# Ethical Control of Unmanned Systems

## Mission Design, Semantic Web Exemplars and Data-Centric Security Considerations for Human Supervision of Lethal/Lifesaving Autonomy

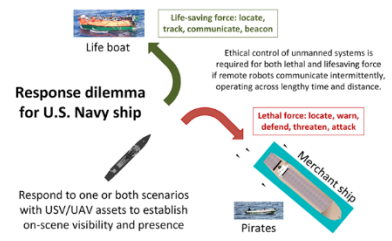
Don Brutzman, Curt Blais and Bob McGhee

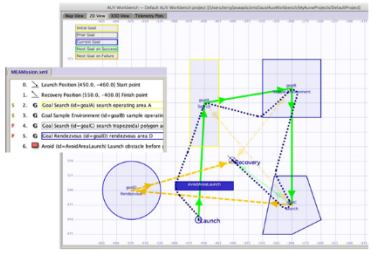
Naval Postgraduate School (NPS)

18 January 2021

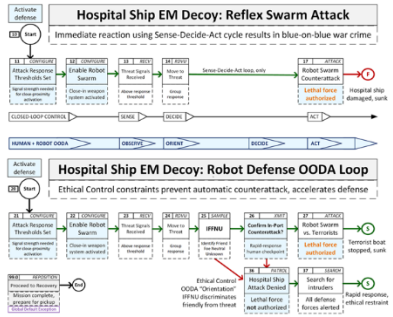
# Ethical Control of Unmanned Systems:
## Keeping Warfighters in Charge of Autonomy

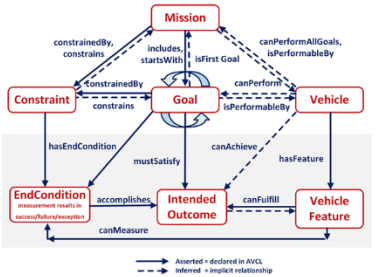
CRUSER
Consortium for Robotics and Unmanned Systems Education and Research



Mission Execution Ontology (MEO) evolving




## Milestones and Transitions

- CRUSER development led to first project selection under CRADA with Raytheon Missile Systems (RMS).
- Successful progress on test missions entering TRL 5 with simulation and Web-sharable 3D visualization.
- Expressing multiple robot mission plans consistently, coherently for diverse UAV, USV, UUV platforms.
- Use Semantic Web Standards to support warfighters.
- Evaluate NAVSEA Unmanned Maritime Autonomy Architecture (UMAA) evolution for robot qualification.

## Why / Objectives

- Ethical control of unmanned systems can be accomplished through structured mission definitions that are trusted, consistently readable, validatable, repeatable and understandable by humans and robots.
- Orders must be lawful. Unmanned systems must behave ethically and comprehensibly if they are to support manned military units effectively.
- Well-structured mission orders can be tested and trusted to give human commanders confidence that offboard systems *will do what they are told to do*, and further *will not do what they are forbidden to do*.
- Demonstrate that no technological limitations exist that prevent applying the same kind of ethical constraints on robots and unmanned vehicles that already apply to humans, in lethal and life-saving scenarios.

https://savage.nps.edu/EthicalControl

## What / Deliverables

- Update Mission Execution Ontology (MEO) concepts demonstrated in tests and simulation, building to perform field experimentation (FX).
- Supervise thesis work to explore canonical exemplar missions that are expected to utilize unmanned systems, looking across the full range of Naval warfare communities. Example scenarios include UAV for sailor overboard, UAV for refugee/lifeboat escort, and adept scouts. All must observe Law of Armed Conflict (LOAC), Rules of Engagement (ROE), and moral guidance of commanders despite long durations/distances.
- Define, simulate, and test combination of real-world goals and ethical constraints to robot mission tasking across set of canonical scenarios.
- Illustrate how human-robot teams meet moral and legal requirements if deploying unmanned systems with potential for lethal, life-saving force.

NPS PRAESTANTIA PER SCIENTIAM

**Building on Concept Demonstrations**


Principal Investigator: Don Brutzman brutzman@nps.edu 1.831.656.2149
Co-Investigator: Curtis Blais clblais@nps.edu 1.831.656.3215

# Synopsis: Ethical Control of Unmanned Systems

- **Project Motivation**: ethically constrained control of unmanned systems and robot missions by human supervisors and warfighters.
- **Precept:** well-structured mission orders can be syntactically and semantically validated to give human commanders confidence that offboard systems
  - *will **do** what they are told to **do***, and further
  - *will **not do** what they are **forbidden to do***.

**Paraphrase: can qualified robots correctly follow human orders?**

- **Project Goal:** apply Semantic Web ontology to scenario goals and constraints for logical validation that human-approved mission orders for robots are semantically coherent, precise, unambiguous, and without internal contradictions.
- **Long-term Objective:** demonstrate that no technological limitations exist that prevent applying the same kind of ethical constraints on robots and unmanned vehicles that already apply to human beings.

# Table of Contents

# Open-Source Licensing for Ethical Control

All NPS work is published as open source with no restrictions on use.

- https://savage.nps.edu/EthicalControl/license.html (also .txt)

An excellent heuristic is to assume success – then what?

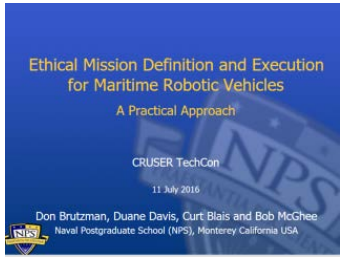
- Ethical Control of unmanned systems is critically important worldwide.
- Ethical Control must be unambiguously defined in order to be actionable by humans (military professionals) under all conditions facing U.S. military.
- Ethical Control must be implementable across a totally diverse set of evolving unmanned and C2 systems (both open and closed) in order to be effective.
- Ethical Control concepts, syntax definitions and semantic relationships for lethal/lifesaving force thus *cannot* be subject to proprietary restrictions.

# Background History

Many years of work have composed multiple fields of study to provide techniques for maintaining human ethical control of unmanned systems.

In this work, ethical theory meets professional practice. Each step must work for human commanders and unmanned systems alike.

# Key Insights regarding Human Ethical Control




1. **Humans in military units are able to deal with moral challenges without ethical quandaries,**
   - **by using formally qualified experience, and by following mission orders that comply with Rules of Engagement (ROE) and Laws of Armed Conflict (LOAC).**

2. Ethical behaviors don't define the mission plan. Instead, **ethical constraints inform the mission plan.**

3. **Naval forces can only command mission orders that are**
   - **Understandable by (legally culpable) humans, then**
   - **Reliably and safely executed by robots.**

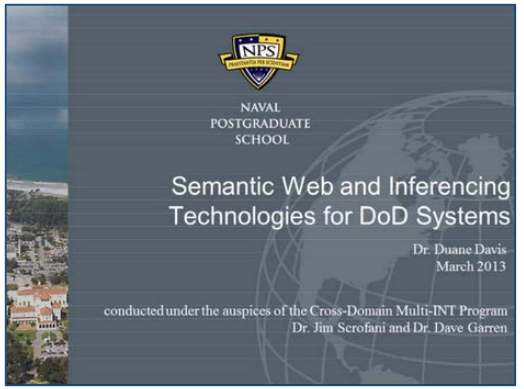
*Reference:* CRUSER TechCon Overview 2016
https://gitlab.nps.edu/Savage/EthicalControl/tree/master/documents/presentations

# Ethical Mission Definition and Execution for Maritime Robots Under Human Supervision


Theoretical Basis

- Don Brutzman, Curtis L. Blais, Duane T. Davis, Robert B. McGhee
- IEEE Journal of Oceanic Engineering (JOE), Volume: 43 , Issue: 2 , April 2018
- *Abstract.* Experts and practitioners have worked long and hard toward achieving functionally capable robots. While numerous areas of progress have been achieved, ethical control of unmanned systems meeting legal requirements has been elusive and problematic. Common conclusions that treat ethical robots as an always-amoral philosophical conundrum requiring undemonstrated morality-based artificial intelligence are simply not sensible or repeatable. Patterning after successful practice by human teams shows that precise mission definition and task execution using well-defined, syntactically valid vocabularies is a necessary first step. Addition of operational constraints enables humans to place limits on robot activities, even when operating at a distance under gapped communications. Semantic validation can then be provided by a Mission Execution Ontology to confirm that no logical or legal contradictions are present in mission orders. Thorough simulation, testing, and certification of qualified robot responses are necessary to build human authority and trust when directing ethical robot operations at a distance. Together these capabilities can provide safeguards for autonomous robots possessing the potential for lethal force. This approach appears to have broad usefulness for both civil and military application of unmanned systems at sea.

# Semantic Web and inferencing technologies for Department of Defense systems




- Duane Davis, NPS Technical Report NPS-CMIS-14-001, 2014
- Available via https://calhoun.nps.edu/handle/10945/43723 (report) (slideset)
- *Abstract*. Operational commanders and intelligence professionals are provided with a continually-increasing volume of data from numerous sources. Effective utilization of this data can be hampered by difficulties in fusing different data streams for presentation, correlating related data from various sources and developing reliable summary and predictive products. An opportunity presently exists to improve this situation through the incorporation of Semantic Web technologies into Department of Defense (DOD) systems. This report provides a didactic overview of Description Logics (DL) and their implementation in Semantic Web languages and technologies to include the mathematical properties supporting robust knowledge representation. Subsequently, the algorithms for automated reasoning and inferencing with DLs are discussed. Included in this discussion is a comparison of available Semantic Web applications for ontology development and realization or DL reasoning capabilities with real-world knowledge bases. Finally, mechanisms for applying artificial intelligence techniques to ontological DL information are presented.

# Description Logic (DL) Rules provide basis for Mission Execution Ontology (MEO)

| Rules | Description Logic Equations | Plain-language description |
|---|---|---|
| **M = Mission Rules** | | |
| **M1** | Mission ⊑ ∀startsWith.Goal ⊓ =1startsWith.Goal | A Mission can only start with a Goal and must start with exactly one Goal |
| **M2** | Mission ⊑ ∀includes.Goal ⊓ ≥1includes.Goal | A Mission can only include Goals and must include one of more Goals |
| **M3** | Mission ⊑ ∀hasConstraint.Constraint | A Mission can only be constrained by Constraints |
| **M4** | startsWith ⊑ includes | A Mission must include the Goal that it starts with |
| **M5** | Mission ⊑ ∀performableBy.Vehicle | A Mission can only be performed by a Vehicle |
| **M6** | Cannot be expressed in DL | A Mission cannot be performable by a Vehicle unless that Vehicle has the ability to identify all Constraints associated with that mission |
| **M7** | Cannot be expressed in DL | A Mission cannot be performable by a Vehicle unless that Vehicle has the capability to accomplish all Goals included in that Mission |

Excerpted from full Mission Execution Ontology Decision Logic Tables

Original author:
Duane Davis

# NPS contributions toward achieving run-time ethical constraints in mission execution by autonomous robots and human/robot teams 1

History

1. NPS began research on autonomous underwater vehicles (AUV) in the early 1990's. The first vehicle, tested at sea in the mid-1990's, was "Phoenix". This vehicle length was around 6 ft., weight approximately 600 lb. At that time, the onboard control system was called RBM for "Rational Behavior Model". RBM was modeled on standard naval practice for deployment and operation of manned submarines at the time.

2. The top level of the three layer RBM software architecture assumed that AI was required to replace the function of a submarine commander. Specifically, we believed that first order logic (predicate calculus) would be needed for mathematical modeling and programming of a commander's function. Therefore a separate "SPARC" workstation running "Prolog" language was included in Phoenix. RBM functioned well in Phoenix at sea testing.

3. In early 2000's, Phoenix was replaced by a larger vehicle called "Aries", that contained a more advanced on board sensor suite, larger batteries, and more powerful main propulsion thrusters. Aries was about 1000 lb. in weight and 8 feet long, and capable of longer duration and more complex missions than its predecessor. Experience with Phoenix led us to full practical realization that mission control based on human style reasoning (predicate logic) cannot, (in general) be proved correct for a given mission. Fortunately, we also came to realize that "finite state" logic is adequate for any mission control tasks we actually anticipated carrying out by autonomous vehicles. In contrast to our experience with Phoenix, for Aries we were able to prove by exhaustion of all possible outcomes that its missions were correctly programmed (that is, that they accomplished what we intended). Aries was a success, and was retired after completion of all planned mission tasks. All results were published.

# NPS contributions toward achieving run-time ethical constraints in mission execution by autonomous robots and human/robot teams 2

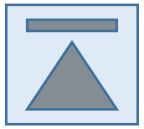
History

4. Early in the current decade, we began to realize that responsible experimentation with larger and more powerful robots (AUVs or others) would require that some run-time ethical constraints be incorporated into their mission control software. This was not done for Phoenix or Aries. As we addressed this requirement, it become apparent to us that inclusion of such constraints requires a possible "exception" outcome of execution of a mission phase goal (command) in addition to the "success" and "failure" outcomes we had previously considered. This understanding was summarized in our 2018 JOE paper that was based on such "tri-state" logic.

5. Since publication of our 2018 paper, we have been concerned with implementation details for tri-state mission logic for autonomous robots and human/robot teams. To date, we have demonstrated, in human interactive form, execution of a simulated 8-state "sailor overboard" mission by a human/robot team, using either Prolog or Common Lisp as a programming language. We expect soon to complete an XML implementation.

6. A key aspect of tri-state logic, including possible violation of an ethical constraint, either pending or actual, is the need for constant situational awareness by mission control software. We believe that mandating this type of software for mission control could possibly have prevented loss of human life in recent passenger aircraft and self-driving car accidents. We urge further research on this issue ASAP.

-- Robert McGhee, NPS, December 2019



# Mission Representation using Autonomous Vehicle Command Language (AVCL)

Structured vocabulary to define unmanned-system missions, understandable by human commanders, useful in multiple programming languages plus Semantic Web logical queries.

# Autonomous Vehicle Command Language (AVCL)

- AVCL is a command and control language for humans supervising autonomous unmanned vehicles.
  - Clarity arises from close correspondence to human naval terminology.

Mission Tasking

- Structured vocabulary defining terms and relationships for mission planning, execution, conduct, recording and replay across diverse robot types.
- Common-ground XML representations for
  - Mission agenda plans, mission scripts, and post-mission recorded telemetry results.
  - ***Future work:*** defining unit tests and expected results for verification and validation.
- Operators have single archivable, validatable format for robot tasking, results
  - directly convertible to and from a wide variety of different robot command languages.

https://savage.nps.edu/Savage/AuvWorkbench/AVCL/AVCL.html

# Example AVCL mission agenda, as pseudo-code XML

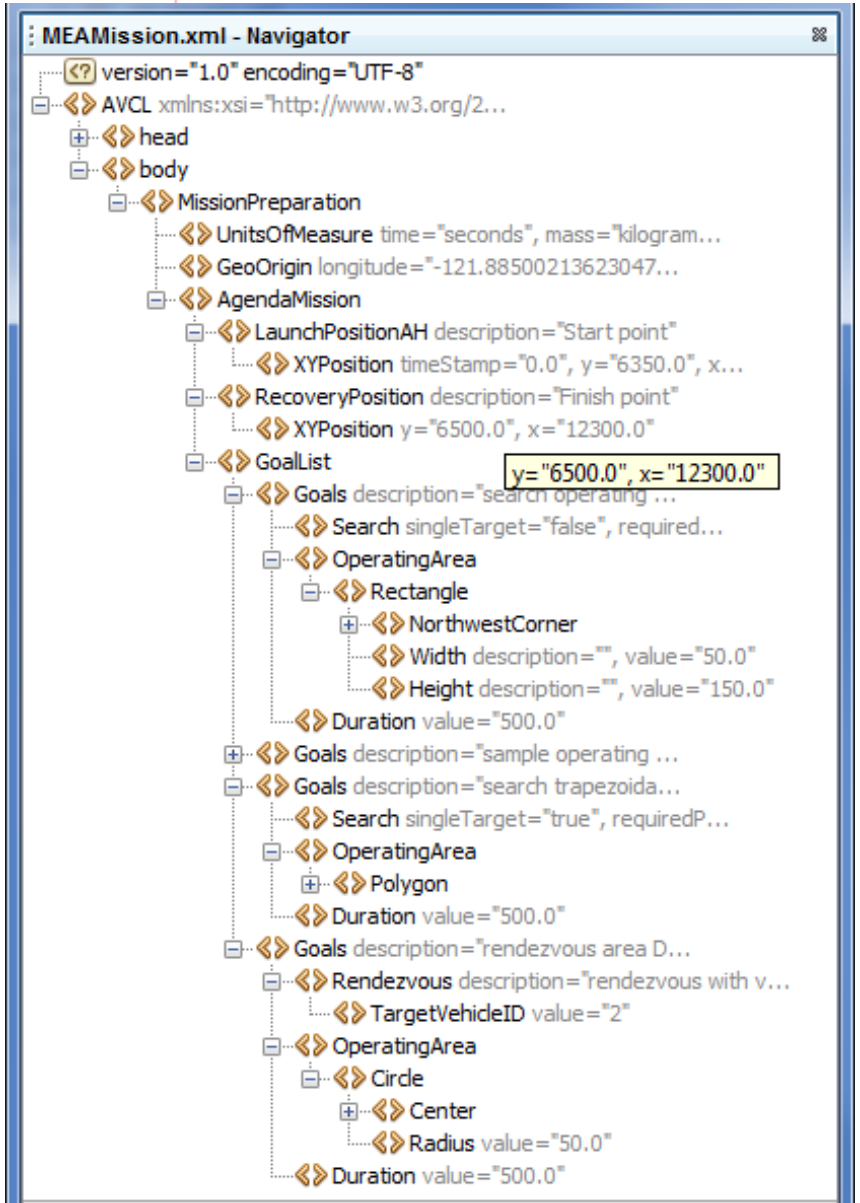
```
<?xml version="1.0" encoding="UTF-8"?>
<UUVMission>
  <GoalSet>
    <Goal area=”A” id=”goal1”>
      <Search nextOnSuccess=”goal2” nextOnFailure=”goal3”/>
    </Goal>
    <Goal area=”A” id=”goal2”>
      <SampleEnvironment nextOnSuccess=”goal3”
          nextOnFailure=”recover”/>
    </Goal>
    <Goal area=”B” id=”goal3”>
      <Search nextOnSuccess=”goal4” nextOnFailure=”goal4”/>
    </Goal>
    <Goal area=”C” id=”goal4”>
      <Rendezvous nextOnSuccess=”recover” nextOnFailure=”recover”/>
    </Goal>
    <Goal area=”recoveryPosition” id=”recover”>
      <Transit nextOnSuccess=”missionComplete”
          nextOnFailure=”missionAbort”/>
    </Goal>
  </GoalSet>
</UUVMission>
```

AVCL is readable by human or robot, captures logic of mission tasking

XML ensures syntactically correct, well-defined, numerically valid

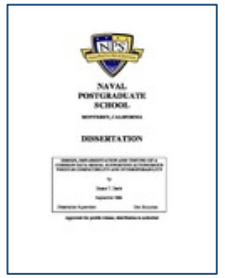
Needed: semantic representation to check ethical, logical consistency

```
<MissionPreparation>
  <UnitsOfMeasure time="seconds" mass="kilograms" angle="degrees" distance="meters" />
  <GeoOrigin longitude="-121.88500213623047" latitude="36.606998443603516" />
  <AgendaMission>
    <LaunchPositionAH description="Start point">
      <XYPosition timeStamp="0.0" y="6350.0" x="12300.0" />
    </LaunchPositionAH>
    <RecoveryPosition description="Finish point">
      <XYPosition y="6500.0" x="12300.0" />
    </RecoveryPosition>
    <GoalList>
      <Goals description="search operating area A" alert="false"
             nextOnFail="goalC" nextOnSucceed="goalB" id="goalA">
        <Search singleTarget="false" requiredPD="0.8"
             datumType="area" />
        <OperatingArea>
          <Rectangle>
            <NorthwestCorner>
              <XYPosition y="6425.0" x="12625.0" />
            </NorthwestCorner>
            <Width description="" value="50.0" />
            <Height description="" value="150.0" />
          </Rectangle>
        </OperatingArea>
        <Duration value="500.0" />
      </Goals>
```




# Corresponding example: MEAMission.xml using AVCL document

# NPS Dissertation: AVCL mission vocabulary




Semantic Web Deep Dive

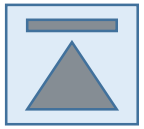
## Design, implementation and testing of a common data model supporting autonomous vehicle compatibility and interoperability

- Duane T. Davis, Naval Postgraduate School (NPS), September 2006
- *Abstract*. Current autonomous vehicle interoperability is limited by vehicle-specific data formats and support systems. Until a standardized approach to autonomous vehicle command and control is adopted, true interoperability will remain elusive. This work explores the applicability of a data model supporting arbitrary vehicles using the Extensible Markup Language (XML). An exemplar, the Autonomous Vehicle Command Language (AVCL), encapsulates behavior-scripted mission definition, goalbased mission definition, inter-vehicle communication, and mission results. Broad applicability is obtained through the development of a behavior set capturing arbitrary vehicle activities, and automated conversion of AVCL to and from vehicle-specific formats. The former uses task-level behaviors suitable for mission scripting and goal decomposition. Translations use the Extensible Stylesheet Language for Transformation, XML data binding, context-free language parsing, and artificial intelligence machine learning and search techniques. Translation capability is demonstrated through mappings of AVCL to and from multiple vehicle-specific formats. A final demonstration of the power of a common autonomous vehicle data model is provided by the implementation of a hybrid control architecture. The model's vehicle-independence and the ability to generate vehicle-specific data are leveraged in the design of an architecture that provides increased autonomy by augmenting a vehicle's existing controller. The utility of this architecture is demonstrated through implementation on the Naval Postgraduate School's ARIES Unmanned Underwater Vehicle.

