution and runtime management.
7. **Runtime monitoring and diagnosis.**
8. **Runtime adaptation.**
9. Runtime visualisation and feedback.



Runtime monitoring framework



SWITCH semantic linking

- Problem
  - What is a suitable ontological framework for describing system components and their interfaces?
  - How best to map semantic information across different layers and link with existing standards (e.g. TOSCA, NML, INDL)?
  - Can we validate our own system design using formal descriptions?
- Approach
  - Develop an upper ontology for SWITCH.
  - Reuse existing models and link them within the SWITCH framework.
  - Apply a multi-viewpoint approach for distributed system design.

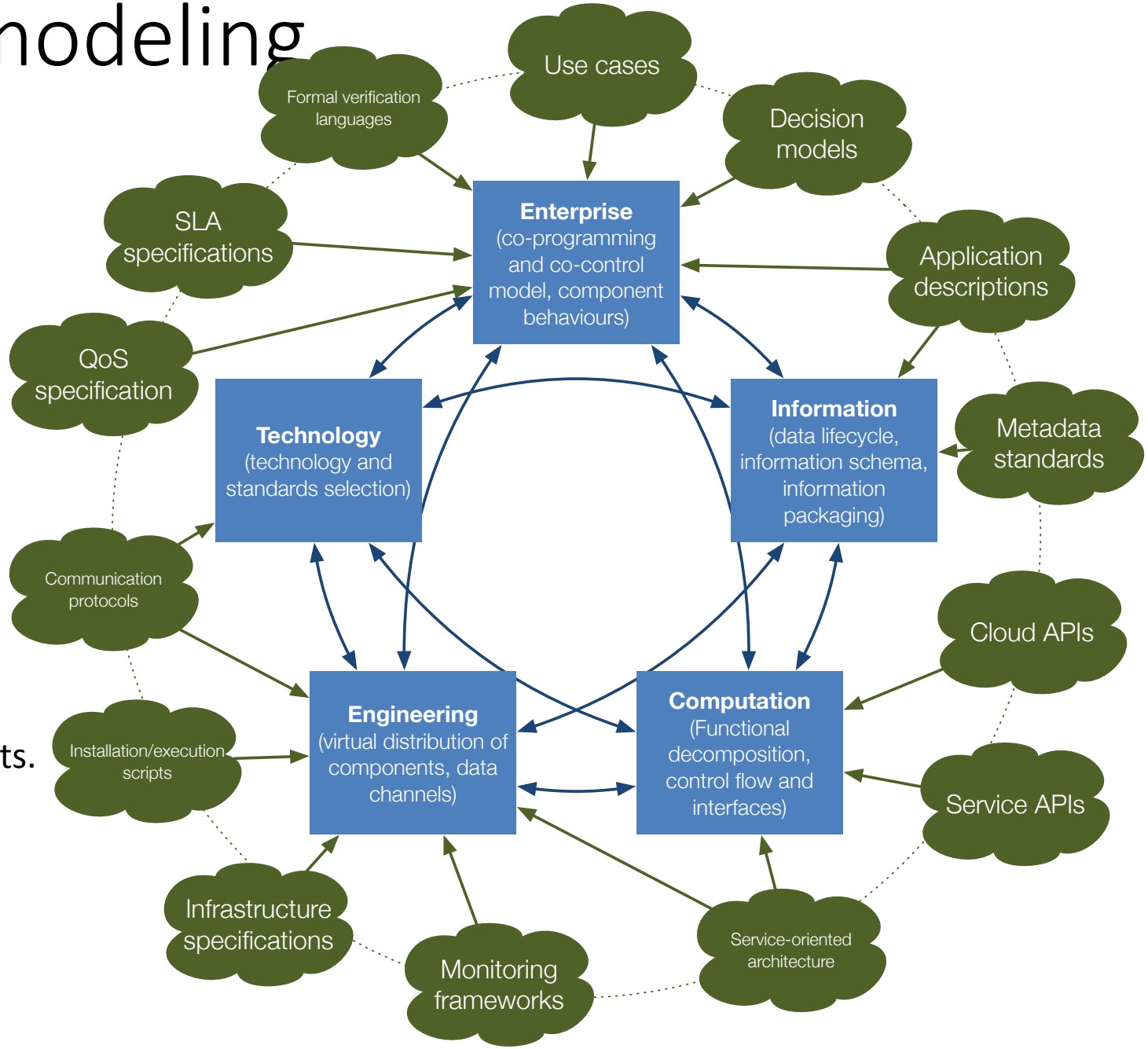
Information modeling

The application lifecycle requires a number of different **information objects**:

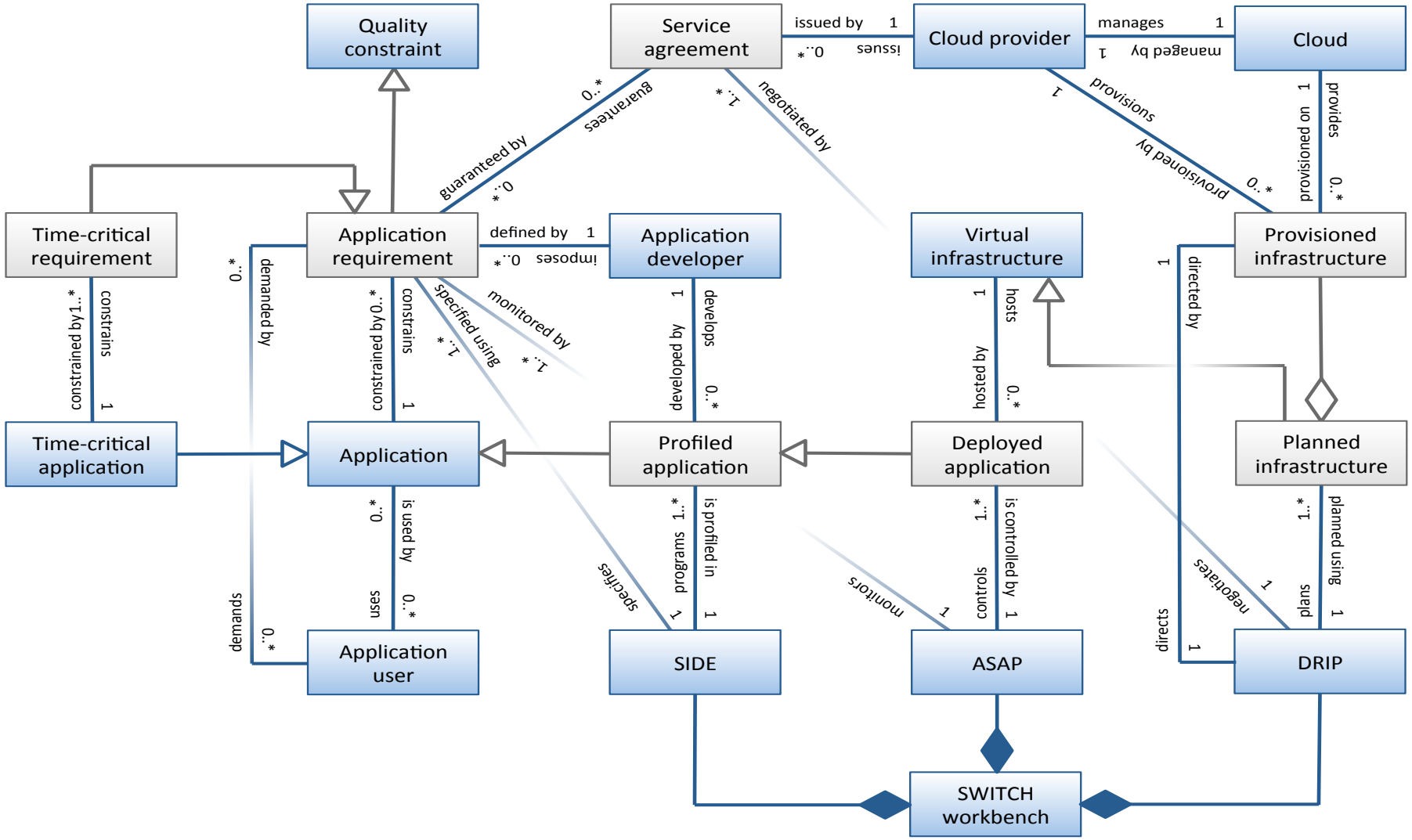
- Application profiles, infrastructure profiles, monitoring profiles, control profiles, adaptation profiles, etc.
- Most objects can be realised using **existing standards** such as BPEL, TOSCA, INDL and OCCI.

