USER-DRIVEN PATH VERIFICATION AND CONTROL FOR INTER-DOMAIN NETWORKS

NWO

UNIVERSITY
OF TWENTE.

UNIVERSITY
OF AMSTERDAM

NLNETLABS

SIDN LABS

SURF

2STIC

UPIN

WORKSHOP #2

RODRIGO, LEONARDO, PAOLA, AIKO, CRISTIAN

JULY 1, 2021

This research received funding from the Dutch Research Council (NWO) as part of the UPIN project

TODAY'S GOAL

- Recap of UPIN
- Update on status and future work
- Get your feedback

- Result: further improve researchers' work based on your feedback

PROPOSED AGENDA

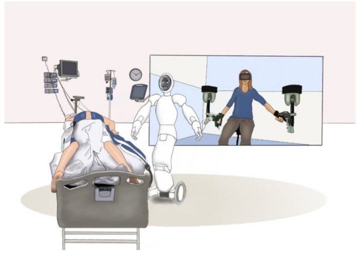
10:00	Opening (Cristian)
10:00	Recap UPIN (Cristian)
10:10	Overall status (Cristian)
10:20	Progress path control (Leonardo)
10:40	Progress path discovery and verification (Rodrigo)
11:00	Discussion (All)
11:30	Partner presentation (optional)
12:00	Adjourn (Cristian)

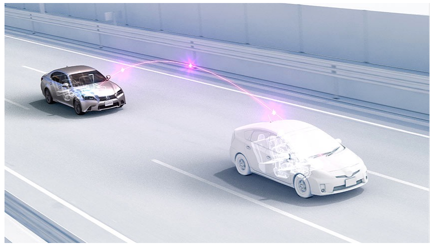
PROBLEM: DATA AUTONOMY “IN TRANSIT”

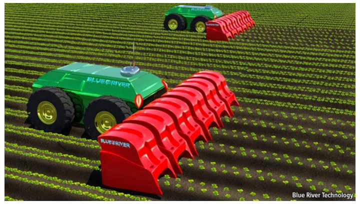
- Lack of **transparency** and **control** of how users’ data flows travel across the Internet
- Which network operators handle my data? How secure are their routers? I only want to use security-audited networks!
- Security risks for **critical services** like remote controlled healthcare robots, energy grids, intelligent transport systems

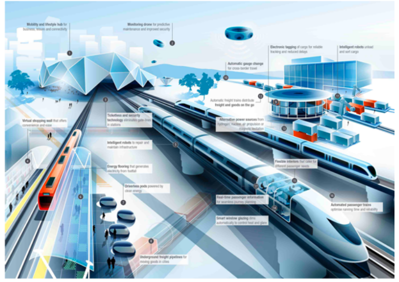
Reduced trust in the Internet infrastructure

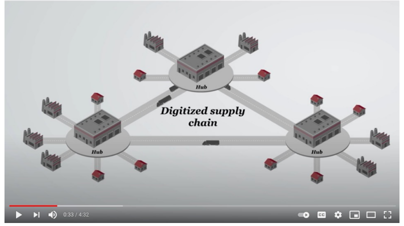
USE CASES: CRITICAL SERVICES

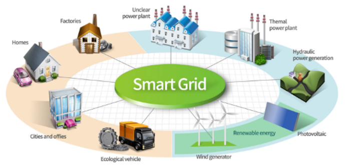












UPIN focus: health, IoT, Intelligent Transport Systems

UPIN GOAL

Provide the building blocks that enable users (e.g., individuals and organizations) to control and verify how their data travels through the Internet or other types of large-scale inter-domain networks, both in terms of hops as well as routers traversed

↓

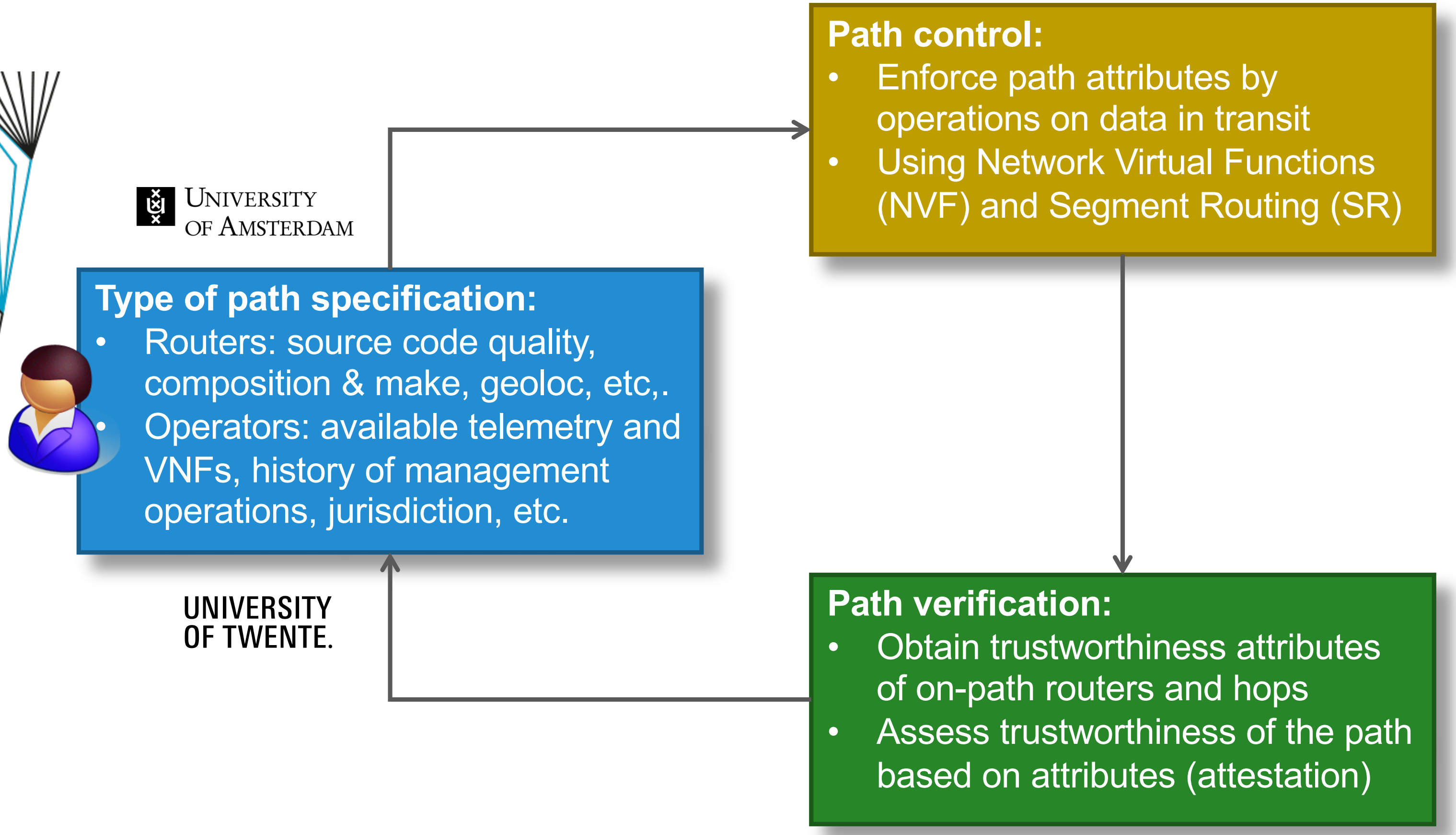
Increase data autonomy in transit

RESEARCH QUESTIONS

- Which mechanisms do we need to make the Internet more transparent and provide Internet users with more control over and verifiability of network paths in a scalable way?

- To what degree can the current Internet architecture accommodate these functions and which other emerging inter-network architectures might potentially be more suitable?

HIGH LEVEL APPROACH



Type of path specification:

- Routers: source code quality, composition & make, geoloc, etc,.
- Operators: available telemetry and VNFs, history of management operations, jurisdiction, etc.

Path control:

- Enforce path attributes by operations on data in transit
- Using Network Virtual Functions (NVF) and Segment Routing (SR)

Path verification:

- Obtain trustworthiness attributes of on-path routers and hops
- Assess trustworthiness of the path based on attributes (attestation)

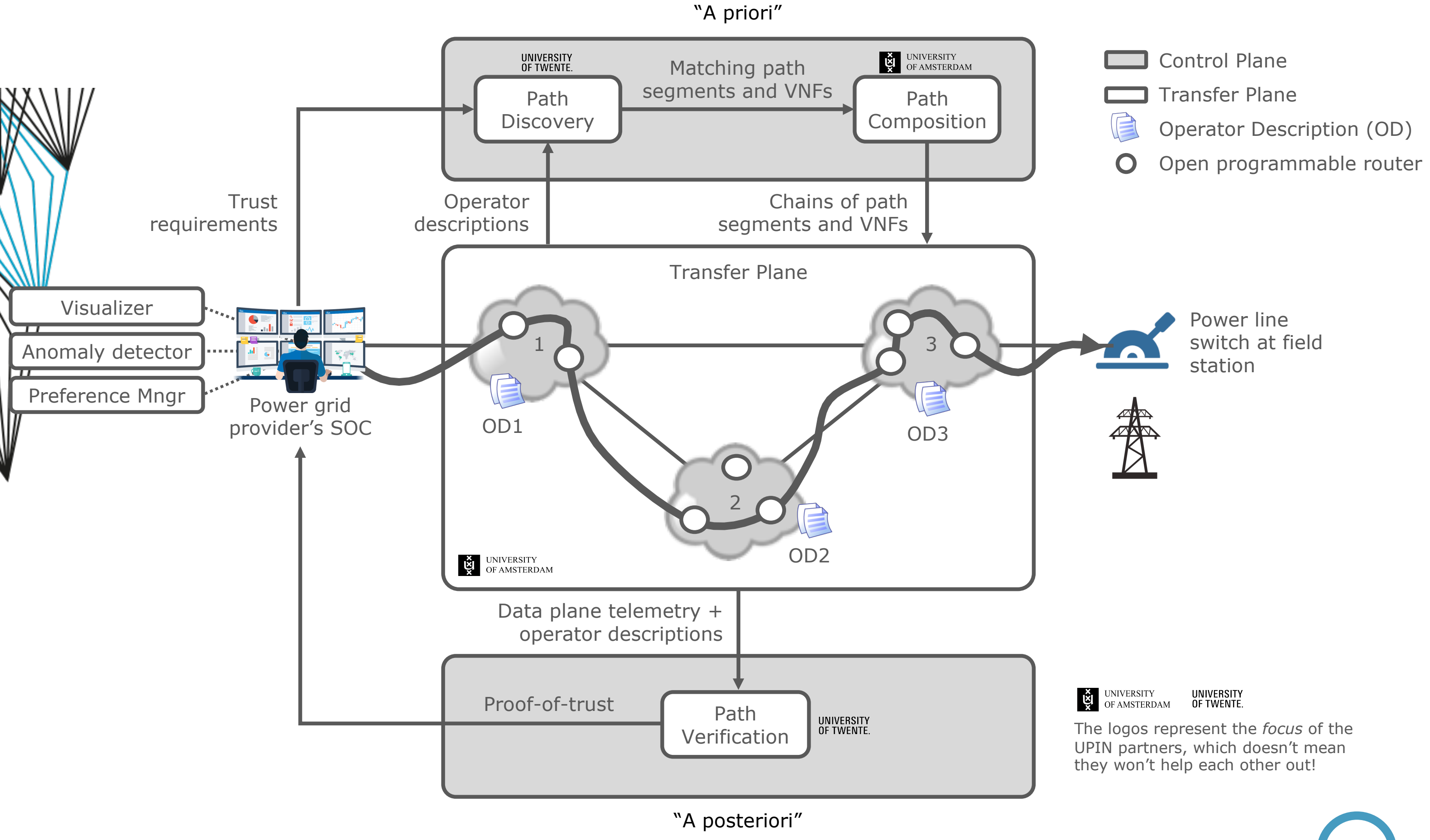
INNOVATIONS

- Novel inter-domain mechanisms for path control and verification based on user’s trust requirement

- New data and control plane protocols that implement these mechanisms using programmable routers and SDN

- Evaluation of the performance and expected scalability of the UPIN system using the 2STiC testbed

UPIN CONCEPT



UNIVERSITY OF AMSTERDAM UNIVERSITY OF TWENTE.

The logos represent the *focus* of the UPIN partners, which doesn't mean they won't help each other out!

KEY RESULTS

- System design and open-source implementation
- Evaluations of through use cases on 2STiC testbed
- Demonstrators of the UPIN concept
- Academic and other publications, annual workshop

TARGETED IMPACT

- Increased **user** control over data in transit
- Enable new types of network and service **operators**
- Advance emerging **standards** (e.g., path-aware networking)
- Increased **pool of knowledge** of academic and operator communities

STATUS

- Poster presentation at ICT.Open (Nov 2020)
- Accepted work-in-progress paper TAURIN workshop (Jun 2021)
- First path control experiments at the UvA
- Website: https://upin-project.nl (work in progress)
- More details in Rodrigo’s and Leonardo’s talks

Contact the UPIN team:

Rodrigo Bazo:	r.bazo@utwente.nl
Leonardo Boldrini:	l.boldrini@uva.nl
Paola Grosso:	p.grosso@uva.nl
Aiko Pras:	a.pras@utwente.nl
Cristian Hesselman:	c.e.w.hesselman@utwente.nl (coordinator)

QUESTIONS AND DISCUSSION

This research received funding from the Dutch Research Council (NWO) as part of the UPIN project

UNIVERSITY
OF TWENTE.