# AVCL mission goals vocabulary (Davis 2015)

| AVCL mission goals | Define | Used | Definition |
|---|---|---|---|
| **Attack** | partial | √ | To conduct a type of offensive action characterized by employment of firepower and maneuver to close with and destroy an enemy. |
| **Decontaminate** | √ | | To provide purification making an area safe by absorbing, destroying, neutralizing, making harmless, or removing chemical, biological, or nuclear contamination. |
| **Demolish** | √ | | To destroy structures, facilities, or material by any available means. |
| **IlluminateArea** | √ | | To provide locale lighting by searchlight or pyrotechnics. |
| **Jam** | √ | | To deliberately radiate, re-radiate or reflect electromagnetic energy with the object of impairing the use of electronic devices or systems. |
| **MarkTarget** | √ | √ | To make visible (by the use of light, infrared, laser, smoke, etc.) of an object in order to allow its identification by another object. |
| **MonitorTransmissions** | √ | √ | To conduct electronic warfare support operations with a view to searching, locating, recording and analyzing radiated electromagnetic energy. |
| **Patrol** | √ | √ | To gather information or carry out a security mission. |
| **Rendezvous** | √ | Partial | Achieve a meeting at a specified time and place. |
| **Reposition** | √ | √ | To change position from one location to another. |
| **SampleEnvironment** | Partial | √ | Collect environmental samples for testing for chemical compounds, biological creatures, or nuclear hazards. |
| **Search** | √ | √ | To look for lost or unlocated objects or persons. |

More Goal Types Foreseen



# Semantic Web and Mission Execution Ontology (MEO)

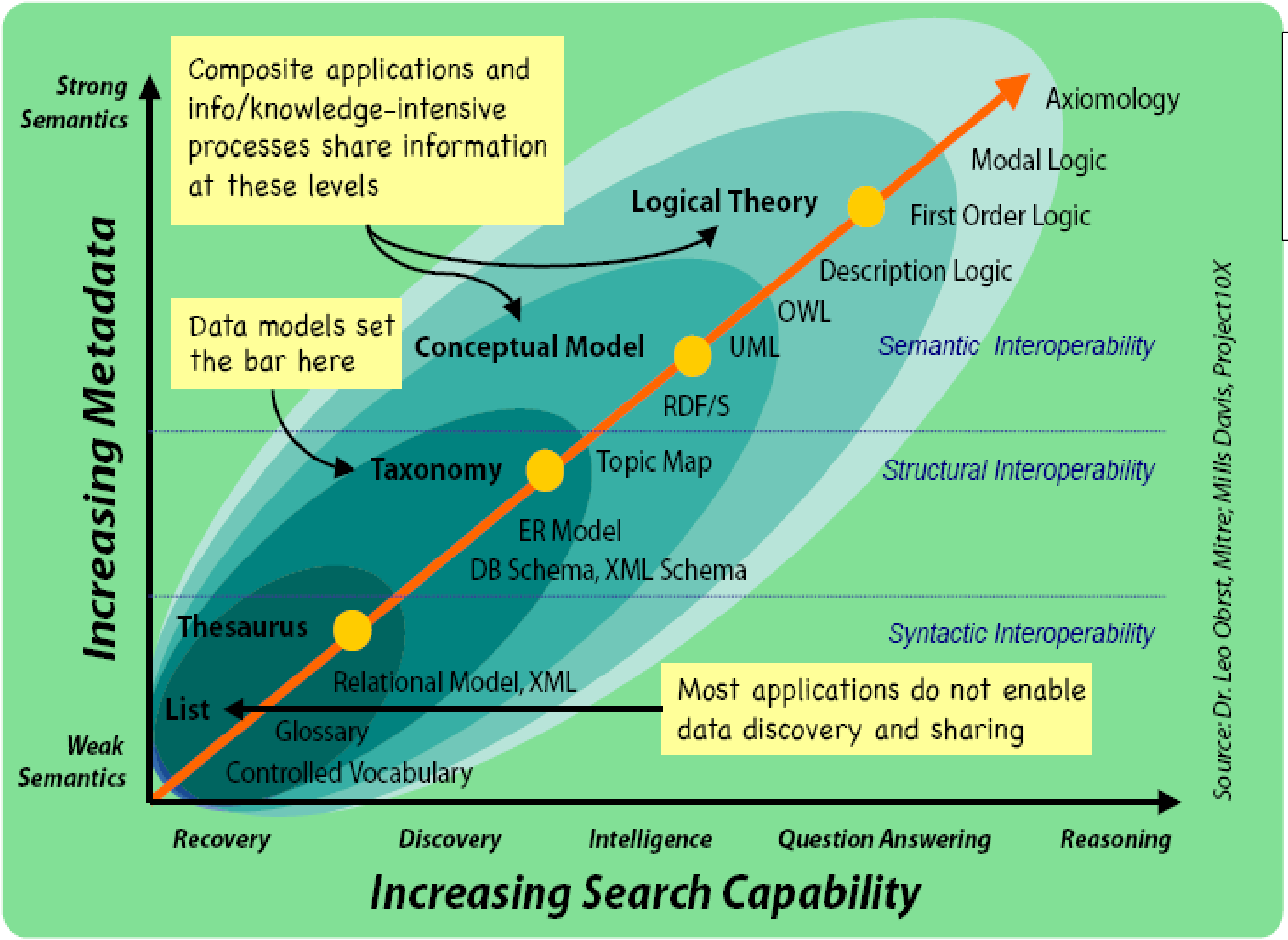
Apply well-developed Semantic Web standards to integrate queries and reasoning to AVCL for logical consistency checks.

# Background: Improving Semantic Representation

- Knowledge Representation (KR) is an area of artificial intelligence (AI) research and practice focused on encoding meaning into data.
- Academia and industry now have a detailed path toward higher levels of machine understanding corresponding to human understanding.
- Refer to the Ontology Spectrum on the next slide (source: Obrst, Leo, and Mills Davis. 2006. *SICoP White Paper Series Module 2: Semantic Wave 2006 - Executive Guide to the Business Value of Semantic Technologies*, Project 10X, May 2015).

Acronyms: Database (DB) ; Extensible Markup Language (XML); Resource Description Language / Schema (RDF/S); Unified Modeling Language (UML); Web Ontology Language (OWL)

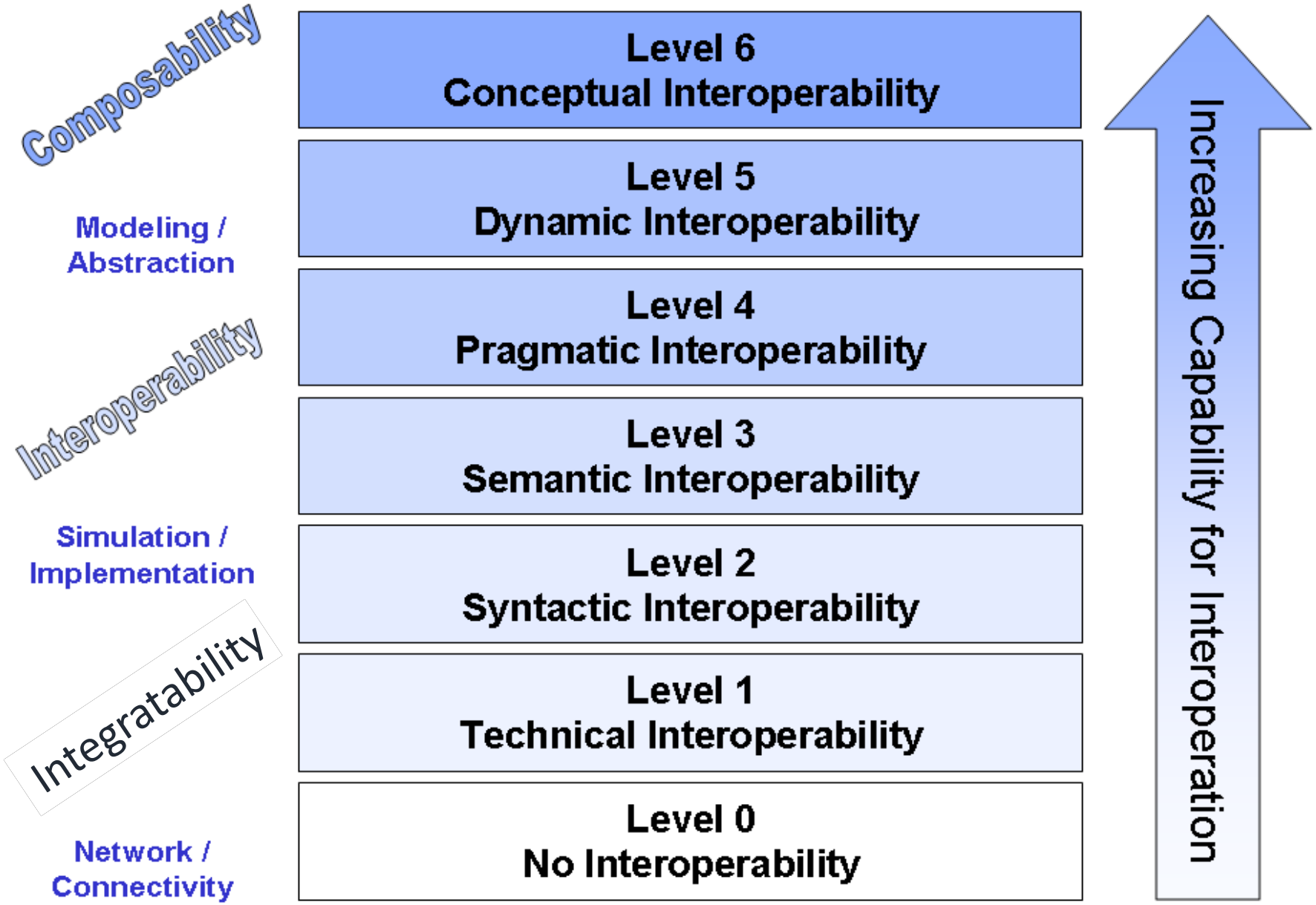
# Improving Semantic Representation



# Background: Improving Interoperability

- Interoperability: “the capability of a system to automatically, without human intervention, provide services to and accept services from other systems, and to use the services so exchanged to enable the systems to work together to achieve a desired outcome” (Blais and Lacy 2004).
- Academia and industry have laid out a path toward higher levels of interoperability: refer to the Levels of Conceptual Interoperability Model (LCIM) on the next slide (source: Tolk, Andreas, Saikou Y. Diallo, Charles D. Turnitsa, and Leslie S. Winters. 2006. “Composable M&S Web Services for Net-centric Applications.” *Journal for Defense Modeling & Simulation (JDMS)*, 3:1:27-44. January.).
- Objective is to achieve conceptual and pragmatic interoperability.

# Improving Interoperability: Levels of Conceptual Interoperability Model



# Background: Semantic Web

- Architects of the World Wide Web have laid out a layered set of standards to achieve the Semantic Web vision: “not a separate Web but an extension of the current one, in which information is given well-defined meaning, better enabling computers and people to work in cooperation” (Berners-Lee et al. 2001)
- Ultimate goal: achieve a scalable trusted information infrastructure where humans and software interact meaningfully, in a repeatable environment where expectations of quality and integrity are met.
- Scalable approach indicates that single (ship + robot) solutions have potential to grow and encompass many simultaneous systems, thus improved data sharing, mission deconfliction, coordinated operations

# Semantic Web Stack

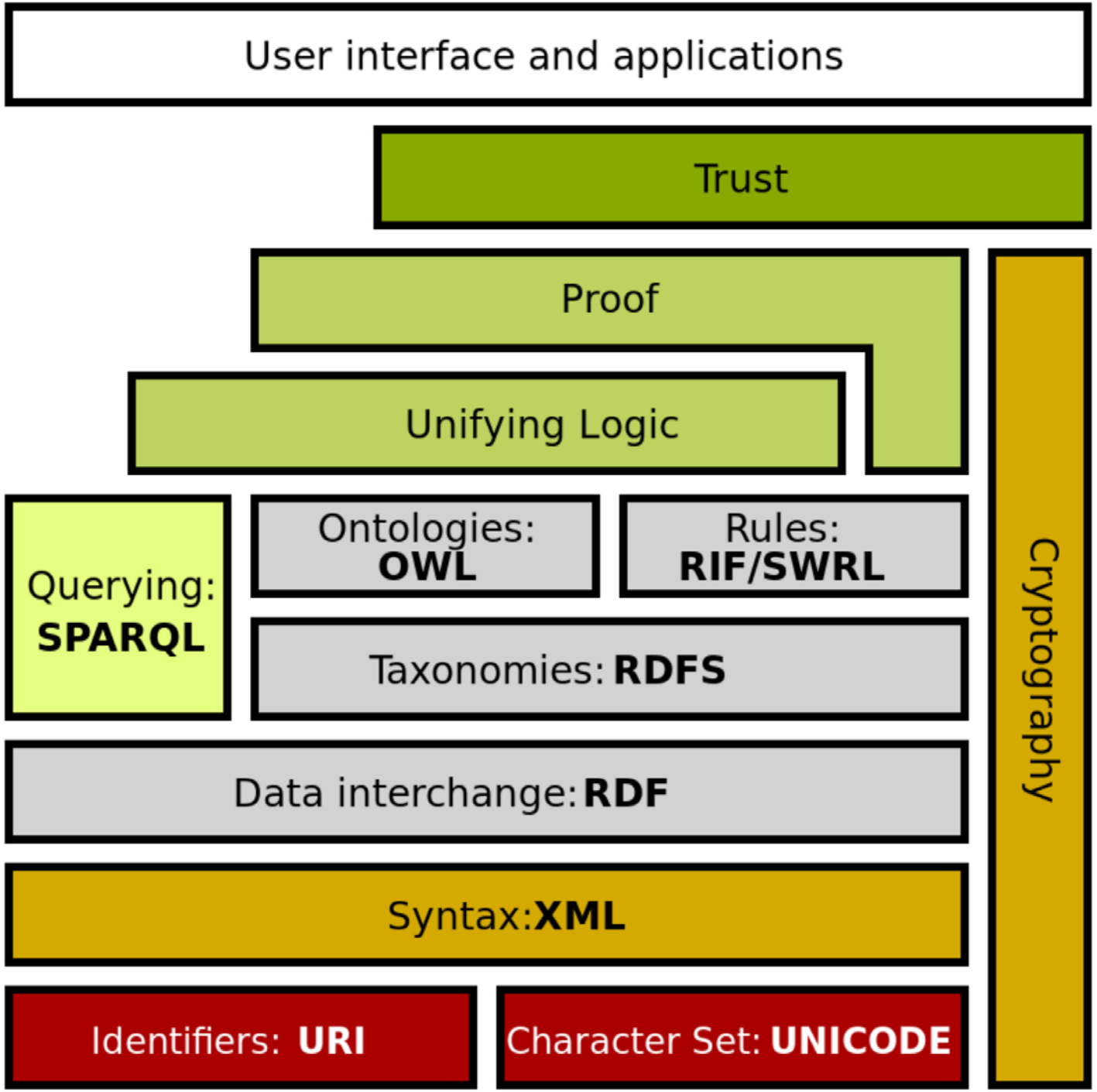
Extends larger Web architecture

- All of these data languages are approved W3C standards
- Proof and unifying logic are mathematically well defined

Trusting derived (composed) statements arises from

- Encryption + digital signature confirms trusted data sources
- Formal logic is basis for deriving new information
- Wikipedia: Semantic Web Stack

Of note: this project is exercising every layer of Semantic Web stack.



# Motivating insights: converting data into logic

"The answer to your question is the response to the query."

- Jim Hendler and Dean Allemang
- Meaningfulness of an answer matches the precise meaning of a question.

"Trying to use the Semantic Web without SPARQL is like trying to use a relational database without SQL.“

- Tim Berners-Lee
- Language representations are needed for query as well as for information.

"The proof of the pudding is in the eating."

- Wiktionary
- Confidence in results requires testing.

# Mission Execution Ontology (MEO) Development

- ✓Define MEO from concepts, properties, relationships using Protégé tool.
- ✓Create full set of canonical missions in AVCL (XML).
- ✓Determine exemplar mappings for AVCL primitives to Turtle for RDF/OWL.
- ✓Write conversion stylesheet AvclToMEO.xslt for full expressiveness.
- ✓Convert all AVCL missions to corresponding triples.
- ✓Confirm AVCL MEO, missions validate satisfactorily using Protégé, ARQ.
- ✓Automate build process as suite of repeatable unit-test queries (log).
- ✓Write SPARQL metaqueries to test, document MEO ontology relationships.
- ✓In progress: SPARQL queries to test AVCL mission representations in Turtle.

