Toronto Water WMS Configuration Schema

Table of contents

.\1-Schemas\A-entity_record_schema\00_common_definitions.yml

Folder: 1-Schemas

Folder: A-entity_record_schema

00_common_definitions.yml

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


definitions:

# ATTACHMENTS

  attachment_def:
    description:
    oneOf:
      - type: null
      - type: array
        items:
          $ref: Maximo_DOCLINKS

# INVENTORY

  rotating_property_def:
    type: boolean
    $comment: |
      A rotating item is a trackable item, represented also as an asset in Maximo.
 ↪ 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_options, 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

```

.\1-Schemas\A-entity_record_schema\00_common_definitions.yml

```yaml
  manufacturer_and_model_def:
    type: object
    properties:
      manufacturer:
        $ref: MaximoCompanyObject
      model_and_sub-model:
        oneOf:
        - type: null
        - type: string
        description: For example, "Multilin 869"
      product_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.

# 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
```

```yaml
     - approved

# RESOURCE FOR PLAN AND WORK

 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:
   name:
    description: is the full name of the unit of measure.
```

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

# FREQ DEFINITION

  frequency_interval_definition:
    properties:
      frequency_quantity:
        type: number
      unit_of_time:
        type: string
        enum:
          - minute
          - hour
          - day
          - month
          - year

# RECORD STATUS AND RETIREMENT

  record_retirement_def:
    properties:

      record_is_active:
        type: Boolean

      retired_by:
        oneOf:
          - type: null
          - $ref: "./08_person.yml"

      reason_for_retirement:
        read-only: true
        #note: reason for requirement should be
        oneOf:
          - type: null
          - type: string
            enum:
              - "record is a duplicate"
              - "what record represents is gone"
              - "what record represents is lost"
              - "record added by mistake"
              - "information out-dated"
              - "information inaccurate"

      date_of_actual_removal_or_departure:
        description: is the date that the entity referent of the record was removed or had departed
        oneOf:
```

```yaml
     - type: null
     - type: string
      format: "date"

   date_of_record_retirement:
    description: is the date when the record was retired
    oneOf:
     - type: null
     - type: string
      format: "date"

# DUPLICATION HANDLING

 duplicate_record_def:
  oneOf:
   - type: null
   - type: array
    items:
     type: object
     $comment: the object may other asset, functional-location, physical-location, ... records, depending on the context of where this property is used.

# METERS

 meter_condition_definition:
  properties:

   meter:
    description: is a selection of a pre-defined meter object.
    $ref: "./41_meter.yml"

   numeric_interval_value:
    oneOf:
     - type: null
     - type: number

   characteristic_trigger_value:
    oneOf:
     - type: null
     - type: string

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

  compliance_class:
    type: string
    enum:
      - legislative
      - corporate policy
```

01_asset.yml

```
---
$schema: "http://json-schema.org/draft-07/schema#"
title: asset
$id: https://raw.githubusercontent.com/TW-ASMP/TWDM/main/1-Schemas/A-entity_record_schema/01_as
type: object

$comment: >

properties:

  ID:
    type: string
    description: A read-only UUID, generated by the system, to uniquely identify the asset record.

```

.\1-Schemas\A-entity_record_schema\01_asset.yml

```
  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 functional-location. An example is an aeration tank.

  class:
    $ref: "../B-entity_class_object_schema/01_asset_item_tool_class.yml"
    description: indicates the class to which this asset is an instance.

  GIS_object_ID:
    type: string
    integration: GIS
    description: is unique ID of the GIS asset record, which represents the same asset as this record.

  specification_data:
    type: object
    description: is a set of specification property data. (The applicable properties are dependent on the definition made in the chosen class.)

  inferred_classes:
    oneOf:
      - type: null
      - type: array
        items:
          $ref: "../B-entity_class_object_schema/01_asset_item_tool_class.yml"
    read-only: true
    integration: data-hub
    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.

# 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 functional-location, 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 functional-location, org-chart_group, or anyone.

# OWNER, OPERATOR, MAINTAINER GROUP

  owned_by_the_group:
    oneOf:
      - type: null
      - $ref: "./04_org-chart_group.yml"
    description: Denotes the org-chart_group that owns the asset.
    integration: GIS

  owned_by_the_unlisted_group:
    oneOf:
    - type: null
    - type: string
    description: name of an org-chart group that is not found in the current list (and should be added)
    integration: GIS
    $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 org-chart_group that owns it had not been added to the value list.

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

  operator_group:
    oneOf:
      - type: null
      - $ref: "./04_org-chart_group.yml"
    description: group responsible for the operation of the asset

# ASSIGNMENTS

  assignment_type:
```

```
  oneOf:
   - type: string
   - type: null
  description:
  enum:
   - to a functional-location
   - 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_asset_functional-location", "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.

 assigned_to_asset_functional-location:

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

 assigned_to_tool_user_group:
  oneOf:
   - $ref: "./04_org-chart_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 functional-location.

 assigned_to_tool_user:
  oneOf:
   - $ref: "./08_person.yml"
   - type: null
  description: Indicates the assignment of an asset (usually a tool) that does not have a system functional-location.

# LOCATION

 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
```

```yaml

  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 functional-location 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

  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

  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.

  is_mobile:
    oneOf:
      - type: boolean
      - type: null

  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.

  tool_product_master_record:
    oneOf:
```

```yaml
   - $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.

# MANUFACTURER AND MODEL

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

# DATE PROPERTIES

 construction_contract_number:
  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
        items:
          oneOf:
            - type: null
            - type: array
              items:
                $ref: "./00_common_definitions.yml/attachment_def"
      - type: null

# COMMON RECORD PROPERTIES

  duplicate_record_of:
    $ref: "./00_common_definitions.yml#/definitions/duplicate_record_def"

  record_retirement_information:
    $ref: "./00_common_definitions.yml#/definitions/record_retirement_def"

# 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.

##############################################################
# RULES
##############################################################

rule_spec:

  - name: Vertical Asset ID
    spec_ID: 41JeoQuvex
    involves_prop: [ID]
    type: [assertion]
    specification: |
```

```yaml
      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: Asset Naming
    spec_ID: 4ykh0m_Dle
    involves_prop: [name]
    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_asset_functional-location".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

  - name: Exclusion Of Part And Material Classes From Asset Classification
    involves_prop: [class]
    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

  - name: Valid Values of owned_by_the_group Property
    involves_prop: [owned_by_the_group]
    spec_ID: 410N2dr_xx
    type: [validation,UI]
    specification: |
      - The valid range of values for selection includes the first or second of the org-chart group hierarchy, specified in the (/TWDM/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

  - name: Inheriting The Asset's Maintenance And Operator Group Values From Its functional-location
    involves_prop: [operator_group, maintenance_group]
    spec_ID: VJ1QRgIclg
    specification: |
      - if the value of asset_x.properties.assigned_to_asset_functional-location is functional-location_y, then
          inherit the value of
            - asset_x.properties.maintenance_group
            - asset_x.properties.operator_group
          from the same properties of functional-location_y
    checked_on: 2024-08-20

  - name: Rendering of assignment_type Data Field.
    involves_prop: [assignment_type]
    spec_ID: 4yARRuvOex
    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 functional-location", then
        - asset_x.properties."assigned_to_asset_functional-location" 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_asset_functional-location" 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_asset_functional-location" 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_asset_functional-location" 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

  - name: asset present at site must have location information on record
    involves_prop: [service_address_or_coordinate]
    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

  - name: Automatic Value Assignment to properties."is_a_commercially_available_product"
    involves_prop: [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]

  - name: Default Value of is_a_tool
    involves_prop: [is_a_tool]
    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]

  # - name: If an asset is commercially available but not a tool, then it must have mfr and model information.
  #   spec_ID: VJY43yI9lx
  #   involves_prop: item_product_master_record
  #   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
  involves_prop: [item_product_master_record]
  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

- name: Serial on Mobile Assets
  involves_prop: [is_mobile]
  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

- name: when to enable the tool_product_master_record
  involves_prop: [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: NyFFWlUcll
  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
  involves_prop: [tool_product_master_record]
  spec_ID:
  type:
  specification:
  status: TBS

- name: Linear Asset Id
  type: [assertion]
  spec_ID: Vku-67dDxx
```

```yaml
    involves_prop: [ID]
    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: 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 functional-location, 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_asset_functional-location"
       - asset_x.properties."assigned_to_tool_user"
       - asset_x.properties."assigned_to_tool_user_group"


  - name: Consistency Between 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_asset_functional-location"
       - 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 functional-location, user, or user group should not be operating and therefore would not have an operating_state"

  - name: Consistency Between Asset's Physical Status, 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"
```

