Compute Resource Modeling Consideration

Presenter: Kehan Yao; yaokehan@chinamobile.com

draft-du-cats-computing-modeling-description

Zongpeng Du, Yuexia Fu, Cheng Li, Daniel Huang

duzongpeng@chinamobile.com, fuyuexia@chinamobile.com, c.l@huawei.com, huang.guangping@zte.com.cn

IETF116

One main problem in CATS

- In CATS (Computing-Aware Traffic Steering), the decision point would make "**a Traffic Steering decision**" considering both network and computing status
- However, as the decision point is a network node, a problem arises:
  - It is straightforward that the decision point, as a network device, can have the network status information by some means
  - But it is challenging for a network device to obtain the computing information
- To enable the Computing-Aware Traffic Steering decision in the network, we need to handle two related issues:
  - Clarify **what** computing information needs to be notified to the decision point and possibly its format (*the draft's motivation*)
  - By which means the computing information can be notified to the decision point

Computing Information Description

- However, differentiated computing capability is reflected in two aspects:
  - Computing capabilities are **various** in different service sites
  - the status of different service sites are **dynamic**

- **An efficient description of computing information** is needed

Computing Information Description (Cont.)

- The decision point needs to know which service site is the best
- The service site should have a suitable service capability, service capability can be expressed by different attributes(CPU/GPU processing speed, memory, host bandwidth, etc.), which are normally static values.
- the workload of service sites is dynamic, the service site can not be overloaded.

The computing evaluating system

- A straightforward way is to run the real service on the service point, and observe the throughput of service
  - For example, images / second for the AI Image Processing

- However, even for the same service, different clients may have different computing requirements, thus
  - In addition, some general capability test results can also be considered as the input of the final score

The computing evaluating system (Cont.)

- **Three levels** of computing information may be considered in the evaluating system
  - It is not to say that a service needs all the information in the evaluating system
  - It is suggested that a service can subscribe the information it cares

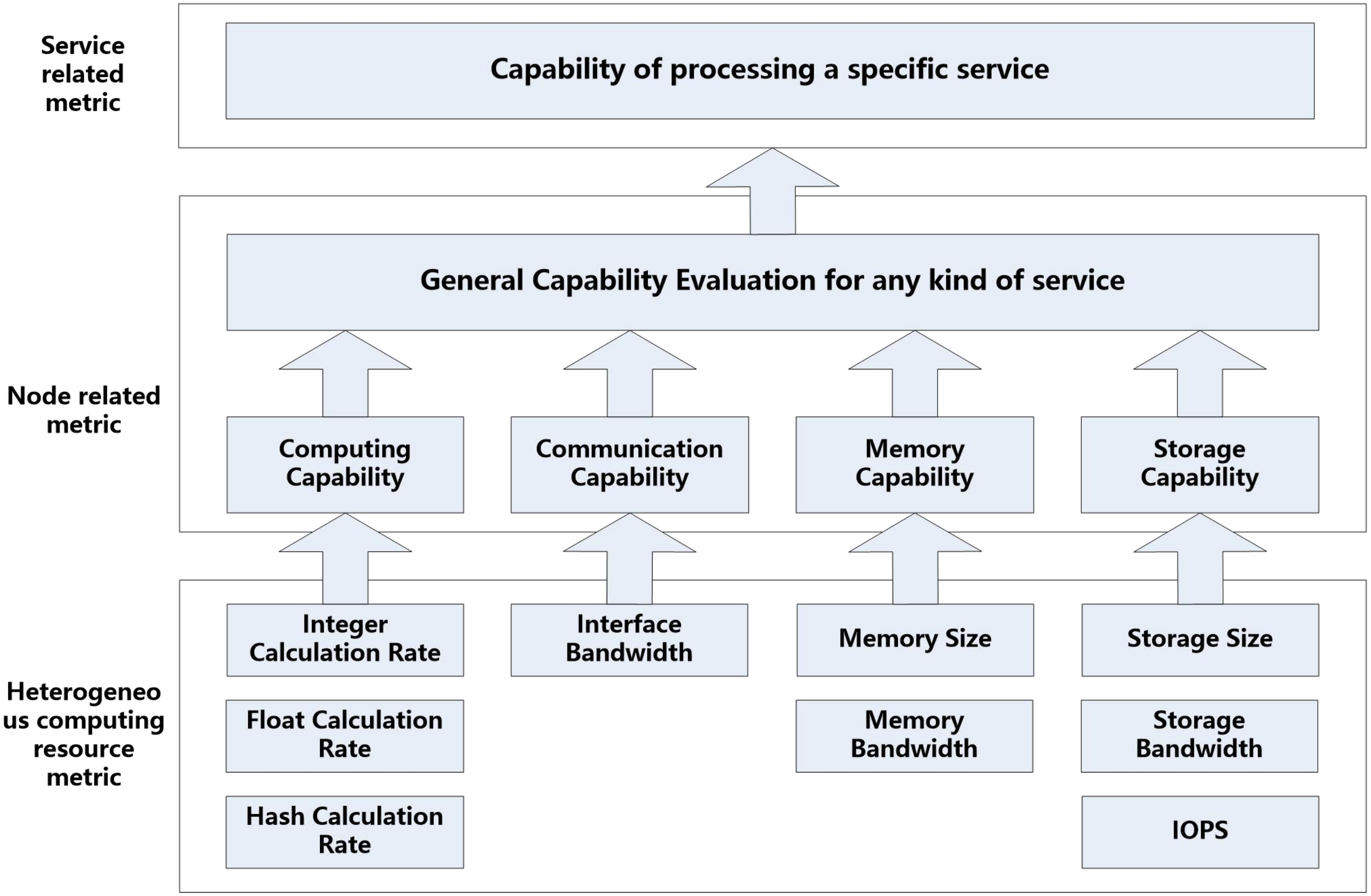
- The first level is about hardware heterogeneity to describe computing capability
  - The indexes of this level can be the performance parameters provided by the manufacturer, such as CPU model, main frequency, number of cores, GPU model, single-precision floating-point performance, etc.
  - Meanwhile, the indexes can also be the test values of commonly used benchmark programs

