Intro to the eduQKD Software Stack

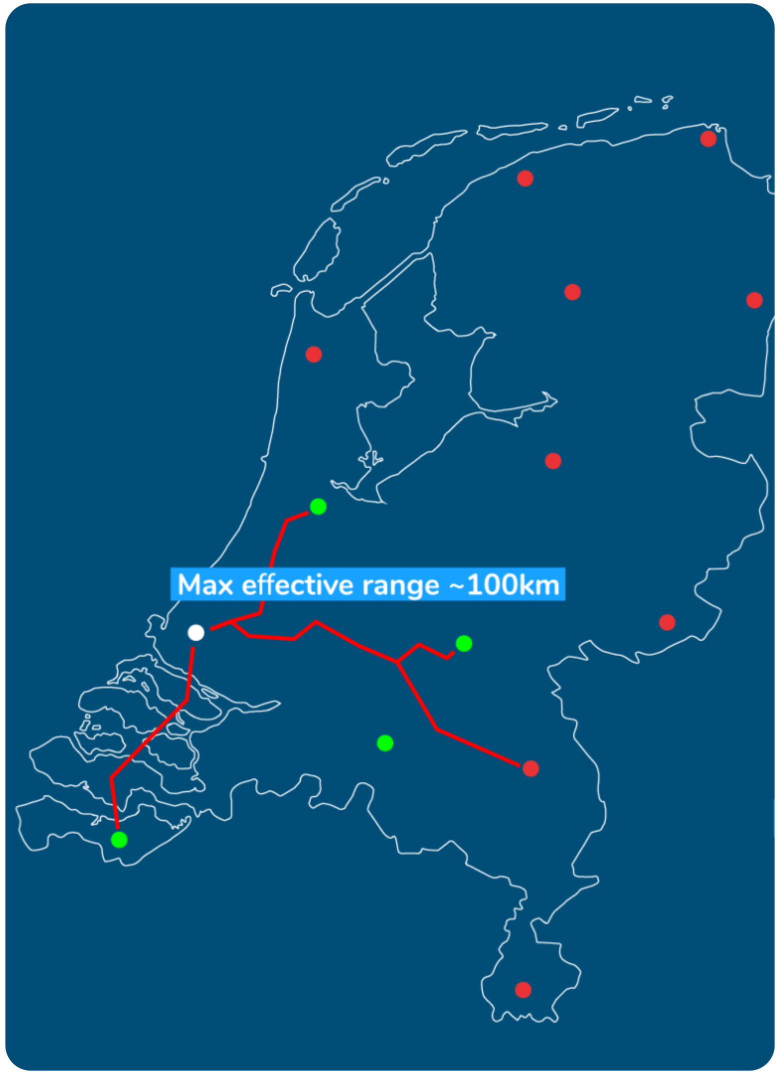
David Maier, SURF NL

david.maier@surf.nl

Overview

Motivation: QCIned 2022

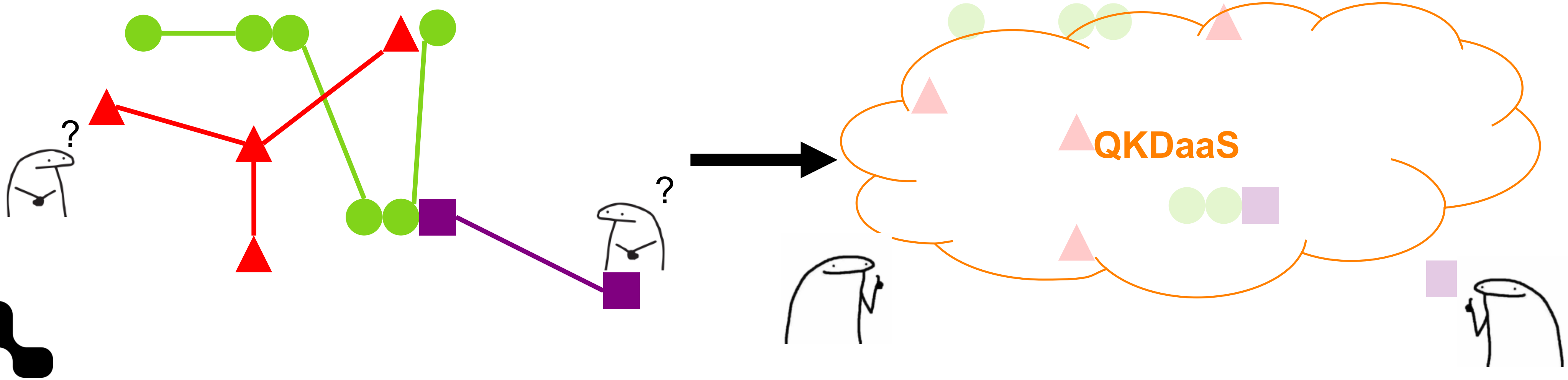
- Software (KMS, QKM) either proprietary, bad or both (back then)
- Vendors not interoperable
- SURF prefers decentralised approach to domain control
- Range limitation of QKD



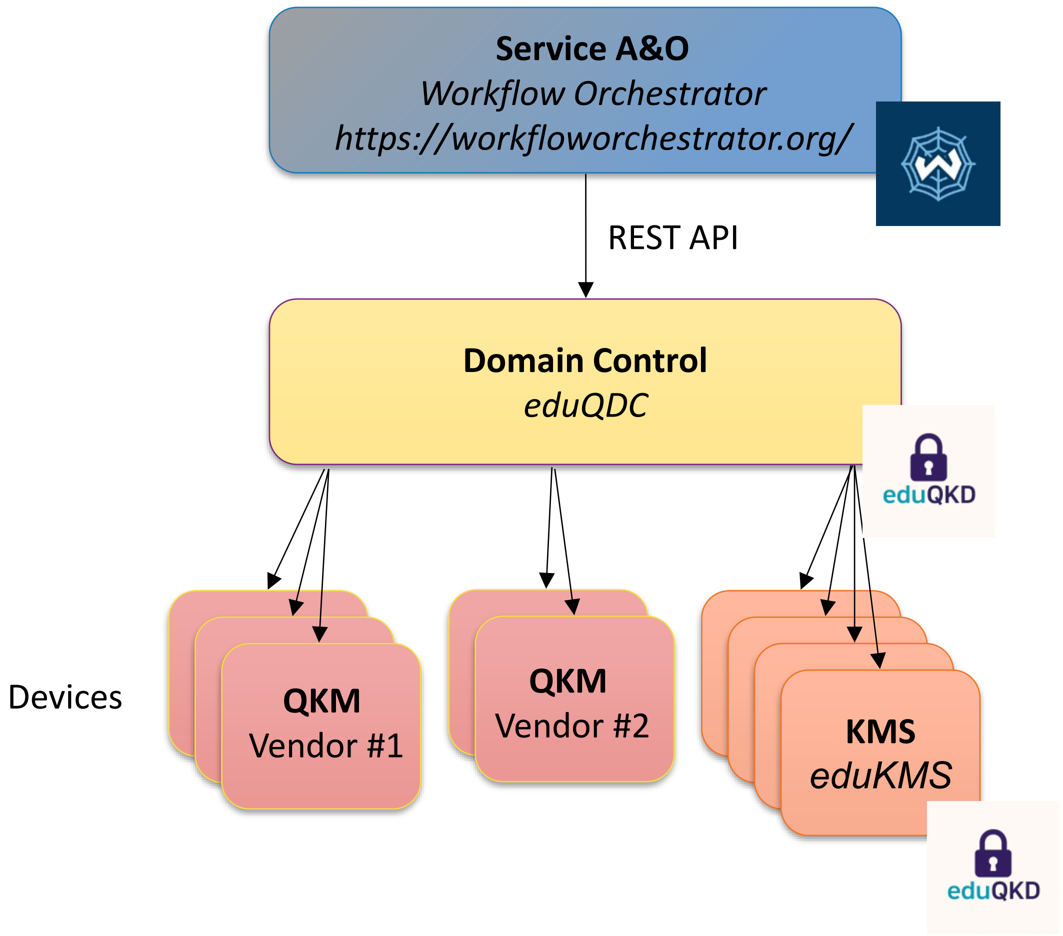
eduQKD: Goal

Develop open-source software stack that is…

- **open-source**, to encourage **collaboration** among QKD deployements
- **ETSI compatible**, to comply with existing **standards**
- Capable of **key-relaying/forwarding**, to enable **practical** deployments
- **vendor-agnostic**, to **avoid lock-in**
- **decentralized**, to **avoid single point of failure**