However, SWITCH needs to be **technology agnostic** where possible.

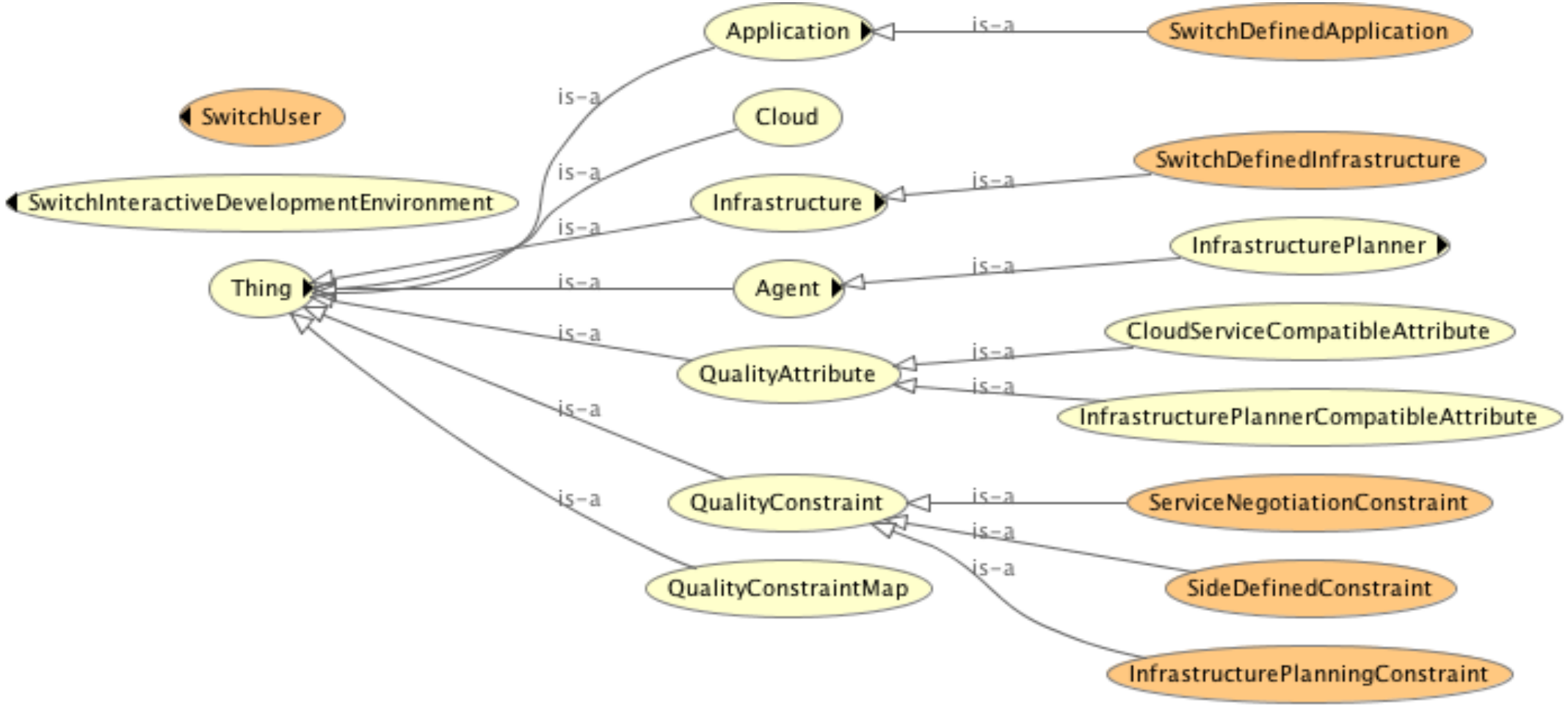
- To support this, need a standard reference model for SWITCH.
- Based on ODP—distributed system, five viewpoints.