The computing evaluating system (Cont.)

- The second-level indexes are abstracted from the first-level indexes, which are mainly used for the comprehensive evaluation of node's computing capability
  - The indexes can provide the ability of a certain aspect of the node, such as in the aspect of computing, communication, cache, and storage, or a general comprehensive service ability of the node

- Level 3 indexes are related to the services deployed on the nodes
  - The indexes mainly provide service-related evaluation parameters, such as the actual processing throughput that nodes can provide for a specific computing service. It can also be a test value, but it is generated by running the real service

The computing evaluating system (Cont.)



Some other sections in the draft

- We introduce only part of the draft, focusing mainly on the section 5 Computing Resource Modeling
- Other information includes:
  - Usage of Computing Resource Modeling
  - Network Resource Modeling
  - Application demand Modeling
  - …
- The draft is still very initiative and welcome more discussions & contributions

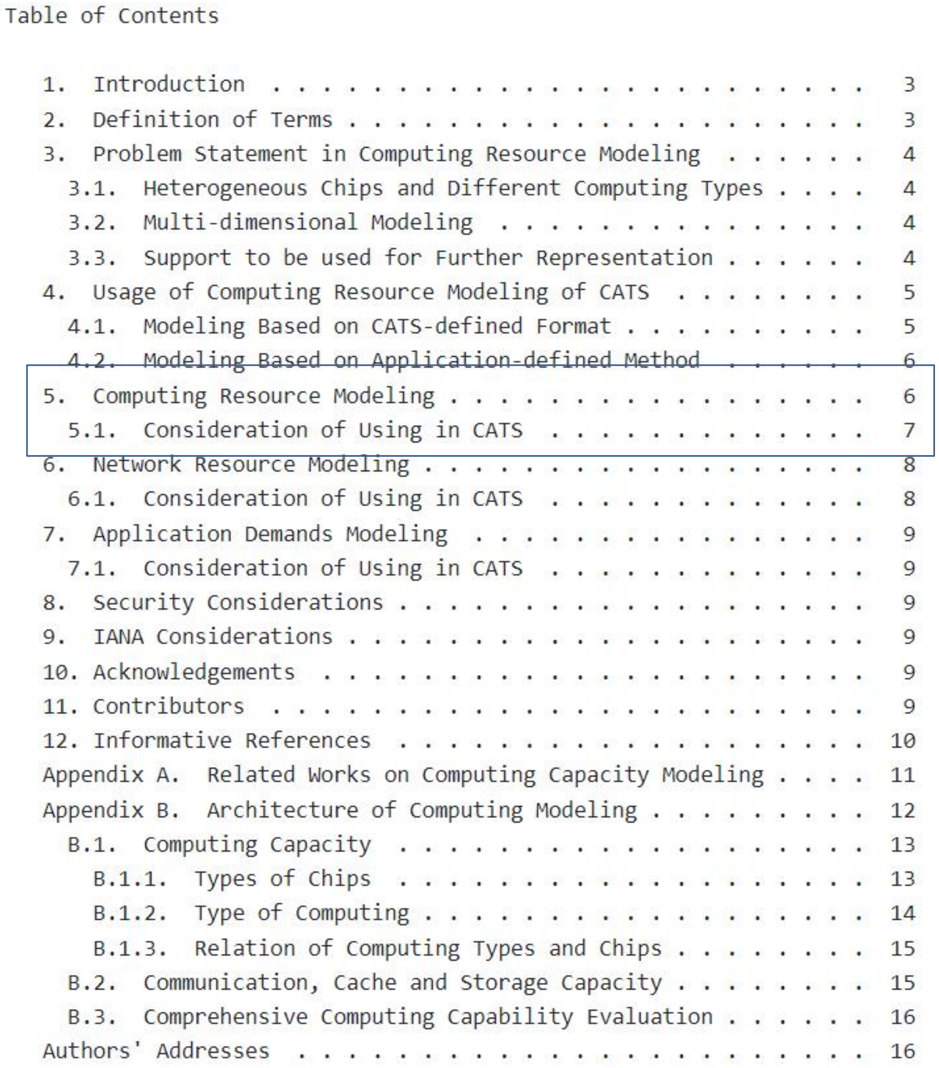


Table of Contents

Thanks !