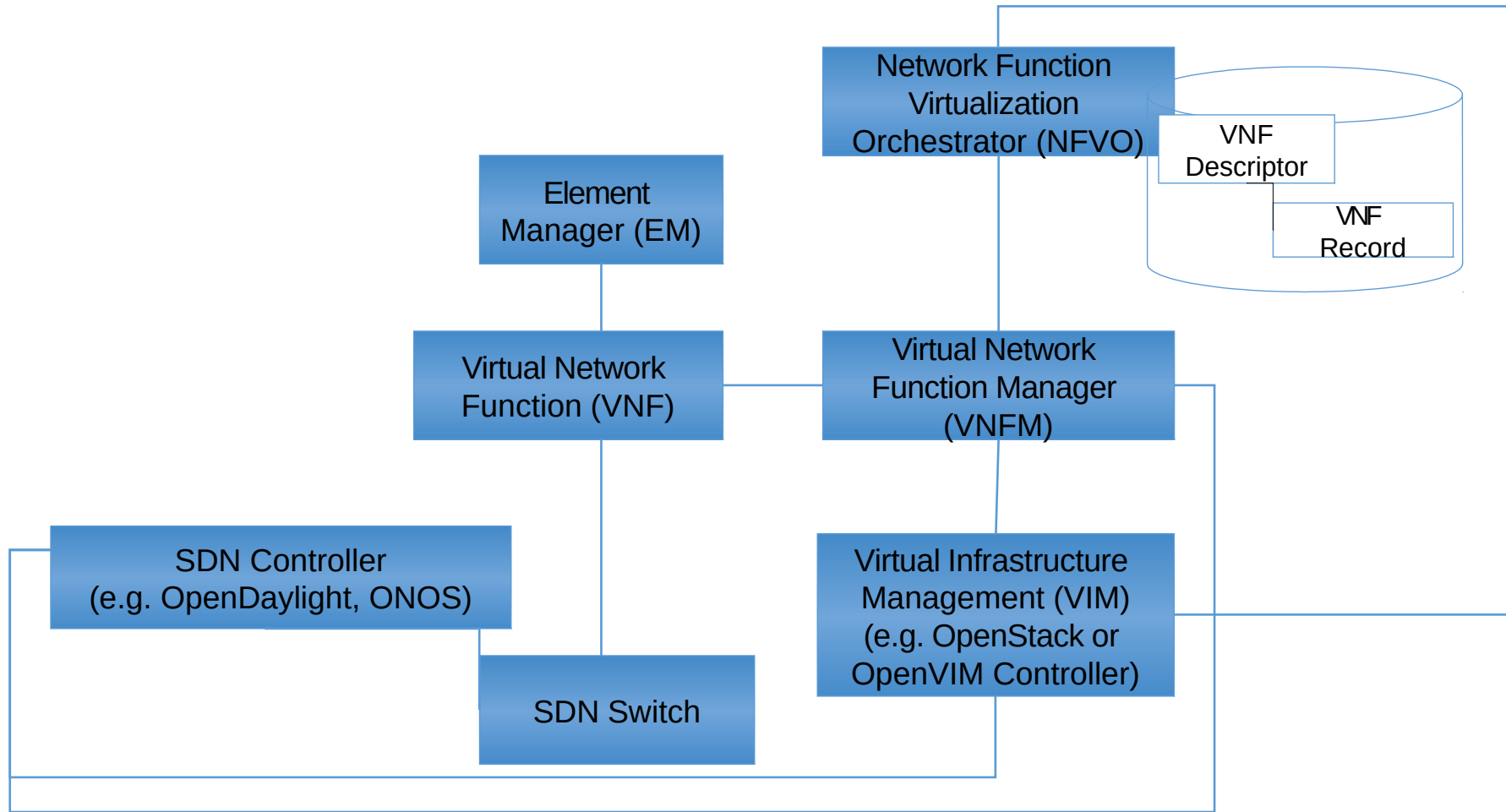# Flexible NFV+SDN Orchestration

Enabled by ALTO

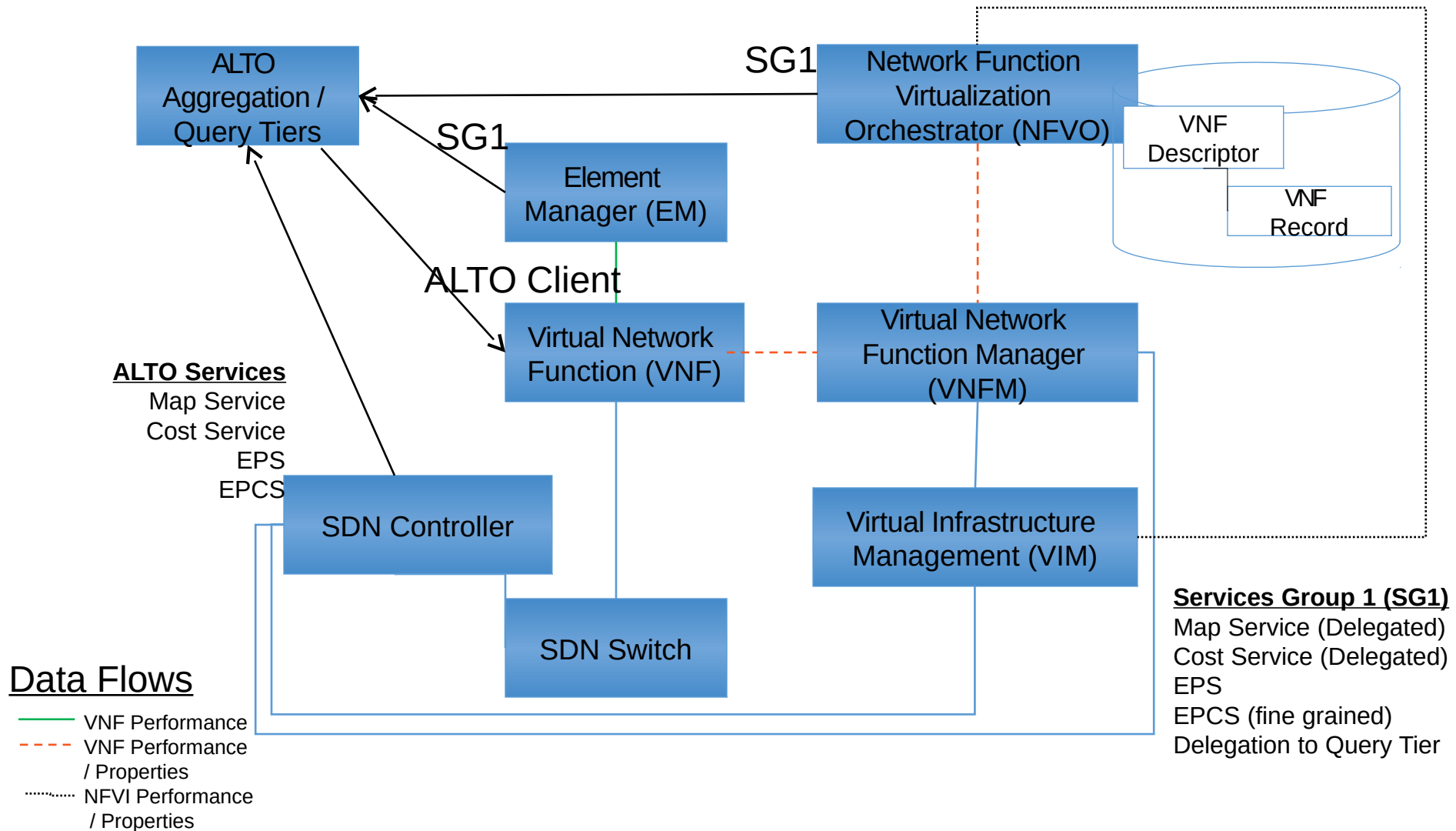draft-bertz-alto-sdnnfvalto-02

Overview

- Network Function Virtualization (NFV)
    - SDN + NFV (SDN pulled out of ETSI NFV View)
- AlTO in the SDN + NFV environment
- Requirement 1: ALTO MUST support Aggregation
- Orchestrator's Task
- Orchestration Algorithm
- New Use Cases for ALTO
- Measurement Initiation
- Proposed System
- Road Ahead

