

ITU-T Technical Report

(02/2025)

QSTR-USO

**Use of open-source and open hardware
projects/products in testbed federations for
IMT-2020 and beyond**



Technical Report ITU-T QSTR-USO

Use of open-source and open hardware projects/products in testbed federations for IMT-2020 and beyond

Summary

This Technical Report addresses the following aspects:

- Perspectives on the use of open-source and open hardware projects/products in building testbeds and in testbed federations for IMT-2020 and beyond, and in implementing the various APIs for testbeds federations described in the testbeds federations reference model [1];
- Mapping of the open-source and open hardware products/solutions to the corresponding functional blocks in testbeds federations reference model [1] that the open source or open hardware can enable to realize or implement – thereby enabling the building of testbeds and facilitating testbeds federations;
- Examples of open-source projects that can be considered in the study on the use of open-source and open hardware projects/products in building testbeds and in testbed federations for IMT-2020 and beyond include the projects such as ONAP [5], ETSI OSM [8], OPNFV [26], ACUMOS [9] [20], OpenDayLight [23], and other open source related initiatives such as those described in [13] and other sources, illustrating how open-source and open hardware projects/products can be used in building testbeds that can also be federated based on the APIs defined in the testbeds federations reference model [1].

Keywords

APIs for testbed automation, APIs for testbeds federation, IMT-2020, instantiations of testbeds federations, open hardware, open networking platforms (ONPs), open source, reference model, testbed domain concept, testbeds.

Note

This is an informative ITU-T publication. Mandatory provisions, such as those found in ITU-T Recommendations, are outside the scope of this publication. This publication should only be referenced bibliographically in ITU-T Recommendations.

Change Log

This document contains Version 1.0 of the ITU-T Technical Report QSTR-USO "Use of open-source and open hardware projects/products in testbed federations for IMT-2020 and beyond" approved at ITU-T SG11 meeting held in Geneva from 19 to 28 February 2025.

Editor: Martin Brand
A1 Telekom Austria
Austria

Email: martin.brand@a1.at

Editor: Dr Sébastien Ziegler
Mandat International
Switzerland

Email: sziegler@mandint.org

Editor: Cédric Crettaz
Mandat International
Switzerland

Email: ccrettaz@mandint.org

© ITU 2025

All rights reserved. No part of this publication may be reproduced, by any means whatsoever, without the prior written permission of ITU.

Table of Contents

		Page
1	Scope.....	1
2	References.....	1
3	Definitions	2
	3.1 Terms defined elsewhere	2
	3.2 Terms defined in this Technical Report	2
4	Abbreviations and acronyms	2
5	Introduction.....	3
6	Use of open-source projects/products in testbed federations for IMT-2020 and beyond.....	3
	6.1 ONF SD-CORE	3
	6.2 OMEC.....	4
	6.3 O-RAN-SC near-RT RIC	4
	6.4 ONAP	4
	6.5 ETSI OSM	4
	6.6 Magma.....	4
	6.7 Juju	5
	6.8 Kubernetes.....	5
	6.9 OpenStack.....	5
	6.10 MAAS.....	5
	6.11 OpenConfig	5
	6.12 ACUMOS	5
	6.13 Other open-source projects and products	6
7	Use of open-hardware projects/products in testbed federations for IMT-2020 and beyond.....	6
	7.1 Other open hardware related projects/products	6
8	Mapping of the open source and open hardware products/solutions to reference model	6

Technical Report ITU-T QSTR-USO

Use of open-source and open hardware projects/products in testbed federations for IMT-2020 and beyond

1 Scope

This Technical Report serves as a guide on the use of open-source and open hardware projects/products in testbed federations for IMT-2020/5G and beyond that conform to the reference model for testbeds federations [1] and its APIs.

