Database-based Open Resource Service (DB-ORS) Framework, Usecases and Requirements

Fenlin Zhou; Dongyu Yuan; Sheng Wang; Xiaocong Qian

ZTE Corporation, China

IETF-115, November 2022

Documents

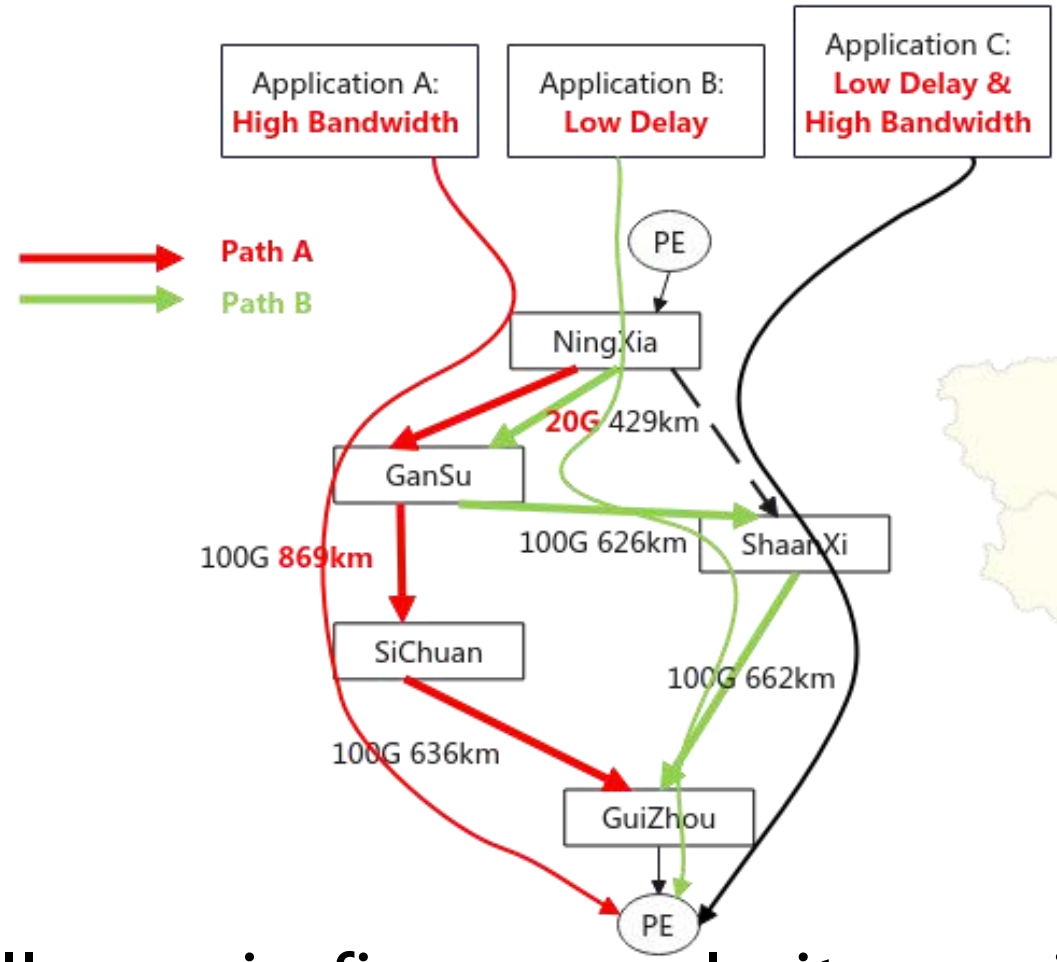
- Requirements and Use Cases of DB-ORS (Database-based Open Resource Service)

