

# Gap Analysis and Requirements of CAN

draft-liu-can-gap-reqs-00

P. Liu, China Mobile

T. Jiang, China Mobile

P. Eardley

D. Trossen, Huawei

C. Li, Huawei

D. Huang, ZTE

115 IETF CAN BOF



# CAN Gap Analysis

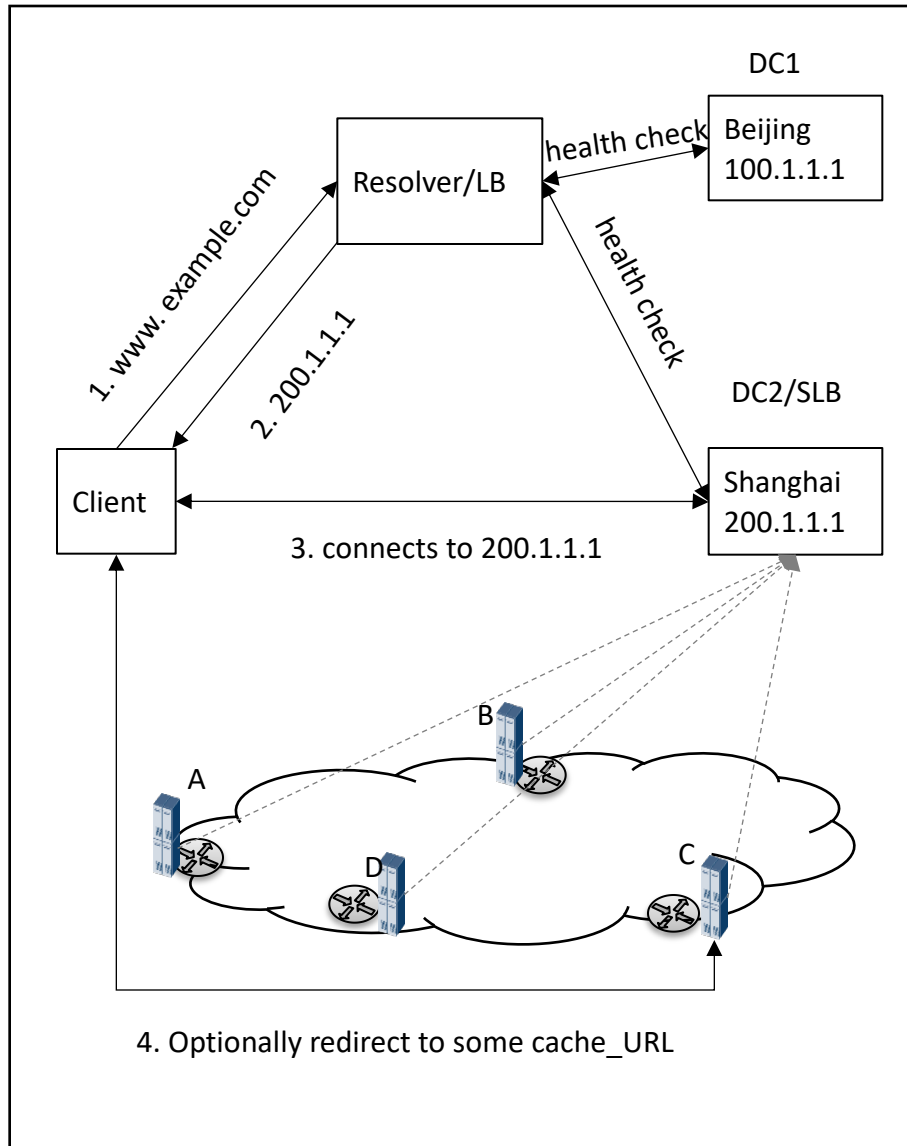
# Existing Solutions

There are existing solutions that utilize end point resource information for selecting server instances, such as DNS, ALTO, or Layer 4 & Layer 7 load balancers.