In applying this Technical Report to the consideration of open-source and open-hardware projects/products in building testbeds and implementing their federation, it is also necessary to consider existing testbeds that may be readily adopted and enhanced with open-source and open-hardware-based components, while at the same time transforming the existing testbeds to conform to the testbeds federations reference model [1]. Use cases [2] for testbeds federations also need to be taken into account when applying the Technical Report. It is important not only to adopt existing testbeds where possible, but also to consider applicable test scenarios, test specifications or test plans that may also be available from certain communities such as standards development organizations (SDOs) or Fora such as [3] [4] [14] [15] [27]. Other perspectives to consider include insights shared by [16].

2 References

- [1] Recommendation ITU-T Q.4068 (2021), *Open application program interfaces (APIs) for interoperable testbed federations*.
- [2] Technical Report ITU-T QSTR-UCFTBS, *Use cases for federated testbeds and business scenarios*.
- [3] ETSI TR 103 761: *Core Network and Interoperability Testing (INT) Methodologies for E2E Testing & Validation of Vertical Applications over 5G & Beyond networks*.
- [4] 5G: Framework to Conduct 5G Testing, November 2020: Federal Mobility Group of USA.
- [5] Open Network Automation Platform (ONAP), <https://www.onap.org>.
- [6] Open Networking Foundation (ONF), <https://www.opennetworking.org/>.
- [7] OpenConfig, <https://www.openconfig.net/>.
- [8] Open-Source MANO, <https://osm.etsi.org/>.
- [9] Acumos AI, <https://www.acumos.org/>.
- [10] ONF SD-Core, <https://opennetworking.org/sd-core>.
- [11] Kubernetes, <https://kubernetes.io>.
- [12] SD-Core deployment guide: <https://docs.sd-core.opennetworking.org/main/deployment/deployment5G.html>
- [13] Ubuntu White Paper: *Telecom infrastructure the open-source way*. October 2021.
- [14] O-RAN SC Near-RT RIC subproject: <https://wiki.o-ran-sc.org/display/RICP/Introduction+and+guides>.
- [15] O-RAN SC Near-RT RIC F release: <https://wiki.o-ran-sc.org/display/REL/F+Release>.

- [16] S. A. Wright and D. Druta, *Open source and standards: The role of open source in the dialogue between research and standardization, 2014 IEEE Globecom Workshops (GC Wkshps)*, 2014, pp. 650-655, doi: 10.1109/GLOCOMW.2014.7063506.
- [17] CANONICAL White Paper by Arno Van Huyusteen: *Telecom infrastructure the open-source way*, October 2021. For more information about Canonical's solutions for telecommunications visit <https://ubuntu.com/telco>.
- [18] OpenConfig Project: <https://www.openconfig.net/>.
- [19] ONF SD-Core: <https://opennetworking.org/sd-core/>.
- [20] ACUMOS: <https://www.acumos.org/>.
- [21] Open Compute Project (OCP): <https://www.opencompute.org/>.
- [22] White Box Switches: <https://www.techtarget.com/searchnetworking/definition/white-box-switch>.
- [23] Open Daylight SDN Controller: <https://www.opendaylight.org/>.
- [24] ONF SD-RAN (Software Defined RAN): <https://opennetworking.org/open-ran/>.
- [25] ONF ONOS: <https://opennetworking.org/onos/>.
- [26] Anuket Project (formerly OPNFV: <https://www.opnfv.org/>): <https://wiki.anuket.io/>.
- [27] Definition of the Testing Framework for the NGMN 5G TTI Interoperability: BY NGMN ALLIANCE: DATE: AUG 2018: VERSION 1.0.

3 Definitions

3.1 Terms defined elsewhere

None.

3.2 Terms defined in this Technical Report

None.