  https://datatracker.ietf.org/doc/**draft-zhou-alto-dbors-requirement-usecase/**

- Database-based Open Resource Service Framework

  https://datatracker.ietf.org/doc/**draft-zhou-alto-dbors-framework/**

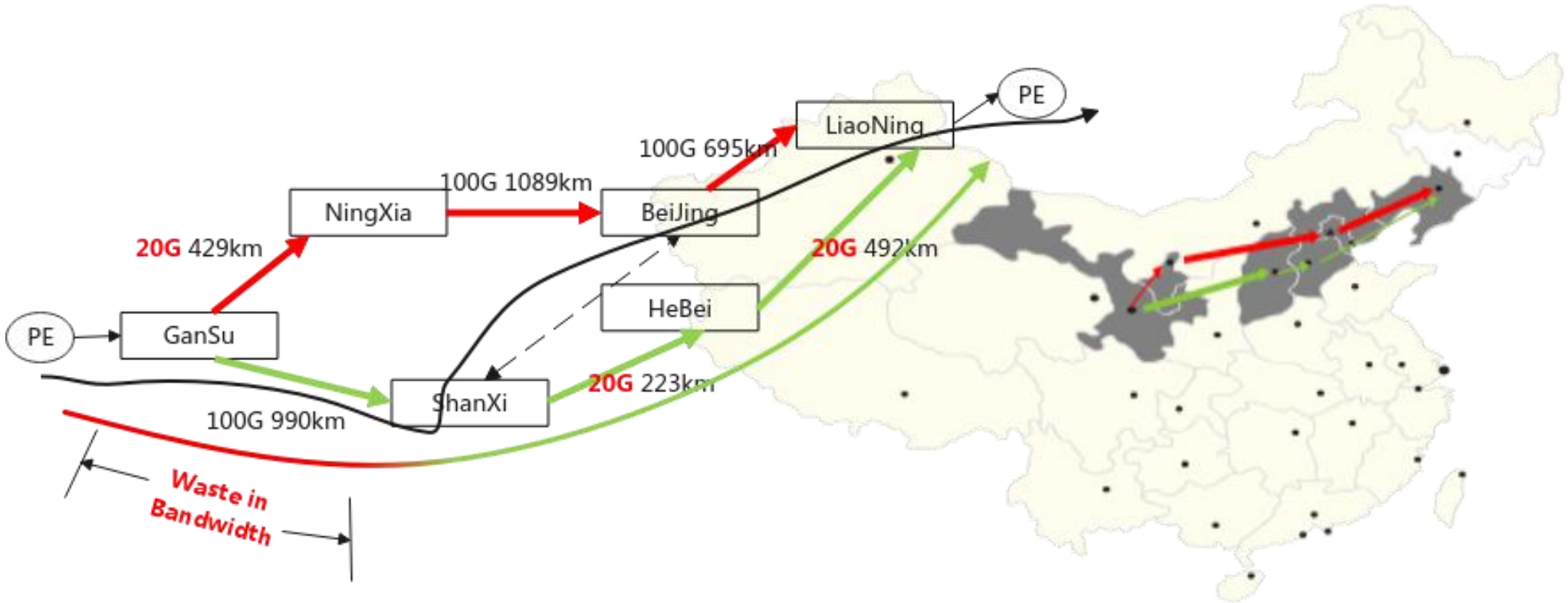
Challenges in Current Network



Challenge in fine-granularity service provisioning:

Capabilities of the network remain invisible，thus **differentiated services** are not achieved. Applications with various requirements can **not be distinguished and served customarily**.

Challenges in Current Network



Challenge in network resources utilization:

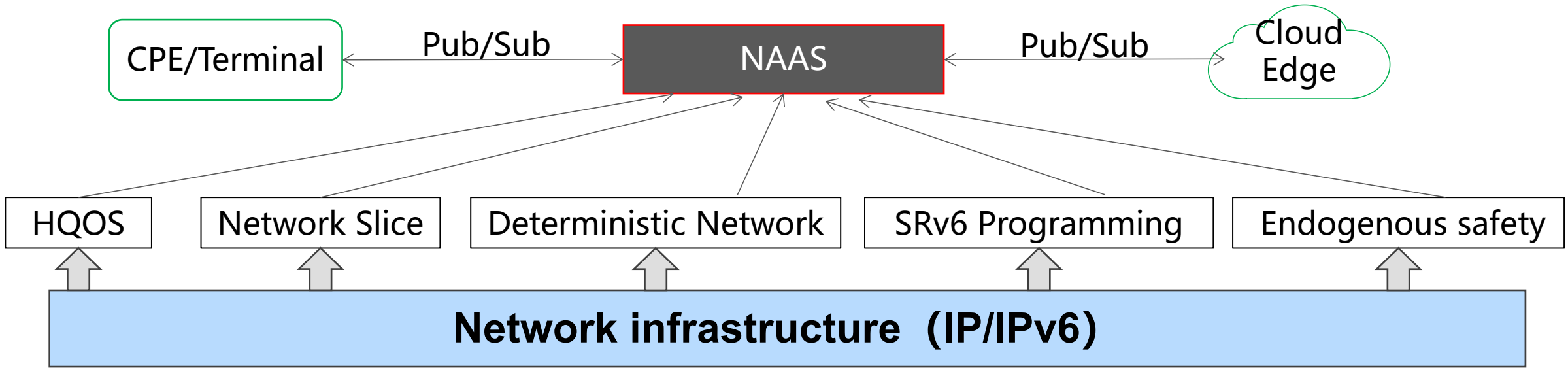
Due to the **bandwidth restrictions** in the Shanxi-Hebei and Hebei-Liaoning sections, the bandwidth provided by the network for applications can not achieve 100 G, and the **bandwidth resources from GanSu to ShanXi is wasted**.

