Toronto Water WMS Configuration Schema

Table of contents

Folder: 1-Schemas

Folder: A-entity_record_schema

00_common_definitions.yml

```yaml
---
$schema: "http://json-schema.org/draft-07/schema#"
title: common properties of all entities
$id: https://raw.githubusercontent.com/TW-ASMP/TWmaximoConfig/main/A-entity_record_schema/00_co


definitions:

  #=================================
  # INVENTORY
  #=================================

  stocked_at_TW_def:

    type: boolean
    description: a true value indicates that the item is a stocked item at the Toronto Water

    TW_rule:
      - name: default value of .properties.stocked_at_TW
        spec_ID: EkxMAfT5ee
        specification: |
          For a given item_or_tool_x,
            the value of item_or_tool_x.properties.stocked_at_TW is set to false by default

  rotating_property_def:

    type: boolean

    $comment: |
      For posterity: a rotating item is a trackable item - each instance of the item being a rotating asset record. When the value of this field is true, we must serialize every instance of the item. This commitment is beyond what we can presently achieve at TW. The more pragmatic starting point considered in 2024 is as follows:
        1) for the initial implementation, specify all items as non-rotating
        2) an non-rotating item definition, containing the mfr, model, and ordering_information, can be associated with any asset, through the the asset's item_product_master_record property.
        2) in the future, if we wish to convert the non-rotating item definition to a rotating item definition, we would serialize all the assets associated to the item definition, and convert them to rotating assets.

    rule_spec:
      - name: value of .properties."rotating item"
        spec_ID: 4JKH1tw9gx
        type: [validation, assertion]
        specification: |
          For a given item_x,
            the value of item_x.properties."rotating flag" is set to false, for all time
        $comment:
        status: specified

  manufacturer_and_model_def:

    type: object
    properties:

      manufacturer:
        $ref: MaximoCompanyObject

      product_model:
        $ref: "#/definitions/product_model_def"

  product_model_def:

    type: object
    properties:

      model_and_sub-model:
        oneOf:
        - type: null
        - type: string
        description: For example, "Multilin 869"

      version_or_model_year:
        oneOf:
        - type: null
        - type: string
        description: Identifies the specific version of the product model. For example "v2" or "2023".

      manufacturer_PN:
        oneOf:
        - type: null
        - type: string
        description: The manufacturer designator identifying the exact product item.

  plain-text_manufacturer_and_model_def:
    type: object

    $comment: |
      The following is an example of the plain text manufacturer and model definition
        manufacturer: General Electric
        model_and_sub-model: Multilin 750
```

```yaml
      version_or_model_year: 2024

#==================================
# FAILURE
#==================================

failure_code:

  type: object
  properties:

    code:
      type: string

    name:
      type: string

    description:
      type: string

    failure_code_type:
      type: string
      enum:
        - problem
        - cause
        - remedy

    site:
      type: object
      $ref: MaximoSiteObject

    failure_classes:
      oneOf:
        - type: null
        - type: array
          items:
            type: object
            $ref: MaximoFailureClass

    status:
      type: string
      enum:
        - draft
        - approved

#==================================
# JOB PLAN RESOURCE
#==================================

item_requirement_definition:
#-------------------------
  properties:

    item_reference:
      $ref: "./05_item_master.yml"

    required_quantity:
      type: number

    unit:
      $ref: "#/definitions/unit_of_measure"

tool_requirements_definition:
#-----------------------------
  properties:

    tool_reference: # reference for both stocked and un-stocked tool
      $ref: "./06_tool_master.yml"

    required_quantity:
      type: number

service_requirement_definition:
#-------------------------------
  properties:

    service_reference:
      $ref: "./07_service_item_master.yml"

    required_quantity:
      type: number

    unit:
      type: string
      enum: >
        - hour
        - instance

trade_requirement_definition:
#-------------------------------
  properties:

    trade_type:
      $ref: "../B-entity_class_object_schema/08_trade_type.yml"

    required_quantity:
      type: number

    qualification_requirement:
      oneOf:
      - type: null
      - type: array
        items:
          $ref: "../B-entity_class_object_schema/09_qualification.yml"

#==================================
# UNIT OF MEASURE
#=================================

unit_of_measure:

  description: Represents a unit of measure (UOM) used in inventory management to track quantities of items.

  properties:

    code:

      description: is the unique identifier or code for the unit of measure.
      type: string
      $comment: e.g., "EA", "kg"

    name:

      description: is the full name of the unit of measure.
      type: string
      $comment: e.g., "Each", "kilogram"

#========================
# Record Retirement
#========================

record retirement definition:

  properties:

    record retired:
      type: Boolean

    reason for retirement:
      oneOf: [type: string, type: null]
      updated by system: true
      sort order: 20-20
      $comment: the value should be written by the system, from a asset or record retirement transaction.

frequency_interval_definition:

  properties:

    frequency_quantity:

      type: number


    unit_of_time:

      type: string
      enum:
        - minute
        - hour
        - day
        - month
        - year

#=========================
# Record Duplication
#=========================

duplicate record of:

  oneOf:
    - type: array
      items:
        $ref: "./01_asset.yml"
    - type: null

#=========================

record retirement information:
  $ref: "./00_common_definitions.yml#/definitions/record retirement definition"

#=========================================
#
#=========================================

meter_condition_definition:

  properties:

    meter:

      description: is a selection of a pre-defined meter.
      $ref: default_WMS_meter_object

    numeric_frequency_value:

      oneOf:
        - type: null
        - type: number

    characteristic_trigger_value:

      oneOf:
        - type: null
        - type: string


reference_attachment:

  properties:

    document:
    document_content_class:

      type: string
      enum:
        - maintenance manual
        - asset photograph
        -

#========================================
# Compliance Requirement Definition
#========================================

compliance_requirement:

  properties:

    name:

      type: string
      $comment: |
        For example, 'ANSI Z358.1-2014 on weekly inspection of self-contained emergency wash equipment'.

    requirement_detail:

      description: provides all relevant descriptions
      type: string
      $comment: |
        The following is an example of the requirement detail text for a compliance requirement.

          applicable_asset_class:
            - emergency eye-wash

          requirement_source(s):
            - ANSI Z358.1-2014 / Emergency Eyewash & Shower Standard / 4 Emergency Showers / 4.6 Maintenance and Training
              ...
            - ANSI Z358.1-2014 / Emergency Eyewash & Shower Standard / 4 Emergency Showers / 4.5 Installation
              ...

          source_content_guide:
            - ANSI Z358.1-2014 4.6 states the requirement to check that shower still meets standards
            - ANSI Z358.1-2014 4.5 states the standards to apply for ht check

          perform_every:
            - year

      requirement_compliance_class:

```

```yaml
          description: indicates the level of compliance, with legislative being the top
          $ref: "#/compliance_class"

  compliance_class:

    type: string
    enum:
      - legislative
      - corporate policy


  ############################################
  # Common Work Entity Definition
  ############################################
```

01_asset.yml

```yaml
---
$schema: "http://json-schema.org/draft-07/schema#"
title: asset
$id: https://raw.githubusercontent.com/TW-ASMP/TWmaximoConfig/main/1-Schemas/A-entity_record_sc
type: object

$comment: >

properties:

  ID:

    type: string
    description: A read-only UUID, generated by the system, to uniquely identify the asset record.
    rule_spec:
      - name: vertical asset ID
        spec_ID: 41JeoQuvex
        type: [assertion]
        specification: |
          Upon the creation of a new vertical facility asset record generate a unique ID (such as UUID Ver4)
        checked_on: 2024-08-15
        $comment: |
          UUID has a distinct advantage over a simple serial number - we do not need a script to check for repetition. For instance, when onboarding assets from another system or a spreadsheet, we don't need to check the WMS to see if the ID was already taken.
      - name: linear_asset_ID
        spec_ID: Vku-67dDxx
        type: [assertion]
        specification: |
          Upon the creation of a new asset record corresponding to a record in TWAG, through the Maximo-TWAG integration,
            populate the TWAG_asset record's "Facility ID" value into the "ID".
        checked_on: 2024-08-15
        $comment: see comment for rule 41JeoQuvex.

  name:

    type: string
    description: The human readable short description of the asset.
    $comment: |
        Assumption: an non-is a specific commercial product is always built on site for a specific purpose, and would permanently occupy a role. An example is an aeration tank.
    rule_spec:
      - name: Asset Naming
        spec_ID: 4ykhOm_Dle
        type: assertion
        specification: |
          if asset_x.properties."is_a_commercially_available_product" = TRUE
            asset_x.properties.name is the semi-colon delimited concatenation of:
              - asset_x.properties.class.properties."class name"
              - asset_x.properties."item_product_master_record".properties.product manufacturer company
              - asset_x.properties."item_product_master_record".properties.model_and_sub-model
              - asset_x.properties."item_product_master_record".properties.product configuration code
              - asset_x.properties."OEM_serial"
          elif asset_x.properties."is_a_commercially_available_product" = FALSE
            asset_x.properties.name is the semi-colon delimited concatenation of:
              - asset_x.properties.class.properties."class name"
              - asset_x.properties."assigned_to_role".properties.name
          # NOTE: actual script should contain additional condition handle formatting of the name text when there is missing data in any concatenated property.
        status:
          checked: 2024-08-15

  class:

    $ref: "../B-entity_class_object_schema/01_asset_item_tool_class.yml"
    description: Indicates the primitive class to which this asset is an instance.
    rule_spec:
      - name: exclusion of parts ("only used as a part asset") from asset classification
        spec_ID: V15NNHZuxl
        type: [validation, UI]
        specification: |
          Assertion Part:
            For all assets "asset_x",
              the value of (asset_x.class.properties.only used as a part asset) must be FALSE
```

```
     UI Part:
      In all asset classification search or selection screens, eliminate or filter out all classes "class_y",

       where (class_y.properties.only used as a part asset) is TRUE
    checked_on: 2024-08-15

 class_dependent_specifications:

  type: object
  description: is a set of properties applicable to the class.

 inferred_classes:

  oneOf:
   - type: array
    items:
     type: string
   - type: null
  read-only: true
  integration: true
  description: Indicates the complex classes to which this asset is an instance. A complex class is defined with reference to a primitive class plus other attributes. An example of a complex class is the TSSA high-pressure boiler class, which is made with reference to th primitive class boiler.
  $comment: |
   To implementer: this field will be populated by a rule processor, operating outside of Maximo and with integration to Maximo. An example of the inferred class is "high-pressure boiler".  The values are strings instead of classification objects because the inferred classes will not be in Maximo's asset classification.


## STATE AND STATUS GROUP OF PROPERTIES
 #===================================

 physical_status:

  type: string
  description: Indicates whether the asset is present at the City, and more precisely, at its working location. It also indicates when the knowledge of the asset's presence is missing (i.e., it is missing or lost).
  enum:
   - planned
   - in possession
   - installed
   - abandoned in place
   - removed from possession
   - missing
   - lost
  $comment: |
   This data field is not nullable because the lack of knowledge is explicitly expressed as "missing" or "lost", and the non-existence is expressed as "planned" or "removed from possession".
```

```yaml

  operating_state:

    type: string
    description: Indicates whether the asset is available for doing the work that it is assigned at a given moment. Only applies to asset that is assigned to a role, user-group, or user.
    enum:
      - available (up)
      - unavailable (down)
      - not applicable
    $comment: |
      To data architect and implementer: the "not applicable" value is important, because when we are reporting on equipment uptime, we need to know about the periods in which the operating state is not applicable. For example, if the asset is not assigned to any role, organization, or anyone.

## OWNER, OPERATOR, MAINTAINER GROUP
  #==================================

  owned_by_organization:

    oneOf: [type: null, $ref: "./04_org_or_group.yml"]
    description: Denotes the organization that owns the asset.
    integration: true

    rule_spec:
      - name: valid values of "owned_by_organization" property
        spec_ID: 410N2dr_xx
        type: [validation,UI]
        specification: |
          - The valid range of values for selection includes the first or second of the organizational hierarchy, specified in the (\TWmaximoConfig\3-System_Hierarchies\04_org_hierarchy.md) . For examples,
            - first level example: York Region,
            - second level example: Toronto Water
          - The UI must only present the valid range of values to the users for selection, and the valid range of values must be presented as a hierarchy.
        checked_on: 2024-08-19

  owned_by_another_organization:

    oneOf:
    - type: null
    - type: string
    description: name of an organization that is not found in the current list (and should be added)
    integration: true
    $comment: |
      []To WIM, this data field should be added to the GIS to allow an asset record to be enter into the system, even if the organization that owns it had not been added to the value list.

```

```yaml
 maintenance_group:

  oneOf: [type: null, $ref: "./04_org_or_group.yml"]
  description: group responsible for the overall maintenance of the asset - for example, a unit, work area, or crew.

 operator_group:

  oneOf: [$ref: "./04_org_or_group.yml"]
  description: group responsible for the operation of the asset
  $comment: |
   This property is not strictly needed for the vertical assets - their group can be inferred from their assignment (to role or user) values.  This value is need for TWAG / linear assets.

  rule_spec:
   - name: inheriting the asset's maintenance and operator group values from its role
    spec_ID: VJ1QRgIclg
    specification: |
     - if the value of asset_x.properties.assigned_to_role is role_y, then
       inherit the value of
        - asset_x.properties.maintenance_group
        - asset_x.properties.operator_group
       from the same properties of role_y
    checked_on: 2024-08-20


## ASSIGNMENT GROUP OF PROPERTIES
 #================================

 assignment_type:

  oneOf:
   - type: string
   - type: null
  description:
  enum:
   - to a role
   - to a user group
   - to a single user
   - not assigned
  $comment: |
   This property is added to assist with the interpretation of the null value in the "assigned_to_role", "assigned_to_tool_user_group", or "assigned_to_tool_user_group" property. If the value here is "not assigned", then we know the asset is not assigned to anything. If the value here is null, we do not know whether this asset is assigned to anything.

  rule_spec:
   - name: Rendering of (asset_x.properties."assignment_type") data field.
    spec_ID: 4yARRuv0ex
    type: UI
```

```
        description: |
          the options of this property should be presented as radial button
        status: specified

      - name: Valid Assignment of an Asset
        spec_ID: NyrzGKwuel
        type: [validation, assertion, UI]
        description: |
          If asset_x.properties."assignment_type" = "to a role", then
            - asset_x.properties."assigned_to_role" must NOT = null;
            - asset_x.properties."assigned_to_tool_user" must = null
            - asset_x.properties."assigned_to_tool_user_group" must = null
          elif .properties."assignment_type" = "to a user group", then
            - asset_x.properties."assigned_to_tool_user_group" must NOT = null;
            - asset_x.properties."assigned_to_role" must = null
            - asset_x.properties."assigned_to_tool_user" must = null
          elif asset_x.properties."assignment_type" = "to a single user", then
            - asset_x.properties."assigned_to_tool_user" must NOT = null;
            - asset_x.properties."assigned_to_role" must = null
            - asset_x.properties."assigned_to_tool_user_group" must = null
          elif asset_x.properties."assignment_type" = null, then
            - asset_x.properties."assigned_to_role" must = null
            - asset_x.properties."assigned_to_tool_user" must = null
            - asset_x.properties."assigned_to_tool_user_group" must = null
          Also, in the UI screen, disable the properties that should = null
        status: specified

  assigned_to_role:

    oneOf:
      - $ref: "./02_role.yml"
      - type: null
    description: Role that the asset is designated to play. This value persists even
 ↪  if the asset is temporarily removed from the location of the role (for reasons
 ↪  such as repair).

  assigned_to_tool_user_group:

    oneOf:
      - $ref: "./04_org_or_group.yml"
      - type: null
    description: A group of people, such as a facility, work area, or crew to whom
 ↪  the asset is assigned for use. Indicates the assignment of an asset (usually a
 ↪  tool) that does not have a system role.

  assigned_to_tool_user:

    oneOf:
      - $ref: "./02_role.yml"
      - type: null
    description: Indicates the assignment of an asset (usually a tool) that does not
 ↪  have a system role.