UPIN PROGRESS MEETING

00. SUMMARY

SUMMARY

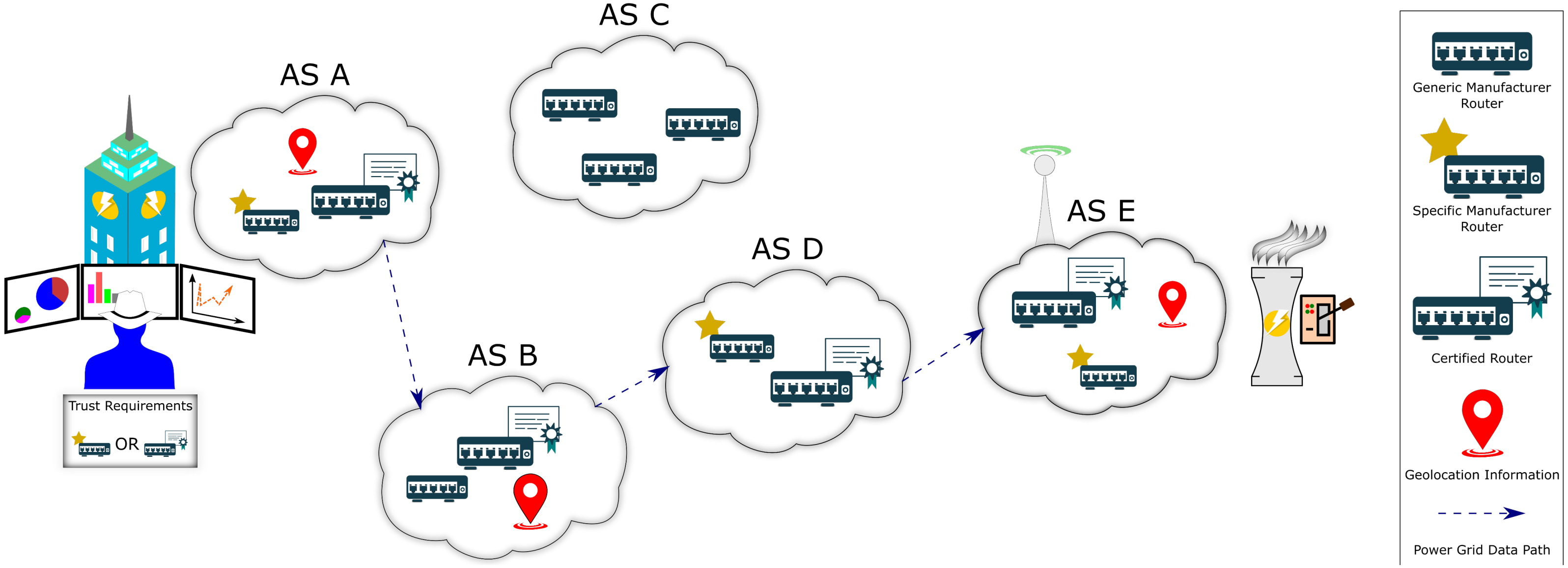
1. Power Grid Use-case
2. Existing Technologies Review
3. UPIN Software Architecture
4. Path Verification Experiments
5. Requirements and Users Surveying

01. POWER GRID USE-CASE

POWER GRID USE CASE

- Decentralized Power Grids will become highly dependant on the security of the network since they will likely depend on multi-domain networks

- Currently, users cannot specify trust requirements such as certified routers or routers from specific manufacturers



POWER GRID USE CASE

- A solution would be for the power grid operators to run their own networks, however this will eventually become unfeasible due to the decentralized nature of the energy grids

- In order to support critical infrastructures such as this power grid, the network must provide higher level of transparency, accountability and controllability to the user
    - Specifically in the multi-domain scenario

02. EXISTING TECHNOLOGIES

EXISTING TECHNOLOGIES

- The requirements observed while analyzing use-cases are not fulfilled by current existing, deployed and production architectures
- On the other hand, a handful of technologies partially solve the problem
- We review the literature for technologies that assess each one of our requirements
  - Transparency
  - Controllability
  - Accountability

EXISTING TECHNOLOGIES

TRANSPARENCY

- There are no solutions that provide:
  - Verifiable metadata of Inter-domain networks properties in an agnostic way
  - Provides metadata of network equipment, domains and network operations on the data path
- SCION for example, provides transparency in many ways. But not transparency about network equipment and domains.
- Programmable Data Planes (PDP), e.g. based on P4, allows fine grained state information from routers and forwarding paths
  - For our goals on transparency, PDPs appear to be the best towards it

EXISTING TECHNOLOGIES

CONTROLLABILITY

- Path-Aware Networks (PANs) enables end-hosts to select the path their data will follow in the level of Autonomous Systems
  - Under the IETF, PANs are considered indispensable towards a secure Internet architecture
  - Several future Internet architectures incorporate path awareness within them (SCION, NEBULA, XIA…)
  - Unfortunately, the current Internet is completely “Path-Unaware”

- Segment Routing is one solution that allows controlling data paths on intra-domain scenarios, partially solving our problem

EXISTING TECHNOLOGIES

ACCOUNTABILITY

- The previously analyzed PANs are also accountable
  - With them, it is possible to achieve even real-time accountability, where packets are verified in a hop-by-hop basis, achieving the finest granularity for path verification
  - The current Internet is “Path-Unaware” so we must search for alternatives that work with the current protocols as well

- Tracing paths (e.g. with Netflow) makes it possible to monitor data paths and provide a posteriori feedback to the user (that is, after message exchange is done)
  - For example, we can infer a combination of segment routing and netflow for giving accountability and controllability to users (unfortunately, on single-domains only)

EXISTING TECHNOLOGIES

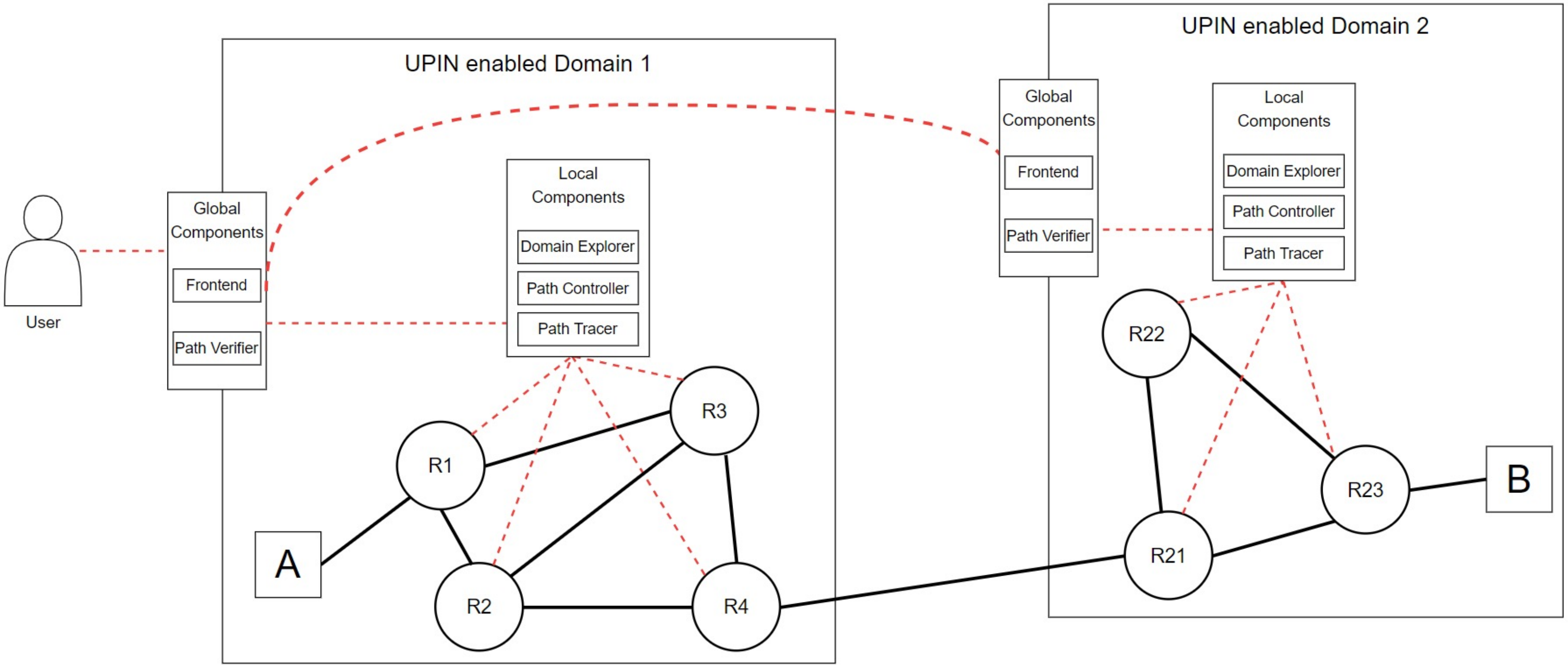
- No single solution offers a solution to all our desired properties

- Affirming our idea that a new design that combines aspects from these technologies is needed

Solution	Transparency	Accountability	Controllability
Programmable Data Planes	X	X	-
Segment Routing	-	-	X
Path-Aware Networking	-	X	X

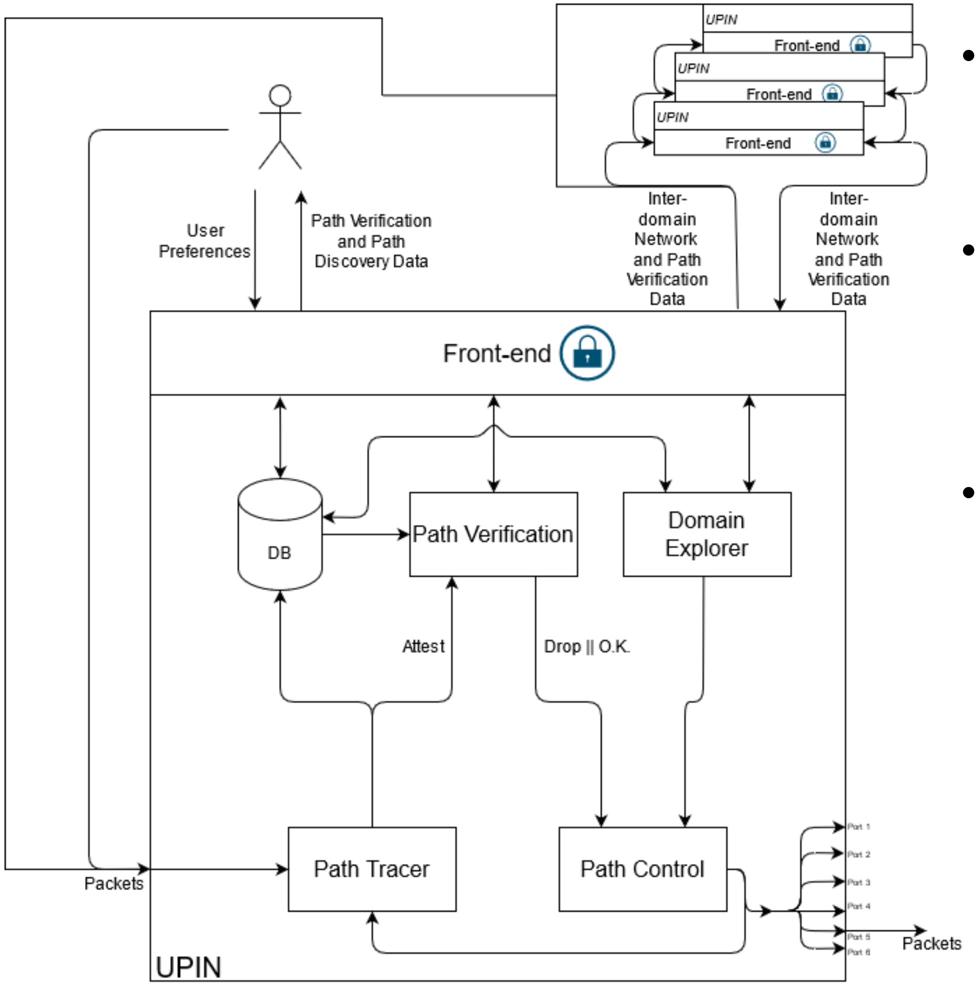
03. UPIN SOFTWARE ARCHITECTURE

UPIN SOFTWARE ARCHITECTURE



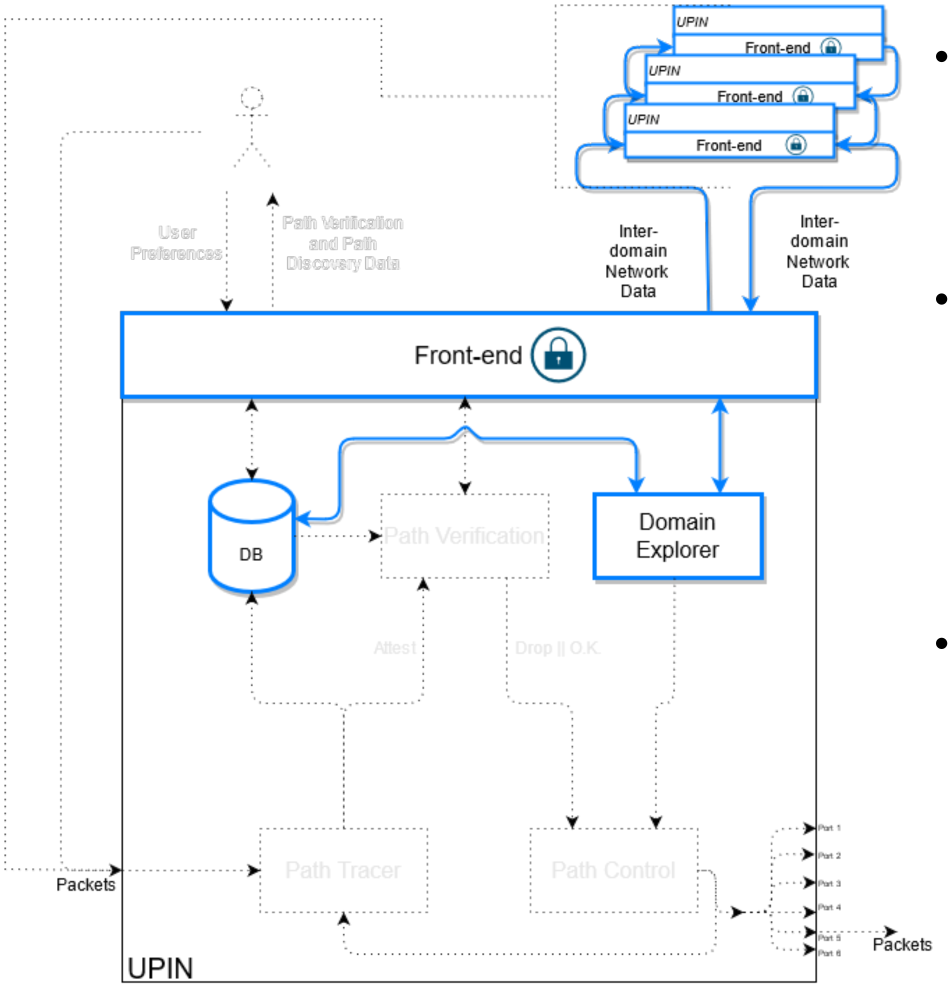
UNIVERSITY
OF TWENTE.