.\1-Schemas\A-entity_record_schema\01_asset.yml

```
          - "in possession"
        then the following properties would take on the stated values
          asset_x.properties."operating_state" = "not applicable"
          asset_x.properties."assigned_to_asset_functional-location" = null
          asset_x.properties."assigned_to_tool_user" = null
          asset_x.properties."assigned_to_tool_user_group" = null

  - name: Asset Can Only Be Assigned To A Discrete Asset functional-location
    spec_ID: 4yBXuH8qle
    type: [validation]
    specification: |
      if (asset_x.properties."assigned_to_asset_functional-location") is NOT null
          then
 ↪  (asset_x.properties."assigned_to_asset_functional-location".properties.class.properties."d
 ↪  asset functional-location") = TRUE
    errorMessage: an asset can only be assigned to a discrete asset
 ↪  functional-location

  - name: inheriting the asset location information from its functional-location
    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_asset_functional-location" is NOT null
          and one of the following is true
            -
 ↪  asset_x.properties."assigned_to_asset_functional-location".properties."asset_installation_
 ↪  is NOT null
            -
 ↪  asset_x.properties."assigned_to_asset_functional-location".properties."service_address_or_
 ↪  is NOT null
        then
            (asset_x.properties."installation_or_parking_location") would be set to
 ↪  the value of

 ↪  (asset_x.properties."assigned_to_asset_functional-location".properties."asset_installation
            (asset_x.properties."service_address_or_coordinate") would be set to the
 ↪  value of

 ↪  (asset_x.properties."assigned_to_asset_functional-location".properties."service_address_or

  - 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_functional-location.yml

```
---
$schema: "http://json-schema.org/draft-07/schema#"
title: functional-location
$id:
 ↪  https://raw.githubusercontent.com/TW-ASMP/TWDM/main/1-Schemas/A-entity_record_schema/02_as
type: object

properties:

  ID:
    type: string
    description: is the unique ID of the functional-location known as the "tag
 ↪  number" or "entity number" in Avantis's vocabulary. (Avantis is the a WMS).

  name:
    type: string
    description: is a structured description of the functional-location.

  parent:
    $ref: "./02_functional-location.yml"
    description: references the functional-location that is served by the larger
 ↪  asset, which physically subsumes the asset serving this functional-location.

  GIS_object_ID:
    type: string
    integration: GIS
    description: is unique ID of the GIS record representing the same pumping
 ↪  station, chamber, or ... as this *functional-location* record in Maximo.

  class:
    $ref: "B-entity_class_object_schema/02_functional-location_classification.md"
    description: denotes the broad types of asset that may play the
 ↪  functional-location (e.g., "breaker functional-location"). It also denotes the
 ↪  useful function provided by an asset in the functional-location to the larger
 ↪  system (e.g., the "tie-breaker functional-location" provides tie-breaking
 ↪  function).

  inferred_classes:
    oneOf:
      - type: null
      - type: array
        items:
          $ref: "B-entity_class_object_schema/02_functional-location_classification.md"
    items:
      type: string
    read-only: true
    description: indicates the complex classes to which this asset is an instance.

  specification_data:
    type: object
    $ref: "."
    description: is a set of functional performance specification data. (The applicable specifications are dependent on the definition made in the chosen class.)

  functional-location_status:
    type: string
    enum:
      - specified
      - active
      - eliminated
    description: is the life-cycle status of a functional-location.
    $comment: |
      "specified" means the functional-location is conceived and exists in some specification or design documentation; "active" means the necessary supports exist for an asset to serve in the functional-location and function of the asset being utilized; "eliminated" represents a negation of either or both conditions of the active status.

  installation_location:
    oneOf:
      - $ref: "./03_space.yml"
      - type: null
    description: refers to the physical-location in which the asset serving the functional-location would be installed.

  service_relations:
    description: identifies the asset, by the functional-location it is occupying, that this functional-location is serving. For example, given a motor starter functional-location, the value in this data field identifies the functional-location of the motor controlled by that motor starter.
    oneOf:
    - type: null
    - type: object
      properties:

        serving:
          $ref: "./02_functional-location.yml"

        service_type:
          oneOf:
          - type: null
```

.\1-Schemas\A-entity_record_schema\02_functional-location.yml

```yaml
     - type: string
      enum:
       - "powers"
       - "supplies material"
       - "supplies energy"
       - "controls or regulates"
      $comment: the list is not exhaustive.

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

# OWNER, OPERATOR, MAINTAINER GROUP

 operator_group:
  oneOf:
   - type: null
   - $ref: "./04_org-chart_group.yml"
  description: group responsible for the operation of the asset in the functional-location.

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

 #note:
  # on: the absence of of a owned_by_the_group property for functional-locations
  # content: assets that are occupying a functional-location are presumed to be owned by the City.

 inherit_operator_group_value:
  type: boolean
  default_value: true

 inherit_operator_group_from_parent:
  type: boolean
  default_value: true

 inherit_maintenance_group_from_parent:
  type: boolean
  default_value: true

 operational_criticality:
  oneOf:
   - $ref: "#/definitions/criticalityRatingDef"
   - type: null
  description: A functional-location bears high operational criticality if the loss of the asset in the functional-location will either reduce throughput or product quality (but not product safety) of the larger system.

  protective_function_criticality:
    oneOf:
      - $ref: "#/definitions/criticalityRatingDef"
      - type: null
    description: A functional-location 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.

# COMMON RECORD PROPERTIES

  duplicate_record_of:
    $ref: "./00_common_definitions.yml#/definitions/duplicate_record_def"

  record_retirement_information:
    $ref: "./00_common_definitions.yml#/definitions/record_retirement_def"


#############################################################
# 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


#############################################################
# RULES
#############################################################

rule_spec:

  - name: ID of Linear Assets Represented as functional-location in Maximo
    spec_ID: Vku-67dDxx
    involves_prop: [ID]
    type: [triggered action]
    specification:
      trigger: replication creation of assetY record from the TWAG
      action: apply Facility ID from TWAG as ID
    status: false

  - name: ID of Linear Assets Represented as functional-location in Maximo
    spec_ID: Vku-67dDxx
    involves_prop: [ID]
    type: [triggered action]
    specification:
      - if:
        oneOf:
          - assetfunctional-locationClass:
              properties:
                  className:
                    const: pumping station
          - assetfunctional-locationClass:
              properties:
                  className:
                    const: water treatment plant
          - assetfunctional-locationClass:
              properties:
                  className:
                    const: large chamber
        then:
          required: GIS_object_ID
    status: false

    #[]RULE VkiDyJcSxg: Before a functional-location can be eliminated, all children, as well as the descendants of the functional-location in the hierarchy must also be eliminated. $comment: a procedure should to be created to allow the recursive elimination of a functional-location and all of its children.

