

Evolving the End-to-End Transport Layer in Times of Emerging Computing In The Network (COIN)

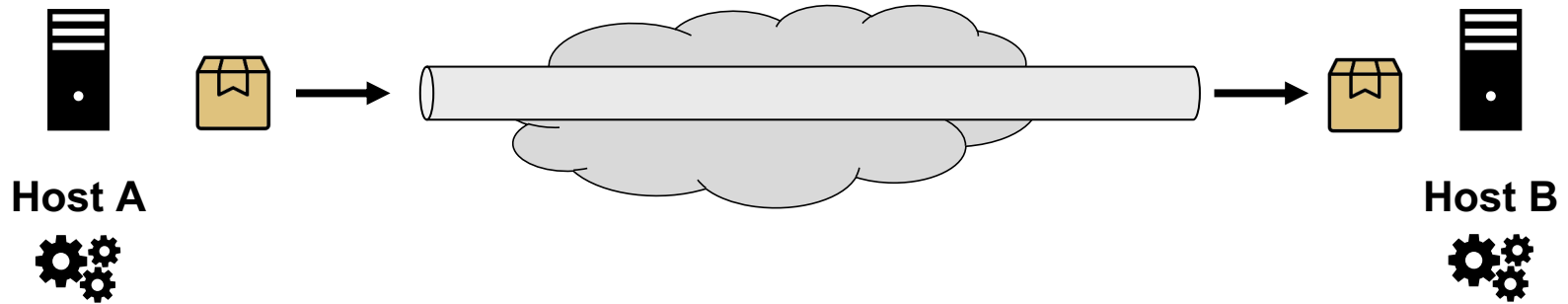
Ike Kunze^{*}, Dirk Trossen[#], Klaus Wehrle^{*}

^{*} RWTH Aachen University [#] Huawei

~~1st Workshop on New IP and Beyond, October 2022~~

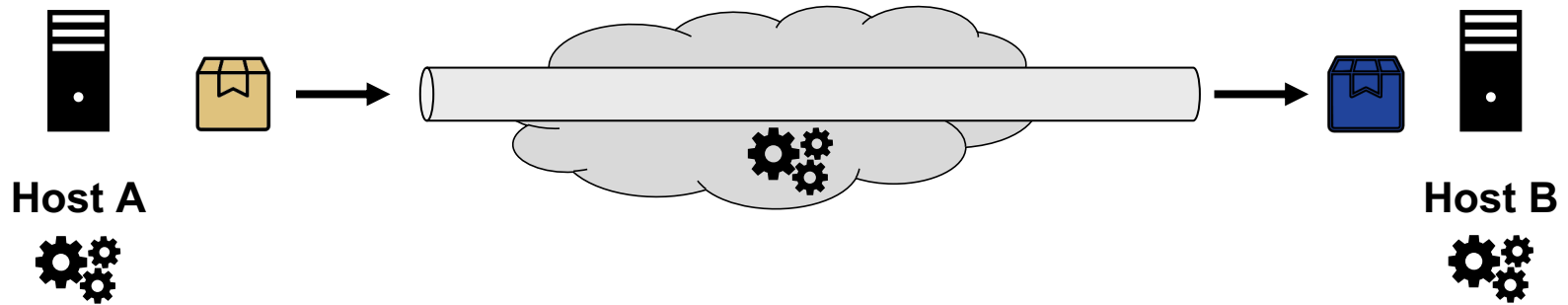
COIN RG @ IETF 115, November 2022

<https://www.comsys.rwth-aachen.de/>



- **End-to-end principle**

- ▶ Network as a dumb pipe
- ▶ Basis for transport protocols



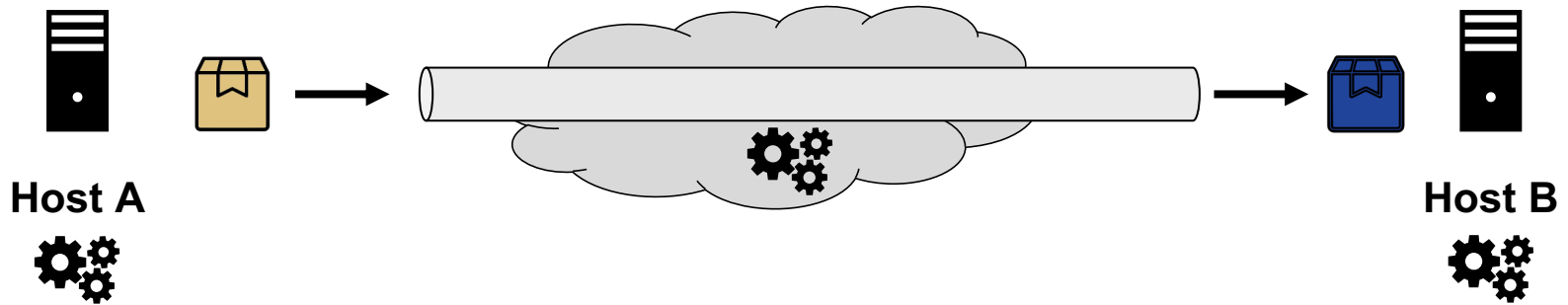
- **End-to-end principle**

- ▶ Network as a dumb pipe
- ▶ Basis for transport protocols

- **COIN**

- ▶ Network no longer a dumb pipe
- ▶ Breaks transport layer assumptions^[1]

[1] Stephens et al.: TCP is Harmful to In-Network Computing: Designing a Message Transport Protocol (MTP), HotNets '21



- **End-to-end principle**

- ▶ Network as a dumb pipe
- ▶ Basis for transport protocols

- **COIN**

- ▶ Network no longer a dumb pipe
- ▶ Breaks transport layer assumptions^[1]

What does it take to have a COIN-enabled transport?

[1] Stephens et al.: TCP is Harmful to In-Network Computing: Designing a Message Transport Protocol (MTP), HotNets '21

- Seminal work by Saltzer et al.^[2]

- ▶ “The **function** [...] can completely and correctly be implemented **only with the knowledge** [...] of the **application** [...] at the **endpoints** [...].”

- Seemingly at odds with COIN

[2] Saltzer et al.: End-To-End Arguments in System Design, *ACM TOCS*, vol. 2, no. 4, 1984

- **Seminal work by Saltzer et al.^[2]**

- ▶ “The **function** [...] can completely and correctly be implemented **only with the knowledge** [...] of the **application** [...] at the **endpoints** [...].”
 - Seemingly at odds with COIN
- ▶ “[...] an **incomplete version** of the function **provided by the communication system** may be useful as a **performance enhancement**.)”
 - Aligns COIN with the end-to-end argument

[2] Saltzer et al.: End-To-End Arguments in System Design, *ACM TOCS*, vol. 2, no. 4, 1984

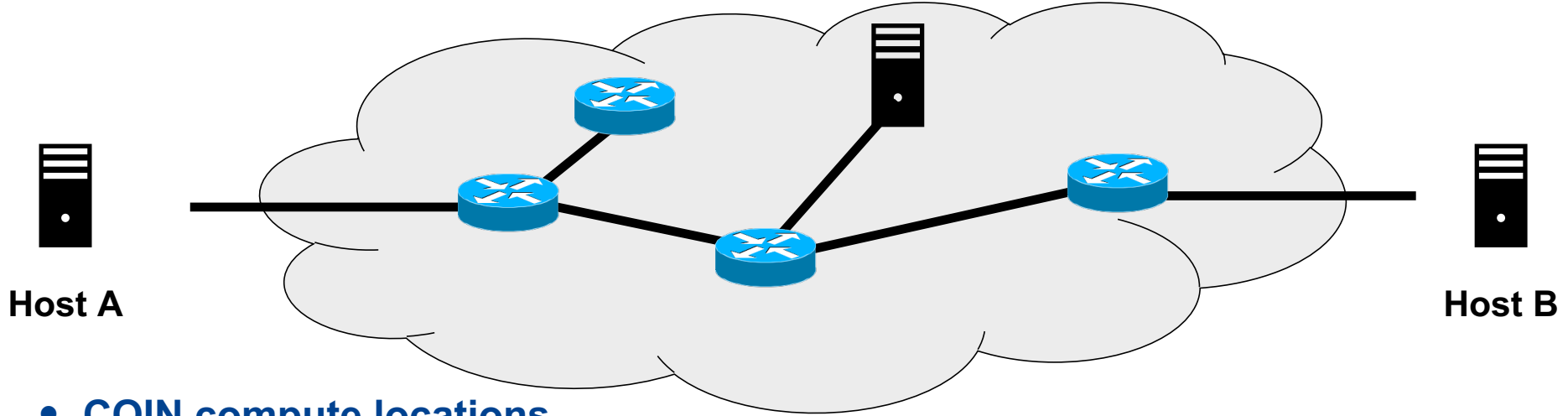
- **Seminal work by Saltzer et al.^[2]**

- ▶ “The **function** [...] can completely and correctly be implemented **only with the knowledge** [...] of the **application** [...] at the **endpoints** [...].”
 - Seemingly at odds with COIN
- ▶ “[...] an **incomplete version** of the function **provided by the communication system** may be useful as a **performance enhancement**.)”
 - Aligns COIN with the end-to-end argument

- **Two critical aspects**

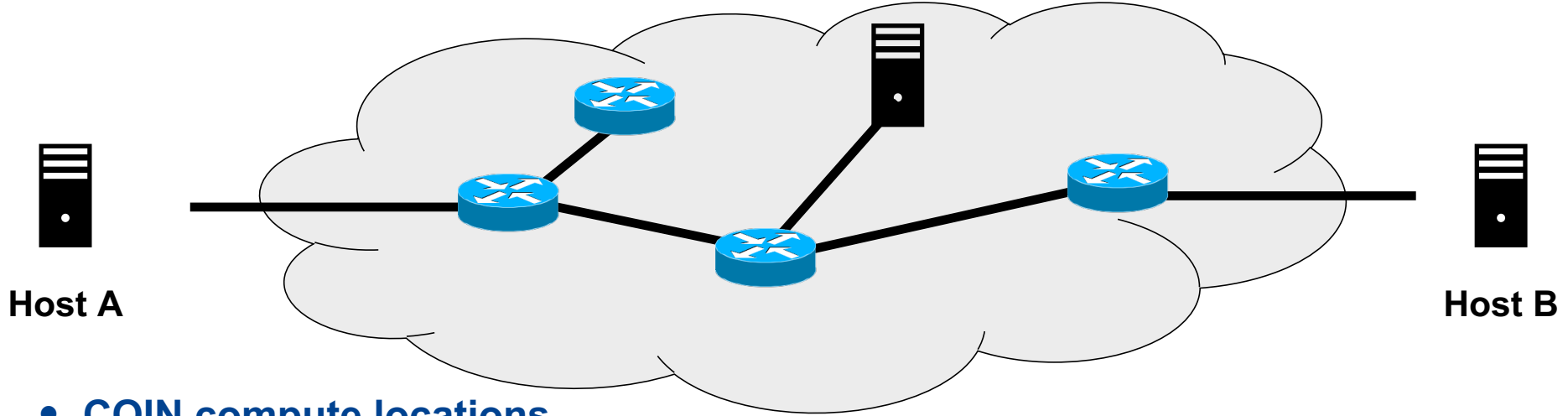
- ▶ Location
- ▶ Which computation?