Requirements of DB-ORS

- Fine-granularity services provisioning
  - Conventional networks only provide clients with coarse-grained connection services.
  - Services which propose diverse network requirements such as ultra-low latency and high reliability emerge increasingly, and therefore differentiated service treatment is desired.
- Network resources utilization enhancement
  - The network resources are not orchestrated appropriately and the resource utilization proves to be relatively low for about 30%-50%.
  - The complexity of the network results in low bandwidth utilization of the real data load.

https://datatracker.ietf.org/doc/**draft-zhou-alto-dbors-requirement-usecase/**

Pivotal Concepts in DB-ORS



SOLUTIONS

- Besides bandwidth for instance, the network has also been endowed with **various other capabilities** including deterministic quality, network slicing, endogenous security, etc.
- Referring to the concept of Software-as-a-Service (SAAS) , capabilities of the network are abstracted into services, namely **Network-as-a-Service(NAAS)**.
- Applications (cloud, terminal, and CPE) **subscribe** corresponding **customized network services**.

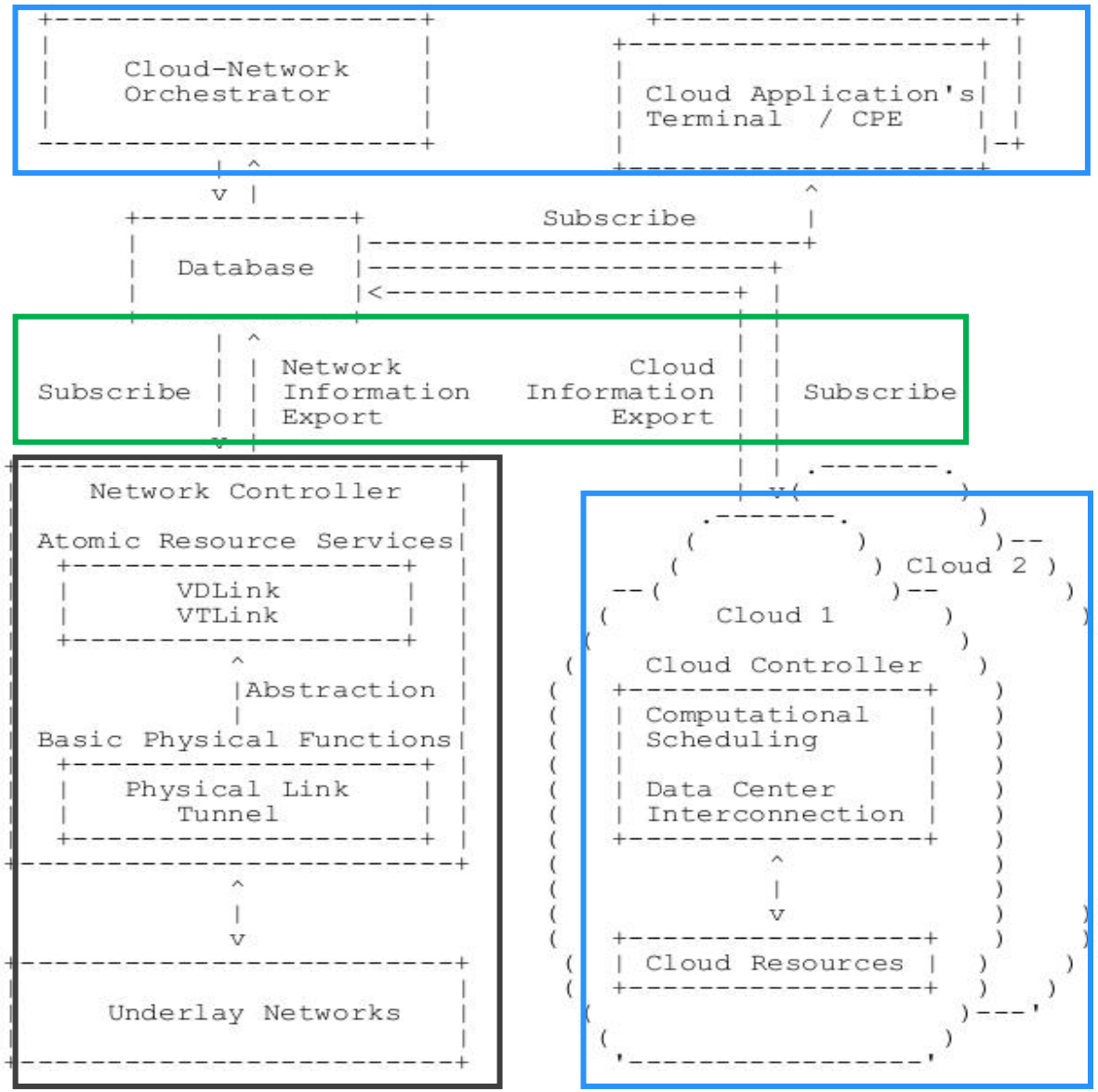
Framework of DB-ORS

Service subscription

- Programmability of DB-Agent
- Efficiency of a ‘watch’ mechanism
- Simplicity of API and Extensibility

Service abstraction

- VDlink
- VTLlink
- Node
- EPE Link



Service publication

Distribute database,e.g ETCD:

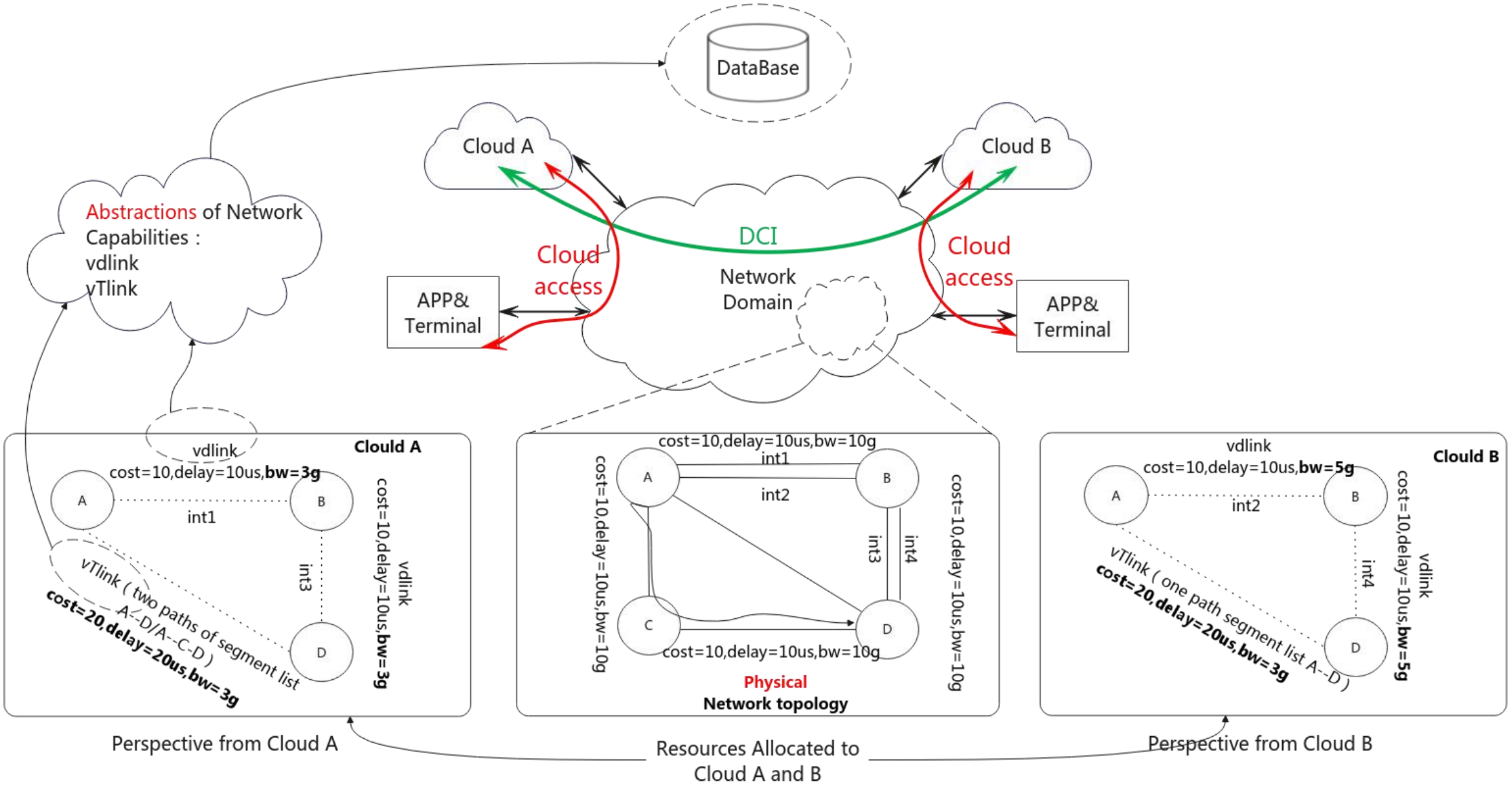
- Key-value model
- Standard schema template