Key concepts in SWITCH



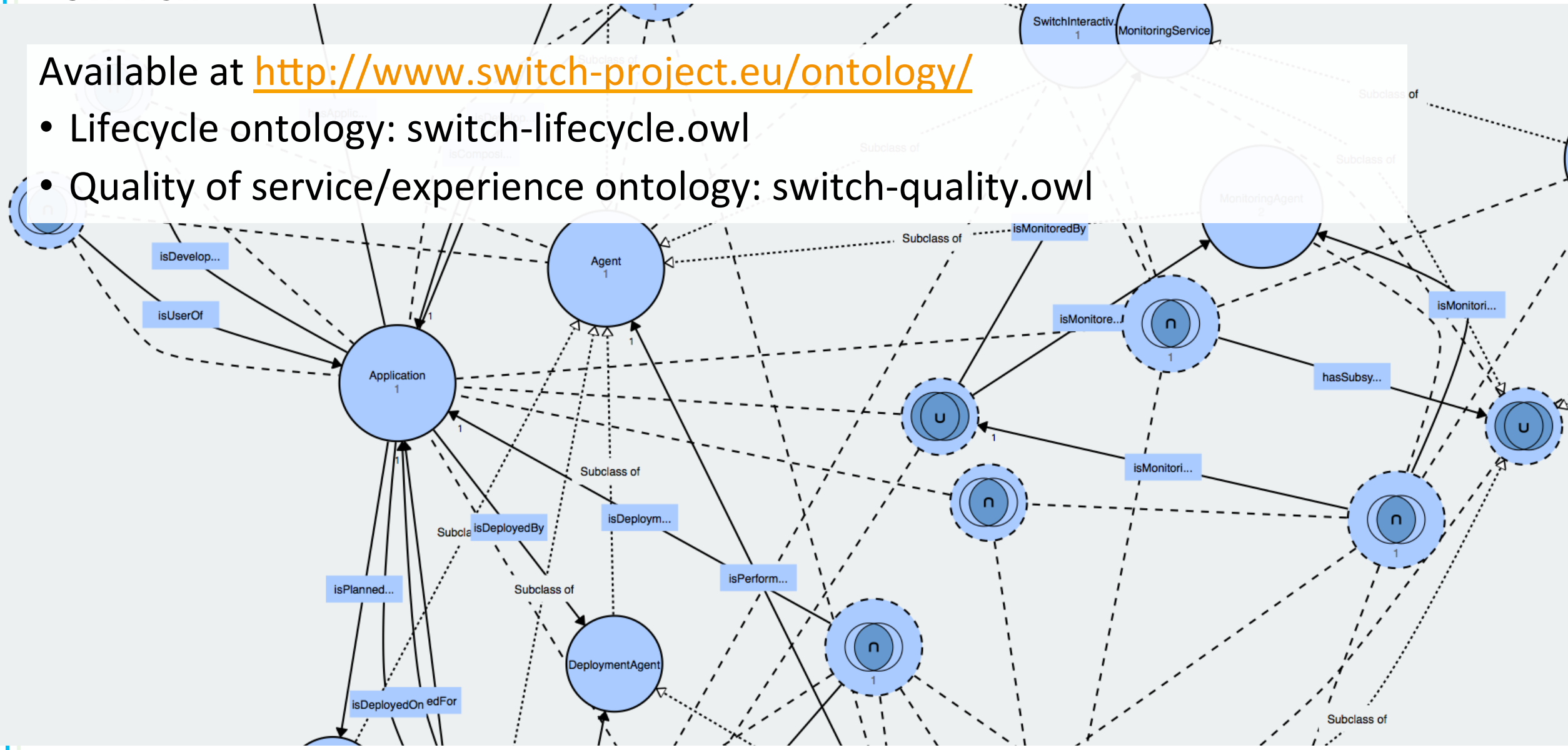
SWITCH quality ontology (initial framework)



