OpenDaylight Project

TAC Project Health Review

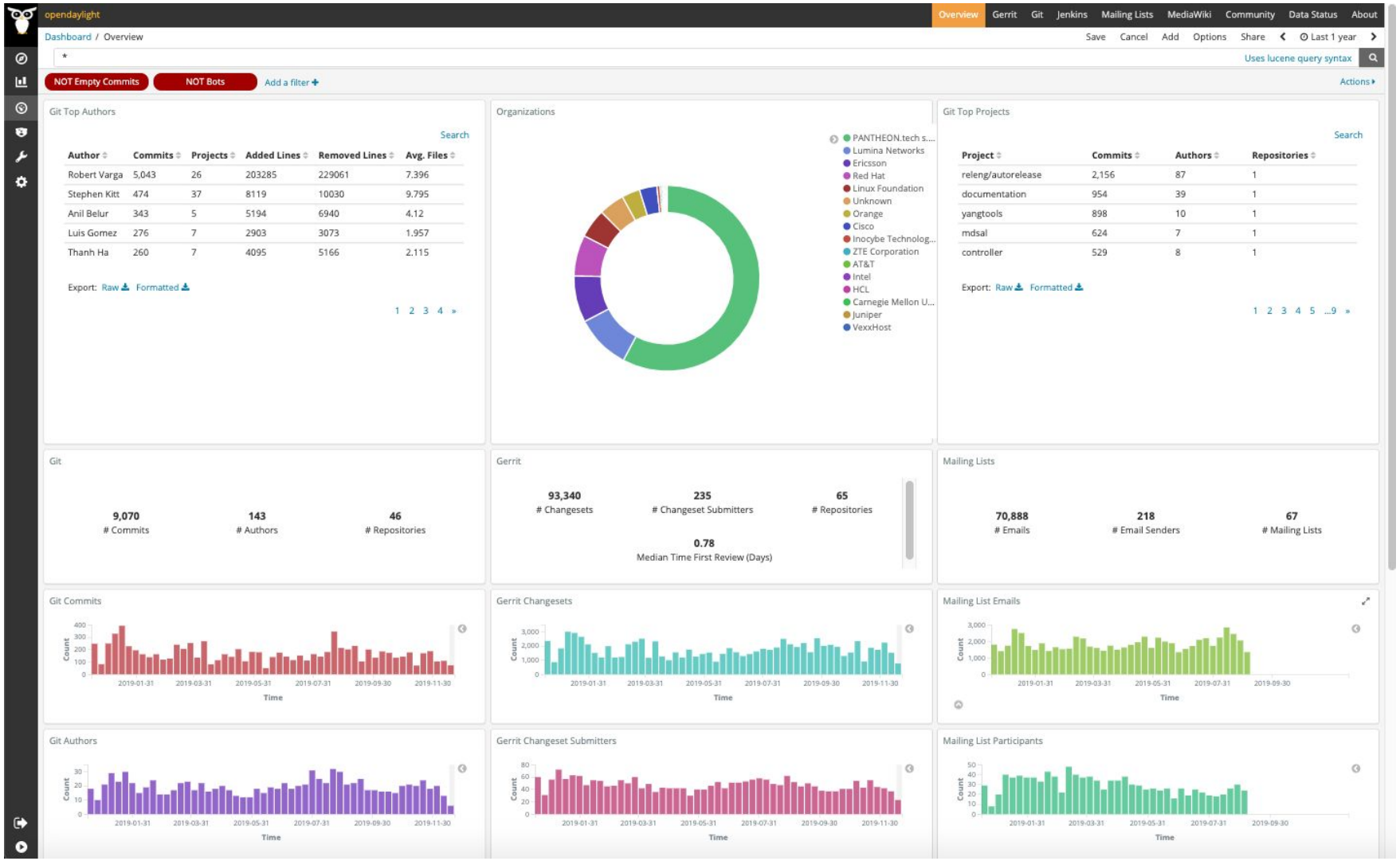
Dec 18, 2019

THE LINUX FOUNDATION

OpenDaylight - Summary

- **Project Functionality**:
  - OpenDaylight project is an open source platform for Software Defined Networking (SDN) that uses open protocols to provide centralized, programmatic control and network device monitoring.
- **Latest Release:**
  - Sodium : September 2019
    - S3P, technical debt reduction, multiple new features in the areas L3 VPN, EVPN, Neutron Integration, Tap as a Service, OpenFlow Tunnels, auto tunnels, CoE, etc.
  - Magnesium (upcoming) : March 2020
    - Java 11, Micro Distribution (initial work)

