Last Updated April 26, 2022

Data Feed Self-Validation Check List

1. □Validated WZDx data feed using JSON schema validator.
2. □Using a mapping tool, visually validated that all road events locations are within my jurisdiction, made sure that the locations marked make sense for work zones, and spot checked the feature's properties.
3. □Reviewed current business rules and checked WZDx data feed for adherence.
4. □Reviewed complex work zones in WZDx feed to validate that your system implements the work zone accurately.
5. □If using event_type detour, have validated that it connects to the appropriate road event(s).
6. □Verified that FeedInfo contact name and email are the appropriate technical contact for questions involving the data feed.
7. □Verified that FeedInfo update_frequency is accurate for your system.
8. □Validated that the geospatial coordinates adhere to the level of accuracy provided by the mapping system.

Name ___________________________________ Contact Email ______________________________

WZDx version_________

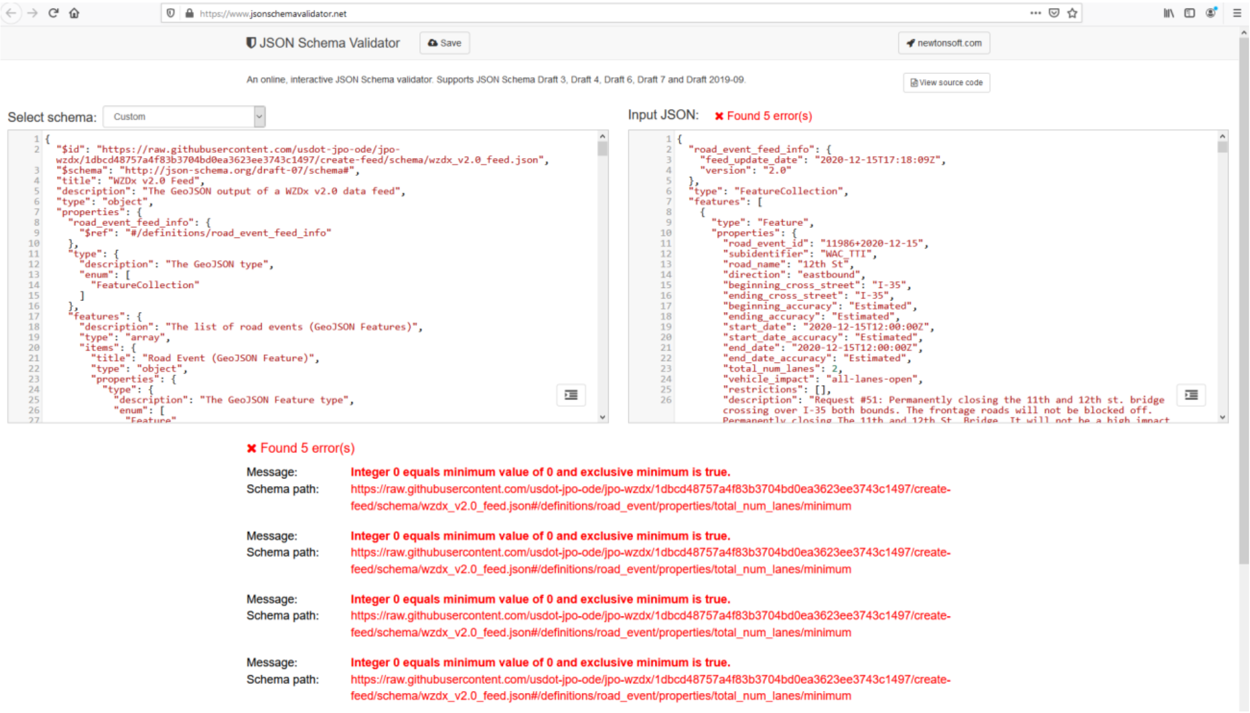
Steps to take when validating if a new WZDx data feed is compliant

1. Validate if the feed follows the required WZDx schema version.

a. Go to https://www.jsonschemavalidator.net/

b. Paste the JSON Schema for the WZDx version you seek to verify on the left-hand box (WZDx schemas can be found here: https://github.com/usdot-jpo-ode/wzdx/tree/main/schemas. Paste the content of the feed on the right-hand box.

c. Validation results will appear below your inputs. Note down any inconsistencies that the feed has with the WZDx JSON Schema. Reach out to help desk or the technical assistance forum if you would like assistance resolving these errors.