# Specific developer build.xml capabilities using Apache Ant

- ✓Test and validation utility tasks, release results checked in (build.all.log.txt)
- ✓AVCL mission conversion to MEO triples: *convertAvclToMeoTurtle.examples.all*
- ✓AVCL mission conversion to other programming languages: *convertAvclToLisp.examples.all*, *convertAvclToProlog.examples.all*
- ✓View gitlab.nps.edu site, locally created owldoc, etc.
- ✓Perform *SPARQL.query.tests* (results)
- ✓Create downloadable .zip of fully documented archive
- ✓Run, test example missions in Lisp, Prolog programming languages
- ✓Deploy, publish development site as https://savage.nps.edu/EthicalControl

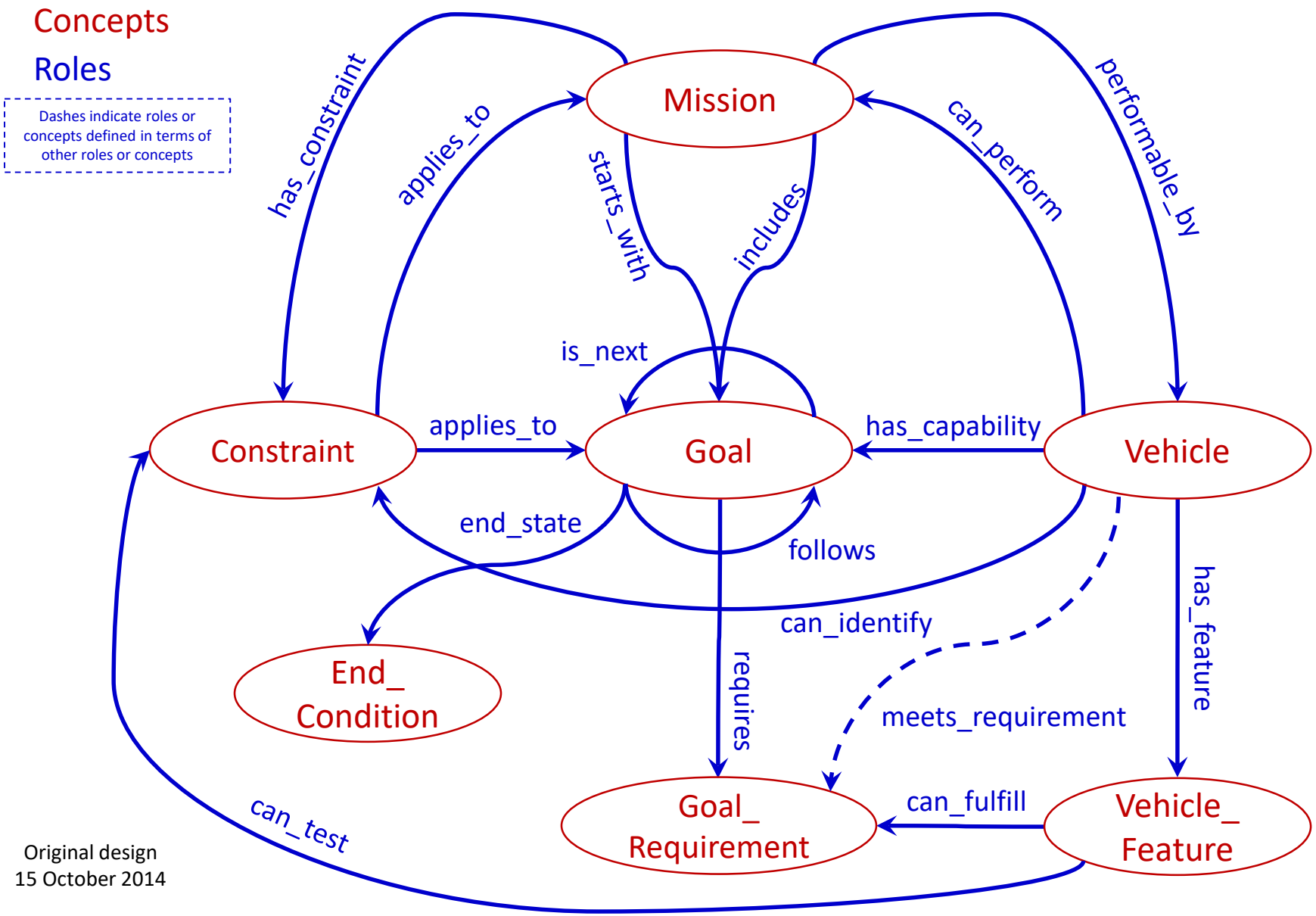
# Mission Execution Ontology (MEO) for Ethical Control of Unmanned Systems in Surrogate Scenarios

Summary of relationships

- **Autonomous Vehicle Command Language (AVCL) for Missions.**
  - Declarative XML, years of NPS research.
- **Multiple Mission Representations.**
  - Imperative commands (orders/waypoints/etc.).
  - Declarative commands (mission goals).
  - Mission results (order log, telemetry etc.).
  - Mission metadata for parameters, settings.
  - Lisp and Prolog examples (Bob McGhee, NPS).
- **Autonomous Unmanned Vehicle (AUV) Workbench Simulation and Visualization Support**
  - Recently restored, debug testing commenced.
  - AVCL 2.1 is prior published version, centered on *syntactic validation*, solo robot operations.
  - AVCL 3.0 is new working version for testing range of multi-participant missions.

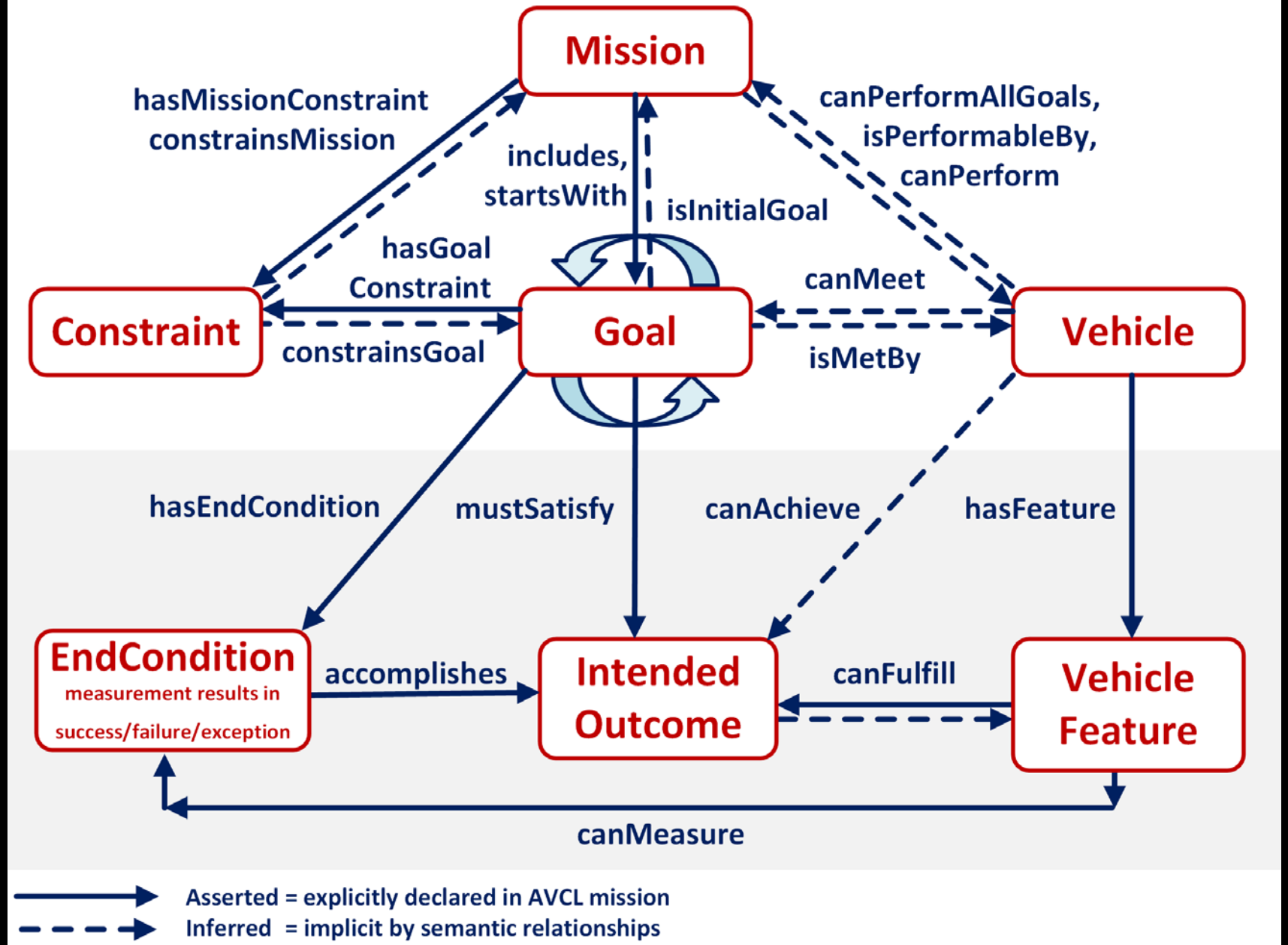
- **Mission Execution Ontology (MEO) for Semantic Validation**
  - Semantic Web framework of rules, relationships for *ethical validation.*
  - Initial examples in IEEE JOE paper.
  - Retested using current Protégé, Jena tools.
- **Sailor Overboard and Other Missions**
  - Hand-crafted triples using Turtle syntax.
  - Beginning to build unit testing framework.
  - Confirming correlation of AVCL information model to existing MEO ontology.
  - Automatic conversion of AVCL missions to match, thus accelerating multiple-mission testing on diverse systems.
  - Visualization, reporting via AUV Workbench can aid understanding, mission planning and further progress.

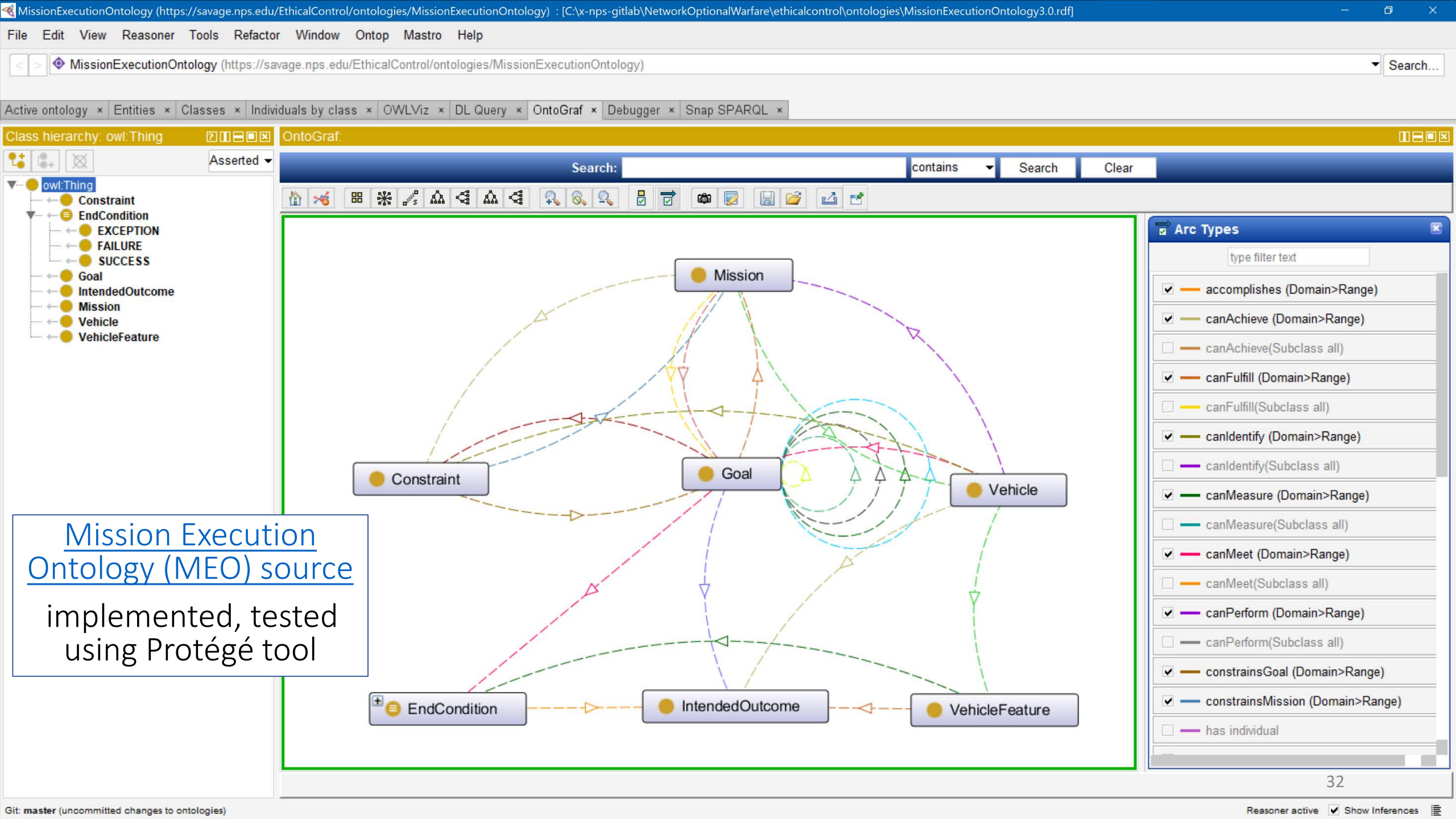
# Unmanned Vehicle Mission Execution Ontology (MEO)




Original design
15 October 2014

# Mission Execution Ontology (MEO) 3.0




MissionExecutionOntology (https://savage.nps.edu/EthicalControl/ontologies/MissionExecutionOntology) : [C:\x-nps-gitlab\NetworkOptionalWarfare\ethicalcontrol\ontologies\MissionExecutionOntology3.0.rdf]
File Edit View Reasoner Tools Refactor Window Ontop Mastro Help
MissionExecutionOntology (https://savage.nps.edu/EthicalControl/ontologies/MissionExecutionOntology)
Search...
Active ontology Entities Classes Individuals by class OWLViz DL Query OntoGraf Debugger Snap SPARQL
Class hierarchy: owl:Thing
Asserted
owl:Thing
Constraint
EndCondition
EXCEPTION
FAILURE
SUCCESS
Goal
IntendedOutcome
Mission
Vehicle
VehicleFeature
OntoGraf:
Search:
contains
Search
Clear
Mission
Constraint
Goal
Vehicle
EndCondition
IntendedOutcome
VehicleFeature
Arc Types
type filter text
accomplishes (Domain>Range)
canAchieve (Domain>Range)
canAchieve(Subclass all)
canFulfill (Domain>Range)
canFulfill(Subclass all)
canIdentify (Domain>Range)
canIdentify(Subclass all)
canMeasure (Domain>Range)
canMeasure(Subclass all)
canMeet (Domain>Range)
canMeet(Subclass all)
canPerform (Domain>Range)
canPerform(Subclass all)
constrainsGoal (Domain>Range)
constrainsMission (Domain>Range)
has individual
Mission Execution Ontology (MEO) source
implemented, tested using Protégé tool
32
Git: master (uncommitted changes to ontologies)
Reasoner active
Show Inferences

```
@prefix :     <https://www.nps.edu/ontologies/MissionExecutionOntology> .
@prefix meo:  <https://www.nps.edu/ontologies/MissionExecutionOntology#> .
@prefix owl:  <http://www.w3.org/2002/07/owl#> .
@prefix rdf:  <http://www.w3.org/1999/02/22-rdf-syntax-ns#> .
@prefix rdfs: <http://www.w3.org/2000/01/rdf-schema#> .
@prefix xml:  <http://www.w3.org/XML/1998/namespace> .
@prefix xsd:  <http://www.w3.org/2001/XMLSchema#> .
@base <https://www.nps.edu/ontologies/MissionExecutionOntology> .

<https://www.nps.edu/ontologies/MissionExecutionOntology> rdf:type owl:Ontology .

#################################################################
#    Object Properties
#################################################################

###  https://www.nps.edu/ontologies/MissionExecutionOntology#appliesTo
meo:appliesTo rdf:type owl:ObjectProperty ;
  rdfs:domain meo:Constraint ;
  rdfs:range [ rdf:type owl:Class ;
               owl:unionOf ( meo:Goal meo:Mission ) ] ;
  owl:propertyChainAxiom ( meo:appliesTo meo:includes ) ;
  rdfs:comment "A Constraint applies to one or more Missions and/or one or more Goals." .
[ rdf:type owl:Axiom ;
  owl:annotatedSource meo:appliesTo ;
  owl:annotatedProperty rdfs:range ;
  owl:annotatedTarget [ rdf:type owl:Class ; owl:unionOf ( meo:Goal meo:Mission ) ] ;
  rdfs:comment "A Constraint can apply to a Mission or a Goal (and nothing else)." ;
  rdfs:label "C1" ] .
[ rdf:type owl:Axiom ;
  owl:annotatedSource meo:appliesTo ;
  owl:annotatedProperty owl:propertyChainAxiom ;
  owl:annotatedTarget ( meo:appliesTo meo:includes ) ;
  rdfs:comment "A Constraint that applies to a Mission must also apply to all of the Goals that Mission includes." ;
  rdfs:label "C3" ] .
```

Mission Execution Ontology (MEO) source

implemented, tested using Protégé tool

Turtle (.ttl) syntax

**MissionExecutionOntology.rdf** 52.7 KB

Edit | Web IDE | Replace | Delete

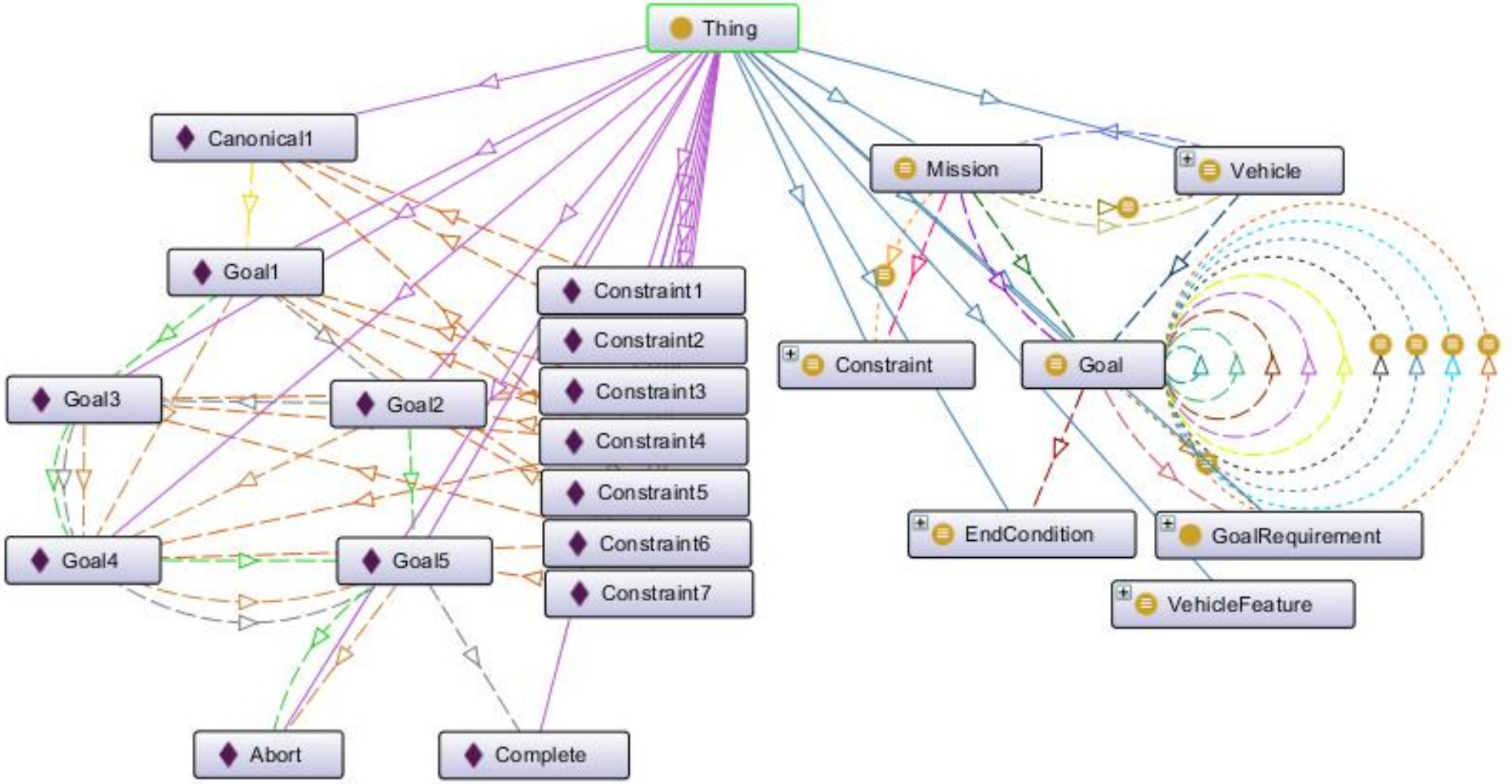
```
<?xml version="1.0"?>
<rdf:RDF xmlns="https://www.nps.edu/ontologies/MissionExecutionOntology"
     xml:base="https://www.nps.edu/ontologies/MissionExecutionOntology"
     xmlns:meo="https://www.nps.edu/ontologies/MissionExecutionOntology#"
     xmlns:owl="http://www.w3.org/2002/07/owl#"
     xmlns:rdf="http://www.w3.org/1999/02/22-rdf-syntax-ns#"
     xmlns:xml="http://www.w3.org/XML/1998/namespace"
     xmlns:xsd="http://www.w3.org/2001/XMLSchema#"
     xmlns:rdfs="http://www.w3.org/2000/01/rdf-schema#">
    <owl:Ontology rdf:about="https://www.nps.edu/ontologies/MissionExecutionOntology"/>
    <!--
    ///////////////////////////////////////////////////////////////////////////////////
    //
    // Object Properties
    //
    ///////////////////////////////////////////////////////////////////////////////////
     -->
    <!-- https://www.nps.edu/ontologies/MissionExecutionOntology#appliesTo -->
    <owl:ObjectProperty rdf:about="https://www.nps.edu/ontologies/MissionExecutionOntology#appliesTo">
        <rdfs:domain rdf:resource="https://www.nps.edu/ontologies/MissionExecutionOntology#Constraint"/>
        <rdfs:range>
            <owl:Class>
                <owl:unionOf rdf:parseType="Collection">
                    <rdf:Description rdf:about="https://www.nps.edu/ontologies/MissionExecutionOntology#Goal"/>
                    <rdf:Description rdf:about="https://www.nps.edu/ontologies/MissionExecutionOntology#Mission"/>
                </owl:unionOf>
            </owl:Class>
        </rdfs:range>
        <owl:propertyChainAxiom rdf:parseType="Collection">
            <rdf:Description rdf:about="https://www.nps.edu/ontologies/MissionExecutionOntology#appliesTo"/>
            <rdf:Description rdf:about="https://www.nps.edu/ontologies/MissionExecutionOntology#includes"/>
        </owl:propertyChainAxiom>
        <rdfs:comment>A Constraint applies to one or more Missions and/or one or more Goals.</rdfs:comment>
    </owl:ObjectProperty>
```

