VNFRQTS Requirement Format discussion

VNFRQTS-23 requires the VNFRQTS project to agree on a "standard" format for requirements for the ONAP Amsterdam release. This task is assigned for resolution in Sprint 2. There are a number of aspects top this task - Key words, processes and tooling

Key Words

RFC 2119 provides definitions for keywords to signify requirements:

- MUST
- MUST NOT
- SHOULD
- SHOULD NOT
- MAY

The Seed documents from ECOMP use the MUST / SHOULD keywords. Some of the other seed documents do not use any Keywords.

Some task is required to ensure consistent use of keywords in the source material

Requirement Subject

From the Project Charter, the **Subject** of the Requirements is the **VNF** and not the platform or its components.

One way to enforce this scope is to structure the requirement such that they are written as complete sentences with VNF as the subject e.g.*The VNF MUST xxxx.*

To allow for conditional requirements, the following is also allowed: *If <condition> the VNF MUST|SHOULD|MAY ..."*

Some of the requirements in the seed documents are not structured as stand alone complete sentences. Some task would be required to align the requirements to a consistent structure.

Requirement Structure

The Project Charter also mentions the development of "EPIC statements". The ONAP JIRA wiki uses the term User Story Syntax for the same structure:

User Story Syntax

```
As a <user> = who
I want to <be able to do ABC> = what
So that <XYZ can be done> = why
```

The requirements from the seed document are not in this form, and so some task would be required to structure them appropriately

Processes

Use Cases for Requirements

- Requirements as text in a .hmtl formatted VNF Requirements deliverable for use in RFPs - *EPIC* - VNFRQTS-6
- Requirements as table for conformance mapping - *User Story* - VNFRQTS-27
- Correlate VNF Requirements and VNF test cases for validation - *EPIC* - VNFRQTS-8
- Subsets of the requirements apply for particular conformance mapping activities beyond complete VNFS e.g.
    - VNF Requirements impacting the VNF Package definition - *User Story* - VNFRQTS-28
    - VNF Requirements for configurable parameters - *User Story* - VNFRQTS-28

Requirement attributes

Attributes may be required to support subsetting the list of requirements.

Once identified, VNF Requirements are not expected to be very dynamic, but there may be some changes required between ONAP releases.

A VNF Requirement version attribute may be required to accommodate this.

Some VNF Requirements may be forward looking ie anticipated to apply in future releases.

Some VNF Requirements may be optional

Tooling

The VNF Requirements "master" in the .rst files in the vnfrqts/requirements repo.

The documentation projects Jenkins job builds the .html onto the http://onap.readthedocs.io website

Tooling is needed to extract just the requirements sentence from the table format. This is expected to require the VNF requirements to be tagged appropriately in the ,rst source files

Requirement identifiers

Requirements should be uniquely identified. Once assigned requirements identified should remain across releases and revisions. New requirments may be added in a release. numbers may be retired.

Currently structured as single sentences starting with R-XXXXX (allows for searching, extraction from narrative text).

Numbering semantics: Yes/No

no semantics: implies a random number/ string as an identifier. Requires a tool of some sort to generate these random numbers. Requires some process for contributors/ editors to aquire and use these random numbers

yes semantics: e.g. sequential, chapter sequential, sequential with spaces etc.

How many digits:

Currently ~300 numbered requirements in the seed documents, but some seed material did not have requirements explicitly numbered.

suggest 5 digits allows plenty of room for expansion/ revision / retirement.

Number of digits could be changed in a future release if needed.

Application for Amsterdam.

M4 is originally 9/14, now 9/28, so tool chain changes at this stage might be problematic.

could manually apply numbers for this release and consider other mechanisms in future?

could investigate sequential numbering mechanisms in Sphinx etc.