eduQKD: Layered Architecture

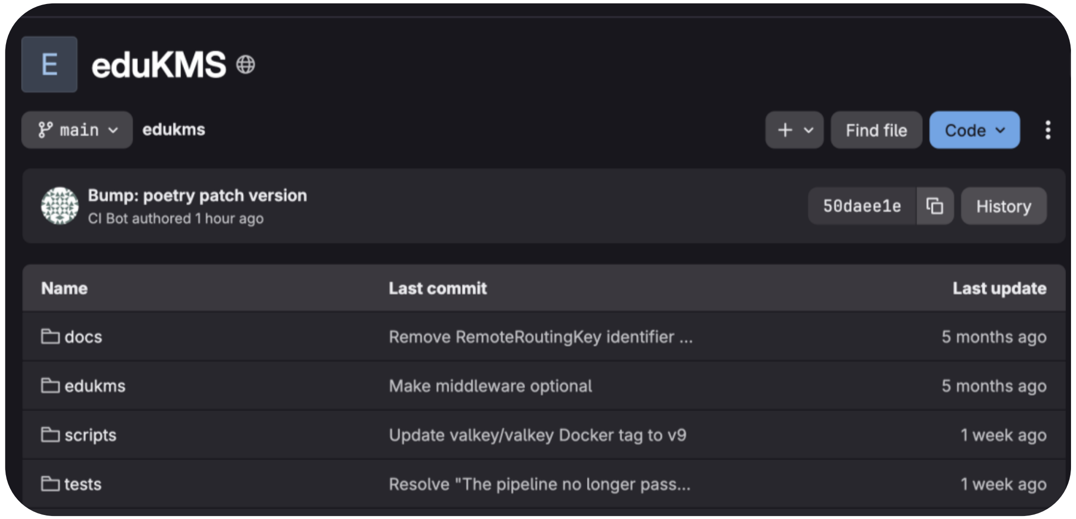


Workflow Orchestrator:
(open source)
Create, terminate, modify, validate subscriptions

Quantum Domain Controller:
set configuration of devices in network

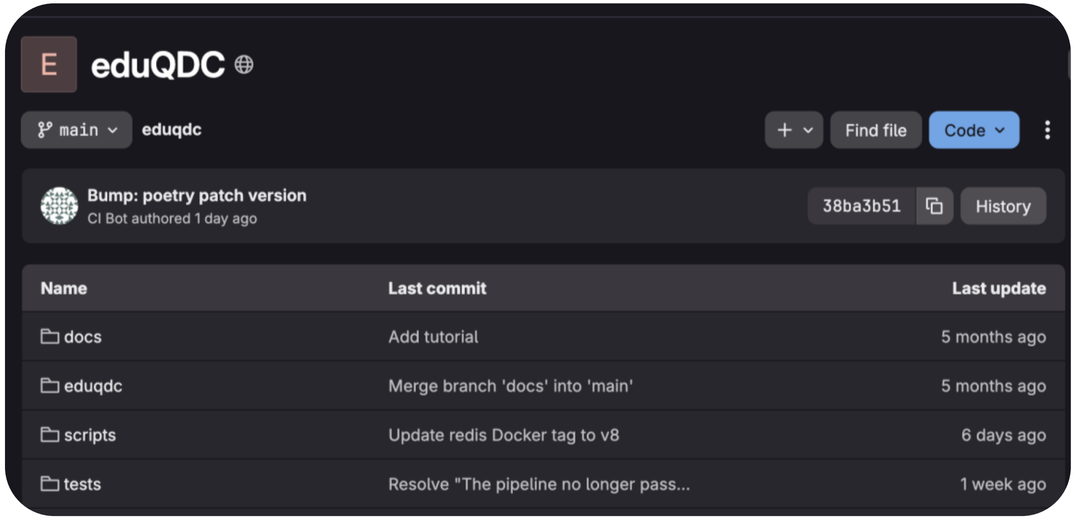
Physical Layer:
QKD hardware:
Point-to-point key generation
Key Management Service:
Key Relaying, Routing, etc.

gitlab.com/surfquantum/eduqkd



Bump: poetry patch version
CI Bot authored 1 hour ago
50daee1e

Name	Last commit	Last update
docs	Remove RemoteRoutingKey identifier ...	5 months ago
edukms	Make middleware optional	5 months ago
scripts	Update valkey/valkey Docker tag to v9	1 week ago
tests	Resolve "The pipeline no longer pass...	1 week ago



Bump: poetry patch version
CI Bot authored 1 day ago
38ba3b51

Name	Last commit	Last update
docs	Add tutorial	5 months ago
eduqdc	Merge branch 'docs' into 'main'	5 months ago
scripts	Update redis Docker tag to v8	6 days ago
tests	Resolve "The pipeline no longer pass...	1 week ago

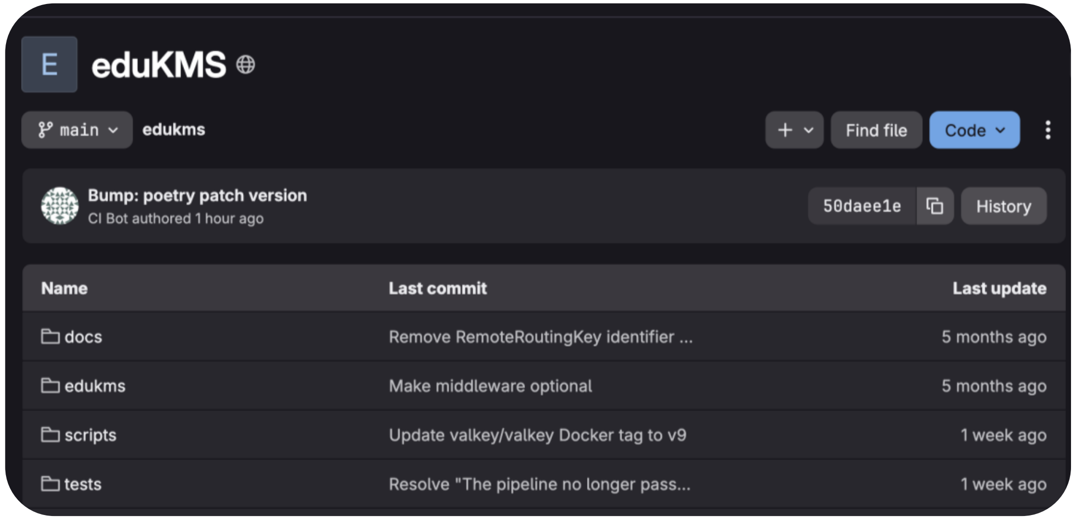
eduKMS

- Implements key relaying and “key flooding”
- Fully vendor-interoperable
- Operates decentralised and independent

