OpenDaylight Project Status

Date

24 Jun 2020

Attendees

- Abhijit Kumbhare
- Daniel De La Rosa
- Jamo Luhrsen
- Luis Gomez
- Robert Varga
- Many others including Daya Kamath, Karthikeyan, Allan Clarke , Jeff Hartley , Arunprakash D , etc.

Overview

- Schedule: https://teamup.com/event/show/id/DYAKPs5RiMDPg6J2mAc1pj1yVfXzMn
- Recording: ODL Project Status.mp4

Minutes

<Misc notes>

General Points

- Casey Cain says we should have concise info to ask companies for help.
- Charles Eckel feels that it maybe useful to have a Zoom session or two to explain.
- Robert Varga and Jamo Luhrsen feel ideally communication should be over emails or IRC/Slack as opposed to office hours, etc. however the issue is the number of resources are too thin.
- Robert Varga believes we need a better pipeline for communicating with users.
- CE: We could host more hackathons to attract developers to specific topics.
    - CC: Maybe we can do monthly hackathons on a specific topic and use MAC/TAC to help drive engagement
- A persistent query is to have a helping office hour for the newbies.
- IRC communication has its own set of problems - not available in all companies. LFN looking to move to some other tool like Slack.
- Hema Gopalakrishnan thinks that tutorials from the experts will be helpful.
- Daya thinks the above have been discussed but have not led to results. More useful would be how to make ODL more usable.
- We need more outreach to users that are already working on OpenDaylight, but not working in the upstream
    - Need more clear messaging to the users that if you are consuming ODL, to please contribute upstream X hours per day or week to help resolve the technical debt.
- Allan Clarke suggests the following:
    - A dedicated public face for helping for new devs get oriented
    - A central location for "low hanging fruit" tasks organized by project, including estimated time and estimated required knowledge.
- Jeff Hartley says an important work item which can be new person friendly is removing stale documentation.
- Tejas Nevrekar says we need to reach out to the end users and document the current usage of opendaylight, whats the deployment, whats the problem being solved, which plugins being used in the solution.
- Karthikeyan Krishnan Thinks that we need to improve the metrics value w.r.t scale and other deployment testing aspect in each ODL release. Also w.r.t stability of the platform is more important to each releases.

Project Status

- ODL Project Status

ODL Community Utilization Improvement suggestion

Suresh PERUMAL

- Jamo Luhrsen
- Luis Gomez
- Robert Varga

ODL Utilization Improvement Suggestion					
Improvement Area	Description	Module	Impact	Approach	Status
Split Platform and Application Projects	There is no clear demarcation between ODL Karaf platform and Karaf Application. There should be a choice or operation provided to migrate application projects for some bug fixes and there should be some migrations allowed in platform to get bug fixes or upgrades	Platform	High	Define features like Platform, Application, 3PP(3rd Party Providers) and come up with a feature template place holders custom build solutions on top of ODL platform solution Define proper bug fixes and benefits of a new platform. This will give an overview about whether to migrate to newer platform or not. We shall take an example java from Oracle or AWS cloud releases, We have been shown more articles and examples or videocalls or recorded sessions that clearly talk about feature and benefits about the version support. We shall also enable AI based on call discussion support from ODL to enable end users to communicate, explain, share more details about the problems faced while using particular feature	
Cluster Improvements	ODL customer supports 1+2(1:2) to start with. But that is not quite stable and it is not running properly and not suitable for customer deployments.	Platform & Application	Critical	ODL shall provide cluster generic usecases - Each Telco based cluster platform should look like and how best the end users can apply their application needs and demands ODL shall ensure the stability of the platform when there is a load and there is no load. ODL shall encourage vendors to add their custom solutions if the current cluster approach is not working for them. ODL shall provide options to integrate any opensource or license commercial tools to get support of customer usecases in case ODL is not able to provide proper stability results	
Performance Test Report	ODL performance results are not tested with real-time customer usecase or customer scenarios. The performance data are mostly from PoC perspective only. Most of the customers are handling Million and Billion of Data inventory with concurrent read/write operations. There were no proper guidance or performance fine tuning support or options provided in ODL platform to get scaled and well deployed in customer environments	Application & Platform	Critical	ODL shall provide more usecases with real-time integration data or Data which can be loaned by vendors for performance testing. ODL shall provide toolsets to test and ensure the performance of the platform irrespective of Domain(L0/L1/L2) usecases. ODL shall come up with performance fine tuning options when there is a issue or problems identified. ODL shall provide guidelines about preparing a better deployment platform with so and so CPU, RAM, memory, Network bandwidth, etc ODL shall provide performance fine tuning on the distributed datastore parameter or attribute changes	
Cloud Native Support	There is no support or option that talk about how ODL shall be deployed as a Container and Cloud Native platforms like K8S or AWS	Platform	Critical	ODL shall come up with another project named Cloud Native Support It shall support containerization as the first feature It shall also support sub projects like AWS deployment and K8S based Good Cloud or Azure based cloud deployments ODL shall provide how the application can seamlessly make use of Cloud Services or components or Technologies	
Integration of Message systems like KAFKA, ZOOKEEPER	Service discovery system is used in all the places to get benefit of expose and integrate the platform services across different systems. There is no much info or discussion or usecase that talks about service discovery framework or solutions	Platform	Critical	ODL shall come up with Load Balancers using Gateway feature for the RESTCONF or REST API or customer UI build by end customer. How the end user or customer shall integrate their solution with Gateway systems. This will also provide better NBI scalability as well. ODL shall provide Service discovery across ODL cluster instances or integration of ODL platform with other non-OSGi or Microservice based Springboot or Spring containers, etc	
Integration of Registry Service to support multiple JVM applications, Discovery, etc	There is no guideline that talk about how ODL shall get integrated with other platforms like other NMS/EMS systems There is no proper guideline or thoughts which has been discussed or architected in the current solution provided by ODL	Platform	High	ODL shall expose options to integrate other VMs or Other servers via Message systems or some generic or broadly used communication interfaces. This will bring new customers to make use of ODL and benefit out of it as well	

Action items

- [ ] Suresh PERUMAL- can you please make the table above editable? Jeff Hartley & Luis Gomez wanted to write comments about either the status or intentions in these areas.