

# Glossary

## See also:

- [SDN-R Glossary](#)

## Definitions

**AAF:** Application Authorization Framework

- fine-grained authorization library and service; one of the ONAP [Common Services](#)

**AAI = A&AI:** Active and Available Inventory (component of ONAP runtime)

- Real-time views of Resources, Services, Products, Customer Subscriptions, and their relationships

**AID:** Architecture Integration Document

**Akka:** handles clustering (used OpenDaylight controllers) [akka.io](#)

**AJSC:** see [JSC](#)

**Amsterdam:** Code name for the [first release](#) of ONAP

**APPC (formerly APP-C):** Application Controller (part of ONAP)

- handles the life cycle management of Virtual Network Functions (VNFs)

**Beijing:** Code name for the [second release](#) of ONAP

**BPEL:** Business Process Execution Language (OASIS Standard)

- XML-based language that allows Web services in a service-oriented architecture (SOA) to interconnect and share data.

**BPMN:** Business Process Model and Notation ([Wikipedia](#)) or Business Process Management Notation (ONAP.pdf)

- graphical representation for specifying business processes

**BRMS:** Business Rules Management System

**BSS:** Business Support System

**CCSDK:** Common Controller SDK project

- Code shared across controllers (e.g. SDNC, APP-C)

**CDAP:** Cask Data Application Platform

- open source framework to build and deploy data applications on Apache™ Hadoop® ([CDAP.io site](#))

**CDS:** Controller Design Studio

- it is not the tool for the design of controllers instead it is common design tool to support both SDNC and generic L4-7 NF controllers integrated with SDC (e.g. APPC, VFC)
- part of [CCSDK project](#),
- see [vFW CDS Casablanca](#) - using CDS in context of virtual Firewall use case

**CDT:** APPC Controller Design Tool

**Chef:**

- server configuration management tool written in Ruby and Erlang. ([Wikipedia](#)) ([Chef site](#))

**CI/CD:** Continuous Integration / Continuous Delivery

- **continuous integration (CI)** is the practice of merging all developer working copies to a shared mainline several times a day ([Wikipedia](#))
- **continuous delivery (CD)** is a software engineering approach in which teams produce software in short cycles, ensuring that the software can be reliably released at any time ([Wikipedia](#))

**CIA:** Container Images - see [Project Docker/OCI Images - Best Practices and Tools](#)

- the abbreviation probably uses the letter A at the end to distinct from commonly used CI - Continuous Integration

**CL:** Control Loop

**CLAMP:** Closed Loop Automation Management Platform (project)

**CLI:** [Command Line Interface](#) (project)

**CMA:** Change Management Application (within ONAP)

**CNF:** Cloud Native network Function. A network function implemented using cloud native principles such as micro services running in containers, immutable infrastructure, etc.

**COE:** Container Orchestration Engine

**Congress:**

- Policy as a service (<https://wiki.openstack.org/wiki/Congress>)

**Controller:**

- manages the state of an Application, Infrastructure, or Network resource
- single service/network domain scope

**CPE:** Customer Premise Equipment

**CSAR:** Cloud Service ARchive ([link](#))

- a package defined by OASIS TOSCA. It is a ZIP file that includes a TOSCA template of a Network Service, and all the scripts or files that a VNF needs for the lifecycle from creation to termination.

**CSMF:** Communication Service Management Function

**DAO:** Data Access Object ([Wikipedia](#))

- an object that provides an abstract interface to some type of database or other persistence mechanism

**DCAE:** Data Collection, Analytics and Events (component of ONAP runtime)

**DDoS:** Distributed Denial-of-Service attack ([Wikipedia](#))

**DG:** Directed Graph

- in ONAP, it is the XML output of DG Builder. Directed graphs are used to encapsulate service logic.

**DG Builder:** Directed Graph Builder

- This is an ONAP-customized version of [Node-RED](#). It is a graphic editor used to generate the resulting XML files that are fed into SLI.