- Common to all those solutions is the **exposure and usage of network conditions** for decision making, ...
- Together with the **use of an indirection approach** for determining the 'best' service instance, while...
- None of the existing solutions **integrate the computing resource conditions with the network conditions** for deciding the optimal paths.

This has implications upon supporting the **dynamicity** of service relations, the **exposure** of holistic compute/network metrics, as well as the **efficiency** and **complexity** of those solutions

# Gap Analysis of Existing Solutions-DNS/GSLB



- **Early binding:** clients resolve IP address first and then send traffic.
  - If using the DNS entry cached at client, stale info may be used.
  - Often, resolver and LB are separate entities which incurs even more signaling overhead by needing to first resolve and then redirect to LB for final decision
  - Resolution is L7 or app-level decision making, i.e. DB lookup. Intended for control, NOT data plane speed!
- **Health check:** on an infrequent base, switch when fail-over
  - Limited computing resources at edge will change rapidly, while more frequent health check is prohibitive in cost
- **Load balance over DNS:** usually focused on edge server load first, then utilizing lowest latency routing to the selected server's IP address
  - Lacks combined consideration of load & latency for a better E2E experience
  - Problem of how to obtain necessary metrics for decision
  - [DNS-SD\[RFC6763\]](#) and [Multi-cast DNS\[RFC6762\]](#) could be used to discover the service, which might be extended to collect the computing information. However, in most cases, they are used in the LAN environment. Use in WAN faces the same DNS stale cache problem.

# Gap Analysis of Existing Solutions - Load Balancer

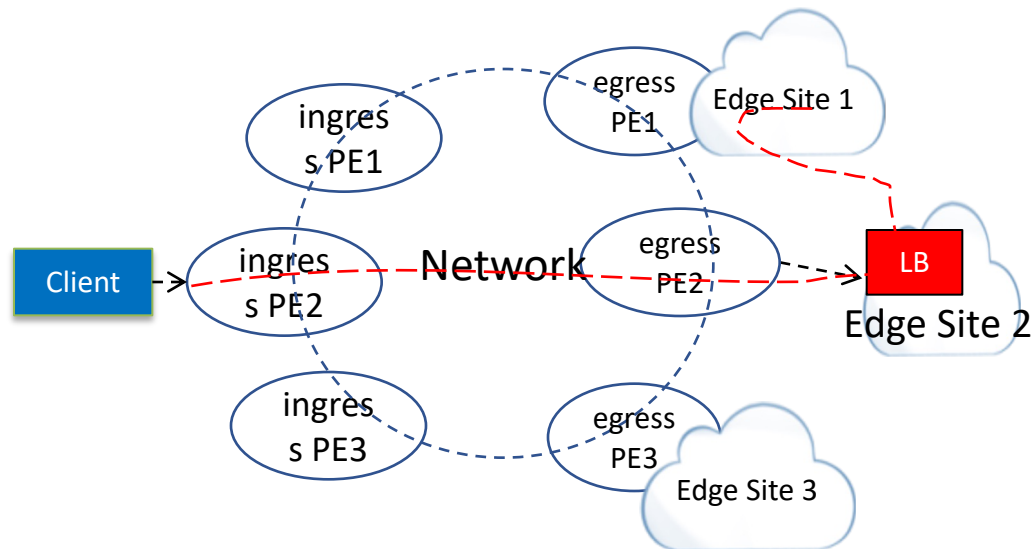
## Single LB in a single site for all server instances:

### Pros:

- easy deployment

### Cons:

- Single point of failure at the LB
- *Path stretch*, i.e., the network path from the LB to server instances at other sites might not be optimal, e.g., the red dotted path