An online, interactive JSON Schema validator. Supports JSON Schema Draft 3, Draft 4, Draft 6, Draft 7 and Draft 2019-09.

Select schema: Custom

```
{
  "$id": "https://raw.githubusercontent.com/usdot-jpo-ode/jpo-
  wzdx/1dbcd48757a4f83b3704bd0ea3623ee3743c1497/create-feed/schema/wzdx_v2.0_feed.json",
  "$schema": "http://json-schema.org/draft-07/schema#",
  "title": "WZDx v2.0 Feed",
  "description": "The GeoJSON output of a WZDx v2.0 data feed",
  "type": "object",
  "properties": {
    "road_event_feed_info": {
      "$ref": "#/definitions/road_event_feed_info"
    },
    "type": {
      "description": "The GeoJSON type",
      "enum": [
        "FeatureCollection"
      ]
    },
    "features": {
      "description": "The list of road events (GeoJSON Features)",
      "type": "array",
      "items": {
        "title": "Road Event (GeoJSON Feature)",
        "type": "object",
        "properties": {
          "type": {
            "description": "The GeoJSON Feature type",
            "enum": [
              "Feature"
```

Input JSON: ✖ Found 5 error(s)

```
{
  "road_event_feed_info": {
    "feed_update_date": "2020-12-15T17:18:09Z",
    "version": "2.0"
  },
  "type": "FeatureCollection",
  "features": [
    {
      "type": "Feature",
      "properties": {
        "road_event_id": "11986+2020-12-15",
        "subidentifier": "WAC_TTI",
        "road_name": "12th St",
        "direction": "eastbound",
        "beginning_cross_street": "I-35",
        "ending_cross_street": "I-35",
        "beginning_accuracy": "Estimated",
        "ending_accuracy": "Estimated",
        "start_date": "2020-12-15T12:00:00Z",
        "start_date_accuracy": "Estimated",
        "end_date": "2020-12-15T12:00:00Z",
        "end_date_accuracy": "Estimated",
        "total_num_lanes": 2,
        "vehicle_impact": "all-lanes-open",
        "restrictions": [],
        "description": "Request #51: Permanently closing the 11th and 12th st. bridge
        crossing over I-35 both bounds. The frontage roads will not be blocked off.
        Permanently closing The 11th and 12th St. Bridge. It will not be a high impact
```

✖ Found 5 error(s)