[2] Saltzer et al.: End-To-End Arguments in System Design, *ACM TOCS*, vol. 2, no. 4, 1984



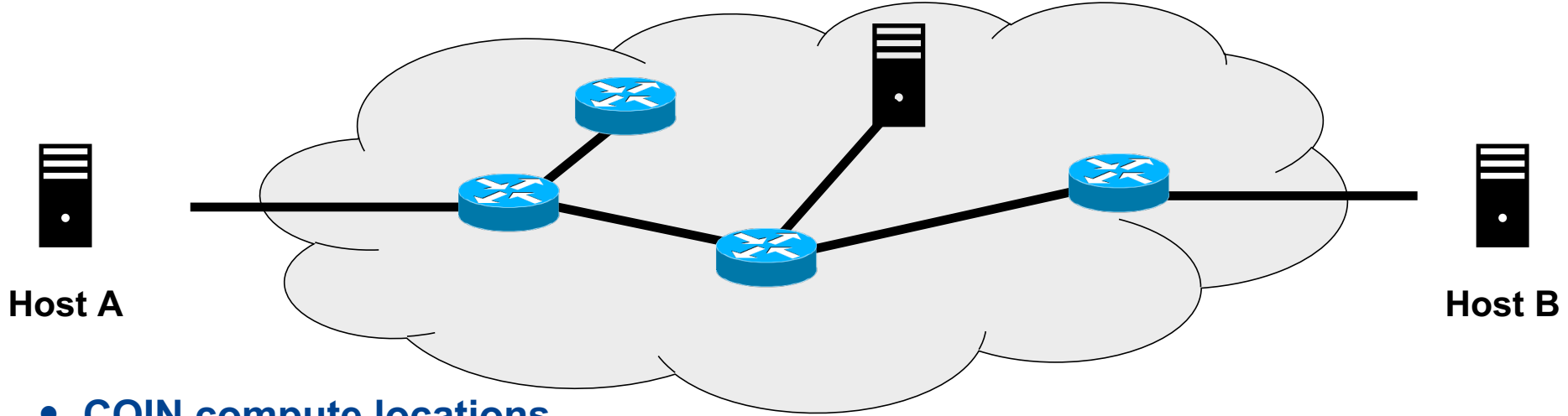
- **COIN compute locations**

- ▶ *Strict* (only networking devices)



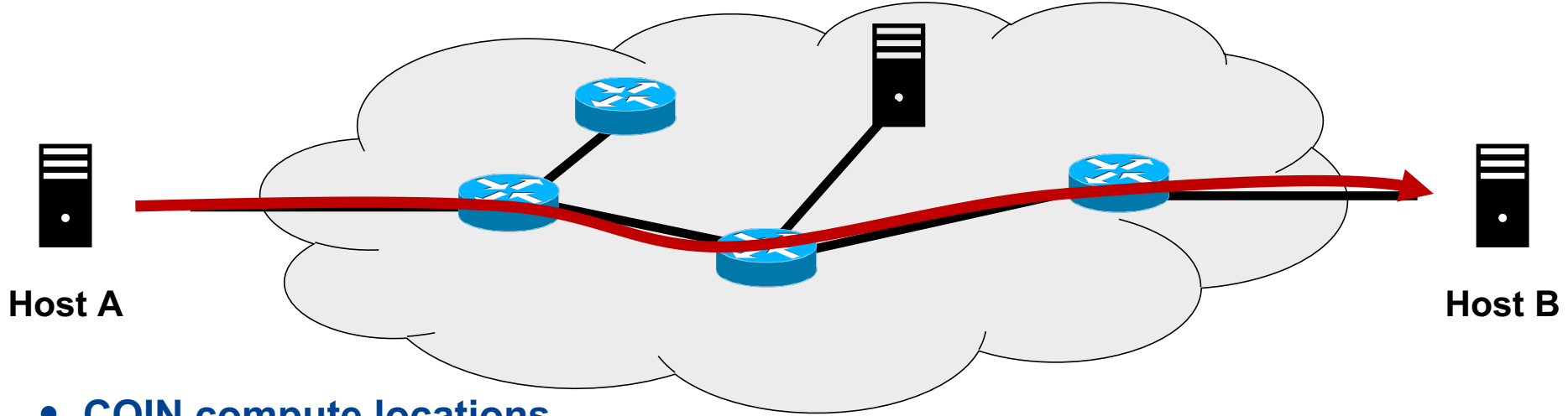
- **COIN compute locations**

- ▶ *Strict* (only networking devices) to *lax* (subset of edge/cloud computing)



- **COIN compute locations**

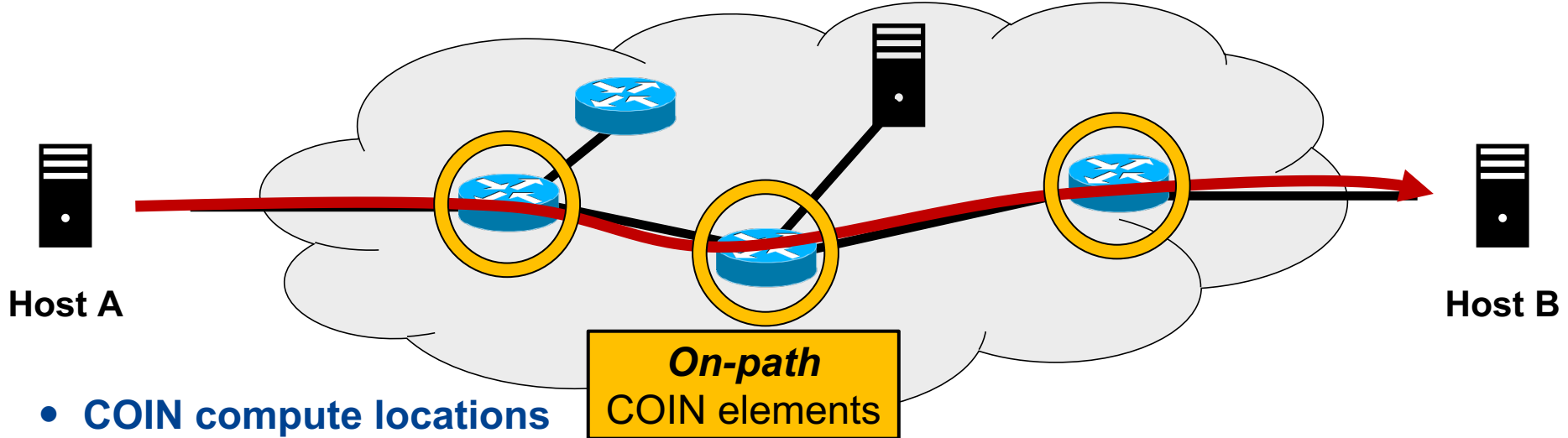
- ▶ *Strict* (only networking devices) to *lax* (subset of edge/cloud computing)
- ▶ Abstraction: *COIN element*



- **COIN compute locations**

- ▶ *Strict* (only networking devices) to *lax* (subset of edge/cloud computing)
- ▶ Abstraction: *COIN element*

- **Location in relation to the endpoints**

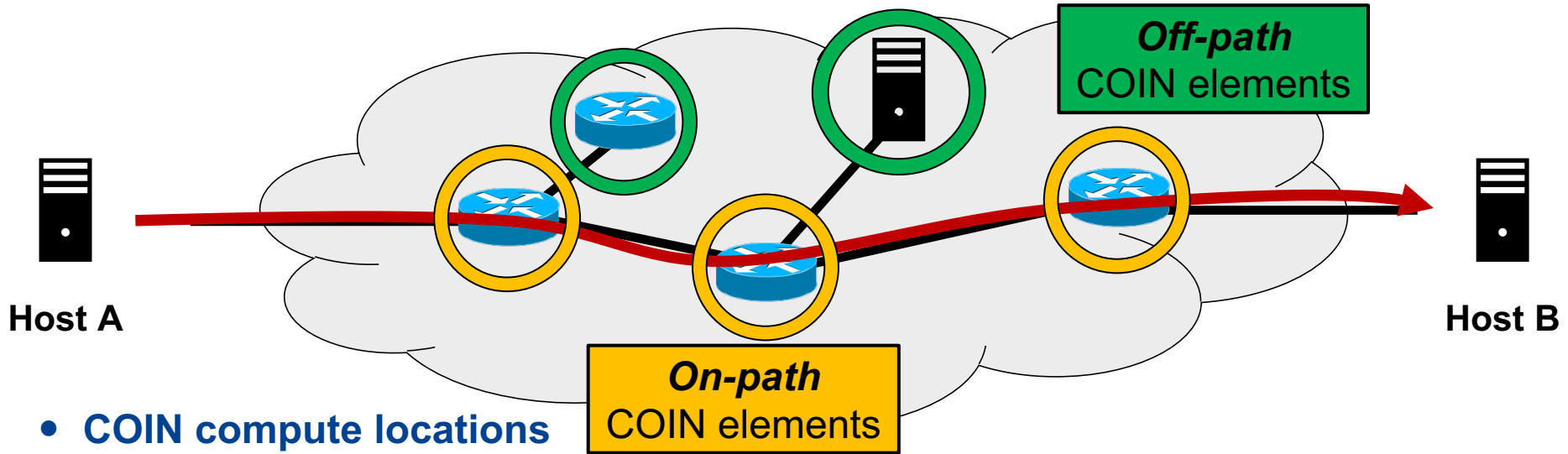


- **COIN compute locations**

- ▶ *Strict* (only networking devices) to *lax* (subset of edge/cloud computing)
- ▶ Abstraction: *COIN element*

