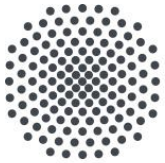
Quantum Workflows



University of Stuttgart

Martin Beisel, Benjamin Weder

{beisel,weder}@iaas.uni-stuttgart.de

Institute of Architecture of Application Systems

PlanQK

SeQuenC

EniQmA

Tutorial Structure

- Session 1 (09:00 - 10:30): An Introduction to Quantum Computing
- Session 2 (11:00 - 12:30): Quantum Software Engineering
- **Session 3 (14:00 - 15:30): Quantum Workflows**
  - Quantum Workflows
  - Service-oriented Quantum Applications
  - Introduction to Hands-On Session
  - Hands-On Session Part 1
- Session 4 (16:00 - 17:30): Operation of Hybrid Quantum Applications

Tutorial Structure

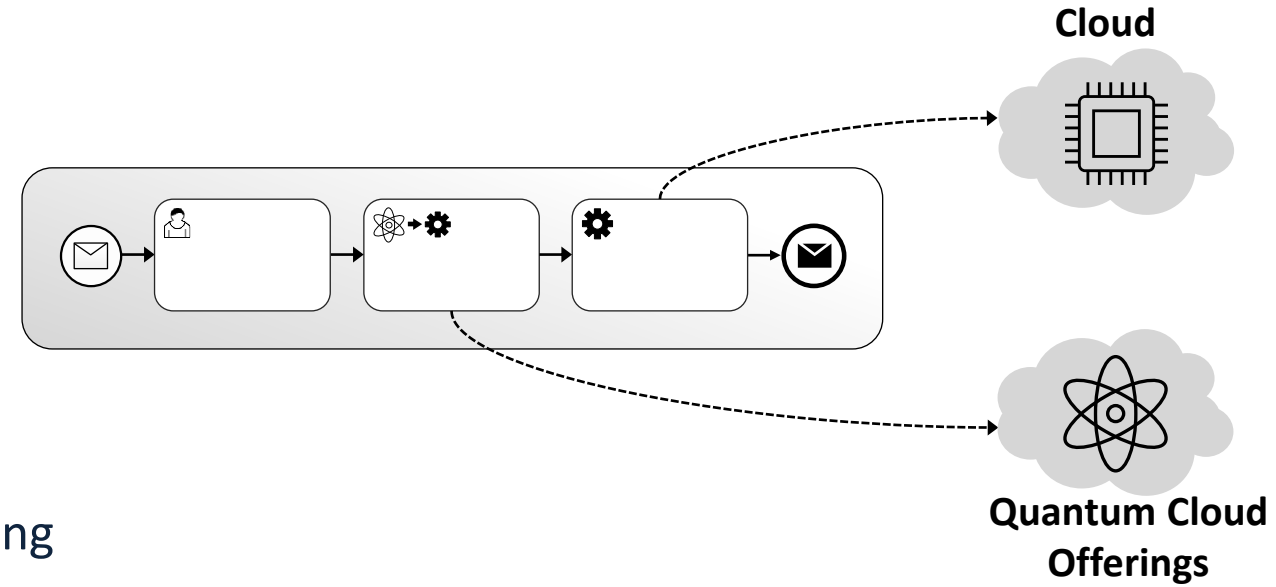
- Session 1 (09:00 - 10:30): An Introduction to Quantum Computing
- Session 2 (11:00 - 12:30): Quantum Software Engineering
- **Session 3 (14:00 - 15:30): Quantum Workflows**
  - **Quantum Workflows**
  - Service-oriented Quantum Applications
  - Introduction to Hands-On Session
  - Hands-On Session Part 1
- Session 4 (16:00 - 17:30): Operation of Hybrid Quantum Applications

Motivation

Motivation

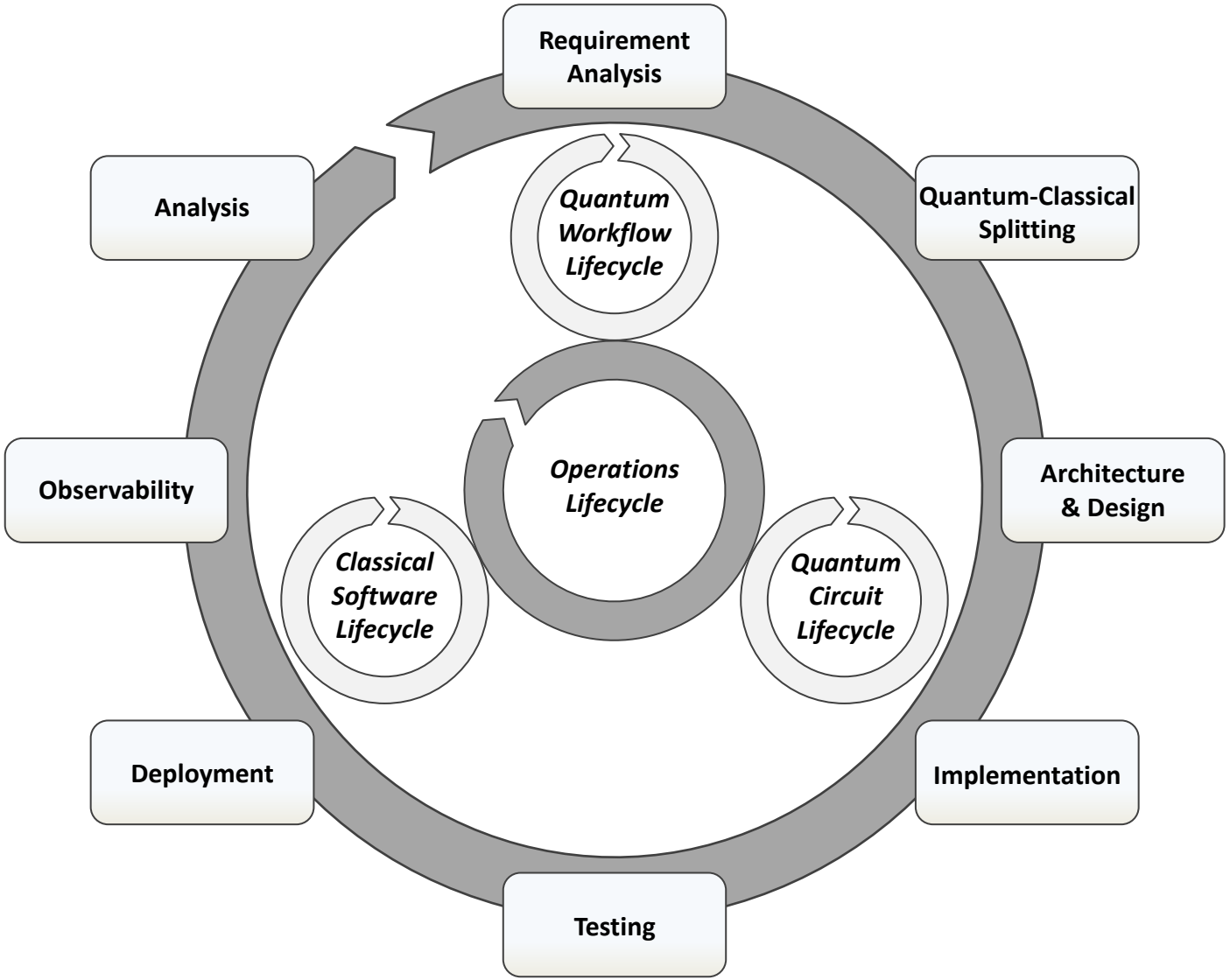
- Workflows enable orchestration and integration of heterogeneous applications
  - Definition of activities, control flow, and data flow

- Advantages:
  - Scalability
  - Robustness
  - Monitoring
  - Advanced Exception Handling
  - Portability via standardized languages (BPMN, BPEL)