eduQDC

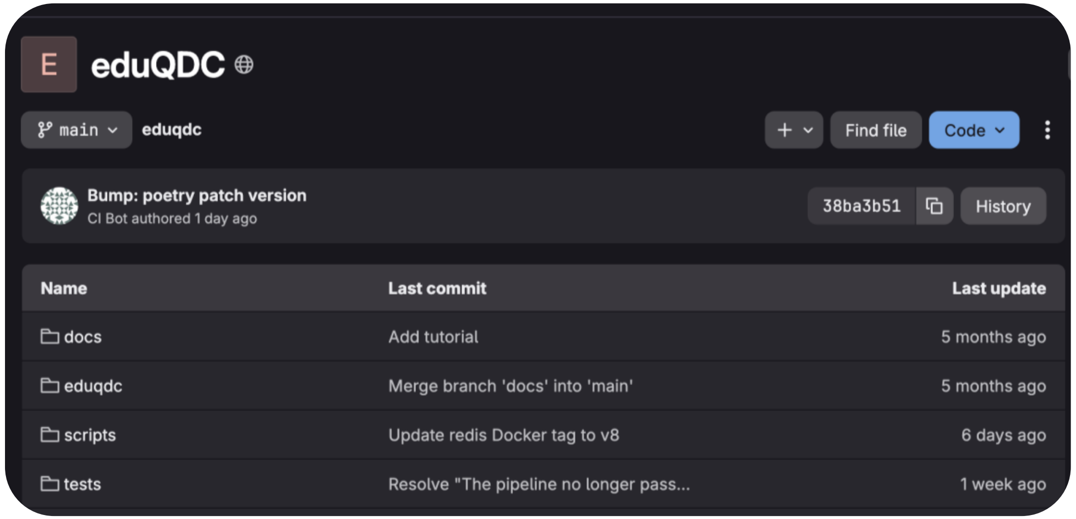
- Defines interface towards WFO
- Deploys KMS
- Pushes configuration
- Optional, stand-alone

gitlab.com/surfquantum/eduqkd



Bump: poetry patch version
CI Bot authored 1 hour ago

Name	Last commit	Last update
docs	Remove RemoteRoutingKey identifier ...	5 months ago
edukms	Make middleware optional	5 months ago
scripts	Update valkey/valkey Docker tag to v9	1 week ago
tests	Resolve "The pipeline no longer pass...	1 week ago



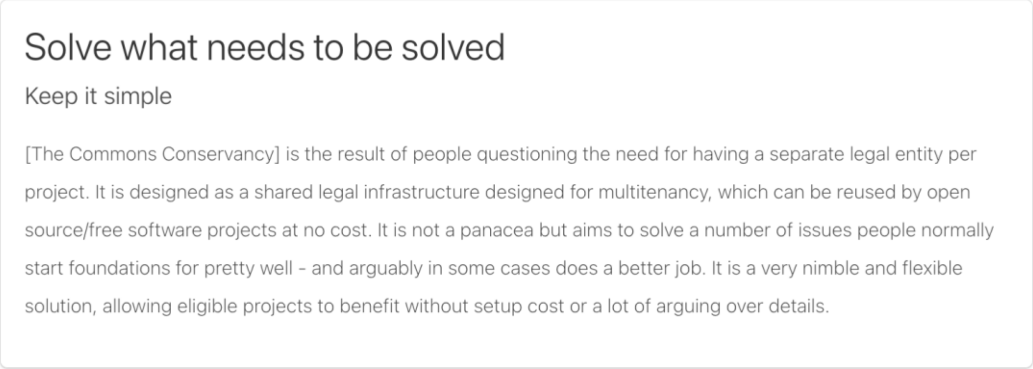
Bump: poetry patch version
CI Bot authored 1 day ago

Name	Last commit	Last update
docs	Add tutorial	5 months ago
eduqdc	Merge branch 'docs' into 'main'	5 months ago
scripts	Update redis Docker tag to v8	6 days ago
tests	Resolve "The pipeline no longer pass...	1 week ago

- **Development will continue** in follow-up projects (e.g. CEF call)
- Interest in **finding co-developers** to ensure sustainability going forward
- Open to **on-boarding of new developers,** in particular if somebody wants to contribute

eduQKD: Governance

- **Currently** (for exclusively practical reasons): SURF
- If interest in **co-development:**
  open to **move to a more open and neutral governance model**
  (e.g., via commons conservancy)



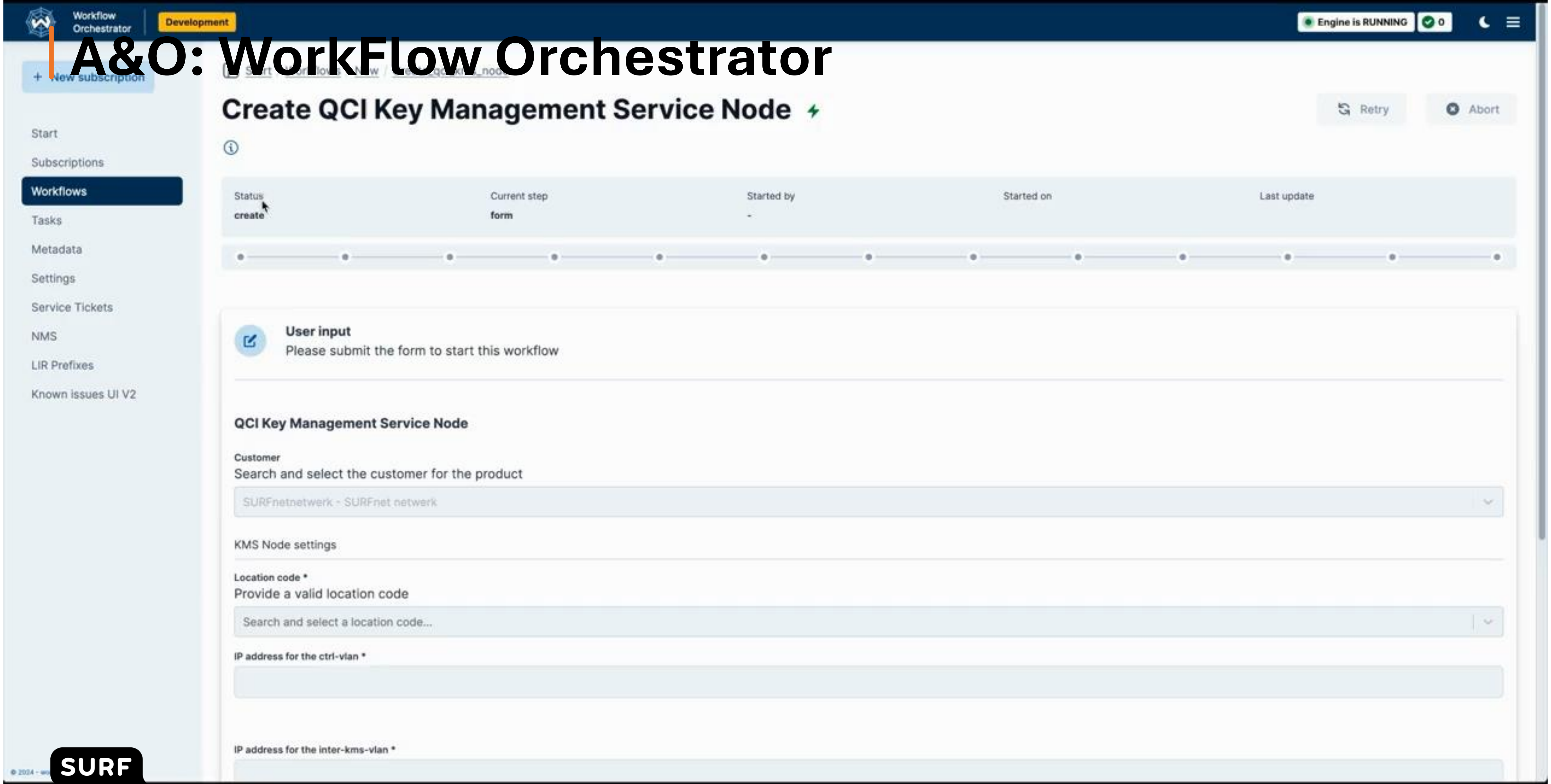
Solve what needs to be solved

Keep it simple

[The Commons Conservancy] is the result of people questioning the need for having a separate legal entity per project. It is designed as a shared legal infrastructure designed for multitenancy, which can be reused by open source/free software projects at no cost. It is not a panacea but aims to solve a number of issues people normally start foundations for pretty well - and arguably in some cases does a better job. It is a very nimble and flexible solution, allowing eligible projects to benefit without setup cost or a lot of arguing over details.