    #[]RULE VygDCOFrxl: When a functional-location is "eliminated", it must no longer be visible in any view of the functional-location 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 functional-locations.

    #[]RULE EkP5qy5Sxl: If change auditing cannot be turned on, then when a functional-location record status is "eliminated", all the specification in the record's data fields must be frozen.

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

.\1-Schemas\A-entity_record_schema\03_physical-location.yml

03_physical-location.yml

```yaml
---
$schema: "http://json-schema.org/draft-07/schema#"
title: physical-location
$id:
↪ https://raw.githubusercontent.com/TW-ASMP/TWDM/main/1-Schemas/A-entity_record_schema/03_sp
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 physical-location that completely contains this
↪   physical-location.

  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
↪   physical-location.


  enclosed_by_asset:

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

.\1-Schemas\A-entity_record_schema\03_physical-location.yml

```
  - equipment cabinet,
  - vehicle

class:

 $ref: "../B-entity_class_object_schema/03_space_class.yml"

specification_data:

 type: object
 description: is a set of specification property data. (The applicable properties are dependent on the definition made in the chosen class.)

inferred_classes:

 oneOf:
  - type: array
    items:
     $ref: "../B-entity_class_object_schema/03_space_class.yml"
  - type: null
 read-only: TRUE

service_address_or_coordinate:

 oneOf:
  - $ref: 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 physical-location is merely specified, or whether the boundary enclosing the physical-location have been contructed (or alternatively, the fiat property / or area boundary around is formally established and approved.) - i.e. "exists".
  Note that an "eliminated" physical-location 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 physical-location disappear by the fact that object bound or defined the physical-location, 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:

    type: string
    enum:
      - 1
      - 2

  hazardous location group:

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

  # COMMON RECORD PROPERTIES


  duplicate_record_of:

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

  record_retirement_information:

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

```

04_org-chart_group.yml

```
---
$schema: "http://json-schema.org/draft-07/schema#"
title: org-chart group
$id:
 ↪  https://raw.githubusercontent.com/TW-ASMP/TWDM/main/1-Schemas/A-entity_record_schema/04_or
type: object
```

.\1-Schemas\A-entity_record_schema\04_org-chart_group.yml

```yaml
properties:

  ID:

    type: string
    description: is the unique ID

  org-chart_group_name:

    type: string

  parent_org-chart_group:

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

  class:

    $ref: B-entity_class_object_schema/04_org_class.yml

  specification_data:

    type: object
    description: is a set of specification property data. (The applicable properties are dependent on the definition made in the chosen class.)

  leader:

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

  equivalent_to_Maximo_site:

    description: indicates that this org-chart group maps to a particular site (a native Maximo object)
    $ref: MaximoSiteObject

  equivalent_to_Maximo_org:

    description: indicates that this org-chart group maps to a particular org-chart group (a native Maximo object)
    $ref: MaximoOrgObject

  equivalent_to_Maximo_crew:

    description: indicates that this org-chart group maps to a particular org-chart group (a native Maximo object)
    $ref: MaximoCrewObject

  # COMMON RECORD PROPERTIES


  duplicate_record_of:

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

  record_retirement_information:

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



```

05_item_master.yml

```
---
$schema: "http://json-schema.org/draft-07/schema#"
title: item master
$id: https://raw.githubusercontent.com/TW-ASMP/TWDM/main/1-Schemas/A-entity_record_schema/05_it
type: object

properties:

  ID:

    description: A read-only UUID, generated by the system, to uniquely identify the item.
    type: string
    implementer_note: |
      Use UUID instead of a serial - this allow us to incorporate future items defined outside of TW.

  description:

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

 description: indicates the class to which this asset is an instance.
 $ref: "../B-entity_class_object_schema/01_asset_item_tool_class.yml"
 $comment: A value from the item classification (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

rotating:

 description: Signifies that instances of this item are tracked as serialized individuals.
 $ref: "./00_common_definitions.yml#/definitions/rotating_property_def"

material_item:

 description: indicates that the item is a material item, which are handled in continuous quantities.
 type: boolean
 $comment: |
  Concept of material: materials are often handled in continuous quantities. For example, you might purchase a certain length of piping or a volume of concrete, and you have to measure and cut or shape these materials to fit your project's requirements.

# GENERIC COMMODITY AND COMMERCIAL PRODUCT DEFINITION


commodity_or_commercial_product:

 description: Indicate whether the item master defines an unspecialized commodity or a specific commercial product of a certain manufacturer.
 type: string
 enum:
  - commodity
```

```
    - commercial product

# GENERIC COMMODITY DESCRIPTION


supplementary_commodity_description:

  description: Supplementary description, in addition to the class value and class-dependent specification values, necessary to differentiate a commodity.
  type: string
  $comment: Toronto Water is not using the UNSPSC or any other commodity taxonomy code for item identification, as Toronto Water's classification system plays the same functional-location and can be mapped to other taxonomies.


# COMMERCIAL PRODUCT DESCRIPTION


commercial_product_definition:

  description: Points to the commercial product definition.
  oneOf:
    - type: null
    - $ref: "./00_common_definitions.yml#/definitions/manufacturer_and_model_def"

commercial_product_description:

  description: A description concatenated from the text components of the commercial product.
  oneOf:
    - type: null
    - type: string
  rule_spec:
    - name: Concatenation of commercial product description
      status: TBD

instant_of_generic_commodity:

  description: Indicates the commercial product is also a type of generic commodities.
  oneOf:
    - type: null
    - type: array
      items:
        $ref: "./05_item_master.yml"

# OTHER ITEMS DESCRIPTIONS


item_format:

  description: Describes the format of the individual units (forming the inventory count)- for example, "can" "functional-location", "sheet", "object", "box", "bag",....
```

```yaml
    type: string

  specification_data:

    type: object
    description: is a set of specification property data. (The applicable properties are dependent on the definition made in the chosen class.)

  cost_in_CAD:

    description: The expected cost of an each unit of the item.
    type: number

  unit_of_issue:

    description: Describes how the quantity of the item is measured, when it is issued out, such as "feet", "kg", "sheet".
    type: string
    rule_spec:
      - name: Unit of issue for assets and parts (non-material items)
        spec_ID: 01JK1VER5T6HK314XPB4W5T27V
        type: validation
        form: short
        description: if the item is not an material item, its unit of issue would be "individual item"
      - name: Default value of  unit_of_issue
        spec_ID: 01JK7AM2RA8S8EPAH57W33SJ79
        type: assertion
        form: short
        description: When an item record is created, the default unit of issue is the same as the item format.