```

```

## LOCATION PROPERTY GROUP
  #========================

  installation_or_parking_location:

    oneOf:
      - $ref: "./03_space.yml"
      - type: null

  service_address_or_coordinate:

    oneOf:
      - $ref: MaximoServiceAddressObject
      - type: null
    $comment: |
      this is referencing Maximo's native service address object
    rule_spec:
        - name: asset present at site must have location information on record
          spec_ID: 01J5R2F9ARJDM3RMGE9WYZWVFE
          type: [validation]
          specification: |
            if the value of asset_x.properties.physical_status is either
              - "in possession", or
              - "installed"
            then at least one of the following properties must NOT be null
              - asset_x.properties.installation_or_parking_location
              - asset_x.properties.service_address_or_coordinate
          check_on: 2024-08-20

  parent_asset:

    oneOf:
      - $ref: "./01_asset.yml"
      - type: null
    description: >
      Indicates the larger discrete asset or defined collection of assets, to which this asset is a part of.  NOTE: this property is not meant to be used for specifying the system hierarchy parent. That property is found on the role record, not the asset record.
    $comment: |
      This field is commonly used when the asset is a part of a skid, structural tank, or switchgear cabinet, in which the asset parent in the system hierarchy should be the line entity. As such we will using this field to track that the asset is also a part of a physical assembly. We would also be using this field to capture a serialized rotating component as a part of another discrete asset. This field can also be used to indicate an asset membership in a Defined Collection of Assets.


## PRODUCT AND TOOL ASSOCIATION GROUP
  #==================================

```

```yaml
  is_a_commercially_available_product:

    type: boolean
    description: An asset is made under as a product of a commercial entity, as opposed to an asset that is assembled on site.
    $comment: No null value allowed because this information is self-evident

    rule_spec:

      - name: Automatic Value Assignment to properties."is_a_commercially_available_product"
        spec_ID: 4Jg2gYSOee
        type: [assertion]
        specification: |
          - Upon record creation, set the value to TRUE.
          - Upon a asset_x.properties.class value change or a re-run of the Maximo rule processor,
              if asset_x.properties.class.properties."non-manufactured" = TRUE;
                set the value to TRUE;
              else set the value to FALSE.
        status: [specified]

  is_a_tool:

    type: boolean
    description: a tool is enables or enhances the ability of a human agent to perform a piece of maintenance, repair, testing, and investigative work. "true" value would designate the asset as a rotating tool, which allows the asset to be 1. reserved for work, or 2. assigned to a staff or group (which includes fixed tools).
    $comment: Note that this property was changed from "mobile" because this designation also applies to fixed tools, such as machine shop or lab tools. All of these assets fall within the definition of a tool.

    rule_spec:

      - name: Value of (.properties."is_a_tool") defaults to false
        spec_ID: 41sz7KSdxe
        type: assertion
        specification: |
          - Upon record creation, set the default value to FALSE.
          - Upon a asset_x.properties.class value change or a re-run of the Maximo rule processor,
              if asset_x.properties.class.properties.tool = TRUE;
                then set the value to TRUE;
        status: [specified, checked]

  is_mobile:

    oneOf:
      - type: boolean
      - type: null

    rule_spec:

      - name: mobiles need to have its serial number on record
        spec_ID: EyA3sYa9le
        type: validation
        specification: |
          For any asset_x,
            if the value of asset_x.properties.is_a_tool is TRUE, and the value of asset_x.properties.is_mobile is also TRUE, then
              the value of asset_x.properties.OEM_serial cannot be null.
        check_on: 2024-08-20

  item_product_master_record:

    oneOf:
      - $ref: "./05_item_master.yml"
      - type: null
    description: This field links the asset to an item record that defines a specific commercial product. By effect, it also deems to asset to be a rotating item.

    rule_spec:

      # - name: If an asset is commercially available but not a tool, then it must have mfr and model information.
      #   spec_ID: VJY43yI9lx
      #   type: [assertion, UI]
      #   specification: |
      #     if asset_x.properties.is_a_commercially_available_product = TRUE AND asset_x.properties.is_a_tool = FALSE, then
      #       - (asset_x.properties."item_product_master_record") is NOT null
      #       - enable (asset_x.properties."item_product_master_record") in UI
      #     else
      #       - (asset_x.properties."item_product_master_record") is null
      #       - disable (asset_x.properties."item_product_master_record") in UI
      #   status: TBS


      - name: valid item master record in .properties.item_product_master_record
        spec_ID: VJGKn1I9ex
        type: validation
        specification: |
          For asset_x.properties.item_product_master_record,
            only accept a master record whose value of asset_x.properties.generic_or_specific_product is "specific commercial product".
        status: [specified, checked]
        $comment: related to 4y3dRfLcee

  tool_product_master_record:

    oneOf:
      - $ref: "./06_tool_master.yml"
```

```
   - type: null
  description: A association with a master record designates the asset as a stocked tool, which allows the tool to be checked into a storeroom and tracked as a part of an inventory. Without an association, the tool would be non-stocked.

  rule_spec:

   - name: when to enable the tool_product_master_record
    spec_ID: NyQBbeL9xl
    specification: |
     if asset_x.properties."is_a_tool" = TRUE
      then enable (asset_x.properties."tool_product_master_record") property.
    status: [specified, checked]


   - name: valid value of asset_x.properties.tool_product_master_record
    spec_ID: NyFFW1Ucll
    type: validation
    specification: |
     only accept a tool_product_master_record whose .properties.generic_or_specific_product property value is "specific commercial product"
    checked_on: 2024-08-20


   - name: an asset may either be associated with a tool or an asset, not both
    spec_ID:
    type:
    specification:
    status: TBS
    checked_on:

  TW_workflow:

   - name: creating a rotating tool directly from an asset record
    specification: TBS []
    status: work in progress


## MANUFACTURER AND MODEL GROUP
 #===============================

 commercial_product_information:
  oneOf: [type: null, $ref:"./00_common_definitions.yml#/definitions/plain-text_manufacturer_and_model_def"]

## DATE PROPERTY GROUP
 #=========================

 construction_contract_number:

```

```yaml
  oneOf:
   - type: string
   - type: null
  description: The construction_contract_number (usually RFQ#) assigned by the City

 first_day_of_City_operation:

  oneOf:
   - type: string
   - type: null
  description: The day that the asset is turned over to the City from a contractor, or if the City installed the asset itself - the day the asset enters operation after testing is completed.
  $comment: |
   This usually coincides with "warranty start date".  However, if the asset is not delivered through a project, "warranty start date" may be empty.

 OEM_serial:

  oneOf:
   - type: string
   - type: null
  description: The serial number, affixed on the asset, designated by the manufacturer.
  $comment: |
   ASMP Discussion Log: The serial number is only populated when an asset experiences a movement (except for movement for removal), or when it is being check into a storeroom. Therefore, when the value of the OEM_serial is null, it represents the fact that we do not know what the serial number is (and whether it has a serial number at all).

 purchase_cost_in_CAD:

  description: the original purchase cost of the asset (not necessary if the asset is associated with a item master record)
  oneOf:
   - type: number
   - type: null

 asset_photos:

  oneOf:
   - type: array # "array" indicates asset may have multiple photos
    items:
     - type: string # photos are converted to a string in JSON
     - oneOf:
      - contentMediaType: image/png
      - contentMediaType: image/jpg
   - type: null


## BACKGROUND PROPERTIES POPULATED AUTOMATICALLY
```

```
  #=============================================

  TW Asset Group:

    oneOf:
      - type: string
      - type: null
    invisible: true
    read-only: true
    enum:
      - Drinking Water Network
      - Drinking Water Treatment Plants
      - Waste and Storm Water Network
      - Wastewater Treatment Plants
      - Independent Building
      - Multiple Major Systems
    $comment: |
      Note that this property is populated automatically, and not available for user to edit. Use-case: asset from the GIS will not be indexed on the hierarchy. The main use of this property is to provide a simple handle term, when one needs to summarize the collection of all assets imported from a certain layer(s) the GIS.

#############################################################
# HIGH LEVEL RULES
#############################################################

rule_spec:

  - name: Asset must have a start of operation date info before we can indicate that it is operationally available.
    spec_ID: NyG2nzL5xg
    type: validation
    specification: |
      if both of the following properties are null
          - (asset_x.properties."first date of City operation")
          - (asset_x.properties."warranty start date")
        then the value of (asset_x.properties."operating_state") CANNOT be "available (up)"


  - name: asset can be assigned exclusively to either a role, user, or user group
    spec_ID: EkD-ZmIceg
    type: validation
    specification: |
      only one of the following properties can have value (i.e., not null) at any given time. (It is also okay for all of them to be null)
          - asset_x.properties."assigned_to_role"
          - asset_x.properties."assigned_to_tool_user"
          - asset_x.properties."assigned_to_tool_user_group"
    errorMessage: "Between \"assigned_to_role\", \"assigned_to_tool_user_group\", \"installed\"assigned_to_tool_user\", every asset may only have one type assignment."

```

```yaml

  - name: consistency between (.properties."operating_state") and assignment values
    spec_ID: 41OFxr8ceg
    type: [validation, assertion]
    specification: |
     if an asset does not have a value in any of the following properties (i.e., all nulls),
       - asset_x.properties."assigned_to_role"
       - asset_x.properties."assigned_to_tool_user"
       - asset_x.properties."assigned_to_tool_user_group"
      then the value of (asset_x.properties."operating_state") must be "not assigned work". The opposite must also be true.
    errorMessage: "An asset NOT assigned to a role, user, or user group should not be operating and therefore would not have an operating_state"


  - name: consistency between asset's physical status and its operating state and assignments
    spec_ID: NyG2nzL5xg
    type: [assertion, validation]
    specification: |
     If the value of (asset_x.properties."physical_status") is NEITHER of the following
       - "installed"
       - "in possession"
      then the following properties would take on the stated values
       asset_x.properties."operating_state" = "not applicable"
       asset_x.properties."assigned_to_role" = null
       asset_x.properties."assigned_to_tool_user" = null
       asset_x.properties."assigned_to_tool_user_group" = null
    errorMessage: "If an asset is not \"installed \\ in possession\" (see the physical_status), then it should not have an operating_state value or any assignment."


  - name: an asset can only be assigned to a discrete asset role
    spec_ID: 4yBXuH8qle
    type: [validation]
    specification: |
     if (asset_x.properties."assigned_to_role") is NOT null
       then (asset_x.properties."assigned_to_role".properties.class.properties."discrete asset role") = TRUE
    errorMessage: an asset can only be assigned to a discrete asset role


  - name: inheriting the asset location information from its role
    spec_ID: NJdGTHLqeg
    type: [assertion]
    specification: |
      For an asset, asset_x, if
       all of the following are true:
        - asset_x.properties."operating_state" = "installed"
```

```
            - asset_x.properties."assigned_to_role" is NOT null
          and one of the following is true
            - asset_x.properties."assigned_to_role".properties."asset installation location" is NOT null
            - asset_x.properties."assigned_to_role".properties."service_address_or_coordinate" is NOT null
        then
            (asset_x.properties."installation_or_parking_location") would be set to the value of
              (asset_x.properties."assigned_to_role".properties."asset installation location")
            (asset_x.properties."service_address_or_coordinate") would be set to the value of
              (asset_x.properties."assigned_to_role".properties."service_address_or_coordinate")


  - name: over-write of manufacturer and model information
    spec_ID: 01J5RPPEKJCP11NBRW3A4XCKF7
    specification: |
      if the value of either
    status: TBS


  - name: a commercially available asset must be associated with manufacturer and model information
    spec_ID: 4y3dRfLcee
    type: validation
    exempt_grandfather: true
    specification: |
      if the value of asset_x.properties.is_a_commercially_available_product is TRUE
        then NONE of the following properties can be null
          - asset_x.properties.item_product_master_record
          - asset_x.properties.manufacturer_name
          - asset_x.properties.product_model_information
```

02_role.yml

```
---
$schema: "http://json-schema.org/draft-07/schema#"
title: role
$id: https://raw.githubusercontent.com/TW-ASMP/TWmaximoConfig/main/1-Schemas/A-entity_record_sc
type: object

properties:

  ID:

    type: string
    description: Also known as the "tag number" or "entity number" in Avantis's vocabulary. (Avantis is the a WMS).
    #[] to do ID for pumping stations an chambers will be from the GIS.
    rule_spec:
      - name: ID of Linear Assets Represented as Role in Maximo
        spec_ID: Vku-67dDxx
        type: triggered action
        specification:
          trigger: replication creation of assetY record from the TWAG
          action: apply Facility ID from TWAG as ID
        status: false

  name:

    type: string
    description: A structured and ideally unique description of the role.
    const: > #[]
      ${properties.class.properties."class name"};
      childOf: ${properties."parent entity".properties."role number"};
      serving: ${properties."client role served".properties."role number"};
    # MT []: would you like me to move this into individual roles?

  parent:

    $ref: "./02_role.yml"
    description: References the role that is served by the larger asset, which physically subsumes the asset serving this role.

  class:

    $ref: B-entity_class_object_schema/02_role_class.yml
    description: The class denote the broad types of asset that may play the role (e.g., "breaker role"), and sometimes, more specifically, it also denotes the useful function provided by that asset in the role to the larger system (e.g., the "tie-breaker role" provides tie-breaking function to the electrical distribution system).

  class_dependent_specifications:

    type: object
    description: is a set of properties applicable to the class.

  inferred_classes:

    oneOf:
      - type: array
        items:
          type: string
      - type: null
    items:
      type: string
    read-only: true
    description: See the description for inferred classe names(s) made in the \TWmaximoConfig\1-Schemas\A-entity_record_schema/01_asset_schema.yml
    $comment: |
      See the comment for inferred classe names(s) made in the
↪   \TWmaximoConfig\1-Schemas\A-entity_record_schema/01_asset_schema.yml

  role_status:

    type: string
    enum:
      - specified
      - active
      - eliminated

    description: |
      This is the basic life-cycle status of a role. Specified means the role is
↪   conceived and exists in some specification or design documentation. Active means
↪   the necessary supports exist for an asset to serve in the role and function of
↪   the asset being utilized. Eliminated represents a negation of either or both
↪   conditions of the active status.

    rule_spec: |
      #[]RULE VkiDyJcSxg: Before a role can be eliminated, all children, as well as
↪   the descendants of the role in the hierarchy must also be eliminated.
        $comment: a procedure should to be created to allow the recursive
↪   elimination of a role and all of its children.
      #[]RULE VygDCOFrxl: When a role is "eliminated", it must no longer be visible
↪   in any view of the role hierarchy. (Its entire branch must not be available
↪   either, because all of its descendants would be eliminated as well.)
        []Review with To ASMP: with this rule, we no longer need to have a hierarchy
↪   branched for retired roles.
      #[]RULE EkP5qy5Sxl: If change auditing cannot be turned on, then when a role
↪   record status is "eliminated", all the specification in the record's datafields
↪   must be frozen.

  role technical requirement description:

    type: string
    description: A free text description of the role's performance requirements that
↪   must be satisfied by the asset. For example, the lifting capacity in kilograms
↪   for a crane.
    $comment: |
      Eventually, this information should be replaced by formalized properties
↪   associated with specific classes. For example, for the motor role class, there
↪   would be a requirement for a minimum amount of horsepower or torque.

  serving_asset_in_role:

    $ref: "./02_role.yml"
    description: Identifies the asset this role is serving. For example, given a
↪   motor starter role, the value in this data field identifies the role of the
↪   motor controlled by that motor starter.

  asset installation location:
    oneOf:
```

```yaml
    - $ref: "./03_space.yml"
    - type: null
  description: References the space in which the asset serving the role would be installed.

service_address_or_coordinate:
  oneOf: [$ref: MaximoServiceAddressObject, type: null]
  description: A geo-coordinate or the nearest street address of the asset.

#=========================

operator_group:

inherit operator group value:
  type: boolean
  default value: true

#=========================

inherit_operator_group_from_parent:

  type: boolean
  default_value: true

#=========================

maintenance_group:

  oneOf: [$ref: "./04_org_or_group.yml", type: null]
  description: Group responsible for the preventive and reactive maintenance of the asset in the role. For example, a unit, work area, or crew.

#=========================

inherit_maintenance_group_from_parent:

  type: boolean
  default_value: true

#=========================

operational criticality:

  oneOf: [$ref:'#/definitions/criticality rating definition', type: null] # see section 3. LOCAL OBJECT DEFINITION
  description: A role bears high operational criticality if the loss of the asset in the role will either reduce throughput or product quality (but not product safety) of the larger system.

#=========================

protective function criticality:

    oneOf: [$ref:'#/definitions/criticalityRatingDef', type: null] # see section 3. LOCAL OBJECT DEFINITION
    description: A role bears protective function criticality if the loss of one of its protective functions (i.e., regulatory/control/protection or containment function) will either result in a consequential release of hazard or the loss of a capability to mitigate a greater level hazard.

  #========================

  duplicate record of:
    oneOf:
      - type: array
        items:
          $ref: "./02_role.yml"
      - type: null

  #========================

  record retirement information:
    $ref: "./00_common_definitions.yml#/definitions/record retirement definition"


  #========================
  # [] REMOVE THE FOLLOWING AFTER RULE ABOUT ID IS REVISED
  # GIS object ID:
  #   type: string
  #   description: This is the ID of the equivalent object (asset or role) in GIS. This value suggests the original record was created in GIS and copied to WMS through the data integration link.



##############################################################
# 2. DATA INTEGRITY RULES
##############################################################
  allOf:
    - if:
      oneOf:
        - roleClass:
            properties:
                className:
                  const: pumping station
        - roleClass:
            properties:
                className:
                  const: water treatment plant
        - roleClass:
            properties:
                className:
                  const: large chamber
      then:
        required: GIS object ID

  #[]RULE:
  # IN COMMON LANGUAGE: At any given time, each role may only have a single asset assigned to it (i.e., associated with the role via the asset's "assigned_to_role" property).

#############################################################
# 3. LOCAL OBJECT DEFINITION
#############################################################

definitions:
  criticality rating definition:
    type: object
    properties:
      rating:
        type: integer
      description:
        type: string
    enum:
      - rating: 1
        description: TBD
      - rating: 2
        description: TBD
      - rating: 3
        description: TBD
      - rating: 4
        description: TBD
      - rating: 5
        description: TBD

$comment: |
  OPEN AND TODO ITEMS:
    [x] To have discussion on what we put in for criticality
    [] explicitly specify an unoccupied role
    [] expression of role equivalence
    [] RULE: certain GIS assets, such as pumping stations, or treatment facilities must be mapped over as roles
```

