OpenFlow 1.4

(Changes compared to 1.3 – OpenDaylight Perspective)

- Abhijit Kumbhare

More extensible wire protocol

- OpenFlow Protocol initially designed w/ many static fixed structures
- OXM (TLV format) added extensibility to match fields
- In 1.4 – many other parts retrofitted with TLV structures
  - allow a much easier way to add new features to the protocol in the future,
  - greatly extend the Experimenter Extension API.
- Probably a major change all over the place – spec cleanup in some ways

More extensible wire protocol (contd)

- New TLV structures
  - Port structures: add port description properties, add port mod properties and add port stats properties
  - Table structures: add table mod properties, add table descriptions multipart, add table status asynchronous message
  - Queue structures: migrate queue description to multipart, convert queue description properties to standardised TLVs, add queue stats properties
  - Set-async structures: convert set-async-config to TLVs, add set-async experimenter property
  - Instruction structures: clarify instruction TLVs
  - Actions structures: clarify actions TLVs
  - Experimenter structures: clarify experimenter TLVs
  - Properties errors: add a set of unified error codes for all properties

More descriptive reasons for packet_in

- Different parts of the pipeline use same reason
- Detailed reasons help debug pipeline
- New reasons :

```
/* Why is this packet being sent to the controller? */
enum ofp_packet_in_reason {
    OFPR_TABLE_MISS = 0, /* No matching flow (table-miss flow entry). */
    OFPR_APPLY_ACTION = 1, /* Output to controller in apply-actions. */
    OFPR_INVALID_TTL = 2, /* Packet has invalid TTL */
    OFPR_ACTION_SET = 3, /* Output to controller in action set. */
    OFPR_GROUP = 4, /* Output to controller in group bucket. */
    OFPR_PACKET_OUT = 5, /* Output to controller in packet-out. */
};
```

- Need to study the impact of handling individual packet in reasons

Optical Port Properties

- New set of port properties to add support for Optical ports to OpenFlow
  - Optical port mod property ofp_port_mod_prop_optical to configure optical ports
  - Optical port stats property ofp_port_stats_prop_optical to monitor optical ports
  - Optical port stats property ofp_port_stats_prop_optical to monitor optical ports
- Minor Change from the controller perspective??

Flow-removed reason for meter removal

- Meter removal
  - All flows pointing to meter are removed
  - Identical to group removal
  - May generate some flow removed messages
  - No flow removed reason in 1.3 !
- Change for 1.4
  - New flow removed reason
  - OFPRR_METER_DELETE
- Minor Change

Flow monitoring

- Allows a controller to monitor in real time the changes to any subsets of the flow table done by other controllers
- Notifications for flow table changes
  - Per controller flow monitors (on the switch)
  - Flow monitor matches flows of interest
  - Generate flow notifications to controller
  - Events message with optional abbreviation
  - Buffer management with pause/resume
- Multi-controller feature
- More than a minor change; but cool, huh?

Role Status Events

- Role change notification
  - Inform a controller when demoted to slave
  - Optionally generated for experimenter causes
- Role status event
  - Similar body as role reply
  - Includes Role + generation_id
  - TLV at the end for experimenter extension
- Multi-controller feature
- Perhaps a minor change after multi-controller support added; fixes a bug in 1.3 (not informing when role changed)

Flow entry eviction

- Extension for 1.3 & proposal for 1.4
- Configure flow entry eviction
  - New messages : set, get request, get reply
  - Per-table configuration, on/off boolean
- Flow importance
  - Encoded as experimenter instruction, per flow
  - Optional hint for eviction algorithm
- Eviction process
  - Entirely switch defined
  - Report flows with reason OFPRR_DELETE
  - Flags in table desc to describe eviction criteria
- Likely a minor change on the OpenFlow Plugin side – from a system perspective (app-controller-plugin-switch) needs flow prioritization
- Cool feature in conjunction with Vacancy Events
- (Should try to support anyway)