Service re-orchestration

- Application identification
- Network path calculation
- Service orchestration & binding

https://datatracker.ietf.org/doc/**draft-zhou-alto-dbors-framework/**

Usecase of DB-ORS： abstract network capabilities



Usecase of DB-ORS： publish and subcribe services

Perspective of Cloud A：

Link	Link type	Logic Link ID	cost	delay	max bw	END.X/INT	BSID（segmentlist）
A----B	vdlink	536870913	10	10us	3g	3e::1/int1	--
B----D	vdlink	536870914	10	10us	3g	3e::2/int3	--
A----D	vTlink	805306369	20	20us	5g	--	4BA::1（sl1:A--Dd sl2:A--C--D）

Perspective of Cloud B：

Link	Link type	Logic Link ID	cost	delay	max bw	END.X/INT	BSID（segmentlist）
A----B	vdlink	1073741825	10	10us	5g	3a::2/int2	--
B----D	vdlink	1073741826	10	10us	5g	3a::2/int4	--
A----D	vTlink	1342177281	20	20us	5g	--	4FE::1（sl1:A--D)

....

Key-Vaule in DataBase

Cloud_A/VTLink/Logic_id	536870913
Cloud_A/VTLink/Logic_id /linktype	vdlink
Cloud_A/VTLink/Logic_id /delay	10
....	

Stardard Schema Template

```
{
 "$schema": "http://json-schema.org/draft-07/schema#",
 "title": "Abstract definition of CN2-DCI network resources.",
 "description": "version v1.0.0",
 "properties": {
   "vTLink": {
     "description": "Logic top",
     "type": "array",
     "item": {
       "type": "object",
       "properties": {
         "id": {
           "description": "logic link id",
           "type": "string"
         },
         "delay": {
           "description": "link delay",
           "type": "string"
         },
         ......
}}}}}
```

Differences between DB-ORS & ALTO

	ALTO	DB-ORS
Scenario	Universal model of network capability exposure for various scenarios	Focus on abstraction of network capabilities, especially for the **integration of the cloud and the network**
Abstraction	Coarse-granularity, simple descriptions	**Fine-granularity**, explicit and diversified abstractions
Application	Endpoints selection in the **Application layer**	Traffic steering and redirection in the **Network layer**
Framework	C/S framework with a unique ALTO server	C/S framework with a distributed database as the server
API	Unified REST+JSON	Unified REST+JSON
Expandability	Easy to expand	Easy to expand

Conclusions and Considerations

What have we done & what do we expect?

- Solve the mentioned problems.
- More perceptions and drafts are expected in the future.
- Promote standardization, cooperate with working groups who have resonated with this issue.

Considerations in the future?

- More abstraction of network capabilities.
- Safety considerations.
- Service affinity.