03_space.yml

```yaml
---
$schema: "http://json-schema.org/draft-07/schema#"
title: space
$id: https://raw.githubusercontent.com/TW-ASMP/TWmaximoConfig/main/1-Schemas/A-entity_record_sc
type: object

properties:

  ID:

    type: string
    description: An unique ID
    $comment: In the future, this value should be validated with a regular expression.

  parent:

    $ref: "./03_space.yml"
    description: The larger space that completely contains this space.

  name:

    type: string
    description: Short name. Should be one that is commonly use by staff in communication. For example, "boardroom"

  complete_name:

    type: string
    read-only: true
    rule_spec:
      - spec_ID:
        name: complete name generation
        id: 01JFVNS0YDFD7K5DP4NVMSKTY8
        status:
    $comment: |
      Automatically generated by the system and not editable. The value is name to that of its parent, its grandparent ... all the way up that facility space.


  enclosed_by_asset:

    oneOf: [$ref: "./01_asset.yml", type: null]
    description: indicates that the space is what is enclosed by (and immediately surrounding) the asset, such as a
      - building,
      - structural tank,
      - equipment cabinet,
      - vehicle

  class:

    $ref: spaceClassObject
    $comment: |
      see space_classification.md file

  class_dependent_specifications:

    type: object
    description: is a set of properties applicable to the class.

  inferred_classes:

    oneOf:
      - type: array
        items:
          type: string
      - type: null
    read-only: TRUE
    $comment: |
      See the comment for inferred class name(s) made in the \TWmaximoConfig\1-Schemas\A-entity_record_schema/01_asset_schema.yml

      #[]REQ 41Vru1Rrxe: This data field should be visible to the users, but should not be editable by the user

  service_address_or_coordinate:
    oneOf: [$ref: .IBM_Maximo_object/MaximoServiceAddressObject, type: null]
    $comment: this is referencing Maximo's native service address object

  status:

    type: string
    enum:
      - specified
      - realized
      - eliminated
    $comment: |
      This field allows the user to specify whether the space is merely specified, or whether the boundary enclosing the space have been contructed (or alternatively, the fiat property / or area boundary around is formally established and approved.) - i.e. "exists".
      Note that an "eliminated" space should be removed from the hierarchy, and should not be visible for users conducting maintenance, reliability, planning and scheduling functions.
      The term "eliminated" is chosen to indicate that space disappear by the fact that object bound or defined the space, such as wall, ceilings, or property lines are removed.

  confined_space:

    type: boolean

  inherit_hazardous_property_values:

    type: boolean
    default_value: true

  hazardous location class:

    type: string
    enum:
      - I
      - II
      - III

  hazardous location division:
```

```yaml

    type: string
    enum:
      - 1
      - 2

  hazardous location group:

    type: string
    enum:
      - A
      - B
      - C
      - D
      - E
      - F
      - G


  duplicate record of:

    oneOf:
      - type: array
        items:
          $ref: "./03_space.yml"
      - type: null

  record retirement information:
    $ref: "./00_common_definitions.yml#/definitions/record retirement definition"

```

04_org_objects_definitions.yml

```yaml
---
$schema: "http://json-schema.org/draft-07/schema#"
title: organization
$id:
 ↪ https://raw.githubusercontent.com/TW-ASMP/TWmaximoConfig/main/1-Schemas/A-entity_record_sc
type: object
properties:

  organization_or_group_name:

    type: string

  parent_organization:

    oneOf:
      - $ref: "./04_org_or_group.yml"
      - type: null

  class:

    $ref: B-entity_class_object_schema/04_org_class.yml

  class_dependent_specifications:

    type: object
    description: is a set of properties applicable to the class.

  leader:

    oneOf:
      - $ref: "./07_person.yml"
      - type: null
    description:

  equivalent_to_Maximo_site:

    description: indicates that this organization maps to a particular site (a native Maximo object)
    $ref: MaximoSiteObject

  equivalent_to_Maximo_org:

    description: indicates that this organization maps to a particular organization (a native Maximo object)
    $ref: MaximoOrgObject

  duplicate record of:

    oneOf:
      - type: array
        items:
          $ref: "./04_org_or_group.yml"
      - type: null

  record retirement information:

    $ref: "./00_common_definitions.yml#/definitions/record retirement definition"

supervisor_group:

crew:


```

05_item_master.yml

```yaml
---
$schema: "http://json-schema.org/draft-07/schema#"
title: item master
```

```yaml
$id: https://raw.githubusercontent.com/TW-ASMP/TWmaximoConfig/main/1-Schemas/A-entity_record_sc
type: object

properties:

  number:

    type: string
    description:
      A read-only UUID, generated by the system, to uniquely identify the item.
    $comment: |
      UUID instead of a simple serial used with the consideration that we may in the future incorporate items defined outside of TW.

  #=======================

  name:

    type: string
    description: The human readable short description of the item.

    rule_spec:

      - name: item master record naming
        form: long
        spec_ID: VkYgCtRPlx
        type: assertion
        specification: |
          if (item_x.properties.commodity_or_commercial_product) = "commodity", then
            the value of item_x.properties.name would be the semi-colon delimited concatenation of the following property values:
              - properties.class.properties.name
              - every non-empty class dependent specification values
              - properties.supplementary_commodity_description
          elif: (item_x.properties.commodity_or_commercial_product = "commercial product"), then:
            the value of item_x.properties.name would be the semi-colon delimited concatenation of the following property values:
              - properties.class.properties.name
              - properties.product manufacturer company.properties."company name"
              - properties.model_and_sub-model
              - properties.version_or_model_year
              - properties.product configuration code
        checked on: 2024-08-15

  #=======================

  class:

    $ref: "../B-entity_class_object_schema/01_asset_item_tool_class.yml"
    $comment: is a value from the item classification, which is a superset of the asset class.

  rule_spec:

   - name: item classification list includes all classes
    spec_ID: EynXVZ-dxg
    specification: |
     Maximo item classification list would include all class instances of B-entity_class_object_schema/01_asset_item_tool_class_object_schema.yml, regardless of the values
      - properties.tool
      - properties.only used as a part asset
    status: specified
    checked_on: 2024-08-15


 class_dependent_specifications:

  type: object
  description: is a set of properties applicable to the class value.

# INVENTORY MANAGEMENT FLAGS
 #========================

 stocked_at_TW:

  $ref: "./00_common_definitions.yml#/definitions/stocked_at_TW_def"

 rotating:

  $ref: "./00_common_definitions.yml#/definitions/rotating_property_def"

 #====================================
 # generic and specific definition
 #====================================

 commodity_or_commercial_product:

  type: string
  description: indicate whether the item master defines an unspecialized commodity or a specific commercial product.
  enum:
   - commodity
   - commercial product

 supplementary_commodity_description:

  type: string
  description: supplementary description, in addition to the class value and and class-dependent specification values, necessary to differentiate a commodity.

 #====================================
 # MANUFACTURER AND MODEL GROUP
 #====================================
```

```yaml

commercial_product:

  oneOf:
    - type: null
    - $ref: "./00_common_definitions.yml#/definitions/manufacturer_and_model_def"

commercial_product_description:

  oneOf:
  - type: null
  - type: string
    $ref: "./00_common_definitions.yml#/definitions/plain-text_manufacturer_and_model_def"

#=========================

instant_of_commodities:

  oneOf:
    - type: null
    - $ref: "./05_item_master.yml"

  rule_spec:
    - description: range of the instant_of_commodities property must be  items whose TW_defined_commodity value is 'true'.
      form: short
      spec_ID: 01JF33GVB2DT7K5FJ3SA3P9EP4

#=========================

same_item_as:

  oneOf:
    - type: array
      items:
        $ref: "./04_item_master.yml"
    - type: null
  description: |
    Identifies the same commercial product item made by the same manufacturer, but differing only in item unit format. For example, the 208-litre drum item and the 5-litre bottle item of Penzoil 5W30 Synthetic Lubricant.

  rule_spec:

    - name: Infer Symmetrical Equivalence between Product Items
      spec_ID: EJCnUukuex
      type: inference
      specification: ""
      status: []

    - name: Range must be a commercial product as well
      spec_ID: 01JF81079K178X9B4NSG23AA0Z
```

```yaml
        type: validation
        specification:
        status: []

  #=========================

  ordering_information:

    oneOf:
      - type: array
        items:
          $ref: "#/definitions/vendor_order_detail"
      - type: null

  #=========================

  issue_unit:

    type: string
    description: A description of each individual unit issued for work, such as a can or roll, of the product item, regardless of the ordering unit, such as a box of 24.  For example, "80 ml can" or "27 x 500in. roll".
    $comment: "To implementer: use the out of the box list from Maximo"

#=========================
# LOCAL OBJECT DEFINITIONS
#=========================

definitions:

  vendor_order_detail:

    properties:

      vendor:

        $ref: MaximoCompanyObject

      #=========================

      order_unit:

        oneOf:
          - type: null
          - type: string
        description: A description of each individual unit of order, such as a can or roll, of the product item, regardless of the ordering unit, such as a box of 24.  For example, "80 ml can" or "27 x 500in. roll".
        $comment: |
          For a commercial product, if there are multiple item unit formats, one item should be create for each format.
          To Implementer,
            Use Maximo's defaults

      #========================

      vendor_item_number:

        oneOf:
          - type: string
          - type: null

      #========================

      unit_cost_in_CAD:

        type: number
        description: The expected cost of a unit of the item. (This should not be
    ↪ the cost of a package of multiple units of the item.)
        $comment: To TW, in the future, this field should contain a running average
    ↪ of the recent purchase costs, possibly also adjusted for recent inflation.


  # 6/25: [] we need to add another field for lead time. This will contribute to the
    ↪ dynamic calculation of criticality among other uses.
```

06_tool_master.yml

```
---
$schema: "http://json-schema.org/draft-07/schema#"
title: tool item master
$id:
    ↪ https://raw.githubusercontent.com/TW-ASMP/TWmaximoConfig/main/1-Schemas/A-entity_record_sc
type: object

properties:

  number:

    type: string
    description:
      A read-only UUID, generated by the system, to uniquely identify the tool.
    $comment: |
      UUID instead of a simple serial used with the consideration that we may in the
    ↪ future incorporate items defined outside of TW.

  #========================

  name:

    type: string
    description: The human readable short description of the tool.
    rule_spec:
      - name: tool master name
    spec_ID: VJpSzGxdxg
    type: implication
    specification: |
     if: toolX.properties."tool master type" = "generic tool"
     then:
      toolX.properties."tool name" value is the semi-colon ("; ") delimited concatenation of the following property values:
       - properties.class.properties."class name"
       - properties."generic tool application definition"
     elif: toolX.properties."tool type" = "specific commercial product"
     then:
      toolX.properties."tool name" value is the semi-colon ("; ") delimited concatenation of the following property values:
       - properties."tool master class".properties."class name"
       - properties.product manufacturer company.properties."company name"
       - properties.model_and_sub-model
       - properties.version_or_model_year
       - properties.product configuration code
    status: specified

 #========================

 class:

  $ref: "../B-entity_class_object_schema/01_asset_item_tool_class.yml"
  description: This is a value from the classification, which is a superset of the asset class.

  rule_spec:
   - name: Tool classification list does not include parts non tools
    spec_ID: V1ulHHWOgx
    specification: |
     Tool classification list include all class instances of B-entity_class_object_schema/01_asset_item_tool_class_object_schema.yml, except ones whose .properties.tool value is FALSE
    status: specified

 class_dependent_specifications:

  type: object
  description: is a set of properties applicable to the class.


# INVENTORY MANAGEMENT FLAGS
 #========================

 rotating:

  $ref: "./00_common_definitions.yml#/definitions/rotating_property_def"

 #========================

 mobile:

    type: boolean
    description: An tool that is used beyond a permanent installation; instead, it is taken from place to place.
    $comment: |
      #PROCESS: SET DEFAULT VALUE:
        At record creation, set value to false.
      #PROCESS: EVENT-DRIVEN VALUE CHANGE:
        Upon the event of a properties.class value change;
          if properties.class.properties."mobile" = true;
          then set the value to true;
          else set the value to false.



$comment: ASMP does not expect the tool item master to be widely used during the initial adoption of Maximo - we expect that most tools would initially be represented as un-stocked.
```

07_service_item_master.yml

```

```

08_person.yml

```
$schema: http://json-schema.org/draft-07/schema#
title: Person
$id: https://raw.githubusercontent.com/TW-ASMP/TWmaximoConfig/main/1-Schemas/A-entity_record_sc
type: object

properties:

  ID:

    type: string
    description: Unique identifier for the person

  first_name:

    type: string
    description: First name of the person

  last_name:

    type: string
    description: Last name of the person

  display_name:

    type: string
    description: Full name displayed (usually a combination of first and last names)

  division:

    type: string
    description: indicates which City division that the person works for

  unit:

    type: string
    description: indicates which business-unit that the person works for

  primary_trade:

    type: string
    description: Primary craft or skill associated with the person

  external_contractor:

    type: boolean
    description: indicates whether the person is not an employee of the City

  status:

    type: string
    enum:
      - ACTIVE
      - INACTIVE
    description: Status of the person

  email_address:

    type: string
    format: email
    description: Email address of the person

  phone:

    type: string
    description: Contact phone number of the person
```

09_qualification.yml

```
$schema: http://json-schema.org/draft-07/schema#
title: Qualification
$id:
 ↪  https://raw.githubusercontent.com/TW-ASMP/TWmaximoConfig/main/1-Schemas/A-entity_record_sc
type: object
```

```yaml

properties:

  ID:
    description: is a read-only, unique, and permanent ID, generated by the system.
    type: string
    read-only: TRUE

  #========================

  name:

    description: a short name given by the creator of the record.
    type: string

  #========================

  class:

    type: string
    enum:
    - professional license
    - industry certification
    - internal certification

  #========================

  issued_by:

    oneOf:
      - type: null
      - $ref: "./04_org_or_group.yml"

```

10_warranty.yml

```yaml
$schema: http://json-schema.org/draft-07/schema#
title: Warranty
$id:
 ↪  https://raw.githubusercontent.com/TW-ASMP/TWmaximoConfig/main/1-Schemas/A-entity_record_sc
type: object

properties:

  #========================

  ID:

    description: is the unique identifier for the warranty contract
    type: string

```

```yaml
#========================

description:

  description: is a brief summary of the warranty contract
  type: string

#========================

vendor:

  description: identifies the vendor or provider of the warranty
  $ref: MaximoCompanyObject

#========================

warranty_start_date:

  description: is the first day that the warranty becomes effective
  oneOf:
    - type: null
    - type: number

#========================

warranty_expiration_date:

  description: is the last effective date of the warranty
  oneOf:
    - type: null
    - type: number

#========================

meter:

  description: if applicable, is the meter reading (e.g., mileage) at which the warranty starts or ends
  oneOf:
    - type: number
    - type: null

#========================

covers_labour:

  description: indicates that the vendor is responsible for providing and covering the cost of labour
  type: boolean

#========================

covers_parts:

  description: indicates that the vendor is responsible for providing and covering
↪ the cost of parts
  type: boolean

#========================

specific_terms:

  description: are the terms and conditions related to the warranty coverage
  type: string

#========================

covers_roles:

  description: is the list of roles, more specifically, the assets installed in
↪ the roles that are covered by the warranty
  oneOf:
    - type: null
    - type: array
      items:
        $ref: "./02_role.yml"

#========================

covers_assets:

  description: is the list of assets that are covered by the warranty
  oneOf:
    - type: null
    - type: array
      items:
        $ref: "./01_asset.yml"
  $comment: note that any item converted by the warranty should be expressed as a
↪ rotating item (i.e. represented as an asset as well)