**Disconnect:** (Vendor specific)

- disconnecting an existing connection sometimes called a delete
- Total Disconnect - Disconnect all ports and VLAN from the Database (SDNC) and Network (NCS)
- VLAN Disconnect - Disconnect an active vlan from Database (SDNC) and Network (NCS)
- Soft Disconnect - Disable the interface for traffic to flow through without deleting the interface from the network

**DLUX:**

- the OpenDaylight community GUI ([DLUX documentation](#))
- typically not needed for ONAP development

**DMaaP:** Data Movement as a Platform

- a set of common services provided by ONAP, including a Message Router, Data Router, and a Data Bus Controller

**DME:** Direct Messaging Engine (common service within ONAP)

**DNS:** Domain Name System

**Docker:**

- containerization platform ([link](#))

**DPDK:** Data Plane Development Kit

- a set of libraries and drivers for fast packet processing ([dpdk.org](http://dpdk.org)). Its optimizations could be used by VNFs requiring high packet processing speeds.

**Drools:** Red Hat's Business Rules Management System solution <https://www.drools.org/>

**EELF:** Event and Error-Logging Framework (common service within ONAP)

**EMS:** Element Management System ([Wikipedia](#))

- systems and applications for managing **network elements** (NE) on the network element-management layer (NEL) of the **Telecommunications Management Network** (TMN) model.

**ESR:** External System Register

- part of AAI - see [External System Register \(5/14/17\)](#)

**ETSI:** European Telecommunications Standards Institute

- a standards body for information and communications technologies (<http://www.etsi.org>). The **ETSI Network Functions Virtualization (NFV) Industry Specification Group (ISG)** promotes standards for Network Function Virtualization.

**EUAG:** ONAP End User Advisory Group, i.e. telecom operators (<https://wiki.lfnetworking.org/pages/viewpage.action?pageId=2916362>)

**FCAPS:** Fault Configuration Accounting Performance Security

**GBP:** Group-Based Policy (<https://wiki.openstack.org/wiki/GroupBasedPolicy>)

**GNFC:** Generic Network Function Controller - A proposed unification of the APP-C and SDN-C for complete L0-L7 control. ([ONAP\\_GNF\\_ControllersSOL003.pptx](#))

**HAS:** [Homing and Allocation Service](#) - part of OOF

**Heat:**

- cloud infrastructure creation template language for Open Stack (<https://wiki.openstack.org/wiki/Heat>)

**HDFS:** Hadoop Distributed File System

**Helm:** application package manager for kubernetes see <https://docs.helm.sh/>

**Holmes:** Holmes project provides alarm correlation and analysis for Telecom cloud infrastructure and services.

**Honeycomb:**

- Java-based agent that runs on the same host as a Vector Packet Processor (VPP); manages the VPP by translating NETCONF/YANG or RESTCONF ([link](#))
- used by vFirewall demonstration service within ONAP

**HPA:** Hardware Platform Awareness

**HTTP:** HyperText Transfer Protocol

**HV VES:** High Volume Virtual function Event Stream

- part of DCAE - see [High Volume VES Collector](#)

**IAM/IDAM:** Identity and Access Management

- security and business discipline that "enables the right individuals to access the right resources at the right times and for the right reasons" ([Wikipedia](#))

**ICE:** Incubation and Certification Environment

- for vendors and 3<sup>rd</sup> parties to develop Virtual Network Functions and other Resources using ONAP and a network cloud

**IDS:** Intrusion Detection System ([Wikipedia](#))

**IETF:** Internet Engineering Task Force (<http://www.ietf.org>)

- a standards body that creates the internet protocol standards.

**IKE:** Internet Key Exchange ([Wikipedia](#))

**IPS:** Intrusion Prevention System ([Wikipedia](#))

**IPSEC:** Internet Protocol Security ([Wikipedia](#))

**JAR:** Java ARchive ([Wikipedia](#))

- a package file format typically used to aggregate many Java class files and associated metadata and resources (text, images, etc.) into one file for distribution. A JAR file is built on the ZIP format and typically has a .jar file extension.