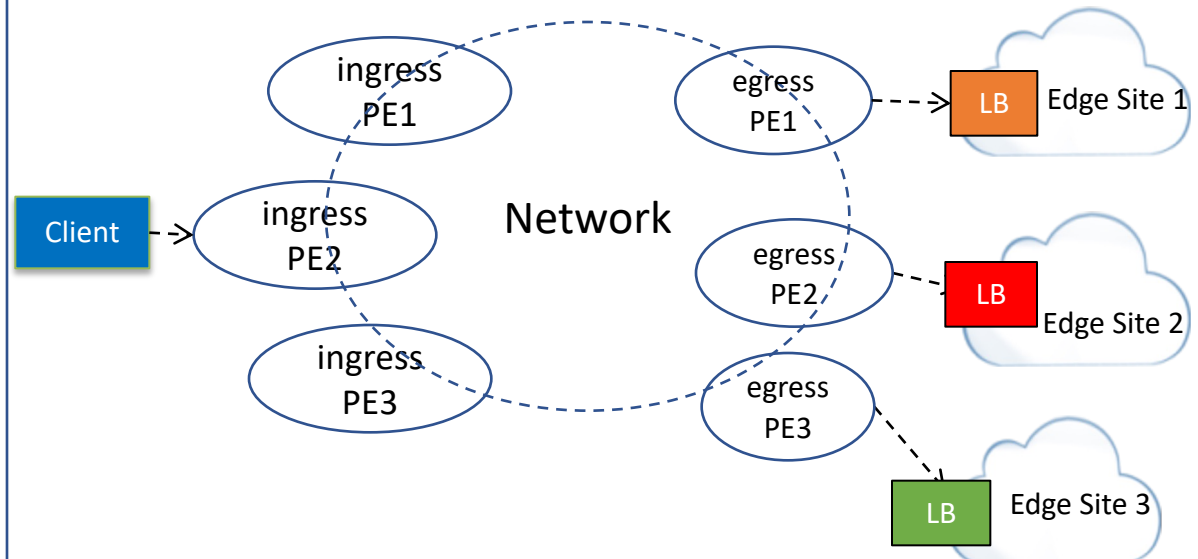
## Each site has its own LB:

### Pros:

- Easy deployment.

### Cons:

- No load balance among multiple sites. Any site might be overloaded.
- Problem on selecting the 'best' site remains unsolved