4 Abbreviations and acronyms

This Technical Report uses the following abbreviations and acronyms:

API	Application Programming Interface
CNF	Cloud Native Function
ETSI	European Telecommunications Standards Institute
IoT	Internet of Things
MAAS	Metal-As-A-Service
MEC	Multi-Access Edge Computing
ONP	Open Networking Platform
OSM	Open-Source MANO
O-RAN	Open Radio Access Network
SDO	Standards Development Organization

5 Introduction

There are various types of testbeds relevant to 5G and beyond that are required by the industry and research communities for running various kinds of test scenarios and test cases. Recommendation ITU-T Q.4068, which defines the reference model for testbeds federation, provides relevant insights on the benefits of testbeds federations to various stakeholders. Examples of testbeds for IMT-2020 and beyond (some of which are briefly described in [1]) can be characterized as follows:

- RAN testbed for traditional RAN architectures for various RATs ranging from 2G, 3G, 4G, 5G,
- Open RAN testbed based on open RAN architectures such as open radio access network (O-RAN) architecture from the O-RAN alliance
- Internet of things (IoT) testbed
- Multi-access edge computing (MEC) testbed
- Fixed/wired access testbed
- Multi-layer transport network testbed
- 5G core network testbed
- Other types of testbeds relevant to 5G and beyond

There are also various other important sources that are relevant when obtaining an outlook on testbeds relevant to 5G and beyond, such as [3] [4] [5].

The following aspects should be considered as part of this Technical Report:

1. Making use of open-source and open hardware projects/products in building testbeds and in testbed federations for IMT-2020 and beyond, and in implementing the various application programming interfaces (APIs) for testbeds federations prescribed by the testbeds federations reference model [1];
2. Provision of a mapping of the open-source and open hardware products/solutions to the corresponding functional blocks in the testbeds federations reference model [1] that the open-source or open-hardware products/solutions can enable to realize or implement – thereby enabling the building of testbeds and facilitating testbeds federations based on the principles and APIs for implementation for that purpose by [1].

6 Use of open-source projects/products in testbed federations for IMT-2020 and beyond

This clause presents some examples of open-source products/solutions that may be considered in building (implementing) testbeds for 5G and beyond. In the same process of using such products to implement a testbed or testbeds that serve a particular use case or use cases, it is necessary to consider the mapping to the particular functional block(s) in the testbeds federations reference model [1] that the open software can enable to realize or implement, as well as the API(s) in the testbeds federations reference model that may be implemented by some APIs provided by the open software product.

NOTE 1 – Regarding the APIs prescribed in [1] that are of relevant to the functional block realized by the open-software product in some way (possibly in combination with other open-hardware and/or open-source products) other products and methods should be considered in implementing the relevant APIs.

NOTE 2 – Implementers should also consider the aspects presented in clause 8 on the mapping of open-source and open-hardware products/solutions to the corresponding functional blocks in the testbeds federations reference model [1] as enablers for implementation.

6.1 ONF SD-CORE

As described by [17], the SD-Core project is a 4G/5G disaggregated mobile core optimized for public cloud deployment in concert with distributed edge clouds and is ideally suited for carrier and private enterprise 5G networks. It exposes standard 3GPP interfaces enabling use of SD-Core [10] [12] [19]

as a conventional mobile core. It is also available pre-integrated with an adapter (which is part of the Aether ROC subsystem) for those deploying it as a mobile core as-a-service solution. SD-Core is an integral component of Aether, ONF's 5G connected edge platform for private mobile connectivity and edge cloud services. It can be rapidly deployed pre-integrated with Aether, as a standalone 5G/4G mobile core, or as control and data plane (UPF) components integrated into custom designed solutions. This range of versatility makes the open-source SD-Core platform ideally suited for the broadest range of use cases and deployment scenarios. Further details can be found in [17].

6.2 OMEC

As described by [17], OMEC [17] is a full-featured, scalable, high-performance and open-source EPC. OMEC has been optimized to handle the ensuing onslaught of devices coming online as part of the move to 5G and IoT. It is designed to be used as a standalone EPC and is also an upstream project for the COMAC platform that (among other things) is integrating mobile and fixed subscriber management functions. It provides 3GPP release 13 compatibility and complete connectivity, billing and charging capabilities. Further details can be found in [17].

6.3 O-RAN-SC near-RT RIC

As described by [14], the O-RAN SC near-RT RIC subproject [14] – created by O-RAN and the Linux Foundation – is a joint effort by contributors from various organizations. It provides two releases per year. The latest release, as of this writing, is the F release which can be downloaded from [15].