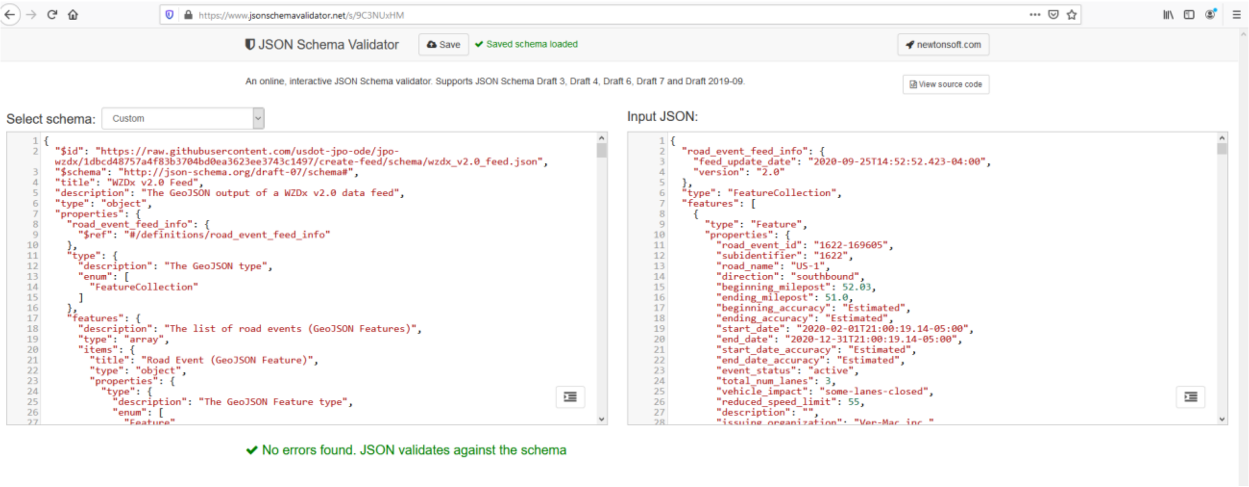
```
Message:        Integer 0 equals minimum value of 0 and exclusive minimum is true.
Schema path:    https://raw.githubusercontent.com/usdot-jpo-ode/jpo-wzdx/1dbcd48757a4f83b3704bd0ea3623ee3743c1497/create-
                feed/schema/wzdx_v2.0_feed.json#/definitions/road_event/properties/total_num_lanes/minimum

Message:        Integer 0 equals minimum value of 0 and exclusive minimum is true.
Schema path:    https://raw.githubusercontent.com/usdot-jpo-ode/jpo-wzdx/1dbcd48757a4f83b3704bd0ea3623ee3743c1497/create-
                feed/schema/wzdx_v2.0_feed.json#/definitions/road_event/properties/total_num_lanes/minimum

Message:        Integer 0 equals minimum value of 0 and exclusive minimum is true.
Schema path:    https://raw.githubusercontent.com/usdot-jpo-ode/jpo-wzdx/1dbcd48757a4f83b3704bd0ea3623ee3743c1497/create-
                feed/schema/wzdx_v2.0_feed.json#/definitions/road_event/properties/total_num_lanes/minimum

Message:        Integer 0 equals minimum value of 0 and exclusive minimum is true.
Schema path:    https://raw.githubusercontent.com/usdot-jpo-ode/jpo-wzdx/1dbcd48757a4f83b3704bd0ea3623ee3743c1497/create-
                feed/schema/wzdx_v2.0_feed.json#/definitions/road_event/properties/total_num_lanes/minimum
```

Figure 1: TxDOT example (retrieved 12/15/20)



An online, interactive JSON Schema validator. Supports JSON Schema Draft 3, Draft 4, Draft 6, Draft 7 and Draft 2019-09.

Select schema: Custom

```
{
  "$id": "https://raw.githubusercontent.com/usdot-jpo-ode/jpo-
  wzdx/1dbcd48757a4f83b3704bd0ea3623ee3743c1497/create-feed/schema/wzdx_v2.0_feed.json",
  "$schema": "http://json-schema.org/draft-07/schema#",
  "title": "WZDx v2.0 Feed",
  "description": "The GeoJSON output of a WZDx v2.0 data feed",
  "type": "object",
  "properties": {
    "road_event_feed_info": {
      "$ref": "#/definitions/road_event_feed_info"
    },
    "type": {
      "description": "The GeoJSON type",
      "enum": [
        "FeatureCollection"
      ]
    },
    "features": {
      "description": "The list of road events (GeoJSON Features)",
      "type": "array",
      "items": {
        "title": "Road Event (GeoJSON Feature)",
        "type": "object",
        "properties": {
          "type": {
            "description": "The GeoJSON Feature type",
            "enum": [
              "Feature"
```

Input JSON:

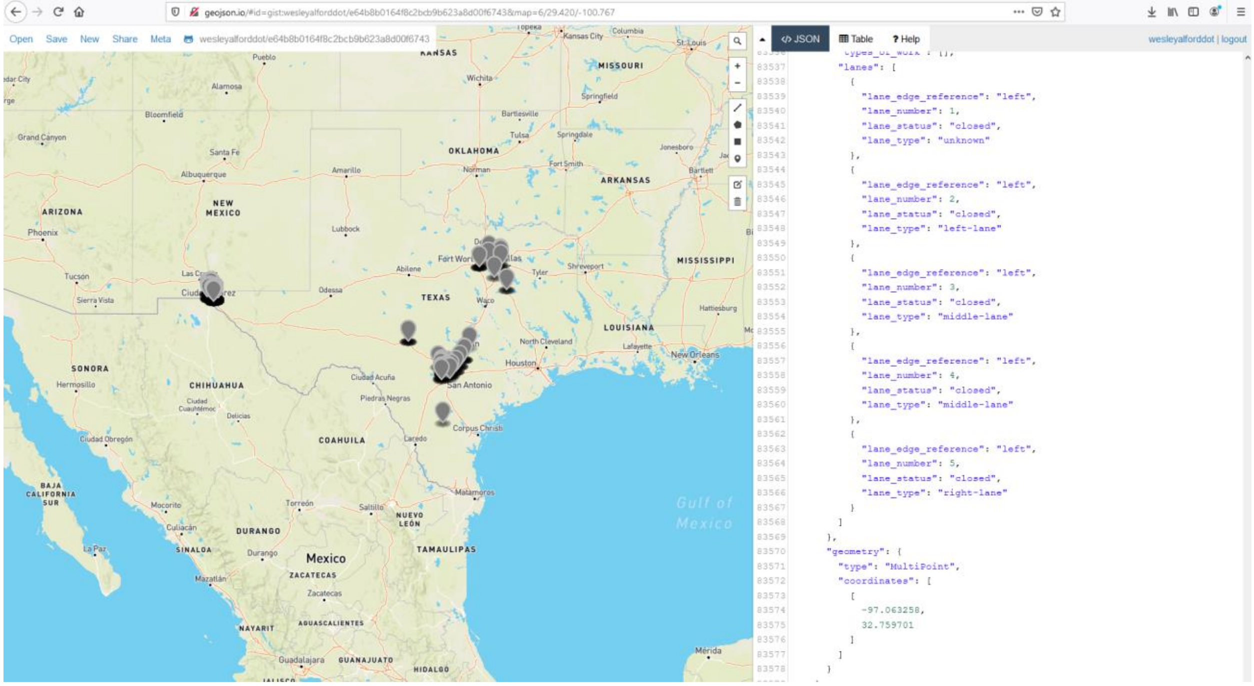
```
{
  "road_event_feed_info": {
    "feed_update_date": "2020-09-25T14:52:52.423-04:00",
    "version": "2.0"
  },
  "type": "FeatureCollection",
  "features": [
    {
      "type": "Feature",
      "properties": {
        "road_event_id": "1622-169605",
        "subidentifier": "1622",
        "road_name": "US-1",
        "direction": "southbound",
        "beginning_milepost": 52.03,
        "ending_milepost": 51.0,
        "beginning_accuracy": "Estimated",
        "ending_accuracy": "Estimated",
        "start_date": "2020-02-01T21:00:19.14-05:00",
        "end_date": "2020-12-31T21:00:19.14-05:00",
        "start_date_accuracy": "Estimated",
        "end_date_accuracy": "Estimated",
        "event_status": "active",
        "total_num_lanes": 3,
        "vehicle_impact": "some-lanes-closed",
        "reduced_speed_limit": 55,
        "description": "",
        "issuing_organization": "Ver-Mac inc."
```

✓ No errors found. JSON validates against the schema

Figure 2: MassDOT example (retrieved 12/15/20)

