

# Blackbird Planning

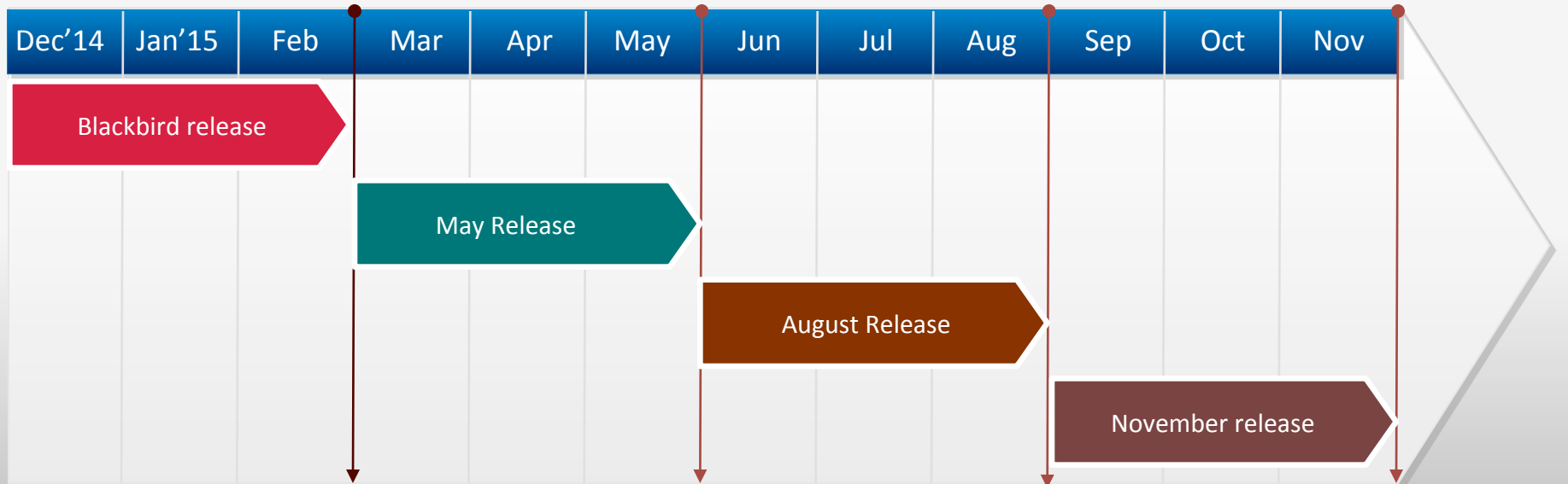
13 January 2015

Bill Snow, Thomas Vachuska, Suibin Zhang and  
Prajakta Joshi  
Partners in crime