6.4 ONAP

As described by [5], ONAP [5] is an open-source solution for network management and orchestration. It addresses global and massive scale (multi-site and multi-VIM) orchestration capabilities for both physical and virtual network elements. It facilitates service agility by providing a common set of Northbound REST APIs that are open and interoperable, and by supporting YANG and TOSCA data models. ONAP's modular and layered nature improves interoperability and simplifies integration, allowing it to support multiple VNF environments by integrating with multiple VIMs, VNFMs, SDN Controllers, and even legacy equipment. This approach allows network and cloud operators to optimize their physical and virtual infrastructure for cost and performance; at the same time, ONAP's use of standard models reduces integration and deployment costs of heterogeneous equipment, while minimizing management fragmentation.

NOTE – ONAP can be used to implement an SMO, and there are already several such ONAP based SMO products in the industry.

6.5 ETSI OSM

As described by [8], the European Telecommunications Standards Institute (ETSI) OSM [8] automates deployment and operations of network functions, and reduces OPEX, by using an open-source implementation of ETSI NFV MANO. Further details can be found in [8] [17].

NOTE – ETSI OSM can be used to implement an SMO, and there are already several such ETSI OSM based SMO products in the industry.

6.6 Magma

As described by [17], Magma [17] is an open-source software platform that gives network operators an open, flexible, and extendable mobile core network solution. Magma is designed to be 3GPP generation and access network (cellular or Wi-Fi) agnostic. Running on Ubuntu, deployed with Juju Charms, and onboarded with OSM is the typical configuration. Further details can be found in [17].

6.7 Juju

As described by [16], Juju [16] is a charmed operator framework, composed of a charmed operator lifecycle manager and the charmed operator SDK. It enables users to deploy, integrate, and manage Kubernetes [11], container, and VM-native applications seamlessly across hybrid clouds. Juju drives day 0 through day 2 operations in a complex environment. More details in [16]. Juju is a powerful tool, and in the context of core network infrastructure, it is used to provide a high level of automation at scale. It is successfully used by many tier-1 network operators and adopted by ETSI as part of the ETSI open-source MANO (OSM) [8]. Further details can be found in [17].

6.8 Kubernetes

As described by [17], Kubernetes (K8s) [17] [11], is an open-source platform, pioneered by Google, which started as a simple container orchestration tool but has grown into a cloud native platform. It is one of the most significant advancements in IT since the public cloud came into existence in 2009 and has an unparalleled 5-year 30% year-on-year growth rate in both market revenue and overall adoption. Charmed Kubernetes gives users perfect portability of workloads across all infrastructures, from the core network to the public cloud. With a strong focus on the cloud native function (CNF) and AI/ML, Ubuntu is the platform of choice for K8s in telco. Further details can be found in [17].

6.9 OpenStack

As described by [17], OpenStack [17] is a carrier-grade virtual infrastructure manager (VIM). In turn, charmed OpenStack is an OpenStack distribution engineered for the best price-performance. Charmed OpenStack is deployable, maintainable, and upgradable economically. This is achieved by putting full automation around its deployments and exposed deployment operations. Further details can be found in [17].

6.10 MAAS

As described by [17], MAAS or Metal-As-A-Service [17], is a service that lets users treat physical servers like virtual machines – instances – in the cloud. No need to manage servers individually – MAAS turns bare metal into an elastic, cloud-like resource. Further details can be found in [17].

6.11 OpenConfig

As described by [17], OpenConfig [17] [7] [18] is an informal working group of network operators sharing the goal of moving networks toward a more dynamic, programmable infrastructure by adopting software-defined networking principles such as declarative configuration and model-driven management and operations. The initial focus of OpenConfig is on compiling a consistent set of vendor-neutral data models (written in YANG) based on actual operational needs from use cases and requirements from multiple network operators. Streaming telemetry is a new paradigm for network monitoring in which data is streamed from devices continuously with efficient, incremental updates. Operators can subscribe to the specific data items they need, using OpenConfig data models as the common interface. Further details are in [17] [7] [18].