**JSC:** [Java Service Container](#): (formerly AJSC)

- Service container library and framework that is provided as part of Common Frameworks open source, separately from ONAP

**JSON:** JavaScript Object Notation

**k8s:** a popular way to abbreviate kubernetes

**Kafka:** a distributed streaming platform created by Apache read <https://kafka.apache.org/intro> for greater depth

**Karaf:**

- a container, sponsored by Apache, fully supporting OSGI with lots of extras ([Apache page](#))

**Kubernetes:** Quoting <https://en.wikipedia.org/wiki/Kubernetes> Kubernetes is, "an open-source container-orchestration system for automating deployment, scaling and management of containerized applications"

**LCM:** Life Cycle Management

**LFN CVC:** Linux Foundation Networking Compliance/Verification Committee

- part of VNFSDK project - see [LFN CVC Testing in VNFSDK](#)

**LFN CVP:** Linux Foundation Networking Compliance/Verification Program

- part of VNFSDK project - see [LFN CVC Testing in VNFSDK](#)
- it uses Dovetail - test framework provided by OPNFV

**LRM:** Local Resource Monitor

**M0:** Release Kick-off milestone. See also [Release Lifecycle](#)

**M1:** Release Planning milestone. See also [Release Lifecycle](#)

**M2:** Release Functionality Freeze milestone. See also [Release Lifecycle](#)

**M3:** Release API Freeze milestone. See also [Release Lifecycle](#)

**M4:** Release Code Freeze milestone. See also [Release Lifecycle](#)

**MACD:** (Vendor specific) Move Add Change Delete/Disconnect

- an existing connection is being changed; thus, MACDs are also called change orders. The connection will have both operational and configuration data.

**MANO:** MANagement and Organization of NFV

- the [ETSI-defined framework](#) for the management and orchestration of all resources in the cloud data center. See the [SDX Central description](#) and the [ETSI MANO page](#)
- ETSI's MANO does not include Controller and Policy components, as ONAP does
- ETSI's MANO resource description does not include complete meta-data for lifecycle management of infrastructure as well as VNFs (ONAP does)

**MD-SAL:** Model Driven Service Abstraction Layer

- OpenDayLight derives service abstractions from YANG models using yang tools. [read about MD-SAL on github](#)

**MR:** Message Router (a [Common Service](#) of ONAP)

**MOP:** Method of Procedure

- set of deployment instructions

**MOTS:** Mechanized Operations Tracking System

**MSB:** Microservice Bus

**MSO:** Master Service Orchestrator (component of ONAP runtime) renamed to SO (Service Orchestrator)

- automates activities, tasks, rules and policies needed for on-demand creation, modification or removal of network, application or infrastructure services

**MUSIC:** [Multi-site State Coordination Service](#) (Project)

**MVP:** Minimum Viable Product

- The minimum set of features/projects determined by the ONAP TSC as required for a specific release. [https://en.wikipedia.org/wiki/Minimum\\_viable\\_product](https://en.wikipedia.org/wiki/Minimum_viable_product)

**NAI:** Network Artificial Intelligence

**NANCSP:** Network Cloud Service Provider

**NBI:** North Bound Interface

**NEP:** Network Equipment Provider

**NETCONF:** Network Configuration Protocol ([Wikipedia](#))

**network cloud:**

- a compute, storage, and network virtualization environment where the network integration and control scope extends beyond the data center

**NFV:** Network Function Virtualization ([Wikipedia](#))

- a network architecture concept that uses the technologies of IT virtualization to virtualize entire classes of network node functions into building blocks that may connect, or chain together, to create communication services
- decouples the network functions, such as DNS, Caching, etc., from proprietary hardware appliances, so they can run in software to accelerate service innovation and provisioning, particularly within service provider environments.
- <https://www.sdxcentral.com/nfv/definitions/which-is-better-sdn-or-nfv/>

**NFVI:** network functions virtualization infrastructure

- <https://www.sdxcentral.com/nfv/definitions/nfv-mano/>

**NFVO:** Network Function Virtualization Orchestrator

- [ONAP SO ETSI-Aligned Hierarchical Orchestration](#)

**NOD:** Network On Demand**Node-RED:**

- open source project by IBM to graphically create flows that configure devices. Projects include managing a raspberry pi. ([official docs](#))

**NS:** Network Services**NS:** Network Slice**NS:** (Vendor Specific) New Start

- a new connection is being setup for the first time. There will only be configuration data until it is activated.