SDN + NFV

Network Function Virtualization Orchestrator (NFVO)

VNF Descriptor

VNF Record

Element Manager (EM)

Virtual Network Function (VNF)

Virtual Network Function Manager (VNFM)

SDN Controller (e.g. OpenDaylight, ONOS)

Virtual Infrastructure Management (VIM) (e.g. OpenStack or OpenVIM Controller)

SDN Switch

NOTE: SDN Controller / Switches are often considered to be part of VIM but may also be VNFs.

NFV / SDN + ALTO

REQ 1: ALTO MUST support Aggregation

- Aggregation of data concept is required in ALTO
  - Client side integration is complex and defeats the ease of Service ALTO provides
  - There will be multiple domains and with filters people are likely to share data
- Biggest issue for ALTO is an incorrect assumption of number of sources, esp. SDN Controllers
  - SDN controllers must remain close to their switches if they are supporting signaling protocols
  - Cannot push a bunch of data from multiple SDN Controllers to an ALTO server then throw 99% of it away
  - Controller separation for security purposes is a reality
  - SDN Controllers are considered part of the VIM
  - The number of NFV instances (VIM + rest of OpenStack) is MUCH higher than anyone anticipated
    - Roughly 4-12 very dense racks can be managed as one set of VIM instances
      - This implies an SDN Controller for every 8-12 racks…
      - The more protocols you pack into controllers the higher probability it can serve less
        - This can be countered in IETF by developing transport / signaling protocols that meet many needs and scale easily but is a lofty goal…
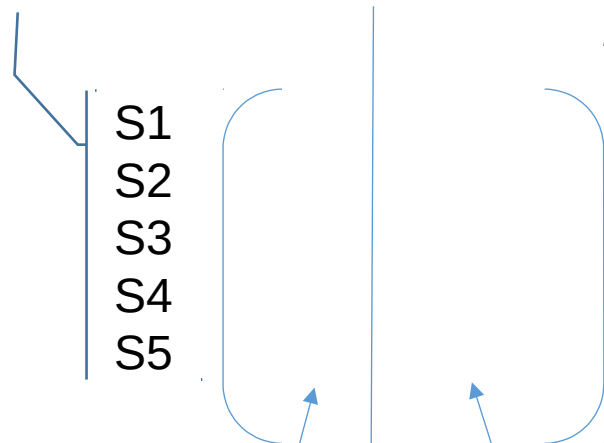
ALTO must support aggregation

Orchestrator's Task(s)

- How to request from VIM(s) what you need (and get it) given a VNF Forwarding Graph (VNF FG)?
  - Splitting up a VNF FG means you need to know how VIMs can connect, i.e. you need network topology.
  - VNFs have constraints
    - Some are resource, e.g. CPU, Memory, etc.
    - Some are performance, e.g. total latency < 50ms
- What is the VIM doing?
  - According to suppliers it is efficiently packing virtual machines, containers, etc.
    - It is doing this on a constant basis
    - Isn't this some form of Bin Packing?

Orchestrator Looking @ a VIMs

STEP 1 –
Orchestrator generates *partial solutions,*
i.e. subgraph, of the VNF FG that can be satisfied by the VIM's resource information.
These become candidate partial solutions CX for a subgraph of the VNF FG X
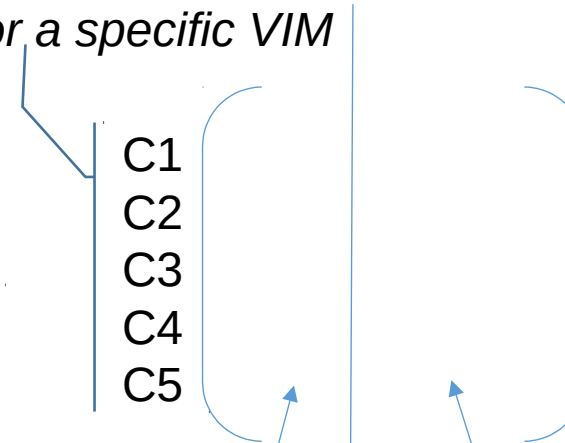
Each row is a subgraph of the VNF FG