- **Location in relation to the endpoints**

- ▶ *On-path*



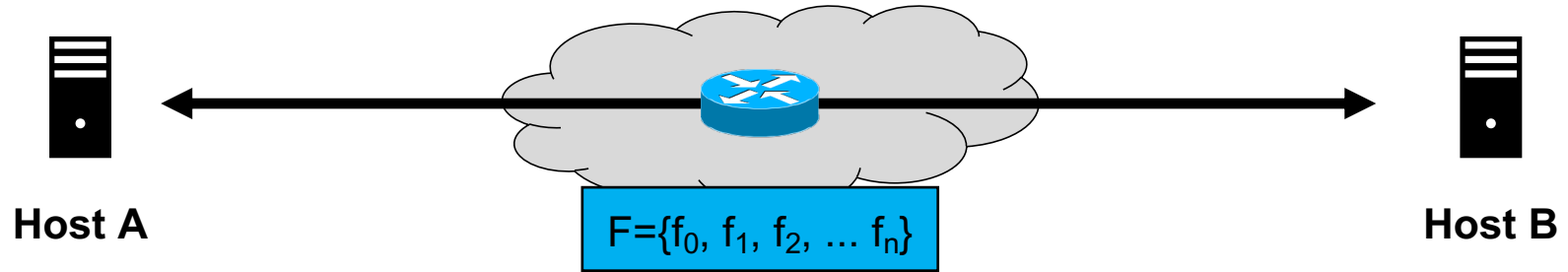
- **COIN compute locations**

- ▶ *Strict* (only networking devices) to *lax* (subset of edge/cloud computing)
- ▶ Abstraction: *COIN element*

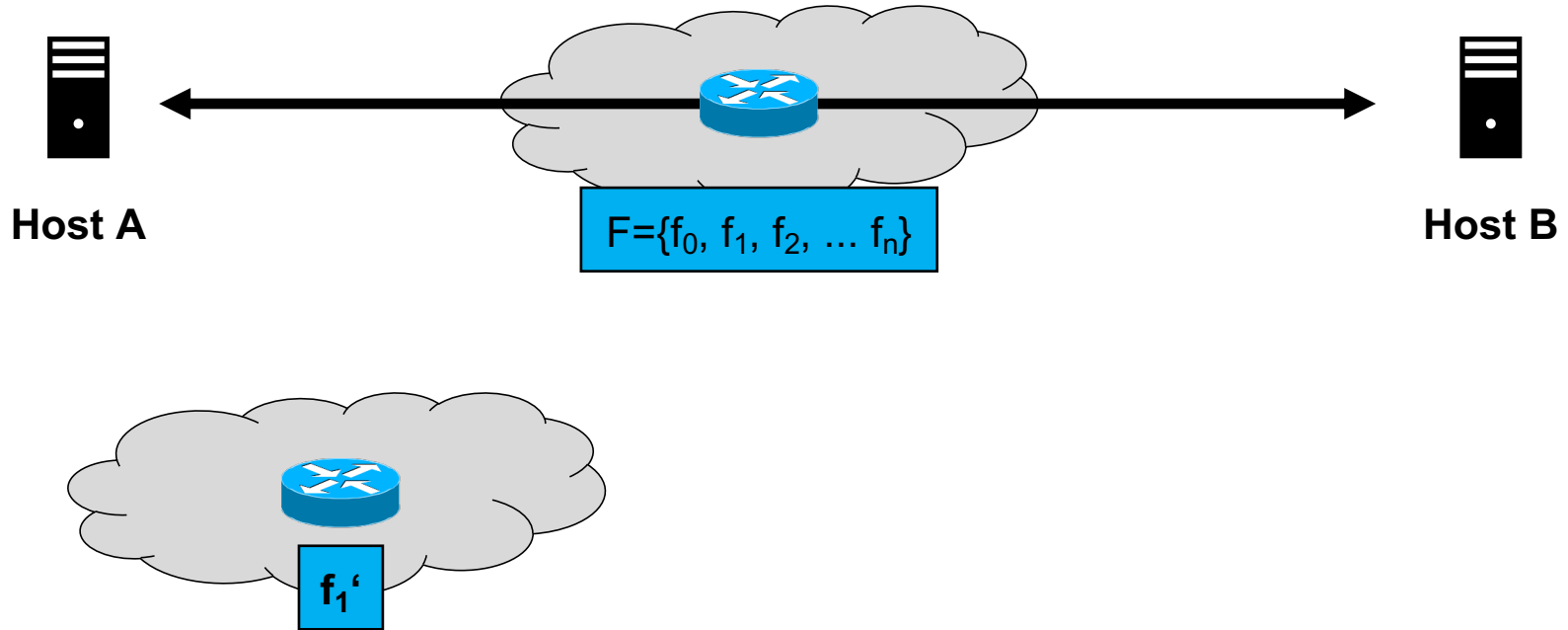
- **Location in relation to the endpoints**

- ▶ *On-path* vs. *off-path*

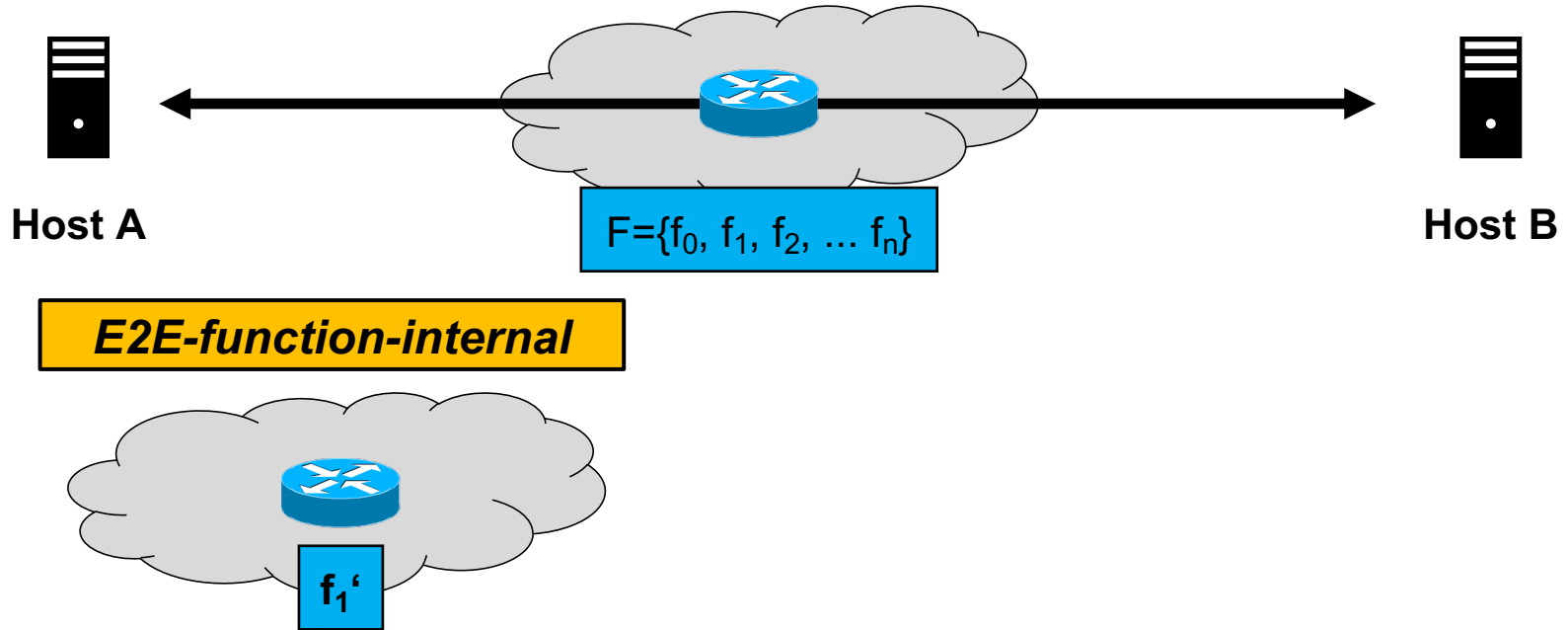
COIN – What kind of functionality?



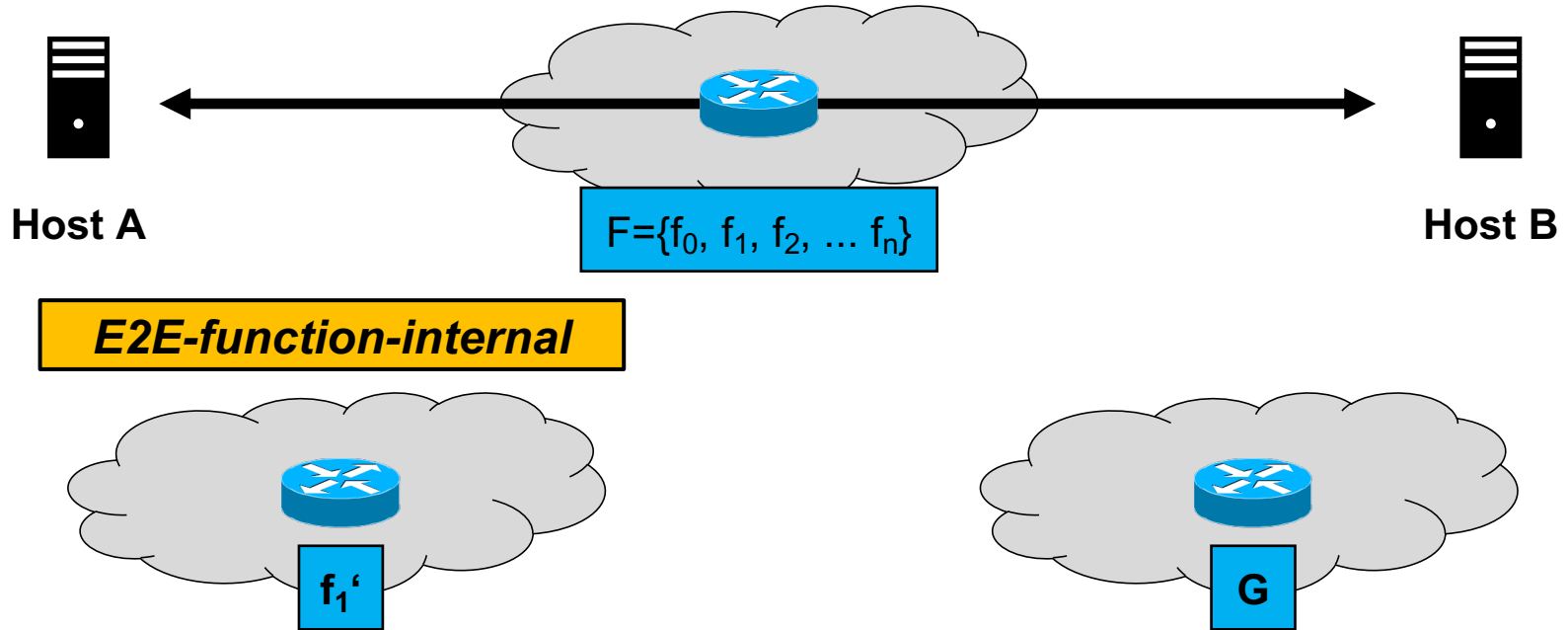
COIN – What kind of functionality?