```

32_job_plan.yml

```
---
$schema: http://json-schema.org/draft-07/schema#
title: Job Plan
$id:
  ↪ https://raw.githubusercontent.com/TW-ASMP/TWmaximoConfig/main/1-Schemas/A-entity_record_sc
description: |
  A job plan is the the lowest level of work description to contain the full
  ↪ planning specifications.
type: object

properties:
```

```yaml

ID:

  description: is a read-only, unique, and permanent ID, generated by the system, to identify the job plan record.
  type: string

#================

name:

  description: is a description of the activity specified in the job plan.
  type: string

#================

discrete_activity_classification:

  description: indicates the type of activity that specified in the job plan
  $ref: "../B-entity_class_object_schema/32_discrete_activity_class.yml"
  $comment: this could also be called the activity classification


#================================
# Job Plan Applicability Notes
#================================

specific_to_asset_classes:

  description: identifies the asset classes on which the activity specified in this job plan can be done.
  type:
    oneOf:
    - type: null
    - type: array
      items:
        $ref: "../B-entity_class_object_schema/01_asset_item_tool_class.yml"

#================

specific_to_role_classes:

  description: identifies the role classes - more specifically, the assets installed in these roles - for which this job plan is customized.
  type:
    oneOf:
    - type: null
    - type: array
      items:
        $ref: "../B-entity_class_object_schema/02_role_class.yml"
  $comment: >
    Examples of a role class include the tie-breaker and effluent turbidity meter role class.
```

```

#================

specific_to_operational_units:

  description: identifies the Toronto Water site(s), defined as a organization in this schema, for which the job plan is specifically customized.
  type:
    oneOf:
    - type: null
    - type: array
      items:
        $ref: "../A-entity_record_schema/04_org_class.yml"
  $comment: >
    Examples of operational group include
      TAB - Ashbridges Bay Wastewater Treatment Plants
      COL - Waste and Storm Water Collection
  rule_spec: >
    - name: job plan's specific_to_operational_units property must be an unit level organization
      spec_ID: 01JD2V5X97J1Y45JWDW4SV1FJ4
      type: [validation]
      specification: |
        Given an job plan, JP_x, all values of JP_x.specific_to_operational_units must be
          - an organization, and
          - whose .class = unit (a subclass of Group in the City)

#================

specific_to_roles:

  description: identifies the asset roles for which the job plan is specifically customized.
  type:
    oneOf:
    - type: null
    - type: array
      items:
        $ref: "../A-entity_record_schema/02_role.yml"

#================

specific_to_commercial_products:

  description: identifies the the commercial products for which the job plan is specifically written/customized.
  type:
    oneOf:
    - type: null
    - type: array
      items:
        $ref: "../A-entity_record_schema/05_item_master.yml"
  rule_spec: >
    - name: Valid Commercial Product Item reference in a Job Plan
      spec_ID: 01JD2V5X97J1Y45JWDW4SV1FJ4
      type: [validation]
      specification: |
         Given an job plan, JP_x, all values of JP_x.specific_to_commercial_products must be
          - a item master record, and
          - whose .generic_or_specific_product = "specific commercial product"

#================================
# Record Provenance
#================================

is_derived_from:

  description: identifies the job plan from which the present job plan specification was based on.
  oneOf:
    - type: null
    - $ref: "./32_job_plan.yml"
  integration: true

#================

failure_codes:

    description: denotes a physical-based failure condition (e.g., shaft misalignment).
    oneOf:
      - type: null
      - type: array
        items:
          $ref: "./00_common_definitions.yml#/definitions/failure_code"

#================

RCM_failure_modes:

  description: identifies the functional failure mode ID, with respect to a specific asset role, mitigated by the work specified in this job plan.
  oneOf:
    - type: null
    - type: array
      items:
        type: string
  integration: true
  $comment: >
    - For the 2026-27 implementation, this data field will start-out as a free-text. In the future, the value will come from a solution such as OnePM.

#================================
# Procedure
```

```yaml
#================================

work_description:

  description: is a single body of text outlining the sequential steps to complete the activity
  type: string
  $comment: >
    Example:
      1) Ensure you have operational approval before performing this task.
      2) Follow Lock-out and Tag-out process before starting this task.
      3) Drain the oil from the gearbox.
      4) Install 25 Litres of UCON 220 ( food grade) oil.
      5) Remove Lock-out and Tag-out and check operation.
      6) Inform operations that the task as assigned is completed.

# ==========

requires_shut_down:

  description:

#========================================
# Resource Requirements
#========================================

estimated_duration:

  description: is the estimated time to complete the activity in the job plan
  $ref: "./00_common_definitions.yml#/definitions/frequency_interval_definition"

#================

parts_or_material_requirements:

  description: identifies the parts and material required to complete the work.
  oneOf:
    - type: null
    - type: array
      items:
        $ref: "./00_common_definitions.yml#/definitions/item_requirement_definition"

#================

tool_requirements:

  description: identifies the tools required to complete the work.
  oneOf:
    - type: null
    - type: array
      items:
        $ref: "./00_common_definitions.yml#/definitions/tool_requirements_definition"

  #================

  skill_and_trade_requirements:

    description: identifies the trades and qualifications of each trade needed to complete the work.
    oneOf:
    - type: null
    - type: array
      items:
        $ref: "./00_common_definitions.yml#/definitions trade_requirement_definition"

  #================

  service_requirements:

    description: identifies (contracted) service needed to complete the work.
    oneOf:
      - type: null
      - type: array
        items:
          $ref: "./00_common_definitions.yml#/definitions/service_requirement_definition"

#=========================================
# Related Activities
#=========================================

  must_be_preceded_by:

    description: identifies activities (specified in other job plans) that must be performed in the same work order before the activity specified in this PM can be performed.
    oneOf:
    - type: null
    - type: array
      items:
        $ref: "./32_job_plan.yml"

  must_be_followed_by:

    description: identifies activities (specified in other job plans) that must be performed in the same work order after the activity specified in this PM can be performed.
    oneOf:
    - type: null
    - type: array
      items:
        $ref: "./32_job_plan.yml"

#=========================================
# Work Trigger Condition Notes
#========================================

  time-based_frequency:

    oneOf:
      - type: null
      - $ref: "./00_common_definitions.yml#/definitions/frequency_interval_definition"

  #================

  meter-based_frequency:

    oneOf:
      - type: null
      - $ref: "./00_common_definitions.yml#/definitions/meter_condition_definition"

  #================

  description_of_event-based_trigger:

    oneOf:
      - type: null
      - type: object
        properties:

          relation_to_event:

            type: string
            enum:
              - before
              - during
              - after
              - at the start of
              - at the end of

          description_of_event:

            description: a free-text description of a event or process, such as "an elevator failure".
            type: string

  #================

  notes_on_trigger_condition:

    description: free-text description on the additional trigger conditions
    oneOf:
      - type: null
      - type: string

#========================================
```

```yaml
# Compliance Information
#=========================================

  compliance_requirement:

    description: identifies the compliance requirement object
    integration: true
    oneOf:
      - type: null
      - type: array
        items:
          $ref: "./00_common_definitions.yml#/definitions/compliance_requirement"

  #=================

  compliance_class:

    description: indicates the level of compliance, with legislative being the top class
    $ref: "./00_common_definitions.yml#/definitions/compliance_class"

  #=====================

  mitigates_safety_risk_to_staff:

    oneOf:
    - type: null
    - type: string
    enum:
      - yes
      - no
      - unspecified

  #=====================

  mitigates_safety_or_health_risk_to_public:

    oneOf:
    - type: null
    - type: string
    enum:
      - yes
      - no
      - unspecified

  #=====================

  mitigates_environmental_risk:

    oneOf:
    - type: null
    - type: string
    enum:
      - yes
      - no
      - unspecified

  #=====================

  safe_work_plan_link:

    description: identifies a safety work plan by a permanent URL to the document (e.g., corporate Safe Procedure or Toronto Water Safe Operating Procedures)
    oneOf:
      - type: null
      - type: array
        items:  #URL strings
          type: string

#=========================================
# Require Information Data Submission Group
#=========================================
```

33_PM.yml

```yaml
---
$schema: http://json-schema.org/draft-07/schema#
title: PM
$id: https://raw.githubusercontent.com/TW-ASMP/TWmaximoConfig/main/1-Schemas/A-entity_record_sc
type: object

properties:

  ID:

    description: is a read-only, unique, and permanent ID, generated by the system, to identify PM.
    type: string
    $comment: This ID is useful for referencing, even when its name changes.

  # ==========

  name:

    description: a short name for the PM, given by the creator of the PM.
    type: string

  # ==========

  operational_unit:

    description: indicates Toronto Water's operational unit, on the organizational hierarchy, such as Waste and Storm Water Pumping (symbol - WASP) or Humber Wastewater Treatment Plant (symbol - THR).
    $ref: "./04_org_or_group.yml"

  # ==========

  supervisor_group:

    description: indicates a sub-group of the operational unit, that is led by a supervisor who is accountable for the performance of the specified work.
    $ref: "./04_org_or_group.yml"

  # ==========

  crew_assignment:

    description: identifies a crew, under the supervisor group, that is always assigned to perform the specified work

    oneOf:
      - type: null
      - $ref: "./04_org_or_group.yml"

    comment: This value will be determined by (and must be consistent with) the maintainer_organization value - situated at a lower level of the organization hierarchy

  # ==========

  member_of_PM_set:

    description: indicates that this PM is a member of a set of related PMs. For example, the PMs for raw water pump 1, 2, 3 are all members of a PM set named Raw Water 5-year Disassembly Maintenance.

    oneOf:
      - type: null
      - $ref: "#/definitions/PM_set"
    rule to add []: only applicable to higher-level PM

  # ==========

  Avantis_PM:

    description: indicate the Avantis PM (the legacy WMS) that this Maximo PM (equivalent to an Avantis PM task) was a part of.

    oneOf:
      - type: null
      - type: string
    read-only: TRUE

    comment: This field can be eliminated in the future.

    work_entity_harmonization: WR(x), WO(x), JP(x)
```

```yaml

# ==========

processes_covered_by_PM:

  description: a list of all major process systems covered by the work specified in this PM.
  read-only: TRUE
  oneOf:
    - type: null
    - type: object
      properties:

        ranking:
          type: number

        system_naming:
          type: string

  rule to add []: only applicable to higher-level PM



# Work Specification at a High-level
#=========================================

role_to_work_on:

  description: indicates role at which the specified must be performed.
  oneOf:
    - type: null
    - $ref: "../A-entity_record_schema/02_role.yml"

#=================

asset:

  description: indicates the asset that is being maintained.
  oneOf:
    - type: null
    - $ref: "../A-entity_record_schema/01_asset.yml"

# ==========

job_plan:

  description: specifies the job plan for the PM, if there is no further specification within the route.


# PM Structure Specification
#=========================================

parent:

  description: indicates the more comprehensive PM, usually a shut-down PM, that this PM is a part of.

  oneOf:
    - type: null
    - $ref: "../A-entity_record_schema/33_PM.yml"

  $comment: PMs should be organized into a PM-set when they are meant to be performed at different times.  For example the PMs for substation line 1 and line 2 maintenance are performed on alternating years. They can be organize into a PM-Set named Main Substation Maintenance. PMs should be organized under a parent PM if they are parts of the same larger continuous process - represented by the parent - such as the winter shutdown maintenance of island treatment plant.


# ==========

route:

  description: a sequential list of work, composed of job plans paired with an asset/role.
  oneOf:
    - type: null
    - type: object
      properties:

        sequence:
          type: number

        asset:
          oneOf:
            - type: null
            - $ref: "./01_asset.yml"

        role:
          oneOf:
            - type: null
            - $ref: "./02_role.yml"

        job_plan:
          oneOf:
            - type: null
            - $ref: "./32_job_plan.yml"

  $comment: the implementation could be done with Maximo route object.

#=========================================
# Resources
#-----------------------------------------
# Note: Travel time and preparation time are not being recorded explicitly on the PM. Instead they could be recorded as contributory work in the job plan route
```

```yaml
  #=========================================

  estimated_duration:

    description: is the estimated time to complete the activity in the job plan
    $ref: "./00_common_definitions.yml#/definitions/frequency_interval_definition"

  #=================

  parts_or_material_requirements:

    description: identifies the parts or material required to complete a work order generated from the PM.
    oneOf:
      - type: null
      - type: array
        items:
          $ref: "./00_common_definitions.yml#/definitions/item_requirement_definition"

  #=================

  tool_requirements:

    description: identifies the tools required to complete a work order generated from the PM.
    oneOf:
      - type: null
      - type: array
        items:
          $ref: "./00_common_definitions.yml#/definitions/tool_requirements_definition"


  #=================

  skill_and_trade_requirements:

    description: identifies the trades and qualifications of each trade needed to complete the work.
    oneOf:
    - type: null
    - type: array
      items:
        $ref: "./00_common_definitions.yml#/definitions trade_requirement_definition"

  #=================

  service_requirements:

    description: identifies (contracted) service needed to complete to complete a work order generated from the PM.
    oneOf:
      - type: null
      - type: array
        items:
          $ref: "./00_common_definitions.yml#/definitions/service_requirement_definition"


  #========================================
  # Work Triggering Specification
  #----------------------------------------
  # Note: the specification is not complete for data mapping purposes, it is complete for requirement gather
  #========================================

  next_due_date_based_on:

    type: string
    enum:
      - work start date
      - work completion date

  trigger_condition:  # aka work generation condition

    oneOf:
    - type: null
    - $ref: "#/definitions/time-based_trigger_specification"
    - $ref: "#/definitions/meter-based_trigger_specification"


  #========================================
  # Legislative Designation
  #========================================

  compliance_level:

    $ref: "./00_common_definitions.yml#/definitions/compliance_class"

    work_entity_harmonization: WR(x), WO(_), JP(_)

##########################################################
# LOCAL OBJECT DEFINITIONS
##########################################################

definitions:

  time-based_trigger_specification:

    next_due_date:
      type: number

    frequency_interval:
      $ref: "./00_common_definitions.yml#/definitions/frequency_interval_definition"

  #================

  meter-based_trigger_specification:

    next_meter_reading:
        type: number

    meter_condition:
      oneOf:
        - type: null
        - $ref:
 ↪  "./00_common_definitions.yml#/definitions/meter_condition_definition"

```

34_FR_WR_WO.yml

```
$schema: http://json-schema.org/draft-07/schema#
title: Failure Report, Work Request, Work Order
$id:
 ↪  https://raw.githubusercontent.com/TW-ASMP/TWmaximoConfig/main/1-Schemas/A-entity_record_sc

$comment: In this design, the work request doubles as a failure report.

properties:

  #================================
  # RECORD
  #================================

  ID:
  #----------------------------
    description: is a read-only unique ID, generated by the system, to uniquely
 ↪  identify the record.
    type: string

    implementation:
      MX_mapping: WONUM

  311_ticket_ID:
  #----------------------------
    implementation:
      MX_mapping: COTTICKETID
      D&C_only: true

  311_request_number:
  #----------------------------
    implementation:
      MX_mapping:
      D&C_only: true