STEP 2 – Compute Matrices

Each row is a candidate partial solution, i.e. subgraph, of the VNF FG for a specific VIM

S1
S2
S3
S4
S5

C1
C2
C3
C4
C5

$-$

$=$

Any row w/o negative values is a valid candidate

Resources Consumed by subgraph

Constraints for Subgraph and adjacent subgraphs

Resources from VIM

Performance Constraints of VIM provided via ALTO

VIM Candidate Suitability Matirx

Subgraph Matrix

VIM Candidate Matrix

All constraints are set up so that if value > 0 constraint is satisfied…

Step 3 - Ranking

Any row w/o negative values is a valid candidate (n x m Matrix) · Weight Vector (m x 1 matrix) = Weighted Solution Values (n x 1 vector)
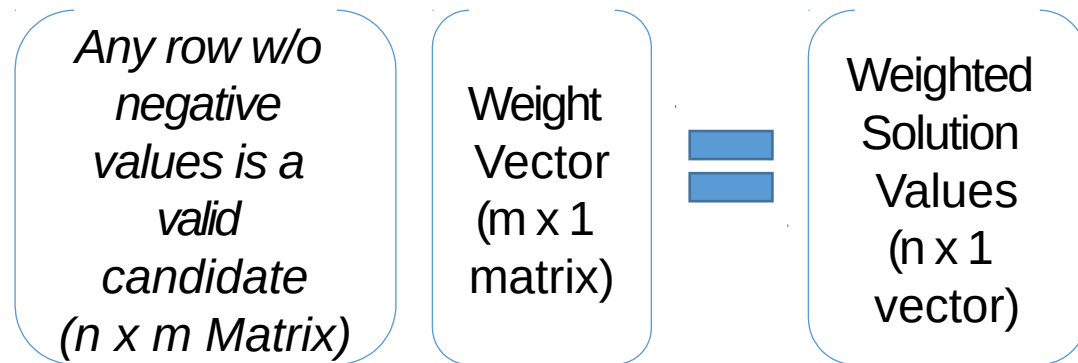
NEW Use Cases for ALTO
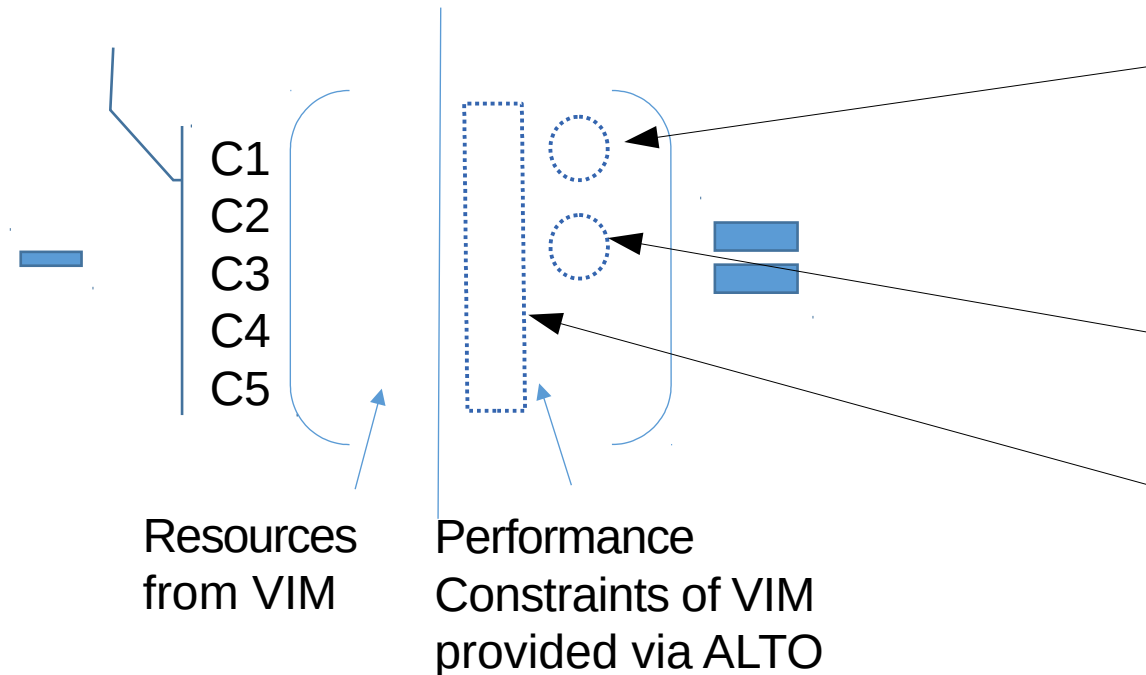
When data was not present in VIM candidate matrix

