



# ONAP Service Assurance VES Spec 6.0 Draft updates – Request for Comments

Alok Gupta  
+1 (732)-420-7007  
[ag1367@att.com](mailto:ag1367@att.com)

Date: May 10<sup>th</sup>, 2018

# VES (NF Event Streaming) Specification 6.0 Changes

## ■ Section 1

- VES Meaning: Expanded to include event streaming for VNF, PNF and Infrastructure
- eventNames Construction: {DomainAbbreviation}\_{vnf/vnfcNameCode}\_{DescriptionOfInfoBeingConveyed}
- Added a list of all the latest field block version numbers in this version of the API spec.

## ■ Section 3

- Removed Throttling requirements. App-C or controller need to support throttling.

## ■ Section 4

- hashMaps. Changed all name-value pair structures to hashMaps
  - Name= "name" value ="value" => "name" = "Value"
- Use of Optional fields: If the event publisher collects a field that is identified as optional in the data structures below, then the event publisher *must* send that field.
- Extensible Field Use Clarification: VES contains various extensible structures (e.g., hashMap) that enable event publishers to send information that has not been explicitly defined in VES data structures.
  - Event publishers must not send information through extensible structures where VES has explicitly defined fields for that information. For example, event publishers must not send information like cpuDdle, through an extensible structure, because VES has explicitly defined a cpuUsage.cpuDdle field for the communication of that information.
  - Keys sent through extensible fields must use camel casing to separate words and acronyms; only the first letter of each acronym shall be capitalized.

- hashMaps. Changed all name-value pair structures to hashMaps causing the following data model and JSON schema (to v29.0) changes:

# VES (NF Event Streaming) Specification 6.0 Changes

## ■ Section 4 – Contd

- Common Event Data Types: Changed vendorVnfNameFields to vendorNfNameFields in SipSignaling and Voice Quality Domains
- Common Event Header clarified the following descriptions:
  - eventId: Event key that is unique to the event source. The key must be unique within notification life cycle similar to EventID from 3GPP. It could be a sequential number, or a composite key formed from the event fields, such as sourceName\_alarmCondition\_startEpoch. The eventId should not include whitespace. For fault events, eventId is the eventId of the initial alarm; if the same alarm is raised again for changed, acknowledged or cleared cases, eventId must be the same as the initial alarm (along with the same startEpochMicrosec and an incremental sequence number).
  - reportingEntityName: Name of the entity reporting the event or detecting a problem in another vnf/vm or pnf which is experiencing the problem. May be the same as the sourceName. For synthetic events generated by DCAE, it is the name of the app generating the event.
  - sourceName: Name of the entity experiencing the event issue, which may be detected and reported by a separate reporting entity. The sourceName identifies the device for which data is collected. A valid sourceName must be inventoried in A&AI. If sourceName is a VNFC or VM, then the event must be reporting data for that particular VNFC or VM. If the sourceName is a VNF, comprised of multiple VNFCs, the data must be reported/aggregated at the VNF level. Data for individual VNFC must not be included in the VNF sourceName event.
  - startEpochMicroseconds: the earliest unix time aka epoch time associated with the event from any component--as microseconds elapsed since 1 Jan 1970 not including leap seconds. For measurements and heartbeats, where events are collected over predefined intervals, startEpochMicrosec shall be rounded to the nearest interval boundary (e.g., the epoch equivalent of 3:00PM, 3:10PM, 3:20PM, etc...). For fault events, startEpochMicrosec is the timestamp of the initial alarm; if the same alarm is raised again for changed, acknowledged or cleared cases, startEpoch Microsec must be the same as the initial alarm (along with the same eventId and an incremental sequence number).
  - added 'notification and pnfRegistrion domains
- Fault Domain Section 4.2.1: clarified the definitions of alarmCondition and specificProblem

- hashMaps. Changed all name-value pair structures to hashMaps causing the following data model and JSON schema (to v29.0) changes:

# VES (NF Event Streaming) Specification 6.0 Changes

- Section 4 – Contd
  - Fault Domain: clarified the following definitions
    - alarmCondition: Short name of the alarm condition/problem, such as a trap name. Should not have white space (e.g., tpLgCgiNotInConfig, BfdSessionDown, linkDown, etc...)
    - specificProblem: Description of the alarm or problem (e.g., ‘This event is sent when the LG is asked to perform a location for a CGI that is not in its configuration’)
  - Measurements Domain: changed the name of this domain from ‘measurementsForVfScaling’ to ‘measurement’
  - Measurement Domain: Updates following fields: cpuUsage, diskUsage, memoryUsage, vNicPerformance to nicPerformance and added load Array, hugePagesArray,
  - Add Notification and pnfRegistrion fields
- Section 6: specified message size limit of 2 Meg

**Note: The Excel file has IPMI metrics in the measurement domain, which will be added to the VES Listener Requirement 6.0 and JSON Schema in the final update.**

- hashMaps. Changed all name-value pair structures to hashMaps causing the following data model and JSON schema (to v29.0) changes:

# VES Event Registration Specification 2.0 Changes

- Updated examples to align with VES Event Listener 6.0
- Clarified Action keyword
- Modified the rules example to conform to the Complex Conditions and Rules sections.

- hashMaps. Changed all name-value pair structures to hashMaps causing the following data model and JSON schema (to v29.0) changes:

# Requirements Files

- VES Listener Requirements 6.0 Draft



Microsoft Word  
Document

- VES JSON Schema 29.0 draft



C:\Users\ag1367\  
nents\Next Gen SA

- VES Event Registration 2.0 Draft



Microsoft Word  
Document

- VES Excel 29.0



Microsoft Excel  
Worksheet