  quantity_in_units_of_issue:

    description: indicates how many units of issue is in the item.
    type: number
    rule_spec:
      - name: Default quantity in units
        spec_ID: 01JK7AFSHFQ2W8G5JD3B9XSWBV
        type: assertion
        form: short
        description: When an item record is created, the default value of quantity_in_units_of_issue is 1.

  alternate_format_of_same_item:

    description: |
      Usually identifies the same commercial product item made by the same manufacturer, but differing only in the format. For example, the 208-litre drum item and the 5-litre bottle item of Penzoil 5W30 Synthetic Lubricant.
    oneOf:
      - type: null
      - type: array
        items:
          $ref: "./04_item_master.yml"
    rule_spec:
      - name: Range must be a commercial product as well
        spec_ID: 01JF81079K178X9B4NSG23AA0Z
        type: validation
        specification:
        status: TBD

  ordering_options:

    description: presents a list of vendors and available order formats
    oneOf:
      - type: null
      - type: array
        items:
          type: object
          properties:

            vendor:

              $ref: MaximoCompanyObject

            unit_of_order:

              type: object
              properties:

                description:

                  description: a description the packages of individual units such as "box of 24" or "individual unit"
                  type: string

                number_in_package:

                  description: indicate the number of items that are in the package
                  type: number

              rule_spec:
                - name: Default order packaging format
                  description: Default order packaging format is "individual unit"
                  form: short
                  spec_ID: 01JK1KEZSR9ASBXMWE406TCBPN

            vendor_item_number:

              oneOf:
                - type: string
                - type: null

            cost_in_CAD:

              description: The expected of the item in the order format
              type: number
              $comment: To TW, in the future, this field should contain a running average of the recent purchase costs, possibly also adjusted for recent inflation.

            contracts:

              description: the existing contracts that can be used for order the product
              $ref: MaximoContractObject

  # COMMON RECORD PROPERTIES


  duplicate_record_of:

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

  record_retirement_information:

    $ref: "./00_common_definitions.yml#/definitions/record_retirement_def"
```

06_tool_master.yml

```
---
$schema: "http://json-schema.org/draft-07/schema#"
title: tool item master
$id: https://raw.githubusercontent.com/TW-ASMP/TWDM/main/1-Schemas/A-entity_record_schema/06_to
type: object

properties:

  ID:
    description: A read-only UUID, generated by the system, to uniquely identify the tool.
    type: string
    implementer_note: |
      Use UUID instead of a serial - this allow us to incorporate future 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.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: to be updated

 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

 specification_data:
  type: object
  description: is a set of specification property data. (The applicable properties are dependent on the definition made in the chosen 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.

```

```
# COMMON RECORD PROPERTIES
#==============================

  duplicate_record_of:
    $ref: "./00_common_definitions.yml#/definitions/duplicate_record_def"

  record_retirement_information:
    $ref: "./00_common_definitions.yml#/definitions/record_retirement_def"

$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

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

properties:

  # COMMON RECORD PROPERTIES

  duplicate_record_of:

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

  record_retirement_information:

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

# Rest to be specified in the future
```

08_person.yml

```
$schema: http://json-schema.org/draft-07/schema#
title: Person
$id:
 ↪  https://raw.githubusercontent.com/TW-ASMP/TWDM/main/1-Schemas/A-entity_record_schema/08_pe
type: object

properties:

  employee_number:
    type: string

```

.\1-Schemas\A-entity_record_schema\08_person.yml

```yaml
  first_name:
    type: string

  last_name:
    type: string

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

  section:
    description: Indicates which section within a division that the person works for
    type: string
    oneOf:
      - type: null
      - type: string

  unit:
    description: Indicates which business-unit within a section that the person works for
    oneOf:
      - type: null
      - type: string

  status:
    type: string
    enum:
      - active
      - inactive
      - departed

  person_roles:

    description: The role, associated with a certain type of system access, that the person plays in the system.
    oneOf:
      - type: null
      - type: array
        items:
          $ref: MaximoSecurityRole

  # NOTE: persons' association to trade in documented in the trade object

  qualifications:

    description: The qualifications that the person gained through training or certification.
    oneOf:
      - type: null
      - type: array
        items:
```

.\1-Schemas\A-entity_record_schema\09_qualification.yml

```yaml
        type: object
        properties:

          qualification:
            $ref: "./09_qualification.yml"

          start_date:
            description: the first effective day
            oneOf:
              - type: null
              - type: string
                format: date

          expiration_date:
            description: the last effective date
            oneOf:
              - type: null
              - type: string
                format: date

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

contract:
  description: identifies the contracting company that the person works for
  oneOf:
    - type: null
    - type: object
      $ref: "./13_service_contract.yml"

contact:

  properties:

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

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

# COMMON RECORD PROPERTIES

duplicate_record_of:
  $ref: "./00_common_definitions.yml#/definitions/duplicate_record_def"

record_retirement_information:
  $ref: "./00_common_definitions.yml#/definitions/record_retirement_def"
```

.\1-Schemas\A-entity_record_schema\10_trade.yml

09_qualification.yml

```yaml
$schema: http://json-schema.org/draft-07/schema#
title: Qualification
$id:
 ↪ https://raw.githubusercontent.com/TW-ASMP/TWDM/main/1-Schemas/A-entity_record_schema/09_qu
type: object

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

  status:
    type: string
    enum:
      - active
      - inactive

  type:
    type: string
    enum:
    - professional license
    - skill certification
    - record of training

  issuing_organization:
    oneOf:
      - type: null
      - $ref: "./04_org-chart_group.yml"

# COMMON RECORD PROPERTIES

  duplicate_record_of:
    $ref: "./00_common_definitions.yml#/definitions/duplicate_record_def"

  record_retirement_information:
    $ref: "./00_common_definitions.yml#/definitions/record_retirement_def"

```

10_trade.yml

```yaml
$schema: http://json-schema.org/draft-07/schema#
title: Trade
```

.\1-Schemas\A-entity_record_schema\10_trade.yml

```yaml
$id:
 ↪  https://raw.githubusercontent.com/TW-ASMP/TWDM/main/1-Schemas/A-entity_record_schema/10_tr
type: object

properties:

  code:
    description: Unique identifier for the craft
    type: string

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

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

  skill_levels:
    type: array
    description: List of skill levels within this craft
    items:
      type: object
      properties:

        skill_level:
          type: string
          description: Skill level associated with the craft (e.g., APPRENTICE,
 ↪  JOURNEYMAN, MASTER)

        hourly_rate:
          description: Standard hourly rate for this skill level
          type: number

        qualifications:
          description: qualifications required to hold
          oneOf:
            - type: null
            - type: array
              items:
                type: object

  association_to_persons:
    description: The primary craft or skill associated with the person, if the
 ↪  person plays the role of a trade person.
    oneOf:
      - type: null
      - type: array
        items:
          type: object
          properties:

            trade:
              $ref: "./10_trade.yml"

            skill_level:
              type: string
              $comment: load the list of skilled trades

  status:
    type: string
    enum:
      - active
      - inactive

# COMMON RECORD PROPERTIES

  duplicate_record_of:
    $ref: "./00_common_definitions.yml#/definitions/duplicate_record_def"
```

11_warranty.yml

```yaml
$schema: http://json-schema.org/draft-07/schema#
title: Warranty
$id: https://raw.githubusercontent.com/TW-ASMP/TWDM/main/1-Schemas/A-entity_record_schema/12_wa
type: object

properties:

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

  description:
    description: 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: the first effective day of the warranty
    oneOf:
      - type: null
      - type: string
        format: date

  warranty_expiration_date:
    description: the last effective date of the warranty
    oneOf:
      - type: null
      - type: string
        format: date

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

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

  covers_asset_functional-locations:
    description: the list of functional-locations or more specifically the assets installed in the functional-locations covered by the warranty
    oneOf:
      - type: null
      - type: array
        items:
          $ref: "./02_functional-location.yml"

  covers_assets:
    description: the list of assets covered by the warranty (any parts in the inventory covered by the warranty must be associated with an asset record)
    oneOf:
      - type: null
      - type: array
        items:
          $ref: "./01_asset.yml"

  contract_document:
    description: the PDF version of the contract document.
    oneOf:
      - type: null
      - type: array
        items:
          $ref: "./00_common_definitions.yml/attachment_def"

# COMMON RECORD PROPERTIES

  duplicate_record_of:
    $ref: "./00_common_definitions.yml#/definitions/duplicate_record_def"

  record_retirement_information:
    $ref: "./00_common_definitions.yml#/definitions/record_retirement_def"
```

12_service_contract.yml

```yaml
$schema: http://json-schema.org/draft-07/schema#
title: Service Contract
$id: https://raw.githubusercontent.com/TW-ASMP/TWDM/main/1-Schemas/A-entity_record_schema/13_se
```

.\1-Schemas\A-entity_record_schema\12_service_contract.yml

```yaml
type: object

properties:

  ID:
    description: the unique identifier for the contract
    type: string

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

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

  start_date:
    description: the first effective day
    oneOf:
      - type: null
      - type: string
        format: date

  expiration_date:
    description: the last effective date
    oneOf:
      - type: null
      - type: string
        format: date

  covers_asset_functional-locations:
    description: the list of functional-locations or more specifically the assets installed in the functional-locations covered by the contract
    oneOf:
      - type: null
      - type: array
        items:
          $ref: "./02_functional-location.yml"

  covers_assets:
    description: the list of assets covered by the contract (any parts in the inventory covered by the contract must be associated with an asset record)
    oneOf:
      - type: null
      - type: array
        items:
          $ref: "./01_asset.yml"

  covers_work:
    description: identifies work (job plans) the contractor would perform.
    oneOf:
      - type: null
      - type: array
```

```
        items:
          $ref: "./32_job_plan.yml"

  contract_document:
    $ref: "./00_common_definitions.yml#/definitions/attachment_def"

  # COMMON RECORD PROPERTIES

  duplicate_record_of:
    $ref: "./00_common_definitions.yml#/definitions/duplicate_record_def"

  record_retirement_information:
    $ref: "./00_common_definitions.yml#/definitions/record_retirement_def"
```

32_job_plan.yml

```
---
$schema: http://json-schema.org/draft-07/schema#
title: Job Plan
$id:
 ↪  https://raw.githubusercontent.com/TW-ASMP/TWDM/main/1-Schemas/A-entity_record_schema/32_jo
description: |
  A job plan is the the lowest level of work description to contain the full
 ↪  planning specifications.
type: object

properties:

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

  name:
    description: 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
```

```yaml
      - type: array
        items:
          $ref: "../B-entity_class_object_schema/01_asset_item_tool_class.yml"

  specific_to_asset_functional-location_classes:
    description: identifies the functional-location classes - more specifically, the assets installed in these functional-locations - for which this job plan is customized.
    type:
      oneOf:
      - type: null
      - type: array
        items:
          $ref: "../B-entity_class_object_schema/02_functional-location_classification.md"
    $comment: >
      Examples of a functional-location class include the tie-breaker and effluent turbidity meter functional-location class.

  specific_to_operational_units:
    description: identifies the Toronto Water site(s), defined as a org-chart group 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 org-chart group
        spec_ID: 01JD2V5X97J1Y45JWDW4SV1FJ4
        type: [validation]
        specification: |
          Given an job plan, JP_x, all values of JP_x.specific_to_operational_units must be
            - an org-chart group, and
            - whose .class = unit (a subclass of Group in the City)

  specific_to_asset_functional-locations:
    description: identifies the functional-locations for which the job plan is specifically customized.
    type:
      oneOf:
      - type: null
      - type: array
        items:
          $ref: "../A-entity_record_schema/02_functional-location.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: PM library

  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 functional-location, mitigated by the work specified in this job plan.
    oneOf:
      - type: null
      - type: array
        items:
          type: string
    integration: PM Library
    $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

```

```
  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:
    type: boolean

# 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
```

```yaml
              - 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

# COMPLIANCE INFORMATION

  compliance_requirement:
    description: identifies the compliance requirement object
    integration: data-hub
    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

# COMMON RECORD PROPERTIES

  duplicate_record_of:
    $ref: "./00_common_definitions.yml#/definitions/duplicate_record_def"

  record_retirement_information:
    $ref: "./00_common_definitions.yml#/definitions/record_retirement_def"
```

33_PM.yml

```
---
$schema: http://json-schema.org/draft-07/schema#
title: PM
$id:
 ↪  https://raw.githubusercontent.com/TW-ASMP/TWDM/main/1-Schemas/A-entity_record_schema/33_PM
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 org-chart group
 ↪  hierarchy, such as Waste and Storm Water Pumping (symbol - WASP) or Humber
 ↪  Wastewater Treatment Plant (symbol - THR).
    $ref: "./04_org-chart_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-chart_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-chart_group.yml"

  comment: This value will be determined by (and must be consistent with) the
↪ maintainer_org-chart_group value - situated at a lower level of the org-chart
↪ group 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)

 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

 functional-location_to_work_on:
```

```yaml
    description: indicates functional-location at which the specified must be performed.
    oneOf:
      - type: null
      - $ref: "../A-entity_record_schema/02_functional-location.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/functional-location.
    oneOf:
      - type: null
      - type: object
        properties:
          sequence:
            type: number
          asset:
            oneOf:
              - type: null
              - $ref: "./01_asset.yml"
          functional-location:
            oneOf:
              - type: null
              - $ref: "./02_functional-location.yml"
          job_plan:
            oneOf:
              - type: null
              - $ref: "./32_job_plan.yml"
    $comment: the implementation could be done with Maximo route object.

# RESOURCES

  #Note:
    # content: 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

  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(_)


  # COMMON RECORD PROPERTIES

  duplicate_record_of:
    $ref: "./00_common_definitions.yml#/definitions/duplicate_record_def"

  record_retirement_information:
    $ref: "./00_common_definitions.yml#/definitions/record_retirement_def"

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

definitions:

  time-based_trigger_specification:
    next_due_date:
      type: string
      format: date
    frequency_interval:
      $ref: "./00_common_definitions.yml#/definitions/frequency_interval_definition"

  meter-based_trigger_specification:
    reading_at_last_service_completion:
      type: number
      description: is the last recorded meter reading when service was completed.
    next_service_trigger_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/TWDM/main/1-Schemas/A-entity_record_schema/34_FR

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

properties:

  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

      $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

  functional-location_to_work_on:

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

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

```yaml

 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_asset_functional-location_criticality:

   description: is the criticality values inherited from the asset or functional-location 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/functional-location.
   oneOf:
     - type: null
     - type: object
       properties:

         sequence:
           type: number

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

         functional-location:
           oneOf:
             - type: null
             - $ref: "./02_functional-location.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