Vacancy Events

- In 1.3 – abrupt behavior once switch flow table gets full
  - New flow entries not inserted – error returned
  - Likely disruption of service
- Provides a mechanism enabling the controller to get an early warning based on a capacity threshold chosen by the controller
  - Allows controller to react in advance and avoid getting the table full
  - New table status event with reasons VACANCY_DOWN & VACANCY_UP
  - Table-mod vacancy property to set vacancy thresholds
- Likely a minor change on the OpenFlow Plugin side – from a system perspective (app-controller-plugin-switch) needs flow prioritization
- Cool feature in conjunction with Flow Entry Eviction
- (Should try to support anyway)

Bundles

- Bundle mechanism enabling to apply a group of OpenFlow message as a single operation
  - Enables the quasi-atomic application of related changes
  - Enables synchronization of changes across a series of switches
- Bundle control message to create, destroy and commit bundles
- Bundle add message to add an OpenFlow message into a bundle
- Bundle error type to report bundle operation errors
- Likely a minor change on the OpenFlow Plugin side – from a system perspective would enable cool applications
- Probably the coolest 1.4 feature
- (Should try to support)

Synchronized tables

- How to express two lookup using same data
  - L2 learning lookup and L2 forwarding lookup
  - RPF check use L3 forwarding data for multicast
- Simplest solution for 1.4
  - Just identify table as “synchronized” + table-id
  - Flow add/remove reflected in synchronized table
  - Actions may be different
  - Match may be transposed
  - Don't express how synchronization is done (hard)
- Likely a minor change on the OpenFlow Plugin side – not sure as to the utility
- (Should support since this may be a minor change)

Group & meter change notifications

- Group and meter notifications
  - Allow a controller to monitor in real time the changes to the group table or meter table done by other controllers
  - Just relay group-mod and meter-mod requests
  - Encapsulate request to other controllers
  - For multi-controller and hybrid
  - Need to think through the scope of changes needed

LPM table support via bad priority error message

- Restrict flow table to Longest Prefix Match
  - Express L3 routing tables
  - Popular hardware feature – simpler than TCAM
- Done by
  - Adding new bad priority error message OFPFMMC_BAD_PRIORITY
- Restrictions
  - Masks need to be contiguous starting with the low order bit
  - Priorities need to be consistent with the masking
- Switch side feature – controller can always enforce LPM support by adding flows to be consistent with the masking
- Nothing to be done here for ODL

Error code for Set-async-config

- New errors codes for OFPT_SET_ASYNC

```
/* ofp_error_msg 'code' values for OFPET_ASYNC_CONFIG_FAILED.
* 'data' contains at least the first 64 bytes of the failed request. */
enum ofp_async_config_failed_code {
    OFPACFC_INVALID = 0, /* One mask is invalid. */
    OFPACFC_UNSUPPORTED = 1, /* Requested configuration not supported. */
    OFPACFC_EPERM = 2, /* Permissions error. */
};
```

- Looks to be a switch side feature for Multi-controller environments – nothing to be done here for ODL

PBB UCA Header Field

- A new OXM field OFPXMT_OFB_PBB_UCA has been added to match the “use customer address” header field from the PBB header
- Minor change

Add error code for duplicate instruction

- Flow entry has an instruction set
  - Only one instruction of each type
  - If duplicate instruction, what error ?
- New in 1.4
  - Detect duplicate instructions
  - Implement new error codes
  - New error OFPBIC_DUP_INST
- Switch side feature
- Nothing to be done here for ODL

Multipart reassembly timeout

- Multipart request and reply sequence
  - End sequence : more-bit clear in header
  - What if more-bit never cleared ?
- Specify min amount time to wait for more-bit (and generate timeout)
  - Switch : 100ms
  - Controller : 1s
- New error messages
  - Optionally sent if timeout expired, no body
  - OFPBRC_MULTIPART_REQUEST_TIMEOUT
  - OFPBRC_MULTIPART_REPLY_TIMEOUT
- Minor change

Change Default TCP Port to 6653

- Already done