**NSMF:** Network Slice Management Function**NSSMF:** Network Slice Subnet Management Function**OA&M:** Operations, Administration and Management**OASIS:**

- Nonprofit consortium that drives the development, convergence and adoption of open standards for the global information society.

**Offer:**

- in ONAP, a bundling of Products with specific Marketing configurations

**OCX/OMX** - AT&T system that sends service orders to SO API Handler to trigger its activities**OMF:** Operational Management Framework (of ONAP)**OMSA:** ONAP Microservice Architecture**OOF:** ONAP Optimization Framework**OpenDaylight = ODL**

- Largest open source SDN controller, or network control plane, written in Java (see <https://www.opendaylight.org/>, also [https://en.Wikipedia.org/wiki/OpenDaylight\\_Project](https://en.Wikipedia.org/wiki/OpenDaylight_Project))

**ONAP:** Open Network Automation Platform, including Open-source Enhanced Control, Orchestration, Management, and Policy**OpenStack:**

- a free and open-source software platform for creating private and public clouds (compute, network, and storage facilities) (<https://www.openstack.org/>)
- mostly deployed as an infrastructure-as-a-service (IaaS). The software platform consists of interrelated components that control hardware pools of processing, storage, and networking resources throughout a data center. Users either manage it through a web-based dashboard, through command-line tools, or through a RESTful API.

**OOM:** ONAP Operations Manager

**OpenAPI Specification:** "defines a standard, language-agnostic interface to RESTful APIs which allows both humans and computers to discover and understand the capabilities of the service without access to source code, documentation, or through network traffic inspection", quoted from <https://swagger.io/specification/>

**OPNFV:** Open Platform for NFV Project

- carrier-grade, integrated reference platform integrating ODL and OpenStack, designed to host VNFs

- will work closely with the [ETSI](#) and others to press for consistent implementation of open standards.
- <https://www.sdxcentral.com/nfv/definitions/opnfv/>

#### **Orchestration:**

- the definition and execution of workflows or processes to manage the completion of a task
- will not involve human intervention/decision/guidance in the vast majority of cases

**OSAM:** Open Source Access Manager

**OSS:** Operations Support System

#### **OSGI:**

- a modular system and a service platform for the Java programming language that implements a complete and dynamic [component model](#) (OpenDaylight uses this) ([Wikipedia](#))

**P4** : "is a programming language designed to allow programming of packet forwarding planes" quoted from [https://en.wikipedia.org/wiki/P4\\_\(programming\\_language\)](https://en.wikipedia.org/wiki/P4_(programming_language))

**PAP:** Policy Administration Point (ONAP)

**PCE:** Path Computation and Element (ONAP)

**PCI:** Physical Cell ID

**pCPE:** physical Customer Premise Equipment

**PDP-x:** Policy Decision Point - XACML (ONAP)

**PDP-d:** Policy Decision Point - Drools (ONAP)

**PO:** Platform Orchestrator

**PoC:** Proof of Concept

**POMBA:** Post Orchestration Model Based Audit

**PNDA:** Open source Platform for Network Data Analytics

- part of DCAE - see [Integrating PNDA](#)

**PNF:** Physical Network Function

- a network function that runs in a hardware appliance

#### **Portal:**

- ONAP user interface; provides access to design, analytics and operational control/administration via a common role-based menu or dashboard. Includes SDK to drive UI consistency.