```

```yaml
    $comment: |
      [] to resolve: there is some uncertainty of this should be either COTREQUESTNUMBER or EXTERNALREFID

record_type:
#----------------------------
  description: indicates whether this work request is being used to track an failure, without being a request for work to address the failure.
  type: string
  enum:
    - failure report
    - work request
    - work order

  rule_spec:
    - name: inference of current_type from status
      status: []

status:
#----------------------------
  description: indicate the status of failure report, work request, and work order
  type: string
  enum:
    - failure reported
    - request made
    - request approved
    - request cancelled
    - waiting on resource
    - ready to schedule
    - scheduled
    - in progress
    - completed
    - closed
    - WO cancelled

  implementation:
    MX_mapping: WO.status

following-up_on:
#----------------------------
  description: identifies the work order which this record is following up on.
  read-only: true
  oneOf:
    - type: null
    - $ref: "../A-entity_record_schema/35_work_order.yml"

#=================================
# THE OBJECT
#=================================

role_to_work_on:
#----------------------------
  description: indicates role at which the specified work is to be performed.
  oneOf:
    - type: null
    - $ref: "../A-entity_record_schema/02_role.yml"
    [] should have name and description, like asset_to_work_on

  rule_spec:
    - name: mutual exclusion of asset_to_work_on and role_to_work_on values
      spec_ID: 01JFK49T43T1TF3HBTBTDPMN5Y
      form: short
      specification: |
        one, but only one, of the following properties can have a non-null value:
          - role_to_work_on
          - asset_to_work_on

asset_to_work_on:
#------------------------------
  description: indicates asset on which the specified work is to be performed.
  oneOf:
    - type: null
    - $ref: "../A-entity_record_schema/01_asset.yml"
      properties:
        ID:
        #---------
          implementation:
            MX_mapping: WO.ASSETNUM
        name:
        #---------

  implementation:
    MX_mapping: WO.ASSETNUM
    $comment: |
      multiple fields may need to represented.

asset_or_role_criticality:
#----------------------------
  description: is the criticality values inherited from the asset or role being worked on.
  read-only: true
  oneOf:
    - type: null
    - type: object
      properties:

        rating:
          type: number

        description:
          type: string

issue_found_at_address:
#------------------------------
  description: indicates the service address in or around which a failed asset requiring maintenance is located.

  oneOf:
    - type: null
    - $ref: MaximoServiceAddressObject

route:
#----------------------------
  description: a sequential list of work, composed of job plans paired with an asset/role.
  oneOf:
    - type: null
    - type: object
      properties:

        sequence:
          type: number

        asset:
          oneOf:
            - type: null
            - $ref: "./01_asset.yml"

        role:
          oneOf:
            - type: null
            - $ref: "./02_role.yml"

        job_plan:
          $ref: "./32_job_plan.yml"

  $comment: the implementation could be done with Maximo route object.

#===================
# FAILURE REPORTING
#===================

description_of_issue:
#----------------------------
  description: is a factual description of the observable aspects of a single issue..
  type: string

  implementation:
    WR_only: true

observed_problems:
#----------------------------
  description: is what's also known as a common symptom code (e.g., making noise, cannot start, not running).
  oneOf:
    - type: null
    - type: array
      items:
        $ref: "./00_common_definitions.yml#/definitions/failure_code"

  implementation:
    WR_only: true

  rule_spec:
    - name: Which problem codes to show
      spec_ID: 01JFH3ERR08WHJ0E4WRK166WRT
      form: short
      specification: []

  $comment: multiple values are allowed

physical_causes:
#----------------------------
  oneOf:
    - type: null
    - type: array
      items:
        properties:
          cause_code:
            description: denotes a physical-based failure condition (e.g., shaft misalignment).
            $ref: "./00_common_definitions.yml#/definitions/failure_code"
            $comment: one failure code per request
            rule_spec:
              - name: Which Cause Codes to Show
                spec_ID: 01JFH2F04P28B4EB2HNWA68KN9
                form: short
                specification: |
                  - type must be "cause"
                  - [] more tbd
            action_spec:
              - name: Creating a New Failure Code
                spec_ID: 01JFH2NE68W0NCSBKKSYPZRA2Z
                form: short
                specification: User must be able to specify new failure codes,
          basis_of_selection:
            description: indicates how the failure code was derived
            type: string
            enum:
              - actual observation
              - educated guess from signs
  implementation:
    WR_only: true

bread_crumb:

found_asset_offline_due_to_this_failure:
#----------------------------
  description: indicates whether the asset was offline, because of the failure
  type: boolean
  implementation:
```

```yaml
      WR_only: true
    $comment: if true, failure reporting is required []Rule

  took_asset_offline_due_to_this_failure:
  #-----------------------------
    description: indicates whether the asset had to be taken offline, because of the failure
    type: boolean
    implementation:
      WR_only: true

  #=========================
  # WORK DETAIL
  #=========================

  work_title:
  #-----------------------------
    description: a short text summarizing the work that is being requested or have been approved to be performed.
    oneOf:
      - type: null
      - type: string

    implementation:
      MX_mapping: WO.description

  work_specification:
  #-----------------------------
    description: a sufficiently detailed description of the work being requested for the approver of the work.

    implementation:
      MX_mapping: WO.DESCRIPTION_LONGDESCRIPTION

  work_priority:
  #-----------------------------
    description: a synthetic number derived from the condition of the asset function being maintained (i.e., how close is it to failure), and the importance (or criticality) of the asset function to the organization's goals.

    implementation:
      MX_mapping: INTERNALPRIORITY

  job_plan:
  #-----------------------------
    description: specifies the job plan for the PM, if there is no further specification within the route.

    oneOf:
      - type: null
      - type: object
        $ref: "../A-entity_record_schema/32_job_plan.yml"

    action_spec:
      name: Importing Specifications from a Job Plan
      form: short
      id: 01JFVCVT6Q5F62WAHEB001J7SX
      specification: TBD []

  work_type:
  #----------------------------
    description: is the classification at the work order level
    $ref: "../B-entity_class_object_schema/33_work_type.yml"

    rule_spec:
      name: Failure Reporting Leads to Investigation or Repair
      form: short
      id: 01JFVCZ9Y7G5MWP2G2DADB2G8Z
      specification: if failure is reported, then work type must either be investigative or repair

  discrete_activity_classification:
  #----------------------------
    description: is a classification often inherited from the job plan specified on the work order
    $ref: "../B-entity_class_object_schema/33_work_type.yml"
    not_on_WR: true

  site:
  #----------------------------
    description: indicates Toronto Water's operational unit, on the organizational hierarchy, such as Waste and Storm Water Pumping (symbol - WASP) or Humber Wastewater Treatment Plant (symbol - THR).
    $ref: "./04_org_or_group.yml"

    implementation:
      $comment: WO.SITEID

  maintenance_group:
  #----------------------------
    description: indicates a sub-group of the operational unit, that is led by a supervisor who is accountable for performing the work.
    $ref: "./04_org_or_group.yml"

    rule_spec:
      name: Inherit the maintenance_group value from either the asset or the role
      spec_ID: 01JFK43CJBC495TB7Y3H3VP172
      form: very short

  requires_asset_offline:
  #--------------------
    description: indicates the work requires the asset to be offline
    oneOf:
      - type: null
      - type: string
    enum:
```

```yaml
    - yes
    - no
    - unknown

  rule_spec:
    - name: Default value of requires_asset_offline is null
      req_spec_ID: 01JFK2J0HWVWKDK4WWK5RZCXWY
      form: very short

#==================================
# RESPONSIBILITIES AND ASSIGNMENTS
#==================================

crew_assignment:
#-----------------------------
  description: identifies a crew, under the supervisor group, that is always assigned to perform the specified work

  oneOf:
    - type: null
    - $ref: "./04_org_or_group.yml"

  WO_only: true

  comment: This value will be determined by (and must be consistent with) the maintainer_organization value - situated at a lower level of the organization hierarchy

trades_assignment:
#-----------------------------
  description: identifies the individual trades-persons who will be performing the work order.
  oneOf:
    - type: null
    - type: array
      items:
        $ref: "./08_person.yml"

asset_covered_by_warranty_contract:
#-----------------------------
  description: indicates that the asset (or the asset in the role) is currently covered by a warranty contract.
  type: boolean

  implementation:
    MX_mapping: WO.WARRANTYEXIST

  todo []: rule - determine the value from the asset's warranty information.

warranty_expiration_date:
#-----------------------------
  description: indicates the date the warranty expires, if the asset is covered by a warranty contract.
```

```yaml
    type: number

    implementation:
      MX_mapping: WO.WARRANTYEXPDATE

  send_work_to_warranty_contractor:
  #------------------------------
    description: a true (or yes) value indicates that the specified work should be performed by the warranty contractor.
    type: boolean
    todo []: rule - enable this field, only if asset_covered_by_warranty_contract is true

  asset_covered_by_service_contract:
  #------------------------------
    description: indicates that the asset (or the asset in the role) is currently covered by a service contract.
    type: boolean

  send_work_to_service_contractor:
  #------------------------------
    description: a true (or yes) value indicates that the specified work should be performed by a the selected service contractor.
    oneOf:
      - type: null
      - $ref: "../A-entity_record_schema/07_service_item_master.yml"

  supports_a_capital_project:
  #------------------------------
    description: indicate that the specified work supports the work being done be a capital project consultant or contractor.
    type: boolean

  supports_the_capital_project:
  #------------------------------
    description: indicates the specific capital project (represented as a work order)
    oneOf:
      - type: null
      - $ref: "../A-entity_record_schema/35_work_order.yml"

  #=========================
  # Resources
  #=========================

  estimated_duration:
  #----------------------
    description: is the estimated time required in hours to complete the activity in the job plan
    oneOf:
      - type: null
      - type: number

```

```
part_or_material_requirements:
#-----------------------------
  description: identifies the parts or material required to complete a work order
↪ generated from the PM.
  oneOf:
    - type: null
    - type: array
      items:
        $ref:
↪ "./00_common_definitions.yml#/definitions/item_requirement_definition"

tool_requirements:
#---------------------
  description: identifies the tools required to complete a work order generated
↪ from the PM.
  oneOf:
    - type: null
    - type: array
      items:
        $ref:
↪ "./00_common_definitions.yml#/definitions/tool_requirements_definition"

service_requirements:
#---------------------
  description: identifies (contracted) service needed to complete to complete a
↪ work order generated from the PM.
  oneOf:
    - type: null
    - type: array
      items:
        $ref:
↪ "./00_common_definitions.yml#/definitions/service_requirement_definition"

skill_and_trade_requirements:
#-----------------------------
  description: identifies the trades and qualifications of each trade needed to
↪ complete the work.
  oneOf:
    - type: null
    - type: array
      items:
        $ref: "./00_common_definitions.yml#/definitions
↪ trade_requirement_definition"

#=======
# DATES
#=======

issue_reported_date:
#---------------------
  description: is the date that the issue or failure was reported.
  oneOf:
    - type: null
```

```
      - type: number

  work_requested_date:
  #---------------------
    description: is the date when the work request was submitted.
    oneOf:
      - type: null
      - type: number

  request_approval_date:
  #---------------------
    description: is the date that the work request was approved (and when it became a work order).
    oneOf:
      - type: null
      - type: number

  target_start_date:
  #---------------------
    description: is the date when the work should begin (according to a certain service standard).
    oneOf:
      - type: null
      - type: number

  target_completion_date:
  #---------------------
    description: is the date when the work should be completed (according to a certain service standard).
    oneOf:
      - type: null
      - type: number

  scheduled_start_date:
  #---------------------
    description: is the date when the work is scheduled (by a scheduler) to begin.
    oneOf:
    - type: null
    - type: number

  scheduled_completion_date:
  #---------------------
    description: is the date when the work is scheduled (by a scheduler) to be completed.
    oneOf:
      - type: null
      - type: number

  actual_start_date:
  #---------------------
    description: is the date when the work actually began.
    oneOf:
      - type: null
```

```yaml
      - type: number

  actual_completion_date:
  #---------------------
    description: is the date when the work was actually completed.
    oneOf:
      - type: null
      - type: number

  cancel_date:
  #---------------------
    description: is the date when the work was cancelled (and the record became either a cancelled work order or work request).
    oneOf:
      - type: null
      - type: number

  WO_closing_date:
  #---------------------
    description: is the date when the work order was closed.
    oneOf:
      - type: null
      - type: number

  #================================
  # Failure Information Inherited
  #================================

  member_of_PM_set:
  #-----------------
    description: indicates that this PM is a member of a set of related PMs. For example, the PMs for raw water pump 1, 2, 3 are all members of a PM set named Raw Water 5-year Disassembly Maintenance.

    oneOf:
      - type: null
      - $ref: "#/definitions/PM_set"

  parent_work_order:
  #-----------------
    description: indicates the more comprehensive PM, usually a shut-down PM, that this PM is a part of.
    oneOf:
      - type: null
      - $ref: "../A-entity_record_schema/33_PM.yml"

    WO_only: true

    rule_spec:
      name: Work Type of Descendant Work Orders
      form: short
      id: 01JFVDM89RVDCE7VBVM7FDQHRD
      specification: In a work order hierarchy, the top-level work order determines the work type of all descendant work orders.

    $comment: |
      PMs should be organized into a PM-set when they are meant to be performed at different times.  For example the PMs for substation line 1 and line 2 maintenance are performed on alternating years. They can be organize into a PM-Set named Main Substation Maintenance. PMs should be organized under a parent PM if they are parts of the same larger continuous process - represented by the parent - such as the winter shutdown maintenance of island treatment plant.

  summary_of_previous_issue_reports:

    description: presents a summary of previously reported issues and failures related to this work.
    oneOf:
      - type: null
      - type: object
        name: compiled_issue_report
        properties:
          # ------
          compiled_text_summary:
            description: the compilation of all text information in a issue report, including problem code, failure code, and description.
            oneOf:
              - type: null
              - type: array
                items:
                  - type: string
          # ------
          photographs:
            description: photographs in the failure report.
            oneOf:
              - type: null
              - type: array
                items:
                  - type: string # photos are converted to a string in JSON
                  - oneOf:
                    - contentMediaType: image/png
                    - contentMediaType: image/jpg

  mitigates_safety_risk_to_staff:
  #-------------------------------
    description: indicates the work has impact on workers' safety
    oneOf:
    - type: null
    - type: string
    enum:
      - yes
      - no
      - unspecified

  mitigates_safety_or_health_risk_to_public:
  #-----------------------------------------
    description: indicates that the work has a direct impact on the well-being of the public
    oneOf:
    - type: null
    - type: string
    enum:
      - yes
      - no
      - unspecified

  mitigates_environmental_risk:
  #-----------------------------
    description: indicates that the work has impact on environmental protection
    oneOf:
    - type: null
    - type: string
    enum:
      - yes
      - no
      - unspecified

  #========================================
  # Legislative Designation
  #========================================

  compliance_class:
  #-----------------------------
    description: indicate that the completion of the specified work would satisfy some compliance requirement of a certain Level.
    $ref: "./00_common_definitions.yml#/definitions/compliance_class"

  attachments:
  #-----------------------------
    description: documents or photographs that provide further supplementary information.
    oneOf:
      - type: null
      - $ref: "../A-entity_record_schema/35_work_order.yml"

    implementation:
      MX_mapping:


###############################
# MINOR OBJECT DEFINITIONS
###############################


```

36_work_order_documentation.yml

```
$schema: http://json-schema.org/draft-07/schema#
title: Work Order Documentation
```

.\1-Schemas\A-entity_record_schema\36_work_order_documentation.yml

```yaml
$id: https://raw.githubusercontent.com/TW-ASMP/TWmaximoConfig/1-Schemas/A-entity_record_schema/
type: object
properties:

  work_order_reference:

    description: identifies the work order, whose actual that is being documented.
    read-only: true
    $ref: "../A-entity_record_schema/35_work_order.yml"


  # OPERATIONAL STATUS INFORMATION
  #================================
  # Note: in a work order containing children work orders, these information only have to be filled in at the parent level.  [] rule

  asset_offline_at_start:

    description: indicates that the asset was offline when the work began
    type: boolean
    $comment:

  asset_brought_back_online:

    description: is only applicable if the asset was offline when the work began; this entry indicates that the work, within the scope of this work order, brought the asset back online
    type: boolean
    $comment:


  # ACTUAL WORK ORDER RESOURCE USAGE
  #================================

  revised_description_of_actual_work:

    description: is a revised and more accurate description of the actual work performed on the asset.
    type: string

  actual_start_time:


  actual_completion_time:


  actual_wrench_time:

    description: is the actual time taken to complete the work order.
    $ref: "./00_common_definitions.yml#/definitions/frequency_interval_definition"


```

```
 actual_parts_and_material_usage:

   description: identifies the parts and material used in completing the work.
   oneOf:
     - type: null
     - type: array
       items:
         $ref: "./00_common_definitions.yml#/definitions/item_requirement_definition"


 actual_tool_usage:

   description: identifies the tools used to perform the work.
   oneOf:
     - type: null
     - type: array
       items:
         $ref: "./00_common_definitions.yml#/definitions/tool_requirements_definition"


 actual_services_usage:

   description: identifies (contracted) service that was actually needed complete the work.
   oneOf:
     - type: null
     - type: array
       items:
         $ref: "./00_common_definitions.yml#/definitions/service_requirement_definition"


 actual_trades_involvement:

   description: identifies the trade and qualifications needed to complete the work.
   oneOf:
   - type: null
   - type: array
     items:
       $ref: "./00_common_definitions.yml#/definitions/trade_requirement_definition"


 # CHILDREN WORK-ORDER DOCUMENTATION
 #=================================
 # The user is able to account for additional work done, by adding new children_work_documentation items. These items would refer any work order.  [] todo: need to define a procedure for creating a new work documentation.

 children_work_documentations:
```

```yaml

    description:
    oneOf:
      - type: null
      - type: array
        items:
          $ref: "../A-entity_record_schema/36_work_documentation.yml"


  # FAILURE REPORTS AND FOLLOWUP REQUESTS
  #=================================
  # Note: the failures are reported on follow-up work requests. The linkage between a followup work order and work order documentation is found on the work order schema
```

41_meter.yml

```yaml
$schema: "http://json-schema.org/draft-07/schema#"
title: Meter
$id: https://raw.githubusercontent.com/TW-ASMP/TWmaximoConfig/main/A-entity_record_schema/41_me

properties:


```

Folder: B-entity_class_object_schema

01_asset_item_tool_class.yml

```yaml
---
$schema: "http://json-schema.org/draft-07/schema#"
title: asset item class
$id: https://raw.githubusercontent.com/TW-ASMP/TWmaximoConfig/main/1-Schemas/B-entity_class_obj
type: object

#############################################################
# 1. Properties
#############################################################

properties:

  allOf:
    - $ref: "/1-Schemas/B-entity_class_object_schema/00_common_class_definitions.yml"
    #inherit the defintions and rules from th "/1-Schemas/B-entity_class_object_schema/00_common_class_definitions.yml"

  #====================
  parent class:
    $ref: B-entity_class_object_schema/01_asset_item_tool_class_object_schema.yml

  #====================
  only used as a part:
    oneOf:
      - type: boolean
      - type: null
    description: A true value indicates that all instances of this class is always
↪   used as a part of another asset, and would never be given an asset tag. E.g.,
↪   bearing.
    rule_spec:
      - name: Do not include only used as a part in the asset classification
        spec_ID: NJ1E1ZbOgg
        status: TBS
    $comment: A false or null value materially mean the same thing.