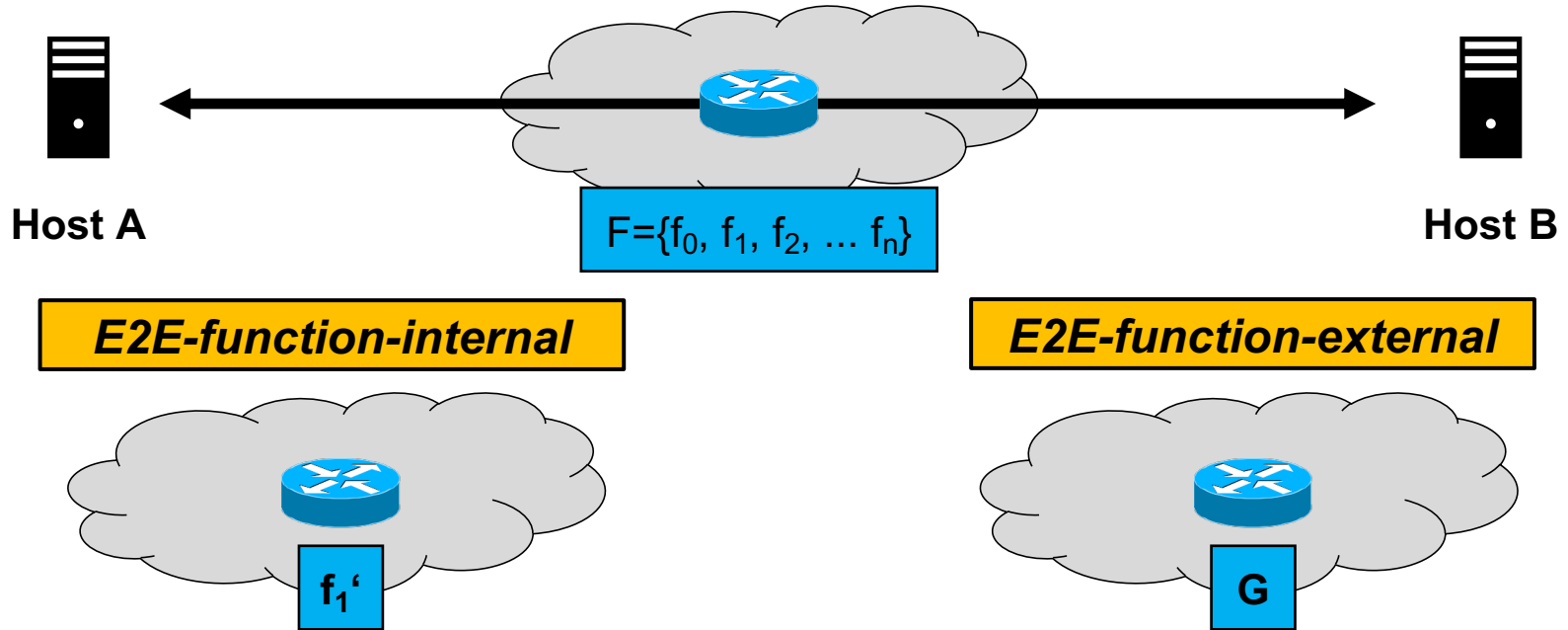
COIN – What kind of functionality?



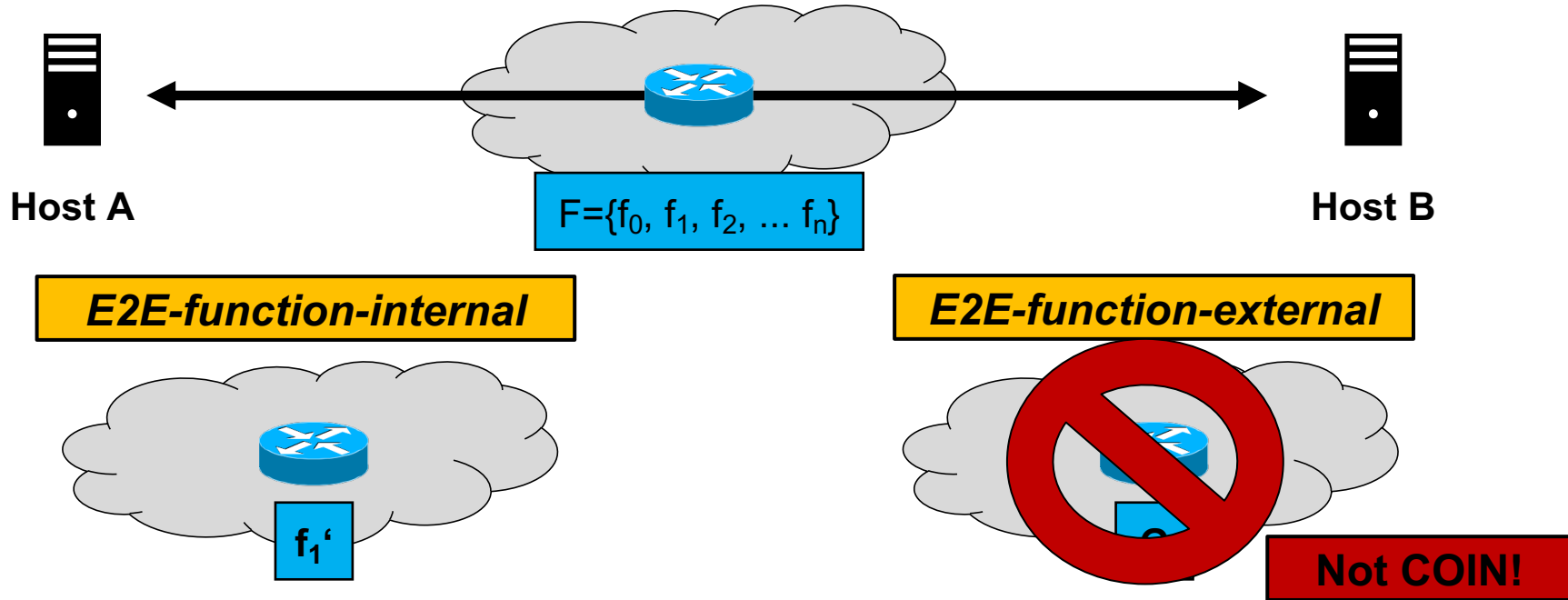
COIN – What kind of functionality?



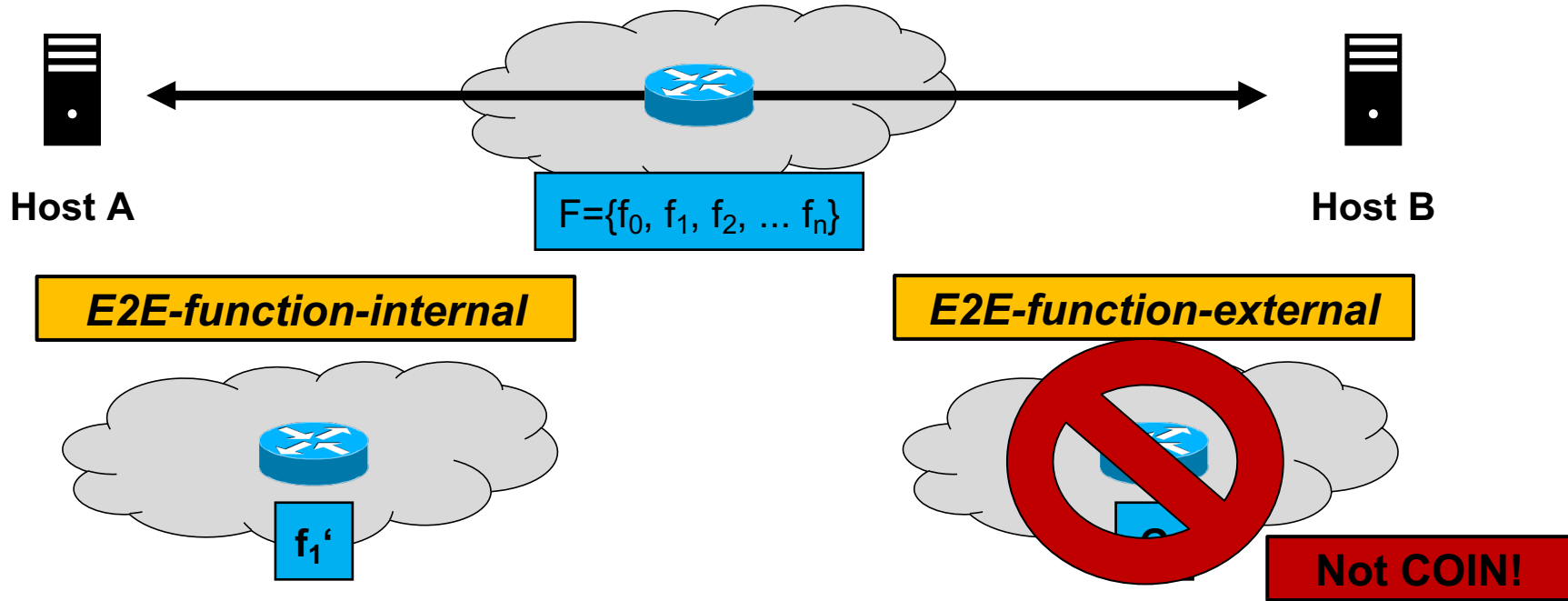
COIN – What kind of functionality?



COIN – What kind of functionality?