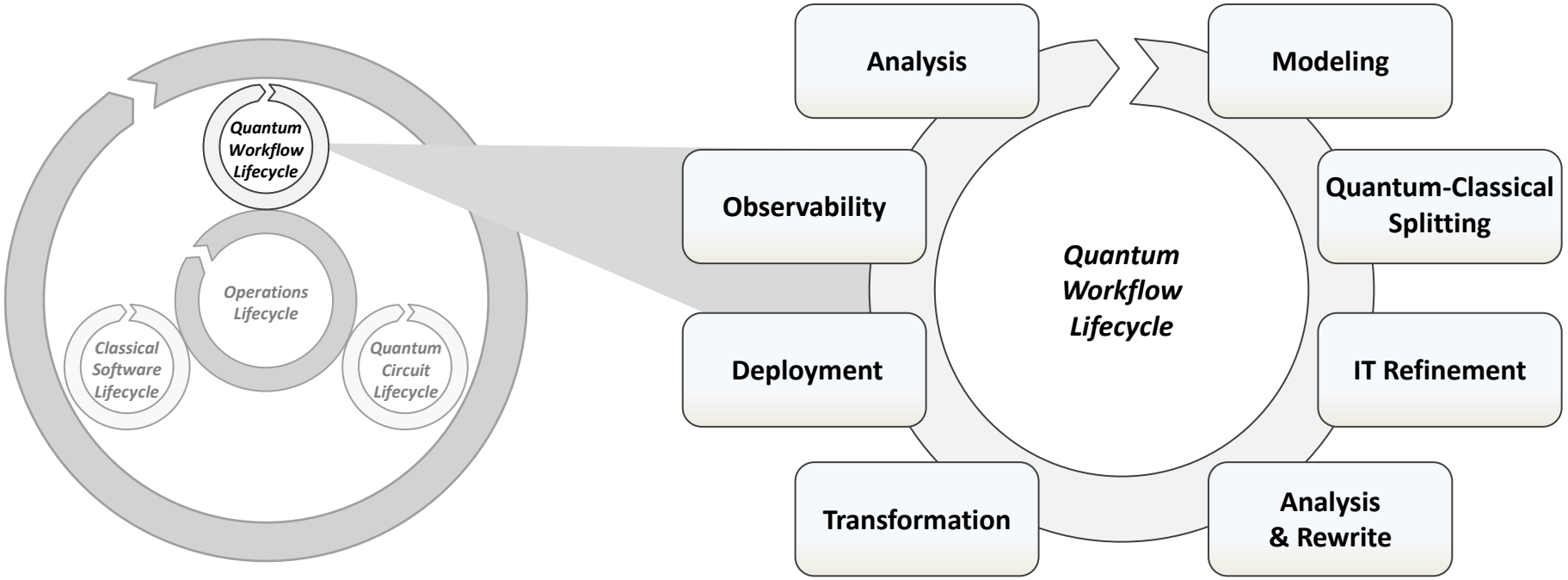


Quantum Workflow Lifecycle

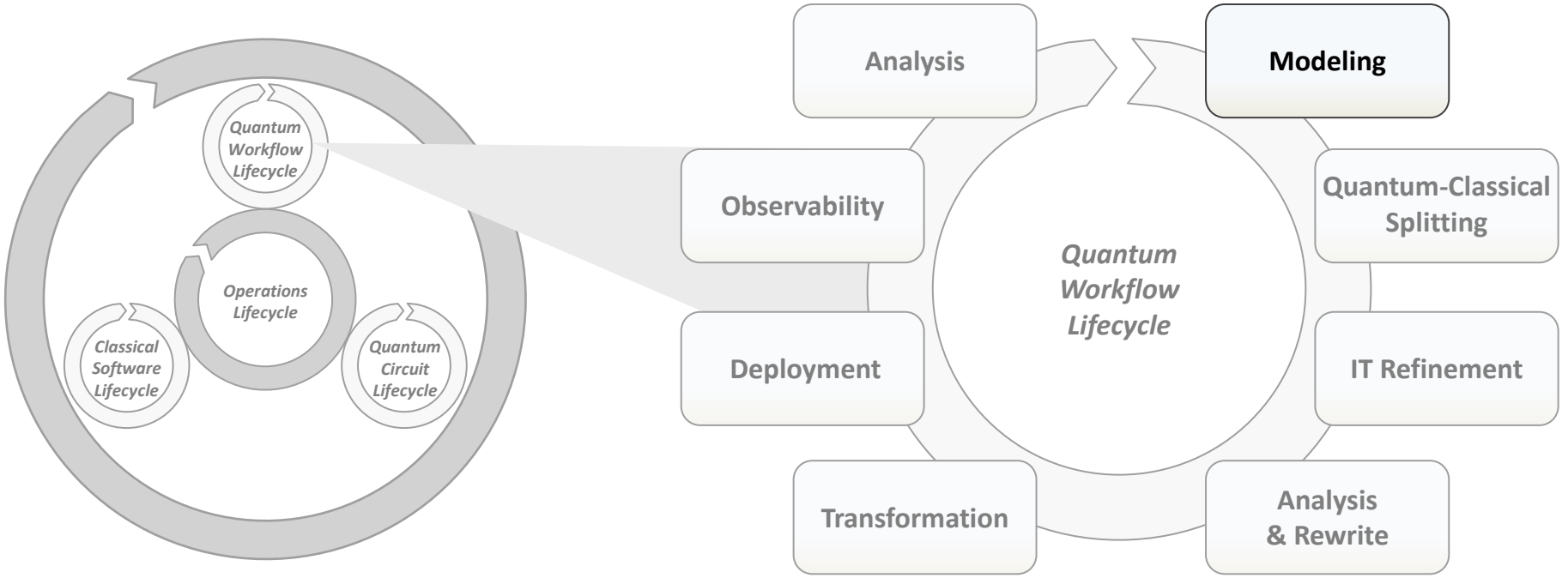
Recap: Quantum Software Lifecycle



Detailed View of the Quantum Workflow Lifecycle



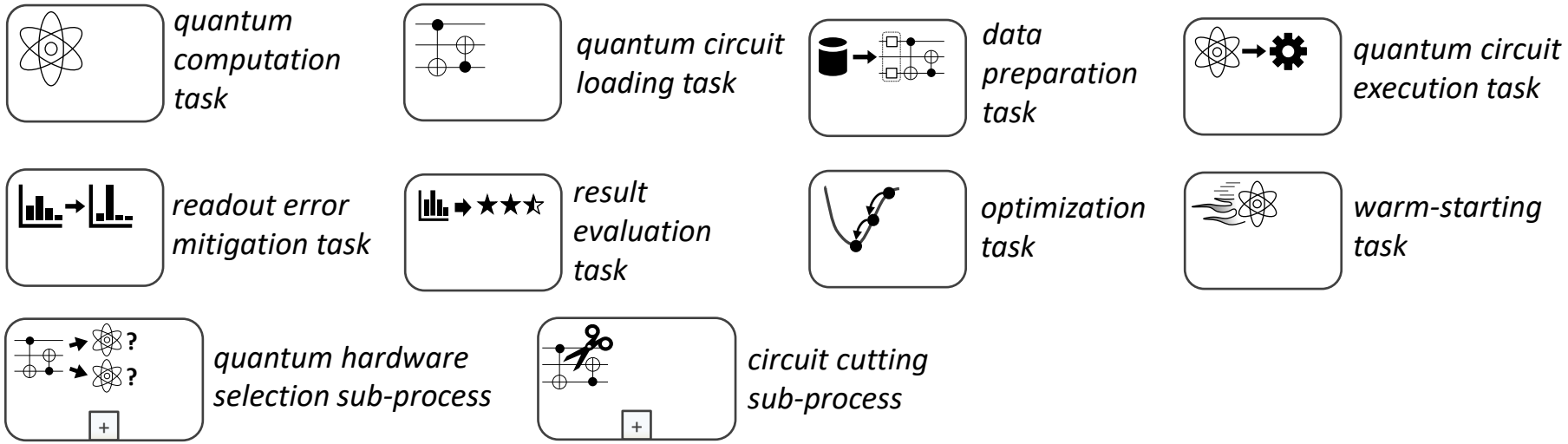
Detailed View of the Quantum Workflow Lifecycle