```

02_role_class.yml

```
---
$schema: "http://json-schema.org/draft-07/schema#"
title: role classification object
$spec_ID: B-entity_class_object_schema/02_role_class.yml
type: object

##############################################################
# 1. Properties
##############################################################

properties:

  allOf:
    - $ref:
↪   "/1-Schemas/B-entity_class_object_schema/00_common_class_definitions.yml"
    #inherit the defintions and rules from th
↪   "/1-Schemas/B-entity_class_object_schema/00_common_class_definitions.yml"

  parent class:
    $ref: B-entity_class_object_schema/02_role_class.yml
    sort order: 1-30

  discrete asset role:
    oneOf: [type: boolean, type: null]
    read-only: true
    description: a role that can be occupied by a single discrete asset.

  functional structure role:
    oneOf: [type: boolean, type: null]
    read-only: true
    description: a role that cannot be occupied any discrete asset, but can be the parent to other roles.

  defined set of roles:
    oneOf: [type: boolean, type: null]
    read-only: true
    description: a role that can be occupied by a collection of descrete assets.

  compatible asset occupant class(s):
    oneOf:
      - type: array
        items:
          $ref: "./01_asset.yml"
      - type: null
    $comment: |
      []RULE NJQ6BwsVee: A asset must be an instance of one of classes listed in this field to be allowed to occupy a role under this class.

##############################################################
# 2. High level rules
##############################################################

rule_spec:
  - name: Is an asset role, functional structure role, or defined set of roles
    spec_ID: NyD4XGbuex
    specification: |
      if roleClassX is a decendent of "Discrete Asset Role" in the role classification hierarchy:
        set roleClassX.properties."a discrete asset role" to TRUE
        set roleClassX.properties."a functional structure role" to FALSE
        set roleClassX.properties."a defined ser of role" to FALSE
      elif roleClassX is a decendent of "functional structure role" in the role classification hierarchy:
        set roleClassX.properties."a discrete asset role" to FALSE
        set roleClassX.properties."a functional structure role" to TRUE
        set roleClassX.properties."a defined ser of role" to FALSE
      elif roleClassX is a decendent of "Defined Set of Roles" in the role classification hierarchy:
        set roleClassX.properties."a discrete asset role" to FALSE
        set roleClassX.properties."a functional structure role" to FALSE
        set roleClassX.properties."a defined ser of role" to TRUE
    status: specified
```

03_space_class.yml

```
title: role classification object
$id: spaceClassObject
```

```
$schema: "http://json-schema.org/draft-07/schema#"
type: object

properties:

  allOf:
    - $ref:
 ↪  "/1-Schemas/B-entity_class_object_schema/00_common_class_definitions.yml"
    #inherit the defintions and rules from th
 ↪  "/1-Schemas/B-entity_class_object_schema/00_common_class_definitions.yml"

  parent class:
    $ref: spaceClassObject
```

04_org_class.yml

```
---
$schema: "http://json-schema.org/draft-07/schema#"
title: role classification object
$id: orgGroupClassObject
type: object

properties:

  allOf:
    - $ref:
 ↪  "/1-Schemas/B-entity_class_object_schema/00_common_class_definitions.yml"
  #inherit the definitions and rules from th
 ↪  "/1-Schemas/B-entity_class_object_schema/00_common_class_definitions.yml"

  parent class:
    $ref: orgGroupClassObject

```

08_trade_type.yml

```

```

101_common_class_definitions.yml

```
---
$schema: "http://json-schema.org/draft-07/schema#"
title: generic class object
$id:
 ↪  https://raw.githubusercontent.com/TW-ASMP/TWmaximoConfig/main/1-Schemas/B-entity_class_obj
type: object
```

```yaml

# This set of properties are used in all classification objects (e.g., asset, organization, etc.)

properties:

  #====================

  class name:
    type: string
    description: a noun or short noun-phrase name of the class

  #====================

  class name with synonym(s):
    oneOf:
      - type: null
      - type: array
        items:
          type: string
    description: other synonymous names that may be used by a user in search

  #====================

  class definition:
    oneOf:
      - type: null
      - type: string

  #====================

  appliable to individual:
    type: boolean
    $comment: |
      a "FALSE" value indicates that the class is meant to be a structural part of the classifcation tree, and cannot be used to classify any entity (i.e. asset, role, space, etc).
  #====================

  retired:
    type: boolean
    rule_spec:
      - name: rules on retire classes
        spec_ID: Vy3qYEZ_ex
        type: [UI]
        specification: |
          a retired class should not appear in the list of classes for user selection. Also, the value of its properties."appliable to individual" must be FALSE.
        status: draft
```

32_discrete_activity_class.yml

```
---
title: discrete activity classification object
$id: discreteActivityClassObject
$schema: "http://json-schema.org/draft-07/schema#"
type: object

properties:

  allOf:
    - $ref: "/1-Schemas/B-entity_class_object_schema/00_common_class_definitions.yml"
    #inherit the defintions and rules from th "/1-Schemas/B-entity_class_object_schema/00_common_class_definitions.yml"

  parent class:
    $ref: discreteActivityClassObject
```

33_work_type.yml

```
title: work type object
$id: workOrderClassObject
$schema: 'http://json-schema.org/draft-07/schema'
type: object
```

Folder: 2-Classification_Trees

01_asset_classification.md

```
>**Important Note:**
>
>**This file is no longer being maintained**. The information had been ported to an ontology file for further development and editing. You can find the file at https://github.com/TW-ASMP/TWONTO/blob/main/OWL/TWONTO.ofn
>
> To view the hierarchy, save it to you computer, and open it with the desktop version of [Stanford Protege](https://protege.stanford.edu/software.php).

```

02_role_classification.md

```
## Top-Level Role Classification

* Discrete Asset Role*
```

```
    * generator role
        * backup generator role
        * emergency generator role
    * breaker role
        * bus feeder breaker role
        * line protection breaker role
        * load breaker role
        * tie-breaker role
* Collection of Assets Role*
    * functional structure role
        * system block
        * facility
            * pumping station
            * water treatment facility
            * wastewater treatment facility
            * lab
            * yard
        * process
        * Linear Functional Structure Role*
            * system train role
            * line role
                * simple line role
                * primary path line role
            * junction role
    * Defined Set of Roles*
        * defined set of discrete assets
        * defined set of functional structures


## Requirements for Implementer

* []REQ Nyh7RPjEgl #IMP "classes names specified in title-case and with an asterik symbol shall have the appliable to individual property, found in the class object, set to false"

## Notes

* []TODO #TW: the following should be moved to the asset classification.
    * system on a skid
    * system of standardized modular parts
```

03_space_classification.md

```
## Top-Level Space Classification and Examples

* Discrete Space*
    * building or structure interior
    * room interior
    * corridor
    * mezzanine
    * stairwell
    * stairwell segment
    * tunnel interior
    * equipment cabinet interior
    * storage cabinet interior
    * facility site space
    * facility site section
    * building or structure exterior
    * vehicle interior
* Collection of Spaces*
    * spaces of a building or structure [^1]
    * define set of spaces

## Requirements for Implementer

[]REQ Nyh7RPjEgl #IMP "classes names specified in camel-case and with an asterik symbol shall have the appliable to individual property, found in the class object, set to false"

## Notes
### Footnotes
[^1]: a collection of indoor and outdoor spaces inside and around a structure. []TODO #TW "add to the TWONTO".
```

04_org_classification.md

```
## Top-Level Org/Group Classification

* Level of Government*
    * provincial government
    * regional government
    * municipal government
* Group in the City*
    * cluster
    * division
    * section
    * unit
    * Group in TW*
        * work area
        * crew
* government agency
* private business
* non-governmental organization

## Requirements for Implementer

[]REQ Nyh7RPjEgl #IMP "classes names with an asterik symbol shall have the appliable to individual property, found in the class object, set to false"
```

31_work_type.md

```
## Work Type [^2]

* corrective / repair
* emergency [^1]
* investigative
* preventive
* asset data
* project
  * contractor support

## Notes
[^1]: An emergency work order is technically a corrective work that must be done urgently; may also involve an investigative component (not unlike other corrective work orders).
[^2]: The commissioning work type has been removed from this list but retained in the discrete activity classification.
```

32_discrete_activity_classification.md

```
## Top-Level Discrete Activity Classification
* Condition Evaluation*
  * quick check
  * inspection and evaluation
  * test and analysis
  * condition analysis
* sample collection
* cause investigation
* repair or service [^1]
  * calibration
  * asset replacement
  * asset part replacement
  * asset part movement
* Move or Replace*
  * new asset installation
  * asset movement
  * asset part movement
  * asset replacement
  * asset part replacement
  * asset hand-over
* Life Cycle Events*
  * asset commissioning
  * asset hand-over [^3]
  * final asset decommissioning
* Asset Modification*
  * modify asset set-point
  * physical modification to asset
  * physical modification to building or structure [^2]
* asset assignment
```

~~~
* Contributory Work*
  * item procurement
  * work coordination
  * safety preparation
  * setup
  * takedown of setup
  * travel
* design or redesign
  * creation of new role
  * removal of existing role
* Asset Data*
  * record information correction
  * record retirement



## Requirements for Implementer

```yaml
  rule_spec:
  - name: Valid Assignment of an Asset
    spec_ID: 01JDCNEFAED17CWF2K851ZAJKW
    type: [assertion]
    description: |
      classes names specified with an asterisk symbol shall have the their .property.appliable_to_individual value set to false
```



### Footnotes
[^1]: more will be added before the final implementation.
[^2]: the physical modification of a building or a structure may result in the creation and removal of a space, hence it is singled out.
[^3]: the process by which a asset's ownership is transferred from a capital project to Toronto Water.
~~~

Folder: 3-System_Hierarchies

02_role_hierarchy.md

```
# TW Highest Level Hierarchy

## The Hierarchy
- TW System
    - Drinking Water Network
        - Drinking Water Treatment Plants [^1]
        - Distribution Pumping Stations [^2]
        - Storage Assets in Drinking Water Supply Network [^3]
    - Waste and Storm Water Network
        - Collection Pumping Stations [^4]
```

```
        - Chambers in Sewer Network
        - Storages of Wet Whether Flow [^5]
        - Wastewater Treatment Plants [^6]
    - Yards
    - Independent Buildings

## The Significance in the Usage of Plurals
Where a plural noun is used, for example: Drinking Water Treatment Plants, the
 ↪  entity represents a set of things. In the case of the example - the set of
 ↪  drinkin water treatment plants in TW.

## Notes

  [^1]: i.e. {[FCL]}, {[FIS]}, {[FHO]}, {[FHA]}
  [^2]: the set of 18 pumping stations
  [^3]: the set of all reservoirs and elevated tanks
  [^4]: pumping station for waste and storm water
  [^5]: the set of all wet-whether storage, inline an offline.
  [^6]: i.e. {[THC]}, {[THR]}, {[TAB]}, {[TNT]}
```

03_space_hierarchy.md

```
# TW Highest Level Spatial Hierarchy

## The Hierarchy
- Spaces in TW
    - Spaces in Drinking Water System
        - Spaces in Drinking Water Treatment
        - Spaces in Distribution Pumping Stations
        - Spaces in Storage of Drinking Water
    - Spaces in Waste and Storm Water System
        - Spaces in Collection Pumping Stations
        - Spaces in Chambers in Sewer Network
        - Spaces in Storages of Wet Whether Flow
        - Spaces in Wastewater Treatment Plants
    - Spaces in Yards
    - Spaces in Independent Buildings

## The Significance in the Usage of Plurals
Where a plural noun is used, for example: Spaces in Drinking Water Treatment Plants,
 ↪  the entity represents a set of things. In the case of the example - the set
 ↪  spaces within the drinking water treatment plants in TW.

## Notes
```

04_org_hierarchy.md

.\4-Class_Dependent_Specifications\01_pump.yml

```
# Organizations in TW and Interact with TW

## The Hierarchy

* Region of Durham
* York Region
* Peel Region
* Province of Ontario
* Metrolinx
* TRCA
* Envave Energy
* City of Toronto
    * Solid Waste
    * Toronto Water
        * Distribution & Collection Section
            * DOS
            * WASP
            * Central Services
            * Program Maintenance
        * Water Treatment & Supply Section
        * Wastewater Treatment Section
            * Ashbridge's Bay Wastewater Treatment Plant
            * Highland Creek Wastewater Treatment Plant
                * Work Area 1
                * Work Area 2

## Notes
* the portion of this hierarchy under TW is not complete; it will be completed before Phase 3 of implementation is complete.
```

Folder: 4-Class_Dependent_Specifications

README.md

```
## Note

The Excel file in this folder contains the attribute/property defintions, including data type and value list specs, for a list of asset_classes commonly encountered or bear legislative significance.
To view the file, download it to your computer, using the marked option in the image below.


<kbd>
  <img src="https://github.com/user-attachments/assets/e62ad9c8-c1fa-401d-8c89-bd4e807372d1">
</kbd>
```

01_pump.yml

```yaml
---
$schema: "http://json-schema.org/draft-07/schema#"
title: asset
$id:
 ↪  https://raw.githubusercontent.com/TW-ASMP/TWmaximoConfig/main/4-Class_Dependent_Specificat
type: object

$comment: >

properties:

  orientation:

    oneOf:
      - type: null
      - type: string
    description: Indicates how the pump is oriented in 3D space
    enum:
      - Horizontal
      - Vertical
      - Angled
      - Inverted

    $comment: |

  #========================

  variable_speed:

    oneOf:
      - type: null
      - type: boolean
    description: Indicates the if the pump has varialbe speed control

    $comment: |

  #========================

  pump_type:

    oneOf:
      - type: null
      - type: string
    description: Indicates the pump type
    enum:
      - "Dynamic, Centrifugal"
      - "Dynamic, Axial"
      - "Reciprocating, Piston or Plunger"
      - "Reciprocating, Diaphram"
      - "Rotary, Vane"
      - "Rotary, Piston"
      - "Screw Pump"
      - "Gear Pump"

    $comment: |

  #========================

  max_rpm:

    oneOf:
      - type: null
      - type: number
    description: Indicates the maximum RPM for the pump

    $comment: |

  #========================

  max_flow:

    oneOf:
      - type: null
      - type: number
    description: Indicates the maximum flow rate at the maximum RPM for the pump in L/s

    $comment: |

  #========================


  pump_head:

    oneOf:
      - type: null
      - type: number
    description: Indicates the pressure head for the pump in metres

    $comment: |

  #========================

  submersible:

    oneOf:
      - type: null
      - type: boolean
    description: Indicates if the pump is submersible

    $comment: |

  #========================


  drive_coupling:

    oneOf:
      - type: null
      - type: string
    description: Indicates how the drive and pump are coupled together
    enum:
      - "Direct Drive"
      - "Belt Drive"
      - "Gear Drive"
      - "Flexible"
      - "Chain Drive"
      - "Hydraulic"

    $comment: |

  #========================

  drive:

    oneOf:
      - type: null
      - type: string
    description: Indicates what the mechanicly drives the pump
    enum:
      - "Electric Motor"
      - "Engine"

    $comment: |

  #========================

  sealed_bearings:

    oneOf:
      - type: null
      - type: boolean
    description: Indicates if the pump has sealed bearings

    $comment: |

  #========================
```