UC 1 : Data is present in underlying Server BUT not visible in ALTO Server (implies a Filter)
=> ALTO MUST support Server side filtering (On Demand Measurements is proposed)

UC 2: Data is not present server at all => ALTO MUST support some form of **Measurement Initiation**

UC 3: Metric not **currently supported by ALTO Sever** => Dynamic loading of measurement data
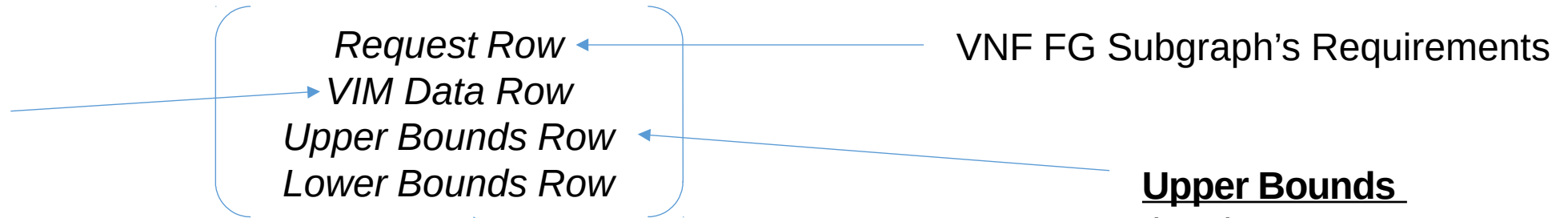
C1
C2
C3
C4
C5