How eduQKD works

A&O: WorkFlow Orchestrator



Create QCI Key Management Service Node

Status	Current step	Started by	Started on	Last update
create	form	-		

User input
Please submit the form to start this workflow

QCI Key Management Service Node

Customer
Search and select the customer for the product
SURFnetnetwerk - SURFnet netwerk

KMS Node settings

Location code *
Provide a valid location code
Search and select a location code...

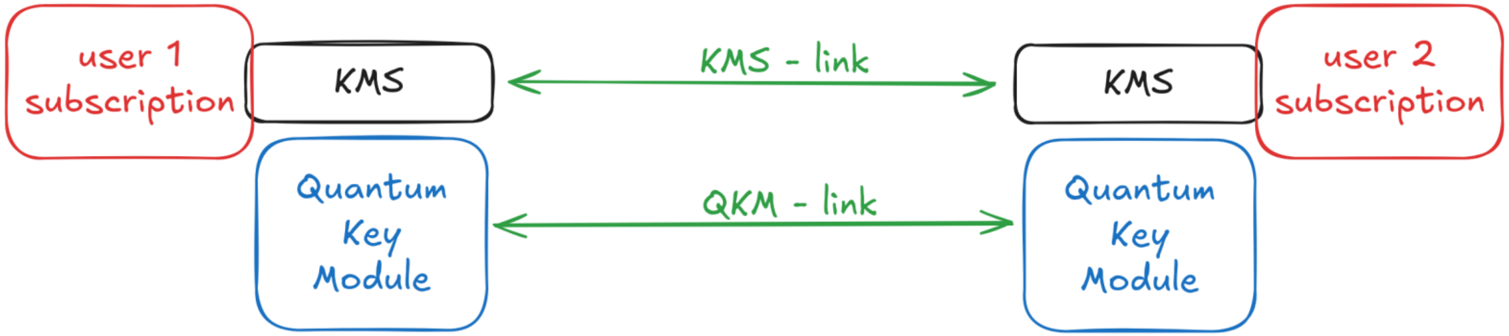
IP address for the ctrl-vlan *

IP address for the inter-kms-vlan *

A&O: Deployment with workflows

Workflows: create/terminate/modify

- kms_node
- kms_link
- qkm
- qkm_link
- qkd_service (user subscription)



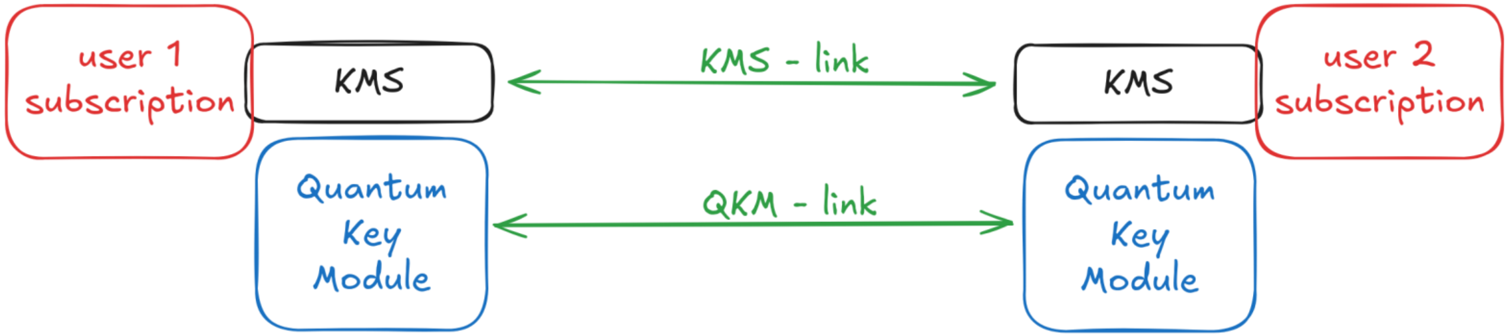
eduQKD Deployment

Prerequisites:

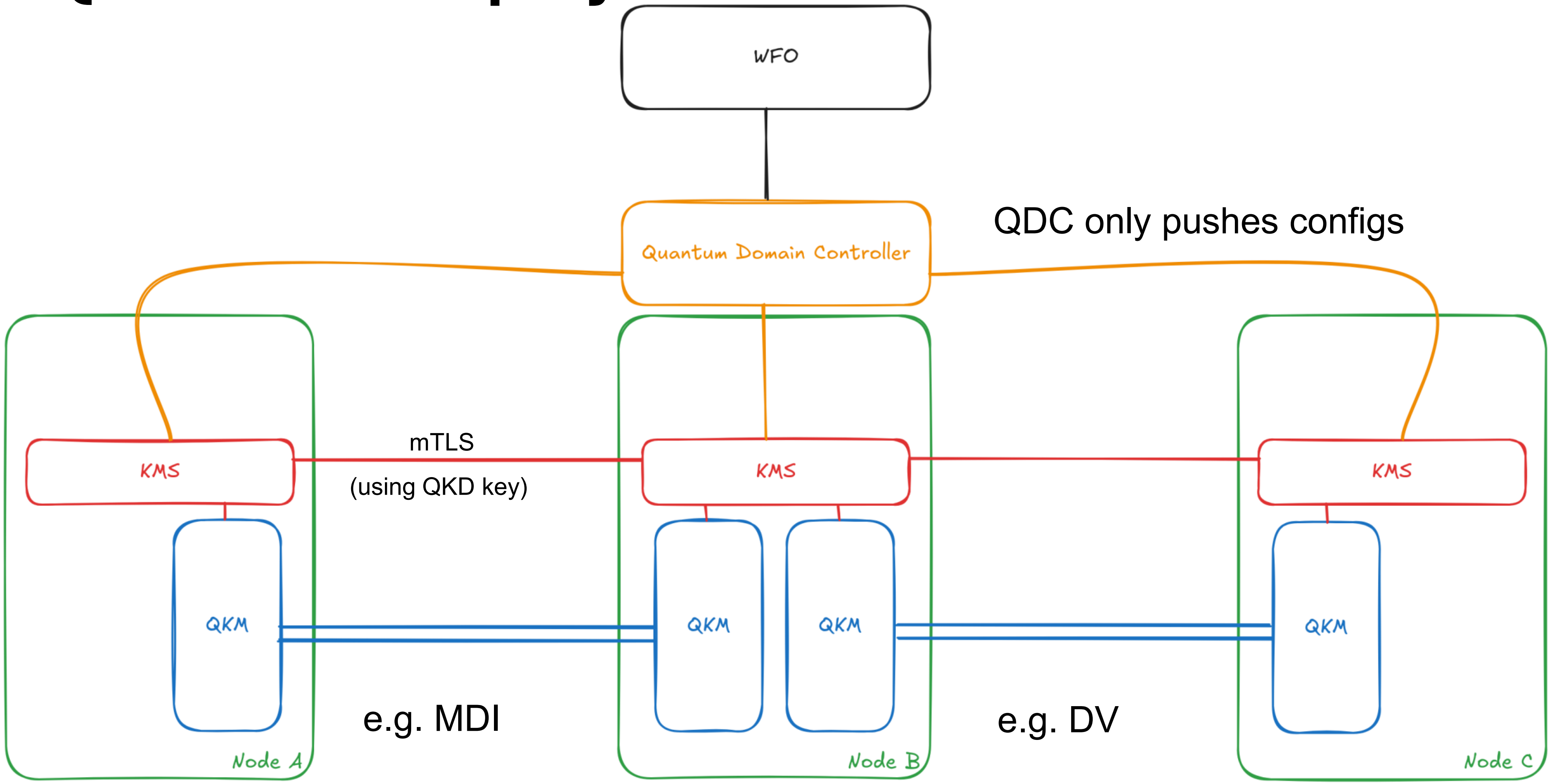
- devices installed with minimal config (need to be reachable remotely)
- classical connectivity (e.g. subnet, l2vpn)
- (optional) WFO deployed

