Danube priorities

Introduction

The goal of this document consist in describing community priorities reported by the community, the projects and the working groups for the next OPNFV release.

Danube will be the 4th OPNFV release. Each version introduced new elements. In the table hereafter, a qualification has been defined for all the previous versions.

Version	Main achievements	Installers	Scenarios	Features
Danube	The priorities are discussed in the following section. We will see another increase in the number of scenarios due to new features and a new installer. The CI evolution already identified in Brahmaputra became critical in order to support a sustainable growth of the community and efficient use of the resources. This version could be qualified as the "**Real CI ready**" version ❓	5 (tentative)	58 (tentative)	
Colorado	**The "Features" version** Colorado confirmed the concept of scenario. While some scenarios remained "generic", there was an increase in the introduction of feature specific scenarios (e.g. bgpvpn, sfc, moon, fdio, lxd, ovs, kvm, etc.). From a testing perspective, new test cases were added and a focus was made on time management. Brahmaputra clearly identified the risk of CI resources with bigger test suites. The notion of category was then created to distinguish between "smoke" tests (e.g. tests mandatory for the release qualification) and advanced tests (e.g.OpenStack, vnf, ...). The objective of limiting the time duration under 3 hours for all the tests was achieved. A system of test promotion was also introduced but not actually used. The test API was completed for reporting pages (Functest & yardstick) in order to give a consistent view to the release manager on the status of the scenario.	4	37	15 including: Policy management (Copper, Domino, Parser) Security (Moon, Security scan) Multisite (Multisite) SFC (ONOS-sfc, ODL-sfc) Container support (LXD scenario) Acceleration (FDIO scenarios) VNF (Automation of vIMS) ARM support
Brahmaputra	**The "Scenario" version** In order to support the increase in the number of installers and the code for the first set of feature projects (initiated in Arno but not ready for the first release), the concept of scenario was created. A scenario was defined as the OPNFV vehicle to support the diversity of integrations and new features and clarify the use of resources. Additionally, Brahmaputra showed a real growth in the OPNFV ecosystem by the introduction of features, new SDN controllers and new installers. New test projects were also created to address the question of qualification beyond functional testing. The testing community introduced a test DB API to collect the test cases and the associated results. A test API was developed to allow any test project to automatically report the results into a pre-defined format.	4	13	6 including: Fault management (Doctor) Ressource management (Promise) bgpvpn (ODL) Introduction of Onos DPDK support KVM optimization (KVM)
Arno	**The "Proof of Concept" version** Arno release was supported by only 5 projects: 2 installer projects, 1 testing project, Releng and Octopus for the the CI. Arno built the foundations of the OPNFV community. It clearly identified the types of projects (installer / testing / infrastructure)	2	1	0

Danube community priorities

The priorities of the Danube release below are based on feedback from OPNFV projects, working groups, and community members participating in the release.

The purpose is not to prioritize the individual projects but to identify at a high level what work is planned in order to ensure a sustainable growth of the community.

The document aims to identify the key items needed to continue producing an OPNFV release that brings value add to the definition of reference SDN /NFV solutions.

The initial list of priorities can be found in https://etherpad.opnfv.org/p/CommunityGoals .

Please note that this page does not aim to present all the priorities of all the OPNFV projects involved in Danube. These priorities correspond to "community priorities", it means the needed glue between several projects in order to ensure the sustainable growth of the community.

Title	Domain	Problem addressed	Description	Links
CI Evolution	Infra	- Inconsistency in how things are done for different installers/scenarios - Inconsistency on resource management towards the installers/waste of CI resources - Different level of code quality on master for different installer/scenarios due to not having patch verificaiton - Long duration/enhanced test cases not run in Colorado	CI Evolution description has been initiated in Brahmaputra. The complete description can be found in CI Evolution Phase 1 and 2 aimed to be implemented for Danube.	CI Evolution
POD description file	Infra	POD description file - define hardware resources in a POD	This effort provides a common configuration file for installer and test projects to use	Infra WG Discussions
Pharos Dashboard	Infra	plan is to get at least one lab supported by the tool	a tool for community to request hardware resources in Pharos labs	Pharos Dashboard
Scenario consolidation	Infra	Too many scenarios, and the number of scenarios can only increase	Proposal to reduce the number of scenarios in a release (Ulrich Kleber)	Slide deck Scenario Considerations
Test consolidation	Test	- Very hard to know the available test cases across the different test and feature projects - Difficult without diving in each project to know what can be used or not - Wheels reinvented in terms of tooling / framework - Only Functest and Yardstick considered for release criteria - Test results/figures not always easy to understand	Creation of a testing landing page allowing any test project to expose test cases and associated constraints Automatic creation of a catalog based on Testing Database Result consolidation	
Introduction of Stress tests	Test	Lots of tests are run including qualification tests. Frameworks (Yardstick, Rally,..) are available but no stress tests run towards the System Under Test	Introduction of first load/stress tests	https://etherpad.opnfv.org/p/DanubeStressTest

Documentation	Documentation	Documentation too segmented per project. it is very hard for a new user to know where to start. And even umbrella documents per domain (installation, infrastructure, feature, testing) is not very consistent.	An effort should be done on documentation. For each release all the projects created their own documentation (user guide, release note, installation guide,..). For a new user it is not easy to read everything and get a global overview. For instance there is no global "Installation" or "Testing" documentation. An effort should be made in Danube to provide more "generic" documentation. This would be helpful in identifying features, test cases without having to dive into each project documentation	
MANO Integration pilot	Integration Pilot	MANO stack Integration pilot in Danube requires identifying ways and means to enable MANO modules to integrate with OPNFV VIM	An effort is made in Danube to add nfvo and vnf artifacts through upstream OPEN-O and OpenBaton and Juju over OPNFV VIM and test sample vnf(vIMS) through functest This can be cleaned for consistency across different stacks in future releases	https://wiki.opnfv.org/display/mano
Test cases recommandation for the compliance	Test and dovetail	in order to show the value of the OPNFV, the test gourp can help to recommande the test cases for the OPNFV compliance	test cases in the compliance set will be important for the OPNFV and will cover the test cases in the test projects and features projects.	Dovetail Test Areas and Test Cases