Mission Execution Ontology (MEO) source

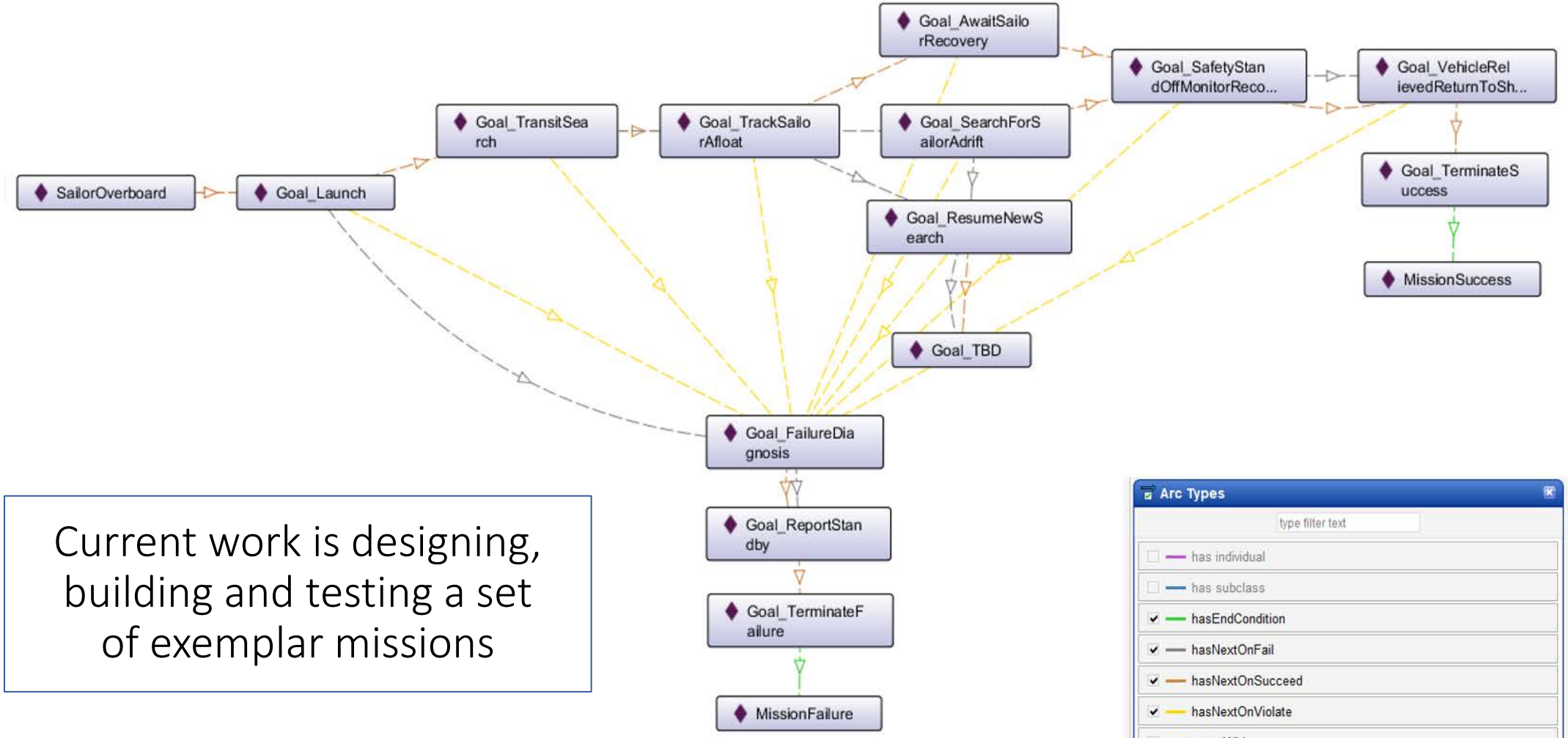
implemented, tested using Protégé tool

RDF (XML) syntax

# Example Mission Validation using Protégé Tool



# Ethical Control of Unmanned Systems in a Surrogate Scenario: Sailor Overboard Mission defined using the MEO Ontology




Current work is designing, building and testing a set of exemplar missions

# Ontology for Ethical Control of Unmanned Systems in a Surrogate Scenario: Example Relationship Definitions

**Mission Definition Expressed in Subject-Predicate-Object Triples Using Semantic Web Standards**

## Excerpt of Sailor Overboard Mission, expressed in Turtle Syntax:

```
###  https://www.nps.edu/ontologies/MissionExecutionOntology/missions#Goal_Launch
        :Goal_Launch rdf:type owl:NamedIndividual ;
                meo:hasNextOnFail :Goal_FailureDiagnosis ;
                meo:hasNextOnSucceed :Goal_TransitSearch ;
                meo:hasNextOnViolate :Goal_FailureDiagnosis .

###  https://www.nps.edu/ontologies/MissionExecutionOntology/missions#Goal_TransitSearch
        :Goal_TransitSearch rdf:type owl:NamedIndividual ;
                meo:hasNextOnFail :Goal_SearchForSailorAdrift ;
                meo:hasNextOnSucceed :Goal_TrackSailorAfloat ;
                meo:hasNextOnViolate :Goal_FailureDiagnosis .
```

# SPARQL Protocol and RDF Query Language

- "SPARQL is an RDF query language — that is, a semantic query language for databases — able to retrieve and manipulate data stored in Resource Description Framework (RDF) format." -- Wikipedia
- Standardized as World Wide Web (W3C) Recommendation
  - Open-source implementations that we use include Apache ARQ and Protégé tool.
- We use SPARQL to express queries against AVCL missions in RDF/OWL (Turtle syntax) together with Mission Execution Ontology (MEO).
- Results reveal interesting properties about missions that are otherwise difficult to determine. Inferences can also be combined and correlated.
- Goal is to express in-depth mission-related queries that determine
  - Whether all logical mission prerequisites and constraints are satisfied, and
  - Whether tactical policies and Rules of Engagement (ROE) are met.

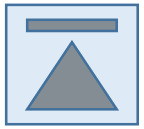
```
Perform Mission Execution Ontology metaquery MissionExecutionOntologyQuery_01.rq
to produce output file MissionExecutionOntologyQuery_01.rq.txt:

PREFIX :     <https://www.nps.edu/ontologies/MissionExecutionOntology/missions#>
PREFIX meo:  <https://www.nps.edu/ontologies/MissionExecutionOntology#>
PREFIX owl:  <http://www.w3.org/2002/07/owl#>
PREFIX rdf:  <http://www.w3.org/1999/02/22-rdf-syntax-ns#>
PREFIX rdfs: <http://www.w3.org/2000/01/rdf-schema#>
PREFIX xml:  <http://www.w3.org/XML/1998/namespace>
PREFIX xsd:  <http://www.w3.org/2001/XMLSchema#>

# @base <https://www.nps.edu/ontologies/MissionExecutionOntology/missions>

# MissionExecutionOntologyQuery_01.rq    Metaquery to list all properties
with corresponding domains and ranges in Mission Execution Ontology.

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

SELECT distinct ?property ?domain ?range
WHERE
{
    ?property rdfs:range  ?range .
    ?property rdfs:domain ?domain .
}
ORDER by ASC(?property) # alphabetize

################################################
```

Confirm MEO via self-centered SPARQL *metaquery*
MissionExecutionOntologyQuery_01.rq

| property | domain | range |
|---|---|---|
| meo:appliesTo | meo:Constraint | _:b0 |
| meo:canFulfill | meo:VehicleFeature | meo:GoalRequirement |
| meo:canIdentify | meo:Vehicle | meo:Constraint |
| meo:canMeet | meo:Vehicle | meo:Goal |
| meo:canPerform | meo:Vehicle | meo:Mission |
| meo:canTest | meo:VehicleFeature | meo:Constraint |
| meo:hasConstraint | meo:Mission | meo:Constraint |
| meo:hasEndCondition | meo:Goal | meo:EndCondition |
| meo:hasFeature | meo:Vehicle | meo:VehicleFeature |
| meo:hasNext | meo:Goal | meo:Goal |
| meo:hasNextOnFail | meo:Goal | |
| meo:hasNextOnSucceed | meo:Goal | |
| meo:hasNextOnViolate | meo:Goal | |
| meo:includes | meo:Mission | |
| meo:isFollowedBy | meo:Goal | |
| meo:isPerformableBy | meo:Mission | meo:Vehicle |
| meo:meetsRequirement | meo:Vehicle | meo:GoalRequirement |
| meo:requires | meo:Goal | meo:GoalRequirement |
| meo:startsWith | meo:Mission | meo:Goal |

Confirm MEO via self-centered SPARQL *metaquery* response

MissionExecutionOntologyQuery_01.rq.txt



# Ethical Control Mission Design

Life-saving missions and missions with lethal force are complementary. Human-robot activity can result in lethal or life-saving outcomes.

Continuing refinement and clarity are opening the path to repeatability. OODA Loop compatibility ensures harmonization with human operations.

# Core Considerations for Artificial Intelligence (AI)

- Effective AI turns data into information for use by humans.
- AI systems do not have capacity for rational thought or morality.
- Unmanned systems require sophisticated control across time, space.
- A large and involved body of internationally accepted law comprises Law of Armed Conflict (LOAC), bounding Rules of Engagement (ROE).
- Only professional warfighters have moral capacity, legal culpability, and societal authority to direct actions applying lethal force.
- Humans must be able to trust that systems under their direction will *do what they are told to do*, and *not do what they are forbidden to do*.
- Successful Ethical Control of unmanned systems must be testable.

# Not suitable for brute-force numerical computation

- AI algorithms for Machine Learning (ML) and Data Mining are often based on statistically training against large datasets to find patterns for filters.
  - For example, **convolutional neural networks**, **genetic algorithms**, **reinforcement learning**, etc.
- Often requires identifying right/wrong matches within large search spaces.
- Such predictive analytics are useful for classification models using detailed and noisy sensor data. Given the central importance of IFFNU and some conditional communications to ethical control, ML filters can be helpful if carefully applied.
- Nevertheless such approaches are not appropriate for carefully following Rules of Engagement (ROEs), Laws of Armed Conflict (LOAC) or other ethical prerequisites, especially when human expertise and judgement is essential for robot teams.
- (Similarly, massive computation or Quantum Computing approaches might be useful in some problems, but are not of practical use for Ethical Control mission orders given by human commanders judiciously guiding remote mobile robots)

Naval history has long shown that sound human judgement is crucial for assessing best strategies and courses of action in ill-structured contexts. Semantic Web approaches are preferable and actionable for Ethical Control.

# Conceptual Chunking: definition

- “In cognitive psychology, chunking is a process by which individual pieces of an information set are broken down and then grouped together.
- “A chunk is a collection of basic familiar units that have been grouped together and stored in a person's memory. These chunks are able to be retrieved more easily due to their coherent familiarity.
- “It is believed that individuals create higher order cognitive representations of the items within the chunk. The items are more easily remembered as a group than as the individual items themselves.
- “These chunks can be highly subjective because they rely on an individuals perceptions and past experiences, that are able to be linked to the information set. The size of the chunks generally range anywhere from two to six items, but often differ based on language and culture.” -- Wikipedia

# Conceptual Chunking for mission design

Command clarity is paramount when giving or receiving mission orders.

- Especially important for humans providing clear directions to teams.
- Must avoid danger of ambiguity or anthropomorphizing robots as human-like.

Multiple AVCL mission design aspects support chunking for clarity:

- Well-defined structured vocabulary describing hierarchy of distinct familiar goals.
- Orders that are readable by both humans and unmanned systems.
- Validatable as syntactically correct, no typos or gaps, avoids GIGO.
- Validatable as semantically correct, no prerequisite omissions or contradictions.
- Shown in this project: can group **goal tasks** within specific **phase definitions.**

***Future work:*** demonstrate best practices for common mission phases.

- Establish testable design patterns as reusable templates, aiding operator clarity.

# Mission clarity for humans – and robots

- Simplicity of success, failure, and (rare) exception outcomes encourages well-defined tasks and unambiguous, measurable criteria for continuation.

Confirmable beforehand: can a tactical officer (or commanding officer) review such a mission and then confidently say

  - "yes I understand and approve this human-robot mission" or, equivalently,
  - "yes I understand this mission and my team can carry it out themselves."

Converse:

- if an officer can't fully review/understand/approve such a mission, then likely it is **ill-defined** and needs further clarification anyway.

Added benefit: missions that are clearly readable/runnable by humans and robots can be further composed and checked by C2 planning tools to test for group operational-space management, avoiding mutual interference, etc.

# Wrong question, right question

Wrong question to ask first when planning a tactical operation:

- "What are my *robots* doing out there?"

Right question to ask first when planning a tactical operation:

- "What is my *human-robot team* doing out there?"

**Human-robot team mission has to be understood first!**

- Robots complement humans, who must remain in charge throughout.
- If you don't have an OODA loop, you don't have a competent plan.

# Observe Orient Decide Act (OODA) Loop

- *"The OODA loop is the cycle observe–orient–decide–act, developed by military strategist and USAF Colonel John Boyd. Boyd applied the concept to the combat operations process, often at the operational level during military campaigns. It is now also often applied to understand commercial operations and learning processes. The approach explains how agility can overcome raw power in dealing with human opponents."* – Wikipedia
- All effective purposeful military activity can be conceived in terms of OODA loop feedback process, especially at tactical/operational levels.
- Aligning Ethical Control mission design with OODA loop ensures that unmanned systems understandably partner within human-run teams.

# OODA Significance for Ethical Control

**Classical robotic Sense-Decide-Act cycle for closed-loop control is insufficient for proper delegation of lethal (or lifesaving) force to unmanned systems.**

**Observe-Orient-Decide-Act (OODA) Loop is essential for coherent operations.**

- *Observe* includes direct sensing and communication inputs.
- *Orientation* includes thorough Rules of Engagement (ROE) constraints and identification friend/foe/neutral/unknown (IFFNU) of all relevant contacts.
- *Decision* logic of unmanned system tactics, techniques, procedures (TTP) includes authorization and confirmation by human supervisors, either in real-time or in advance, for critical steps leading to lethal force.
- *Actions* in tandem with direct or intermittent human supervisory command enables effective Ethical Control of remote systems.

Feedback loops are essential, generally leading to… *more effective operations.*

# OODA Clarity for Mission Definitions

Robot activity must complement, not contradict, human OODA loop.

- Application of ROE and LOAC requirements can be confirmed present as part of mission definition, typically as
  - Goal success/failure criteria, preset authorities or time-outs for delegation, and
  - Constraints on conduct (safe zones, permission periods/requirements, etc.)
- When human leaders confirm correct inclusion of ROE requirements in mission orders, they essentially perform an audit of doctrine and TTPs.
- Similar audit confirmation can be applied to well-structured orders:
  - Autonomous Vehicle Command Language (AVCL) expresses well-defined goals,
  - Mission Execution Ontology (MEO) defines relationships and requirements,
  - SPARQL queries can be written to perform such logical-confirmation checks.
- Resulting mission orders are thus coherent from OODA perspective.

# Loop management: principles

Proper algorithms always includes one or more termination conditions.

- Sequence of operations proceeds through a finite number of steps, otherwise the system is performing an infinite loop without end.
- Infinite-loop sequencing or recursion is a common computational failure mode and must be protected against for reliable operations.

Nevertheless a frequent characteristic of at-sea operations is to perform repeated tasks in an interactive fashion until complete…

- Either via task success, or a terminating condition that forces halting.

The presence of termination conditions can be verified in mission logic and also tested in simulation.

# Loop management: practice

Example terminating conditions, some fixed/adaptive and some iterative/exceptional:

- Reach maximum number of iterations.
- Point of diminishing returns (e.g. unchanging search effectiveness).
- Time-out deadline reached, or else no longer feasible to continue.
- Insufficient power remains, conduct graceful shutdown for recovery.
- Equipment damage or unexpected software failure; log and shutdown.
- Interfering operational conditions (e.g. potential hazard to friendlies).
- Human direction asserts higher priority and overrides decision logic.

***Future work:*** expressing all these constraints precisely in AVCL missions.

# Canonical Ethical Mission Development 1

Progressive sophistication to test and evaluate Ethical Control design.

a. *Sailor Overboard* for life-saving force under close coordination,
b. *Lifeboat Tracking* for life-saving force under remote conditions,
c. *Pirate Boats Seizing Merchant* for steady escalation to lethal force,
d. *Robot Swarm Attacks Hospital Ship* to prevent hostile provocation of sense-decide-act vulnerabilities through Ethical Control safeguards. Two variants show fundamental importance of ethical constraints.

Demonstrating a range of functionality tests majority of Ethical Control capabilities that are currently envisioned. These canonical capabilities also "set the stage" for further work on advanced problems of interest.

# OODA Loops for Ethical Control Canonical Missions

| Ethical Control OODA Loops | Observe | Orient | Decide | Act |
|---|---|---|---|---|
| Sailor Overboard | Find Sailor | Report status | Avoid interference | Track sailor until rescued or relieved |
| Lifeboat Rescue | Find Lifeboat | Report status | Two-way communication | Track life raft until relieved |
| Pirate Seizure of Merchant Ship | Find merchant ship, pirate small boats | Identity Friend Foe Neutral Unknown (IFFNU)<br>Issue warnings | Human commander authorization to use lethal force | Attack to defend ship if provoked, stay with merchant |
| Hospital Ship Swarm Attack | EM threat signals detected | (no orientation step in Sense Decide Act) | Reflex-response weapons attack | Mistaken attack on friendly = war crime |
| Hospital Ship Defense detects spoofing anti-pattern | EM threat signals detected | IFFNU including correlation | Human requirement for lethal force unmet, attack avoided | Report threat alert, commence search for hostile actors |

# Canonical Ethical Mission Development 2

Next steps: completing this initial set of missions is expected to

- Demonstrate both logical soundness and human comprehensibility of Ethical Control methodology, for both life-saving and lethal force.
- “Tune up” AVCL vocabulary for mission orders, revealing good practices and repeatable design patterns for common activities in diverse missions.
- Translation of AVCL missions into corresponding MEO representations to allow deeper semantic validation of correctness of ethical constraints.
- Performing AVCL missions in AUV Workbench tool is expected to show mission execution working in simulation, or else reveal hidden flaws.
- ***Future work:*** prepare for comprehensive testing of unmanned systems across range of requirements necessary for afloat operational qualification.