Then:

- Deploy eduQDC on server
- from WFO **execute workflows** to deploy (or directly via QDC):
  KMS -> KMS links -> add QKMs -> QKM links -> add user subscriptions



eduQKD: basic deployment



QDC only pushes configs

mTLS

(using QKD key)

e.g. MDI

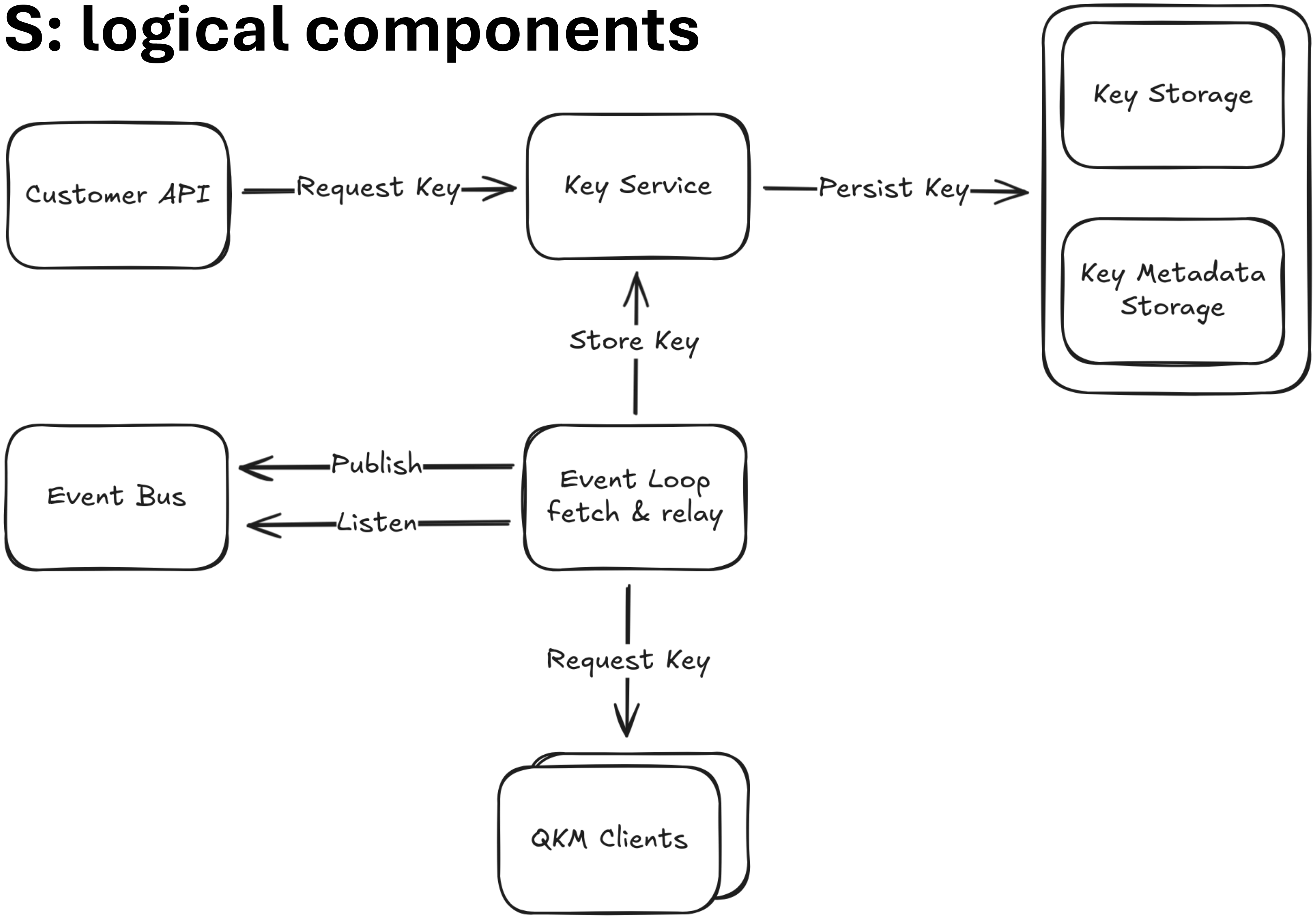
e.g. DV

eduQKD: current limitations

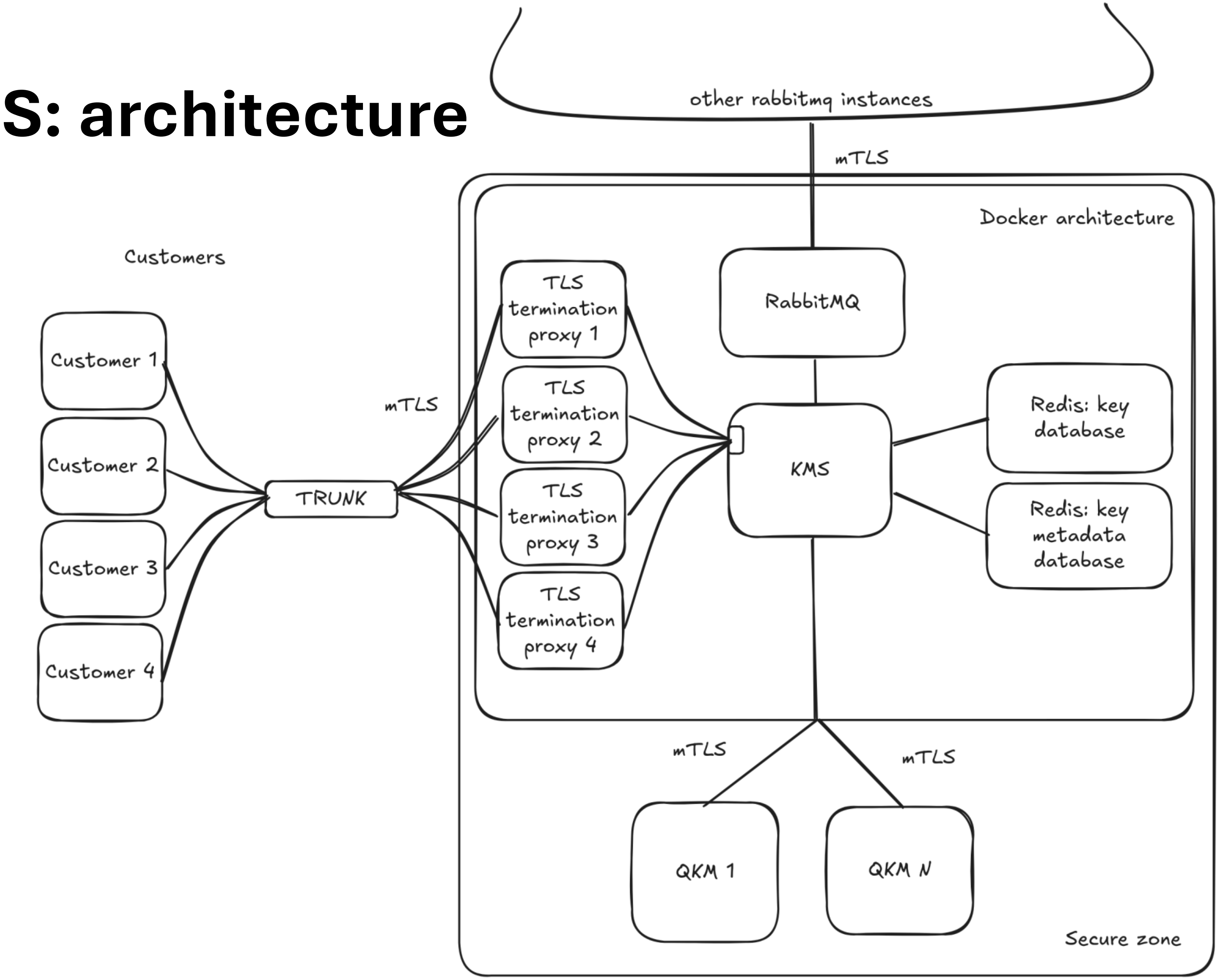
- not pen-tested / authentication still not quantum safe
- static routes
- key flooding very basic (round-robin)
- Few things still hardcoded (e.g. key_size)
- Telemetry/monitoring
- Scalability