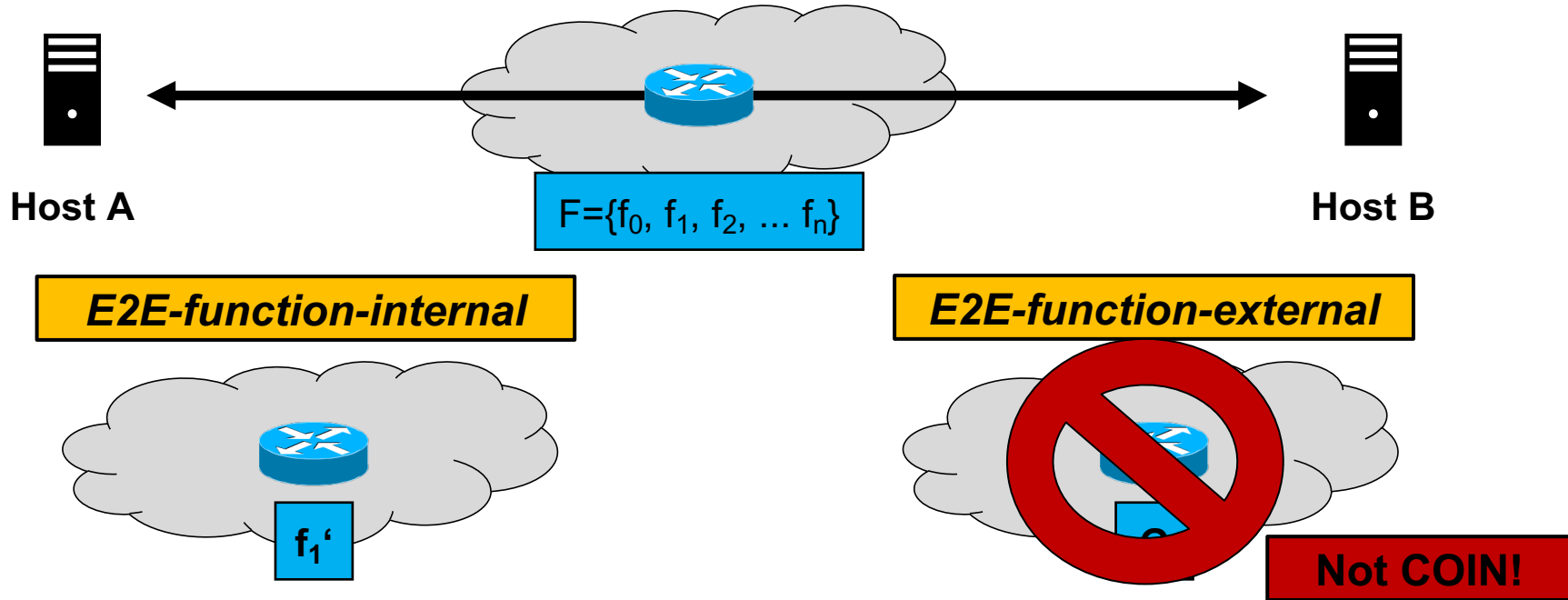


COIN – What kind of functionality?



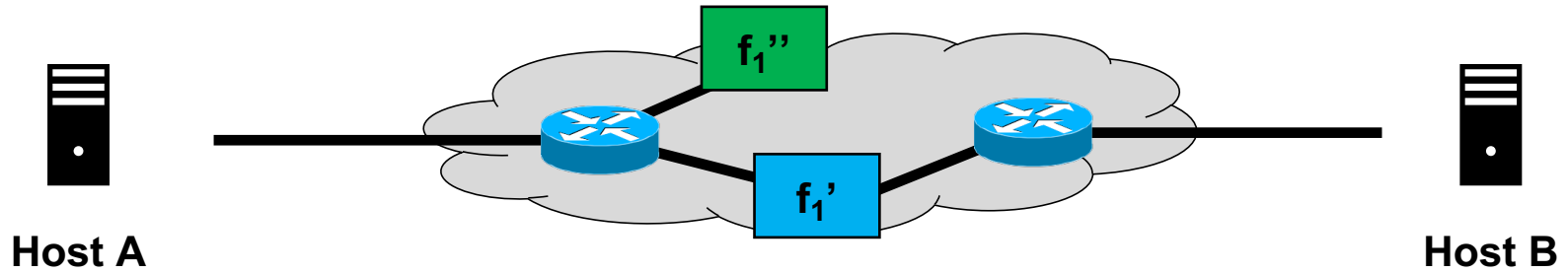
- *E2E-function-internal* computation on **on- & off-path COIN elements**

COIN – What kind of functionality?

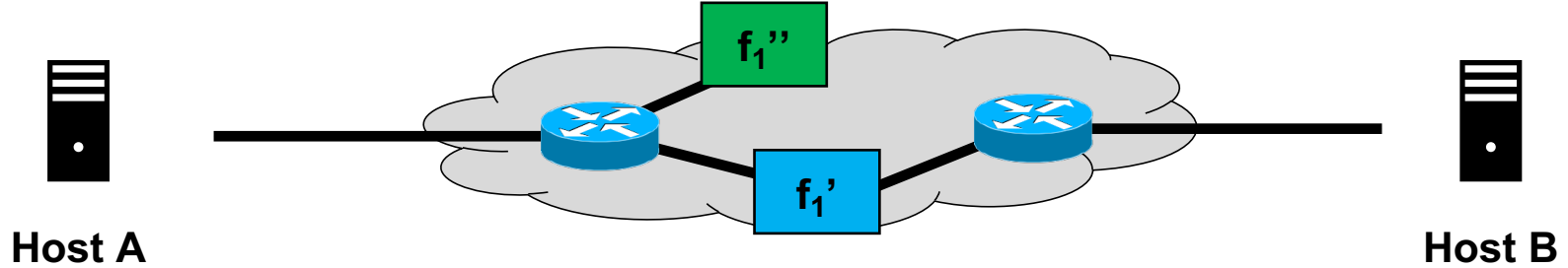


- *E2E-function-internal* computation on **on- & off-path COIN** elements
- How and where to place functionality?

First Design Principle for COIN



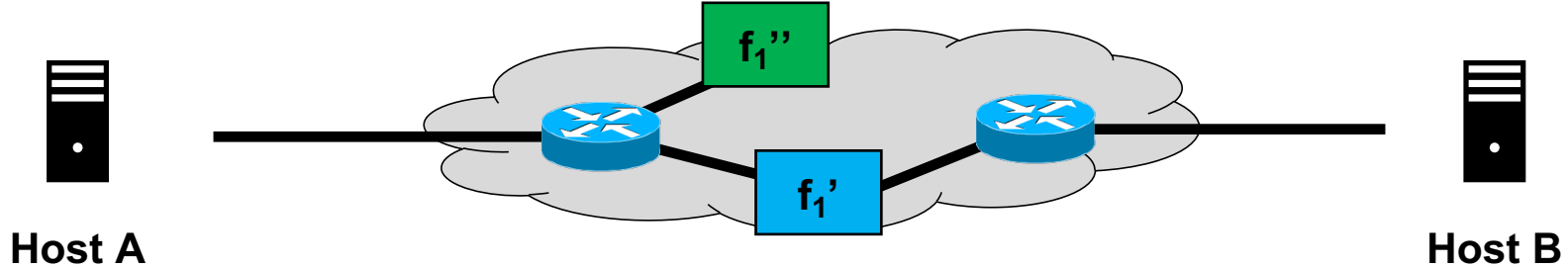
First Design Principle for COIN



- ***First COIN Design Principle***

- ▶ Adhere to the ***original requirements*** of the E2E function,
- ▶ ***Enrich its functionality***

First Design Principle for COIN



- **First COIN Design Principle**

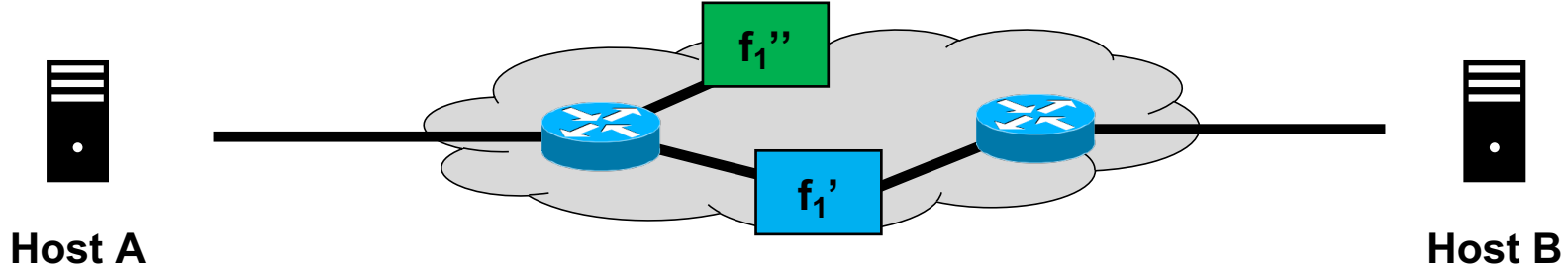
- ▶ Adhere to the **original requirements** of the E2E function,
- ▶ **Enrich its functionality**

- **Simplicity Principle^[3]**

- ▶ “[...] we must drive our architectures and designs toward the simplest possible solutions.”

[3] Bush and Meyer: RFC 3439

First Design Principle for COIN



- **First COIN Design Principle**

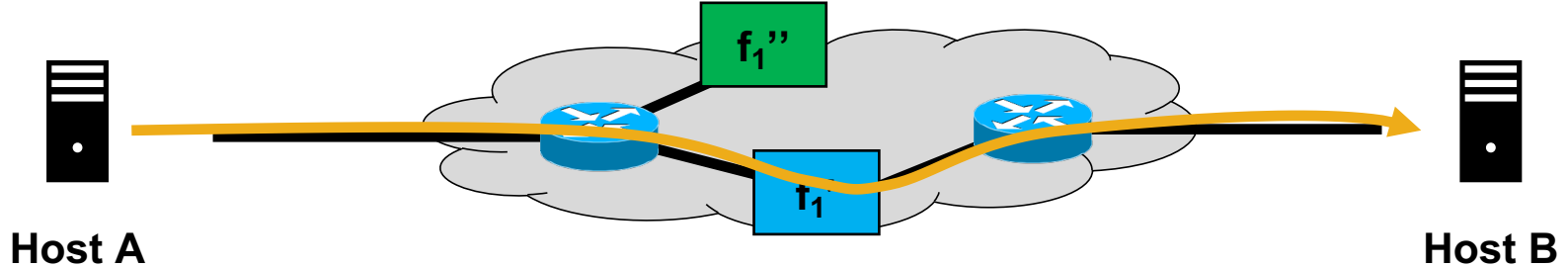
- ▶ Adhere to the **original requirements** of the E2E function,
- ▶ **Enrich its functionality**
- ▶ **Optimize functional complexity** against its **key communication requirement**

- **Simplicity Principle^[3]**

- ▶ “[...] we must drive our architectures and designs toward the simplest possible solutions.”