# UPCOMING ONOS RELEASES

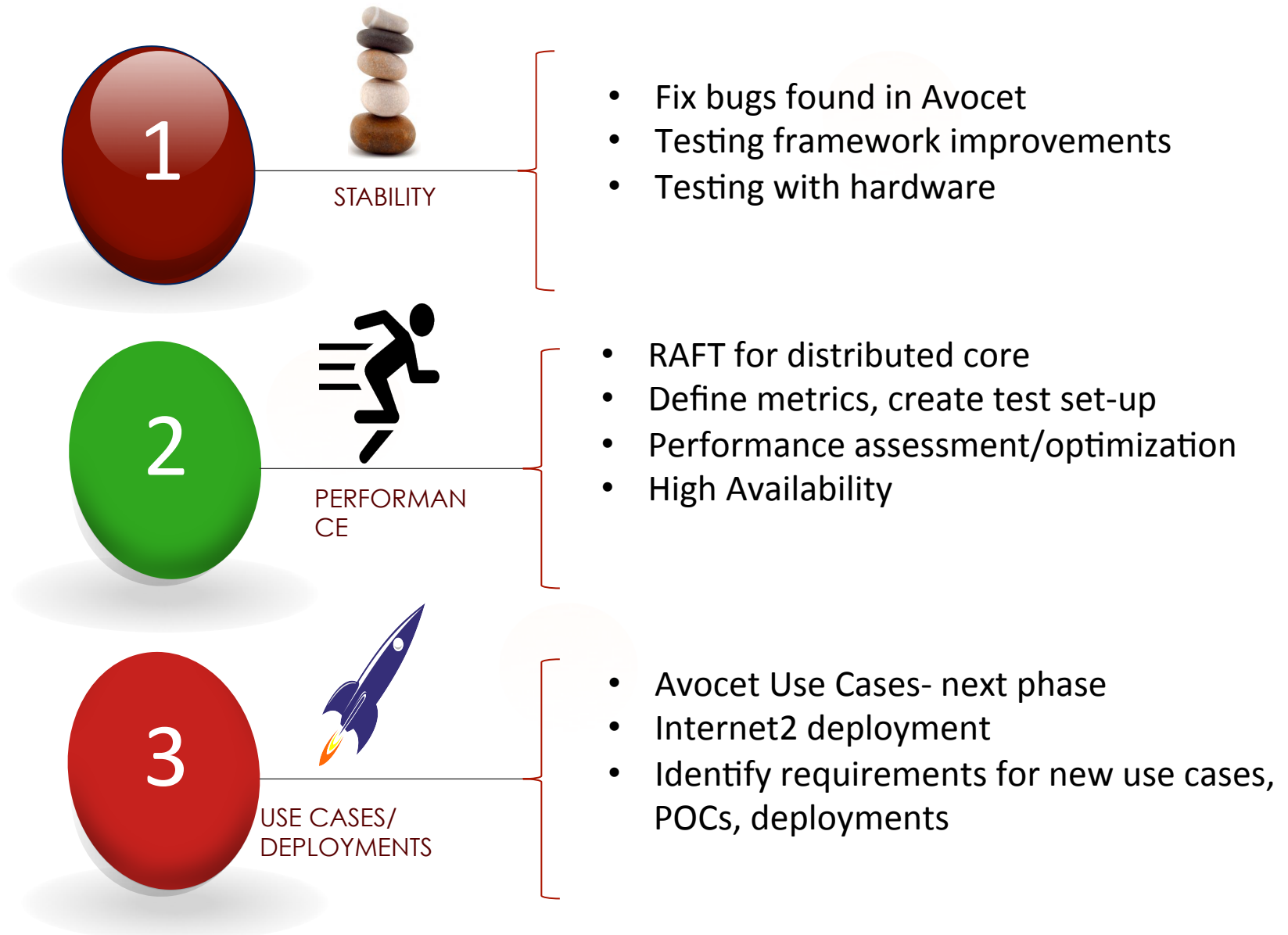
2014/15



## ONOS Releases

- Regular 3 month release cadence
- Named after birds in alphabetical order (Avocet, Blackbird...)
- Community release planning event at the start of the release cycle
- Everything tracked in JIRA ([jira.onosproject.org](http://jira.onosproject.org))
- Agile development at ON.Lab + freedom to choose whatever process works for you

# Blackbird Release (28<sup>th</sup> Feb 2015)



# Timelines

- This week
  - 1<sup>st</sup> of four 2 week sprints
    - 1/5-1/16, 1/19-1/30, 2/2-2/13, 2/16-2/27
- Week of 1/26 – May release planning session
- 2/15 Blackbird code freeze
- 2/28 Blackbird release
- 4/1 May release feature freeze
- 5/1 May release code freeze
- 5/30 May release

# Priorities and leads

- Blackbird (Feb)
  - Bugs (**Suibin**/all)
  - Performance and Scale – **Tom/Suibin**
    - NB – **Brian**, Sho, Ray
    - Core – **Madan**, Jono, Ali
    - SB – **Ali**, Ayaka
  - SDN-IP at Internet2 – **Luca**, Pingping, Pavlin
  - UI improvements – **Simon**, Bri
  - Test improvements – **Suibin**, Hari, Shreya, Jon, Cameron, Kelvin, Andrew
- May Release
  - OF 1.3 data pipelines – **Ali**, Saurav, Sangho, Srikanth
  - IPv6 - **Pavlin**
  - Plug in examples
    - OVSDDB
  - NB, Core, SB enhancements TBD
  - Use case features+infra support for them
    - AT&T IP Optical – **Marc**, Tom T, Wei, Simon
    - NTT Optical – NTT/NEC – Yuta, Toru, Naoki
    - NFaaS – Luca? Definition phase
    - IP RAN – Huawei **Patrick**, Hongtao
    - IoT – NEC – Toshi, Teru
    - DREAMER – Matteo, Michele
    - DirectTV - TBD