UPIN SOFTWARE ARCHITECTURE



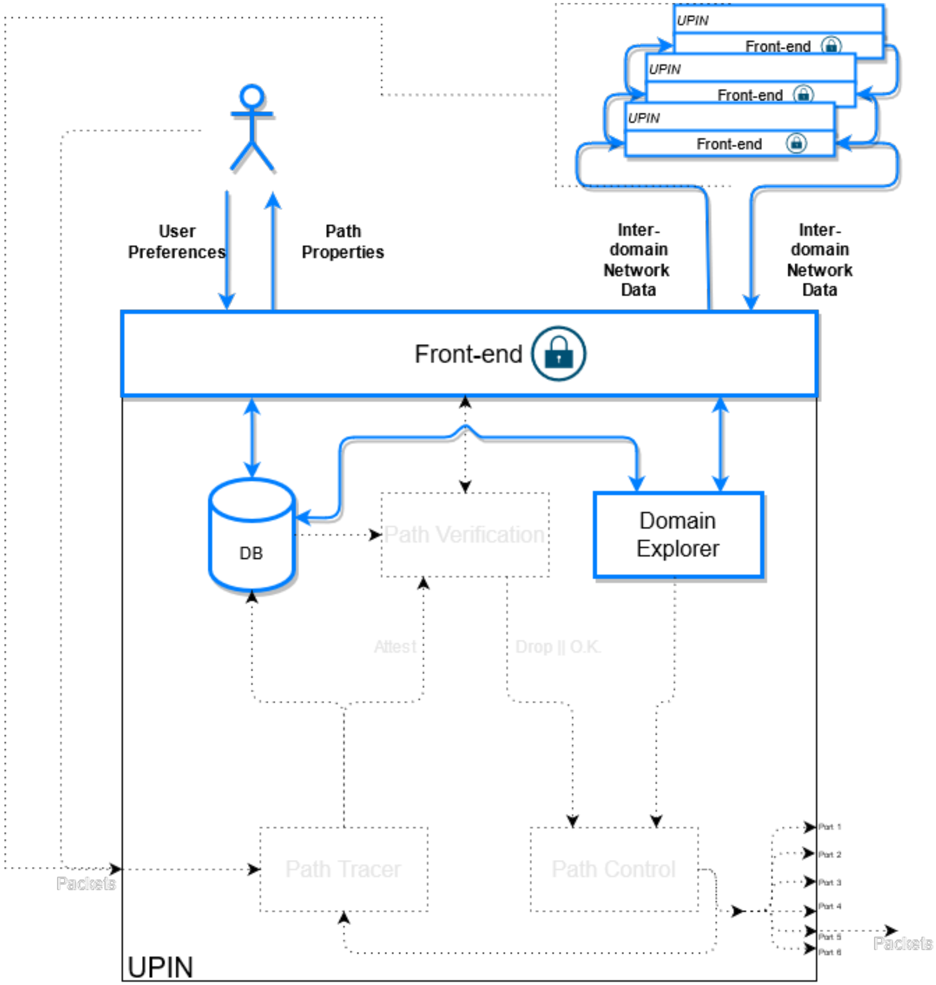
- Initial software architecture of the UPIN prototype
- All components from our network architecture are mapped as functions in the diagram
- Current Envisioned “Threads”:
  1. Domain Exploring
  2. User Settings
  3. Path Controlling
  4. Path Verification

DOMAIN EXPLORING



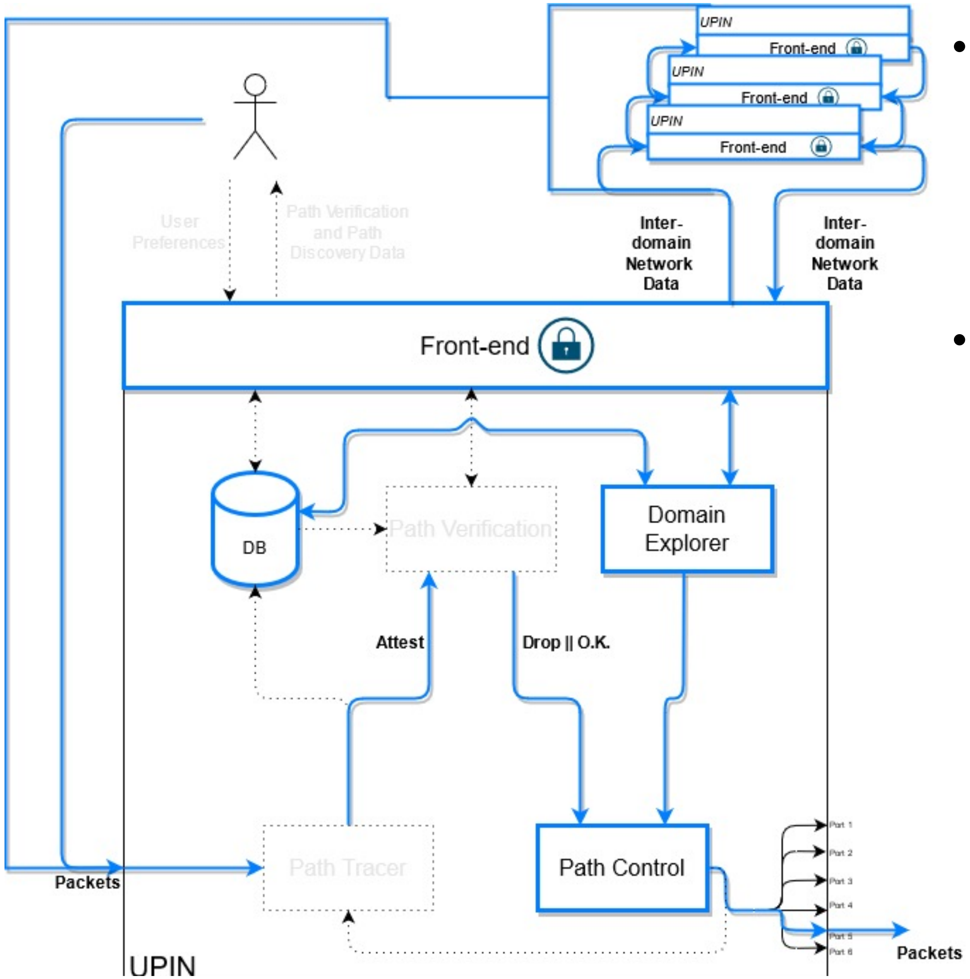
- The UPIN prototype will constantly probe other domains for their information
- Our database will constantly be updated with data of other domains in order to keep overhead and latency to a minimum
- All information flowing through the Front-end must be encrypted

USER SETTINGS



- Users access the system through the Front-end. Existing information of users is fetched if it exists
- When setting their preferences, users add a specific destination and the system returns the available properties for that path to the user (if there is no info for that destination on the DB, the Domain Explorer will be prompted to fetch it)
- All information flowing through the Front-end must be encrypted

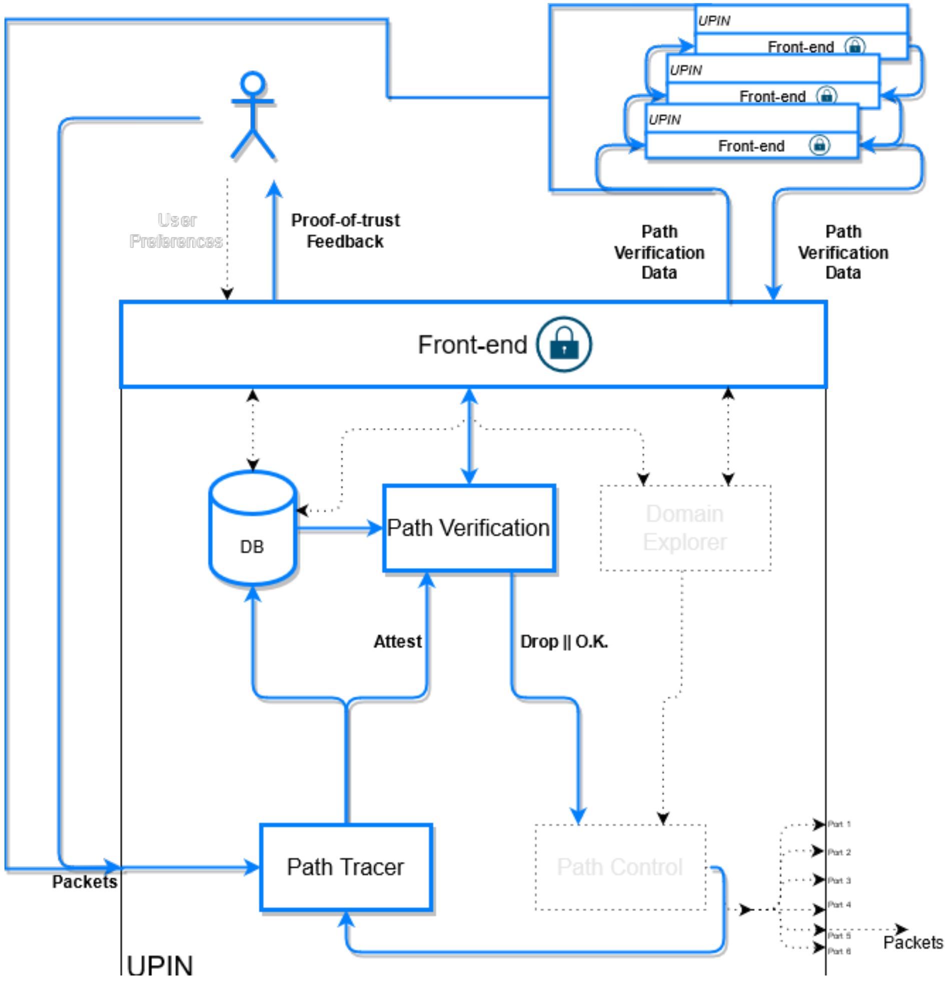
PATH CONTROLLING



- With settings in place, the system starts routing data based on preferences set or Inter-domain data received by other domains

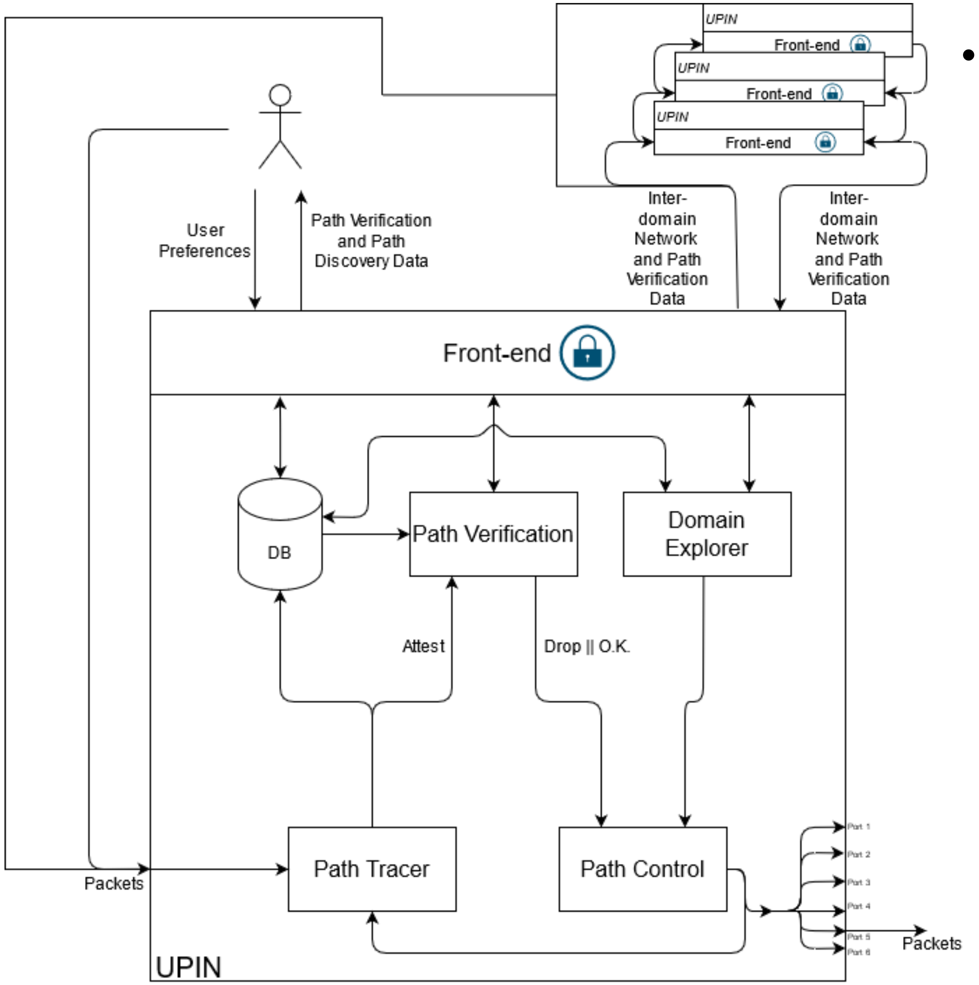
- Upon receival of packets by the Path Control module, it forwards the packets based on data provided by the Domain Explorer and/or embedded in the packet's headers

PATH VERIFICATION



- Upon receival of new packets or Path Verification Data from other domains, the software proceeds to conduct the Path Verification
- Traces are gathered in the ingress and egress of the router for verification purposes
- Traces are saved on the DB and verification is executed, forwarding a proof-of-trust to the user
- Verification can happen on real-time or a posteriori, depending of the verification method desired by the user or requested by other domains

UPIN SOFTWARE ARCHITECTURE



- This is the first version of the software architecture of UPIN router that will be implemented in P4 as a deliverable for the project
  - Modifications may be necessary

04. PATH VERIFICATION EXPERIMENTS

PATH VERIFICATION EXPERIMENTS

- A verification system in a simulated environment will be developed in P4 as my next UPIN task

- Fake GPS coordinates will be embedded into packets headers for verification purposes
  - Create simple verification rules for these GPS coordinates, to be embedded by the source. This will define the routers that should route the data.

