IETF 100 Singapore - Nov 14, 2017

Optimized Service Function Chaining

draft-khalili-optimized-service-function-chaining-00

R. Khalili*, Z. Despotovic, A. Hecker, Huawei ERC, Germany

D. Purkayastha, A. Rahman, D. Trossen, InterDigital Communications

* *Corresponding author:* ramin.khalili@huawei.com

Motivation

- **Investigate the possibility of using ’transport derived service function forwarders’ (tSFFs)**
  - Using existing transport information for service path information
  - Optional use of the SFC encapsulation in the SFF
  - Can be optimized to reduce initial request latency

- **Two solutions are proposed**
  - SDN-based solution using aggregated flow identifier [Khalili2016]
  - Using service request routing (SRR) function [Purka2017]

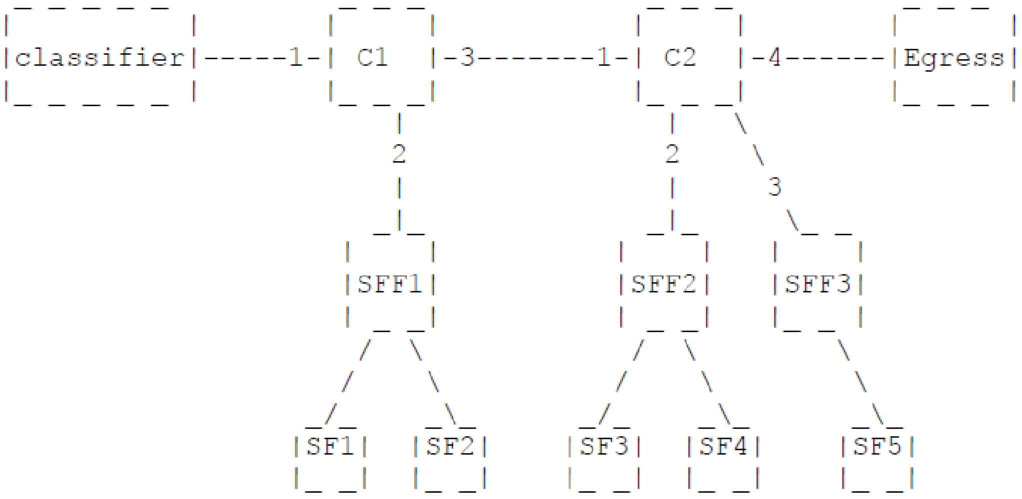
SDN based solution

- **Edge classification and network forwarding aggregation:**
  - Assign network locators (IDs) to edge nodes
  - Routers/switches can forward based on wildcard matching on bits of these IDs
- **Pre-configuration of all the paths between edges: the network operates as a fabric connecting edges**
- **[Khalili2016] proposed a multi-level classification of edge nodes, such that the internal state at routes/switches are minimal**

Example for fabric

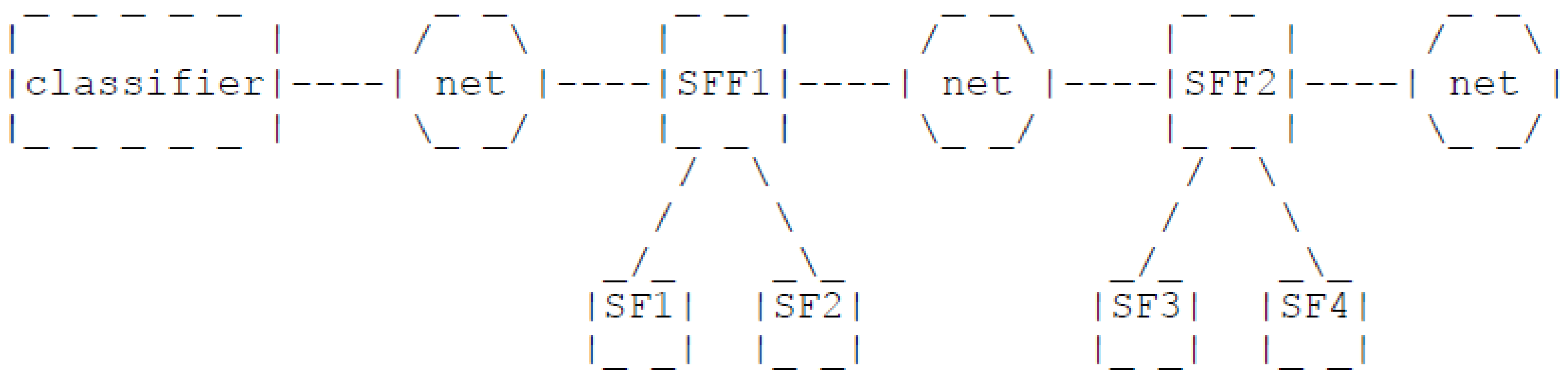
- **Assume that SFF1, SFF2, and Egress node are edges**
- **Applying ASC algorithm in [Khalili2016], we get**
  - □ IDSFF1 = (0, 1, 0, 0, 0),
  - □ IDSFF2 = (1, 0, 1, 0, 0),
  - □ IDSFF3 = (1, 0, 0, 1, 0),
  - □ IDEgress = (1, 0, 0, 0, 1).
- **Assume that these locator IDs are embedded in transport header (will be explained in next slides)**