#### **Product:**

- in ONAP, a composition of Services

**RCA:** Root Cause Analysis

**RCT:** Reference Connection Tool

#### **Recipe:**

- within Chef, the most fundamental configuration element (see the [Style Guide](#))

#### **Resource:**

- in ONAP, a fundamental capability

**REST:** REpresentational State Transfer ([Wikipedia](#))

**RESTCONF:** REST + NETCONF

- IETF draft that describes how to map a YANG specification to a RESTful interface. Read the IETF draft and see its current status [here](#) or [here](#).

**RO:** Resource Orchestrator

**RPC:** Remote Procedure Call

**S3P:** Stability, Security, Scalability, Performance. See also [Platform Maturity Requirements \(S3P\)](#).

**SDC:** Service Design and Creation (component of ONAP for visual modeling and design)

**SDN:** Software-Defined Networking

- separates the control (brains) and forwarding (muscle) planes for a centralized view of the network, for more efficient orchestration and automation of network services.
- <https://www.sdxcentral.com/nfv/definitions/virtual-network-function/>
- <https://www.sdxcentral.com/nfv/definitions/which-is-better-sdn-or-nfv/>
- [Wikipedia article](#)

**SDNC (formerly SDN-C):** Software Defined Network Controller (part of ONAP)

- its operators accept Comma Separated Value files describing networks

**SDN-GP:** Software Defined Network - Global Platform

**Service:**

- in ONAP, a composition of Resources

**SDN-R:** application of SDNC - see [SDN-R objectives](#)

**SEBA:** SDN-Enabled Broadband Access, see also:

- <https://www.opennetworking.org/reference-designs/>
- <https://www.opennetworking.org/seba/>

**SLA:** Service Level Agreement

- a contract between a service provider (either internal or external) and the end user that defines the level of service expected from the service provider. SLAs are output-based in that their purpose is specifically to define what the customer will receive.

**SLI:** Service Logic Interpreter

- within the Application Controller (APP-C), executes Directed Graphs (DGs)

**SME:** Subject Matter Expert

**SMTP:** Simple Mail Transfer Protocol

**SNMP:** Simple Network Management Protocol

**SO:** [Service Orchestrator](#) (Project)

**SOT:** Source Of Truth

- external system where data object originates

**SR-IOV:** Single-Root Input/Output Virtualization ([Wikipedia](#))

- a network interface that allows the isolation of the [PCI Express](#) resources for manageability and performance reasons

**SSH:** Secure Shell

**SSL:** Secure Sockets Layer ([Wikipedia](#)), precursor to TLS

**SUPP:** (Vendor Specific) short for supplement, changing a connection before activation

**SVNFM:** (Vendor) Specific Virtual Network Function Manger

**Swagger:** legacy name for the OpenAPI Specification

**TCP:** Transmission Control Protocol

**TEM:** Telecom Electronics Manufacturer

**tenant:**

- a group of users who share a common access with specific privileges to a software instance on a server ([Wikipedia](#)). This terminology is used in [OpenStack](#)

**TLS:** Transport Layer Security ([Wikipedia](#)), standardized replacement for Secure Sockets Layer (SSL)

**TOSCA:** [Topology and Orchestration Specification for Cloud Applications](#) (OASIS spec)

**TPS:** Transactions Per Second

**TSC:** Technical Steering Committee. Establishes work flows and procedures, criteria for contributors and committers, and any additional roles and responsibilities required on ONAP projects.

**UEB:** An event bus interface that has been replaced by DMaaP.

**U-UI:** Usecase UI = [Usecase User Interface](#) (Project)

**vCE:** virtual CE (Customer Edge) router (an example VNF)

**vCPE:** Virtual Customer Premise Equipment

**vDNS:** Virtual Domain Name Server (an example VNF)

**VDU:** Virtualisation Deployment Unit - The Virtualisation Deployment Unit (VDU) is a construct supporting the description of the deployment and operational behaviour of a VNFC.