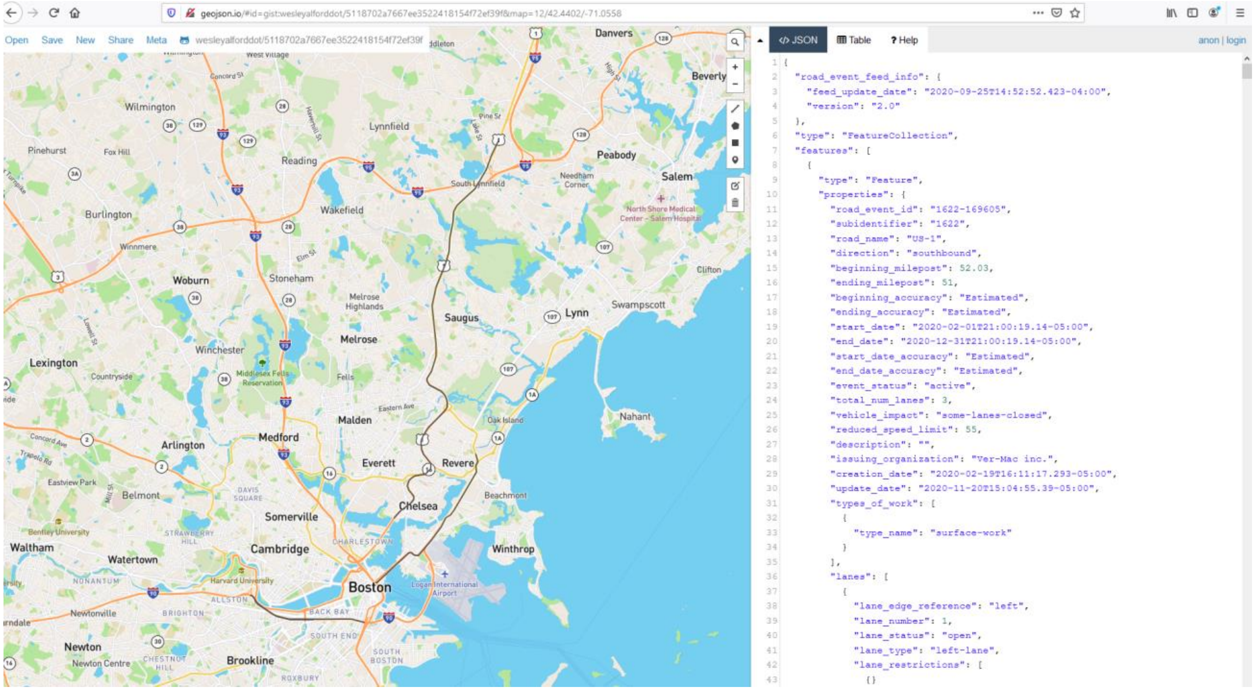
2. Validate if the feed contains valid GeoJSON and valid locations

a. Go to https://GeoJSON.io/
b. Paste the feed's current content in the right-hand box. The map should change on the left if the GeoJSON is valid. If it is not valid GeoJSON, the right-hand box will have red margins where the GeoJSON is invalid.
c. Spot check for the following in the map:
   i. Are the locations marked within the feed boundary indicated by the data provider (by state or other administrative boundaries, depending on how you are aggregating data)?
   ii. Make sure that the locations marked make sense for work zones. (e.g., Are the locations on roadways or in the middle of a lake?)
   iii. Click on some markers to spot check the feature's properties - does the road_names value match the road that it is on?


geojson.io/#id=gist:wesleyalforddot/e64b8b0164f8c2bcb9b623a8d00f6743&map=6/29.420/-100.767
Open
Save
New
Share
Meta
wesleyalforddot/e64b8b0164f8c2bcb9b623a8d00f6743
JSON
Table
Help
wesleyalforddot | logout

```
        "types_of_work": [],
        "lanes": [
          {
            "lane_edge_reference": "left",
            "lane_number": 1,
            "lane_status": "closed",
            "lane_type": "unknown"
          },
          {
            "lane_edge_reference": "left",
            "lane_number": 2,
            "lane_status": "closed",
            "lane_type": "left-lane"
          },
          {
            "lane_edge_reference": "left",
            "lane_number": 3,
            "lane_status": "closed",
            "lane_type": "middle-lane"
          },
          {
            "lane_edge_reference": "left",
            "lane_number": 4,
            "lane_status": "closed",
            "lane_type": "middle-lane"
          },
          {
            "lane_edge_reference": "left",
            "lane_number": 5,
            "lane_status": "closed",
            "lane_type": "right-lane"
          }
        ]
      },
      "geometry": {
        "type": "MultiPoint",
        "coordinates": [
          [
            -97.063258,
            32.759701
          ]
        ]
      }
```