6.12 ACUMOS

As described by [17], Acumos AI [17] [9] [20] is a platform and open-source framework that makes it easy to build, share, and deploy AI apps. Acumos standardizes the infrastructure stack and components required to run an out-of-the-box general AI environment. This frees data scientists and model trainers to focus on their core competencies and accelerates innovation. Acumos is part of the LF AI Foundation, an umbrella organization within The Linux Foundation that supports and sustains open-source innovation in artificial intelligence, machine learning, and deep learning while striving

to make these critical new technologies available to developers and data scientists everywhere. Further details can be found in [17] [9] [20].

6.13 Other open-source projects and products

Other open-source projects/products worthy to mention, include the following:

- OpenDayLight SDN controller: Open-source SDN controller [23]
- ONOS SDN controller: Open-source SDN controller [25]
- ONF SD-RAN: Software defined RAN [24]
- Anuket project (formerly OPNFV): Open NFV software [26]

7 Use of open-hardware projects/products in testbed federations for IMT-2020 and beyond

This clause presents some examples of open hardware products/solutions that may be considered in building (implementing) testbeds for IMT-2020 and beyond. In the same process of using such products to implement a testbed or testbeds that serve a particular use case or use cases, the mapping to a particular functional block(s) in the testbeds federations reference model [1] that the open hardware can help realize or implement, as well as the API(s) in the testbeds federations reference model that may be realized by some APIs that the open hardware product provides, should be considered.

NOTE 1 – Regarding the APIs prescribed in [1] that are of relevance to the functional block realized by the open hardware product in some way (possibly in combination in some other open hardware and/or open-source products) there may be the need to consider other products and methods that should be considered in implementing the relevant APIs.

NOTE 2 – Implementers should also consider the aspects presented in clause 8 on mapping of the open source and open hardware products/solutions to corresponding functional blocks in testbeds federations reference model [1] as enablers for implementation.

As described by [21], the open compute project (OCP) is creating a set of technologies that are disaggregated and fully open, allowing for rapid innovation in the network space, including white box hardware switches. The project covers not only open hardware but also open-source as summarized below (more details in [21]):

- Fully disaggregated and open networking hardware (HW) and software (SW)
- Operating system (OS) - Linux based operating systems and developer tools, and REST APIs
- Fully automated configuration management & bare metal provisioning
- Universal and multi-form factor switch motherboard hardware
- Fully open integration and connectivity
- Energy efficient power and cooling designs
- Software defined networking (SDN)

7.1 Other open hardware related projects/products

There are other open hardware projects/products that leverage the concept of white box networking such as the ones mentioned in [22].

8 Mapping of the open source and open hardware products/solutions to reference model

This clause provides a mapping of the open-source and open hardware products/solutions to corresponding functional blocks in the testbeds federations reference model [1] that the open-source or open hardware can enable to realize or implement.

NOTE – Tables 1 to 3 are meant to only indicate where the open-source and open-hardware products/solutions may play a role in full or in combination with other solutions in complementing each other in implementing specific targeted functional blocks of the testbeds federations reference model [1].

Table 1 – Mapping of the open-source products/solutions to corresponding functional blocks in the testbeds federations reference model [1] that can be implemented