- see [class Vdu](#)

**VES:** Virtual function Event Stream

- OPNFV proposed standard common event data model for telemetry-related data ([PowerPoint of proposal](#))
- used by vFirewall ONAP demonstration VNF to report heartbeats, faults, measurements, etc.
- DCAE has a VES Collector - see [High Volume VES Collector](#)

**vf:** Virtual Firewall (an example VNF)

**VF:** Virtual Function

- an entity that may be modeled and subsequently instantiated, which takes on the responsibility of handling a particular function that when instantiated, will run on one or more virtual machines within the cloud.
- in [48534184](#), a VF is equivalent to a Resource

**VFC:** [Virtual Function Controller](#) (the ONAP project)

**VFC:** Virtual Function Component ([Resource Onboarding](#))

- software component of a [48534184](#) that is packaged into one or more images and is capable of running in its own container
- in [48534184](#), a VFC is configured as a sub-component of a [VSP](#).
- the smallest granularity of function visible to ONAP designers

**vfModule:** Virtual Function Module

- A [48534184](#) stack that defines the compute, network, licensing and other Heat resources needed to instantiate one or more VFCs associated with a VNF. In a VNF, one vfModule is typically designated as a base module. Any resources defined in the base module Heat template can be exposed to all other vfModules by declaring their resource UUID as Heat outputs. Beyond the base module, other VNF vfModules are typically referred to as expansion modules.

**VID:** Virtual Instantiation Deployment

- a Portal GUI to trigger MSO instantiation of services and components

**VID:** [Virtual Infrastructure Deployment](#) (Project)

**VIM:** Virtualized Infrastructure Manager

- part of MANO
- controls and manages the NFVI compute, storage, and network resources

**VLAN:** Virtual Local Area Network

**VM:** Virtual Machine

**VNF:** Virtual Network Function ([link](#))

- a virtualized task formerly carried out by proprietary, dedicated network hardware. (Examples: virtual firewall, virtual DNS.)
- in ONAP, a VNF is a Resource
- a VNF is a specific kind of Vendor Software Product

**VNFC:** Virtual Network Function Component

- a part of a VNF. It is a stand-alone executable that is loosely-coupled, granular, re-usable, and responsible for a single capability.

**VNFD:** VNF Descriptor

- the term used in VNF SDK project - it refers to modeling of VNF - see [Models - VNF Descriptor & Service Descriptor](#)

**VNFM:** VNF Manager

**VNO:** Virtual Network Operator

**VNFSDK:** ONAP project

**vPE:** virtual PE (Provider Edge) router (an example of a VNF)



**VPP:** Vector Packet Processing ([link](#))

- open-source version of Cisco's VPP
- a platform that provides switch/router functionality
- used by vPacketGenerator, vFirewall, and vLoadBalancer Virtual Network Functions in ONAP

**VSP:** Vendor Software Product (from SDC Demo Guide)

- example: a VNF
- In ONAP, a VSP is a Resource.

**VTP:** VNF Test Platform

- part of VNFSDK - see [VNF Test Platform \(VTP\)](#)

**VVP:** [VNF Validation Program](#) (Project)

**WAR:** Web application ARchive ([Wikipedia](#))

- a JAR file used to distribute a collection of JavaServer Pages, Java Servlets, Java classes, XML files, tag libraries, static web pages (HTML and related files) and other resources that together constitute a web application

**XACML++:** eXtensible Access Control Markup Language (OASIS standard, extended)

- a declarative fine-grained, attribute-based access control policy language, an architecture, and a processing model describing how to evaluate access requests according to the rules defined in policies. It is an [Attribute-Based Access Control](#) system (ABAC), where attributes (bits of data) associated with a user or action or resource are inputs into the decision of whether a given user may access a given resource in a particular way.

**YAML:**

- machine parsable data serialization format designed for human readability and interaction with scripting languages such as Perl and Python ([yaml.org](#)) ([Wikipedia](#))

**YANG:** A Data Modeling Language for the Network Configuration Protocol (NETCONF)

- <https://www.rfc-editor.org/info/rfc6020> or <https://tools.ietf.org/html/rfc6020>
- <https://datatracker.ietf.org/doc/html/rfc7950>