Perf/Scale-out Characterization  
Status V1.0 Release

- ❖ Perf/Scale-out was not the target on the release
- ❖ Only results obtained were on small-scale network latency
  - ✧ <https://wiki.onosproject.org/display/ONOS/Test+Results+-+Perf+and+Scale-out>
- ❖ A number of factors affected the tests:
  - ✧ Bugs affected Intent testing (ONOS-384,389)
  - ✧ Network Performance Test: limited by some OVS behaviors (min. switch back-off timer;)

## Targets for Blackbird release

**Env:** metrics tested on 1-node, 3-node, 5-node, 7-node cluster; nodes are bare-metal, dual-Xeon, 10G Network.

### **Target: Produce a Reasonable ONOS Performance Baseline**

#### ❖ Network Performance

- ✧ Latency to discover switch, port connected to ONOS1
- ✧ Throughput (# of network events per second)
  - ✧ Implement and Use “Null Topo Provider” as network event generator

#### ❖ Intent Performance

- ✧ Latency to “install”, “re-route” and “withdraw” intents from ONOS1 with various of batch sizes
- ✧ Throughput (# of intents install/withdraw per second)
  - ✧ Use “demo installer” as intent load generator
  - ✧ Implement and use “Null Device Provider” to bypass switch performance bottleneck

# Blackbird Performance Themes

- Demonstrate scale-out effect
  - Intent subsystem will be used to demonstrate that as load scales up, performance can be maintained by adding more ONOS instances
- Demonstrate low-latency operation
  - Topology subsystem will be used to demonstrate system ability to timely respond to topology change events
- Develop partitioned RAFT implementation
  - Multi-consensus ring implementation will be used as a basis for strongly consistent store semantics while avoiding the pitfall of degraded performance as the size of cluster grows