[3] Bush and Meyer: RFC 3439

First Design Principle for COIN



- **First COIN Design Principle**

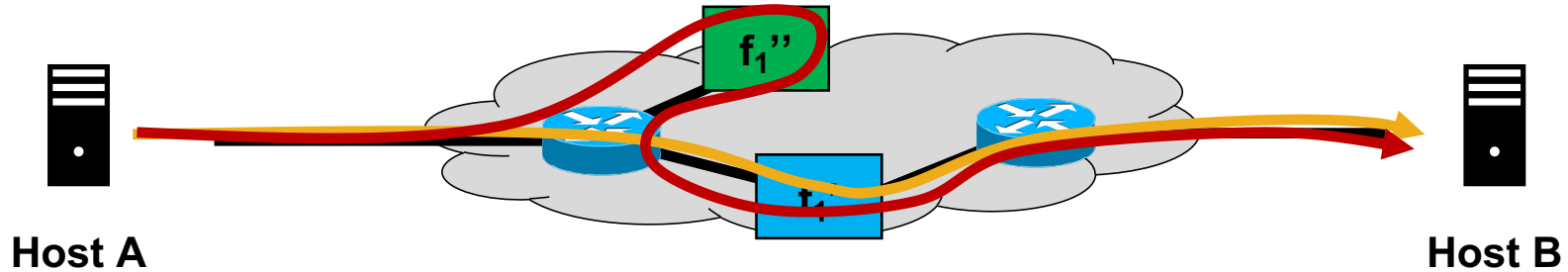
- ▶ Adhere to the **original requirements** of the E2E function,
- ▶ **Enrich its functionality**
- ▶ **Optimize functional complexity** against its **key communication requirement**

- **Simplicity Principle^[3]**

- ▶ “[...] we must drive our architectures and designs toward the simplest possible solutions.”

[3] Bush and Meyer: RFC 3439

First Design Principle for COIN



- **First COIN Design Principle**

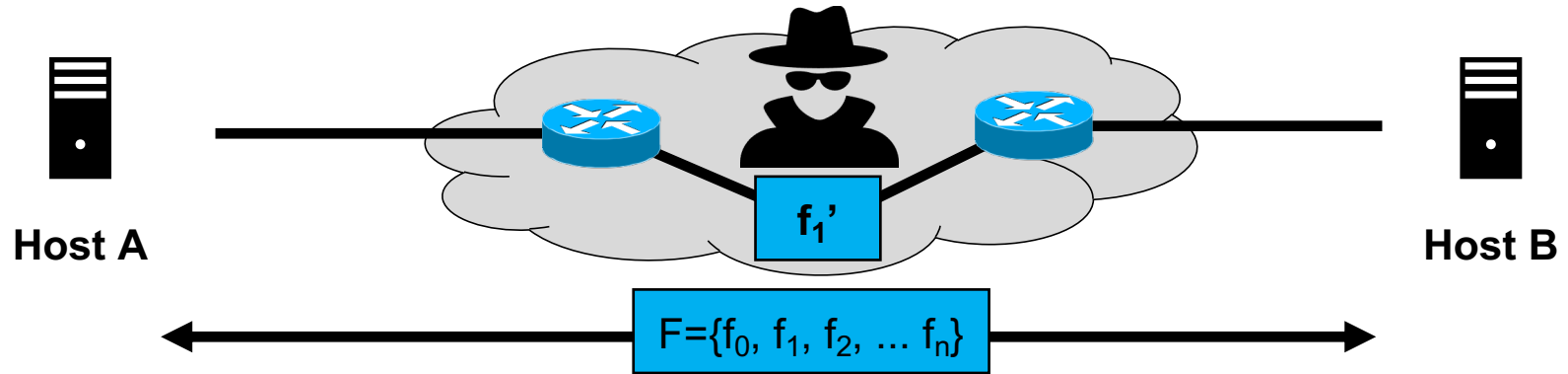
- ▶ Adhere to the **original requirements** of the E2E function,
- ▶ **Enrich its functionality**
- ▶ **Optimize functional complexity** against its **key communication requirement**

- **Simplicity Principle^[3]**

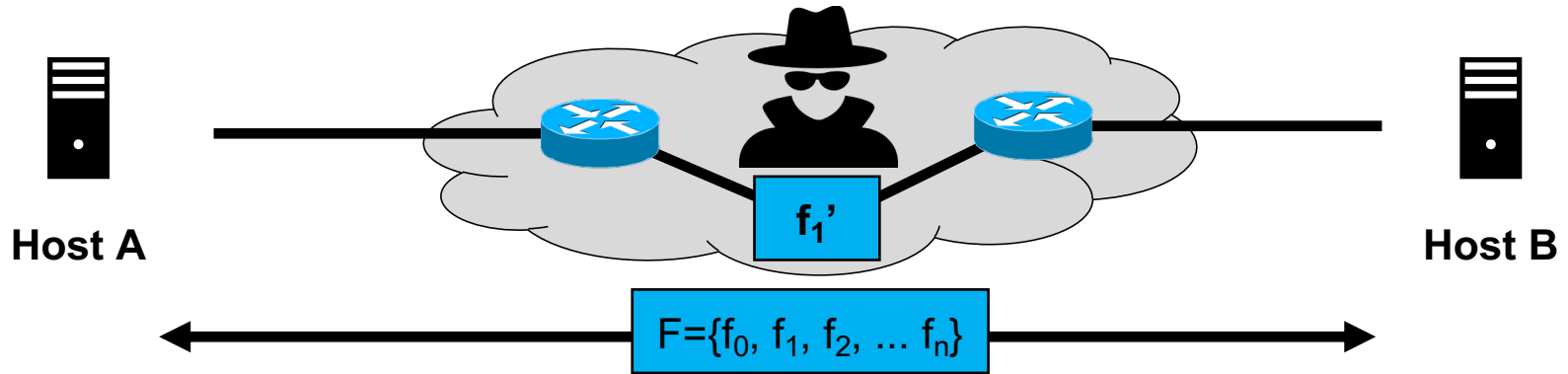
- ▶ “[...] we must drive our architectures and designs toward the simplest possible solutions.”

[3] Bush and Meyer: RFC 3439

Second Design Principle for COIN

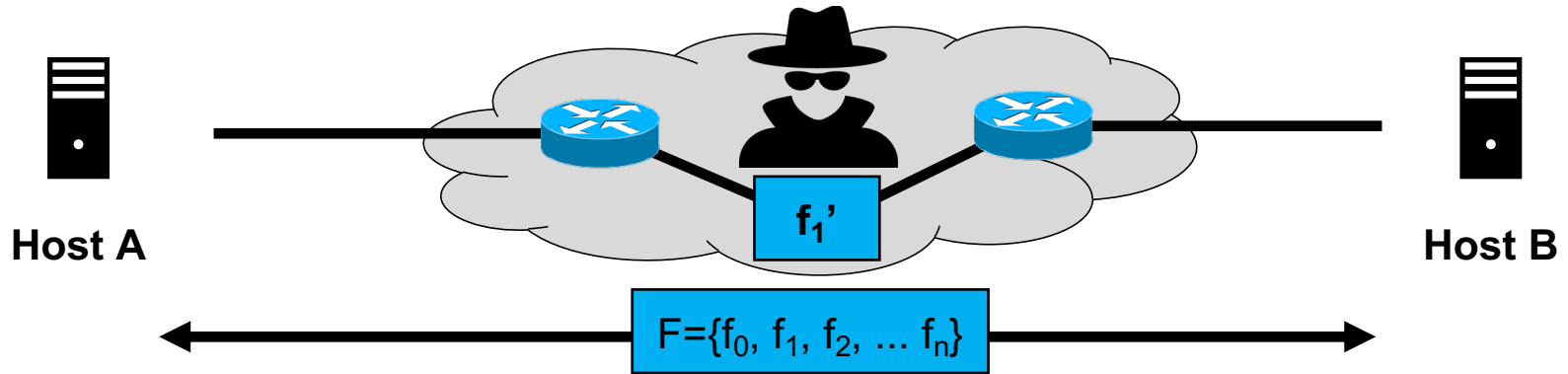


Second Design Principle for COIN



- Missing transparency might lead to problems

Second Design Principle for COIN

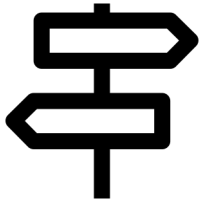


- Missing transparency might lead to problems
- *Second COIN Design Principle*
 - ▶ Insert in **full transparency** to the endpoints

- **Functionality traditionally implemented in endpoints only**
- **Directly impacted by emerging COIN capabilities**

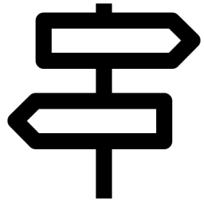
- **Functionality traditionally implemented in endpoints only**
- **Directly impacted by emerging COIN capabilities**
- **Several aspects to consider**

Adressing



- **Functionality traditionally implemented in endpoints only**
- **Directly impacted by emerging COIN capabilities**
- **Several aspects to consider**

Adressing

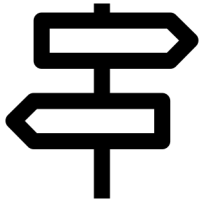


Flow Granularity



- **Functionality traditionally implemented in endpoints only**
- **Directly impacted by emerging COIN capabilities**
- **Several aspects to consider**

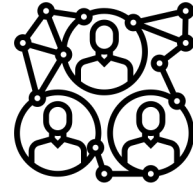
Adressing



Flow Granularity



Collective Communication

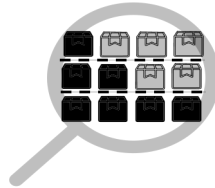


- **Functionality traditionally implemented in endpoints only**
- **Directly impacted by emerging COIN capabilities**
- **Several aspects to consider**