- **At C1:**
  - □ if 1st bit is 1, forward over port 3,
  - □ if 2nd bit is 1, forward over port 2.
- **At C2:**
  - □ if 2rd bit is 1, forward over port 1,
  - □ If 3th bit is 1, forward over port 2,
  - □ If 4th bit is 1, forward over port 3,
  - □ If 5th bit is 1, forward over port 4.

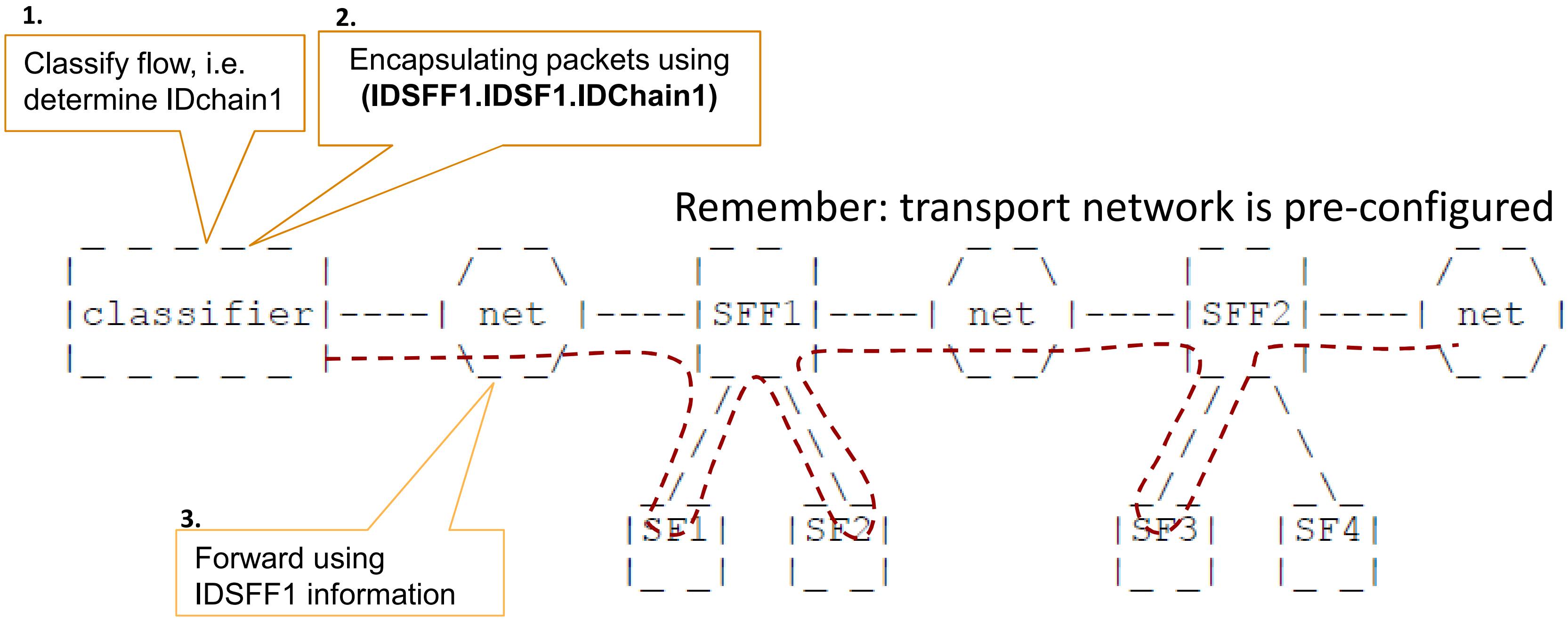


Hierarchical addressing for service chaining

- **Treats SFFs as edge nodes**
- **Address structure: (IDSFF.IDSF.IDChain)**