Figure 3: TxDOT example (retrieved 12/15/20)


geojson.io/#id=gist:wesleyalforddot/5118702a7667ee3522418154f72ef39f&map=12/42.4402/-71.0558
Open
Save
New
Share
Meta
wesleyalforddot/5118702a7667ee3522418154f72ef39f
JSON
Table
Help
anon | login

```
{
  "road_event_feed_info": {
    "feed_update_date": "2020-09-25T14:52:52.423-04:00",
    "version": "2.0"
  },
  "type": "FeatureCollection",
  "features": [
    {
      "type": "Feature",
      "properties": {
        "road_event_id": "1622-169605",
        "subidentifier": "1622",
        "road_name": "US-1",
        "direction": "southbound",
        "beginning_milepost": 52.03,
        "ending_milepost": 51,
        "beginning_accuracy": "Estimated",
        "ending_accuracy": "Estimated",
        "start_date": "2020-02-01T21:00:19.14-05:00",
        "end_date": "2020-12-31T21:00:19.14-05:00",
        "start_date_accuracy": "Estimated",
        "end_date_accuracy": "Estimated",
        "event_status": "active",
        "total_num_lanes": 3,
        "vehicle_impact": "some-lanes-closed",
        "reduced_speed_limit": 55,
        "description": "",
        "issuing_organization": "Ver-Mac inc.",
        "creation_date": "2020-02-19T16:11:17.293-05:00",
        "update_date": "2020-11-20T15:04:55.39-05:00",
        "types_of_work": [
          {
            "type_name": "surface-work"
          }
        ],
        "lanes": [
          {
            "lane_edge_reference": "left",
            "lane_number": 1,
            "lane_status": "open",
            "lane_type": "left-lane",
            "lane_restrictions": [
              {}
```

Figure 4: MassDOT example (retrieved 12/15/20)

3. Validate adherence to business rules.

a. Current business rules are located here https://github.com/usdot-jpo-ode/wzdx/blob/main/Creating_a_WZDx_Feed.md#business-rules. As you go through the business rules check that existing road events adhere to the rules. Consider documenting examples by using a GeoJSON visualizer and the corresponding code snippet(s) from the WZDx data feed.

NOTE: lane_number has been deprecated in v4.0 and has been replaced with order.

4. Review complex work zones in WZDx feed.

a. Complex work zones often require several linked road events.
b. A single road event can only describe individual changes to each field, so a work zone that has differing number of lane closures, numerous speed restriction zones, etc. must be described by more than one road event.
c. For guidance in constructing complex work zones see draft example work zone implementations here: https://github.com/usdot-jpo-ode/jpo-wzdx/discussions/131.

5. If using DetourRoadEvent Object, validate that it connects to the appropriate road event(s).

a. Check that the relationship is defined for the detour and the connected road events.
    i. road_event_id for the detour is listed in the children array of each road event that detour is routing around
    ii. road_event_id for each road event that the detour is routing around is listed in the parents array of the detour

6. Validate the feed's metadata

a. Validate that the links and contact information are correct and active for the RoadRestrictionFeed Object and WorkzoneRoadEvent Object.

7. Verify that the FeedInfo Object update_frequency is accurate for your system.

a. How often is your system capable of updating? The following are some useful questions to determine this:
    i. Are there manual validation steps before the data feed is updated?
    ii. What is the total time for a change at the work zone to be reflected in the data feed?
    iii. If your system automatically polls data to populate the data feed, how often does this automatic polling occur?

8. Check geospatial coordinates for precision.

a. Make sure coordinates don't have excessive decimal places that would infer more accuracy than your mapping system has. (Most systems support fewer than 5 or 6 decimal places.)