# Tactical span of control for ship with many robots

- Span of control is number of subordinates reporting to a supervisor.
  - Classic definition from business management, also applies to leadership.
- In effect, multiple offboard unmanned systems supervised by a ship comprise its span of control across the tactical battlespace.
- Greater tactical presence across distances of time and space means ship commanders have greater ability to influence their local theater.
- Clear mission guidance on human-checkpoint requirements reduces dependency on communication links (i.e. Network Optional Warfare).
- Next diagram illustrates how such increased ability to project power enables ship to maintain chosen standoff location while nevertheless focusing direct attention and actions in multiple locations at once.

Life boat

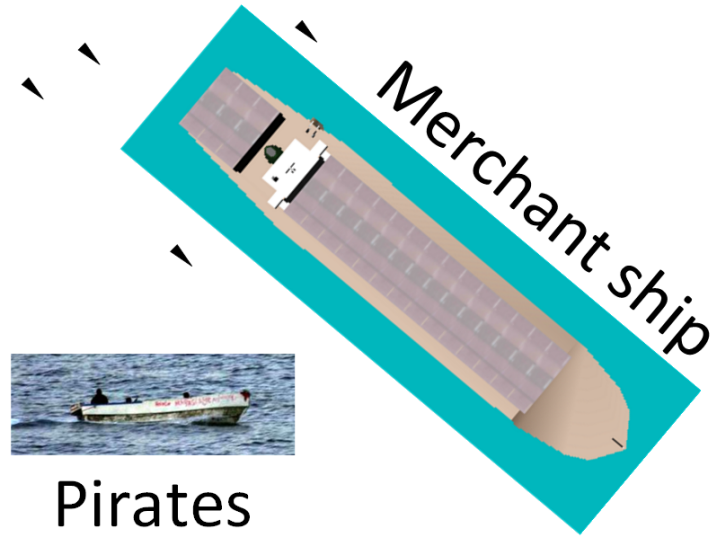
**Life-saving force: locate, track, communicate, beacon**

# Response dilemma for U.S. Navy ship

Ethical control of unmanned systems is required for both lethal and lifesaving force if remote robots communicate intermittently, operating across lengthy time and distance.

**Lethal force: locate, warn, defend, threaten, attack**

Respond to one or both scenarios with USV/UAV assets to establish on-scene visibility and presence



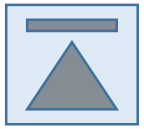



Pirates

# Exemplar Missions

Unmanned systems working in tandem with human forces, authorized by commander for life-saving or lethal force, can handle progressive challenges in distance and time.

# Catalog of Exemplar Missions

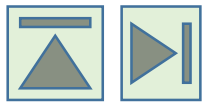


Structure for each section

- Design and Description
- Mission decision-flow diagram
- AVCL implementation source: version control, XML Spy table
- Semantic Web Mission MEO .ttl Turtle representation
- SPARQL semantic queries
- Lisp and Prolog autocode
- AUVW simulation

- Sailor Overboard
- Lifeboat Tracking
- Pirates Seizing Merchant Ship
- Hospital Ship EM Decoy Scenario compares Sense-Decide-Act vs. Observe-Orient-Decide-Act (OODA)

# Sailor Overboard Mission: Design



**Motivations**

- Explore how *life-saving force* is complementary to *lethal force*, with many similar considerations for remote supervision.
- First demonstrate human-system teaming in close proximity to ship, where direct override of robot control by human operator is possible.
- No temporal delays, all actions and reactions must be immediate.
- Show that mission design can complement shipboard procedures.

**Lessons learned**

- Good design patterns to visually represent AVCL mission satisfactorily.
- Temporal flow (left to right, e.g. Gantt chart) best suited for diagram.

# Sailor Overboard Mission: Description

**Purpose**

- Life saving: single unmanned air/surface vehicle actions to complement human responses when performing *“SAILOR OVERBOARD”* operations.
- Carried out in direct concert with formal *shipboard emergency procedures*.
- Multiple UAVs/USVs might be employed in parallel with ships and aircraft, avoid mutual interference by each following deconflicted mission orders.

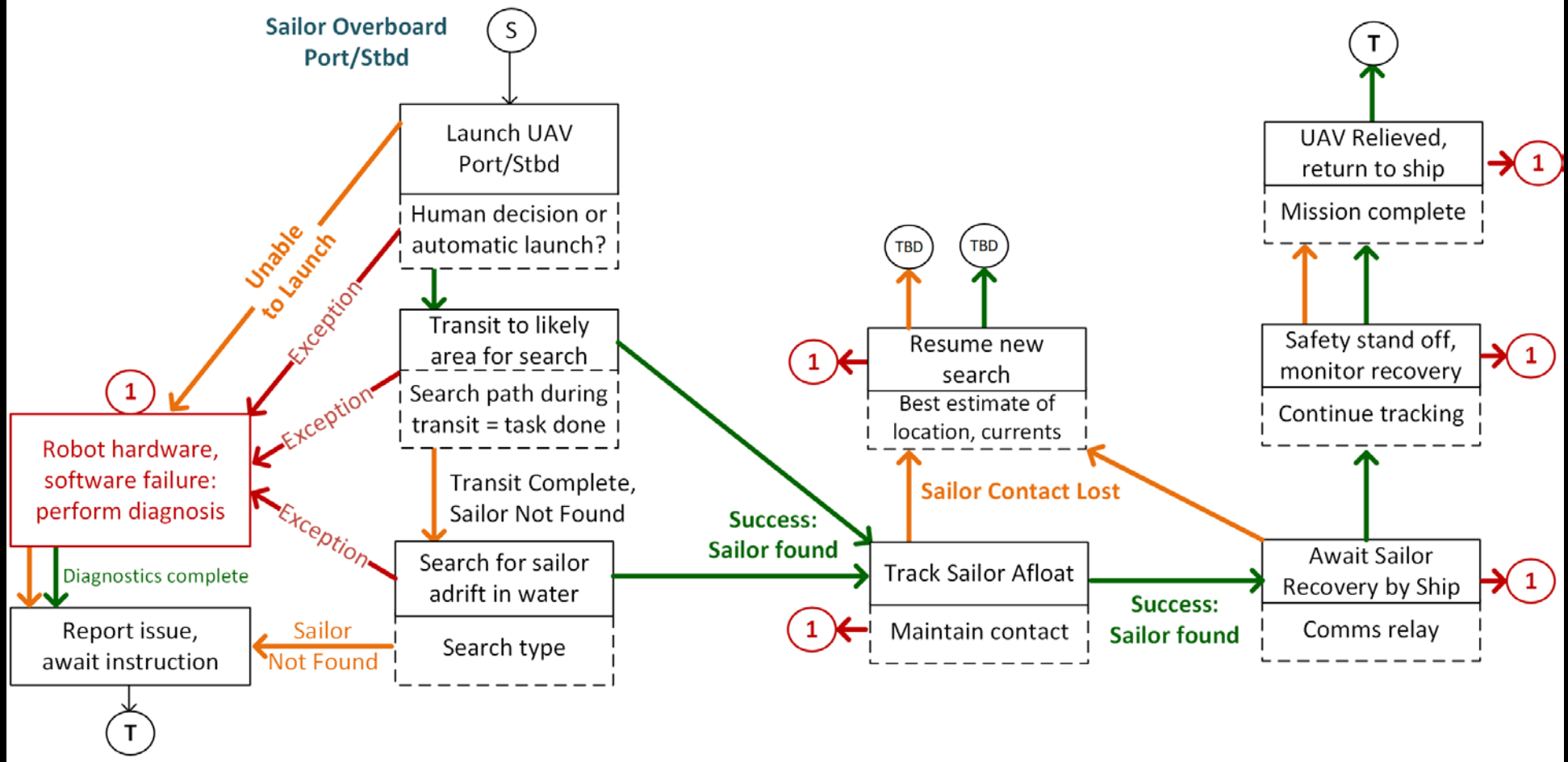
**Phases**

- Deploy/Launch, Rendezvous, Track Sailor until Safe, Return/Recovery.

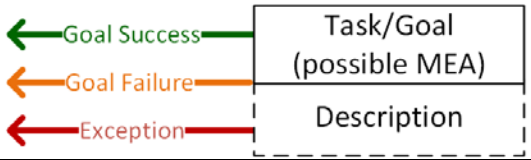
**Human Supervisory Role**

- Order standoff if interfering, manual control is possible due to proximity, can communicate to sailor via loudspeaker or beacon light.

Single unmanned vehicle responses to complement existing **shipboard emergency procedures.**
Multiple UAVs might be employed that each follow these mission orders, in parallel.



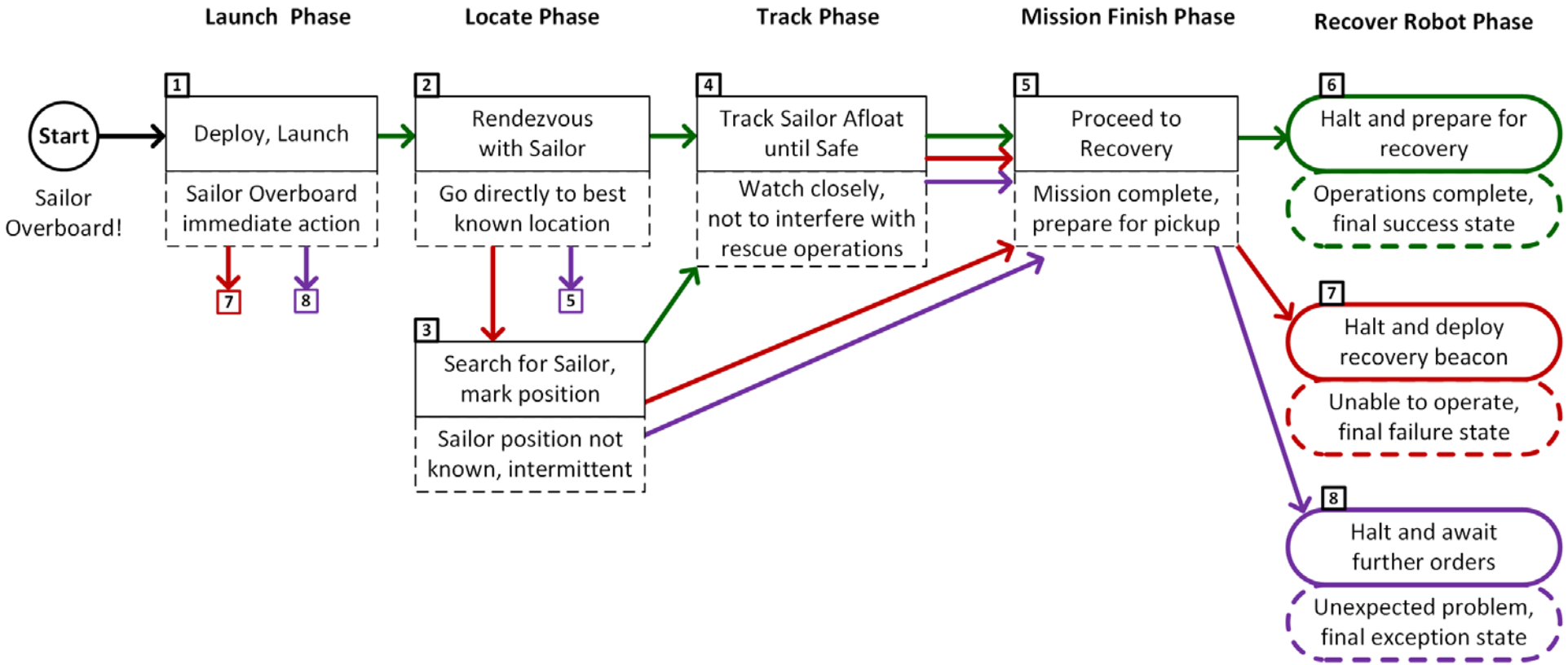

**Original Version**



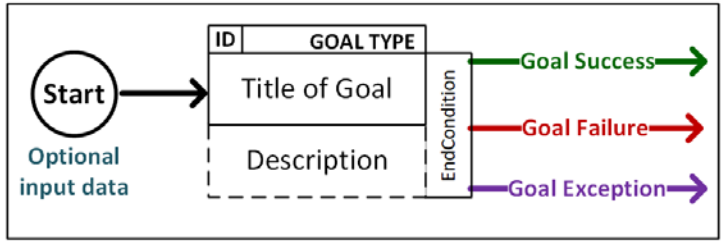

Don Brutzman and Bob McGhee
12 AUG 2019

# Sailor Overboard, 8 Phases – Mission Execution Automaton (MEA)

Single unmanned air/surface vehicle actions to complement human response when performing "**SAILOR OVERBOARD**" operations, carried out in concert with **shipboard emergency procedures**. Multiple UAVs/USVs can be employed in parallel with ships/aircraft, each following mission orders.




Legend





Don Brutzman and Bob McGhee
Mission upgrade 19 NOV 2019

# SailorOverboard.xml in gitlab.nps.edu version control

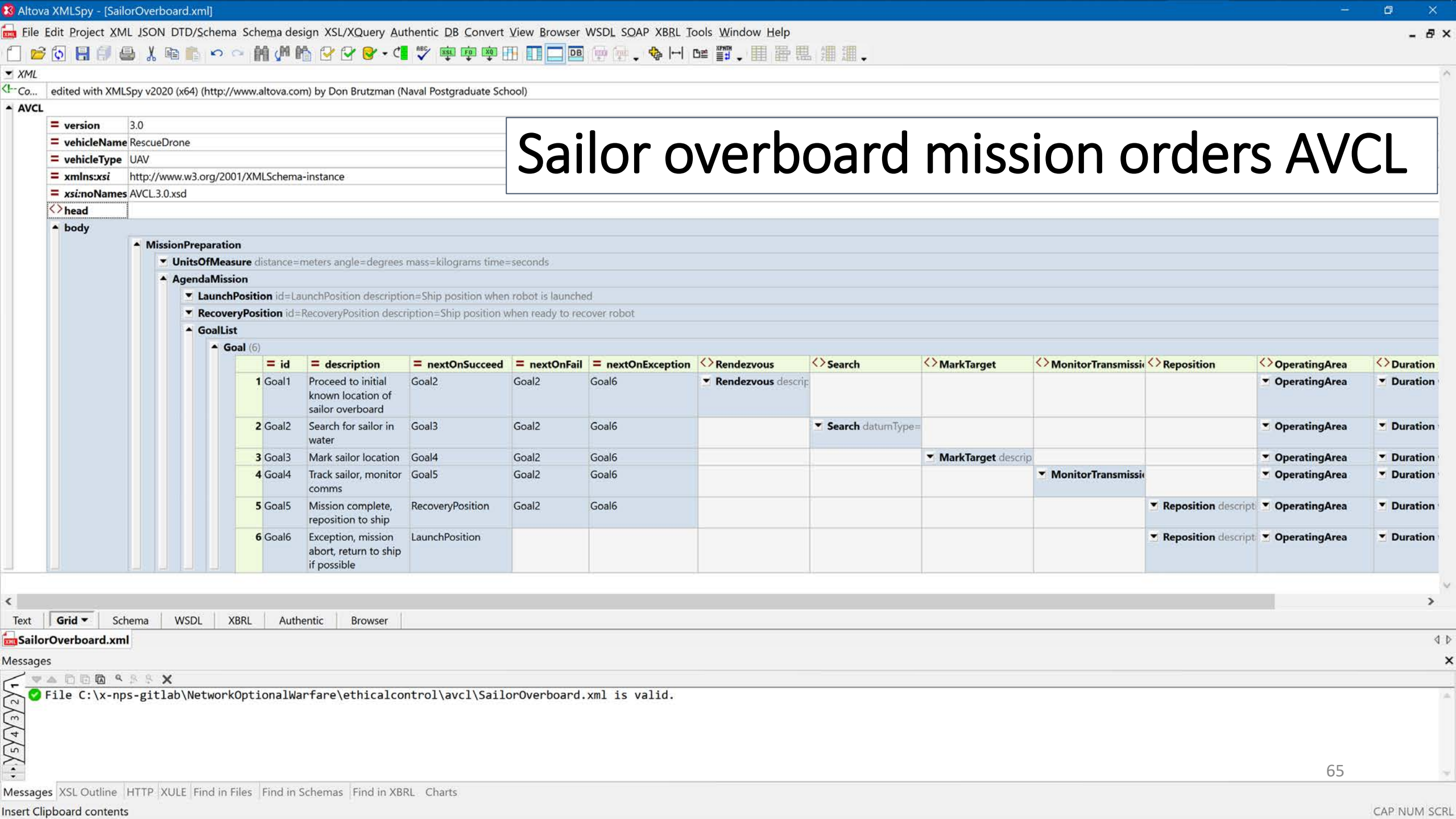
https://gitlab.nps.edu/Savage/EthicalControl/blob/master/missions/avcl/SailorOverboard.xml

GitLab Projects Groups More Search or jump to...

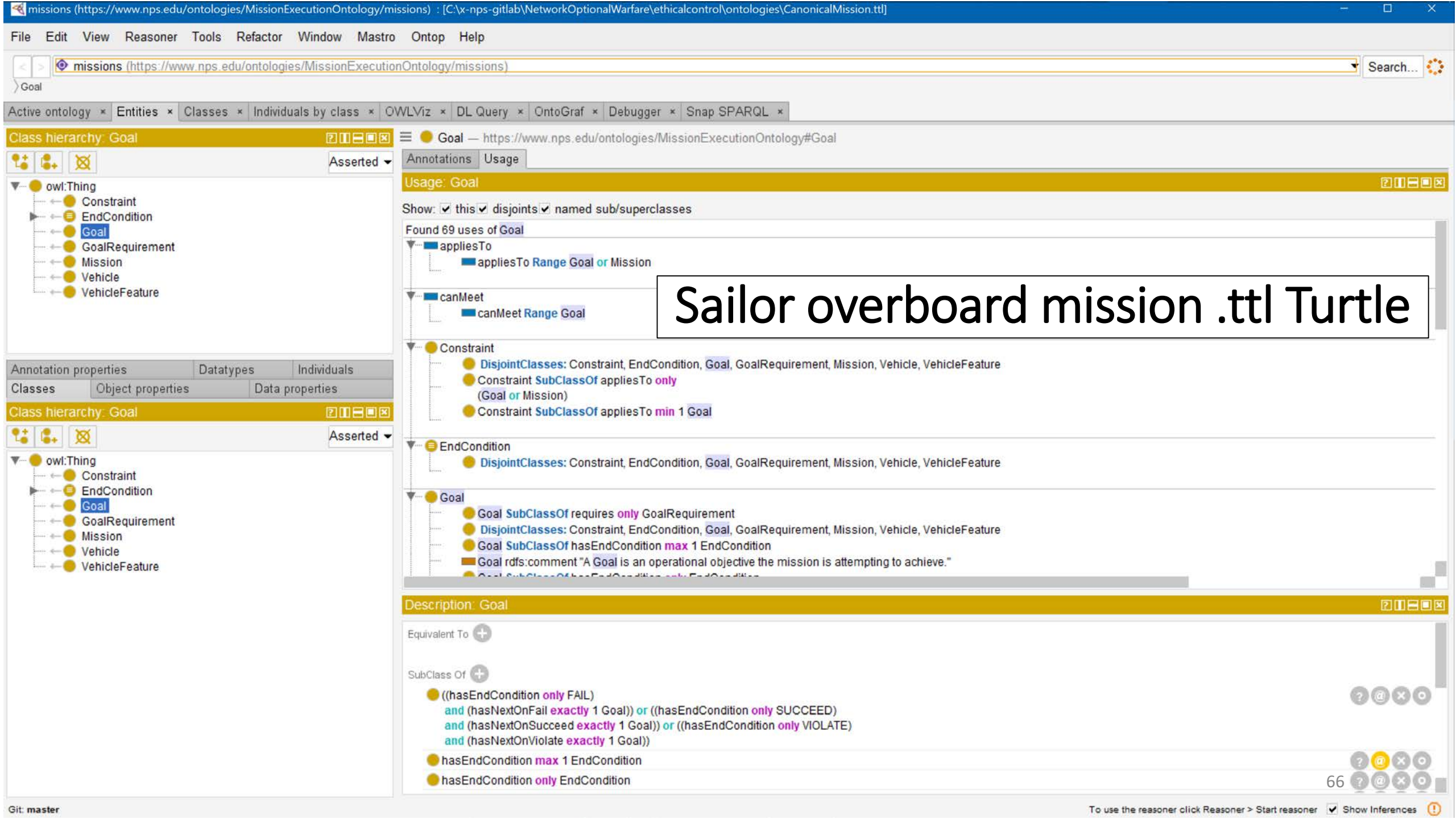
**SailorOverboard.xml** 5.63 KB

Edit | Web IDE | Replace | Delete

```
<?xml version="1.0" encoding="UTF-8"?>
<AVCL version="3.0" vehicleName="RescueDrone" vehicleType="UAV" xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance" xsi:noNamespaceSchemaLocation="../../avcl/AVCL.3.0.xsd">
	<head>
		<meta name="title" content="SailorOverboard.xml"/>
		<meta name="created" content="19 November 2019"/>
		<meta name="modified" content="31 December 2019"/>
		<meta name="creator" content="Don Brutzman"/>
		<meta name="reference" content="https://wiki.nps.edu/display/NOW/Ethical+Control+of+Unmanned+Systems"/>
		<meta name="reference" content="https://gitlab.nps.edu/Savage/EthicalControl/raw/master/missions/SailorOverboard8PhaseMission.png"/>
		<meta name="identifier" content="https://gitlab.nps.edu/Savage/EthicalControl/tree/master/missions/avcl/SailorOverboard.xml"/>
		<meta name="generator" content="Altova XMLSpy, https://www.altova.com"/>
		<meta name="generator" content="Apache NetBeans, https://netbeans.apache.org"/>
		<meta name="generator" content="NPS Autonomous Unmanned Vehicle (AUV) Workbench, https://savage.nps.edu/AuvWorkbench"/>
		<meta name="license" content="../license.html"/>
	</head>
	<body>
		<MissionPreparation>
			<UnitsOfMeasure distance="meters" angle="degrees" mass="kilograms" time="seconds"/>
			<AgendaMission>
				<LaunchPosition id="LaunchPosition" description="Ship position when robot is launched">
					<LatitudeLongitude latitude="0" longitude="0"/>
				</LaunchPosition>
				<RecoveryPosition id="RecoveryPosition" description="Ship position when ready to recover robot">
					<LatitudeLongitude latitude="0" longitude="0"/>
				</RecoveryPosition>
				<GoalList>
					<Goal id="Goal1" title="Deploy, Launch" description="Sailor Overboard immediate Action" nextOnSucceed="Goal2" nextOnFail="Goal2" nextOnException="Goa
						<Rendezvous description="">
							<TargetVehicleID value="101" description="Sailor in water"/>
						</Rendezvous>
						<OperatingArea>
							<Point>
								<LatitudeLongitude latitude="36.62" longitude="121.506"/>
<!--36°36´11˝N 121°53´37˝W-->
							</Point>
						</OperatingArea>
						<Duration value="300"/>
```