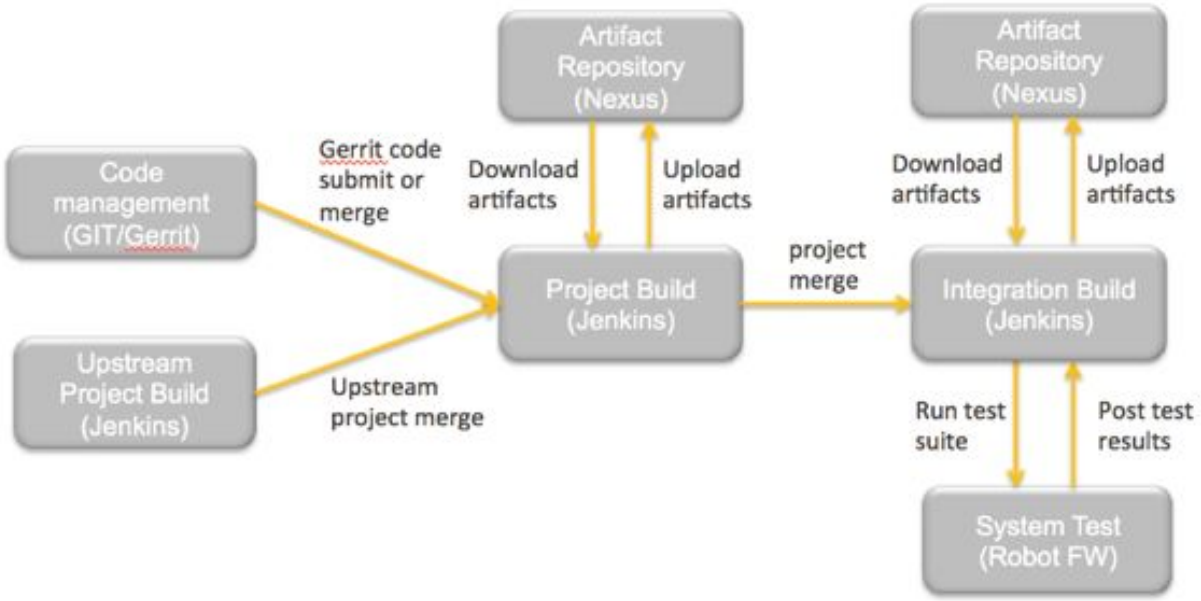
Opendaylight - Community Health



Last one year stats show:

- 143 authors, 9070 commits across 46 repositories
- Contributions from several organizations - but increasingly more from the top 3 (Pantheon, Lumina & Ericsson) and particularly from the top contributor.
- Total Git commits lower than an year ago - but reasonably steady
  - Note: Bitergia has not been updated to track emails from the new email system - groups.io (Sept 2019 onwards).
- Source of info: https://opendaylight.biterg.io/app/kibana#/dashboard/Overview?_g=()

OpenDaylight - Infrastructure Tooling



- The latest LFN Infra "IT Point of View" had noted that the migration for OpenDaylight would be particularly difficult.
- From ODL's point of view this needs more exploration
  - Some job templates are designed to work with both github & gerrit (from global-jjb). Others may need more work/testing (releng/builder)

- Beyond the tools stated above for bug tracking - ODL uses JIRA (previously bugzilla); for Wiki migrating to Confluence and for mailing lists groups.io (previously mailman)

LFN Readiness Summary - CI/CD transformation

IT Point of View of level or readiness – this is what the Infra WG should help "validate"

LFN Project	Readiness for Infra / Tools / CI/CD Transformation	Development Area	Readiness Level
OPNFV	**Probable**: Hardware lab centric; desire to mature virtualization capability relative to their lab configurations. New CI platforms Azure Pipelines (AZP) and Gitlab CI (GCI) don't cater to their emphasis on multi-node testing. May be possible to use GCI or AZP to do 'lead node' builds that then bring up virtual labs in a connected cloud – requires project alignment and development time. Primarily Python, no Nexus environment.	Project alignment and discovery; relevance	
ONAP	**Relevant**: GCI or AZP would be good candidates given project desire to move a percentage of the build infra to containers instead of VMs. Potential problem with build node sizes being "free". Many builds require heavier weight instances and build management is premium to this project. Would require negotiated paid tiers or continue existing on low cost and LF managed Open Stack hosting in Vexx or a hybrid configuration to keep costs manageable.	Container competencies; cost analysis	
ODL	**Relevant / Difficult:** Virtually all builds could be handled by either GCI or AZP, provided that the build instances available in the free pools have enough resources. All integration testing is highly dependent on OpenStack configuration due to development around multi-node testing. ODL has jobs that are by design 24+ hour for stability testing. AZP "unlimited" build time jobs are capped at ~6 hours. Possible approach: limited move of builds to AZP (or GCI). Integration testing continues on Jenkins / OpenStack initially, then transform them from current "freestyle" job design (global-jjb) to new Jenkins pipeline library support – this gives benefits of being pipeline based CI/CD.	Project alignment; hybrid setup to enable pipelines	
Fd.IO	**Most Likely**: The most active sub-project already using containers for builds. Unmanaged releases. Probably they could transition to either GCI or AZP. With either of these platforms they could add in virtualized resources, or their own hardware labs, as needed.	Project alignment	
Tungsten Fabric	**Uncertain**: Currently managing their own Zuul deployment which requires they remain on Gerrit. Zuul does not support GitHub as an SCM source and only supports Gerrit.	n/a	

OpenDaylight - Cross-Project Interaction

- ONAP has dependencies on OpenDaylight
  - ODL & ONAP have had couple of meetings together in the last 2-3 months (at the DDF and one after that)
  - An outcome of the meeting was for ODL to provide ONAP with an ODL distribution tailored to ONAP

- OPNFV and ODL
  - In the early days some use cases were end-to-end system tested in OPNFV

Feedback from OpenDaylight to LFN

- What's working well that others should know about?
  - Overall the transition to LFN from being a standalone project in LF has been smooth.
- Where is help needed?
  - More developers from LFN companies not currently participating in OpenDaylight
  - ONAP community uses ODL and it would be nice to see more upstream ODL contributions from ONAP community in strengthening ODL in areas needed by ONAP
- What would do differently over past year?
  - Take advantage of LFN cross project collaboration events