eduKMS

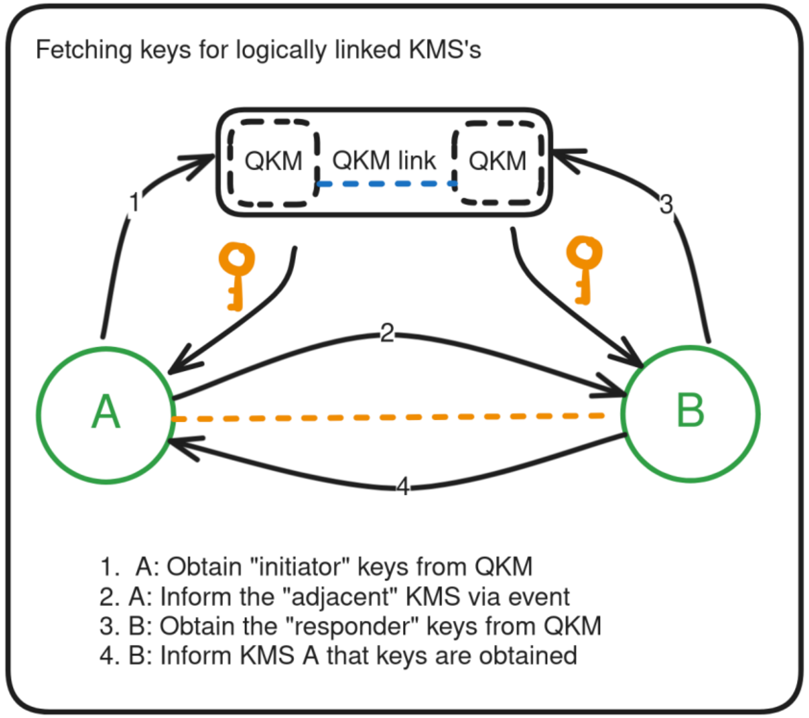
eduKMS: logical components



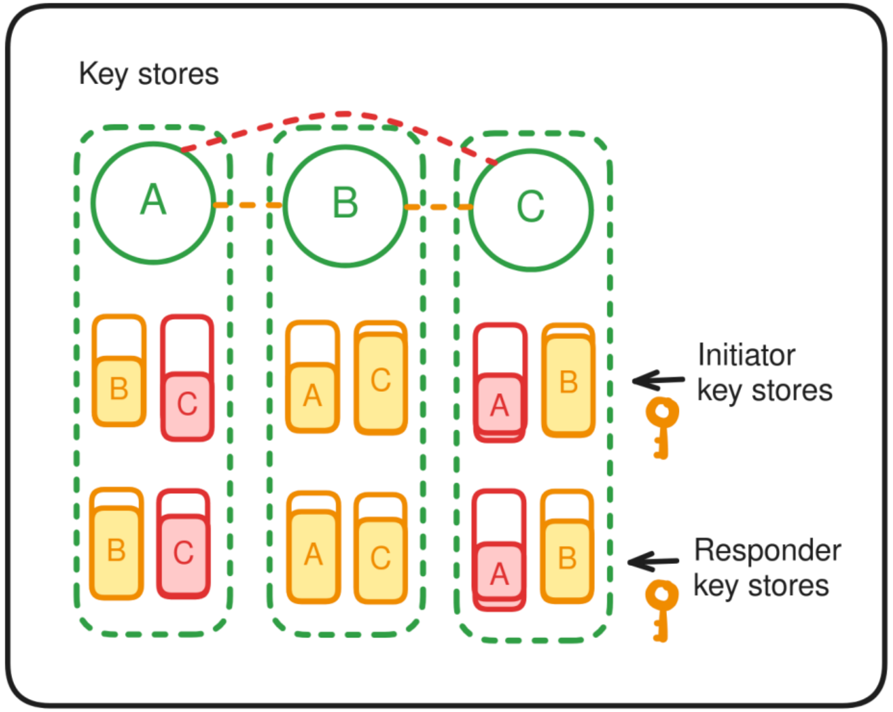
eduKMS: architecture



eduKMS: key-relaying

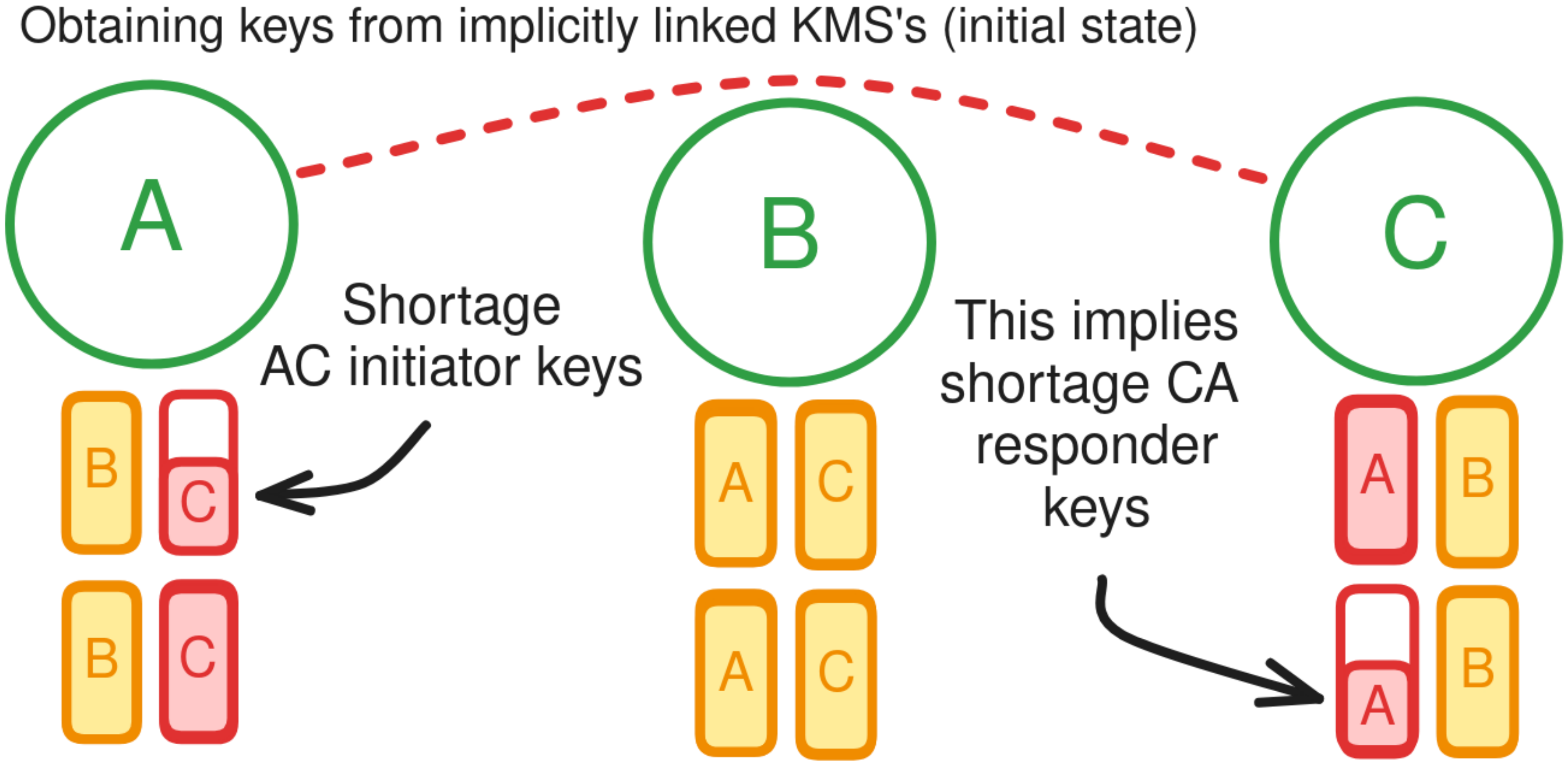


1. A: Obtain "initiator" keys from QKM
2. A: Inform the "adjacent" KMS via event
3. B: Obtain the "responder" keys from QKM
4. B: Inform KMS A that keys are obtained



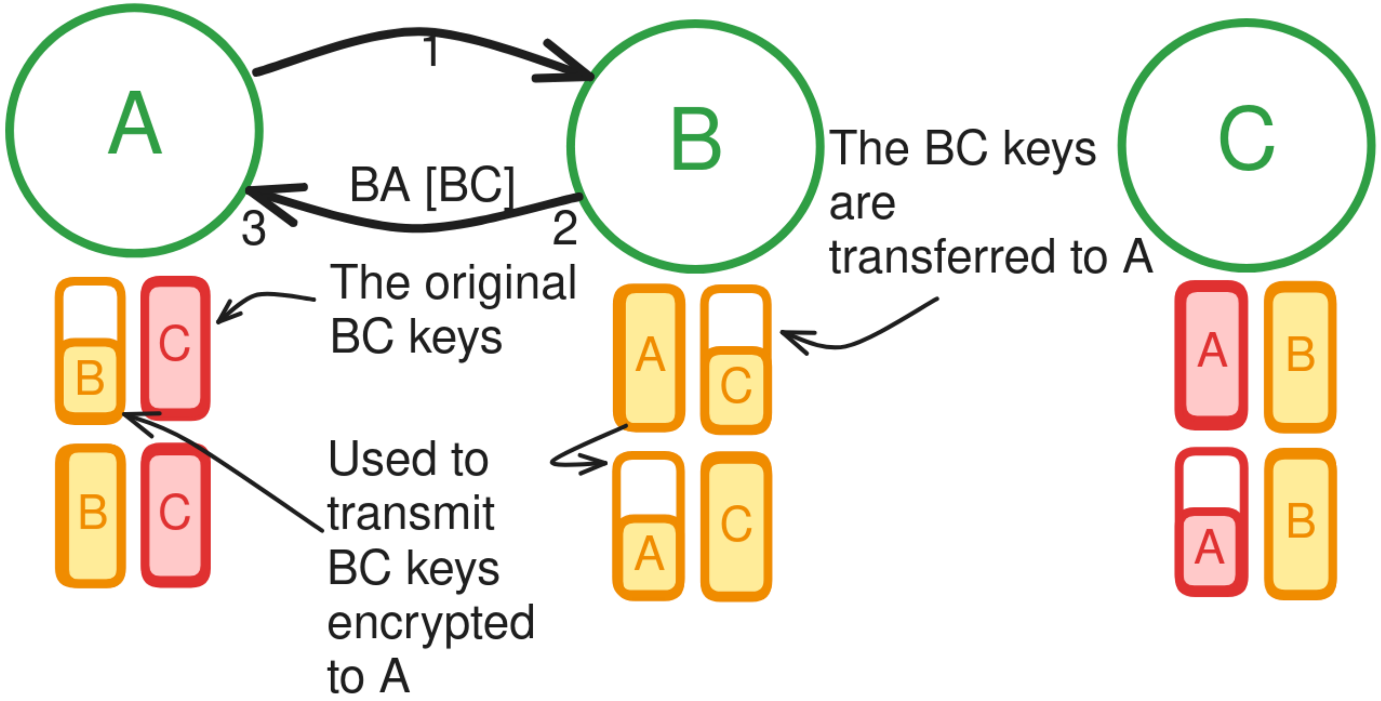
Split into "mono-directional" initiator/responder keys to avoid race conditions