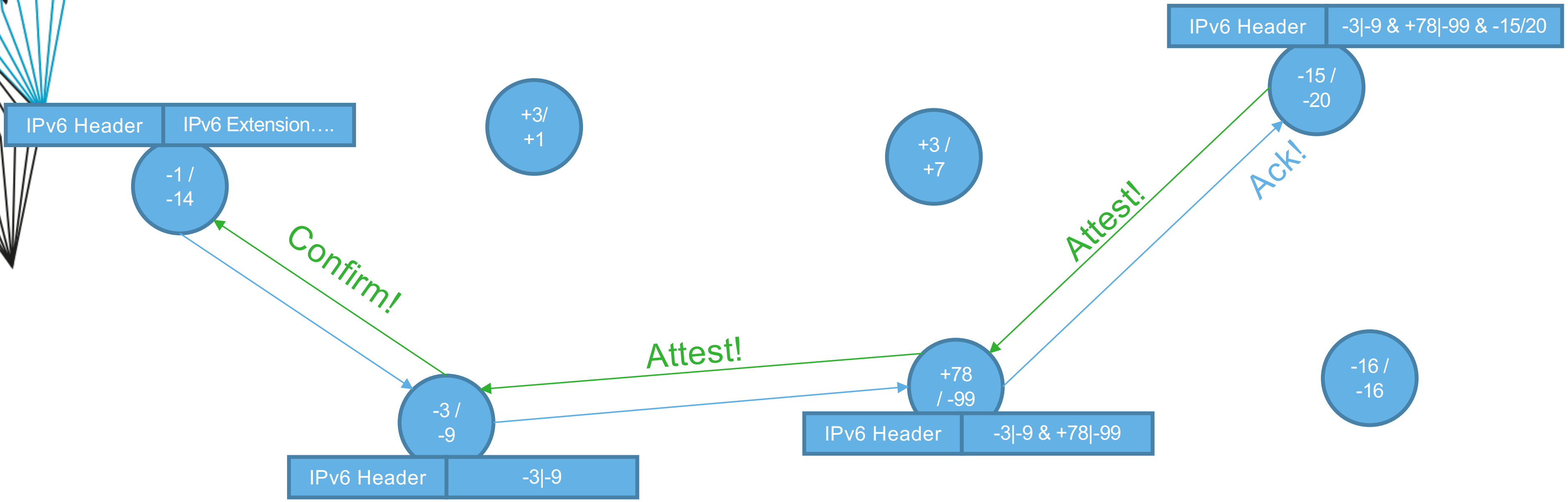
- First idea is to verify the data with the use of public-key cryptography
  - Cryptography will most likely be developed with python due to easy prototyping

PATH VERIFICATION EXPERIMENTS

- Verify the added latency and overall overheads of adding these labels into the packets.
  - And cryptographically verifying them.

- Have the system to send data with and without probing other routers about their gps coordinates. Simulating a simple version of “Domain Explorer” component.
  - Hypothesis: Forwarding data without this will take way longer and significantly increase the latency.

PATH VERIFICATION EXPERIMENTS



05. REQUIREMENTS AND USERS SURVEYING

REQUIREMENTS AND USERS SURVEYING

- We started initial contact with other researchers from other areas in the last months.
- Initial meetings with a researcher from the robotics team from the UT were conducted in order to gather requirements
  - Notes and observations from our meetings will be written in the form of a blog in the future

- We are looking at conducting further surveys with other industry people and researchers in order to gather more requirements for further elaborating the research

UPIN PROGRESS MEETING

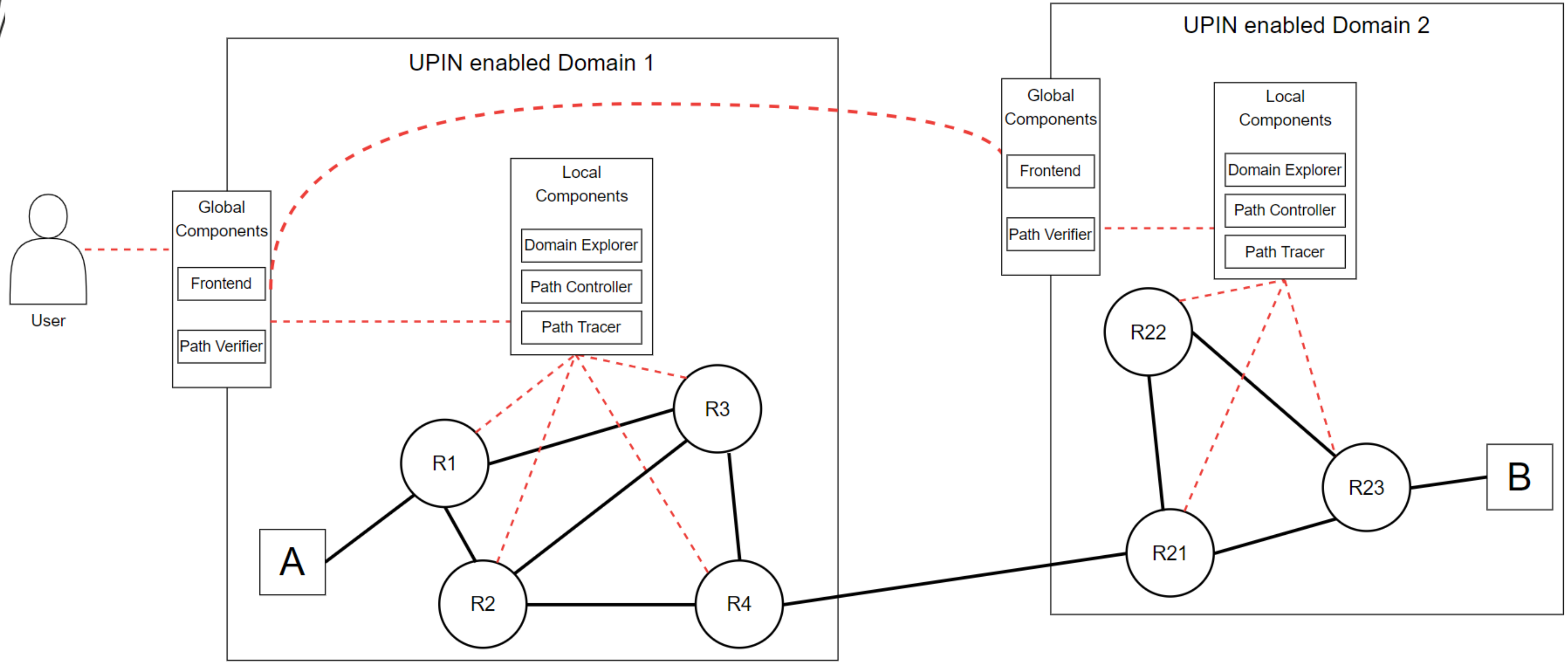
LEONARDO BOLDRINI

UPIN FRAMEWORK

- The presented use-cases and existing technologies are the background of the UPIN framework
- The framework intends to achieve larger levels of transparency, accountability and controllability in Inter-domain networks
- Each component of the architecture assesses one or more of our desired properties
- The framework does not mandate the underlying data plane technology in each domain

UPIN FRAMEWORK

ARCHITECTURE



UNIVERSITY OF AMSTERDAM

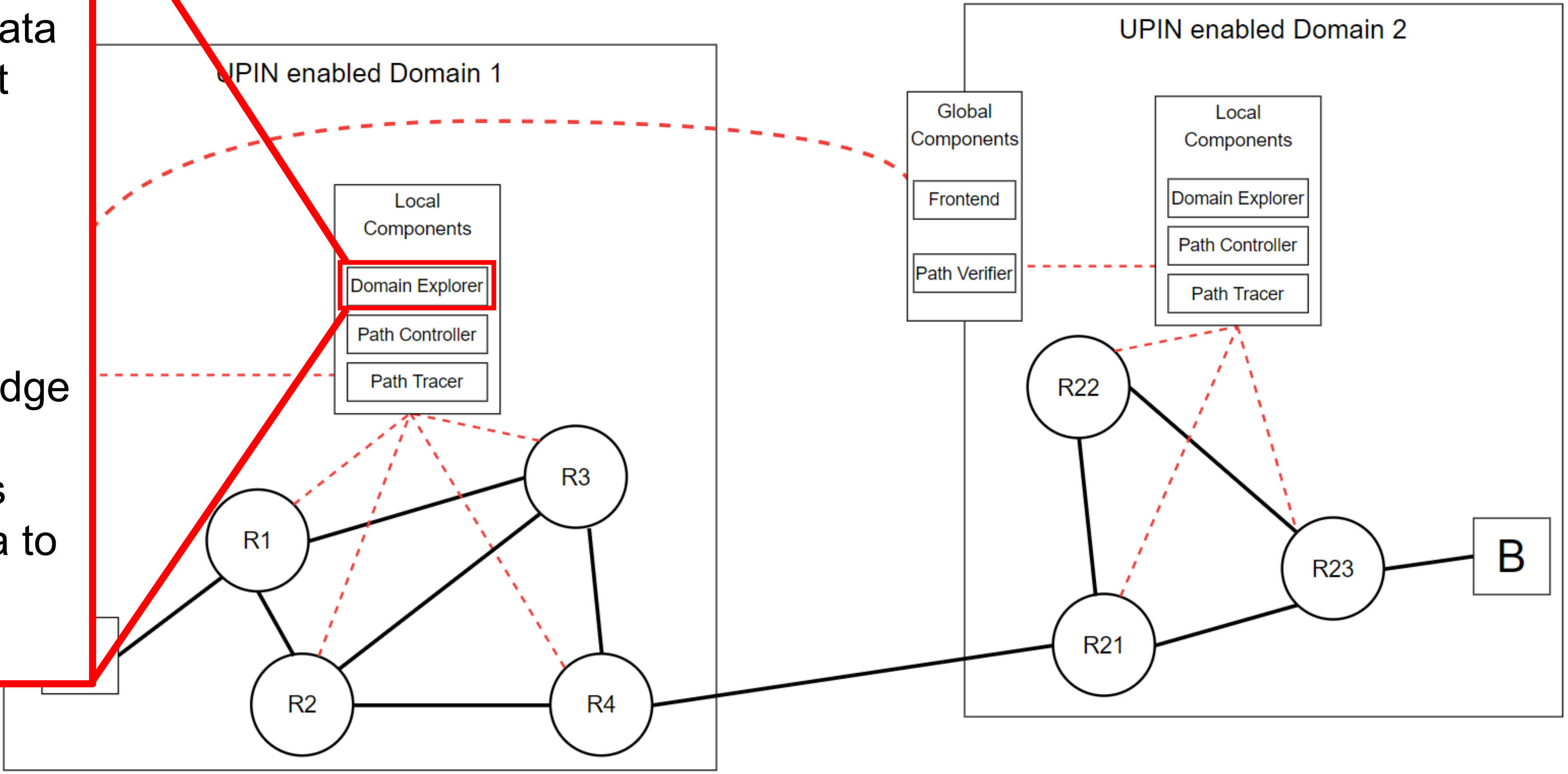
UNIVERSITY OF TWENTE.

UPIN FRAMEWORK

TECTURE

Domain Explorer

- Obtains and stores metadata about domain’s equipment and keeps data updated
- Topology, source code of routers, geographical characteristics, …
- Local view on its domain
- Deep and detailed knowledge on its nodes
- Domain’s operator defines policies of which metadata to share with other domains

