Beehive: Simple Distributed Programming in Software-Defined Networks

Soheil Hassas Yeganeh†
(Google Inc., University of Toronto)
Yashar Ganjali
(University of Toronto)

SOSR 2016

Michael Franz
SDN Seminar 2017
Abstract

- **Beehive**: Distributed control platform
  - Simple programming model
  - Asynchronous message handlers
  - State stored in dictionaries
- Centralized application → distributed system
- Runtime instrumentation
- Dynamically optimizing control plane
- Fault-tolerant
- Open source
- SDN controller on top
  - 200K OpenFlow

→ No external data store
Programming Model

A look under the hood
Programming Model
Programming Model

- single-threaded message handlers
  - general purpose programming language
  - triggered by asynchronous messages
  - can emit further messages

- dictionaries are transactional, replicated and persistent

- Inter-Dependencies
  - Messages
  - Dictionary access

- Consistent Concurrency
  - Centralized → distributed
  - State synchronisation
Control Platform

What about the *bees* and *hives*?
Control Platform

- **Hive:** controller instance

- **Bee:** light-weight thread of execution
  - Owns **cells** (keys)
  - **Colonies:** Replication
Control Platform

- Find the corresponding *bee* for a message handler
  - *Map* function
    - Automatically generated by compiler
    - Used keys
  - *Queen bee*: message router
Control Platform

- Transactions
  - Rcv functions transactional
  - Dictionary modifications
  - Emitted messages

- Runtime Instrumentation
  - Resource consumption
  - Exchanged messages

- Optimization
  - Automatic placement
  - Migrate bees
SDN Control Platform

Can do – Kandoo
SDN Control Platform

- SDN Controller Suite
  - Network object model (NOM)
  - Triggers
  - Paths

- Pollers
  - Consolidator: Discrepancy
  - Monitor: Triggers
Fault-Tolerance

https://github.com/kandoo/beehive-netctrl
Fault-Tolerance

(a)

(b)

Latency (ms)

Disconnected

Optimized

Time (sec)
Scalability

![Scalability Graph]

**Replication**
- Beehive-0
- Beehive-1
- Beehive-3
- Beehive-5
- ONOS-0
- ONOS-1
- ONOS-3
- ONOS-5

**TPut (msg/sec)** vs **Number of Nodes**
Discussion
Discussion / Related Work

- :-) Analogy / pictorially
- :-) Simplicity
- :-) Instrumentation / automatic optimization
- :-) scalability graph

- Raft (Election / Consensus)

- General-purpose programming language
- Persistence / Consistency
- Beehive vs. ONOS
- Automatic placement (Greedy heuristic)
Thank you …

- Questions?