Adressing



Flow Granularity



Collective Communication

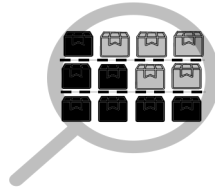


- **Functionality traditionally implemented in endpoints only**
- **Directly impacted by emerging COIN capabilities**
- **Several aspects to consider**

Adressing



Flow Granularity



Collective Communication

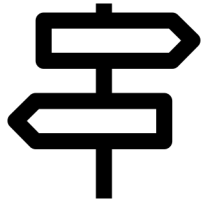


- **More details**



- **Functionality traditionally implemented in endpoints only**
- **Directly impacted by emerging COIN capabilities**
- **Several aspects to consider**

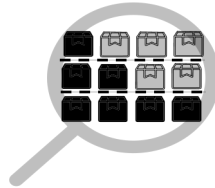
Adressing



- **More details**



Flow Granularity



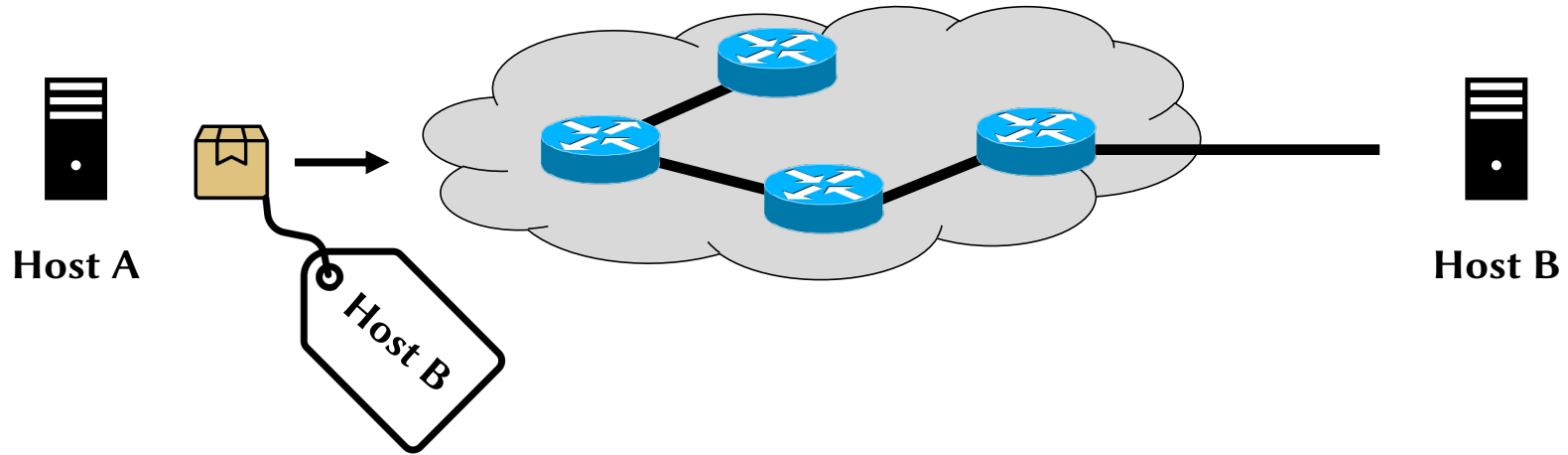
- **More aspects**



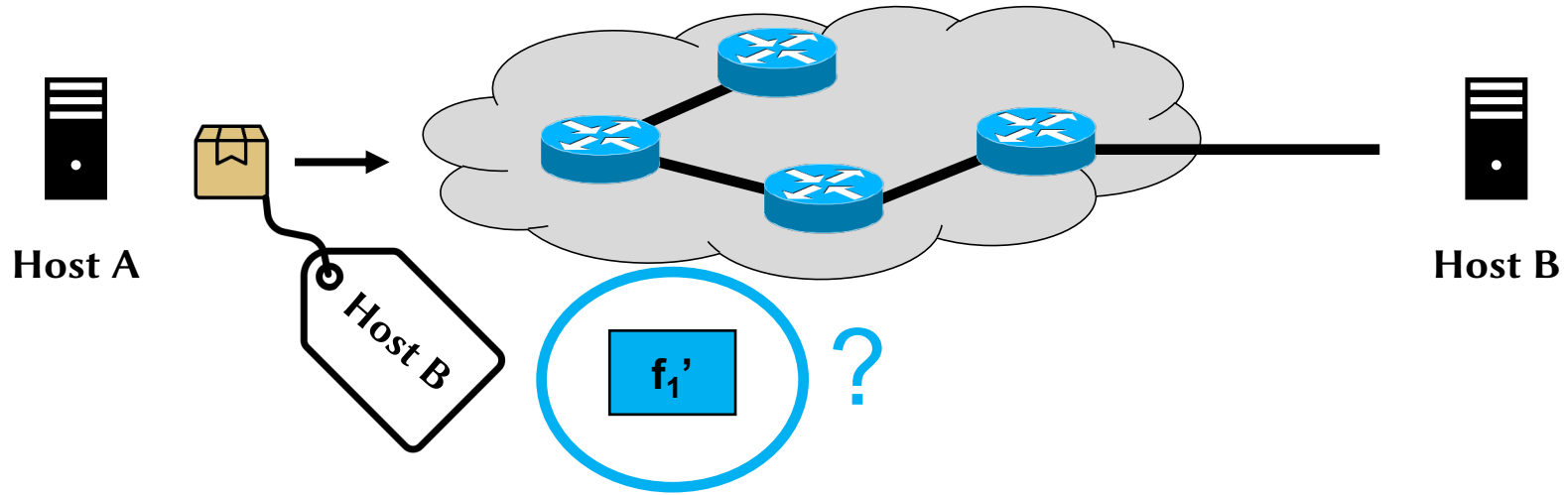
Collective Communication

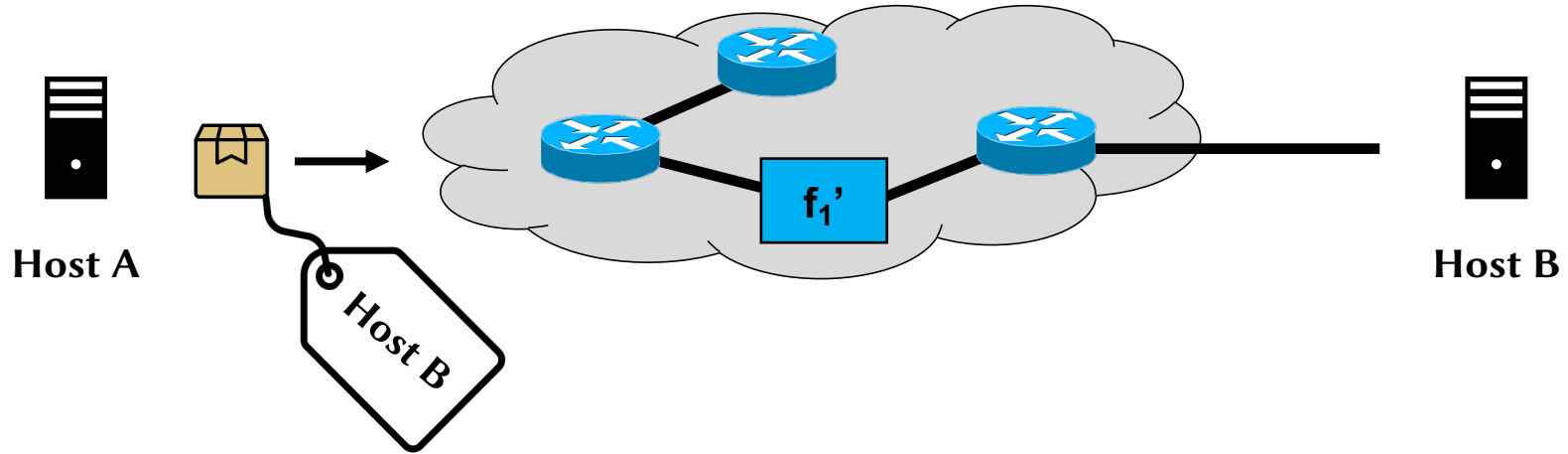


Transport Addressing (1/2)



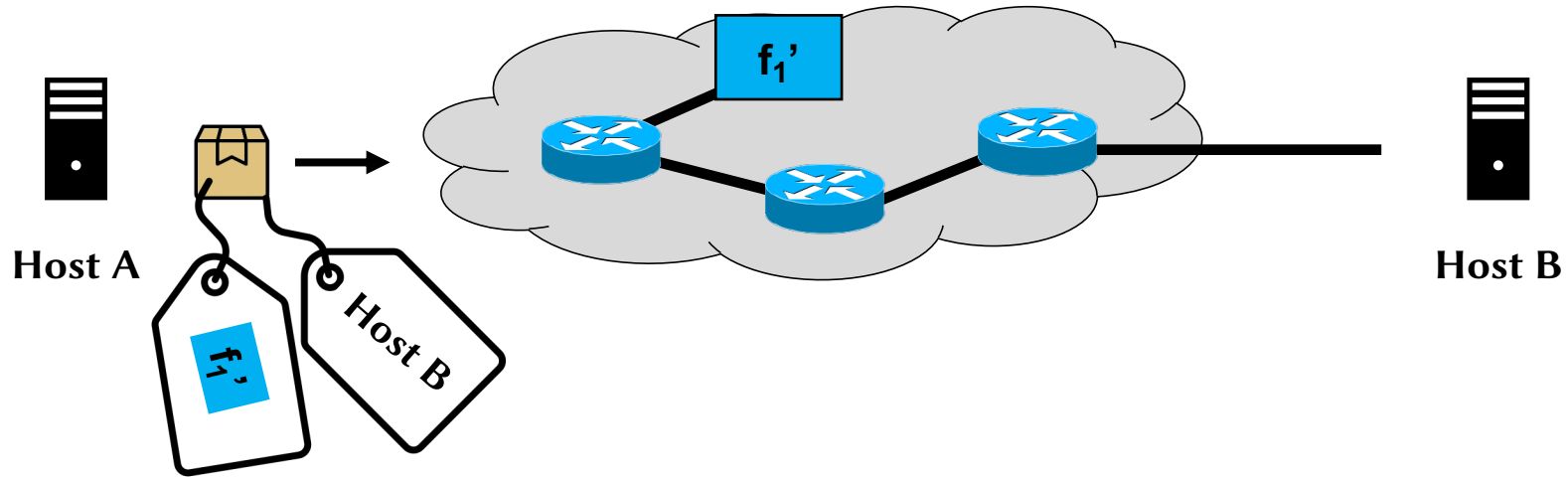
Transport Addressing (1/2)