Resources from VIM

Performance Constraints of VIM provided via ALTO

VIM Candidate Matrix

UC2 Overview
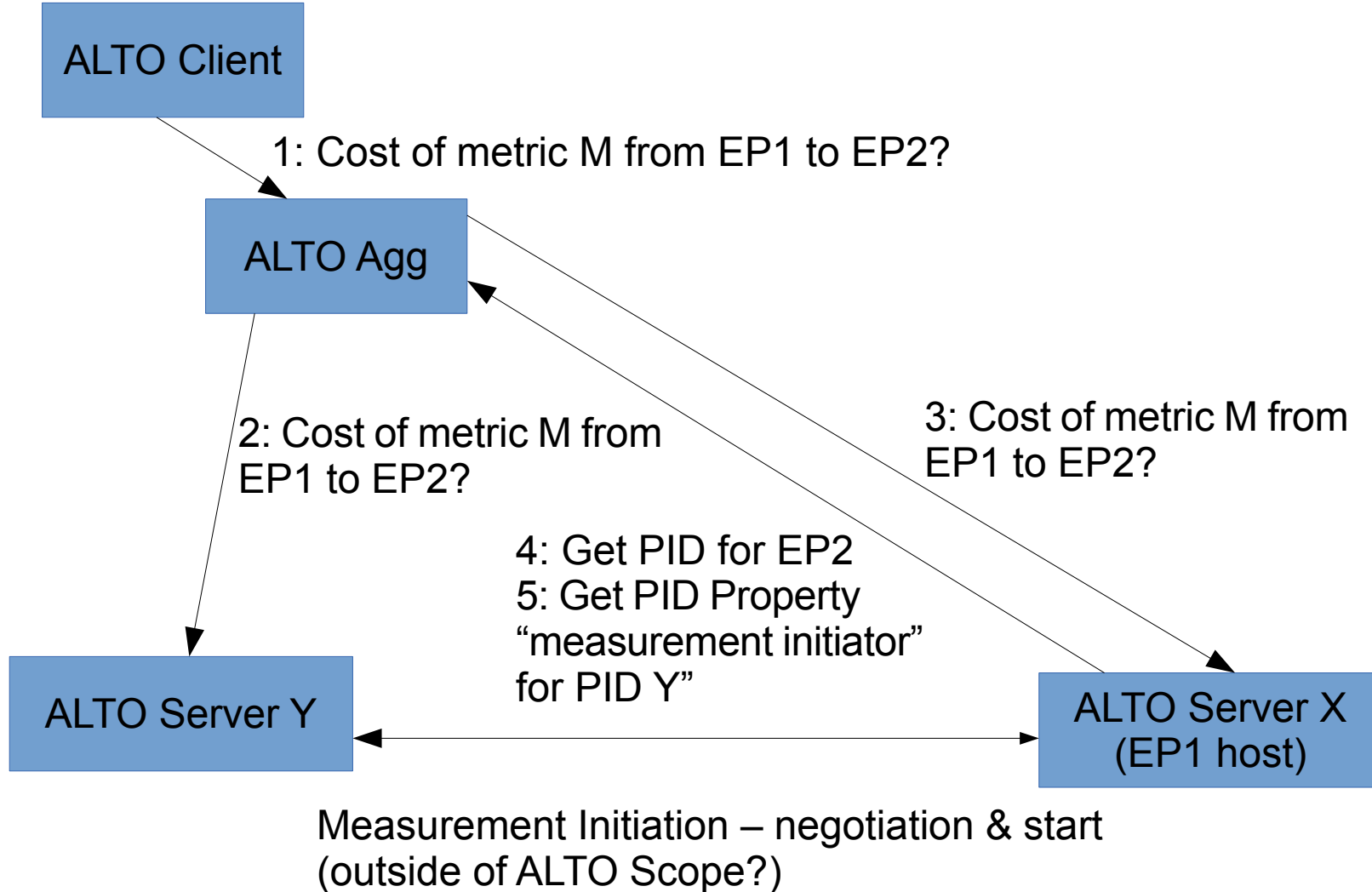
Communication with a Programmable VIM

Orchestrator 'asks' VIM by sending a set of 4 rows

Request Row
VIM Data Row
Upper Bounds Row
Lower Bounds Row

VNF FG Subgraph's Requirements

VIM Data – It shows the ALTO (performance) + Resource Information that was used to determine the selection

Upper Bounds that the VIM may auto-scale the VNFs in the VNF FG subgraph to w/o querying the VIM

Lower Bounds where if they are maintained for a pre-defined period will require the VIM to notify the Orchestrator.

Proposed System

- Elements
  - ALTO Server integrated SDN Controller
  - ALTO aggregators
  - Measurement Initiators (may be part of Controllers)
  - ALTO integrated NFVO
  - ALTO integrated VNF EM (optional)
  - ALTO Client based Orchestrator
- Features
  - ALTO aggregation for Scaling over multiple SDN Controllers / NFVI domains
  - On Demand Measurements / Measurement Initiation to adapt to different Orchestrator information needs
  - Minimal interchange between VIM and Orchestrator
  - Orchestrator can get data from ANY ALTO source
  - System can
    - Adapt to new metrics from VNFs
    - Send only data that is required
    - Auto discovery of VIMs, Aggregators and Orchestrators via ALTO
- Still permits differentiation for Orchestrators and VIMs

Road Ahead

- Feedback requested.
    - (Thank you to those who have already provided it!)
- Should this become a Requirements Document for ALTO?