Open-Source product/solution	Functional block(s) of testbeds federations reference model [1]
<i>ONF SD-CORE; OMEC</i>	<i>Level-0 resource(s)</i> in the testbeds federations reference model [1] that could play a role of a test component(s) <i>as a test component in a test architecture</i> upon which the test scenario is to be executed, or role of a component(s) under test (CUT); Some of the components of the open source solution can be treated individually as test component or component under test (CUT)
<i>O-RAN-SC near-RT RIC</i>	<i>Level-1 resource</i> in the testbeds federations reference model [1] that could be used within a test scenario as a test component in a test architecture upon which the test scenario is to be executed, or could serve in the role of a component(s) under test (CUT)
<i>ONAP</i>	<i>Level-1 resource</i> in the testbeds federations reference model [1] that could be used within a test scenario as a test component in a test architecture upon which the test scenario is to be executed, or could serve in the role of a component(s) under test (CUT)
<i>ETSI OSM</i>	<i>Level-1 resource</i> in the testbeds federations reference model [1] that could be used within a test scenario as a test component in a test architecture upon which the test scenario is to be executed, or could serve in the role of a component(s) under test (CUT)
<i>Magma</i>	<i>Level-0 Resource(s)</i> in the testbeds federations reference model [1] that could play the role of a test component(s) <i>as a test component in a test architecture</i> upon which the test scenario is to be executed, or could serve in the role of a component(s) under test (CUT); some of the components of the open-source solution can be treated individually as test component or component under test (CUT)
<i>Juju</i>	<i>Level-1 resource</i> in the testbeds federations reference model [1] that could be used within a test scenario as a test component in a test architecture upon which the test scenario is to be executed, or could serve in the role of a component(s) under test (CUT)
<i>Kubernetes</i>	<ul style="list-style-type: none"> a) <i>Level-1 resource</i> in the testbeds federations reference model [1] that could be used within a test scenario as a test component in a test architecture upon which the test scenario is to be executed, or could serve in the role of a component(s) under test (CUT). b) There may be also resources within the open-source solution that realize <i>level-0 resource(s)</i> in the testbeds federations reference model [1] that could play a role of a test component(s) as a test component in a test architecture upon which the test scenario is to be executed, or could serve in the role of a component(s) under test (CUT).
<i>OpenStack</i>	<ul style="list-style-type: none"> a) <i>Level-1 resource</i> in the testbeds federations reference model [1] that could be used within a test scenario as a test component in a test architecture upon which the test scenario is to be executed, or could serve in the role of a component(s) under test (CUT). b) There are also resources within the open-source solution that realize <i>Level-0 resource(s)</i> in the testbeds federations reference model [1] that could play the role of a test component(s) <i>as a test component in a test architecture</i> upon

Table 1 – Mapping of the open-source products/solutions to corresponding functional blocks in the testbeds federations reference model [1] that can be implemented

Open-Source product/solution	Functional block(s) of testbeds federations reference model [1]
	which the test scenario is to be executed, or could serve in the role of a component(s) under test (CUT).
<i>MAAS</i>	<i>Level-1 resource</i> in the testbeds federations reference model [1] that could be used within a test scenario as a test component in a test architecture upon which the test scenario is to be executed, or could serve in the role of a component(s) under test (CUT)
<i>OpenConfig</i>	<i>Level-1 resource</i> in the testbeds federations reference model [1] that could be used within a test scenario as a test component in a test architecture upon which the test scenario is to be executed, or could serve in the role of a component(s) under test (CUT)
<i>ACUMOS</i>	<i>Level-1 resource</i> in the testbeds federations reference model [1] that could be used within a test scenario as a test component in a test architecture upon which the test scenario is to be executed, or could serve in the role of a component(s) under test (CUT)
<i>OpenDayLight SDN controller</i>	<i>Level-1 resource</i> in the testbeds federations reference model [1] that could be used within a test scenario as a test component in a test architecture upon which the test scenario is to be executed, or could serve in the role of a component(s) under test (CUT)
<i>ONOS SDN controller</i>	<i>Level-1 resource</i> in the testbeds federations reference model [1] that could be used within a test scenario as a test component in a test architecture upon which the test scenario is to be executed, or could serve in the role of a component(s) under test (CUT)
<i>ONF SD-RAN</i>	<p>a) <i>Level-1 resource</i> in the testbeds federations reference model [1] that could be used within a test scenario as a test component in a test architecture upon which the test scenario is to be executed, or could serve in the role of a component(s) under test (CUT).</p> <p>b) There are also resources within the open-source solution that realize <i>level-0 resource(s)</i> in the testbeds federations reference model [1] that could play the role of test component(s) <i>as a test component in a test architecture</i> upon which the test scenario is to be executed, or could serve in the role of a component(s) under test (CUT).</p>
<i>Anuket project</i>	<p>a) <i>Level-1 resource</i> in the testbeds federations reference model [1] that could be used within a test scenario as a test component in a test architecture upon which the test scenario is to be executed, or could serve in the role of a component(s) under test (CUT).</p> <p>b) There are also resources within the open-source solution that realize <i>level-0 resource(s)</i> in the testbeds federations reference model [1] that could play the role of a test component(s) <i>as a test component in a test architecture</i> upon which the test scenario is to be executed, or could serve in the role of a component(s) under test (CUT).</p>