  - □ IDSFF1 and IDSFF2: locator IDs assigned to SFF1 and SFF2
  - □ IDSF1 and IDSF2: IDs assigned to SF1 and SF2 by SFF1
  - □ IDSF3 and IDSF4: IDs assigned to SF3 and SF4 by SFF2
  - □ IDChain: ID assigned to the chain by the classifier (or the controller)
- **Assumption: SFFs and SFs IDs are known by the classifier**



Example: chain SF1-SF2-SF3 (1)


1\.

Classify flow, i.e. determine IDchain1

2\.

Encapsulating packets using **(IDSFF1.IDSF1.IDChain1)**

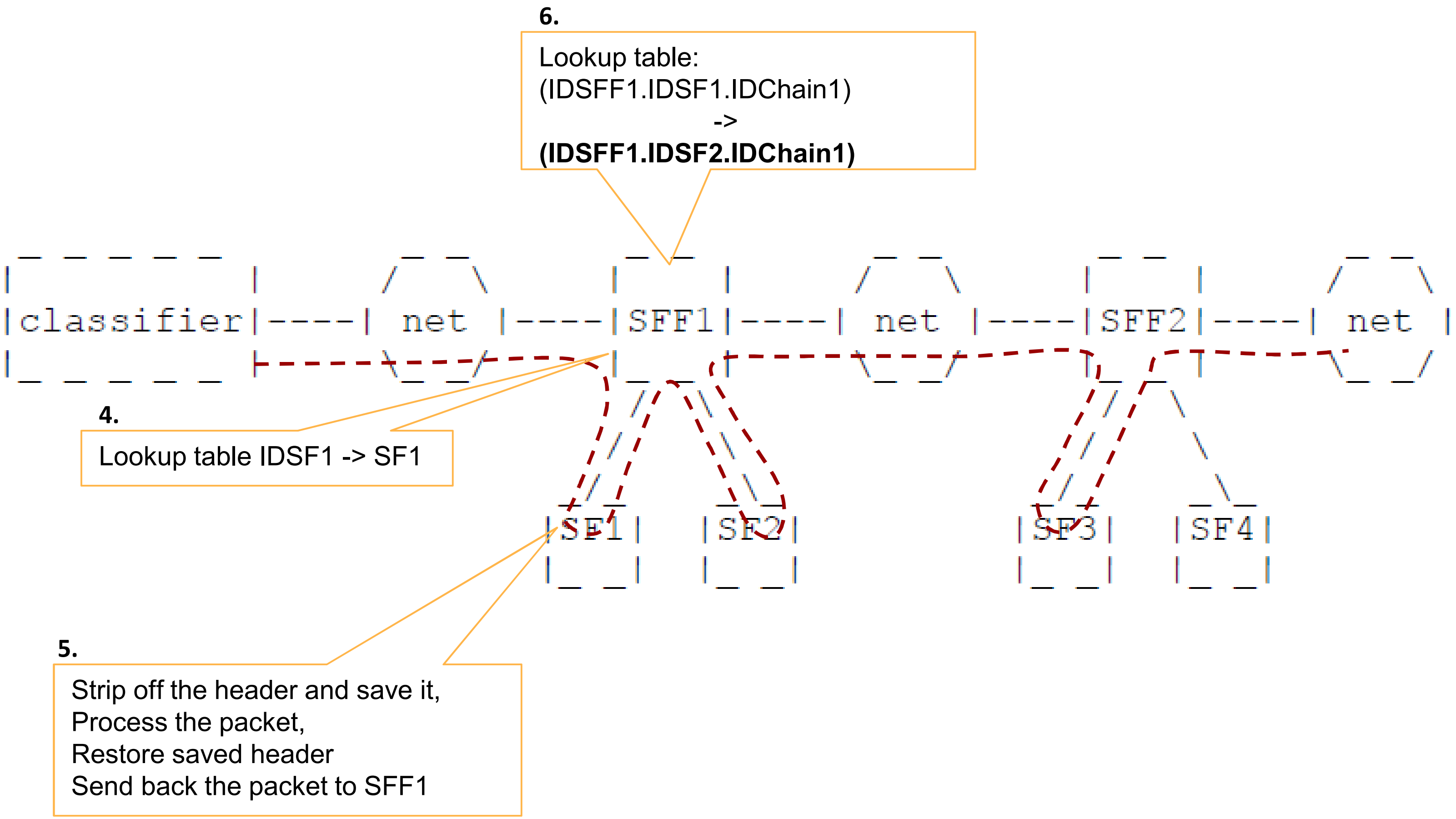
3\.

