Open vSwitch

December 10-11, 2019 | Westford, MA

DDlog in OVN
Current State and Future Challenges

Mark Michelson, Dumitru Ceara, Red Hat

What is DDlog? Why does OVN need it?

Differential Datalog (DDlog)

- *bottom-up, incremental, in-memory, typed Datalog engine* [1]
- automatic incremental behavior
- compiles into Rust
- uses Rust for utility functions and standard library

OVN (ovn-northd) processes NB/SB databases completely (with every change!)

- does this scale (lots of recomputation without changing outputs)?
- can we replace the DB processing with DDlog? Yes, we could [2]

[1] https://github.com/vmware/differential-datalog

[2] https://github.com/ovn-org/ovn/tree/ddlog-dev-v2

Dev story #1 - IP Multicast in ovn-northd-ddlog

The “good”:

- resulting code is more concise, the developer can focus on the functionality. Basic metric (LoC): ~650 LoC in DDlog vs ~1778 LoC in C
- (some) things are easier to implement: e.g., stable ID allocation for multicast groups
  - no need for IDL lookups and such.. it’s incremental, right?
- DDlog feature requests/bug reports get immediate attention from DDlog maintainers: e.g., debug options to dump intermediate relations, ovsdb2ddlog enhancements

The “bad”:

- new language, different programming paradigm. Good topic coverage in documentation [1]. Some hard-to-spot bottlenecks for DDlog beginners [2].
- debugging is mainly offline; analyzing DDlog relations (tables) dumps.

[1] https://github.com/vmware/differential-datalog/blob/master/doc/tutorial/tutorial.md

[2] https://github.com/ovn-org/ovn/pull/28#discussion_r348680499

Dev story #2 - Dynamic MAC for static IPv6

The “good”:

- DDlog tutorials were enough to get the job done
- error messages for DDlog were clear and made it easy fix mistakes

The “bad”:

- the feature implementation involved writing more Rust code vs DDlog.
- ended up not really getting to see the benefits of DDlog
- compilation takes long
- never managed to get the ovn-northd-cli to compile properly; a unit test still failing
- therefore, the change never got to be submitted for review

Highlights/Lowlights

The “good”:

- simple translation of input to output relations; focus is on the data itself
- automatic incremental updates
- stronger type system than C; safer memory usage
- ovn-northd-ddlog performs better than the C version [1] [2]

The “bad”:

- “unproven” new language
- tooling is minimal, especially for debugging (a live DDlog debugger would be great)
- standard library is ad-hoc and quite minimal
- no DDlog v1.0/stable release yet

[1] https://mail.openvswitch.org/pipermail/ovs-dev/2019-July/360876.html
[2] https://mail.openvswitch.org/pipermail/ovs-dev/2019-July/360889.html

Challenges for conversion to DDlog

- ovn-northd (C) is a moving target that often gets new features/patches; these should also be ported to ovn-northd-ddlog!
- who does the porting?
- the contributor must learn DDlog on top of understanding the rest of OVN code
- ovn-northd-ddlog uses a custom ovsdb client to push updates from DDlog to the NB/SB databases; code duplication and missed edge cases (e.g., transaction failure reporting). Should we use IDL instead?
- is ovn-northd the only bottleneck?
  - ovn-controller also has incremental processing (C)
  - maintain two incremental processing approaches or choose one?

Thanks!