02_motor.yml

```
---
$schema: "http://json-schema.org/draft-07/schema#"
title: asset
$id:
 ↪  https://raw.githubusercontent.com/TW-ASMP/TWmaximoConfig/main/4-Class_Dependent_Specificat
```

```yaml
type: object

$comment: >

properties:

  type:

    oneOf:
      - type: null
      - type: string
    description: Indicates the motor type
    enum:
      - "AC"
      - "AC, Squirrel Cage Induction"
      - "AC, Wound Rotor Induction"
      - "AC, Synchronous"
      - "DC"
      - "DC, Separately Excited"
      - "DC, Self Excited"
      - "DC, Permanent Magnet"


    $comment: |

  #========================


  voltage:

    oneOf:
      - type: null
      - type: number
    description: Indicates the voltage of the motor in Volts

    $comment: |

  #========================

  horse_power:

    oneOf:
      - type: null
      - type: number
    description: Indicates the horse power of the motor

    $comment: |

  #========================

  nema_frame:

    oneOf:
      - type: null
      - type: string
    description: Indicates the NEMA frame type for the motor
    enum:
      - "42"
      - "48"
      - "56"
      - "66"
      - "182"
      - "184"
      - "213"
      - "215"
      - "1412AT"
      - "143T"
      - "145T"
      - "146AT"
      - "148AT"
      - "149AT"
      - "182AT"
      - "182T"
      - "184T"
      - "186ACY"
      - "186AT"
      - "189AT"
      - "203#"
      - "204#"
      - "2110AT"
      - "213T"
      - "215T"
      - "219AT"
      - "224#"
      - "225#"
      - "254#"
      - "254T"
      - "254U"
      - "256T"
      - "256U"
      - "284#"
      - "284T"
      - "284TS"
      - "284U"
      - "286T"
      - "286TS"
      - "286U"
      - "324#"
      - "324T"
      - "324TS"
      - "324U"
      - "326#"
      - "326T"
      - "326TS"
      - "326U"
      - "364#"
      - "364S#"
      - "364T"
      - "364TS"
      - "364U"
      - "365#"
      - "365T"
      - "365TS"
      - "365U"
      - "404T"
      - "404TS"
      - "404U"
      - "405T"
      - "405TS"
      - "405U"
      - "444T"
      - "444TS"
      - "444U"
      - "445T"
      - "445TS"
      - "445U"
      - "447T&&"
      - "447TS&&"
      - "449T"
      - "449TS"
      - "48H"
      - "56H"
      - "56HZ"
      - "L182ACY"
      - "L186AT"


    $comment: |

  #========================


  nema_enclosure:

    oneOf:
      - type: null
      - type: string
    description: Indicates what the NEMA enclosure type for the motor
    enum:
      - ODP
      - TEFC
      - TENV
      - TEAO
      - TEWD
      - EXPL
      - HAZ


    $comment: |

  #========================

  sealed_bearings:

    oneOf:
      - type: null
      - type: boolean
    description: Indicates if the motor has sealed bearings

    $comment: |

  #========================
```

03_valve.yml

```
---
$schema: "http://json-schema.org/draft-07/schema#"
title: asset
$id:
 ↪  https://raw.githubusercontent.com/TW-ASMP/TWmaximoConfig/main/4-Class_Dependent_Specificat
type: object

$comment: >

properties:

  type:

    oneOf:
      - type: null
      - type: string
    description: Indicates the valve type
    enum:
      - "Ball"
      - "Butterfly"
      - "Cone"
      - "Diaphragm"
      - "Gate Valve"
      - "Globe Valve"
      - "Knife Valve"
      - "Needle Valve"
      - "Pinch Valve"
      - "Plug Valve"

    $comment: may be redundent to lablel them valve in the valve type

  #========================

  special_type:

```

```yaml
    oneOf:
      - type: null
      - type: string
    description: Indicates what type of specialised valve it is if it is a specialised valve
    enum:
      - "Air release valve"
      - "Backflow Preventer"
      - "Check"
      - "Pressure Relief Valve"
      - "Solenoid"


    $comment: may be redundent to lablel them valve in the valve type

  #========================

  size:

    oneOf:
      - type: null
      - type: number
    description: Indicates the valve size in inches

    $comment: |

  #========================

  ansi_type:

    oneOf:
      - type: null
      - type: string
    description: Indicates the ansi type for the valve
    enum:
      - 150
      - 300
      - 400
      - 600
      - 900
      - 1500
      - 2500
      - 4500

    $comment:

  #========================

  ansi_class:

    oneOf:
      - type: null
      - type: string
    description: Indicates the ansi class for the valve
    enum:
      - "A - Standard"
      - "B - Special"
      - "Limited"

    $comment:

  #========================

  horse_power:

    oneOf:
      - type: null
      - type: number
    description: Indicates the horse power of the motor

    $comment: |

  #========================

  nema_frame:

    oneOf:
      - type: null
      - type: string
    description: Indicates the NEMA frame type for the motor
    enum:
      - "42"
      - "48"
      - "56"
      - "66"
      - "182"
      - "184"
      - "213"
      - "215"
      - "1412AT"
      - "143T"
      - "145T"
      - "146AT"
      - "148AT"
      - "149AT"
      - "182AT"
      - "182T"
      - "184T"
      - "186ACY"
      - "186AT"
      - "189AT"
      - "203#"
      - "204#"
      - "2110AT"
      - "213T"
      - "215T"
```

```
      - "219AT"
      - "224#"
      - "225#"
      - "254#"
      - "254T"
      - "254U"
      - "256T"
      - "256U"
      - "284#"
      - "284T"
      - "284TS"
      - "284U"
      - "286T"
      - "286TS"
      - "286U"
      - "324#"
      - "324T"
      - "324TS"
      - "324U"
      - "326#"
      - "326T"
      - "326TS"
      - "326U"
      - "364#"
      - "364S#"
      - "364T"
      - "364TS"
      - "364U"
      - "365#"
      - "365T"
      - "365TS"
      - "365U"
      - "404T"
      - "404TS"
      - "404U"
      - "405T"
      - "405TS"
      - "405U"
      - "444T"
      - "444TS"
      - "444U"
      - "445T"
      - "445TS"
      - "445U"
      - "447T&&"
      - "447TS&&"
      - "449T"
      - "449TS"
      - "48H"
      - "56H"
      - "56HZ"
      - "L182ACY"
      - "L186AT"
```

```


  $comment: |

#========================


nema_enclosure:

  oneOf:
    - type: null
    - type: string
  description: Indicates what the NEMA enclosure type for the motor
  enum:
    - ODP
    - TEFC
    - TENV
    - TEAO
    - TEWD
    - EXPL
    - HAZ


  $comment: |

#========================

sealed_bearings:

  oneOf:
    - type: null
    - type: boolean
  description: Indicates if the motor has sealed bearings

  $comment: |

#========================


cwp:

  oneOf:
    - type: null
    - type: number
  description: Indicates the cold working pressure of the valve in psi

  $comment: |

#========================


nominal_pressure:

  oneOf:
    - type: null
    - type: number
  description: Indicates the nominal pressure of the valve in psi

  $comment: |

#========================


valve_body:

  oneOf:
    - type: null
    - type: string
  description: Indicates the valve body material
  enum:
    - "Carbon Steel"
    - "Stainless Steel"
    - "Duplex"
    - "Alloy"
    - "Composite"
    - "Titanium"

  $comment: |

#========================

actuator_type:

  oneOf:
    - type: null
    - type: string
  description: Indicates the type of actuator
  enum:
    - Electric
    - Pneumatic
    - Hydraulic
    - Manual

  $comment: |

#========================

stem_seal:

  oneOf:
    - type: null
    - type: string
  description: Indicates the type of stem seal for the valve
  enum:
    - "Duplex"
    - "Lip Seal"
    - "O-ring"
    - "Stuffing Box"

  $comment: |

#========================


```

04_breaker.yml

```
---
$schema: "http://json-schema.org/draft-07/schema#"
title: asset
$id:
 ↪  https://raw.githubusercontent.com/TW-ASMP/TWmaximoConfig/main/4-Class_Dependent_Specificat
type: object

$comment: >

properties:

  type:

    oneOf:
      - type: null
      - type: string
    description: Indicates the breaker type
    enum:
      - "Insulated Case"
      - "Metal Clad or Enclosed"
      - "Molded Case"

    $comment:

  #========================

  max_voltage:

    oneOf:
      - type: null
      - type: number
    description: Indicates what the maximum continuous voltage rating for the
 ↪  breaker in Volts

    $comment:

  #========================

  max_amperage:

    oneOf:
      - type: null
      - type: number
    description: Indicates what the maximum continuous current rating for the breaker in Amps

    $comment: |

  #========================

  main_contactor:

    oneOf:
      - type: null
      - type: string
    description: Indicates the ansi type for the valve
    enum:
      - "Air Insulated"
      - "Air Insulated, Air Blast"
      - "Vacuum Insulated"
      - "Oil Insultated"
      - "Gas Insultated"

    $comment:

  #========================

```

05_starter.yml

```
---
$schema: "http://json-schema.org/draft-07/schema#"
title: asset
$id: https://raw.githubusercontent.com/TW-ASMP/TWmaximoConfig/main/4-Class_Dependent_Specificat
type: object

$comment: >

properties:

  voltage_rating:

    oneOf:
      - type: null
      - type: number
    description: Indicates the continuous voltage rating for the starter in Volts

    $comment:

  #========================
```

```yaml

  current_rating:

    oneOf:
      - type: null
      - type: number
    description: Indicates the continuous current rating for the starter in Amps

    $comment:

  #=========================

  vfd_present:

    oneOf:
      - type: null
      - type: boolean
    description: Indicates the presence of a variable frequency drive within the starter

    $comment:

  #=========================

  soft_start_present:

    oneOf:
      - type: null
      - type: boolean
    description: Indicates the function of a soft starter in  the starter

    $comment:

  #=========================

  main_contactor:

    oneOf:
      - type: null
      - type: string
    description: Indicates the ansi type for the valve
    enum:
      - "Air Insulated"
      - "Air Insulated, Air Blast"
      - "Vacuum Insulated"
      - "Oil Insultated"
      - "Gas Insultated"

    $comment:

  #=========================

```

06_transformer.yml

```
---
$schema: "http://json-schema.org/draft-07/schema#"
title: asset
$id:
 ↪  https://raw.githubusercontent.com/TW-ASMP/TWmaximoConfig/main/4-Class_Dependent_Specificat
type: object

$comment: >

properties:

  voltage_primary:

    oneOf:
      - type: null
      - type: number
    description: Indicates the primary voltage rating for the transformer in Volts

    $comment:

  #========================

  voltage_secondary:

    oneOf:
      - type: null
      - type: number
    description: Indicates the secondary voltage rating for the transformer in Volts

    $comment:

  #========================

  power_rating:

    oneOf:
      - type: null
      - type: number
    description: Indicates the power rating for the transformer in kVA

    $comment:

  #========================

  oil_filled:

    oneOf:
      - type: null
      - type: boolean
    description: Indicates the requirement for oil cooling for the transformer
```

```yaml

  $comment:

#========================

pressure_relay:

  oneOf:
    - type: null
    - type: boolean
  description: Indicates the presence of a sudden pressure relay

  $comment:

#========================

cooling_air_fan:

  oneOf:
    - type: null
    - type: boolean
  description: Indicates the presence of a cooling air fan(s)

  $comment:

#========================

coolant_pump:

  oneOf:
    - type: null
    - type: boolean
  description: Indicates the presence of a coolant pump

  $comment:

#========================

gas_monitor:

  oneOf:
    - type: null
    - type: boolean
  description: Indicates the presence of a gas monitor

  $comment:

#========================

```

07_hvac.yml

```
---
$schema: "http://json-schema.org/draft-07/schema#"
title: asset
$id:
 ↪ https://raw.githubusercontent.com/TW-ASMP/TWmaximoConfig/main/4-Class_Dependent_Specificat
type: object

$comment: >

properties:

  cfm_rating:

    oneOf:
      - type: null
      - type: number
    description: Indicates the capacity rating for the hvac unit in CFM

    $comment:

  #========================

  static_pressure:

    oneOf:
      - type: null
      - type: number
    description: Indicates the static pressure for the hvac unit in inch water
 ↪  column

    $comment:

  #========================

  heating:

    oneOf:
      - type: null
      - type: boolean
    description: Indicates the presence of a heating function in the hvac unit such
 ↪  as heating coil or gasburner

    $comment:

  #========================

  cooling:

    oneOf:
      - type: null
      - type: boolean
    description: Indicates the presence of a cooling function in the hvac unit such as cooling coil

    $comment:

  #=========================

  dehumidifier:

    oneOf:
      - type: null
      - type: boolean
    description: Indicates the presence of a dehumidifier

    $comment:

  #=========================

  maintainable_damper:

    oneOf:
      - type: null
      - type: boolean
    description: Indicates the presence of a maintainable damper
    $comment:

  #=========================

  belts_present:

    oneOf:
      - type: null
      - type: boolean
    description: Indicates the presence of replacable belts

    $comment:

  #=========================

```

08_blower_fan.yml

```
---
$schema: "http://json-schema.org/draft-07/schema#"
title: asset
$id: https://raw.githubusercontent.com/TW-ASMP/TWmaximoConfig/main/4-Class_Dependent_Specificat
type: object

$comment: >
```

.\4-Class_Dependent_Specifications\08_blower_fan.yml

```yaml

properties:

  cfm_rating:

    oneOf:
      - type: null
      - type: number
    description: Indicates the capacity rating for the hvac unit in CFM

    $comment:

  #========================

  static_pressure:

    oneOf:
      - type: null
      - type: number
    description: Indicates the static pressure for the hvac unit in inch water column

    $comment:

  #========================

  drive_coupling:

    oneOf:
      - type: null
      - type: string
    description: Indicates the type of drive coupling
    enum:
      - "Direct Drive"
      - "Belt Drive"
      - "Gear Drive"
      - "Flexible"
      - "Chain Drive"
      - "Hydraulic"

    $comment:

  #========================

  sealed_bearings:

    oneOf:
      - type: null
      - type: boolean
    description: Indicates the presence of sealed of shielded bearings

    $comment:

```

```
  #========================
```

09_compressor.yml

```
---
$schema: "http://json-schema.org/draft-07/schema#"
title: asset
$id: https://raw.githubusercontent.com/TW-ASMP/TWmaximoConfig/main/4-Class_Dependent_Specificat
type: object

$comment: >

properties:

  rated_pressure:

    oneOf:
      - type: null
      - type: number
    description: Indicates the capacity rating for the compressor in kPa

    $comment:

  #========================

  rated_flow:

    oneOf:
      - type: null
      - type: number
    description: Indicates the capacity rating for the compressor in SCMH

    $comment:

  #========================

  drive_coupling:

    oneOf:
      - type: null
      - type: string
    description: Indicates the type of drive coupling
    enum:
      - "Direct Drive"
      - "Belt Drive"
      - "Gear Drive"
      - "Flexible"
      - "Chain Drive"
      - "Hydraulic"

    $comment:

  #========================

  sealed_bearings:

    oneOf:
      - type: null
      - type: boolean
    description: Indicates the presence of sealed of shielded bearings

    $comment:

  #========================
```

10_generator.yml

```
---
$schema: "http://json-schema.org/draft-07/schema#"
title: asset
$id:
 ↪  https://raw.githubusercontent.com/TW-ASMP/TWmaximoConfig/main/4-Class_Dependent_Specificat
type: object

$comment: >

properties:


  mobile:

    oneOf:
      - type: null
      - type: boolean
    description: Indicates if the generator is mobile

    $comment:

  #========================

  power_rated:

    oneOf:
      - type: null
      - type: number
    description: Indicates the power rating in kilo watts

    $comment:

  #========================

```

```yaml
  voltage_rating:

    oneOf:
      - type: null
      - type: number
    description: Indicates the capacity rating for the generator in Volts

    $comment:

  #========================

  drive_type:

    oneOf:
      - type: null
      - type: string
    description: Indicates the type of drive
    enum:
      - "Engine, Diesel or Bio-diesel"
      - "Engine, Natural Gas"
      - "Turbine"

    $comment:

  #========================

  brushless:

    oneOf:
      - type: null
      - type: boolean
    description: Indicates the presence of brushes within the generator

    $comment:

  #========================

  test_load:

    oneOf:
      - type: null
      - type: boolean
    description: Indicates the presence of an electrical connection to attach a test load

    $comment:

  #========================

  drive_coupling:

    oneOf:
      - type: null
      - type: string
    description: Indicates the type of drive coupling
    enum:
      - "Direct Drive"
      - "Belt Drive"
      - "Gear Drive"
      - "Flexible"
      - "Chain Drive"
      - "Hydraulic"

    $comment:

  #========================

  sealed_bearings:

    oneOf:
      - type: null
      - type: boolean
    description: Indicates the presence of sealed of shielded bearings

    $comment:

  #========================
```