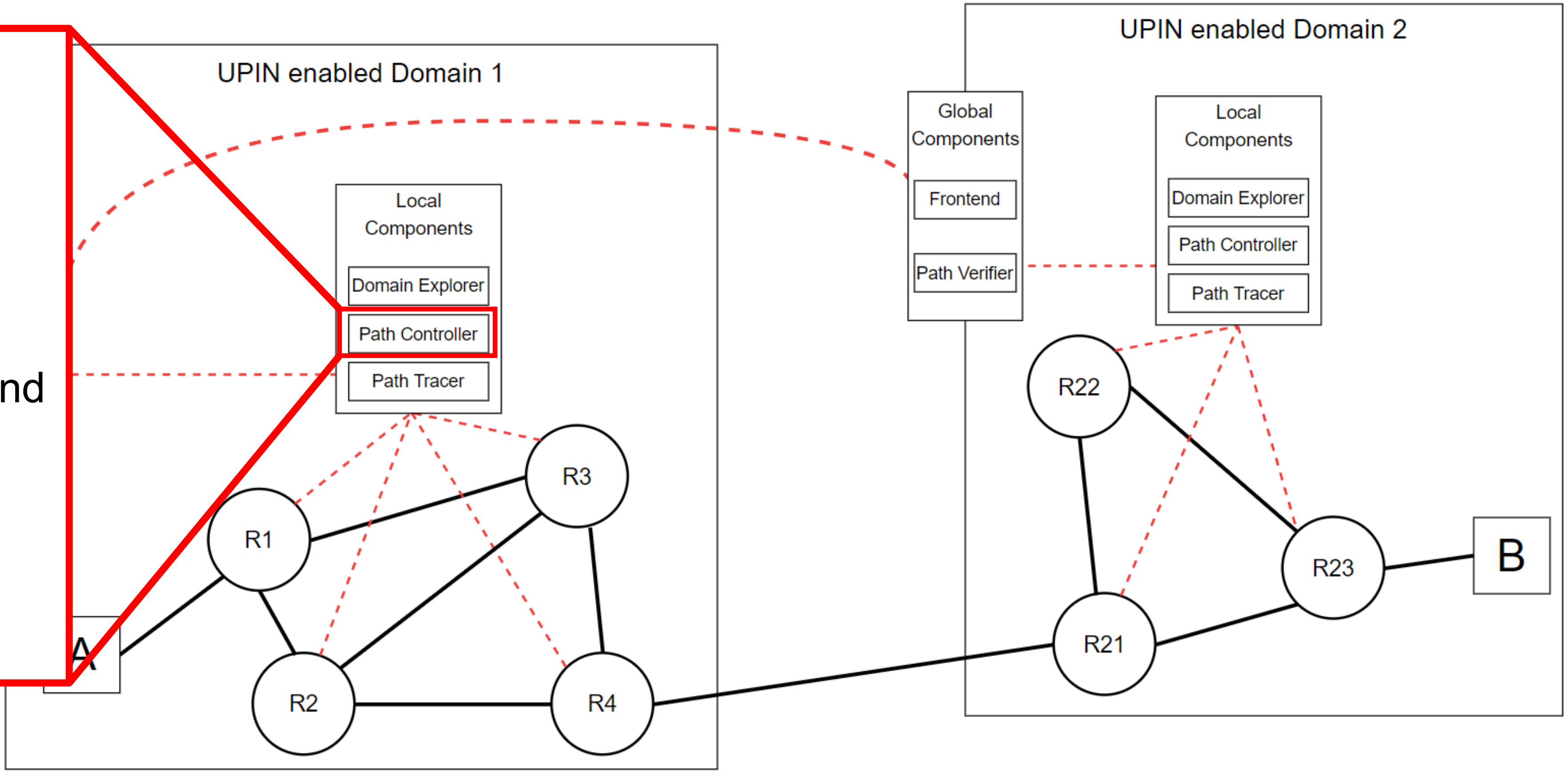


UPIN FRAMEWORK

ARCHITECTURE

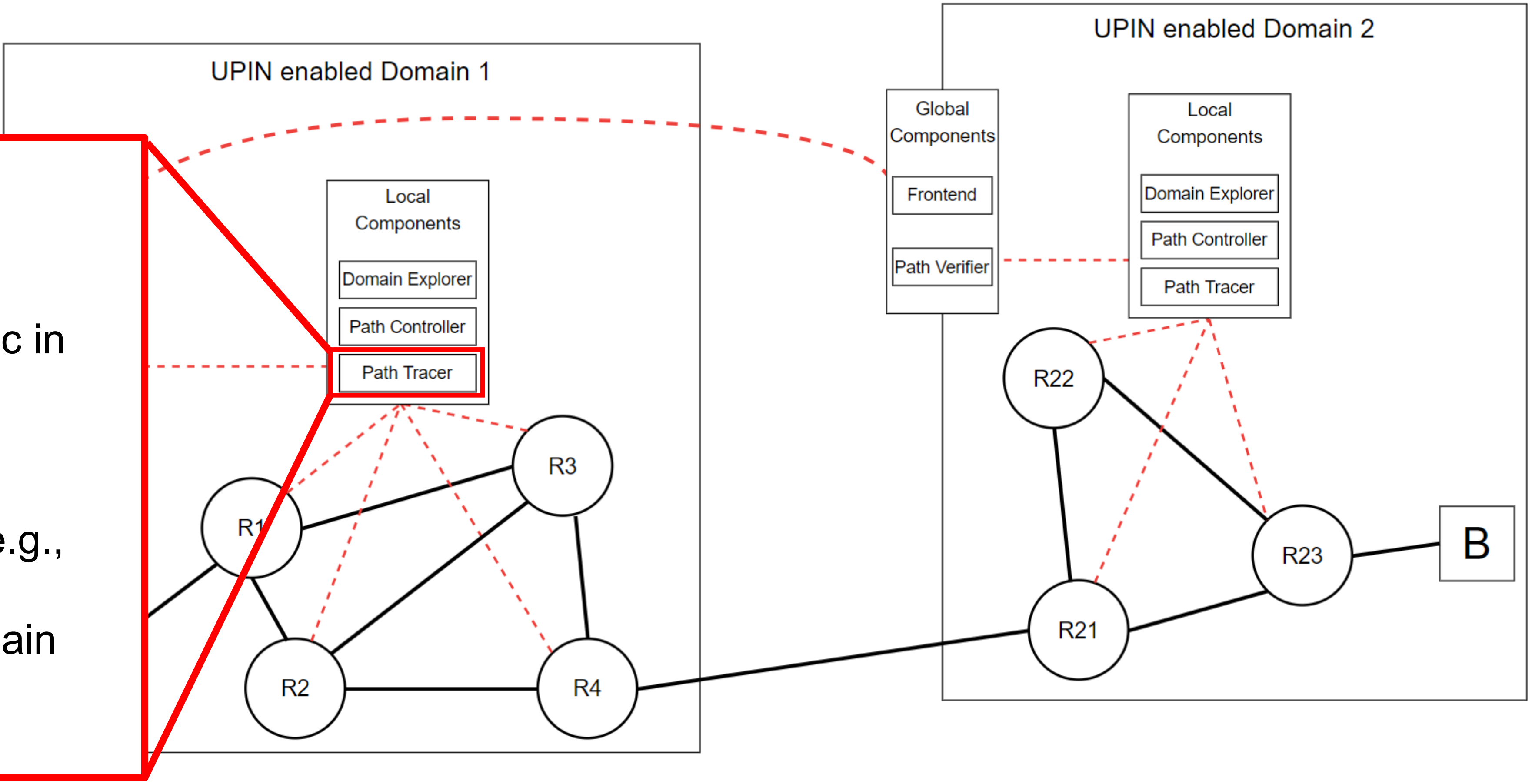
Path Controller

- Sets forwarding rules in its domain
- Rules based on user's request
- Instructions and rules depend on the technologies that nodes use (e.g., Segment Routing)
- Local scope on its domain



UPIN FRAMEWORK

ARCHITECTURE

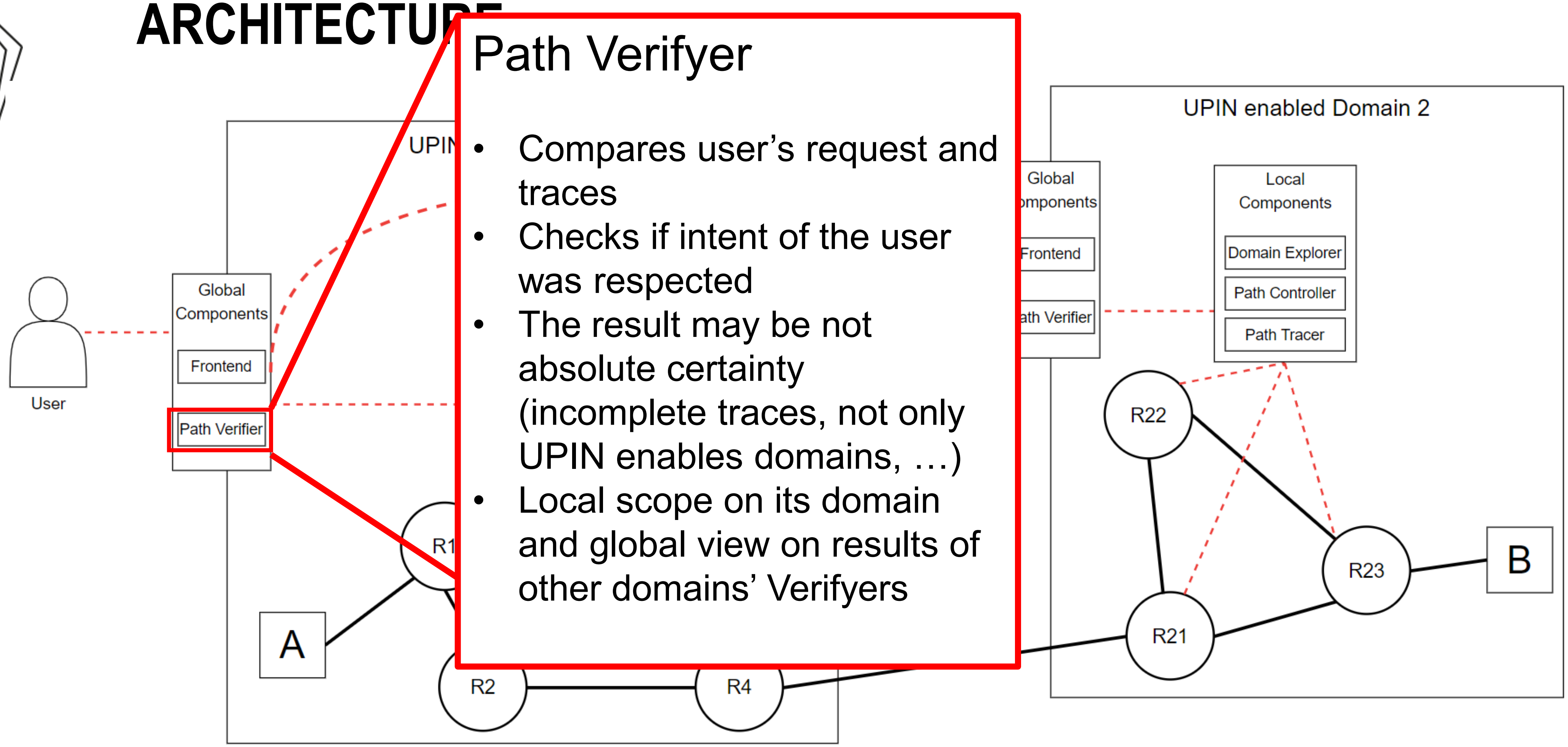


Path Tracer

- Gathers real-time measurements on traffic in the data plane
- Stores traces and any information useful for verification purposes (e.g., nodes traversed)
- Local scope on its domain only and technology dependent

UPIN FRAMEWORK

ARCHITECTURE

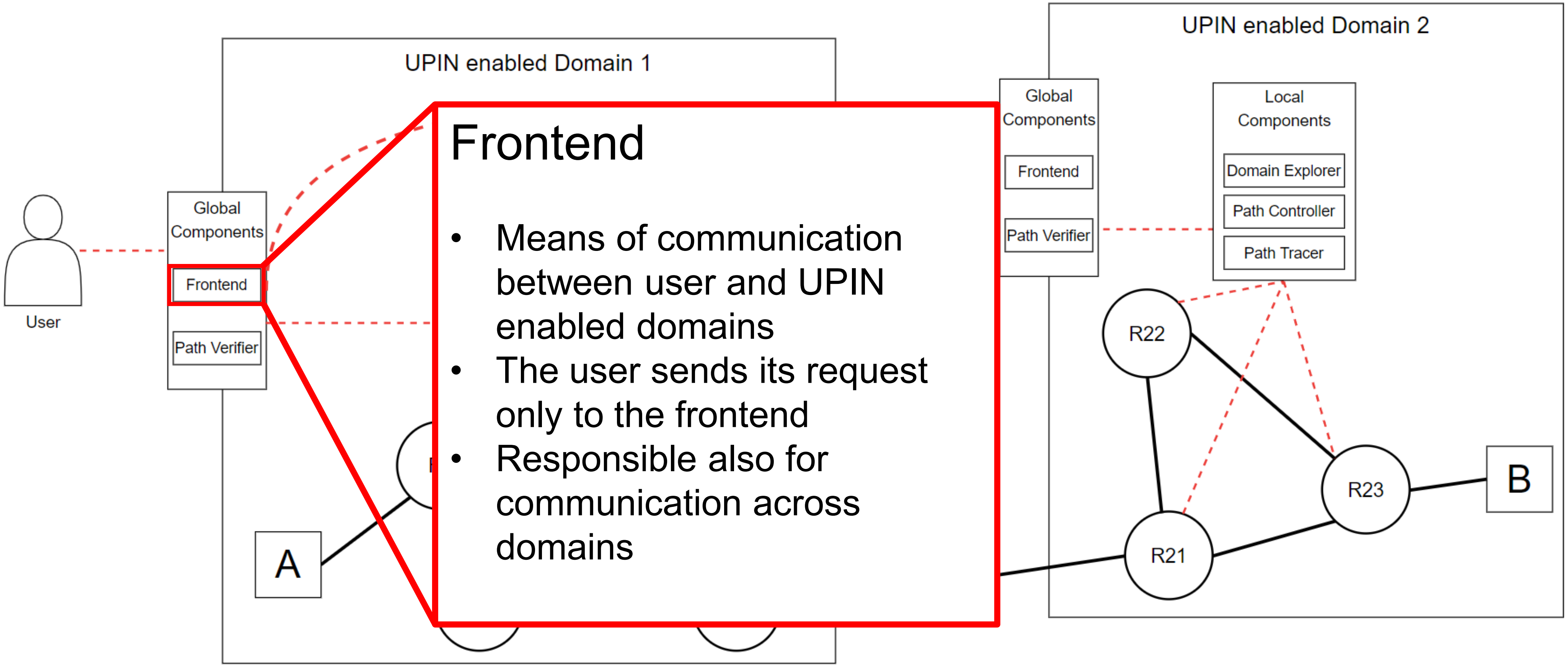


Path Verifyer

- Compares user's request and traces
- Checks if intent of the user was respected
- The result may be not absolute certainty (incomplete traces, not only UPIN enables domains, …)
- Local scope on its domain and global view on results of other domains' Verifyers

UPIN FRAMEWORK

ARCHITECTURE

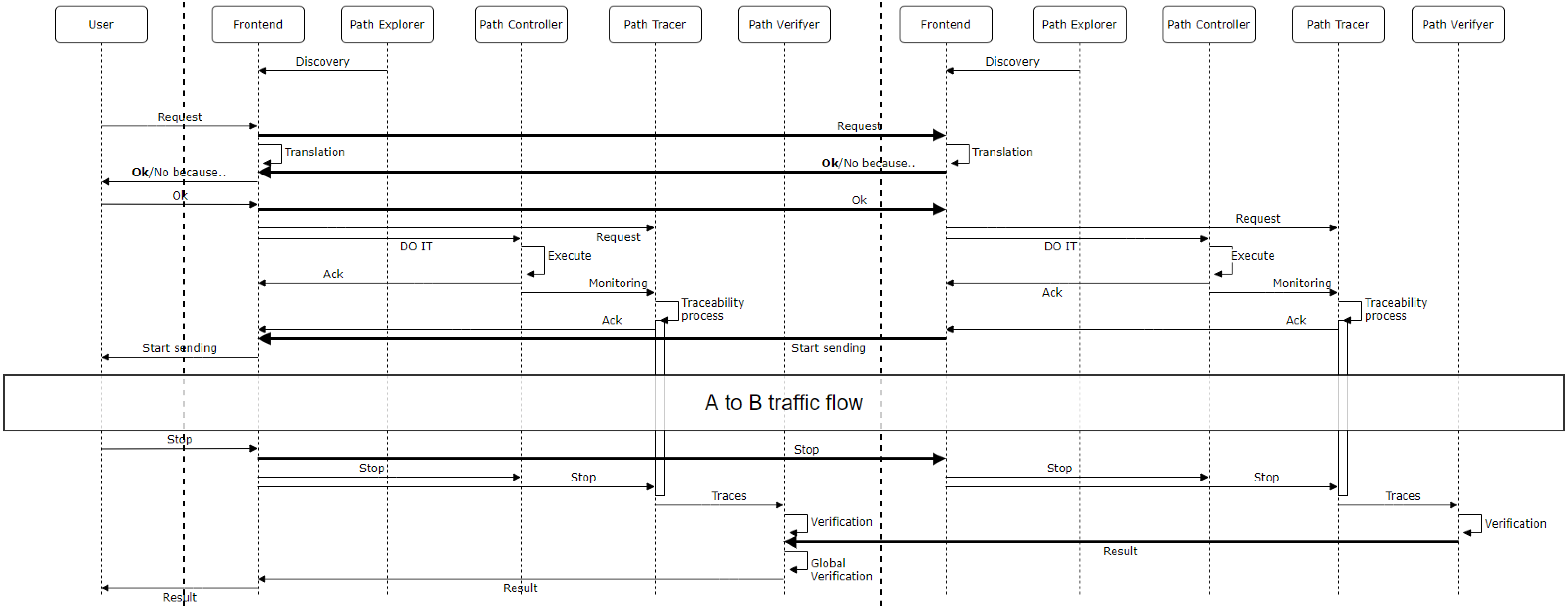


Frontend

- Means of communication between user and UPIN enabled domains
- The user sends its request only to the frontend
- Responsible also for communication across domains

UPIN FRAMEWORK

ARCHITECTURE



UNIVERSITY OF AMSTERDAM

UNIVERSITY OF TWENTE.

