

TASTE-Dataview **DEFINITIONS ::=**

BEGIN

IMPORTS T-Int32, T-UInt32, T-Int8, T-UInt8, T-Boolean **FROM** TASTE-BasicTypes;

--declerations of global variables

--AgentNumber T-UInt8

--pathLocationNumber T-UInt8

--nodeNumber T-UInt8

--actionmaxNumber T-UInt8

-- A few simple types to start with ASN.1

MyInteger ::= T-UInt8

MyReal ::= REAL (-1000.0 .. 100000.0)

MyBool ::= BOOLEAN

MyEnum ::= ENUMERATED { hello, world, howareyou }

MySeq ::= SEQUENCE {
 input-data MyInteger,
 output-data MyInteger,
 validity ENUMERATED { valid, invalid }
 }

--User Commands

UserCommand ::= ENUMERATED {sysStart, sysPause, sysStop}

Color ::= ENUMERATED {black, pink, red}

SystemState ::= ENUMERATED {idle, initializing, runningOk, emergencyLand, safeReturn}

SafetyEvent ::= ENUMERATED {safe,collision,wallColl,ceilColl}

FeedbackToGUI ::= SEQUENCE {
 flight WorldData,
 notification SystemState
 }

PositionSystemData ::= SEQUENCE {
 xAct MyReal,
 yAct MyReal
 }

DroneSensorData ::= SEQUENCE {
 yawAct MyReal,
 pitchAct MyReal,
 rollAct MyReal,
 baropAct MyReal,
 accxAct MyReal,
 accyAct MyReal,
 acczAct MyReal
 }

DroneControllerInput ::= SEQUENCE {
 yawrateRef MyReal,
 pitchRef MyReal,
 rollRef MyReal,
 thrustRef MyReal
 }

MyChoice ::= CHOICE {
 a BOOLEAN,
 b MySeq
 }

Position ::= SEQUENCE {
 x MyReal,
 y MyReal,
 z MyReal
 }

```
Orientation ::= SEQUENCE{
    roll MyReal,
    pitch MyReal,
    yaw MyReal
}

Velocity ::= SEQUENCE {
    vX MyReal,
    vY MyReal,
    vZ MyReal
}

Agent ::= SEQUENCE {
    agentID T-Int8,
    agentColor Color,
    currentPosition Position,
    currentOrientation Orientation,
    currentVelocity Velocity
}

Area ::= SEQUENCE {
    height T-Int8,
    width T-Int8,
    depth T-Int8,
    refLocation Position
}

HoverAction ::= SEQUENCE {
    yaw-rate MyReal,
    duration MyReal
}

ReferencePath ::= SEQUENCE {
    locations SEQUENCE (SIZE (3)) OF Position,
    actions SEQUENCE (SIZE (3)) OF HoverAction
}

ReferenceFormation ::= SEQUENCE {
    --name STRING,
    nodes SEQUENCE (SIZE (5)) OF T-Int8, -- AgentID
    anchor T-Int8, --AgentID
    formationAnchorPosition Position
}

WorldData ::= SEQUENCE {
    myArea Area,
    refPath ReferencePath,
    refFormation ReferenceFormation,
    agents SEQUENCE (SIZE (3)) OF Agent
}

MySeqOf ::= SEQUENCE (SIZE (2)) OF MyEnum

MyOctStr ::= OCTET STRING (SIZE (3))

-- You can also declare constants
myVar MySeqOf ::= { hello, world }

END
```