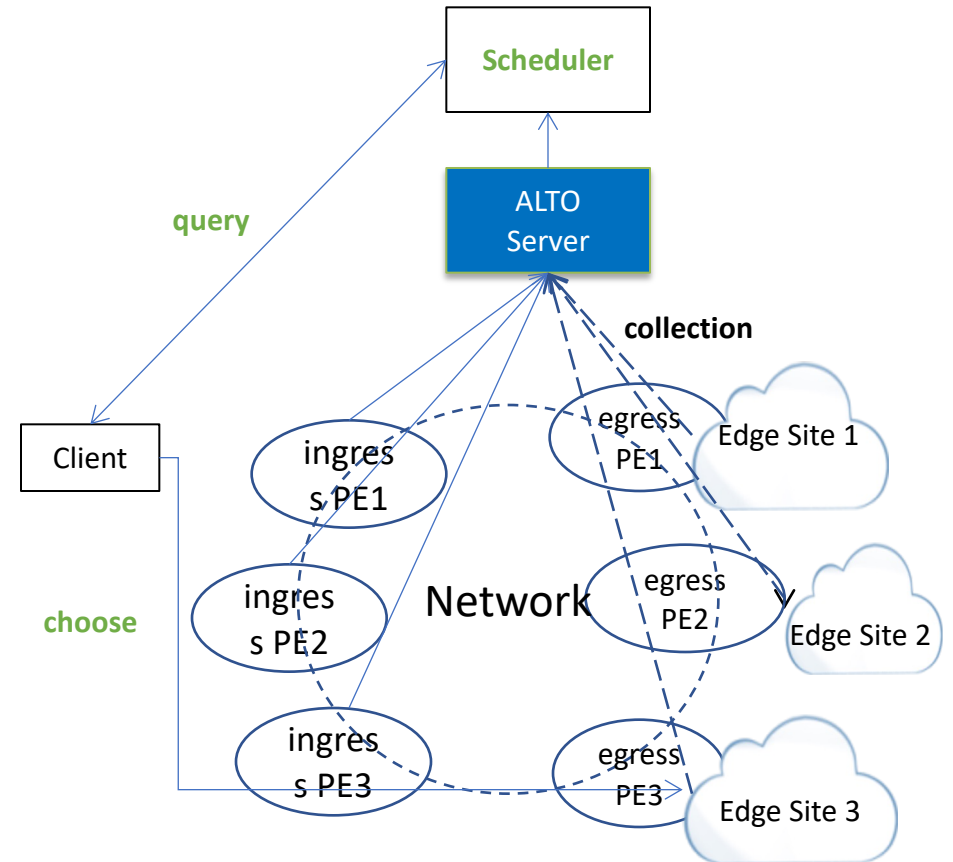


# Gap Analysis of Existing Solutions - ALTO

ALTO [RFC7285] addresses the problem of selecting the 'optimal' service instance as an **off-path solution**, which can be seen as an alternative to DNS for tackling the problem space of CAN at the Application Layer.

The critical aspect is the signaling latency and the control plane load that a service-instance selection process may incur in both on-and off-path solutions. This in turn may impact the frequency with which applications will query ALTO server(s), especially in the mobile system.

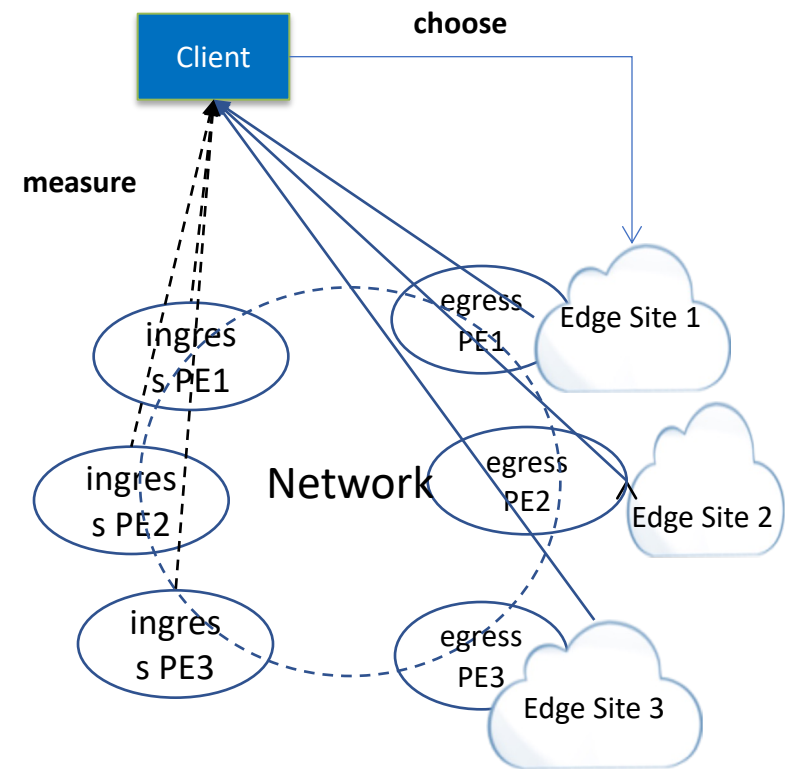
As a result, off-path systems, e.g., ALTO, which are based on receiving replies for applications/services before traffic could be delivered, might not keep optimal or even valid after the handover.



# Gap Analysis of Existing Solutions – Client-based solution

The individual destination, i.e., the network identifier for a service instance, must be known to the client a priori for direct service dispatching. While this may be viable for certain applications, it suffers from following issues:

- Unwanted service instance exposure: It may be undesirable for clients to learn all available service instance identifiers for reasons of Service Providers' being reluctant to expose their 'valuable' information to clients.
- Can not learn all network paths: Operators would not expose all available paths to clients, forcing clients to measure the network by itself.
- Scalability: The number of service instances and network paths may be very high.