- **Implicit integration**

- ▶ Hard to maintain
- ▶ Only on-path notion



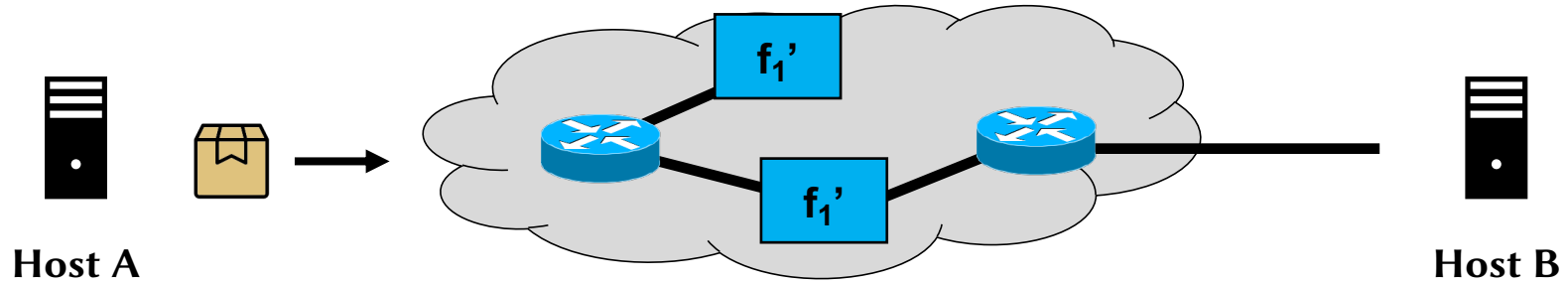
- **Implicit integration**

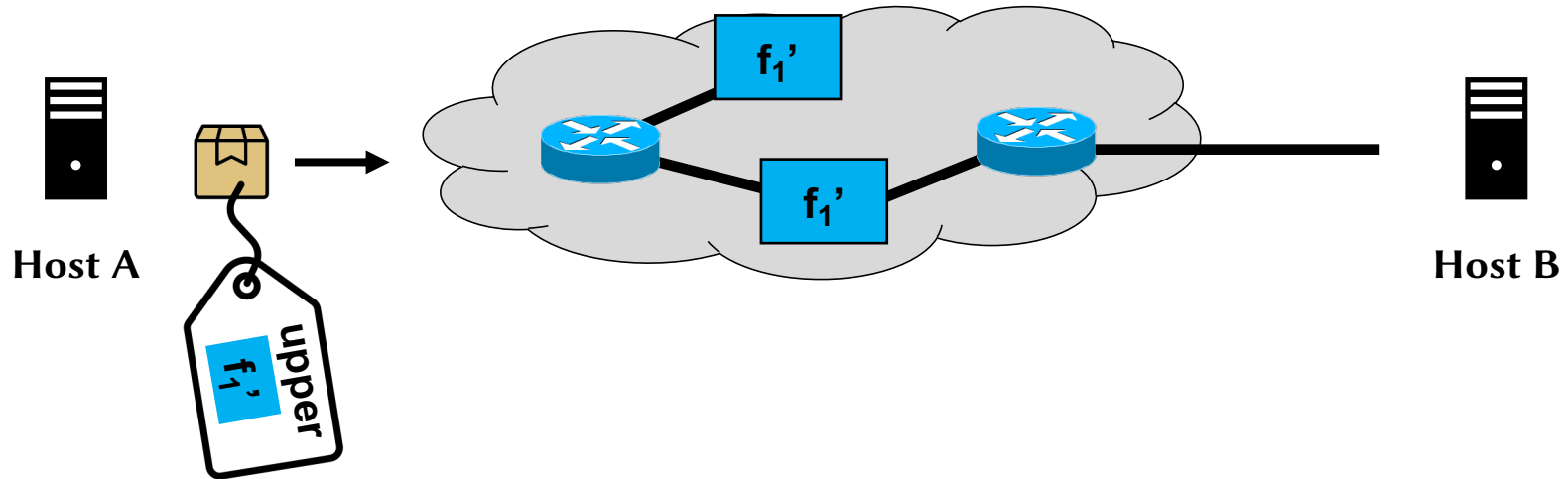
- ▶ Hard to maintain
- ▶ Only on-path notion

- **Explicit steering**

- ▶ Explicitly address
- ▶ How to do the addressing?

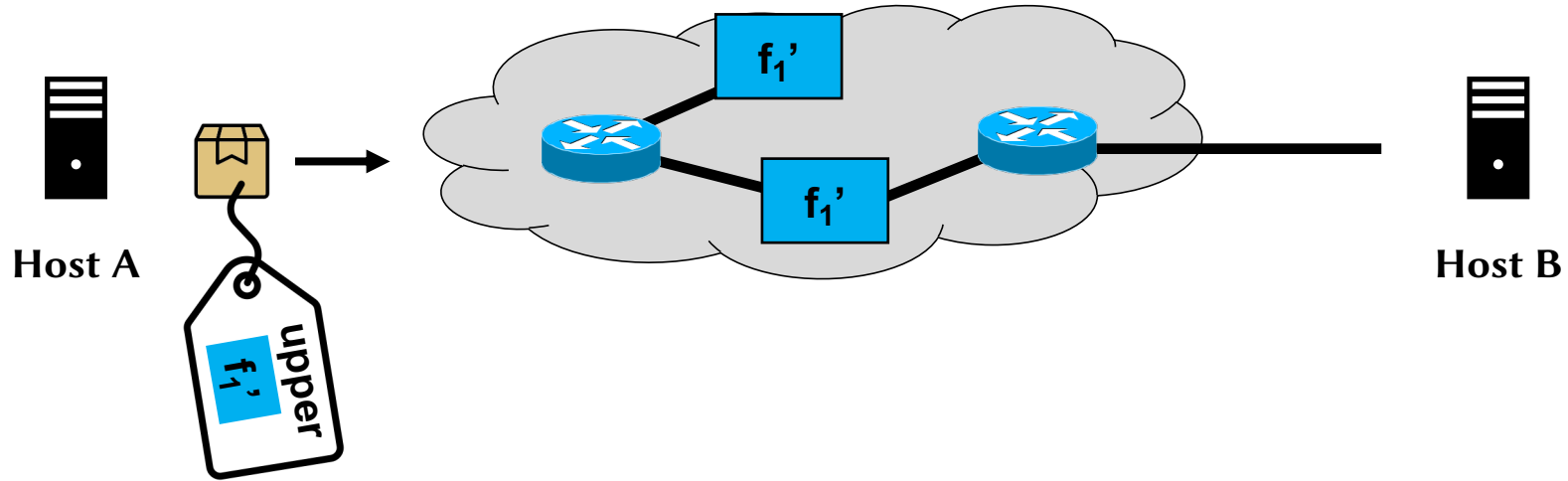
Transport Addressing (2/2)





- **Instance selection**

- ▶ Multiple possible locations
- ▶ How to select them?
 - Specify constraints?
 - Specify locator?



- **Instance selection**

- ▶ Multiple possible locations
- ▶ How to select them?
 - Specify constraints?
 - Specify locator?

- **Affinity**

- ▶ Might be needed
- ▶ How to realize that?
 - Setup during “orchestration”?
 - Route based on service identifier?

- **Transport Addressing**

- ▶ Source routing

- Only which path not which functionality

- **Transport Addressing**

- ▶ Source routing
 - Only which path not which functionality
- ▶ Service function chaining
 - Steer traffic through functions

- **Transport Addressing**

- ▶ Source routing
 - Only which path not which functionality
- ▶ Service function chaining
 - Steer traffic through functions
- ▶ Information centric networking
 - Address information rather than endpoints

- **Transport Addressing**

- ▶ Source routing
 - Only which path not which functionality
- ▶ Service function chaining
 - Steer traffic through functions
- ▶ Information centric networking
 - Address information rather than endpoints

- **Many possible solutions and open questions**

- **Main takeaway of this paper: COIN can align with end-to-end principle**

- **What should be our goal?**

- ▶ Have one global E2E-compliant transport protocol for everything COIN-related?

- MTP^[1]

- TinTin^[4]

[1] Stephens et al.: TCP is Harmful to In-Network Computing: Designing a Message Transport Protocol (MTP), HotNets '21

[4] Makhijani et al.: TinTin: Tiny In-Network Transport for High Precision Industrial Communication, NIB '22

- **What should be our goal?**

- ▶ Have one global E2E-compliant transport protocol for everything COIN-related?
 - MTP^[1]
 - TinTin^[4]
- ▶ Specialized transport protocols for limited domains?
 - Required core features?
 - Standardized interaction?

[1] Stephens et al.: TCP is Harmful to In-Network Computing: Designing a Message Transport Protocol (MTP), HotNets '21

[4] Makhijani et al.: TinTin: Tiny In-Network Transport for High Precision Industrial Communication, NIB '21

- **What should be our goal?**

- ▶ Have one global E2E-compliant transport protocol for everything COIN-related?
 - MTP^[1]
 - TinTin^[4]
- ▶ Specialized transport protocols for limited domains?
 - Required core features
 - Standardized interaction?
- ▶ Abandon E2E principle for the limited domains?
 - Industrial networks

[1] Stephens et al.: TCP is Harmful to In-Network Computing: Designing a Message Transport Protocol (MTP), HotNets '21

[4] Makhijani et al.: TinTin: Tiny In-Network Transport for High Precision Industrial Communication, NIB '21

- **COIN seemingly at odds with end-to-end principle**

- **COIN seemingly at odds with end-to-end principle**
- **E2E-function-internal COIN computations align**
 - ▶ On- vs. off-path COIN elements

- **COIN seemingly at odds with end-to-end principle**
- **E2E-function-internal COIN computations align**
 - ▶ On- vs. off-path COIN elements
- **Two design principles for COIN**

- COIN seemingly at odds with end-to-end principle
- E2E-function-internal COIN computations align
 - ▶ On- vs. off-path COIN elements
- Two design principles for COIN
- Transport considerations



- COIN seemingly at odds with end-to-end principle
- E2E-function-internal COIN computations align
 - ▶ On- vs. off-path COIN elements
- Two design principles for COIN
- Transport considerations



Paper



Draft

