Data center resource management for in-network processing

Marco Micera

March 12, 2020

Table of contents

1

Introduction

(2.4)

...RMs do not consider server and switch resources **conjunctly**...

Goals

(abstract)

1. Model and evaluate an API through which applications can ask for INP resources
2. Discuss the importance of a scheduler which can reject INP requests and propose their server-only equivalent when needed (e.g., high switch utilization)

Analysis

Currently existing In-Network Processing (INP) solutions

- In-network data aggregation
    - Properties (5.3.1). . .
    - Daiet, SHArP
- In-network storage
    - Properties (5.3.1). . .
    - NetChain, IncBricks

Aggregation protocol: SHArP

Coordination services: NetChain

Resource models (3.4)

- VC
  - Image
- VOC
  - Image
- TAG
  - Image

1. Fine-grained resource requests
2. High-level goals

(3.3)

(3.3.2)

FIXME Requirements? It is necessary?

Design

Composites

(Thesis repository `dock/thesis/figures/design/model/presentation.pdf`)

The extended-Tenant Application Graph (eTag)

- (5.2.1 why existing resource models do not satisfy all requirements)
- (5.2.2 eTag)

The template database

- (5.3 generic groups)
- (5.1.2 template database role)(5.1.2 template database role)

FIXME I should introduce composites earlier

Bibliography

Some references to showcase [allowframebreaks] [2, 3, 4, 1]

Conclusions

Questions?

Thank you

📄 R. L. Graham, D. Bureddy, P. Lui, H. Rosenstock, G. Shainer, G. Bloch,
D. Goldenerg, M. Dubman, S. Kotchubievsky, V. Koushnir, et al.
**Scalable hierarchical aggregation protocol (sharp): a hardware architecture
for efficient data reduction.**
In *Proceedings of the First Workshop on Optimization of Communication in HPC*,
pages 1–10. IEEE Press, 2016.

📄 X. Jin, X. Li, H. Zhang, N. Foster, J. Lee, R. Soulé, C. Kim, and I. Stoica.
Netchain: Scale-free sub-rtt coordination.
In *15th USENIX Symposium on Networked Systems Design and Implementation
(NSDI 18)*, pages 35–49, Renton, WA, 2018. USENIX Association.

M. Liu, L. Luo, J. Nelson, L. Ceze, A. Krishnamurthy, and K. Atreya.
Incbricks: Toward in-network computation with an in-network cache.
In *Proceedings of the Twenty-Second International Conference on Architectural Support for Programming Languages and Operating Systems*, ASPLOS '17, pages 795–809, New York, NY, USA, 2017. ACM.

A. Sapio, I. Abdelaziz, A. Aldilaijan, M. Canini, and P. Kalnis.
In-network computation is a dumb idea whose time has come.
In *Proceedings of the 16th ACM Workshop on Hot Topics in Networks*, HotNets-XVI, pages 150–156, New York, NY, USA, 2017. ACM.