# Summary of Gap Analysis

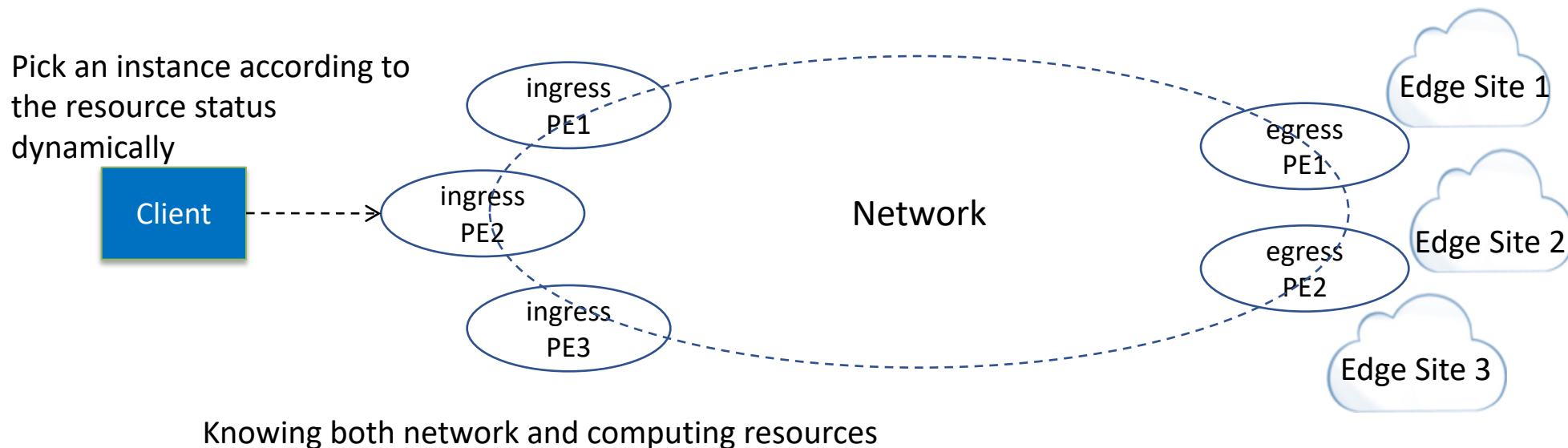
- **Dynamicity:** Existing solutions exhibit limitations in providing dynamic instance affinity
  - E.g., DNS is not designed for this level of dynamicity (i.e., minute level originally, client needs to flush the local DNS cache, frequent resolving may lead to overloading DNS)
- **Efficiency:** Existing solutions may introduce additional latencies and inefficiencies (e.g., more messages) in packet transmission, e.g., path stretch.
- **Complexity and Accuracy:** Existing solutions require careful planning for the placement of necessary control plane functions in relation to the resulting data plane traffic, which is difficult and may lead to the inaccuracy of the scheduling.
- **Metric exposure and use:** Existing solutions lack the necessary information to make the right decision on the selection of the suitable service instance due to the limited semantic or due to information not being exposed.



# CAN Requirements

# Main Goals

- **Service equivalence/Multi instances:** considering to access one of possibly many instances deployed across multiple sites
- **Multi-metric:** considering both the network and computing resource status
- **Dynamic decision:** select the appropriate computing resource dynamically



# Potential Main Requirements

- **Support access to the available edge sites dynamically**
  - Allow for making dynamic selections of service instances across multiple sites, based on suitable metrics
- **Provide both network and computing metrics for further use**
  - Computing resource related model for the metric
  - Control the rate of metrics distribution
- **Support effective computing resource representation and encapsulation**
  - Single index or multi-dimensional information for specific purpose
- **Support the session continuity and service continuity**
  - Functional equivalency
  - Preserve affinity in light of metric-based instance selection decisions
- **Preserve communication confidentiality**
  - Avoid unnecessary exposure of service as well as metric information