Forward using IDSFF1 information

Remember: transport network is pre-configured

- **IDchain1: ID assigned to the chain by the classifier**
  - This ID should be unique among all chains starting by SF1

Example: chain SF1-SF2-SF3 (2)



6.

Lookup table:
(IDSFF1.IDSF1.IDChain1)
->
(IDSFF1.IDSF2.IDChain1)

4.

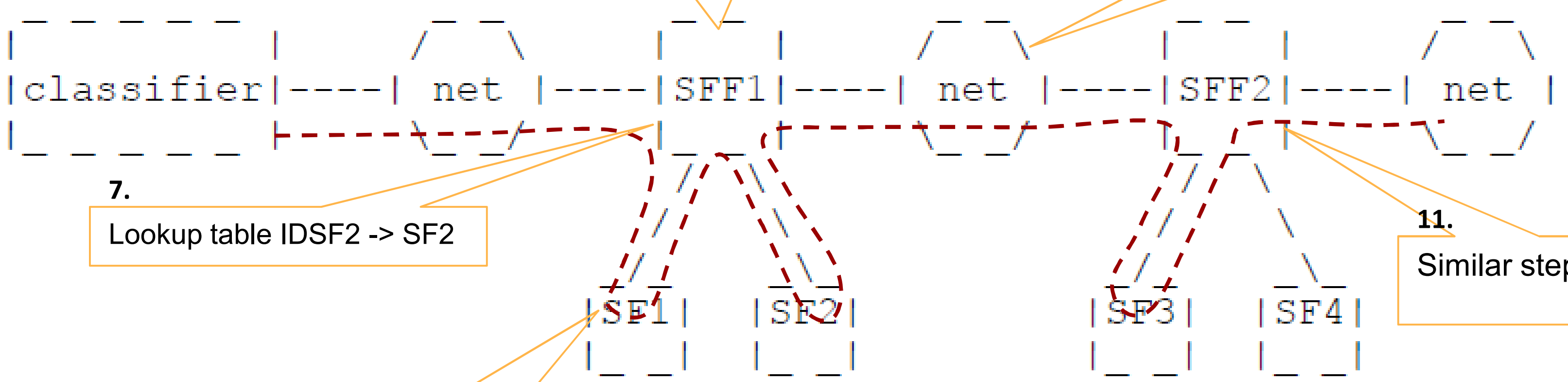
Lookup table IDSF1 -> SF1

5.

Strip off the header and save it,
Process the packet,
Restore saved header
Send back the packet to SFF1

Lookup table information at SFF1 and SFF2 are updated by classifier (or the controller)

Example: chain SF1-SF2-SF3 (3)



7.

Lookup table IDSF2 -> SF2

8.

Strip off the header and save it,
Process the packet,
Restore saved header
Send back the packet to SFF1

9.

Lookup table:
(IDSFF1.IDSF2.IDChain1)
->
(IDSFF2.IDSF3.IDChain1)

10.

Forward using
IDSFF2 information

11.