eduKMS: key-relaying



eduKMS: key-relaying

1. A asks B for initiator keys to C
2. B encrypts BC initiator keys with BA responder keys
3. A decrypts these with AB responder keys,
   obtaining the BC keys (that are now AC keys from A's perspective)



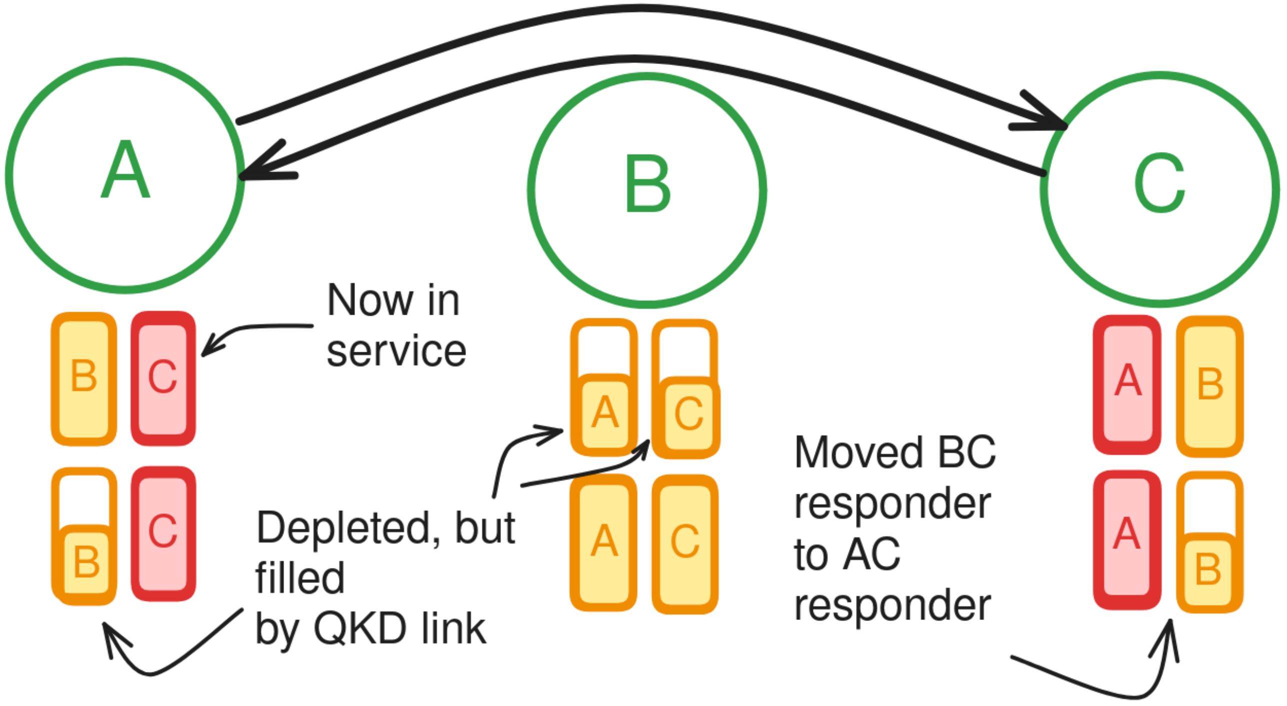
The BC keys are transferred to A

The original BC keys

Used to transmit BC keys encrypted to A

eduKMS: key-relaying

A sends a rename request to C, after confirmation, the keys are "in service"