# Sailor overboard mission orders AVCL




# Sailor overboard mission .ttl Turtle

https://gitlab.nps.edu/Savage/EthicalControl/blob/master/missions/turtle/SailorOverboardConverted.ttl

More

**SailorOverboardConverted.ttl** 2.3 KB

```
### Namespace declarations
@prefix :     <https://www.nps.edu/ontologies/MissionExecutionOntology/missions#> .
@prefix meo:  <https://www.nps.edu/ontologies/MissionExecutionOntology#> .
@prefix owl:  <http://www.w3.org/2002/07/owl#> .
@prefix rdf:  <http://www.w3.org/1999/02/22-rdf-syntax-ns#> .
@prefix rdfs: <http://www.w3.org/2000/01/rdf-schema#> .
@prefix xml:  <http://www.w3.org/XML/1998/namespace> .
@prefix xsd:  <http://www.w3.org/2001/XMLSchema#> .

@base <https://www.nps.edu/ontologies/MissionExecutionOntology/missions> . # TODO consider URI matching URL

### Import base ontology
<https://www.nps.edu/ontologies/MissionExecutionOntology/missions> rdf:type owl:Ontology ;
owl:imports <https://www.nps.edu/ontologies/MissionExecutionOntology> .

#################################################################
#    Individuals
#################################################################

:SailorOverboard rdf:type owl:NamedIndividual , owl:Thing ;
  meo:startsWith :Goal1 ;
  rdfs:comment "AVCL mission to perform SAILOR OVERBOARD operations, carried out in concert with shipboard emergency procedures." .

:Goal1 rdf:type owl:NamedIndividual , owl:Thing ;
  meo:hasNextOnFail :Goal2 ;
  meo:hasNextOnSucceed :Goal2 ;
  meo:hasNextOnViolate :Goal5 ;
  rdfs:comment "Deploy, Launch" .

:Goal2 rdf:type owl:NamedIndividual , owl:Thing ;
  meo:hasNextOnFail :Goal3 ;
  meo:hasNextOnSucceed :Goal4 ;
  meo:hasNextOnViolate :Goal5 ;
  rdfs:comment "Rendezvous with Sailor" .

:Goal3 rdf:type owl:NamedIndividual , owl:Thing ;
  meo:hasNextOnFail :Goal5 ;
  meo:hasNextOnSucceed :Goal4 ;
  meo:hasNextOnViolate :Goal5 ;
  rdfs:comment "Search for Sailor " .
```

SailorOverboardConverted.ttl
Turtle excerpt autogenerated in gitlab.nps.edu version control

first mission query MissionGoalsQuery_01 - hooray!
Brutzman, Don authored 1 hour ago

**MissionGoalsQuery_01.rq** 1.7 KB

```
PREFIX :     <https://www.nps.edu/ontologies/MissionExecutionOntology/missions#>
PREFIX meo:  <https://www.nps.edu/ontologies/MissionExecutionOntology#>
PREFIX owl:  <http://www.w3.org/2002/07/owl#>
PREFIX rdf:  <http://www.w3.org/1999/02/22-rdf-syntax-ns#>
PREFIX rdfs: <http://www.w3.org/2000/01/rdf-schema#>
PREFIX xml:  <http://www.w3.org/XML/1998/namespace>
PREFIX xsd:  <http://www.w3.org/2001/XMLSchema#>

# @base <https://www.nps.edu/ontologies/MissionExecutionOntology/missions>

# MissionGoalsQuery_01.rq    Query to list all goals with corresponding description information and sequencing logic.

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

SELECT ?goal ?nextOnSucceed ?nextOnFail ?nextOnViolate ?isPartOfPhase ?description # ?GoalFound ?phase

WHERE
{
    ?goal    a meo:Goal ; # Shorthand expression: a = rdf:type
             meo:isPartOfPhase      ?isPartOfPhase ; # TODO what about when no value is provided
             meo:hasNextOnSucceed   ?nextOnSucceed ;
             meo:hasNextOnFail      ?nextOnFail ;
             meo:hasNextOnViolate   ?nextOnViolate ;# TODO rename as Exception
             rdfs:comment           ?description .

    # https://stackoverflow.com/questions/11234371/sparql-query-results-without-namespace
    BIND (strafter(xsd:string(?goal),"#")            AS ?GoalFound)
    BIND (strafter(xsd:string(?nextOnSucceed),"#")   AS ?GoalNextOnSucceed)
    BIND (strafter(xsd:string(?nextOnFail),"#")      AS ?GoalNextOnFail)
    BIND (strafter(xsd:string(?nextOnViolate),"#")   AS ?GoalNextOnViolate)
#   BIND (coalesce(?isPartOfPhase,"")                AS ?phase)
}
ORDER BY (?GoalFound) # alphanumeric order results in order given by each name
################################################
```

# SPARQL mission query

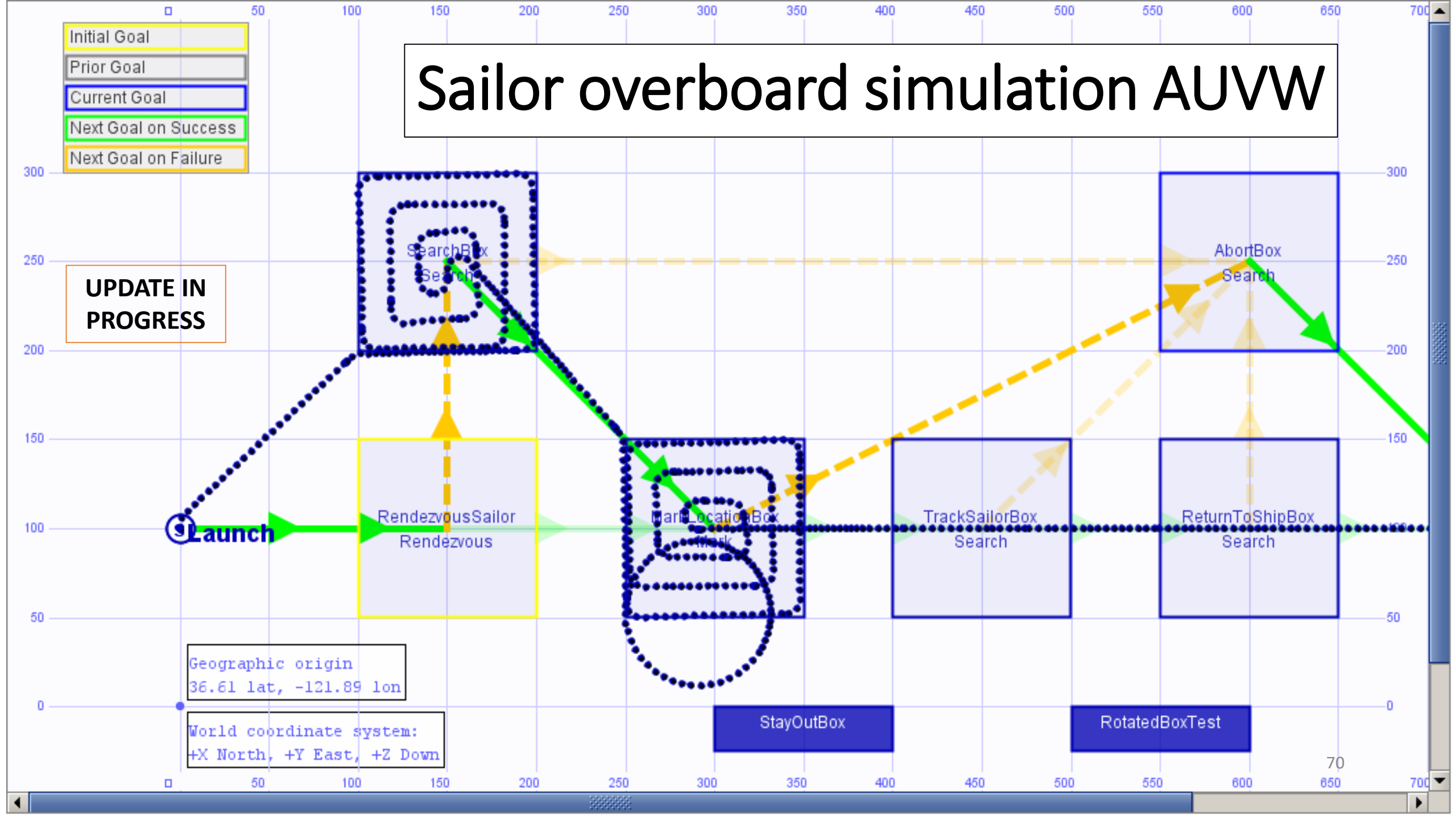
# MissionQuery_01_GoalBranches.rq

# SPARQL query response
## SailorOverboardConverted. MissionQuery_01_GoalBranches.rq.txt

| goal | nextOnSuccess | nextOnFailure | nextOnException | isPartOfPhase | description |
|---|---|---|---|---|---|
| :Goal1 | :Goal2 | :Goal2 | :Goal5 | "Launch" | "Deploy, Launch: Sailor Overboard Immediate Action" |
| :Goal2 | :Goal4 | :Goal3 | :Goal5 | "Locate" | "Rendezvous with Sailor: Go directly to best known location" |
| :Goal3 | :Goal4 | :Goal5 | :Goal5 | "Locate" | "Search for Sailor: Sailor position not known, intermittent" |
| :Goal4 | :Goal5 | :Goal5 | :Goal5 | "Track" | "Track Sailor afloat until safe: Watch closely, not to interfere with rescue operations" |
| :Goal5 | :Goal6 | :Goal2 | :Goal6 | "Mission Finish" | "Proceed to Recovery: Mission complete, prepare for pickup" |
| :Goal6 | | | | "Recover Robot" | "Halt and prepare for recovery: Operations complete, final success state" |
| :Goal7 | | | | "Recover Robot" | "Halt and deploy recovery beacon: Unable to continue, final failure state" |
| :Goal8 | | | | "Recover Robot" | "Halt and await further orders: Unexpected problem, final exception state" |

# SPARQL query response
## SailorOverboardConverted. MissionQuery_02_InitialGoal.rq.txt

| Mission | InitialGoal | isPartOfPhase | goalDescription |
|---|---|---|---|
| :SailorOverboard | :Goal1 | "Launch" | "Deploy, Launch: Sailor Overboard Immediate Action" |


Initial Goal
Prior Goal
Current Goal
Next Goal on Success
Next Goal on Failure
Sailor overboard simulation AUVW
UPDATE IN PROGRESS
SearchBox
Search
AbortBox
Search
Launch
RendezvousSailor
Rendezvous
Mark_LocationBox
Mark
TrackSailorBox
Search
ReturnToShipBox
Search
Geographic origin
36.61 lat, -121.89 lon
World coordinate system:
+X North, +Y East, +Z Down
StayOutBox
RotatedBoxTest

# Lifeboat Tracking Mission: Design

**Motivations**

- Similar to Sailor Overboard in demonstrating use of life-saving force.
- Far greater distances, over the horizon, increases ship span of control.
- Potential for intermittent or lost communications in real time, requires advance guidance for default behaviors desired by human controller.
- Consider possible transfer of supervisory control mid-mission to another cooperating vessel – appears feasible.

**Lessons learned**

- Vertical grouping of related subtasks helps in structuring goal sets, without requiring change to ternary logic of AVCL mission goals.
- Coexistence of multiple constraints is possible, requires careful thought.

# Lifeboat Tracking Mission: Description

**Purpose**

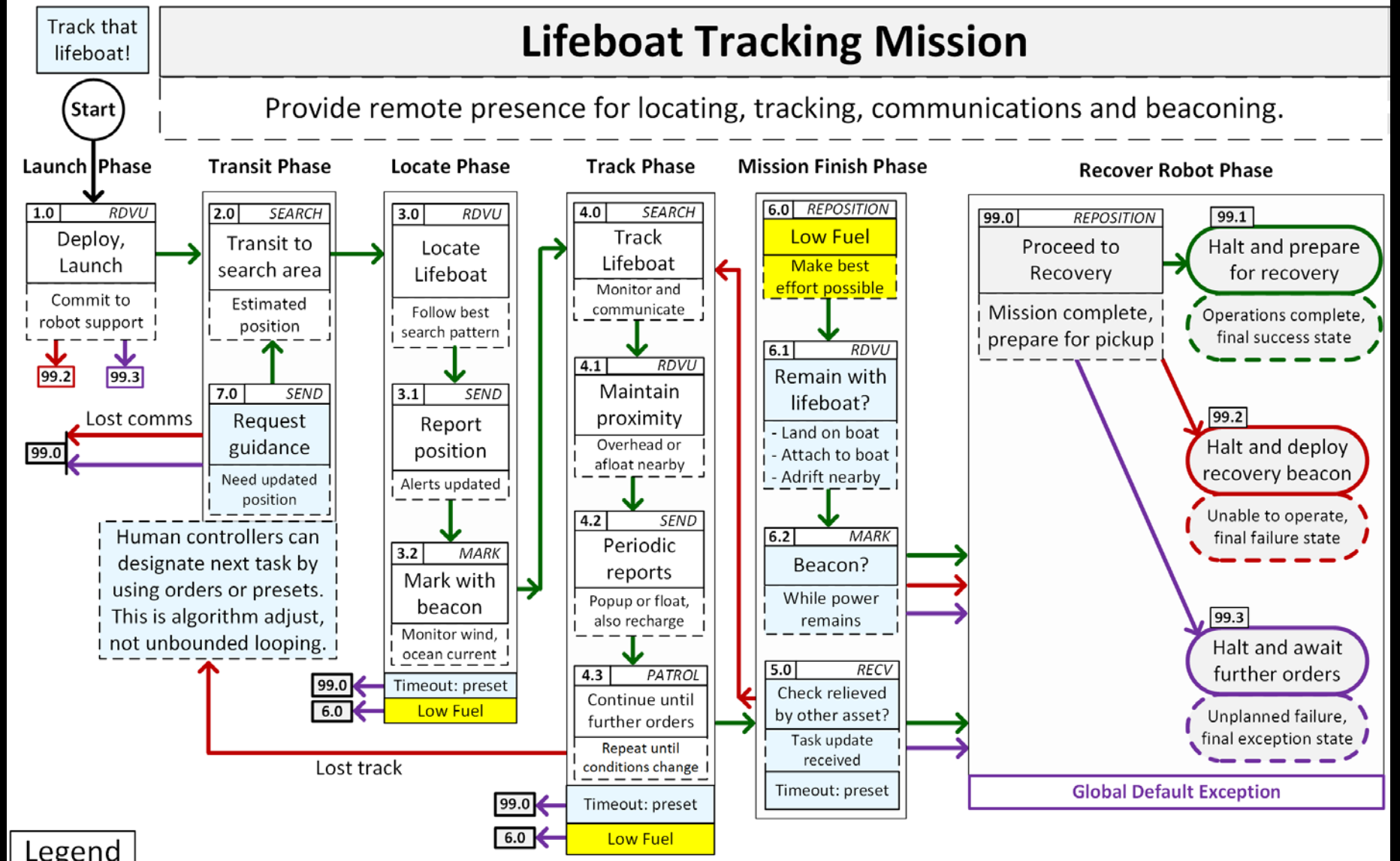
- Provide remote presence for locating, tracking, communications and beaconing.

**Phases**

- Deploy/Launch, Rendezvous, Track Lifeboat, Beacon/Communicate, Return/Recovery.

**Human Supervisory Role and Constraints**

- Monitor, communicate, respond or coordinate rescue effort.
- Low fuel condition and graceful-degradation response.