11_ups.yml

```
---
$schema: "http://json-schema.org/draft-07/schema#"
title: asset
$id:
 ↪  https://raw.githubusercontent.com/TW-ASMP/TWmaximoConfig/main/4-Class_Dependent_Specificat
type: object

$comment: >

properties:


  fls:

    oneOf:
      - type: null
      - type: boolean
    description: Indicates if the UPS directly supports any part of the fire
 ↪  supression or any life safety systems

    $comment:

  #========================

```

```yaml
battery_capacity:

  oneOf:
    - type: null
    - type: number
  description: Indicates the power rating in kilo watt hours

  $comment:

#========================

voltage_output:

  oneOf:
    - type: null
    - type: number
  description: Indicates the output voltage of the UPS in Volts

  $comment:

#========================

battery_type:

  oneOf:
    - type: null
    - type: string
  description: Indicates the type of drive
  enum:
    - "Sealed / Valve Regulated Lead Acid"
    - "Flooded / Vented Lead Acid"
    - "NiCad"
    - "Li-ion"

  $comment:

#========================

intergrated_charger:

  oneOf:
    - type: null
    - type: boolean
  description: Indicates that the UPS is physically integrated with charger

  $comment:

#========================
```

.\4-Class_Dependent_Specifications\12_boiler.yml

12_boiler.yml

```yaml
---
$schema: "http://json-schema.org/draft-07/schema#"
title: asset
$id:
 ↪  https://raw.githubusercontent.com/TW-ASMP/TWmaximoConfig/main/4-Class_Dependent_Specificat
type: object

$comment: >

properties:

  energy_source:

    oneOf:
      - type: null
      - type: string
    description: Indicates the fuel source for the boiler
    enum:
      - "Natural Gas"
      - "Biogas / Digester Gas"
      - "Propane"
      - "Electric"
      - "Fuel Oil"

    $comment:

  #=========================

  boiled_medium:

    oneOf:
      - type: null
      - type: string
    description: Indicates the boiled/heated medium within the boiler
    enum:
      - Water
      - Steam
      - Glycol

    $comment:

  #=========================

  heat_rate:

    oneOf:
      - type: null
      - type: number
    description: Indicates the rated heat rate in the boiler in kilo Watts

    $comment:

  #========================

  max_pressure:

    oneOf:
      - type: null
      - type: number
    description: Indicates the maximum operating pressure for the boiler in psi

    $comment:

  #========================

  max_temperature:

    oneOf:
      - type: null
      - type: number
    description: Indicates the maximum operating temperature for the boiler in degrees celsius

    $comment:

  #========================

  heated_surface:

    oneOf:
      - type: null
      - type: number
    description: Indicates the heating surface area of the boiler in meters squared

    $comment:

  #========================

  capacity:

    oneOf:
      - type: null
      - type: number
    description: Indicates the size capacity of the boiler in Litres

    $comment:

  #========================

  tssa_crn:

    oneOf:
      - type: null
      - type: string
    description: Indicates the CRN number issued by the TSSA

    $comment:

  #========================
```

13_pressure_vessel.yml

```
---
$schema: "http://json-schema.org/draft-07/schema#"
title: asset
$id:
 ↪  https://raw.githubusercontent.com/TW-ASMP/TWmaximoConfig/main/4-Class_Dependent_Specificat
type: object

$comment: >

properties:

  contained_medium:

    oneOf:
      - type: null
      - type: string
    description: Indicates the medium within the pressure vessel
    enum:
      - "Water"
      - "Steam"
      - "Glycol"
      - "Refrigerant"
      - "Compressed Air"
      - "Digester Gas"
      - "Ozone"

    $comment:

  #========================

  capacity:

    oneOf:
      - type: null
      - type: number
    description: Indicates the size capacity of the pressure vessel in Litres

    $comment:

  #========================

  max_pressure:

    oneOf:
      - type: null
      - type: number
    description: Indicates the maximum operating pressure for the pressure vessel in
 ↪  psi

    $comment:

  #========================

  tssa_crn:

    oneOf:
      - type: null
      - type: string
    description: Indicates the CRN number issued by the TSSA

    $comment:

  #========================
```

14_pressure_piping.yml

```
---
$schema: "http://json-schema.org/draft-07/schema#"
title: asset
$id:
 ↪  https://raw.githubusercontent.com/TW-ASMP/TWmaximoConfig/main/4-Class_Dependent_Specificat
type: object

$comment: >

properties:

  max_size:

    oneOf:
      - type: null
      - type: number
    description: Indicates the maximum pipe size in inches

    $comment:

  #========================

  max_pressure:

    oneOf:
      - type: null
      - type: number
    description: Indicates the maximum working pressure within the pressure piping in kPa

    $comment:

  #========================

  max_temperature:

    oneOf:
      - type: null
      - type: number
    description: Indicates the maximum medium temperature within the pressure piping in degrees celsius

    $comment:

  #========================

  contained_medium:

    oneOf:
      - type: null
      - type: string
    description: Indicates the medium within the pressure piping
    enum:
      - "Water"
      - "Steam"
      - "Glycol"
      - "Refrigerant"
      - "Compressed Air"
      - "Digester Gas"
      - "Ozone"

    $comment:

  #========================

  special_application:

    oneOf:
      - type: null
      - type: string
    description: Indicates the speical application required for pressure piping
    enum:
      - "piping in fire protection system"
      - "piping in heating system"
      - "piping in refrigeration system"
      - "compressed air piping"
      - "hot oil piping"
      - "buried water piping"

    $comment:
```

```

  #========================

  tssa_crn:

    oneOf:
      - type: null
      - type: string
    description: Indicates the CRN number issued by the TSSA

    $comment:

  #========================
```

15_instrumentation.yml

```
---
$schema: "http://json-schema.org/draft-07/schema#"
title: asset
$id:
 ↪  "https://raw.githubusercontent.com/TW-ASMP/TWmaximoConfig/main/4-Class_Dependent_Specifica
type: object

properties:

  parameter:
    oneOf:
      - type: null
      - type: string
    description: Indicates the parameter that is being measured
    enum:
      - "Density"
      - "Flow Rate"
      - "Humidity"
      - "Level"
      - "Current"
      - "Power"
      - "Position"
      - "Pressure"
      - "Speed"
      - "Temperature"
      - "Torque"
      - "UV"
      - "Vibration"
      - "Weight"
      - "Specific Gravity"
      - "Ammonia"
      - "Carbon Monoxide"
      - "Chloramination"
      - "Chlorine"
      - "Dissolved Oxygen"
```

```yaml
      - "Fluoride"
      - "Methane/LEL"
      - "Total Hydrocarbon"
      - "ORP"
      - "Ozone"
      - "Particulate"
      - "PH"
      - "Sulphite"
      - "Sulphur Dioxide"
      - "Suspended Solids"
      - "Turbidity"

  uom:
    oneOf:
      - type: null
      - type: string
    description: Indicates the unit of measurement that the instrumentation is reporting values in

allOf:
  - if:
      properties:
        parameter:
          const: "Density"
    then:
      properties:
        uom:
          enum:
            - "Kilograms Per Cubic Meter (kg/m³)"
            - "Grams Per Cubic Centimeter (g/cm³)"
            - "Grams Per Milliliter (g/mL)"
            - "Pounds Per Cubic Foot (lb/ft³)"

  - if:
      properties:
        parameter:
          const: "Flow Rate"
    then:
      properties:
        uom:
          enum:
            - "Liters Per Second (L/s)"
            - "Cubic Meters Per Second (m³/s)"
            - "Gallons Per Minute (GPM)"
            - "Cubic Feet Per Minute (CFM)"
            - "Liters Per Hour (L/h)"
            - "Standard Cubic Feet Per Minute (SCFM)"

  - if:
      properties:
        parameter:
          const: "Humidity"
    then:
      properties:
        uom:
          enum:
            - "Percentage (%)"
            - "Grams Per Milliliter (g/mL)"
            - "Milligrams Per Liter (mg/L)"
            - "Parts Per Million (ppm)"
            - "Parts Per Billion (ppb)"

  - if:
      properties:
        parameter:
          const: "Level"
    then:
      properties:
        uom:
          enum:
            - "Centimeters (cm)"
            - "Meters (m)"
            - "Inch (in)"
            - "Percentage (%)"
            - "Feet (ft)"

  - if:
      properties:
        parameter:
          const: "Current"
    then:
      properties:
        uom:
          enum:
            - "Ampere (A)"
            - "Milliampere (mA)"

  - if:
      properties:
        parameter:
          const: "Power"
    then:
      properties:
        uom:
          enum:
            - "Watt (W)"
            - "kilowatt (kW)"
            - "Megawatt (MW)"

  - if:
      properties:
        parameter:
          const: "Position"
    then:
      properties:
        uom:
          enum:
            - "Centimeters (cm)"
            - "Meters (m)"
            - "Inch (in)"
            - "Millimeter (mm)"
            - "Feet (ft)"
            - "Degree (°)"
            - "Randians (rad)"
            - "Unitless"

  - if:
      properties:
        parameter:
          const: "Pressure"
    then:
      properties:
        uom:
          enum:
            - "Pascal (Pa)"
            - "Kilopascal (kPa)"
            - "Bar"
            - "Atmosphere (atm)"
            - "Pounds Per Square Inch (PSI)"
            - "Millimeter of Mercury (mmHg)"
            - "Millimeter of Water (mmH2O)"
            - "Inch of Water (\"WC)"

  - if:
      properties:
        parameter:
          const: "Speed"
    then:
      properties:
        uom:
          enum:
            - "Meters Per Second (m/s)"
            - "Kilometers Per Hour (km/h)"
            - "Feet Per Second (ft/s)"
            - "Mile Per Hour (mph)"
            - "Revolutions Per Minute (RPM)"

  - if:
      properties:
        parameter:
          const: "Temperature"
    then:
      properties:
        uom:
          enum:
            - "Degree Celsius (°C)"
            - "Degree Fahrenheit (°F)"
  - if:
      properties:
        parameter:
          const: "Torque"
    then:
      properties:
        uom:
          enum:
            - "Newton-meters (N·m)"
            - "Foot-pounds (ft·lb)"

  - if:
      properties:
        parameter:
          const: "UV"
    then:
      properties:
        uom:
          enum:
            - "Watts Per Square Meter (W/m²)"
            - "Percentage (%)"

  - if:
      properties:
        parameter:
          const: "Vibration"
    then:
      properties:
        uom:
          enum:
            - "Meters Per Second (m/s)"
            - "Centimeters Per Second (cm/s)"
            - "Feet Per Seconf (ft/s)"
            - "Inch Per Second (in/s)"
            - "Meters Per Second Square (m/s2)"
            - "Centimeters Per Second Square (cm/s2)"
            - "Feet Per Second Square (ft/s2)"
            - "Inch Per Second Square (in/s2)"
            - "Hertz (Hz)"

  - if:
      properties:
        parameter:
          const: "Weight"
    then:
      properties:
        uom:
          enum:
            - "Grams (g)"
            - "Kilograms (kg)"
            - "Pounds (lb)"
            - "Metric Tons (tonne)"

```

```yaml
  - if:
      properties:
        parameter:
          const: "Specific Gravity"
    then:
      properties:
        uom:
          enum:
            - "Unitless"

  - if:
      properties:
        parameter:
          const: "Ammonia"
    then:
      properties:
        uom:
          enum:
            - "Parts Per Million (ppm)"
            - "Parts Per Billion (ppb)"
            - "Milligrams Per Cubic Meter (mg/m³)"
            - "Percentage (%)"
            - "Micrograms Per Cubic Meter (µg/m³)"

  - if:
      properties:
        parameter:
          const: "Carbon Monoxide"
    then:
      properties:
        uom:
          enum:
            - "Parts Per Million (ppm)"
            - "Parts Per Billion (ppb)"
            - "Milligrams Per Cubic Meter (mg/m³)"
            - "Percentage (%)"
            - "Micrograms Per Cubic Meter (µg/m³)"

  - if:
      properties:
        parameter:
          const: "Chloramination"
    then:
      properties:
        uom:
          enum:
            - "Grams Per Milliliter (g/mL)"
            - "Milligrams Per Liter (mg/L)"
            - "Parts Per Million (ppm)"
            - "Parts Per Billion (ppb)"


  - if:
      properties:
        parameter:
          const: "Chlorine"
    then:
      properties:
        uom:
          enum:
            - "Grams Per Milliliter (g/mL)"
            - "Milligrams Per Liter (mg/L)"
            - "Parts Per Million (ppm)"
            - "Parts Per Billion (ppb)"

  - if:
      properties:
        parameter:
          const: "Dissolved Oxygen"
    then:
      properties:
        uom:
          enum:
            - "Grams Per Milliliter (g/mL)"
            - "Milligrams Per Liter (mg/L)"
            - "Parts Per Million (ppm)"
            - "Parts Per Billion (ppb)"

  - if:
      properties:
        parameter:
          const: "Fluride"
    then:
      properties:
        uom:
          enum:
            - "Grams Per Milliliter (g/mL)"
            - "Milligrams Per Liter (mg/L)"
            - "Parts Per Million (ppm)"
            - "Parts Per Billion (ppb)"

  - if:
      properties:
        parameter:
          const: "Methane/LEL"
    then:
      properties:
        uom:
          enum:
            - "Parts Per Million (ppm)"
            - "Parts Per Billion (ppb)"
            - "Milligrams Per Cubic Meter (mg/m³)"
            - "Percentage (%)"
            - "Micrograms Per Cubic Meter (µg/m³)"

  - if:
      properties:
        parameter:
          const: "Total Hydrocarbon"
    then:
      properties:
        uom:
          enum:
            - "Grams Per Milliliter (g/mL)"
            - "Milligrams Per Liter (mg/L)"
            - "Parts Per Million (ppm)"
            - "Parts Per Billion (ppb)"

  - if:
      properties:
        parameter:
          const: "ORP"
    then:
      properties:
        uom:
          enum:
            - "Grams Per Milliliter (g/mL)"
            - "Milligrams Per Liter (mg/L)"
            - "Parts Per Million (ppm)"
            - "Parts Per Billion (ppb)"

  - if:
      properties:
        parameter:
          const: "Ozone"
    then:
      properties:
        uom:
          enum:
            - "Grams Per Milliliter (g/mL)"
            - "Milligrams Per Liter (mg/L)"
            - "Parts Per Million (ppm)"
            - "Parts Per Billion (ppb)"

  - if:
      properties:
        parameter:
          const: "Particulate"
    then:
      properties:
        uom:
          enum:
            - "Grams Per Milliliter (g/mL)"
            - "Milligrams Per Liter (mg/L)"
            - "Parts Per Million (ppm)"
            - "Parts Per Billion (ppb)"

  - if:
      properties:
```

```yaml
        parameter:
          const: "PH"
    then:
      properties:
        uom:
          enum:
            - "Unitless"

  - if:
      properties:
        parameter:
          const: "Sulphite"
    then:
      properties:
        uom:
          enum:
            - "Grams Per Milliliter (g/mL)"
            - "Milligrams Per Liter (mg/L)"
            - "Parts Per Million (ppm)"
            - "Parts Per Billion (ppb)"

  - if:
      properties:
        parameter:
          const: "Sulphur Dioxide"
    then:
      properties:
        uom:
          enum:
            - "Parts Per Million (ppm)"
            - "Parts Per Billion (ppb)"
            - "Milligrams Per Cubic Meter (mg/m³)"
            - "Percentage (%)"
            - "Micrograms Per Cubic Meter (µg/m³)"

  - if:
      properties:
        parameter:
          const: "Suspended Solids"
    then:
      properties:
        uom:
          enum:
            - "Grams Per Milliliter (g/mL)"
            - "Milligrams Per Liter (mg/L)"
            - "Parts Per Million (ppm)"
            - "Parts Per Billion (ppb)"

  - if:
      properties:
        parameter:
          const: "Turbity"
    then:
      properties:
        uom:
          enum:
            - "Grams Per Milliliter (g/mL)"
            - "Milligrams Per Liter (mg/L)"
            - "Parts Per Million (ppm)"
            - "Parts Per Billion (ppb)"
```

Folder: 5-Functions