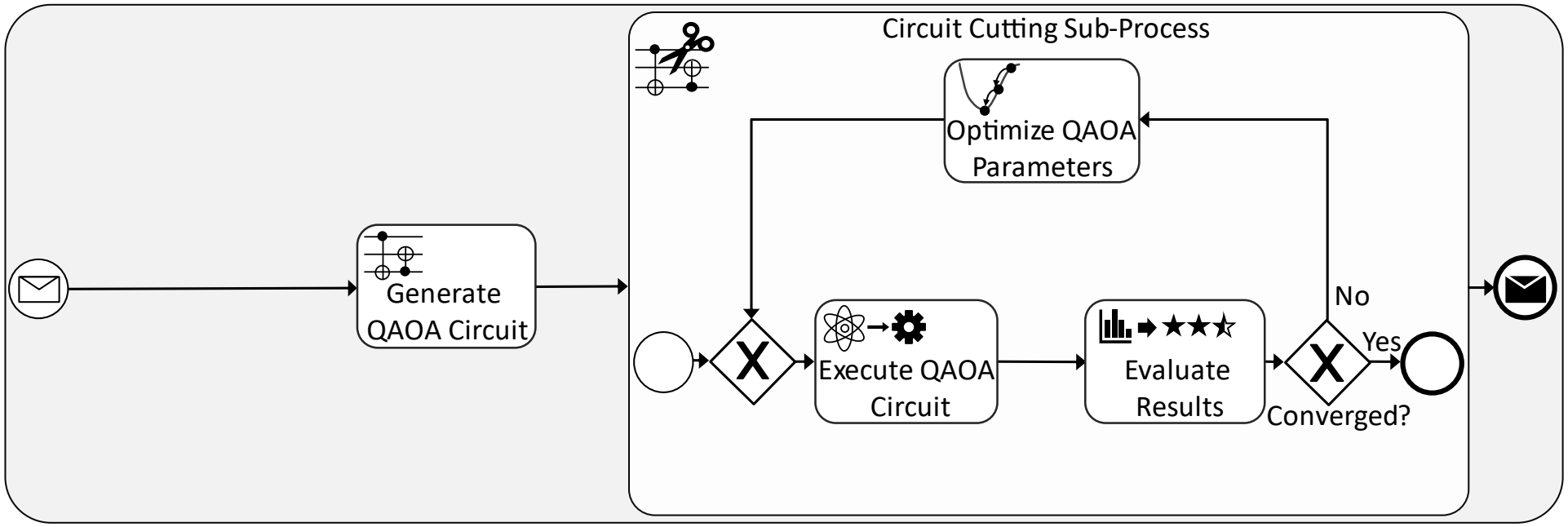
Quantum Modeling Extension (QuantME)

- Modeling extension for imperative workflow languages

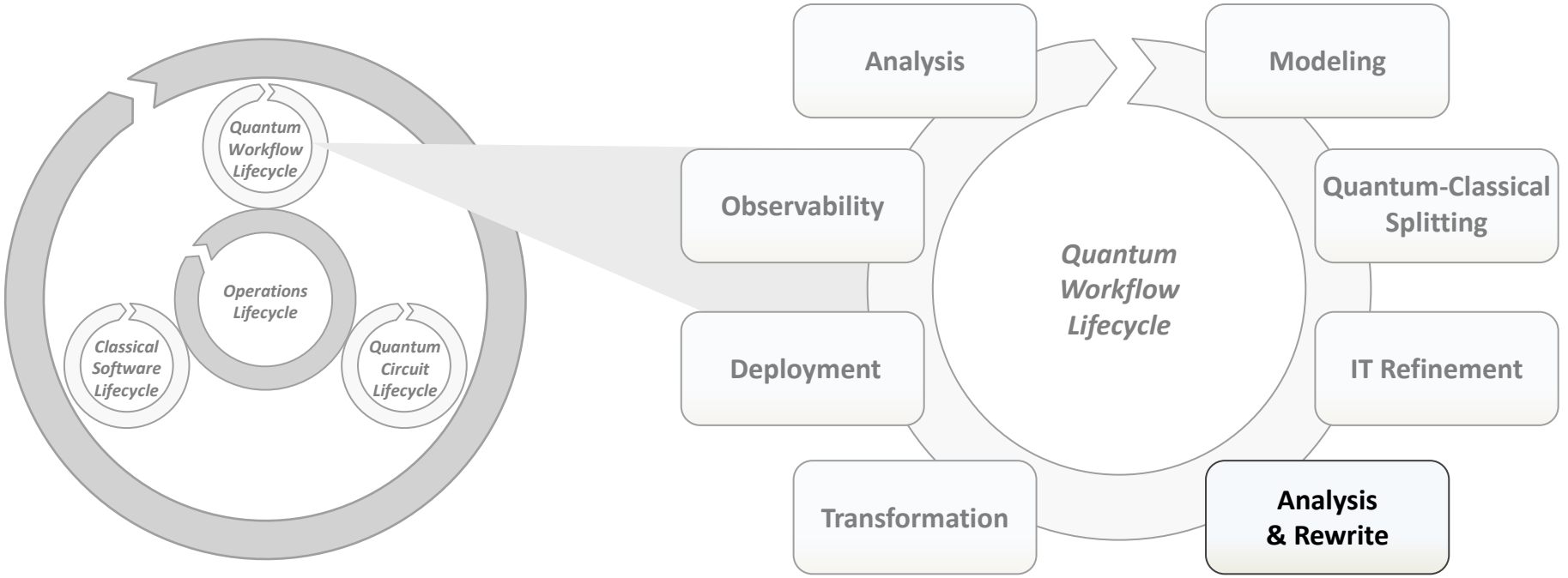
- Facilitates the modeling of quantum applications
  - Quantum-specific modeling constructs



Exemplary Orchestration of a Variational Quantum Algorithm

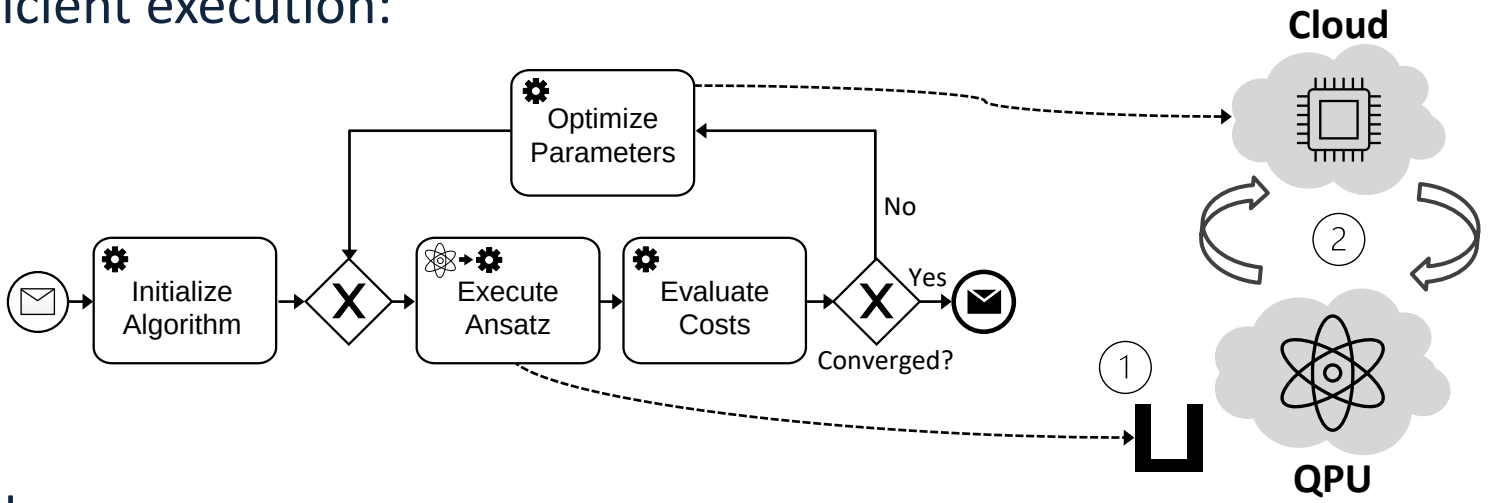


Detailed View of the Quantum Workflow Lifecycle



Executing Hybrid Loops using Workflows

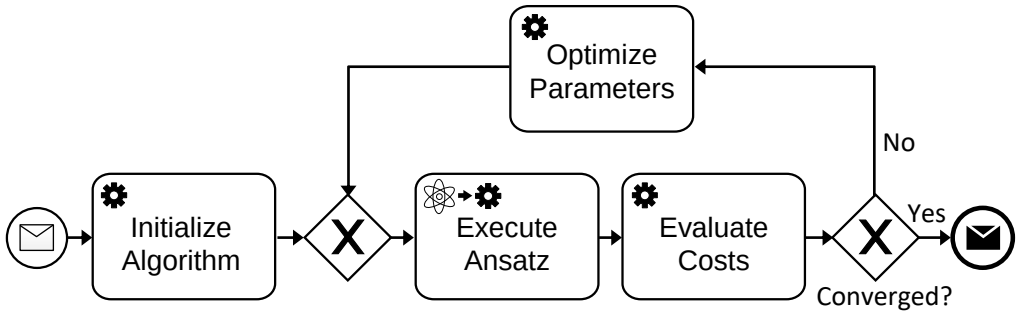
- Inefficient execution:

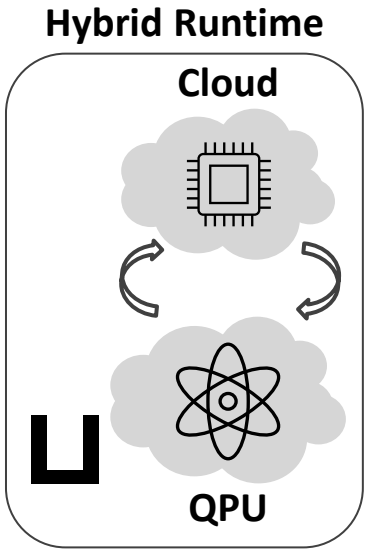


- Problems:
  - 1. Latency due to queue-based access
    ➔ Reserve time slot for complete execution
  - 2. Data transfer between classical and quantum parts inefficient
    ➔ Deploy quantum and classical parts closely together

Executing Hybrid Loops using Workflows

- Inefficient execution:

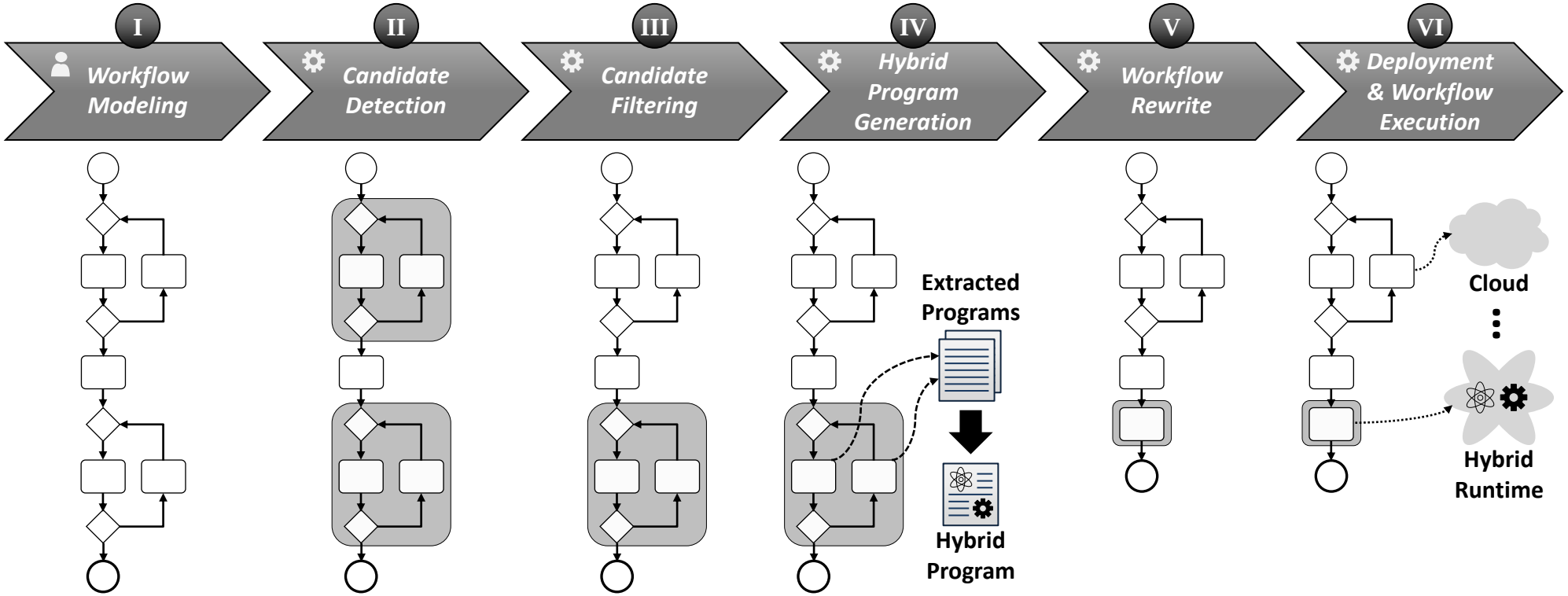




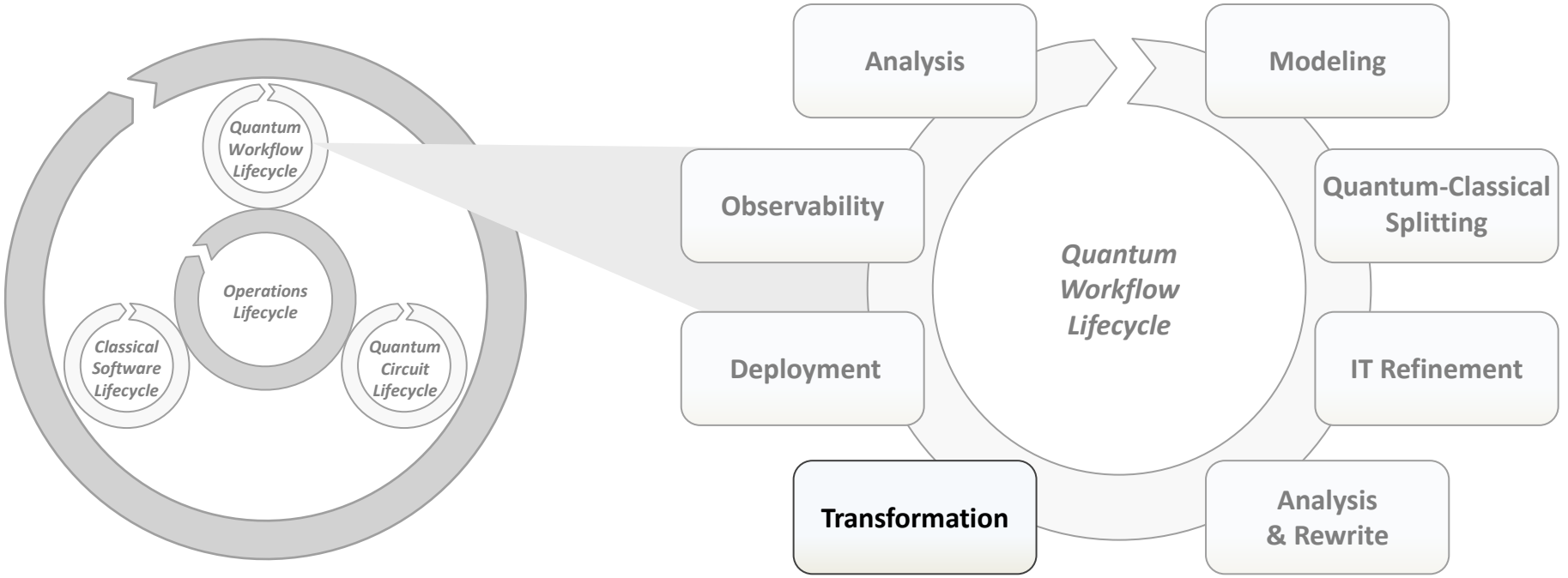
- Problems:
  - 1. Latency due to queue-based access
    ➔ Reserve time slot for complete execution
  - 2. Data transfer between classical and quantum parts inefficient
    ➔ Deploy quantum and classical parts closely together