```

```yaml
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:
    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
```

```yaml


  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 org-chart group'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 org-chart group hierarchy, such as Waste and Storm Water Pumping (symbol - WASP) or Humber Wastewater Treatment Plant (symbol - THR).
    $ref: "./04_org-chart_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-chart_group.yml"

    rule_spec:
      name: Inherit the maintenance_group value from either the asset or the functional-location
      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:
      - 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-chart_group.yml"

  WO_only: true

  comment: This value will be determined by (and must be consistent with) the maintainer_org-chart_group value - situated at a lower level of the org-chart group 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 functional-location) 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.
  type: string
  format: date

  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
```

```yaml

asset_covered_by_service_contract:

  description: indicates that the asset (or the asset in the functional-location) 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: string
        format: date

  work_requested_date:

    description: is the date when the work request was submitted.
    oneOf:
      - type: null
      - type: string
        format: date

  request_approval_date:

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

  target_start_date:

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

  target_completion_date:

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

  scheduled_start_date:

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

  scheduled_completion_date:

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

  actual_start_date:

    description: is the date when the work actually began.
    oneOf:
      - type: null
      - type: string
        format: date

  actual_completion_date:

```

```yaml
    description: is the date when the work was actually completed.
    oneOf:
      - type: null
      - type: string
        format: date

  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: string
        format: date

  WO_closing_date:

    description: is the date when the work order was closed.
    oneOf:
      - type: null
      - type: string
        format: date


  # 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:
                  $ref: "./00_common_definitions.yml/attachment_def"

  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
      - type: array
        items:
          $ref: "./00_common_definitions.yml/attachment_def"

    implementation:
      MX_mapping:


  # COMMON RECORD PROPERTIES

  duplicate_record_of:

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

  record_retirement_information:

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

```

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

36_work_order_documentation.yml

```yaml
---
$schema: http://json-schema.org/draft-07/schema#
title: Work Order Documentation
$id: https://raw.githubusercontent.com/TW-ASMP/TWDM/1-Schemas/A-entity_record_schema/36_work_do
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:
    # content: 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:
    description: the date the work started
    oneOf:
      - type: null
      - type: string
        format: date

  actual_completion_time:
    description: the date the work was actually completed
    oneOf:
      - type: null
      - type: string
```

```
        format: date

  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:
    description:
    oneOf:
```

```yaml
      - 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

# COMMON RECORD PROPERTIES

  duplicate_record_of:
    $ref: "./00_common_definitions.yml#/definitions/duplicate_record_def"
```

41_meter.yml

```yaml
$schema: "http://json-schema.org/draft-07/schema#"
title: Maximo Meter
description: A JSON Schema representing a Meter object in IBM Maximo.
$id: https://raw.githubusercontent.com/TW-ASMP/TWDM/1-Schemas/A-entity_record_schema/41_meter.y
type: object
properties:

  Id:
    type:
      - string
      - "null"
    description: Unique identifier for the meter record.

  description:
    type:
      - string
      - "null"
    description: optional description or notes about the meter.

  asset:
    oneOf:
      - $ref: "./01_asset.yml"
      - type: null
    description: identifier of the asset associated with this meter.

  functional-location:
    oneOf:
      - $ref: "./02_functional-location.yml"
      - type: null
    description: identifier of the functional-location associated with this meter.

  meter_name:
    type:
      - string
      - "null"
    description: name of the meter.

  meter_type:
    type:
      - string
      - "null"
    enum:
      - HOURS
      - CYCLES
      - MILES
      - KILOMETERS
      - ...
    description: type of meter measurement.
    $comment: the enum values is not meant to be comprehensive

  unit_of_measure:
    type:
      - string
      - "null"
    description: unit of measure for the meter (e.g., hours, cycles).

  current_reading:
    type:
      - number
      - "null"
    description: is the current meter reading.

  last_reading:
    type:
      - number
      - "null"
    description: is the latest know meter reading

# COMMON RECORD PROPERTIES

  duplicate_record_of:
    $ref: "./00_common_definitions.yml#/definitions/duplicate_record_def"

  record_retirement_information:
    $ref: "./00_common_definitions.yml#/definitions/record_retirement_def"
```

Folder: B-entity_class_object_schema

00_common_class_definitions.yml

```
---
$schema: "http://json-schema.org/draft-07/schema#"
title: generic class object
```

.\1-Schemas\B-entity_class_object_schema\01_asset_item_tool_class.yml

```yaml
$id: https://raw.githubusercontent.com/TW-ASMP/TWDM/main/1-Schemas/B-entity_class_object_schema
type: object

# This set of properties are used in all classification objects (e.g., asset, org-chart group, etc.)

properties:

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

  class_description:
    oneOf:
      - type: null
      - type: string

  synonym_names:
    oneOf:
      - type: null
      - type: array
        items:
          type: string
    description: other synonymous names that may be used by a user in search

  can_be_applied_to_instances:
    type: boolean
    $comment: |
      a "FALSE" value indicates that the class is meant to be a structural part of the classification tree, and cannot be used to classify any entity (i.e. asset, functional-location, physical-location, etc).

  # COMMON RECORD PROPERTIES

  duplicate_record_of:

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

  record_retirement_information:

    $ref: "./00_common_definitions.yml#/definitions/record_retirement_def"
```

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/TWDM/main/1-Schemas/B-entity_class_object_schema
type: object
```

```

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

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: B-entity_class_object_schema/01_asset_item_tool_class.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_functional-location_class.yml

```
---
$schema: "http://json-schema.org/draft-07/schema#"
title: functional-location classification object
$id:
 ↪  https://raw.githubusercontent.com/TW-ASMP/TWDM/main/1-Schemas/B-entity_class_object_schema
type: object

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

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: B-entity_class_object_schema/02_functional-location_class.yml
    sort order: 1-30

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

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

  defined_set_of_functional-locations:
    oneOf: [type: boolean, type: null]
    read-only: true
    description: a functional-location that can be occupied by a collection of discrete assets.

  compatible_asset_occupant_classes:
    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 functional-location under this class.

##############################################################
# RULES
##############################################################

rule_spec:
  - name: Is an functional-location, functional structure functional-location, or defined set of functional-locations
    spec_ID: NyD4XGbuex
    specification: |
      if functional-locationClassX is a descendent of "Discrete Asset functional-location" in the functional-location classification hierarchy:
        set functional-locationClassX.properties."a discrete asset functional-location" to TRUE
        set functional-locationClassX.properties."a functional structure functional-location" to FALSE
        set functional-locationClassX.properties."a defined ser of functional-location" to FALSE
      elif functional-locationClassX is a descendent of "functional structure functional-location" in the functional-location classification hierarchy:
        set functional-locationClassX.properties."a discrete asset functional-location" to FALSE
        set functional-locationClassX.properties."a functional structure
↪   functional-location" to TRUE
        set functional-locationClassX.properties."a defined ser of
↪   functional-location" to FALSE
      elif functional-locationClassX is a descendent of "Defined Set of
↪   functional-locations" in the functional-location classification hierarchy:
        set functional-locationClassX.properties."a discrete asset
↪   functional-location" to FALSE
        set functional-locationClassX.properties."a functional structure
↪   functional-location" to FALSE
        set functional-locationClassX.properties."a defined ser of
↪   functional-location" to TRUE
    status: specified
```