Track that lifeboat!
Lifeboat Tracking Mission
Provide remote presence for locating, tracking, communications and beaconing.
Start
Launch Phase
Transit Phase
Locate Phase
Track Phase
Mission Finish Phase
Recover Robot Phase
1.0 RDVU
Deploy, Launch
Commit to robot support
99.2
99.3
2.0 SEARCH
Transit to search area
Estimated position
7.0 SEND
Request guidance
Need updated position
Lost comms
99.0
3.0 RDVU
Locate Lifeboat
Follow best search pattern
3.1 SEND
Report position
Alerts updated
3.2 MARK
Mark with beacon
Monitor wind, ocean current
Timeout: preset
Low Fuel
99.0
6.0
4.0 SEARCH
Track Lifeboat
Monitor and communicate
4.1 RDVU
Maintain proximity
Overhead or afloat nearby
4.2 SEND
Periodic reports
Popup or float, also recharge
4.3 PATROL
Continue until further orders
Repeat until conditions change
Timeout: preset
Low Fuel
99.0
6.0
Lost track
Human controllers can designate next task by using orders or presets. This is algorithm adjust, not unbounded looping.
6.0 REPOSITION
Low Fuel
Make best effort possible
6.1 RDVU
Remain with lifeboat?
- Land on boat
- Attach to boat
- Adrift nearby
6.2 MARK
Beacon?
While power remains
5.0 RECV
Check relieved by other asset?
Task update received
Timeout: preset
99.0 REPOSITION
Proceed to Recovery
Mission complete, prepare for pickup
99.1
Halt and prepare for recovery
Operations complete, final success state
99.2
Halt and deploy recovery beacon
Unable to operate, final failure state
99.3
Halt and await further orders
Unplanned failure, final exception state
Global Default Exception
Legend
Start
Optional input data
ID GOAL TYPE
Title of Goal
Description
Constraints
Goal Success
Goal Failure
Goal Exception
Default Condition Transitions
Goal Success condition must be defined for non-terminal Goals
If no Failure condition defined, then Failure matches Success
If no Exception defined, then Exception condition matches Global Exception or else Failure
Human orders
Guidance or presets needed
Terminal States
S Success
F Failure
X Exception

# LifeboatTracking.xml in gitlab.nps.edu version control

https://gitlab.nps.edu/Savage/EthicalControl/blob/master/missions/avcl/LifeboatTracking.xml

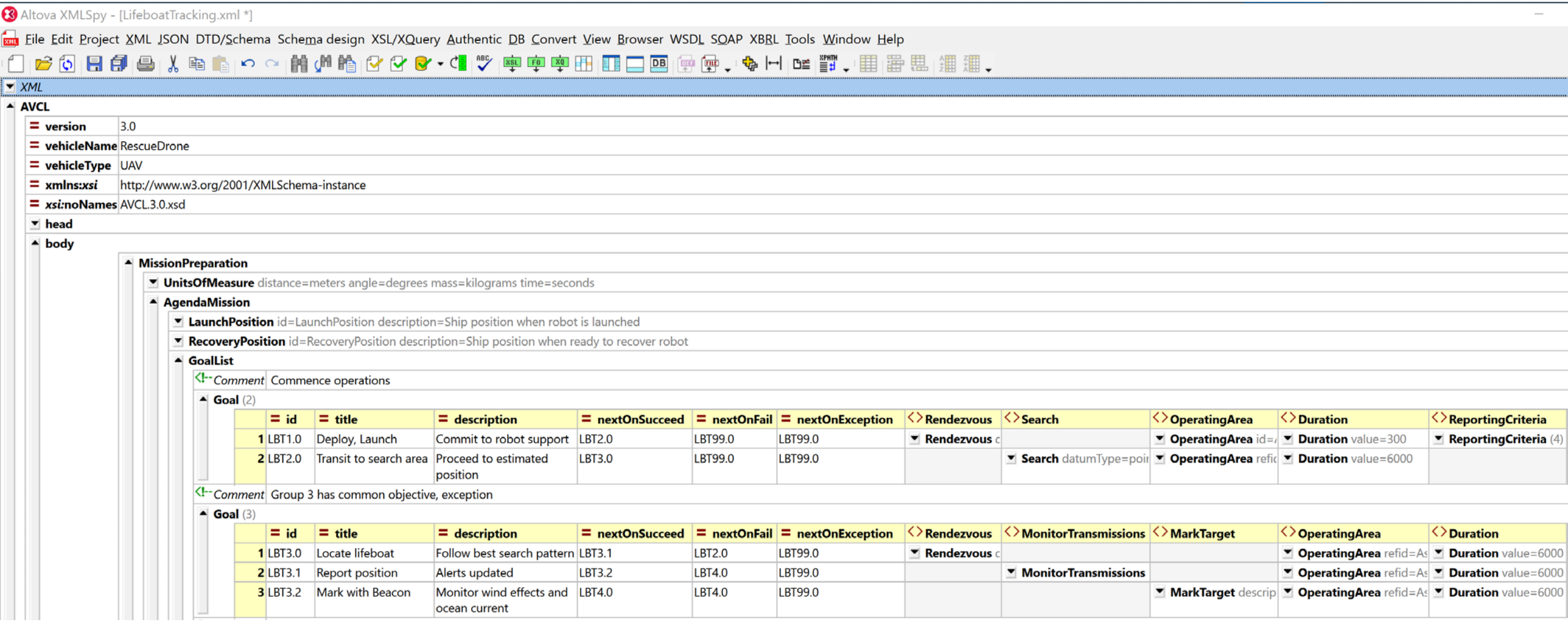
GitLab Projects Groups More Search or jump to...

**LifeboatTracking.xml** 9.32 KB

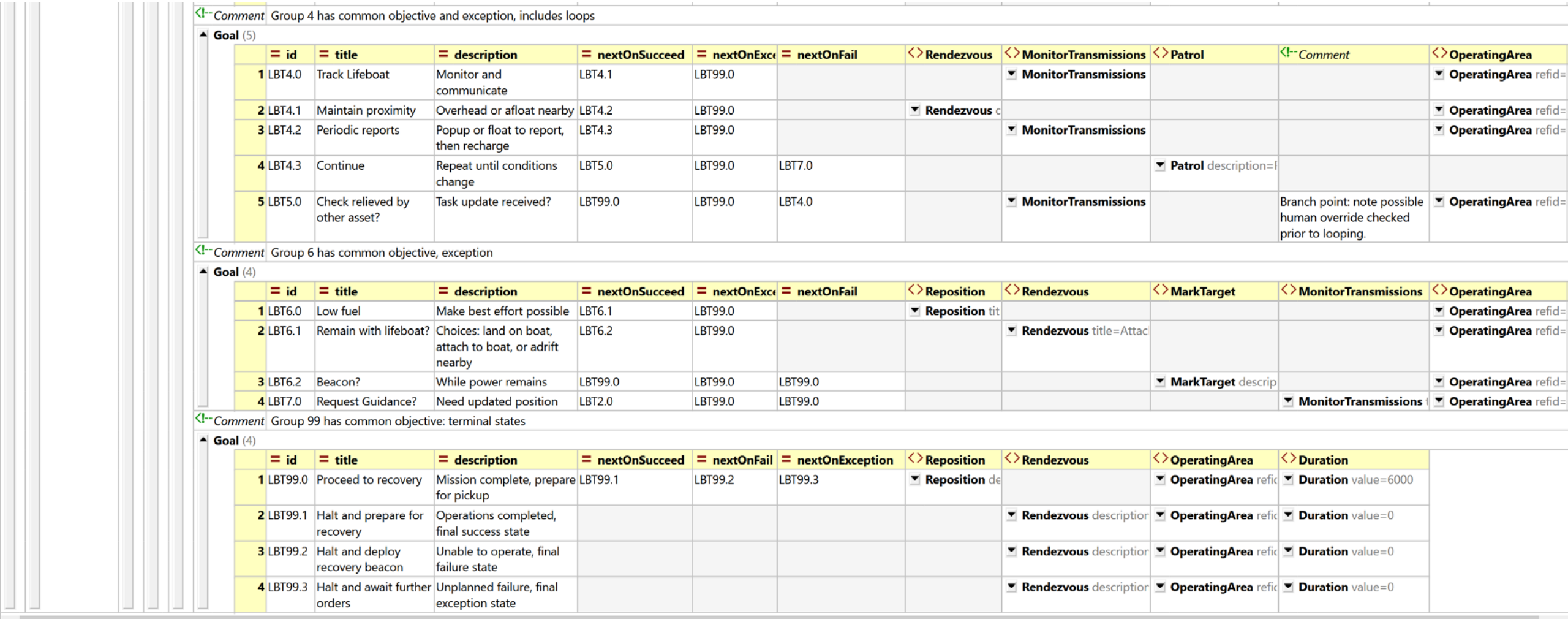
Edit | Web IDE | Replace | Delete

```
<?xml version="1.0" encoding="UTF-8"?>
<AVCL version="3.0" vehicleName="RescueDrone" vehicleType="UAV" xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance" xsi:noNamespaceSchemaLocation="../../avcl/AVCL.3.0.xsd">
	<head>
		<meta name="title" content="LifeboatTracking.xml"/>
		<meta name="created" content="26 December 2019"/>
		<meta name="modified" content="31 December 2019"/>
		<meta name="creator" content="Don Brutzman"/>
		<meta name="reference" content="https://wiki.nps.edu/display/NOW/Ethical+Control+of+Unmanned+Systems"/>
		<meta name="reference" content="https://gitlab.nps.edu/Savage/EthicalControl/raw/master/missions/LifeboatTrackingMission.png"/>
		<meta name="identifier" content="https://gitlab.nps.edu/Savage/EthicalControl/tree/master/missions/avcl/LifeboatTracking.xml"/>
		<meta name="generator" content="Altova XMLSpy, https://www.altova.com"/>
		<meta name="generator" content="Apache NetBeans, https://netbeans.apache.org"/>
		<meta name="generator" content="NPS Autonomous Unmanned Vehicle (AUV) Workbench, https://savage.nps.edu/AuvWorkbench/"/>
		<meta name="license" content="../license.html"/>
	</head>
	<body>
		<MissionPreparation>
			<UnitsOfMeasure distance="meters" angle="degrees" mass="kilograms" time="seconds"/>
			<AgendaMission>
				<LaunchPosition id="LaunchPosition" description="Ship position when robot is launched">
					<LatitudeLongitude latitude="0" longitude="0"/>
				</LaunchPosition>
				<RecoveryPosition id="RecoveryPosition" description="Ship position when ready to recover robot">
					<LatitudeLongitude latitude="0" longitude="0"/>
				</RecoveryPosition>
				<GoalList>
					<!-- Commence operations -->
					<Goal id="LBT1.0" title="Deploy, Launch" description="Commit to robot support" nextOnSucceed="LBT2.0" nextOnFail="LBT99.0" nextOnException="LBT99.0">
						<Rendezvous description="">
							<TargetVehicleID value="101" description="This robot"/>
							<TargetVehicleID value="102" description="Lifeboat"/>
						</Rendezvous>
						<OperatingArea id="AssignedOpArea">
							<Point>
								<LatitudeLongitude latitude="36.62" longitude="121.506"/>
								<!--36°36´11˝N 121°53´37˝W-->
							</Point>
```

# Lifeboat Tracking Mission in AVCL, goals 1..3



# Lifeboat Tracking Mission in AVCL, goals 4..end



*Comment* Group 4 has common objective and exception, includes loops

**Goal** (5)

| | id | title | description | nextOnSucceed | nextOnExce | nextOnFail | Rendezvous | MonitorTransmissions | Patrol | Comment | OperatingArea |
|---|---|---|---|---|---|---|---|---|---|---|---|
| 1 | LBT4.0 | Track Lifeboat | Monitor and communicate | LBT4.1 | LBT99.0 | | | MonitorTransmissions | | | OperatingArea refid= |
| 2 | LBT4.1 | Maintain proximity | Overhead or afloat nearby | LBT4.2 | LBT99.0 | | Rendezvous c | | | | OperatingArea refid= |
| 3 | LBT4.2 | Periodic reports | Popup or float to report, then recharge | LBT4.3 | LBT99.0 | | | MonitorTransmissions | | | OperatingArea refid= |
| 4 | LBT4.3 | Continue | Repeat until conditions change | LBT5.0 | LBT99.0 | LBT7.0 | | | Patrol description=I | | |
| 5 | LBT5.0 | Check relieved by other asset? | Task update received? | LBT99.0 | LBT99.0 | LBT4.0 | | MonitorTransmissions | | Branch point: note possible human override checked prior to looping. | OperatingArea refid= |

*Comment* Group 6 has common objective, exception

**Goal** (4)

| | id | title | description | nextOnSucceed | nextOnExce | nextOnFail | Reposition | Rendezvous | MarkTarget | MonitorTransmissions | OperatingArea |
|---|---|---|---|---|---|---|---|---|---|---|---|
| 1 | LBT6.0 | Low fuel | Make best effort possible | LBT6.1 | LBT99.0 | | Reposition tit | | | | OperatingArea refid= |
| 2 | LBT6.1 | Remain with lifeboat? | Choices: land on boat, attach to boat, or adrift nearby | LBT6.2 | LBT99.0 | | | Rendezvous title=Attac | | | OperatingArea refid= |
| 3 | LBT6.2 | Beacon? | While power remains | LBT99.0 | LBT99.0 | LBT99.0 | | | MarkTarget descrip | | OperatingArea refid= |
| 4 | LBT7.0 | Request Guidance? | Need updated position | LBT2.0 | LBT99.0 | LBT99.0 | | | | MonitorTransmissions t | OperatingArea refid= |

*Comment* Group 99 has common objective: terminal states

**Goal** (4)

| | id | title | description | nextOnSucceed | nextOnFail | nextOnException | Reposition | Rendezvous | OperatingArea | Duration |
|---|---|---|---|---|---|---|---|---|---|---|
| 1 | LBT99.0 | Proceed to recovery | Mission complete, prepare for pickup | LBT99.1 | LBT99.2 | LBT99.3 | Reposition de | | OperatingArea refic | Duration value=6000 |
| 2 | LBT99.1 | Halt and prepare for recovery | Operations completed, final success state | | | | | Rendezvous description | OperatingArea refic | Duration value=0 |
| 3 | LBT99.2 | Halt and deploy recovery beacon | Unable to operate, final failure state | | | | | Rendezvous description | OperatingArea refic | Duration value=0 |
| 4 | LBT99.3 | Halt and await further orders | Unplanned failure, final exception state | | | | | Rendezvous description | OperatingArea refic | Duration value=0 |

Text | Grid | Schema | WSDL | XBRL | Authentic | Browser

AVCL.3.0.xsd | AVCL.3.0.changes.txt | SailorOverboard.xml | LifeboatTracking.xml

**LifeboatTrackingConverted.ttl** 4.56 KB

```
### Namespace declarations
@prefix :     <https://www.nps.edu/ontologies/MissionExecutionOntology/missions#> .
@prefix meo:  <https://www.nps.edu/ontologies/MissionExecutionOntology#> .
@prefix owl:  <http://www.w3.org/2002/07/owl#> .
@prefix rdf:  <http://www.w3.org/1999/02/22-rdf-syntax-ns#> .
@prefix rdfs: <http://www.w3.org/2000/01/rdf-schema#> .
@prefix xml:  <http://www.w3.org/XML/1998/namespace> .
@prefix xsd:  <http://www.w3.org/2001/XMLSchema#> .

@base <https://www.nps.edu/ontologies/MissionExecutionOntology/missions> . # TODO consider URI matching URL

### Import base ontology
<https://www.nps.edu/ontologies/MissionExecutionOntology/missions> rdf:type owl:Ontology ;
owl:imports <https://www.nps.edu/ontologies/MissionExecutionOntology> .

#################################################################
#    Individuals
#################################################################

:LifeboatTracking rdf:type owl:NamedIndividual , owl:Thing ;
  meo:startsWith :LBT1.0 ;
  rdfs:comment "AVCL mission to 0rovide remote presence for locating, tracking, communications and beaconing." .

:LBT1.0 rdf:type owl:NamedIndividual , owl:Thing ;
  meo:hasNextOnFail :LBT99.0 ;
  meo:hasNextOnSucceed :LBT2.0 ;
  meo:hasNextOnViolate :LBT99.0 ;
  meo:isPartOfPhase "Launch" ;
  rdfs:comment "Deploy, Launch" .

:LBT2.0 rdf:type owl:NamedIndividual , owl:Thing ;
  meo:hasNextOnFail :LBT99.0 ;
  meo:hasNextOnSucceed :LBT3.0 ;
  meo:hasNextOnViolate :LBT99.0 ;
  meo:isPartOfPhase "Transit" ;
  rdfs:comment "Transit to search area" .

:LBT3.0 rdf:type owl:NamedIndividual , owl:Thing ;
  meo:hasNextOnFail :LBT2.0 ;
  meo:hasNextOnSucceed :LBT3.1 ;
  meo:hasNextOnViolate :LBT99.0 ;
  meo:isPartOfPhase "Locate" ;
  rdfs:comment "Locate Lifeboat" .
```

LifeboatTrackingConverted.ttl
Turtle excerpt autogenerated in gitlab.nps.edu version control

# SPARQL query response

## LifeboatTrackingConverted.MissionGoalsQuery_01_GoalBranches.rq.txt

```
-----------------------------------------------------------------------------------------------------------------------------------------------------------------
| goal     | nextOnSucceed | nextOnFail   | nextOnViolate | isPartOfPhase    | description                                                                                |
=================================================================================================================================================================
| :LBT1.0  | :LBT2.0       | :LBT99.0     | :LBT99.0      | "Launch"         | "Deploy, Launch: Commit to robot support"                                                  |
| :LBT2.0  | :LBT3.0       | :LBT99.0     | :LBT99.0      | "Transit"        | "Transit to search area: Proceed to estimated position"                                    |
| :LBT3.0  | :LBT3.1       | :LBT2.0      | :LBT99.0      | "Locate"         | "Locate Lifeboat: Follow best search pattern"                                              |
| :LBT3.1  | :LBT3.2       | :LBT4.0      | :LBT99.0      | "Locate"         | "Report position: Alerts updated"                                                          |
| :LBT3.2  | :LBT4.0       | :LBT4.0      | :LBT99.0      | "Locate"         | "Mark with Beacon: Monitor wind effects and ocean current"                                 |
| :LBT4.0  | :LBT4.1       | "undefined"  | :LBT99.0      | "Track"          | "Track Lifeboat: Monitor and communicate"                                                  |
| :LBT4.1  | :LBT4.2       | "undefined"  | :LBT99.0      | "Track"          | "Maintain proximity: Overhead or afloat nearby"                                            |
| :LBT4.2  | :LBT4.3       | "undefined"  | :LBT99.0      | "Track"          | "Periodic reports: Popup or float to report, also recharge"                                |
| :LBT4.3  | :LBT5.0       | :LBT7.0      | :LBT99.0      | "Track"          | "Continue: Repeat until conditions change"                                                 |
| :LBT5.0  | :LBT99.0      | :LBT4.0      | :LBT99.0      | "Mission Finish" | "Check relieved by other asset: Task update received?"                                     |
| :LBT6.0  | :LBT6.1       | "undefined"  | :LBT99.0      | "Mission Finish" | "Low Fuel: Make best effort possible"                                                      |
| :LBT6.1  | :LBT6.2       | "undefined"  | :LBT99.0      | "Mission Finish" | "Remain with lifeboat? Choices: land on boat, attach to boat, or adrift nearby"            |
| :LBT6.2  | :LBT99.0      | :LBT99.0     | :LBT99.0      | "Mission Finish" | "Beacon? While power remains"                                                              |
| :LBT7.0  | :LBT2.0       | :LBT99.0     | :LBT99.0      | "Transit"        | "Request Guidance? Need updated position"                                                  |
| :LBT99.0 | :LBT99.1      | :LBT99.2     | :LBT99.3      | "Recover Robot"  | "Proceed to recovery: Mission complete, prepare for pickup"                                |
-----------------------------------------------------------------------------------------------------------------------------------------------------------------
```

# Pirates Seizing Merchant Mission: Design

**Motivations**

- Necessity to apply lethal force against pirates distant from own ship, corresponding life-saving force potential for hostage merchant crew.
- Must operate over long time period, emphasize restraint throughout.
- Provide soft and strict supervisory checkpoints for human control.

**Lessons learned**

- Concept of phases helped organize overall mission structure sensibly (approach, warning, attack, recovery).
- Looping is necessary, human control checkpoints help avoid deadlock.

