**Discussion and resolutions:**

1. Some messages will have both request and response packets. We need to discuss how to handle these as an artifact of your CSV files. I can “fake” these, but need an identifier and a trigger from your CSV file to maintain fidelity in the JSON posts to L2.
2. We need “identifiers” as the first parameter in your CSV. Each line needs a code (identifier) that will allow me to trigger the correct “fill-in” or hard coded parameters that I would normally get from the PCAP TCP packets. These are things like IP addresses, ports, etc.

I suggest the following:

MS-ODEventNotification = MS-ODEN

MS-GetLatestReadings = MS-GLR

MS-GetLatestReadingsByMeterGroup = MS-GLRBMG

MS-IniateDisconnectConnect = MS-IDC

DNP-SubstationKwH = DNP-SKW

DNP-SubstationBreakerSwitchStatus = DNP-SBSS

DNP-SubstationVoltageControl = DNP-SVC

These codes will direct the line of CSV you have scheduled to the correct JSON message structure and fill in the hard-coded parameters that are “faked” for the TCP stuff.

1. Example CSV based on the message mapping tables.

Data Dictionary Elements:

*packet\_timestamp, Identifier, application\_timestamp, operation, string\_value*

1506033686.12, **MS-ODEN**, 15060334230.12, Outage, tpm3\_C1\_R2-12-47-2\_tm\_1

1506033695.24, **MS-ODEN**, 1506034123.12, Outage, tpm3\_C1\_R2-12-47-2\_tm\_2

.

.

.

*\*The first epoch.ms timestamp is in effect your schedule of events*

**MS-METHOD:** ODEventNotification (Unsolicited RESPONSE message)

|  |  |  |  |
| --- | --- | --- | --- |
| **CSV-Parameter{}** | **Msg Creation** | **L1-JSON** | **L2 Fields** |
| PCAP TimeStamp {epoch.ms}1  Actual time of event {epoch.ms}2  From List3 {string}  MeterNo {string} | \*MultiSpeak Payload  packet\_timestamp  eventTime  outageEventType  meterNo | \*Code Below  TCP-Frame  eventTime  outageEventType  meterNo | \*Based on Data Dict  receipt\_timestamp  **packet\_timestamp**  **application\_timestamp**  message\_id {generated GUID}  source\_actor “RNI”  src\_ip “192.168.1.1”  src\_port “16000”  dest\_actor “OMS”  dest\_ip 192.168.1.2  dest\_port “12345”  **operation {enum}**  **string\_value**  double\_value {0}  units “NULL”  stream “METER-EVENTS”  protocol “MULTISPEAK” |

**NOTES**:

1-This is effectively the receipt of the message in the PCAP “arrival time”, it is the *scheduled* packet time for all messages in the CSV

2- The ODEvent contains a timestamp of when the event was logged

3-Inferred, Instantaneous, NoResponse, Outage, PowerOff, PowerOn, Restoration

<tns:ODEvents>

<tns:outageDetectionEvent utility="PNNL-FNCS-SYSTEM">

<tns:eventTime>**{datetimestamp}**</tns:eventTime>

<tns:outageEventType>{enum: “**Inferred, Instantaneous, NoResponse, Outage, PowerOff, PowerOn,   
 Restoration”**}</tns:outageEventType>

<tns:outageDetectDeviceID>MeterID</tns:outageDetectDeviceID>

<tns:outageDetectDeviceType>”Meter**”**</tns:outageDetectDeviceType>

<tns:outageLocation>

<tns:servLoc>MeterID</tns:servLoc>

<tns:meterNo>**{MeterNum}**</tns:meterNo>

</tns:outageLocation>

<tns:messageList>

<tns:message>

<tns:comments>”SuccessEvent**”**</tns:comments>

</tns:message>

</tns:messageList>

<tns:priority>Enum: “NeedsAttention, Normal, Urgent**”**</tns:priority>

</tns:outageDetectionEvent>

</tns:ODEvents>

</tns:ODEventNotification>

</soap:Body>

</soap:Envelope>

**BELOW IS FOR MY REFERENCE ONLY!**

# MultiSpeak

**MS-METHOD:** GetLatestMeterReadingsByMeterGroupResponse (RQST/RSPN)

Multispeak

**GetLatestMeterReadingsByMeterGroup**

Message Encoder

* PNNL method complete has data generated via PCAP and L1-L2 complete 9/18/17

**MS-METHOD:** GetLatestReadings (RQST/RSPN)

Multispeak

**GetLatestReadings**

Message Encoder

* PNNL has NO method implemented, definition/msg structure sent Sep 9, 2017

**MS-METHOD:** ODEventNotification (RQST/RSPN)

Multispeak

**ODEventNotification**

Message Encoder

* PNNL has NO method implemented, definition/msg structure sent Sep 9, 2017

**MS-METHOD:** IniateConnectDisconnect (RQST/RSPN)

Multispeak

**IniateConnectDisconnect**

Message Encoder

* PNNL has NO method implemented, definition/msg structure sent {TBD}

# DNP3 (RESPONSE OBJECTS)

**DNP-GRP/VAR:** Substation Analog Power Readings (8-elements 2X Subs)

DNP3 Value Status Message Encoder

**GRP-30 VAR-5**

IEEE Spec. Pgs. 442, 568

* PNNL method complete and has data generated via PCAP and L1-L2 complete 9/18/17

**DNP-GRP/VAR:** Substation Switches (69-elements @ 3 per switch, 2X Subs)

DNP3 Switch Status Control Message

**GRP-1 VAR-2**

IEEE Spec. Pgs. 438, 512

* PNNL method complete and has data generated via PCAP and L1-L2 complete 9/18/17

**DNP-GRP/VAR:** Substation Voltage Control (X-elements 2X Subs)

DNP3 Voltage Control Message

**GRP-x VAR-y**

IEEE Spec. Pgs. xxx, xxx

**NOTE: this is a control signal “to” the substation. Need to determine exactly what value it controls to ensure units, and expected grid effect.**

* PNNL has NO method implemented, definition/msg structure sent {TBD}