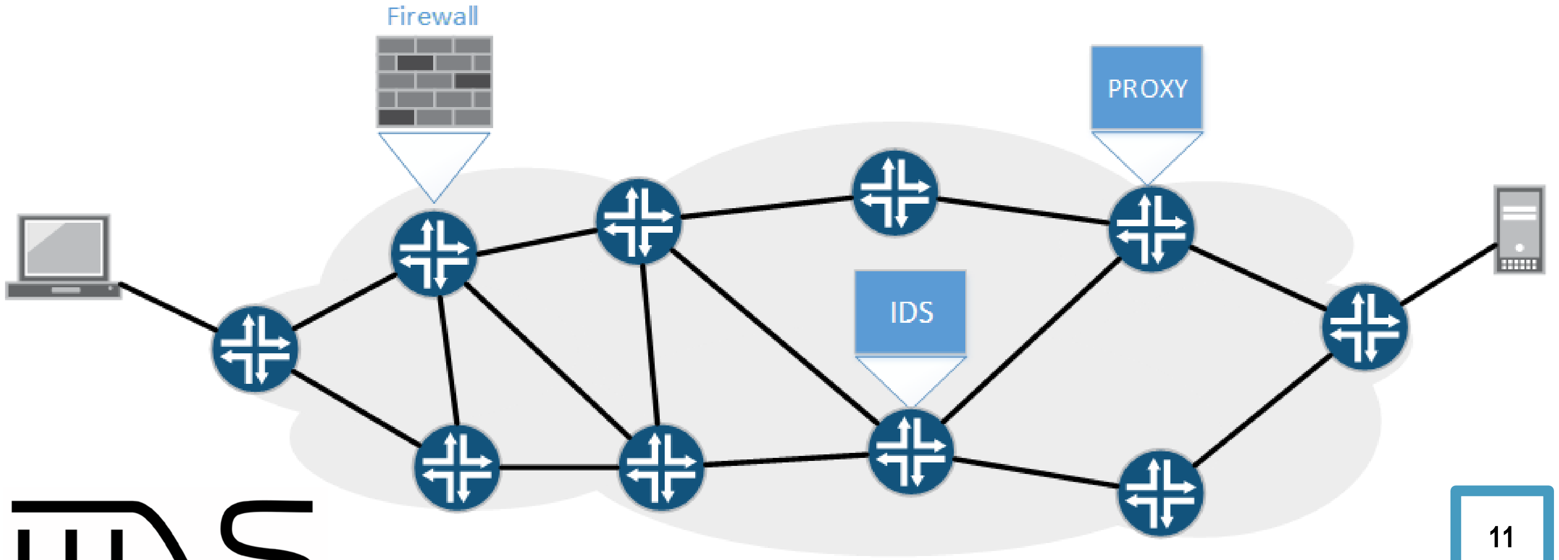
UPIN PROGRESS MEETING

EXPERIMENTS

EXPERIMENTS

PATH CONTROLLER

- We focus our first experiments on the Path Controller component
- The user already expressed his intent for his traffic to follow a specific path, and to visit determined functions
- How do we steer traffic following the user intent?

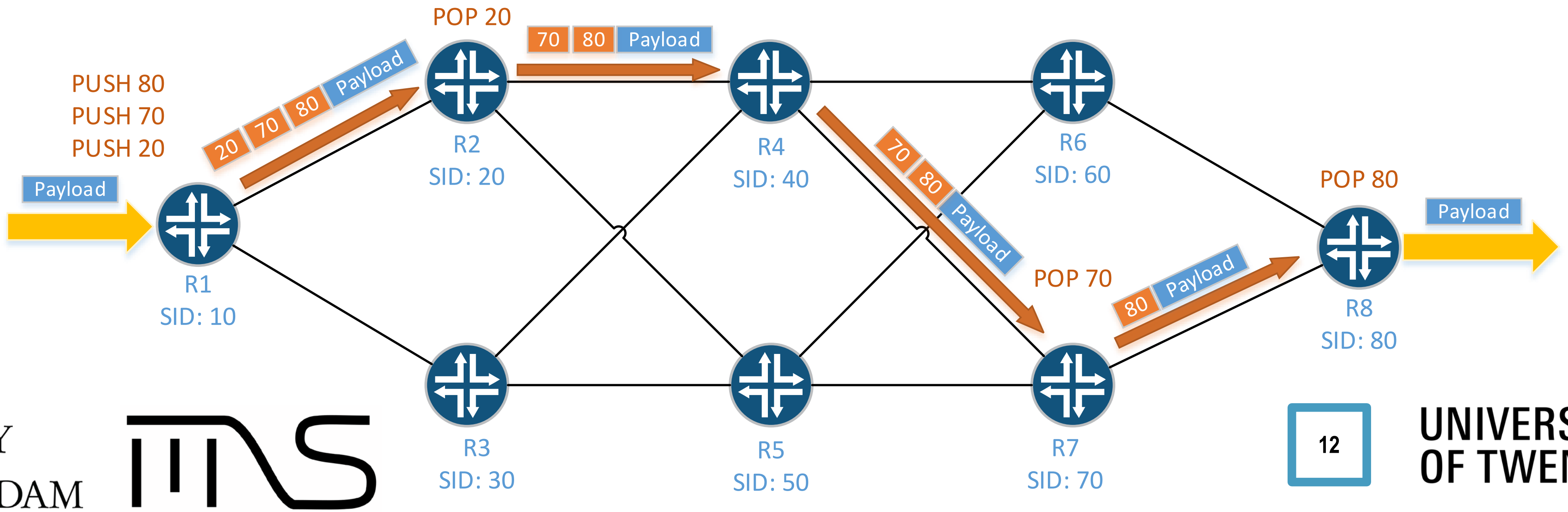


EXPERIMENTS

PATH CONTROLLER

- IP routing: destination based — Destination | Payload
- Segment Routing: source based — Segment ● ● Segment | Destination | Payload

- In the example, we push Segment Identifiers (SIDs) to steer traffic through R2 and R7



UNIVERSITY OF AMSTERDAM

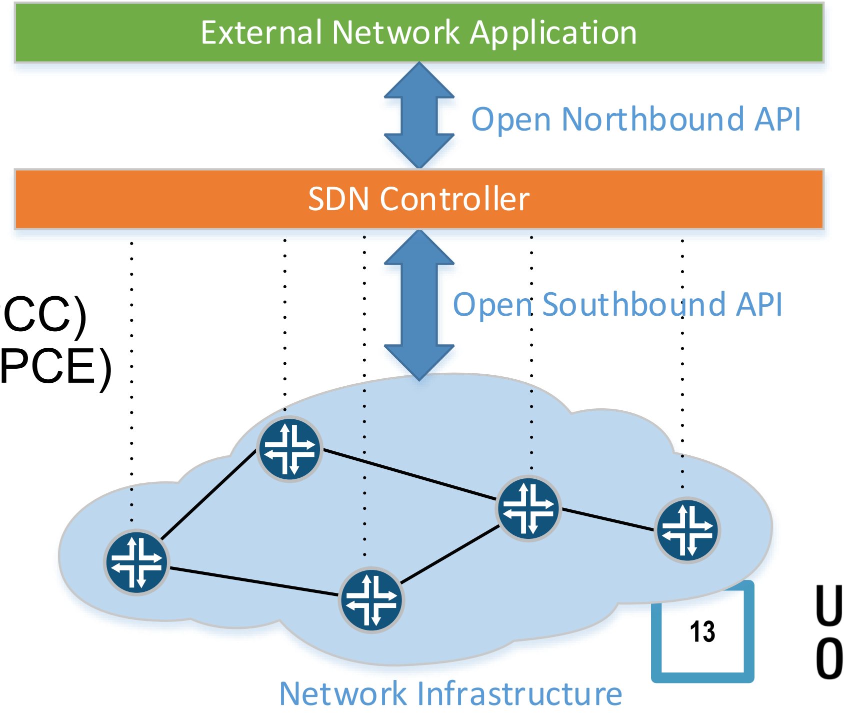
UNIVERSITY OF TWENTE.

EXPERIMENTS

PATH CONTROLLER

- SR-MPLS re-uses Multi Protocol Label Switching dataplane
- SR-MPLS paths are called Segment Routed Label Switched Paths (SR-LSP)
- IGPs with SR support: IS-IS, OSPF. In our Proof of Concept we used IS-IS

- We use the Path Computation Element Protocol (PCEP) to build paths
- Paths as Explicit Route Objects (ERO)
- Consists of Path Computation Client (PCC) and Path Computation Element (PCE)
- SDN Controller



UNIVERSITY OF AMSTERDAM

UNIVERSITY OF TWENTE.

EXPERIMENTS

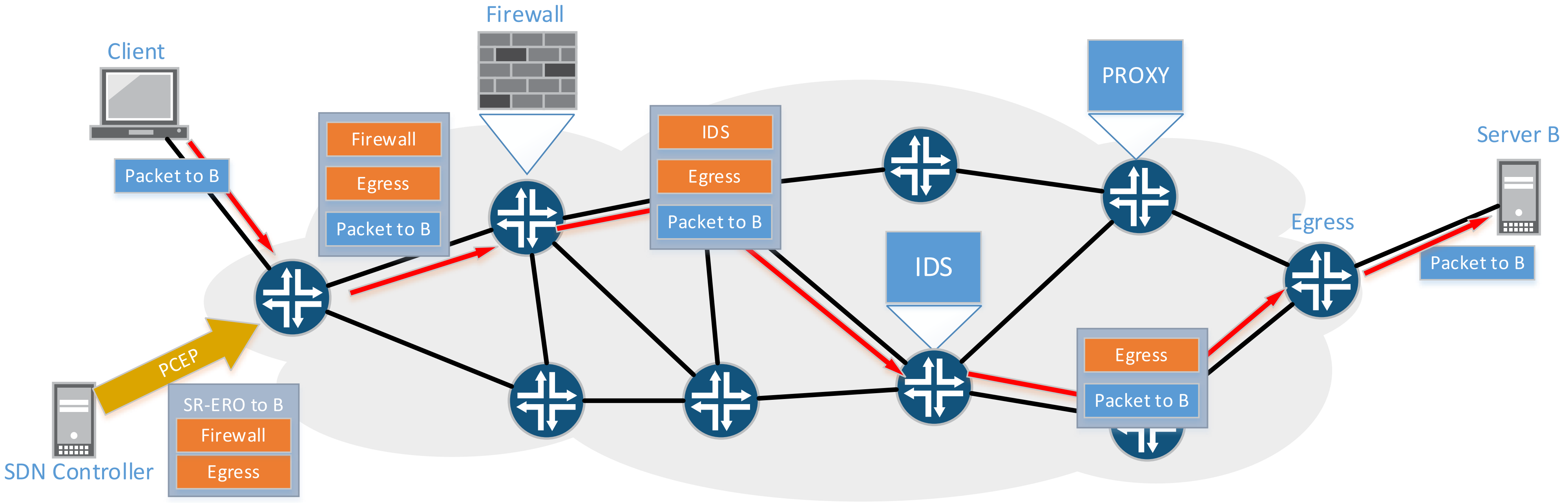
PATH CONTROLLER

- “How can we steer traffic to services deployed in the network?”

More technically:

“How can we create SR-MPLS network paths to assist the network integration of VNFs?”

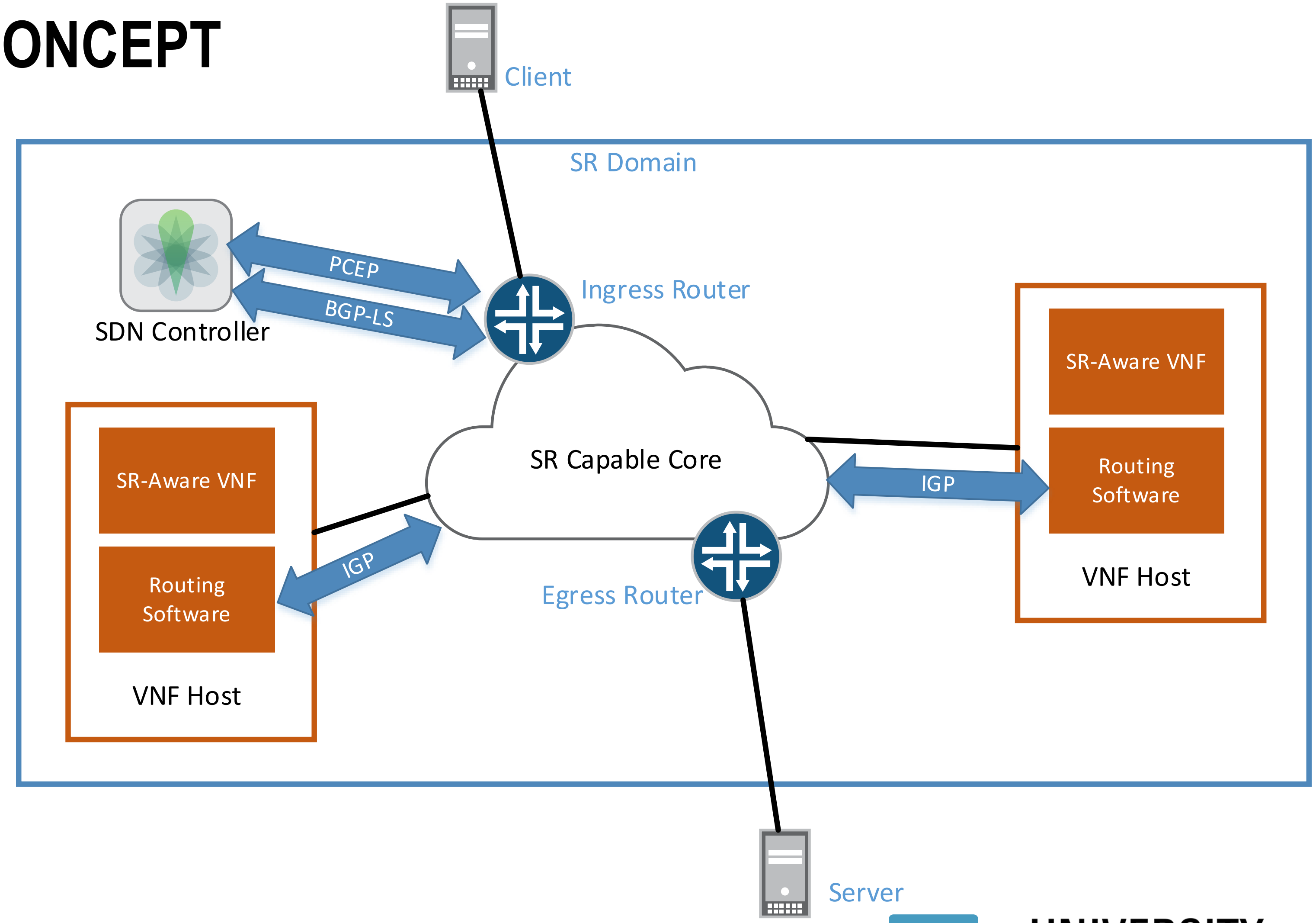
“Can PCEP be used for this purpose?”