03_physical_location_class.yml

```
title: functional-location classification object
$id:
↪   https://raw.githubusercontent.com/TW-ASMP/TWDM/main/1-Schemas/B-entity_class_object_schema
$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 definitions and rules from th
↪   "/1-Schemas/B-entity_class_object_schema/00_common_class_definitions.yml"

  parent_class:
    $ref: B-entity_class_object_schema/03_space_class.yml
```

04_org-chart_group_class.yml

```
---
$schema: "http://json-schema.org/draft-07/schema#"
title: functional-location classification object
$id:
↪   https://raw.githubusercontent.com/TW-ASMP/TWDM/main/1-Schemas/B-entity_class_object_schema
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: B-entity_class_object_schema/04_org_class.yml
```

32_discrete_activity_class.yml

```
---
title: discrete activity classification object
$id: https://raw.githubusercontent.com/TW-ASMP/TWDM/main/1-Schemas/B-entity_class_object_schema
$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 definitions and rules from th "/1-Schemas/B-entity_class_object_schema/00_common_class_definitions.yml"

  parent_class:
    $ref: B-entity_class_object_schema/32_discrete_activity_class.yml
```

33_work_type_class.yml

```
title: work type object
$id: https://raw.githubusercontent.com/TW-ASMP/TWDM/main/1-Schemas/B-entity_class_object_schema
$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 definitions and rules from th "/1-Schemas/B-entity_class_object_schema/00_common_class_definitions.yml"

  parent_class:
    $ref: B-entity_class_object_schema/33_work_type_class.yml
```

Folder: 2-Classification_Trees

01_asset_classification.md

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

```

02_functional-location_classification.md

```
## Top-Level functional-location Classification

The following is a sample of the Toronto Water's functional-location classification.

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

```

.\2-Classification_Trees\04_org-chart_group_classification.md

```

## Requirements for Implementer

* []REQ Nyh7RPjEgl #IMP "classes names specified in title-case and with an asterisk
 ↪  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_physical-location_classification.md

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

The following is a sample of the Toronto Water's physical-location classification.

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

## Requirements for Implementer

[]REQ Nyh7RPjEgl #IMP "classes names specified in camel-case and with an asterisk
 ↪  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 physical-locations inside and around a
 ↪  structure. []TODO #TW "add to the TWONTO".
```

.\2-Classification_Trees\32_discrete_activity_classification.md

04_org-chart_group_classification.md

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

The following is a sample of the Toronto Water's org-chart 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 asterisk symbol shall have the appliable to individual property, found in the class object, set to false"
```

31_work_type.md

```
## Work Type [^2]

The following is the complete set of Toronto Water's work type classification.

* corrective
* emergency [^1]
* investigative
* preventive
* informational
* 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.
```

.\2-Classification_Trees\32_discrete_activity_classification.md

32_discrete_activity_classification.md

~~~
## Top-Level Discrete Activity Classification

The following is is the complete set of Toronto Water's 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 functional-location
  * removal of existing functional-location
* 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.can_be_applied_to_instances 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 physical-location, 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_functional-location_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: Water Treatment Plants, the entity represents a set of things. In the case of the example - the set of 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_physical-location_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
 ↪ physical-locations within the drinking water treatment plants in TW.

## Notes
```

04_org-chart_group_hierarchy.md

```
# 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

Folder: A-asset_class_properties

01_pump.yml

```
---
$schema: "http://json-schema.org/draft-07/schema#"
title: pump
$id: https://raw.githubusercontent.com/TW-ASMP/TWDM/main/4-Class_Dependent_Specifications/01_pu
type: object

properties:

  pump_type:
    oneOf:
      - type: null
      - type: string
    description: Indicates the pump type
    enum:
      - "dynamic, centrifugal"
      - "dynamic, axial"
      - "reciprocating, piston or plunger"
      - "reciprocating, diaphragm"
      - "rotary, vane"
      - "rotary, piston"
      - "screw pump"
      - "gear pump"

  orientation:
    oneOf:
      - type: null
      - type: string
    description: Indicates how the pump is oriented in 3D physical-location
    enum:
      - "horizontal"
      - "vertical"
      - "angled"
```

.\4-Class_Dependent_Specifications\A-asset_class_properties\01_pump.yml

```yaml
      - "inverted"

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

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

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

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

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

  drive_coupling_type:
    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"

  drive_type:
    oneOf:
      - type: null
      - type: string
    description: Indicates what the mechanically drives the pump
    enum:
      - "electric motor"
      - "engine"

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

```

02_motor.yml

```
---
$schema: "http://json-schema.org/draft-07/schema#"
title: motor
$id:
 ↪ https://raw.githubusercontent.com/TW-ASMP/TWDM/main/4-Class_Dependent_Specifications/02_mo
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"

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

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

  NEMA_frame_type:
    oneOf:
```

```yaml
      - 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"

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

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

```

.\4-Class_Dependent_Specifications\A-asset_class_properties\03_valve.yml

03_valve.yml

```yaml
---
$schema: "http://json-schema.org/draft-07/schema#"
title: valve
$id: https://raw.githubusercontent.com/TW-ASMP/TWDM/main/4-Class_Dependent_Specifications/03_va
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 redundant to label them valve in the valve type

  special_type:
    oneOf:
      - type: null
      - type: string
    description: Indicates what type of specialized valve it is if it is a specialized valve
    enum:
      - "air release valve"
      - "backflow preventer"
      - "check"
      - "pressure relief valve"
      - "solenoid"
    $comment: may be redundant to label them valve in the valve type

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

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

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

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

  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"
```

.\4-Class_Dependent_Specifications\A-asset_class_properties\03_valve.yml

```
      - "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&&"
```

```yaml
      - "447TS&&"
      - "449T"
      - "449TS"
      - "48H"
      - "56H"
      - "56HZ"
      - "L182ACY"
      - "L186AT"

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

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

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

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

  valve_body_type:
    oneOf:
      - type: null
      - type: string
    description: Indicates the valve body material
    enum:
      - "carbon steel"
      - "stainless steel"
      - "duplex"
      - "alloy"
      - "composite"
      - "titanium"

  actuator_type:
    oneOf:
      - type: null
      - type: string
    description: Indicates the type of actuator
    enum:
      - "electric"
      - "pneumatic"
      - "hydraulic"
      - "manual"

  stem_seal_type:
    oneOf:
      - type: null
      - type: string
    description: Indicates the type of stem seal for the valve
    enum:
      - "duplex"
      - "lip seal"
      - "o-ring"
      - "stuffing box"

  valve_turn_direction:
    oneOf:
      - type: null
      - type: string
    enum:
      - "counter-clockwise"
      - "clockwise"
    $comment: This specification is specifically requested by DOS via Silvia Sawada

```

04_breaker.yml

```yaml
---
$schema: "http://json-schema.org/draft-07/schema#"
title: breaker
$id:
 ↪  https://raw.githubusercontent.com/TW-ASMP/TWDM/main/4-Class_Dependent_Specifications/04_br
type: object

$comment: >

properties:

  breaker_type:
    oneOf:
      - type: null
      - type: string
    description: Indicates the breaker type
    enum:
      - "insulated case"
      - "metal clad or enclosed"
      - "molded case"

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

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

  main_contactor_type:
    oneOf:
      - type: null
      - type: string
    description: Indicates the ANSI type for the valve
    enum:
      - "air insulated"
      - "air insulated, air blast"
      - "vacuum insulated"
      - "oil insulated"
      - "gas insulated"


```

05_starter.yml