➔ Usage of hybrid runtimes (e.g., Qiskit Runtime) for execution of hybrid loops

Analysis and Rewrite Method

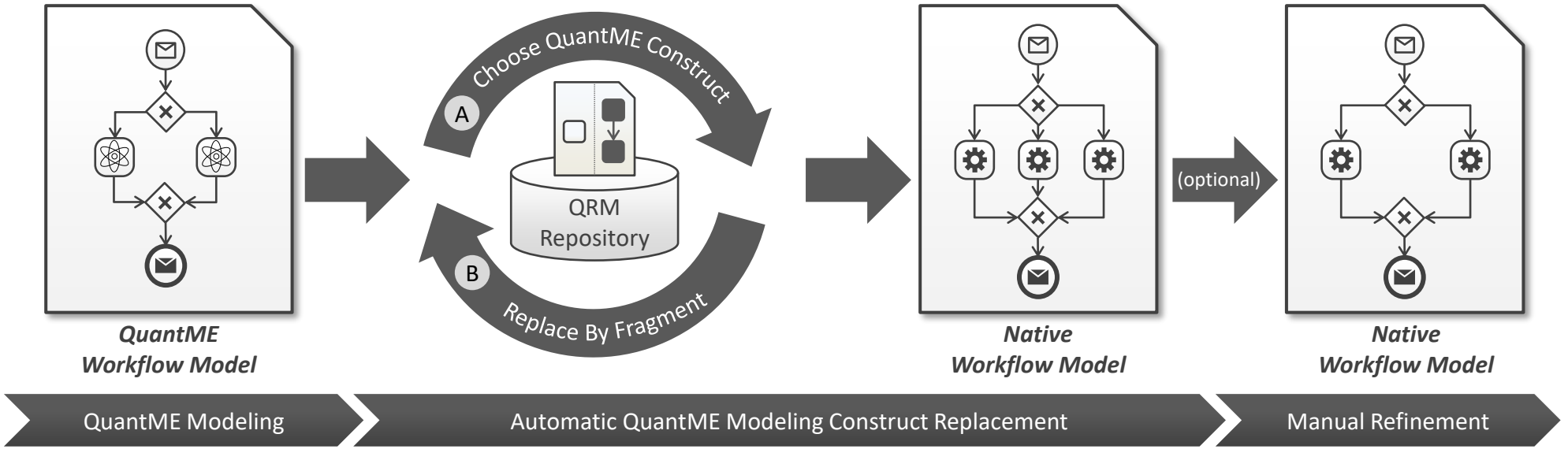


Detailed View of the Quantum Workflow Lifecycle



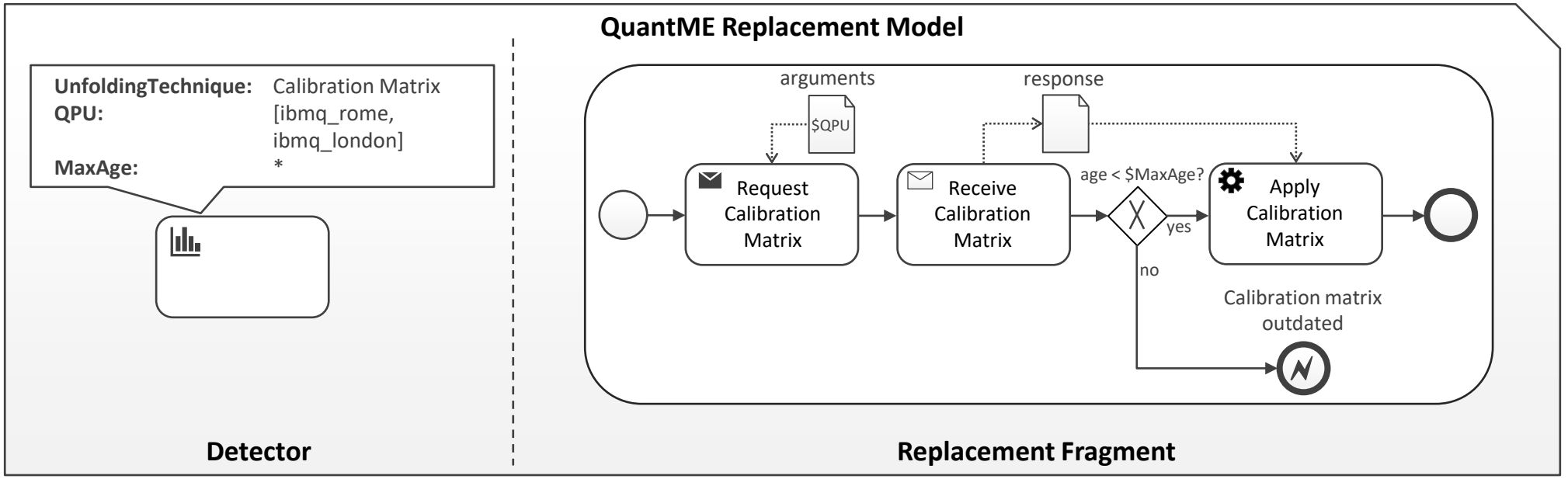
Transformation

- Transformation to native workflow language, e.g., BPMN
  - Portability
  - Compatibilty with existing workflow engines
- Transformation based on reusable workflow fragments

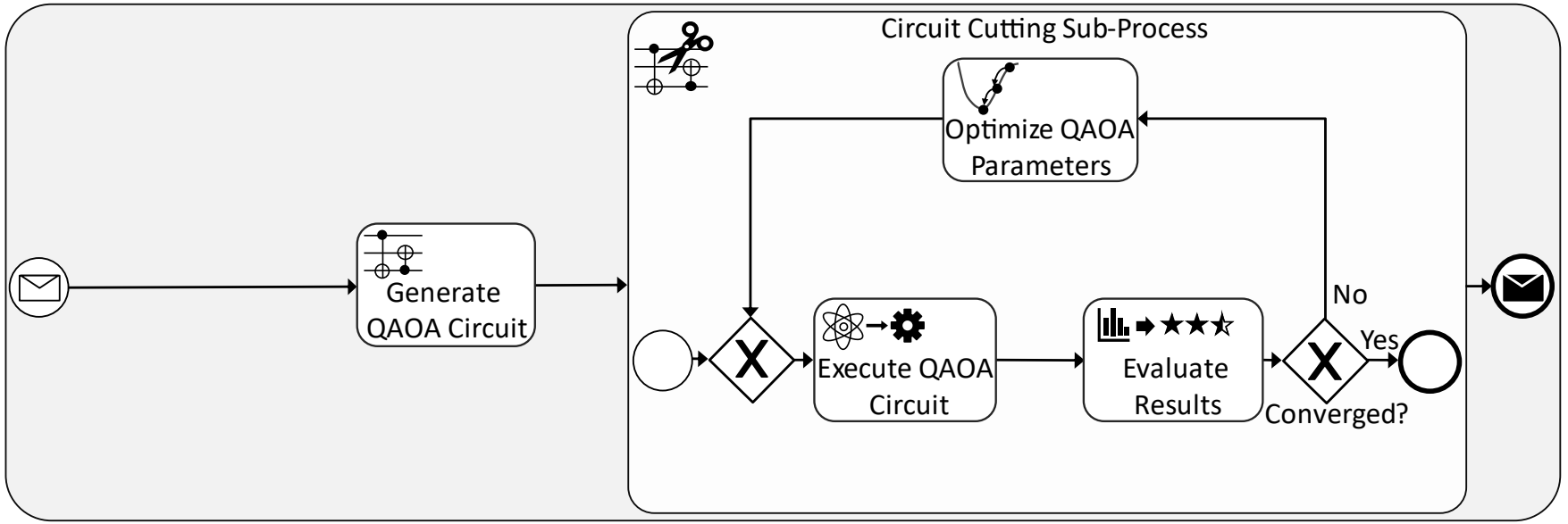


QuantME Replacment Models (QRMs)

- Defining QuantME tasks to replace together with replacing workflow fragments
- Exemplary QRM:

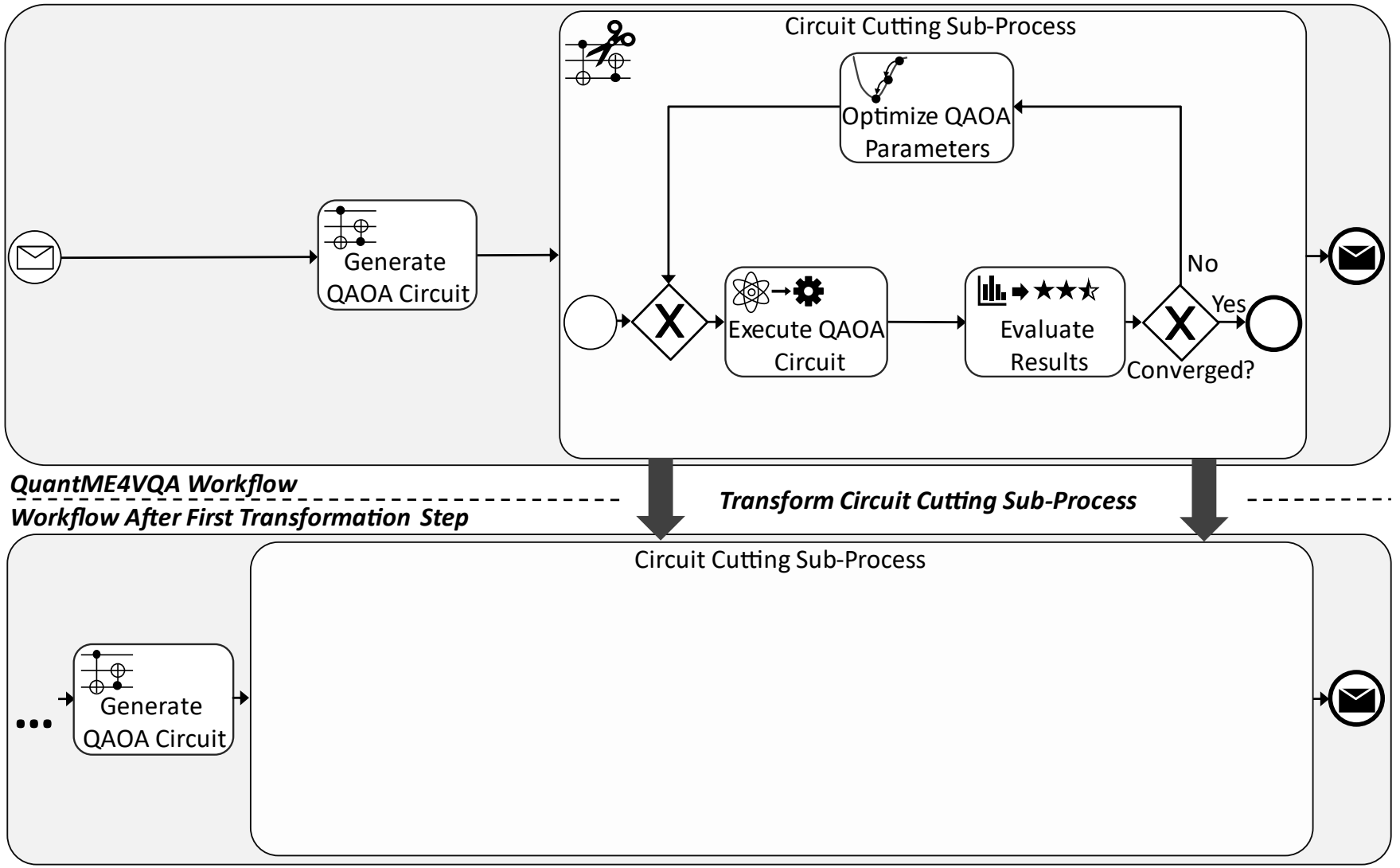


Custom Transformation Rules

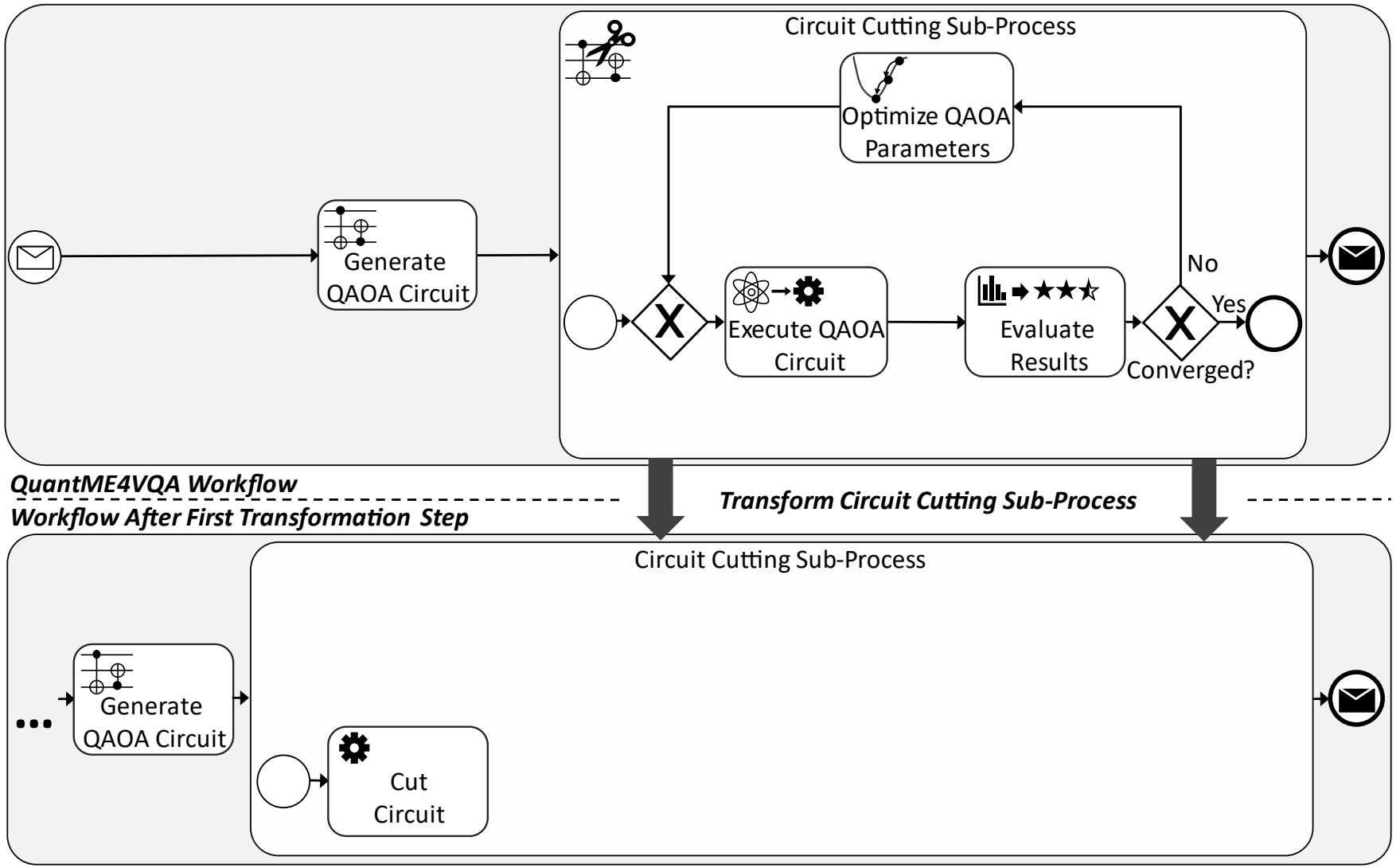


QuantME4VQA Workflow

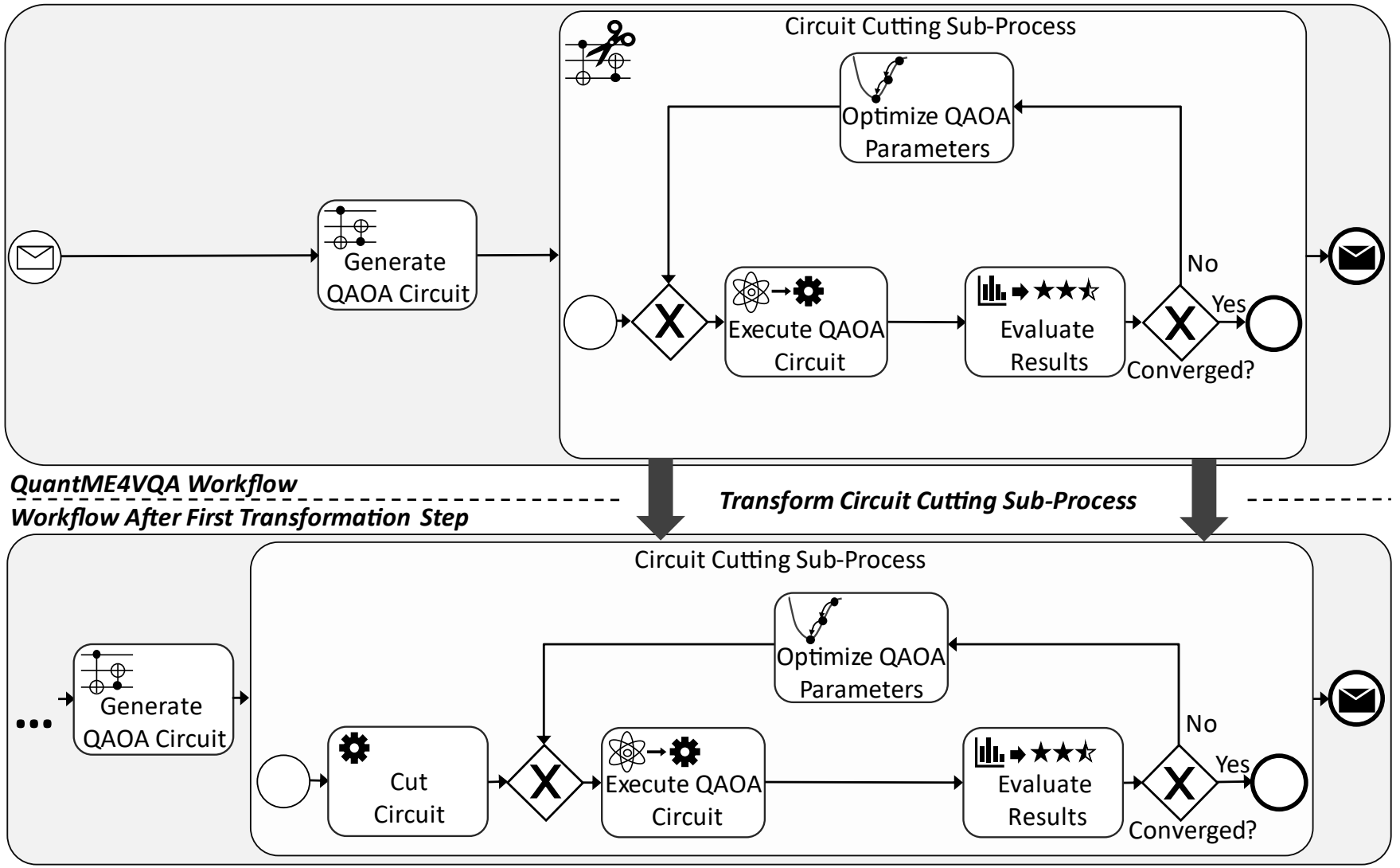
Custom Transformation Rules



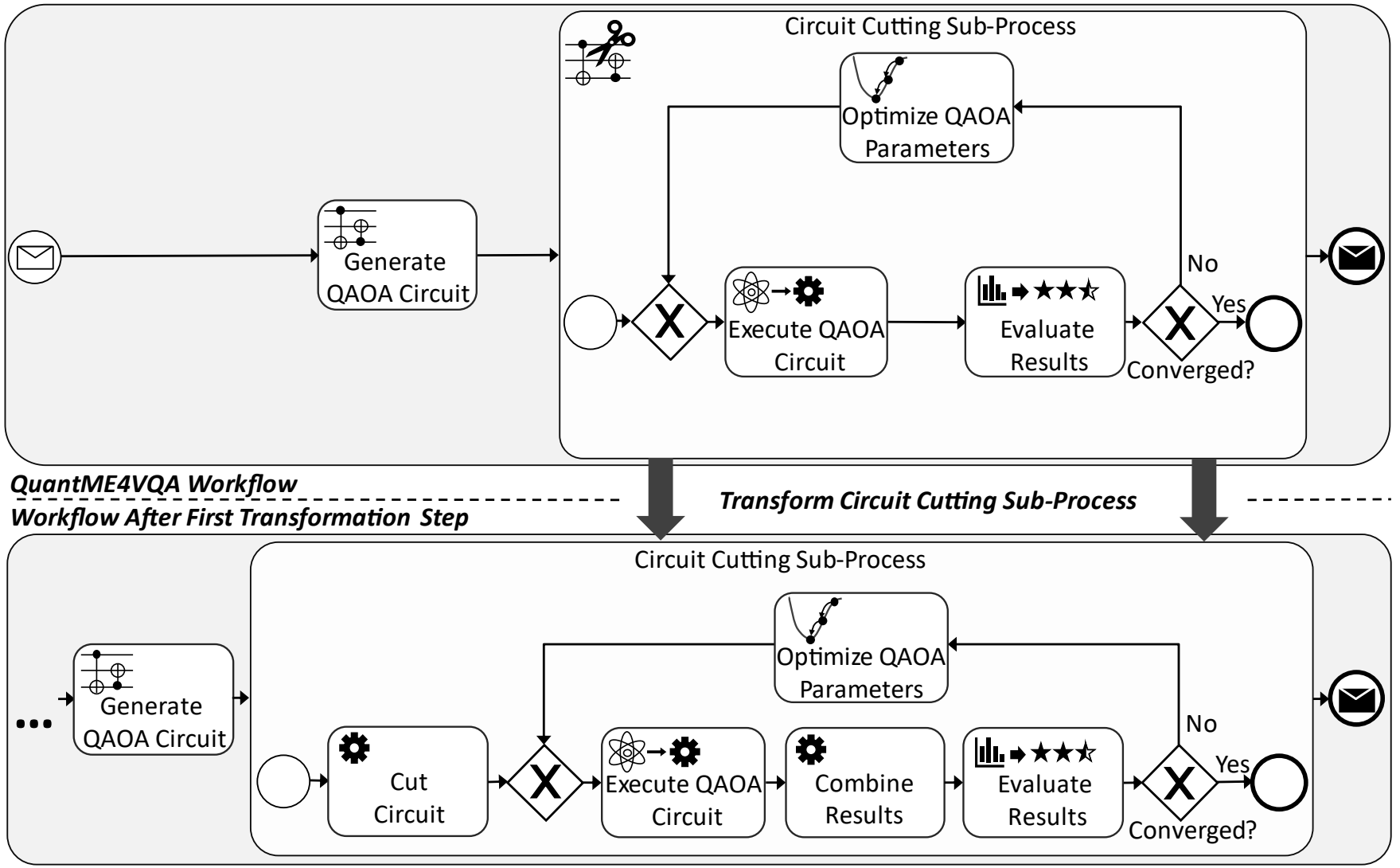
Custom Transformation Rules



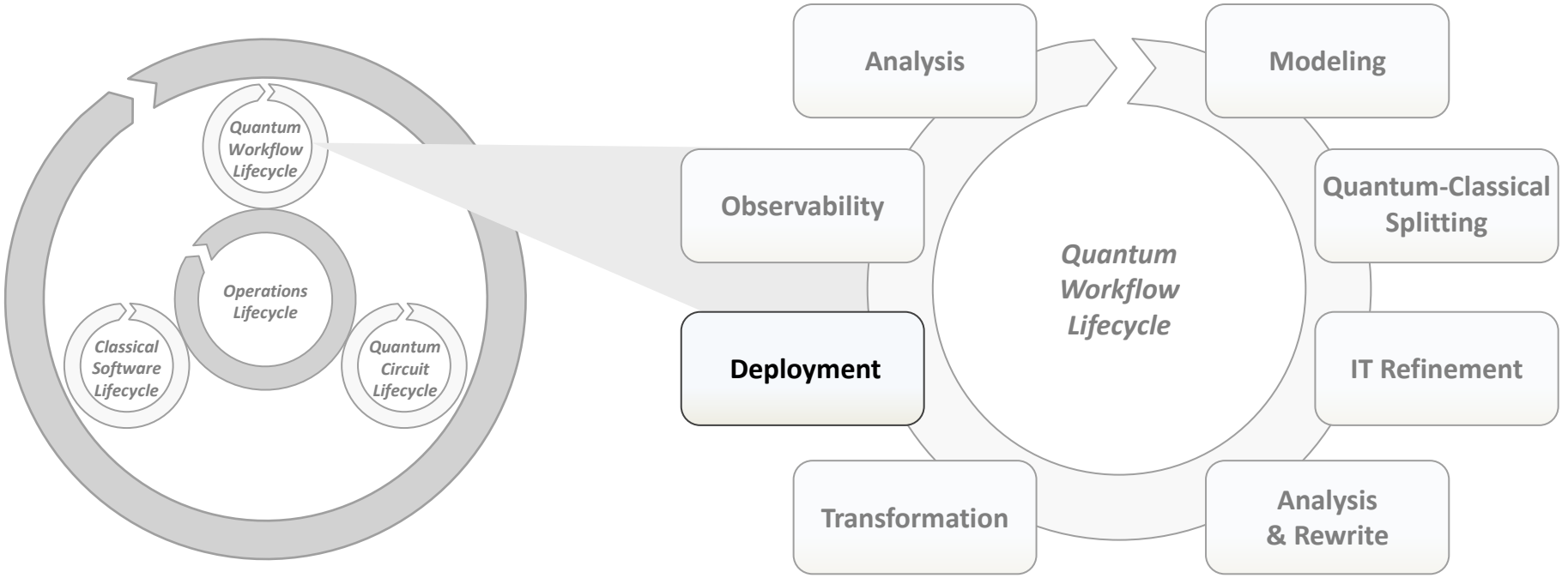
Custom Transformation Rules



Custom Transformation Rules

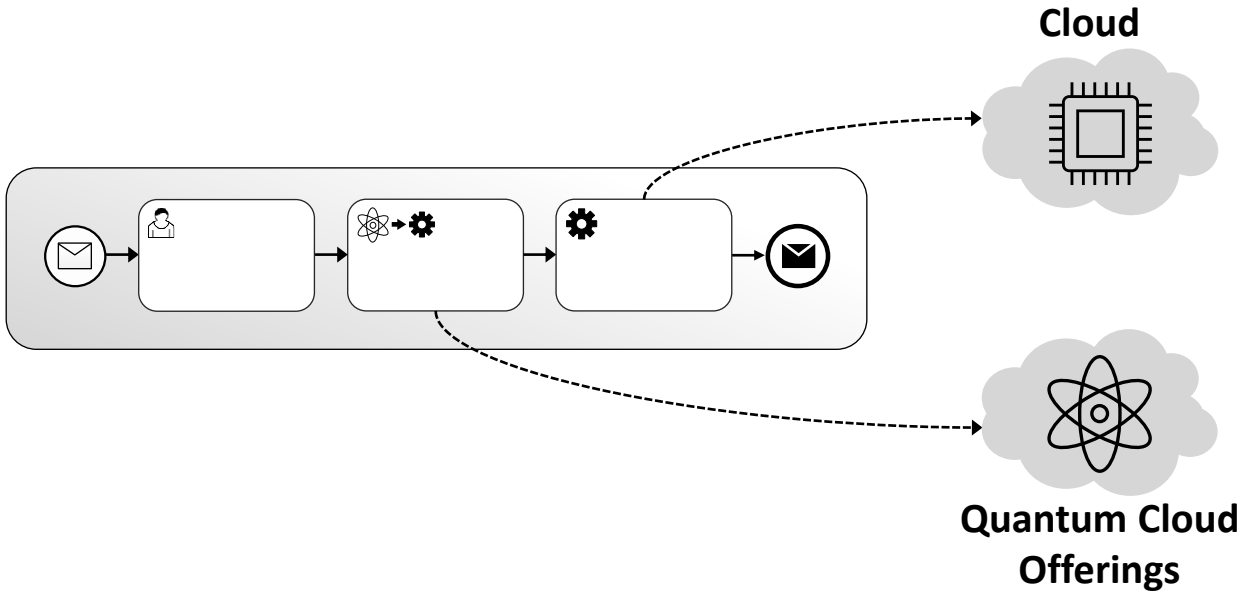


Detailed View of the Quantum Workflow Lifecycle



Invocation of Quantum and Classical Programs

- Workflow invokes quantum and classical programs during runtime:

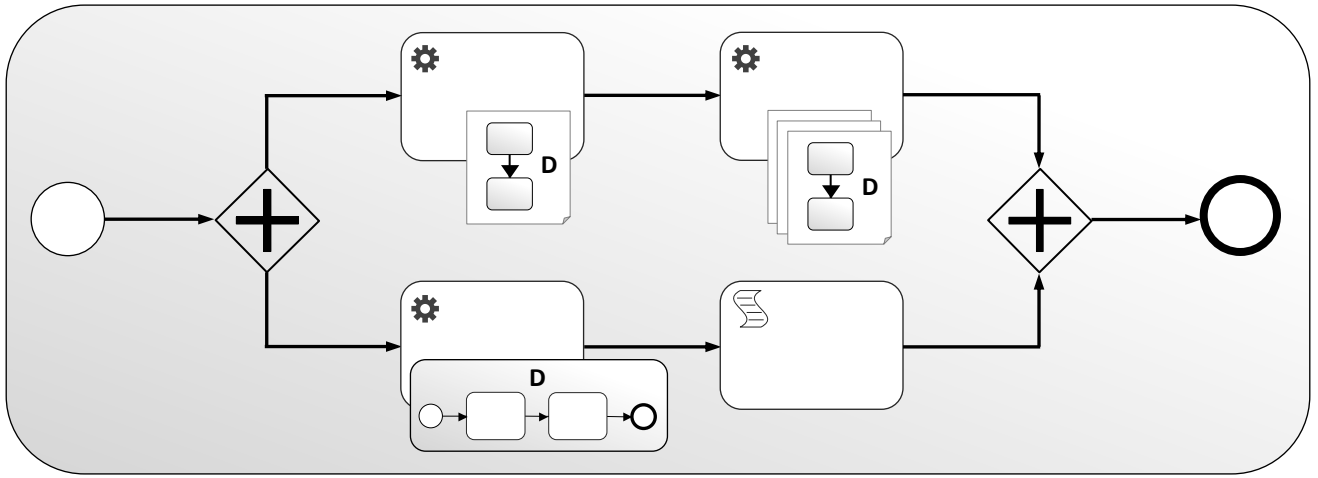


- Required programs and service are often not “always on”:
  - Must be deployed before using them in a workflow
  - Error-prone, time-consuming deployment

➔ Automation using deployment technologies

Self-Contained Workflow Models

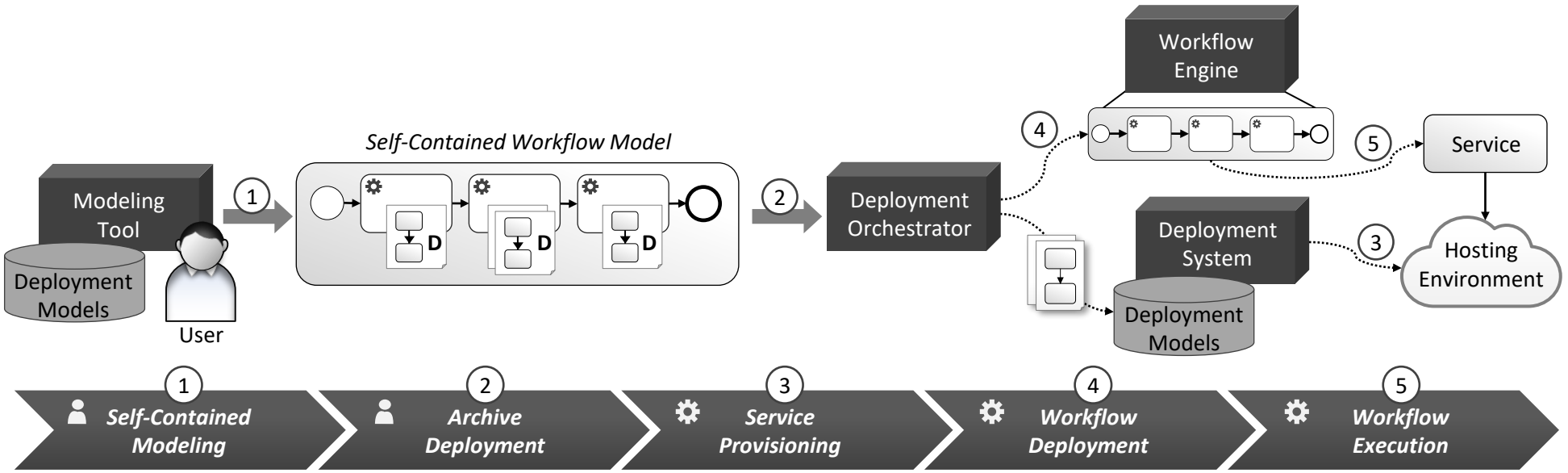
- Service deployment models are attached to activities:



- Self-contained archive can be transferred into the target environment

➔ However, deployment and binding must still be done by the user

Automatic Service Deployment



Conclusion & Outlook

Conclusion & Outlook

- Today's quantum applications are mostly hybrid and require to orchestrate various tasks
- Workflows are a means for this orchestration, providing robustness, scalability, …
  ➔ Quantum Modeling Extension (QuantME)
- Rewriting of quantum workflows required to achieve higher efficiency
- Transformation ensures portability of workflow models
- Outlook:
  - Service-based development ensures reuse and separation of concerns
  - Tooling support to model, rewrite, transform, and deploy quantum workflows required

Thank you for your attention ☺