EXPERIMENTS

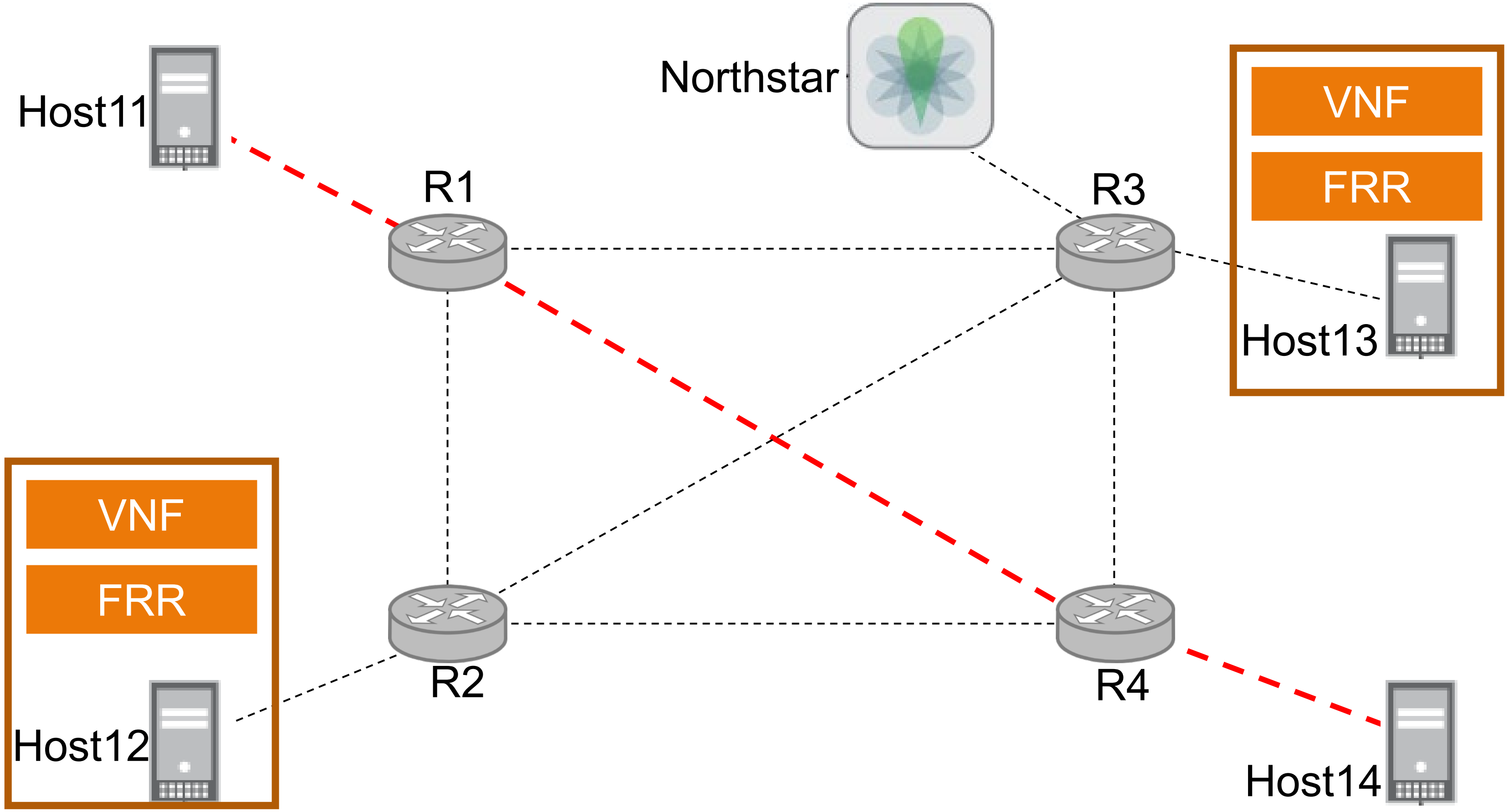
PROOF OF CONCEPT

- Juniper vQFX routers
- NorthStar SDN Controller
- Free Range Routing
  - On VNF Hosts
- Custom eBPF SR-Aware VNFs