Current ontology

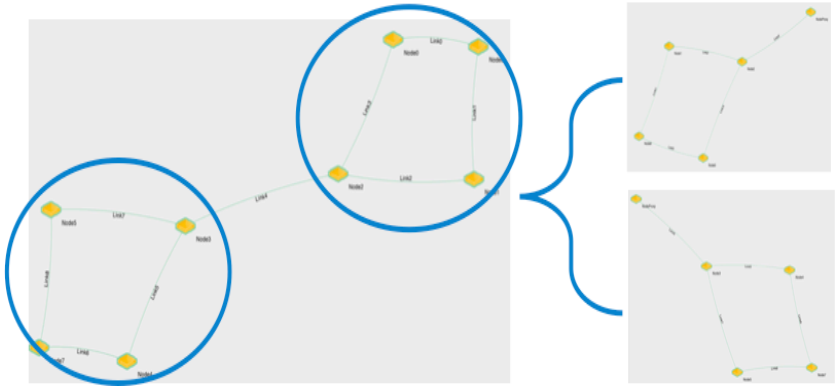
Available at http://www.switch-project.eu/ontology/

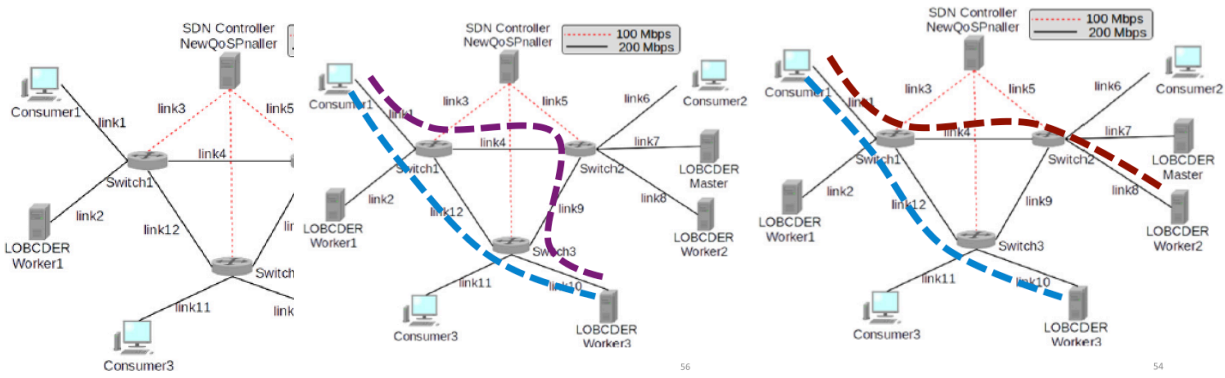
- Lifecycle ontology: switch-lifecycle.owl
- Quality of service/experience ontology: switch-quality.owl