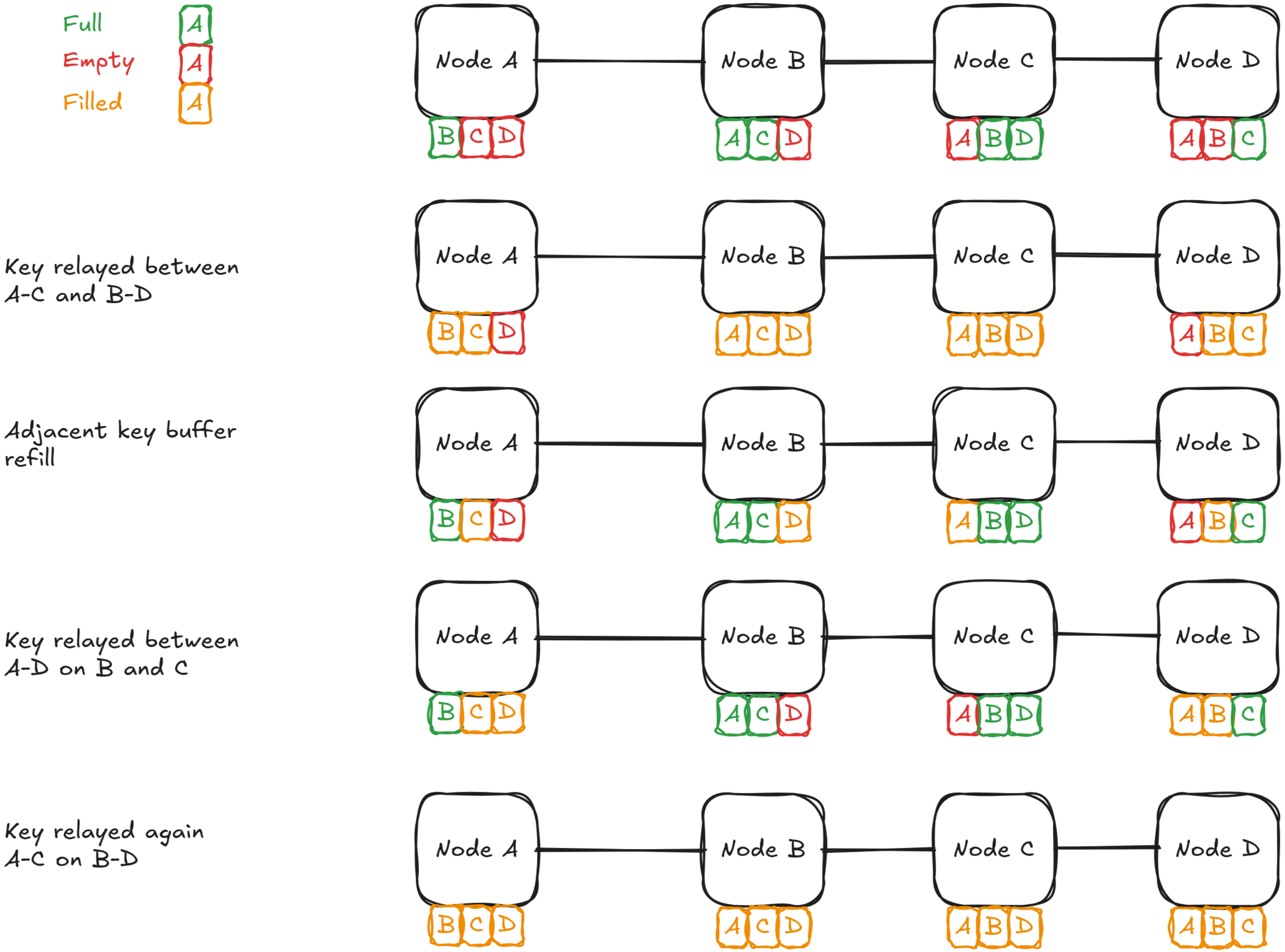


Now in service

Depleted, but filled by QKD link

Moved BC responder to AC responder

eduKMS: key flooding



Full
Empty
Filled

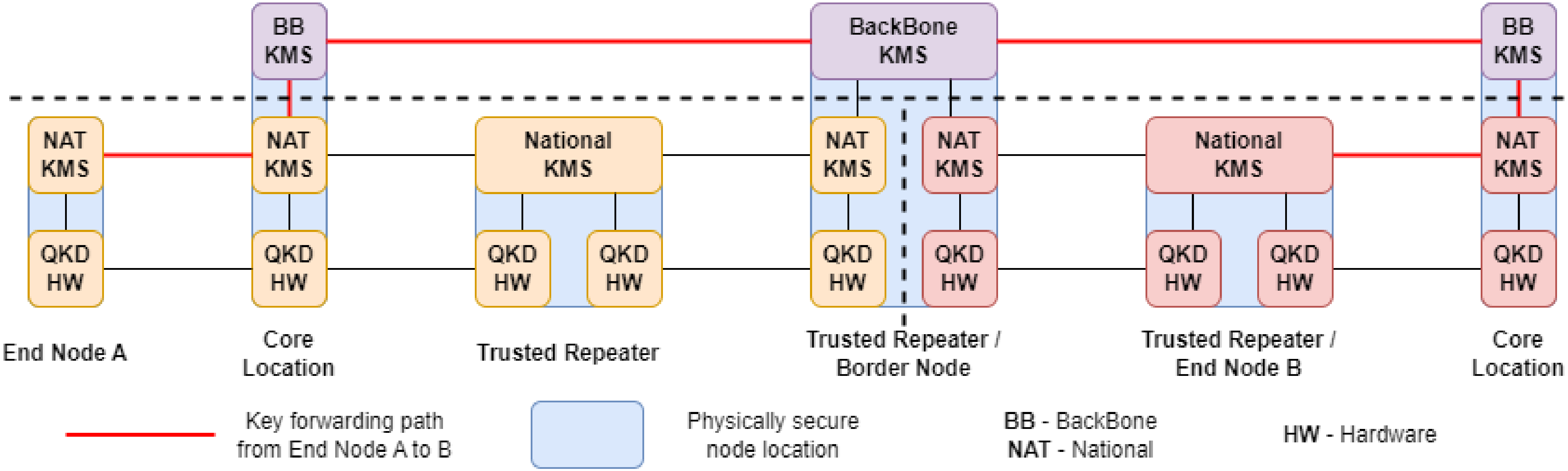
Key relayed between A-C and B-D

Adjacent key buffer refill

Key relayed between A-D on B and C

Key relayed again A-C on B-D

eduQKD across domains: software backbone (draft)



Devlopment Roadmap

- ETSI 020 integration
- Cross-domain software backbone
- Publish monitoring and telemetry code
- Security audit

Intro to the eduQKD Software Stack

David Maier, SURF NL

david.maier@surf.nl