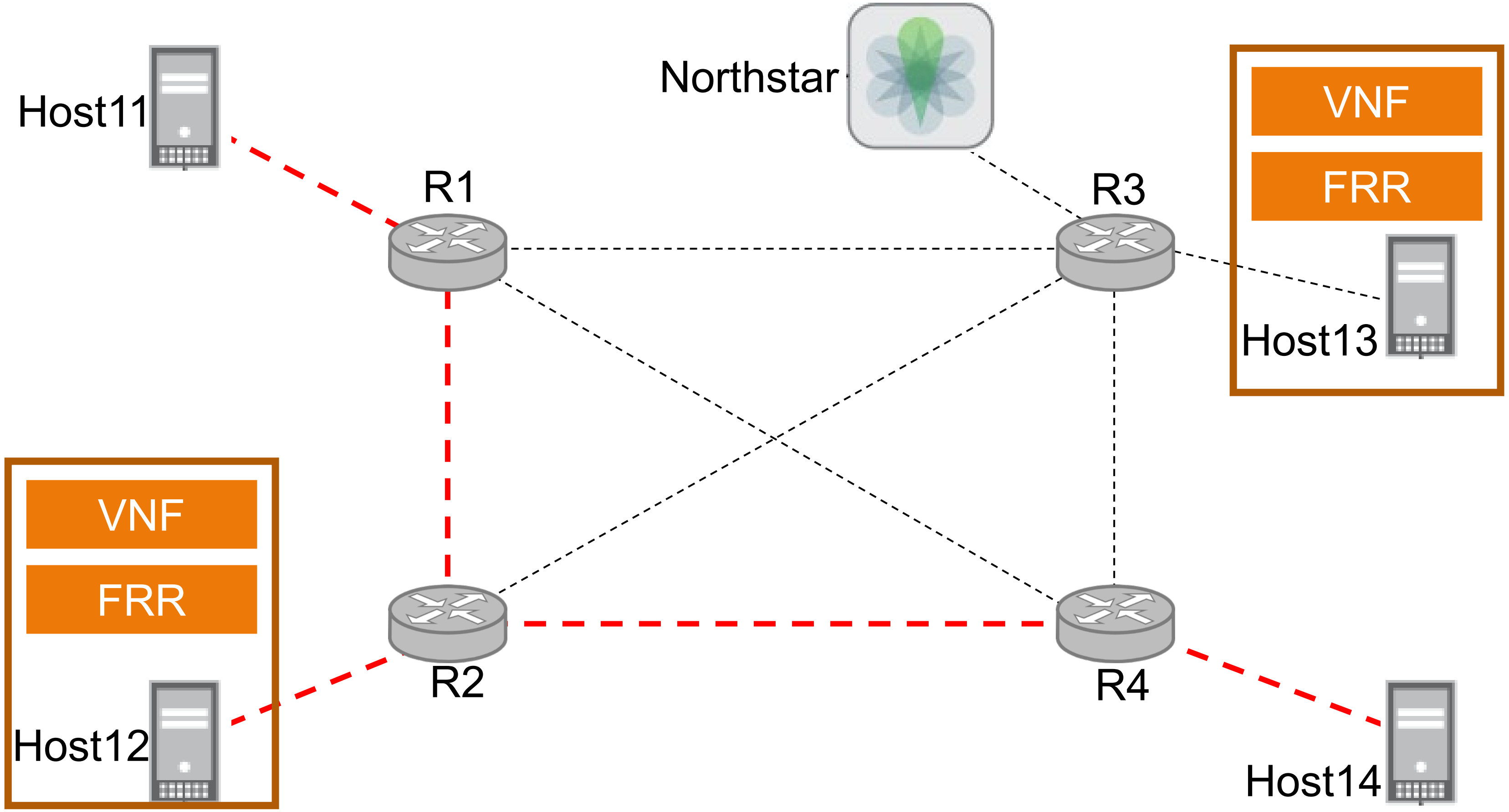
EXPERIMENTS

START STATE OF THE NETWORK



EXPERIMENTS

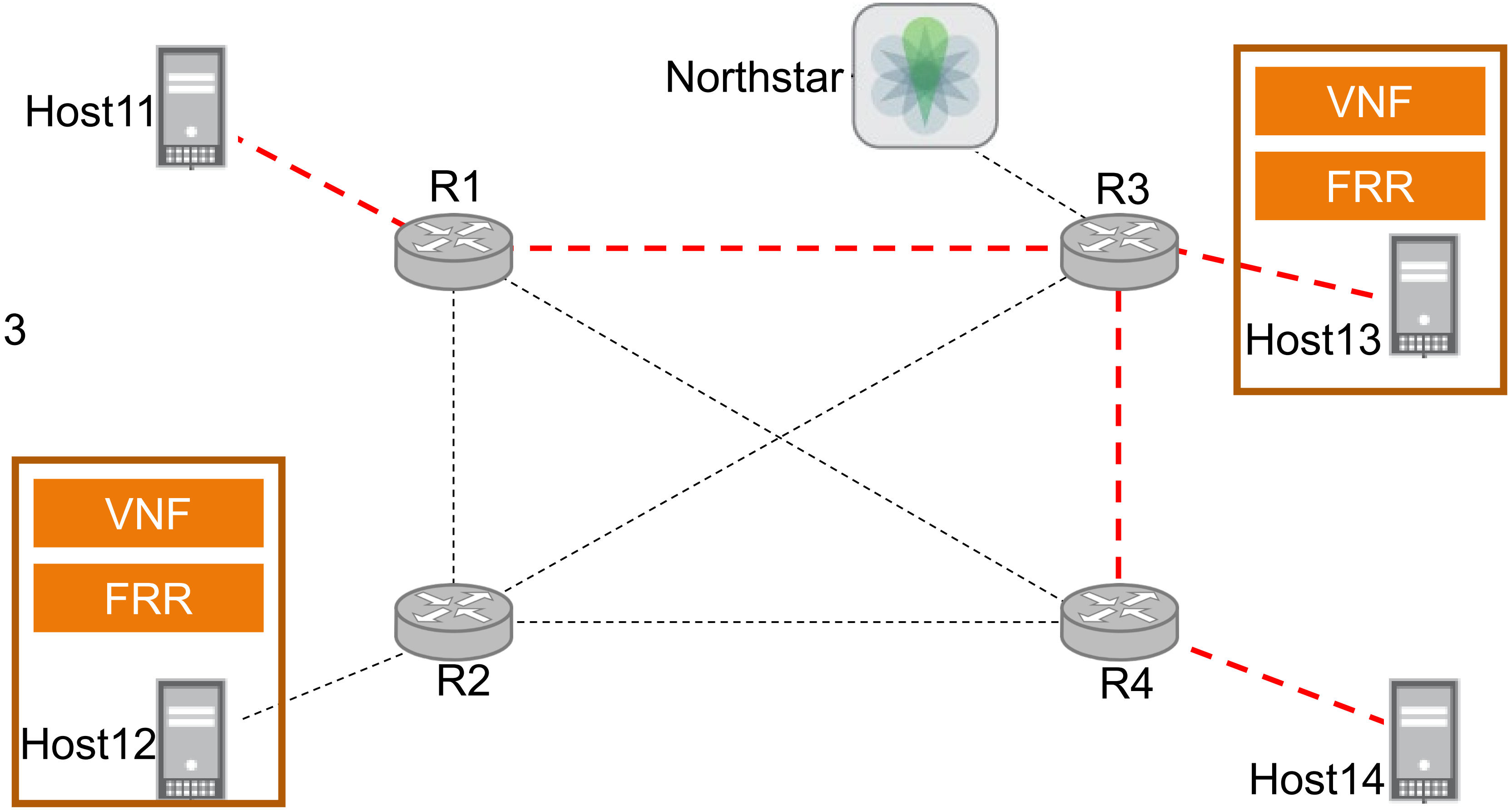
PATH THROUGH VNF IN HOST12



EXPERIMENTS

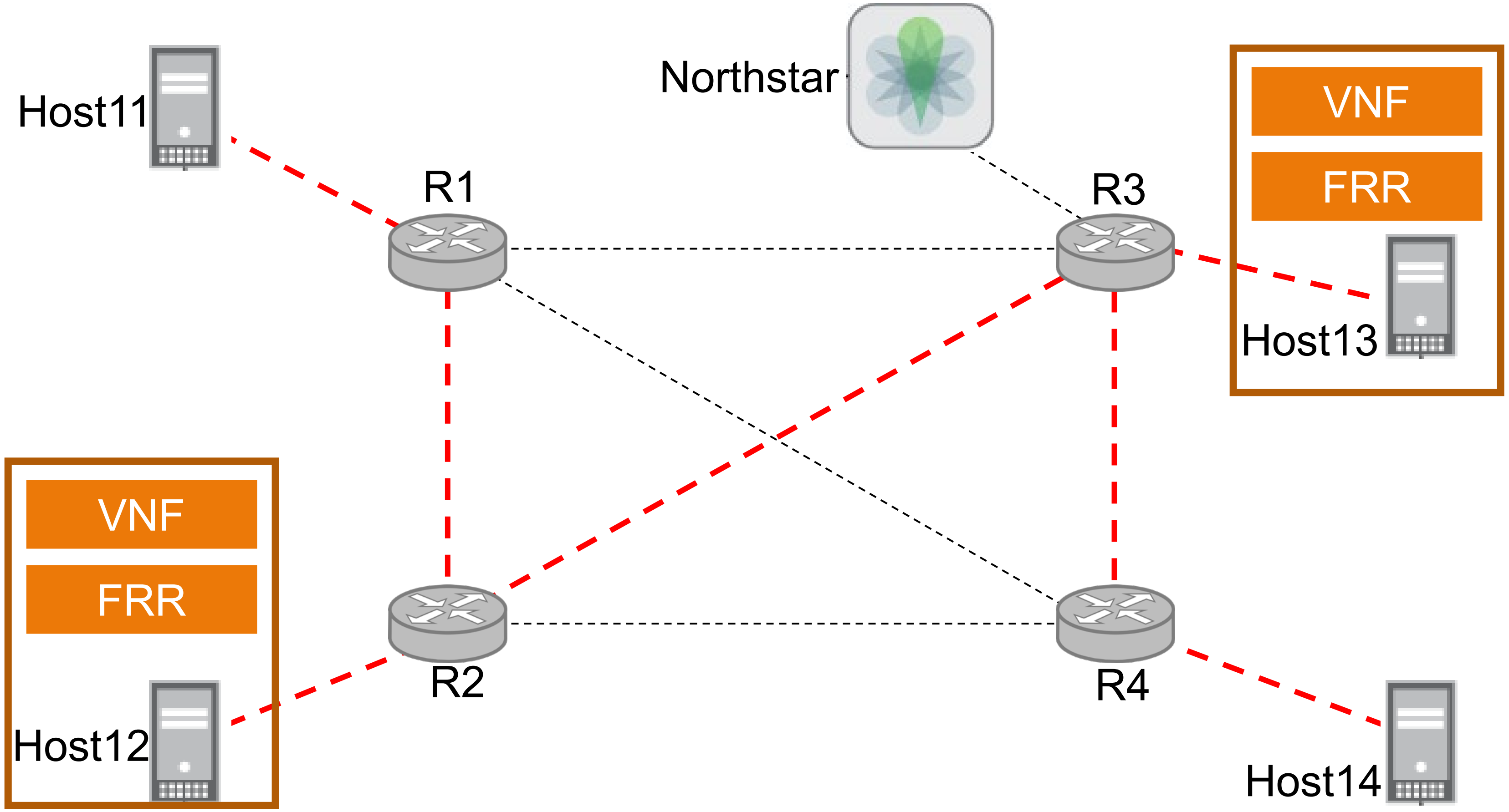
RE-INSTANTIATION OF VNF

Host12 -> Host13



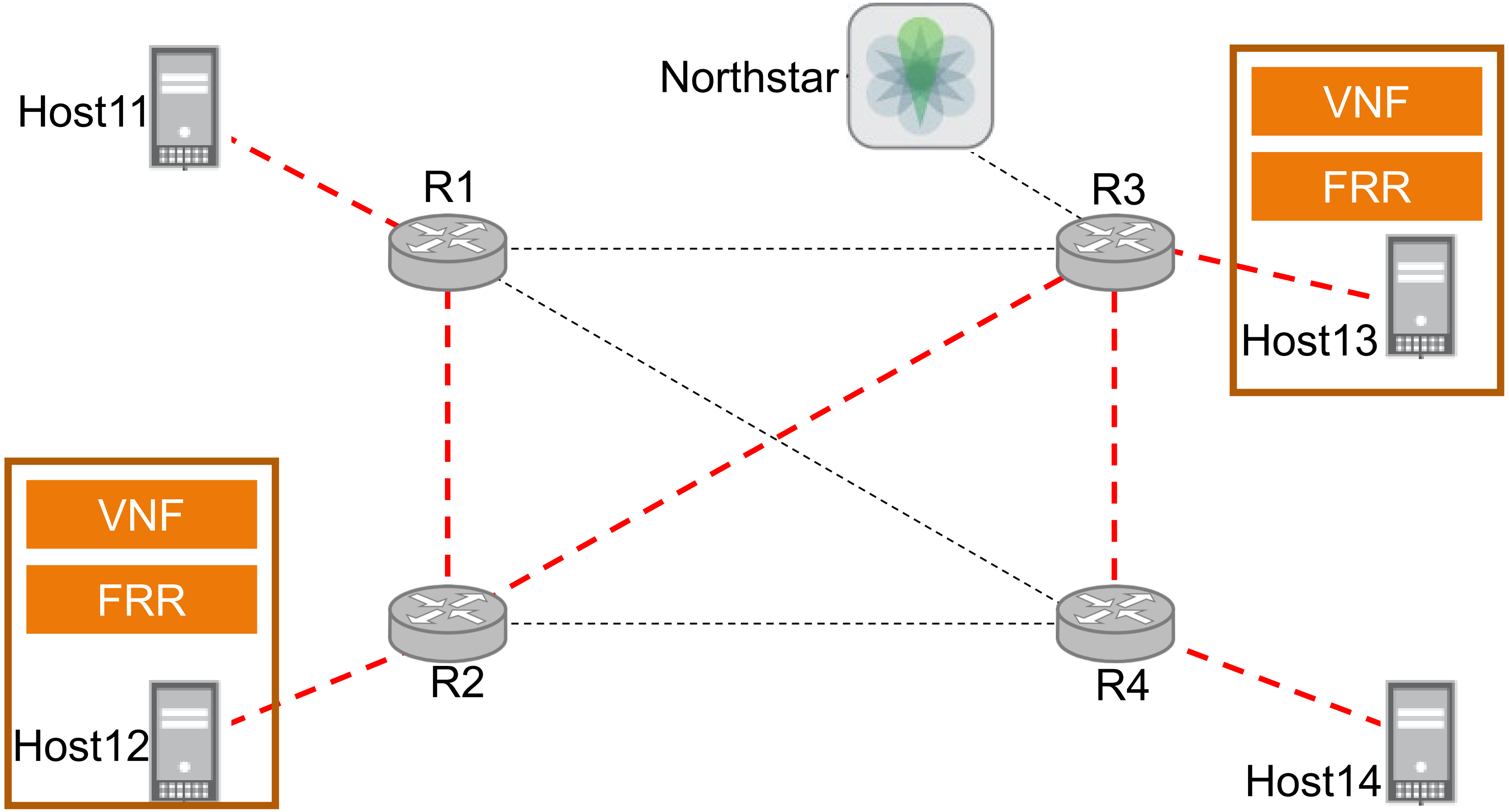
EXPERIMENTS

CHAIN OF 2 VNFS



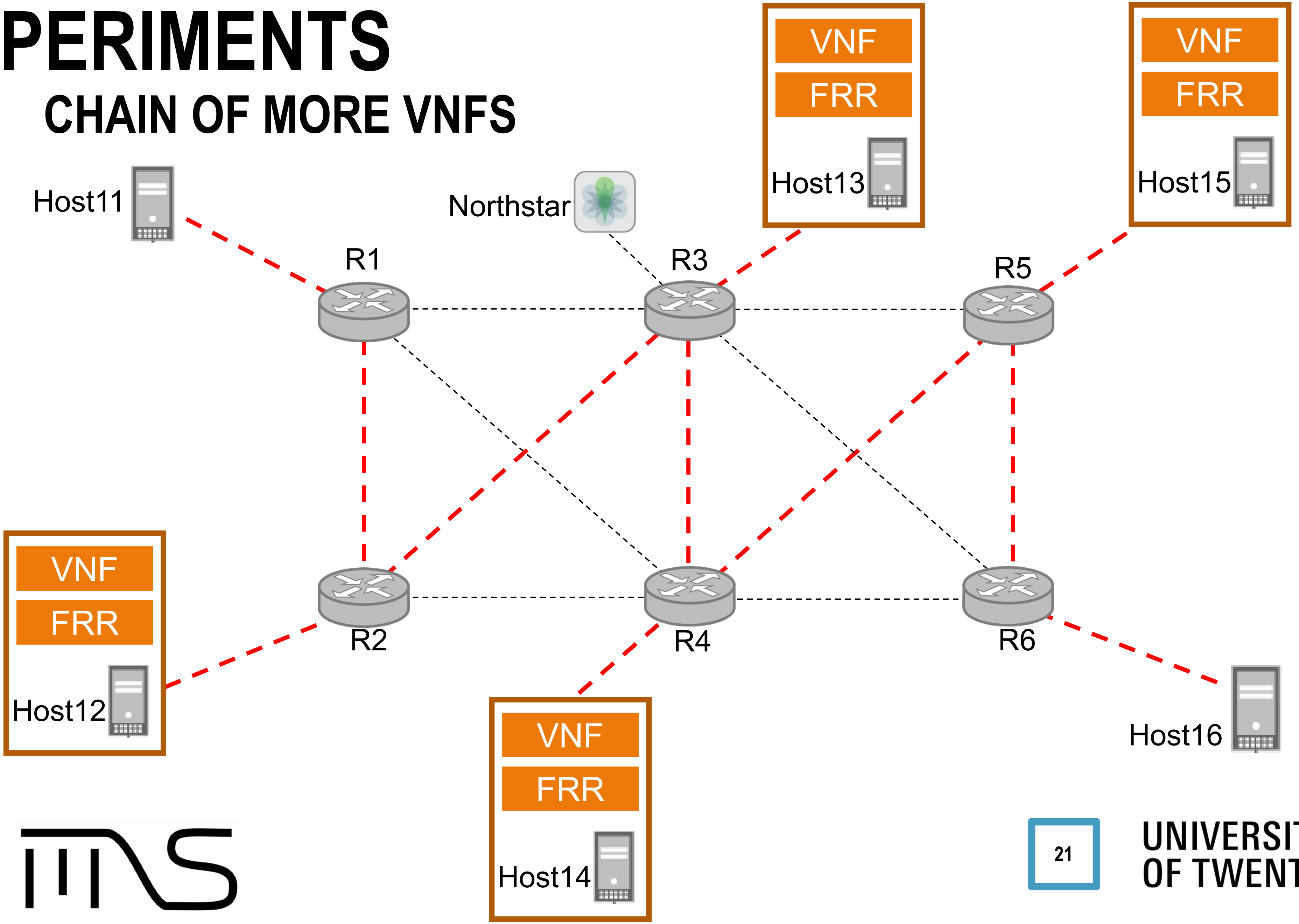
EXPERIMENTS

CHAIN OF MORE VNFS



EXPERIMENTS

CHAIN OF MORE VNFS



EXPERIMENTS

PATH CONTROLLER

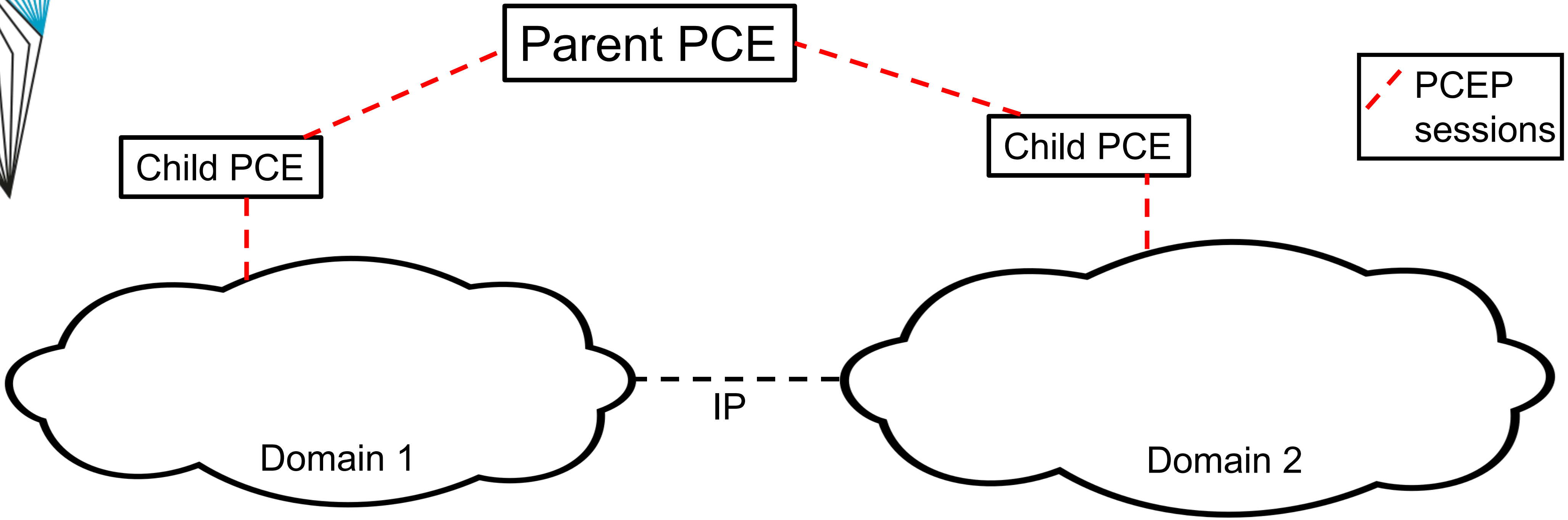
- **SINGLE DOMAIN**
  - Our implementation successfully uses PCEP to assist the network integration of VNFs
  - Re-instantiation and service chain of 2 VNFs could successfully be constructed
  - The main limitation encountered is the strongly varying support for the various IGP SR extensions
  - More interesting implementation of longer VNF chains under revision

- **WHAT ABOUT MULTI DOMAIN?**

EXPERIMENTS

PATH CONTROLLER – MULTI DOMAIN

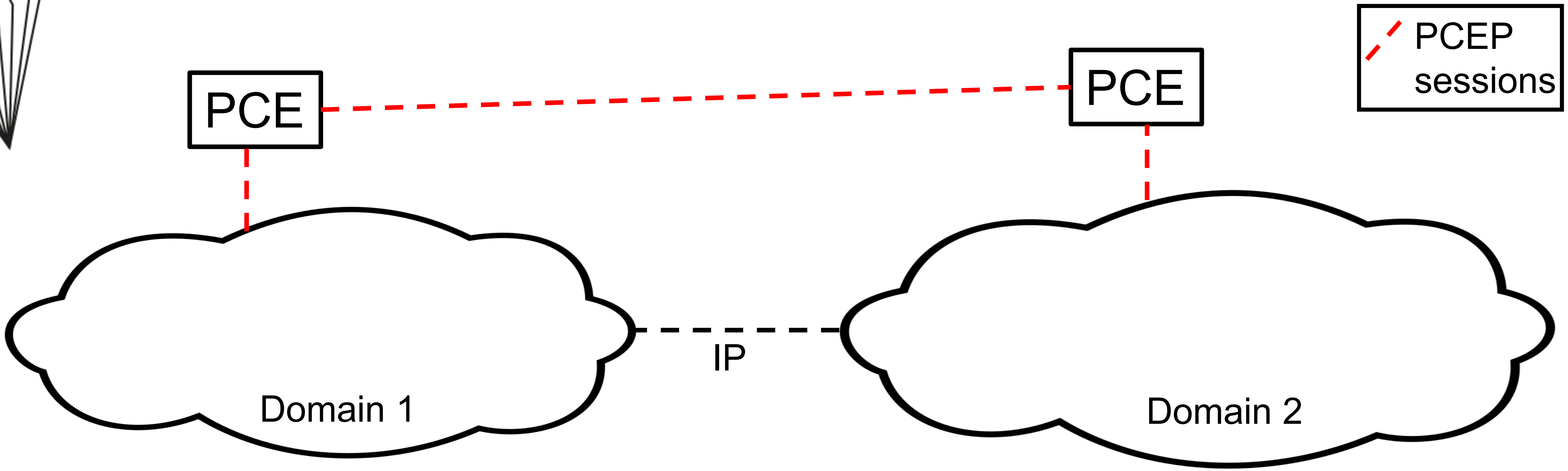
- PCEP can be used to set routes in multi-domain scenarios
- The hierarchy of PCE and PCC needs to change accordingly



EXPERIMENTS

PATH CONTROLLER – MULTI DOMAIN

- PCEP can be used to set routes in multi-domain scenarios
- The hierarchy of PCE and PCC needs to change accordingly



EXPERIMENTS

PATH CONTROLLER – MULTI DOMAIN

- PCEP can be used to set routes in multi-domain scenarios
- The hierarchy of PCE and PCC needs to change accordingly

  - Can we trust the other party to keep the path we instructed?
  - How does the provider get payed for the use of the service?
  - How can we prevent leakage of internal network topology?