Table 2 – Mapping of the open hardware products/solutions to corresponding functional blocks in the testbeds federations reference model [1] that can be implemented

Open Hardware product/solution	Functional block(s) in the testbeds federations reference model [1]
<i>Open compute project (OCP)</i>	<p>a) <i>Level-1 resource</i> in the testbeds federations reference model [1] that could be used within a test scenario as a test component in a test architecture upon which the test scenario is to be executed, or could serve in the role of a component(s) under test (CUT).</p> <p>b) There are also resources within the open-source solution that realize <i>level-0 resource(s)</i> in the testbeds federations reference model [1] that could play the role of a test component(s) <i>as a test component in a test architecture</i> upon which the test scenario is to be executed, or could serve in the role of a component(s) under test (CUT).</p>
<i>White Box Networking</i>	<p><i>Level-0 Resource(s)</i> in the testbeds federations reference model [1] that could play the role of a test component(s) <i>as a test component in a test architecture</i> upon which the test scenario is to be executed, or could serve in the role of a component(s) under test (CUT); some of the components of the open-source solution can be treated individually as a test component or component under test (CUT)</p> <p>NOTE – Within white box networking product or solution there may be <i>level-1 resource</i> in the testbeds federations reference model [1] that could be used within a test scenario as a test component in a test architecture upon which the test scenario is to be executed, or could serve in the role of a component(s) under test (CUT).</p>

Table 3 – Gaps on functional block(s) in the testbeds federations reference model [1] for which there appear to be no open-source projects/products available yet that could play a role in implementing the functional blocks

Open-source product/solution	Functional block(s) in the testbeds federations reference model [1]
There appears to be no open-source product that can directly and fully implement the <i>testbed resource broker</i> . However, there may be certain open-source code that could be used in implementing some code elements of a <i>testbed resource broker</i> . This indicates that further study on this subject is required.	<i>Testbed resource broker</i>
There appears to be no open-source product that can directly and fully implement an <i>inter-testbed E2E universal resource broker for testbeds federation</i> . However, there may be certain open-source code that could be used in implementing some code elements of an <i>Inter-testbed E2E universal resource broker for testbeds federation</i> . This indicates that further study on this subject is required.	<i>Inter-testbed E2E universal resource broker for testbeds federation</i>
There appears to be no open-source product that can directly and fully implement <i>testbed management system</i> . However, there may be certain open-source code that could be used in implementing some code elements of a <i>testbed management system</i> . This indicates that further study on this subject is required.	<i>Testbed management system</i>

Table 3 – Gaps on functional block(s) in the testbeds federations reference model [1] for which there appear to be no open-source projects/products available yet that could play a role in implementing the functional blocks

Open-source product/solution	Functional block(s) in the testbeds federations reference model [1]
There appears to be no open-source product that can directly and fully implement <i>test manager</i> . However, there may be certain open-source code that could be used in implementing some code elements of a <i>test manager</i>	<i>Test manager</i>
There appears to be no open-source product that can directly and fully implement <i>Real-time resources state repository</i> . However, there may be certain open-source code that could be used in implementing some code elements of a <i>real-time resources state repository</i> . This indicates that further study on this subject is required.	<i>Real-time resources state repository</i>