Similar steps at SFF2

Alternative chaining addressing

- **Considers SFs as edge nodes**
- **Multiple IDs could be assigned to an SF, each representing one of the chains that contains this SF**
  - An extension of the edge classification algorithm in [Khalili2016] can be used for this purpose
- **In the previous example:**
  - We assign IDSF1Chain1, IDSF2Chain1, and IDSF3Chain1 to SF1, SF2, and SF3, for the chain SF1-SF2-SF3
- **Similar steps, with following changes**
  - Step 2: encapsulating packets using IDSF1Chain1
  - Step 6: lookup table IDSF1Chain1 -> IDSF2Chain1
  - Step 9: lookup table IDSF2Chain1 -> IDSF3Chain1

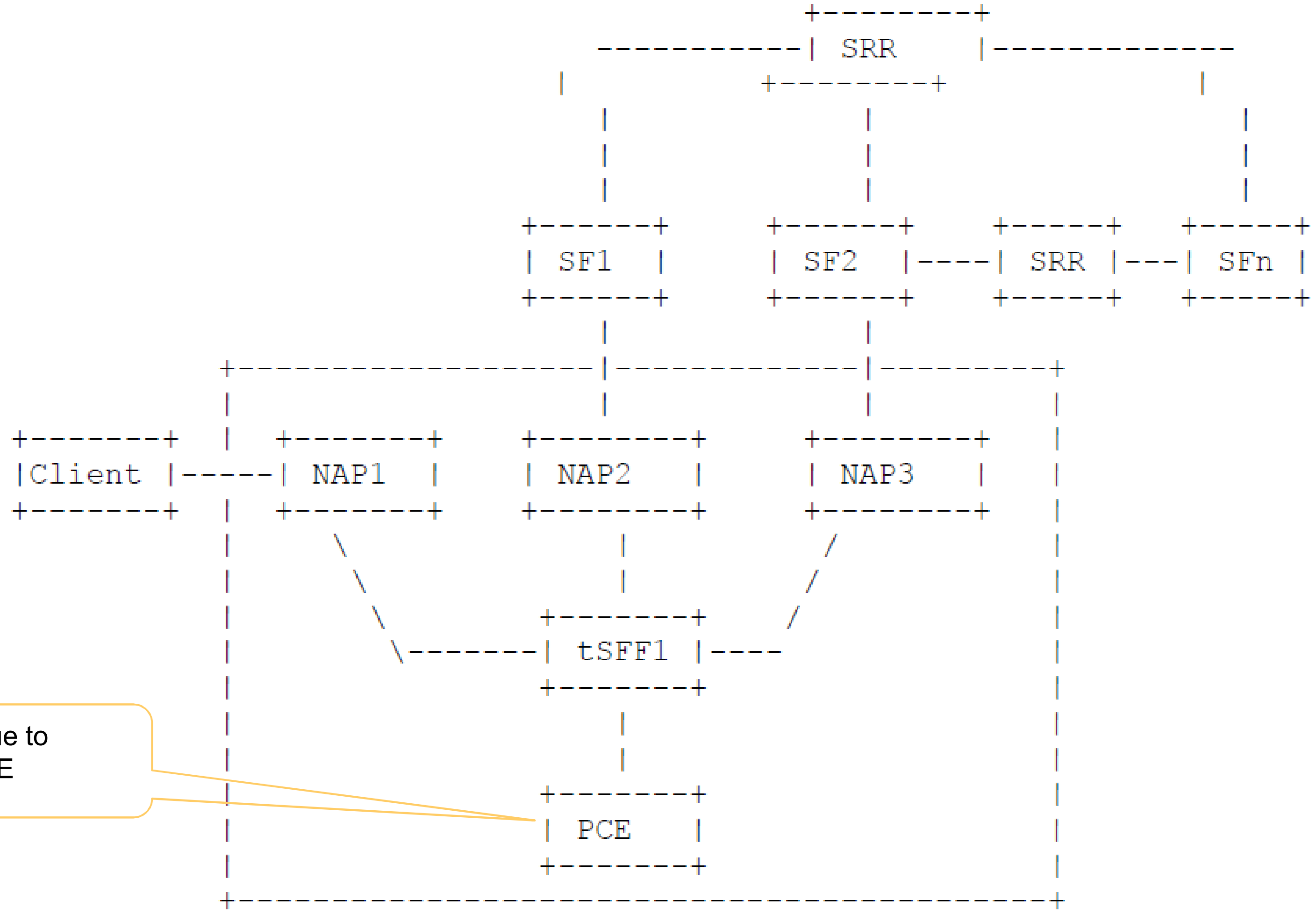
SRR-Based Solution

- **HTTP-level transport being introduced in *draft-purkayastha-sfc-service-indirection-01* for chaining at named service function level**
- **Service-request routing (SRR) function has been proposed as means to flexible re-chain virtualized (named) SF endpoints**

