

# RI testing that enumerated values are correct

Written by: David Herron - Evoke Systems - [david@davidherron.com](mailto:david@davidherron.com) or [david.herron@evokesystems.com](mailto:david.herron@evokesystems.com)

Date: February 1, 2025

## Introduction

In [Recommended Implementation for utilizing enumeration schemas in the OpenADR RI](#) we discussed how the RI, or any VEN or VTN, can use the schema files to implement data validation.

In [Definition.md](#), we have several enumeration tables describing certain values that can be sent via the OpenADR protocol. Some are sent by the VTN to VENs, and others are sent by VENs to the VTN.

This paper focuses on how the RI and the test suite should validate conformance with the OpenADR protocol.

We start from assuming that the RI VTN and VEN implementations will incorporate the schema files, and utilize auto-generated validation functions in a manner similar to what is described. Obviously the tools used on Node.js/TypeScript do not apply to the RI which is implemented in Python.

By having implemented data validation for the values corresponding to the enumeration tables the RI will exhibit this behavior:

- RI VEN will detect invalid values supplied by the VTN, and throw errors
- RI VEN will ensure it does not generate invalid values
- RI VTN will detect invalid values supplied by a VEN, and throw errors
- RI VTN will ensure it does not generate invalid values

## Compliance Testing

Certain software products - such as communications protocol, programming platforms where there are multiple implementations - need to validate that an implementation is compliant with the specification.

There is a specification - such as the Java APIs or the OpenADR 3.x protocol - and there are multiple implementations of that specification created by multiple

organizations. Compliance testing aims to ensure that the implementation is correctly conforming to the specification.

Therefore the compliance tests for OpenADR must exercise every API endpoint as well as ensure that data is correctly structured. The correctly structured data includes the objects whose content is defined in the enumeration tables.

## Testing a VTN for compliant behavior on enumerated values

The VTN being tested, to be compliant, should reject objects that have incorrect enumerated values, and, it should not produce objects with incorrect enumerated values.

Therefore - compatibility tests to ensure that behavior should

1. POST and UPDATE methods - For each of the enumerated tables - the tests sends objects containing legitimate values -- RESULT is success code, and returned object also validates
2. POST and UPDATE methods - The tests send objects with bad values but match the correct data type -- RESULT is the correct failure code
3. POST and UPDATE methods - The tests send objects with legitimate values but with too many items -- RESULT is the correct failure code
4. POST and UPDATE methods - The tests send objects with incorrect data type -- RESULT is the correct failure code

All objects sent by the VTN to the test suite must be validated.

## Testing a VEN for compliant behavior on enumerated values

The VEN being tested, to be compliant, should reject objects that have incorrect enumerated values, and, it should not produce objects with incorrect enumerated values.

Therefore - compatibility tests to ensure that behavior should

1. POST and UPDATE methods - For each of the enumerated tables - the VEN can correctly send objects containing legitimate values -- RESULT is that the VEN sends it, the VTN validates it, and returns a success code, and returned object also validates

2. POST and UPDATE methods - The VEN fails to send objects with bad values but match the correct data type -- RESULT is that the request doesn't make it out of the VEN
3. POST and UPDATE methods - The VEN fails to send objects with legitimate values but with too many items -- RESULT is that the request doesn't make it out of the VEN
4. POST and UPDATE methods - The VEN fails to send objects with incorrect data type -- RESULT is that the request doesn't make it out of the VEN

All objects received by from the VTN must be validated.