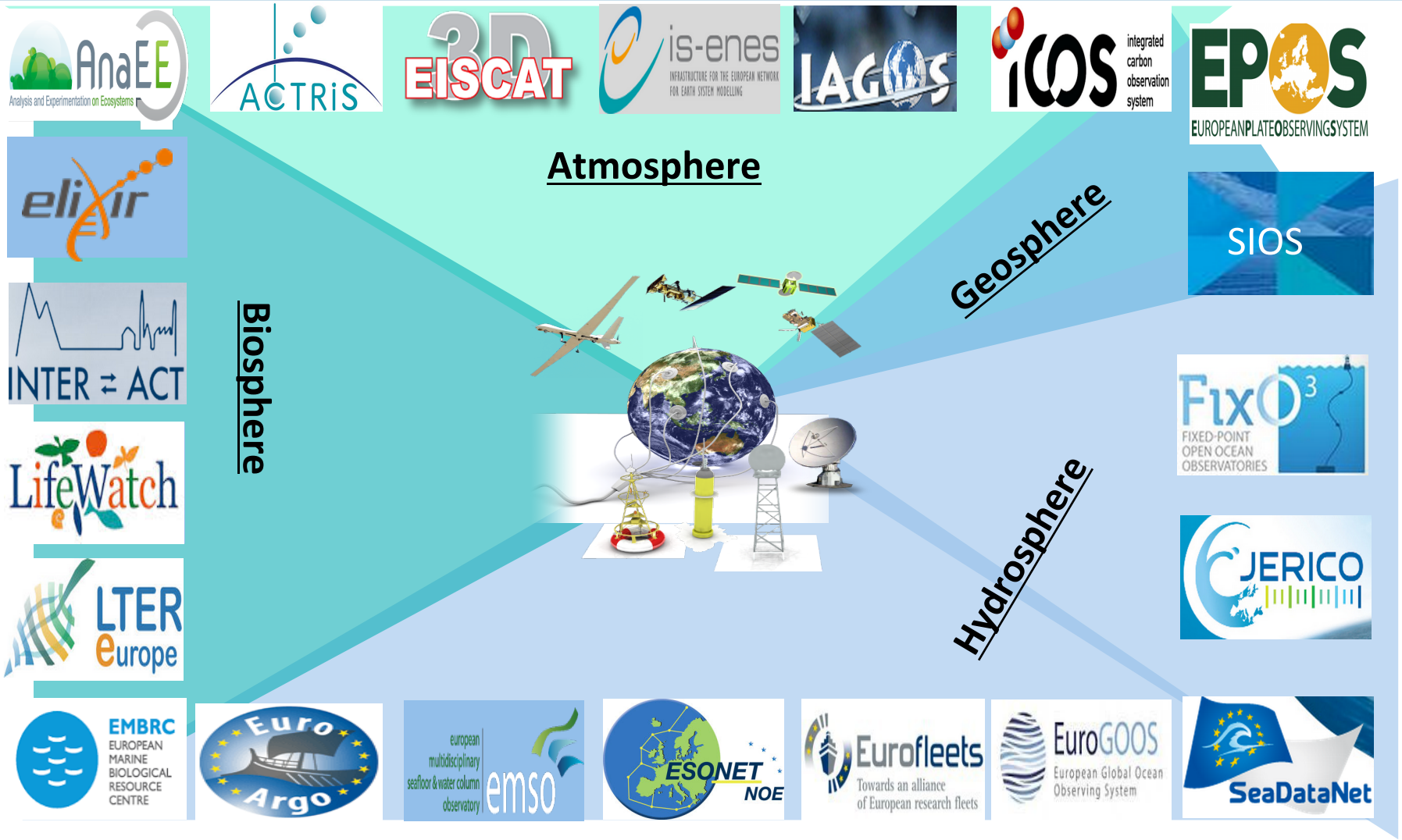
Current results

- Describe network topology, virtual resources and software services (INDL) , and workflow (DAG) to plan virtual infrastructure
- Describe the virtual infrastructure for parallel partitioning
- Describe network topology for adapting network paths using SDN





Exploitation in RI development



From RI to VRE

(a) Access to datasets
(b) Access to (APIs of) software services
(c) Access to computing resources (including GRIDs and CLOUDs)
(d) Access to equipment/detectors for data taking and parameter control

Summary

- Semantic linking is important for time critical cloud applications
- Identifying potential tasks help us limit the scope and focus on urgent semantic linking issues
- Open distributed processing provides an effective multi viewpoint mechanism to structure and link different semantic models

About the SWITCH project

- Running from Feb 2015 – 2018.
- Total budget 2.92 M.
- EU H2020 ICT-9.
- Web: http://www.switchproject.eu/.

- Affiliated projects/initiatives:

Environmental Research Infrastructures Providing Shared Solutions for Science and Society