```
                                       +---------+
                                       |         |
    |--------------------|-------------+   SRR   + <-------------|
    |                    |             |         |               |
    |                    |             +----/|\--+               |
    |                    |                   |                   |
+--\|/--+   +-----+   +-\|/-+   +-----+   +--+--+   +-----+   +--+--+
|       |   |     |   |     |   |     |   |     |   |     |   |     |
+Client +-->+SRR  +-->+ SF1 +-->+SRR  +-->+ SF2 +-->+SRR  |-->| SFN |
|       |   |     |   |     |   |     |   |     |   |     |   |     |
+-------+   +-----+   +-----+   +-----+   +-----+   +-----+   +-----+
```

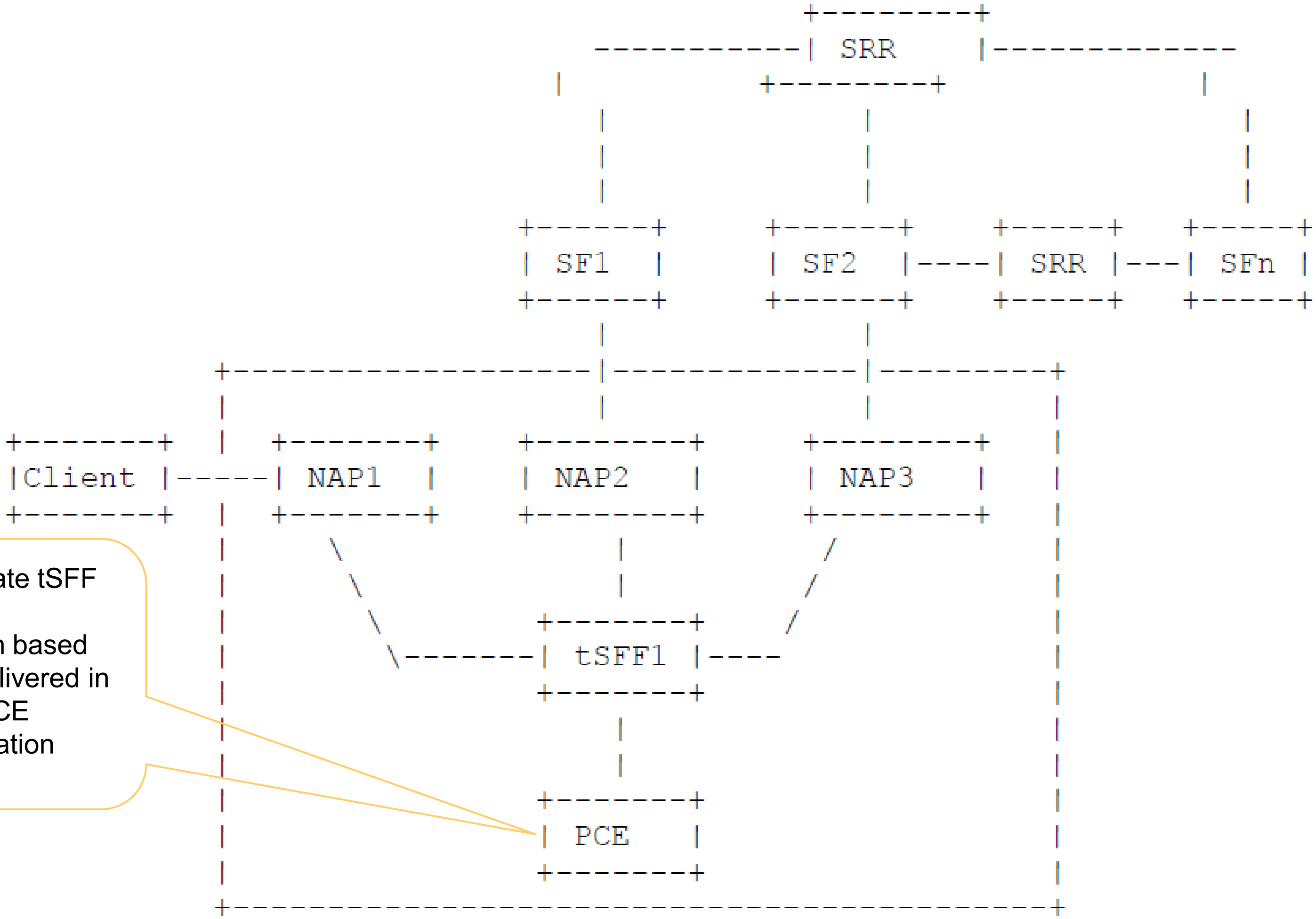
Figure 2: Dynamic Chaining SFC, as proposed in [Purka2017]. SFP: 192.168.x.x -> www.foo.com -> 192.168.x.x -> www.foo2.com -> ... -> www.fooN.com