# Frequent Comments/Questions and Answers

- **Is upper layer solution such as ALTO, GSLB enough?**
  - Upper layer solution is off-path solution which will introduce redirection delay. CAN is about on-path solution which has lower latency and signaling load.
- **Do not load underlay network with application details**
  - The goal is to impact routing system decisions using computing-related informations. How to achieve that is still open for discussing. Solutions related discussion should be avoided in this stage.
  - However, some common concerns of solutions can be addressed by following potential design principles.
    - Only few routes associated to the CAN services will be impacted
    - Avoid impacting routes to instances, but **impacting the aggregated routes**(routes of services in a edge site)
    - The information MAY NOT be distributed by distributed protocols, it may be collected by controller, and the ingress node can learn the decision from the controller.
    - Common computing metric (if needed) needs to be distributed
      - to a **limited set of nodes** (only the ingress nodes related to the services)
      - In a **small scale network** (only the edge computing sites, or access rings, related to the services)
      - In a **limited frequency** (a reasonable frequency need to be set for each services)

Like other info that impacts the routing decisions, it is worth trying if enough benefits can get from doing so.

If the direction is roughly correct, let's work together to find the solutions.

# Thanks to People Who Contribute to /Comment on CAN

Luismiguel Contrerasmurillo<luismiguel.contrerasmurillo@telefonica.com>,  
Mohamed Boucadair<mohamed.boucadair@orange.com>,  
Christian Jacquenet<christian.jacquenet@orange.com>,  
Philip Eardley<ietf.philip.eardley@gmail.com>,  
Tim Costello<tim.costello@bt.com>,  
Gyan Mishra<hayabusagsm@gmail.com>,  
Luc Fabrice<Luc-Fabrice.Ndifor@mtn.com>  
Markus Amend<Markus.Amend@telekom.de>,  
Tianji Jiang<tianjijiang@chinamobile.com>,  
Adrian Farrel <adrian@olddog.co.uk>,  
Daniel King <daniel@olddog.co.uk>,  
Lanlan Rui<llrui@bupt.edu.cn>  
Wenjing Li<wjli@bupt.edu.cn>  
Yongqiang Dong<dongyq@seu.edu.cn>  
Qiang Dai<qdai0006@student.monash.edu>  
Zhengyuan Diao<zhengyuandiao@outlook.com>  
Changwang Lin<linchangwang.04414@h3c.com>,  
Jason Zhang<zhangjiaxing20g@ict.ac.cn>  
Dirk Trossen<dirk.trossen@huawei.com>,  
Luigi Iannone<luigi.iannone@huawei.com>,  
Zongpeng Du<duzongpeng@chinamobile.com>,  
Cheng Li<c.l@huawei.com>  
Linda Dunbar<linda.dunbar@futurewei.com>  
Yizhou Li<liyizhou@huawei.com>,

Kyoungjae Sun<gomjae@dcn.ssu.ac.kr>,  
Gomjae Younghak<younghak@ssu.ac.kr>,  
Mingyu Wu<wumy@centec.com>  
Junjie Wang<wangjj@centec.com>  
Peng Liu <liupengyjj@chinamobile.com>

Wim Henderickx<wim.henderickx@nokia.com>  
Marco Liebsch<marco.liebsch@neclab.eu>  
Joel Halpern<jmh@joelhalpern.com>  
Tony Li<tony.li@tony.li>  
Pete Resnick<resnick@episteme.net>  
Richard Barnes<rlb@ipv.sx>,  
Erik Nordmark<erik@zededa.com>  
David Oran<daveoran@orandom.net>,  
Jeff Tantsura<jefftant.ietf@gmail.com>  
Carlos J. Bernardos<cjbc@it.uc3m.es>  
Greg Mirsky<gregimirsky@gmail.com>,  
Christian Hopps<chopps@chopps.org>,  
Srihari Sangli<ssangli@juniper.net>  
Shraddha Hegde<shraddha@juniper.net>  
Lars Eggert<lars@eggert.org>  
Jens Finkhaeuser<jens@interpeer.io>

... ..