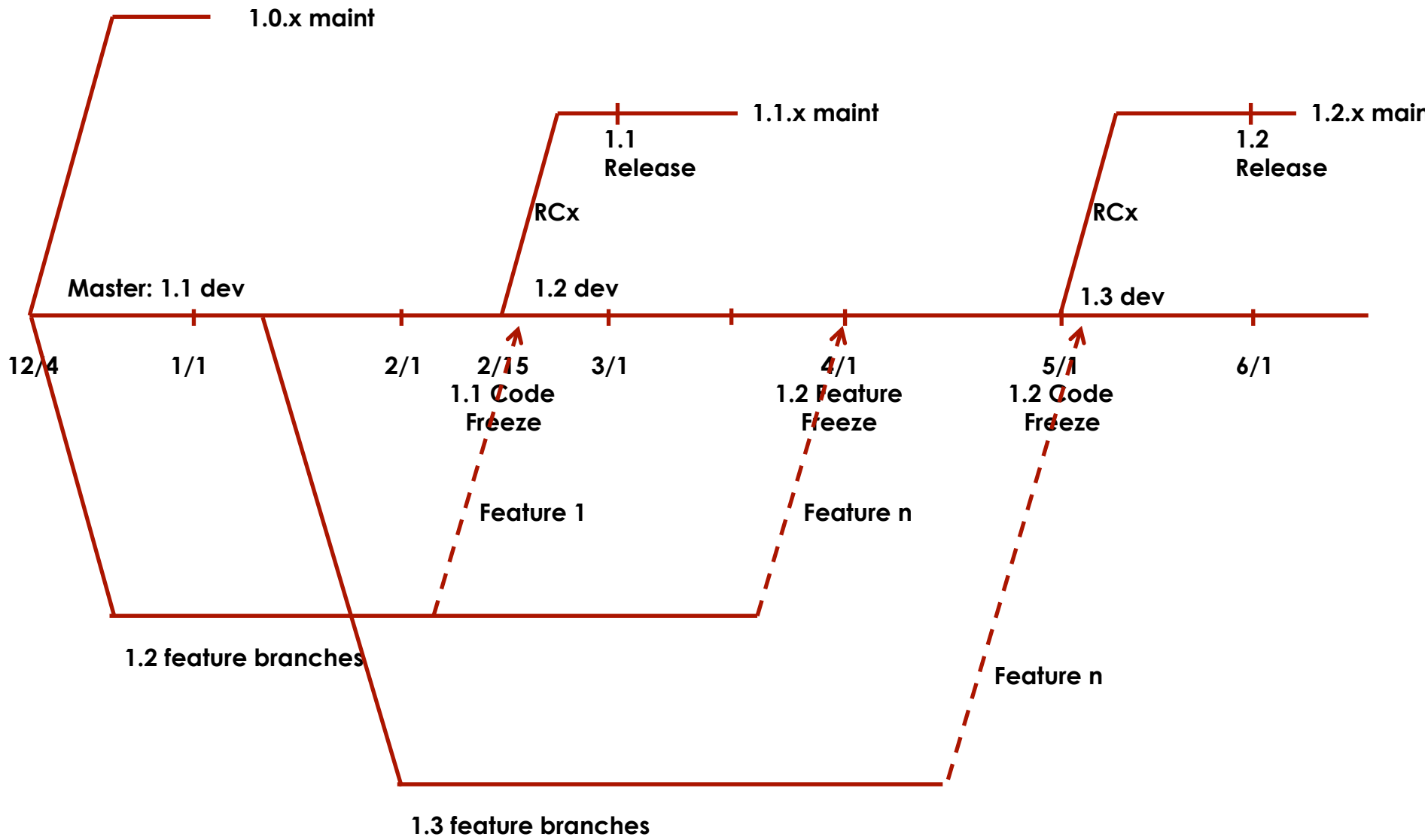
# Performance Focus Areas

- NULL providers
  - Used to remove any friction from the device I/O and instead put stress on the ONOS subsystems, specifically the distributed stores
- Flow & Intent subsystems
  - Enhance *IntentService* API to support explicit batching & enable more parallelism
  - Enhance *FlowRuleService* API to support explicit operation ordering
  - Refactor *IntentManager* implementation to decouple N2S & E2W work streams
  - Refactor *IntentStore* implementation to be eventually consistent & fully replicated
  - Refactor *FlowRuleManager* implementation
- Topology subsystem
  - Eliminate pre-computing all shortest paths as part of topology processing
- Partitioned RAFT implementation
  - Implement partitioned RAFT implementation as a basis for durable and strongly consistent stores
  - (Stretch goal) Implement *LinkResourceStore* using the new RAFT-based store

Perf/Scale-out Characterization  
Plan for Blackbird Release

**Main Stories:**

- ❖ First-Pass Code Analysis, Tuning on NB+ Core, SB for Performance (Sprint2)
- ❖ Finalize Test Methodology (Sprint2)
- ❖ Complete Test Scripts (Sprint3)
- ❖ Iterations of Test-Profiling-Tuning-Retest cycles (starting Sprint3)



# Code Contribution Methods

- Small submission – feature or bug fix
  - Follow the process on the wiki
- Large submission
  - Process under definition – but some important points:
    - Must get agreement ahead of time from technical steering team
    - Must align with ONOS architecture/direction – technical steering team
    - Must conform with software design / abstractions – technical steering team
    - Must include unit tests, unit tests must all pass
    - Must pass all sanity and static checks
    - Must be a support resource committed to it
    - Javadocs can be created
    - Can not adversely affect performance, scale, HA
    - Should have CLI / debug support
    - Code reviews must pass
    - Can enter release train after meeting the feature freeze date and all above
- Work directly with ON.Lab
  - Contact project lead

# Governance Update

- Technical Steering
  - Thomas Vachuska
  - Regular meetings to be set up for review of design/architectural issues
- Use Case Steering
  - Tom Anschutz
  - Regular meetings to be set up for review of use cases and their priorities
- Release Steering
  - Bill Snow
  - February release (Blackbird)
  - May release planning session to be set up for 2 weeks from now
- Community Steering
  - Prajakta Joshi
  - Regular meetings to be set up for review of community issues