SRR Decomposition Across Two SRR NAPs



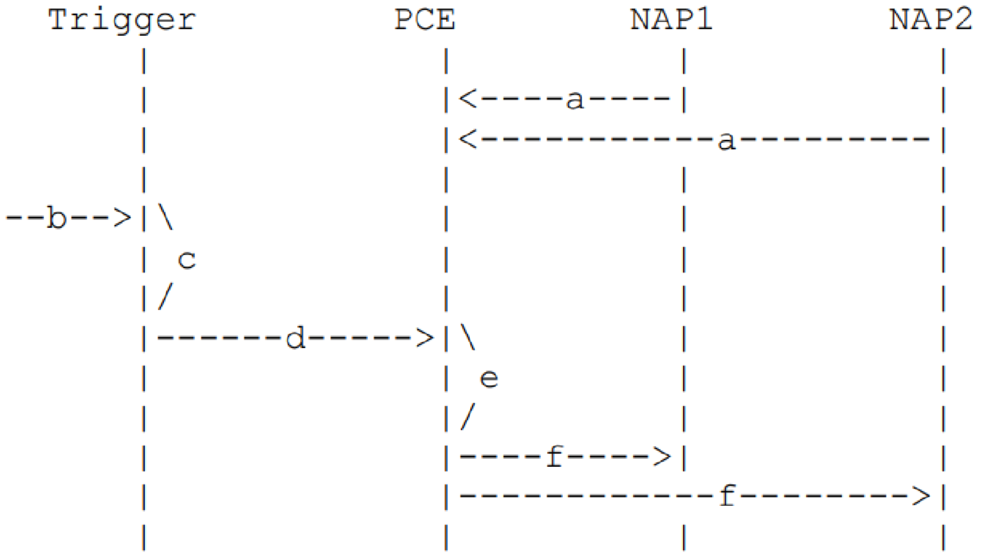
Latency due to
NAP-> PCE
interaction

Proposal: Pre-Warming of tSFF Entries



Pre-calculate tSFF forwarding information based on SFP delivered in trigger->PCE communication

Signaling for Pre-Warming of tSFF Entries



```
  Trigger         PCE        NAP1        NAP2
      |             |          |           |
      |             |<----a----|           |
      |             |<-----------a---------|
      |             |          |           |
--b-->|\            |          |           |
      |  c          |          |           |
      |/            |          |           |
      |------d----->|\         |           |
      |             |  e       |           |
      |             |/         |           |
      |             |----f---->|           |
      |             |------------f-------->|
      |             |          |           |
```

Message Sequence Chart Resulting in Pre-Warming of Routing Entries. a) subscribe to pre-warming information, b)initiate service chaining based on external mgmt. trigger, c) compute SFP, d) send SFP, e) map SFP information onto paths from incoming to ongoing NAPs, f) push path information with forwarding/path identifier and URL.

Triggers

- **Could be orchestration-based**

- **Could be determination of SFPs through some external entity**
  - Needs protocol between SRR PCE to said external entity

- **NOTE: re-chaining for re-routing to alternative SF endpoints is done SRR-internal and does not require pre-warming yet uses similar tSFF information dissemination as pre-warming!**

Discussion

- **Transport encapsulation can be used for service function chaining**
  - Easy to implement: SFs and SFFs do not have to be sfc aware
  - Reduce flow setup latency
- **Dynamic service chaining should be possible and will be discussed in future versions**
- **We need also to study which existing transport encapsulation give us the needed feature**
- **Question: if sfc group finds this line of work interesting and related?**