```yaml
---
$schema: "http://json-schema.org/draft-07/schema#"
title: starter
$id: https://raw.githubusercontent.com/TW-ASMP/TWDM/main/4-Class_Dependent_Specifications/05_st
type: object

properties:

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

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

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

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

  main_contactor_type:
    oneOf:
      - type: null
      - type: string
    description: Indicates the ANSI type for the valve
    enum:
      - "air insulated"
      - "air insulated, air blast"
      - "vacuum insulated"
      - "oil insulated"
      - "gas insulated"

```

06_transformer.yml

```yaml
---
$schema: "http://json-schema.org/draft-07/schema#"
title: transformer
$id: https://raw.githubusercontent.com/TW-ASMP/TWDM/main/4-Class_Dependent_Specifications/06_tr
type: object

$comment: >

properties:

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

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

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

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

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

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

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

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


```

07_HVAC.yml

```
---
$schema: "http://json-schema.org/draft-07/schema#"
title: HVAC
$id:
 ↪  https://raw.githubusercontent.com/TW-ASMP/TWDM/main/4-Class_Dependent_Specifications/07_HV
```

.\4-Class_Dependent_Specifications\A-asset_class_properties\07_HVAC.yml

```yaml
type: object

$comment: >

properties:

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

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

  has_heating_function:
    oneOf:
      - type: null
      - type: boolean
    description: Indicates the presence of a heating function in the HVAC unit such as heating coil or gas burner

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

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

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

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


```

```


```

08_blower_fan.yml

```yaml
---
$schema: "http://json-schema.org/draft-07/schema#"
title: blower_fan
$id: https://raw.githubusercontent.com/TW-ASMP/TWDM/main/4-Class_Dependent_Specifications/08_bl
type: object

$comment: >

properties:

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

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

  drive_coupling_type:
    oneOf:
      - type: null
      - type: string
    description: Indicates the type of drive coupling
    enum:
      - "direct drive"
      - "belt drive"
      - "gear drive"
      - "flexible"
      - "chain drive"
      - "hydraulic"

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


```

.\4-Class_Dependent_Specifications\A-asset_class_properties\10_generator.yml

09_compressor.yml

```yaml
---
$schema: "http://json-schema.org/draft-07/schema#"
title: compressor
$id:
 ↪  https://raw.githubusercontent.com/TW-ASMP/TWDM/main/4-Class_Dependent_Specifications/09_co
type: object

$comment: >

properties:

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

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

  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"

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

10_generator.yml

```yaml
---
$schema: "http://json-schema.org/draft-07/schema#"
title: generator
$id:
 ↪  https://raw.githubusercontent.com/TW-ASMP/TWDM/main/4-Class_Dependent_Specifications/10_ge
```

```yaml
type: object

properties:

  fire_life-safety_elevator:
    oneOf:
    - type: null
    - type: boolean
    description: Indicates if the device directly supports any part of the fire suppression or any life safety systems

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

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

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

  drive_type:
    oneOf:
      - type: null
      - type: string
    description: Indicates the type of drive
    enum:
      - "engine, diesel or bio-diesel"
      - "engine, natural gas"
      - "turbine"

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

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

  drive_coupling_type:
    oneOf:
      - type: null
      - type: string
    description: Indicates the type of drive coupling
    enum:
      - "direct drive"
      - "belt drive"
      - "gear drive"
      - "flexible"
      - "chain drive"
      - "hydraulic"

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

11_UPS.yml

```
---
$schema: "http://json-schema.org/draft-07/schema#"
title: ups
$id: https://raw.githubusercontent.com/TW-ASMP/TWDM/main/4-Class_Dependent_Specifications/11_UP
type: object

$comment: >

properties:

  powers_safety_load:
    oneOf:
      - type: null
      - type: boolean
    description: Indicates if the device directly supports any part of the fire suppression system, life safety system, or elevator of a high building systems

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

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

  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"

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

```

12_boiler.yml

```
---
$schema: "http://json-schema.org/draft-07/schema#"
title: boiler
$id:
 ↪  https://raw.githubusercontent.com/TW-ASMP/TWDM/main/4-Class_Dependent_Specifications/12_bo
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"

  boiled_medium:
    oneOf:
      - type: null
      - type: string
    description: Indicates the boiled/heated medium within the boiler
    enum:
      - water
      - steam
      - glycol
```

```yaml

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

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

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

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

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

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

13_pressure_vessel.yml

```yaml
---
$schema: "http://json-schema.org/draft-07/schema#"
title: pressure_vessel
$id: https://raw.githubusercontent.com/TW-ASMP/TWDM/main/4-Class_Dependent_Specifications/13_pr
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"

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

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

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

14_pressure_piping.yml

```yaml
---
$schema: "http://json-schema.org/draft-07/schema#"
title: pressure_piping
$id: https://raw.githubusercontent.com/TW-ASMP/TWDM/main/4-Class_Dependent_Specifications/14_pr
type: object

$comment: >

properties:

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

.\4-Class_Dependent_Specifications\A-asset_class_properties\15_instrumentation.yml

```yaml

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

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

  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"

  special_application:
    oneOf:
      - type: null
      - type: string
    description: Indicates the special 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"

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

.\4-Class_Dependent_Specifications\A-asset_class_properties\15_instrumentation.yml

15_instrumentation.yml

```
---
$schema: "http://json-schema.org/draft-07/schema#"
title: instrumentation
$id:
 ↪  "https://raw.githubusercontent.com/TW-ASMP/TWDM/main/4-Class_Dependent_Specifications/15_i
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"
      - "chlorination"
      - "chlorine"
      - "dissolved oxygen"
      - "fluoride"
      - "methane/lel"
      - "total hydrocarbon"
      - "orp"
      - "ozone"
      - "particulate"
      - "ph"
      - "sulphide"
      - "sulphur dioxide"
      - "suspended solids"
      - "turbidity"

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

  lower_bound:
    oneOf:
      - type: null
      - type: number
    description: Indicates the lower bound for the parameter of the instrument in
 ↪  the UOM

  upper_bound:
    oneOf:
      - type: null
      - type: number
    description: Indicates the upper bound for the parameter of the instrument in
 ↪  the UOM

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 (°)"
            - "Radians (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 (mmH20)"
            - "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:
```

```yaml
        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 Second (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)"

  - 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: "Chlorination"
    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"
```

```yaml
    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: "Fluoride"
    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:
        parameter:
          const: "PH"
    then:
```

```
    properties:
      uom:
        enum:
          - "Unitless"

- if:
    properties:
      parameter:
        const: "Sulphide"
  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: "Turbidity"
  then:
    properties:
      uom:
        enum:
```

```
      - "Grams Per Milliliter (g/mL)"
      - "Milligrams Per Liter (mg/L)"
      - "Parts Per Million (ppm)"
      - "Parts Per Billion (ppb)"
```

Folder: B-role_class_properties

01_pump_role.yml

```
---
$schema: "http://json-schema.org/draft-07/schema#"
title: pump functional-location
$id: https://raw.githubusercontent.com/TW-ASMP/TWDM/4-Class_Dependent_Specifications/B-function
type: object

properties:

  requires_variable_speed:
    oneOf:
      - type: null
      - type: boolean
    description: indicates the the pump serving in the functional-location must

  required_max_flow:
    oneOf:
      - type: null
      - type: number
    description: indicates the maximum flow rate required to serve the functional-location

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

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

Folder: 5-Functions