# Pirates Seizing Merchant Mission: Description

**Purpose**

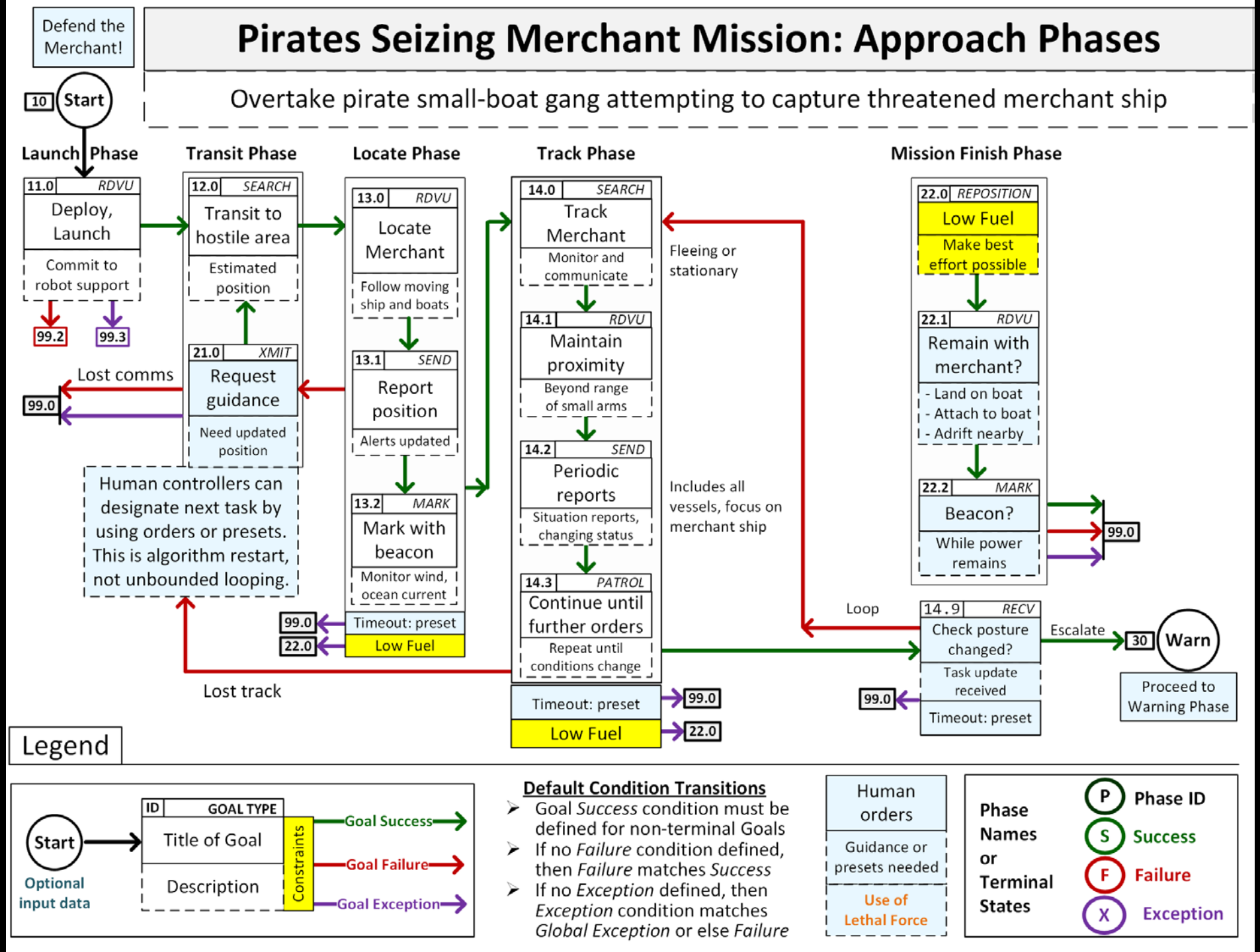
- Overtake pirate small-boat gang attempting to capture threatened evading merchant ship.

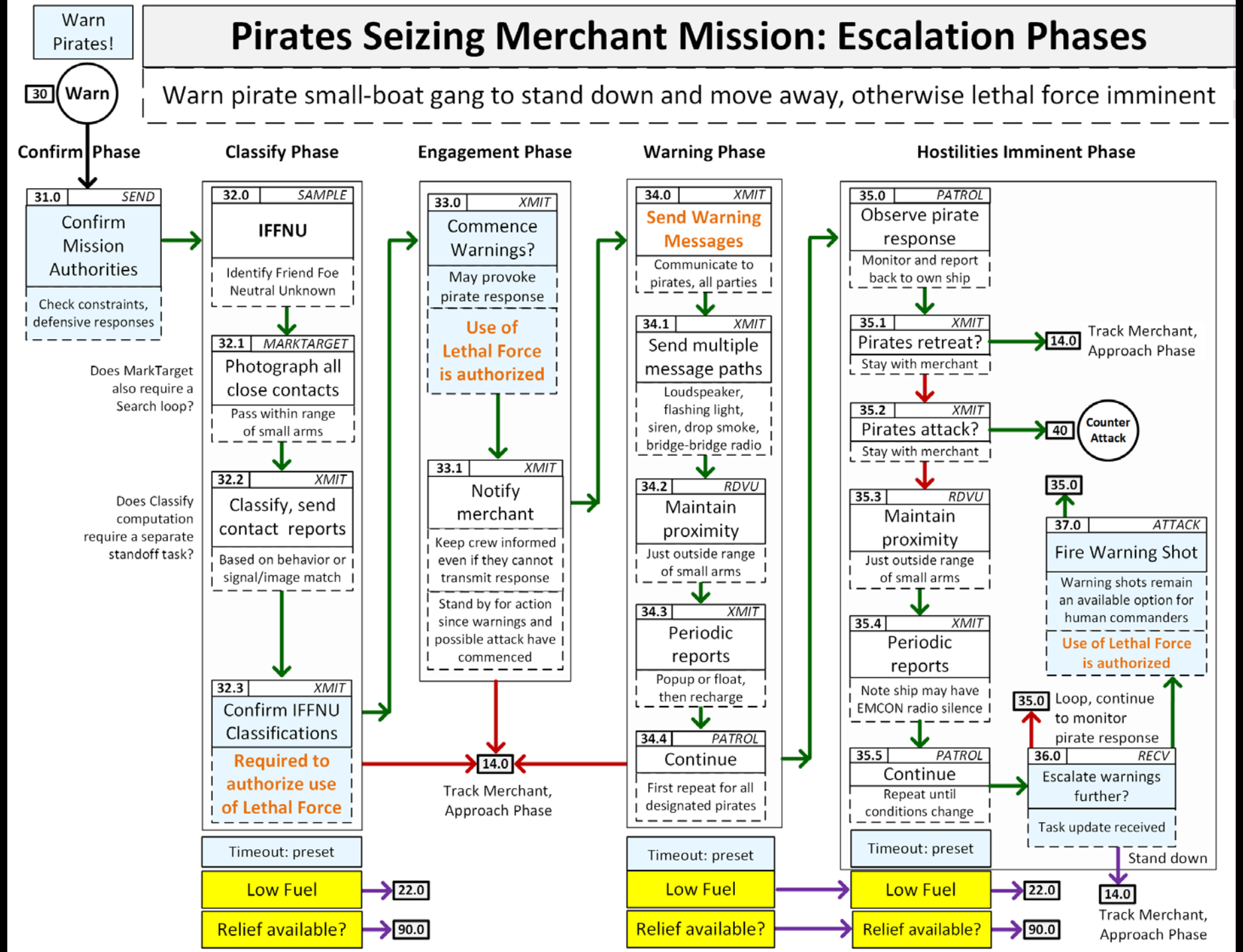
**Phases**

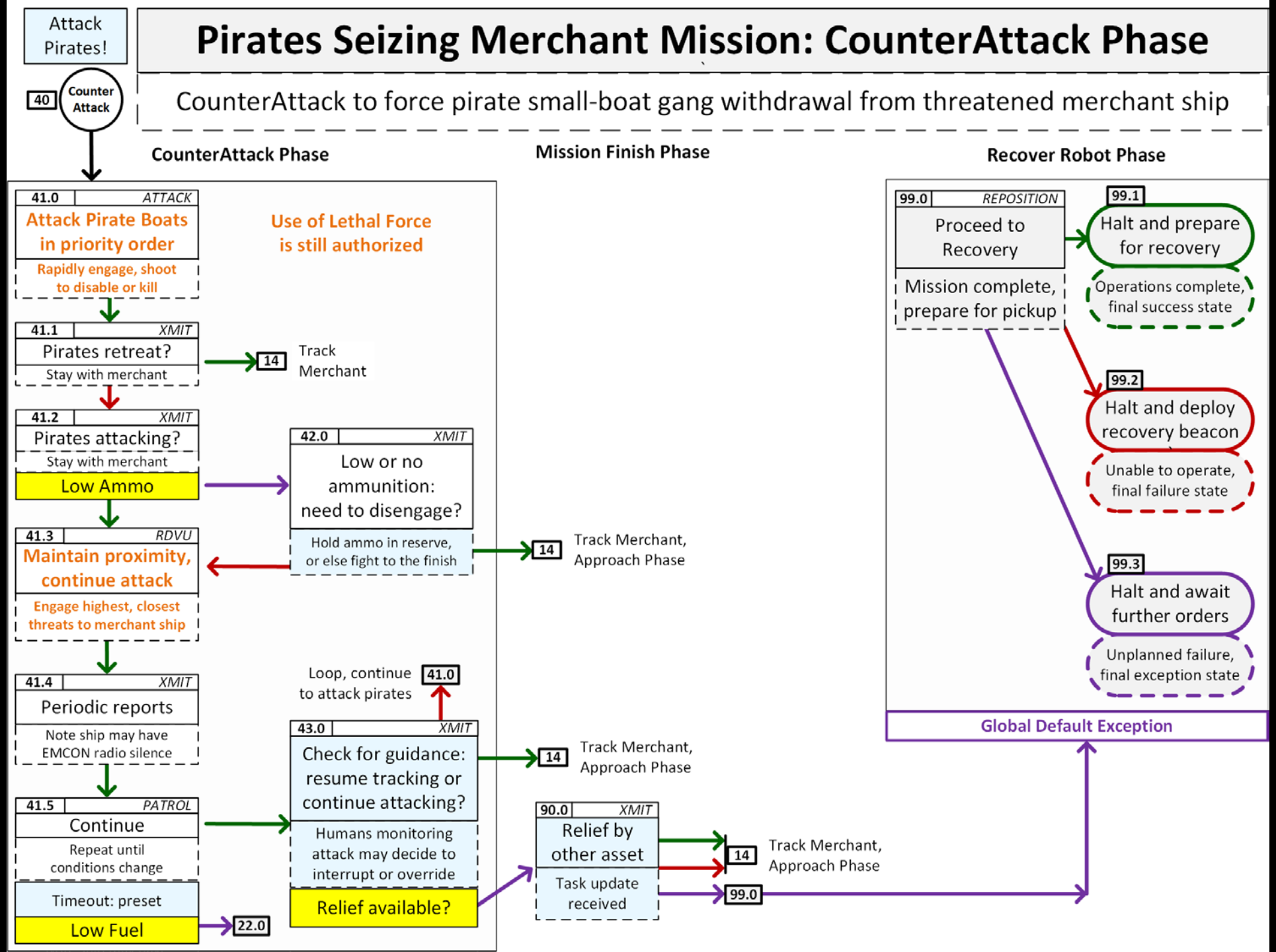
- Deploy/Launch, Search, Approach and Track, Warning, Attack.

**Human Supervisory Role and Constraints**

- Control pace of engagement, careful deliberate escalation.
- Confirm IFFNU classification, must order lethal force prior to use.
- Low fuel condition and graceful-degradation response.
- Low ammunition condition: fight to finish, or stand in reserve?


Defend the Merchant!
Pirates Seizing Merchant Mission: Approach Phases
Overtake pirate small-boat gang attempting to capture threatened merchant ship
10 Start
Launch Phase
Transit Phase
Locate Phase
Track Phase
Mission Finish Phase
11.0 RDVU
Deploy, Launch
Commit to robot support
99.2
99.3
12.0 SEARCH
Transit to hostile area
Estimated position
21.0 XMIT
Request guidance
Need updated position
Lost comms
99.0
Human controllers can designate next task by using orders or presets. This is algorithm restart, not unbounded looping.
13.0 RDVU
Locate Merchant
Follow moving ship and boats
13.1 SEND
Report position
Alerts updated
13.2 MARK
Mark with beacon
Monitor wind, ocean current
Timeout: preset
Low Fuel
99.0
22.0
Lost track
14.0 SEARCH
Track Merchant
Monitor and communicate
14.1 RDVU
Maintain proximity
Beyond range of small arms
14.2 SEND
Periodic reports
Situation reports, changing status
14.3 PATROL
Continue until further orders
Repeat until conditions change
Timeout: preset
Low Fuel
99.0
22.0
Fleeing or stationary
Includes all vessels, focus on merchant ship
22.0 REPOSITION
Low Fuel
Make best effort possible
22.1 RDVU
Remain with merchant?
- Land on boat
- Attach to boat
- Adrift nearby
22.2 MARK
Beacon?
While power remains
99.0
Loop
14.9 RECV
Check posture changed?
Task update received
Timeout: preset
99.0
Escalate
30 Warn
Proceed to Warning Phase
Legend
Start
Optional input data
ID GOAL TYPE
Title of Goal
Description
Constraints
Goal Success
Goal Failure
Goal Exception
Default Condition Transitions
Goal Success condition must be defined for non-terminal Goals
If no Failure condition defined, then Failure matches Success
If no Exception defined, then Exception condition matches Global Exception or else Failure
Human orders
Guidance or presets needed
Use of Lethal Force
Phase Names or Terminal States
P Phase ID
S Success
F Failure
X Exception


Warn Pirates!
Pirates Seizing Merchant Mission: Escalation Phases
30 Warn
Warn pirate small-boat gang to stand down and move away, otherwise lethal force imminent
Confirm Phase
Classify Phase
Engagement Phase
Warning Phase
Hostilities Imminent Phase
31.0 SEND
Confirm Mission Authorities
Check constraints, defensive responses
32.0 SAMPLE
IFFNU
Identify Friend Foe Neutral Unknown
32.1 MARKTARGET
Photograph all close contacts
Pass within range of small arms
Does MarkTarget also require a Search loop?
32.2 XMIT
Classify, send contact reports
Based on behavior or signal/image match
Does Classify computation require a separate standoff task?
32.3 XMIT
Confirm IFFNU Classifications
Required to authorize use of Lethal Force
33.0 XMIT
Commence Warnings?
May provoke pirate response
Use of Lethal Force is authorized
33.1 XMIT
Notify merchant
Keep crew informed even if they cannot transmit response
Stand by for action since warnings and possible attack have commenced
14.0
Track Merchant, Approach Phase
34.0 XMIT
Send Warning Messages
Communicate to pirates, all parties
34.1 XMIT
Send multiple message paths
Loudspeaker, flashing light, siren, drop smoke, bridge-bridge radio
34.2 RDVU
Maintain proximity
Just outside range of small arms
34.3 XMIT
Periodic reports
Popup or float, then recharge
34.4 PATROL
Continue
First repeat for all designated pirates
35.0 PATROL
Observe pirate response
Monitor and report back to own ship
35.1 XMIT
Pirates retreat?
Stay with merchant
14.0
Track Merchant, Approach Phase
35.2 XMIT
Pirates attack?
Stay with merchant
40
Counter Attack
35.3 RDVU
Maintain proximity
Just outside range of small arms
35.4 XMIT
Periodic reports
Note ship may have EMCON radio silence
35.5 PATROL
Continue
Repeat until conditions change
Timeout: preset
35.0
37.0 ATTACK
Fire Warning Shot
Warning shots remain an available option for human commanders
Use of Lethal Force is authorized
35.0
Loop, continue to monitor pirate response
36.0 RECV
Escalate warnings further?
Task update received
Stand down
14.0
Track Merchant, Approach Phase
Timeout: preset
Low Fuel
22.0
Relief available?
90.0
Timeout: preset
Low Fuel
22.0
Relief available?
90.0


Attack Pirates!
Pirates Seizing Merchant Mission: CounterAttack Phase
40
Counter Attack
CounterAttack to force pirate small-boat gang withdrawal from threatened merchant ship
CounterAttack Phase
Mission Finish Phase
Recover Robot Phase
41.0 ATTACK
Attack Pirate Boats in priority order
Rapidly engage, shoot to disable or kill
Use of Lethal Force is still authorized
41.1 XMIT
Pirates retreat?
Stay with merchant
14
Track Merchant
41.2 XMIT
Pirates attacking?
Stay with merchant
Low Ammo
42.0 XMIT
Low or no ammunition: need to disengage?
Hold ammo in reserve, or else fight to the finish
14
Track Merchant, Approach Phase
41.3 RDVU
Maintain proximity, continue attack
Engage highest, closest threats to merchant ship
41.4 XMIT
Periodic reports
Note ship may have EMCON radio silence
Loop, continue to attack pirates
41.0
43.0 XMIT
Check for guidance: resume tracking or continue attacking?
14
Track Merchant, Approach Phase
41.5 PATROL
Continue
Repeat until conditions change
Humans monitoring attack may decide to interrupt or override
Timeout: preset
Low Fuel
22.0
Relief available?
90.0 XMIT
Relief by other asset
Task update received
14
Track Merchant, Approach Phase
99.0
Global Default Exception
99.0 REPOSITION
Proceed to Recovery
Mission complete, prepare for pickup
99.1
Halt and prepare for recovery
Operations complete, final success state
99.2
Halt and deploy recovery beacon
Unable to operate, final failure state
99.3
Halt and await further orders
Unplanned failure, final exception state

# PiratesSeizingMerchantDefense.xml in gitlab.nps.edu version control

https://gitlab.nps.edu/Savage/EthicalControl/blob/master/missions/avcl/PiratesSeizingMerchantDefense

GitLab Projects Groups More Search or jump to...

PiratesSeizingMerchantDefense.xml 20.3 KB

Edit Web IDE Replace Delete

```xml
<?xml version="1.0" encoding="UTF-8"?>
<AVCL version="3.0" vehicleName="RescueDrone" vehicleType="UAV" xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance" xsi:noNamespaceSchemaLocation="../../avcl/AVCL.3.0.xsd">
	<!-- vehicleType="USV" is also possible -->
	<head>
		<meta name="title" content="PiratesSeizingMerchantDefense.xml"/>
		<meta name="created" content="27 December 2019"/>
		<meta name="modified" content="31 December 2019"/>
		<meta name="creator" content="Don Brutzman"/>
		<meta name="reference" content="https://wiki.nps.edu/display/NOW/Ethical+Control+of+Unmanned+Systems"/>
		<meta name="reference" content="https://gitlab.nps.edu/Savage/EthicalControl/raw/master/missions/PiratesSeizingMerchantDefense.png"/>
		<meta name="identifier" content="https://gitlab.nps.edu/Savage/EthicalControl/tree/master/missions/avcl/PiratesSeizingMerchantDefense.xml"/>
		<meta name="generator" content="Altova XMLSpy, https://www.altova.com"/>
		<meta name="generator" content="Apache NetBeans, https://netbeans.apache.org"/>
		<meta name="generator" content="NPS Autonomous Unmanned Vehicle (AUV) Workbench, https://savage.nps.edu/AuvWorkbench"/>
		<meta name="license" content="../license.html"/>
	</head>
	<body>
		<MissionPreparation>
			<UnitsOfMeasure distance="meters" angle="degrees" mass="kilograms" time="seconds"/>
			<AgendaMission>
				<LaunchPosition id="LaunchPosition" description="Ship position when robot is launched">
					<LatitudeLongitude latitude="0" longitude="0"/>
				</LaunchPosition>
				<RecoveryPosition id="RecoveryPosition" description="Ship position when ready to recover robot">
					<LatitudeLongitude latitude="0" longitude="0"/>
				</RecoveryPosition>
				<GoalList>
					<!--*Initiation Phase* to commence operations-->
					<Goal id="PSMD11.0" title="Deploy, Launch" description="Commit to robot support" nextOnSucceed="PSMD12.0" nextOnFail="PSMD99.0" nextOnException="PSMD
						<Rendezvous description="">
							<TargetVehicleID value="100" description="Launching ship"/>
							<TargetVehicleID value="101" description="This robot"/>
							<!--
							<TargetVehicleID value="102" description="Recovery Ship"/>
							<TargetVehicleID value="103" description="Merchant Ship"/>
							-->
						</Rendezvous>
```

# PiratesSeizingMerchantDefense.xml AVCL 1

- **AgendaMission**
  - **LaunchPosition** id=LaunchPosition description=Ship position when robot is launched
  - **RecoveryPosition** id=RecoveryPosition description=Ship position when ready to recover robot
  - **GoalList**
    - *Comment* *Initiation Phase* to commence operations
    - **Goal**

| | |
|---|---|
| **id** | PSMD11.0 |
| **title** | Deploy, Launch |
| **description** | Commit to robot support |
| **nextOnSucceed** | PSMD12.0 |
| **nextOnFail** | PSMD99.0 |
| **nextOnException** | PSMD99.0 |

      - **Rendezvous** description=
      - **OperatingArea** id=AssignedOpArea
      - **Duration** value=300
      - **ReportingCriteria** (4)
    - *Comment* *Approach Phase* to overtake pirate small-boat gang attempting to capture threatened merchant ship
    - **Goal**

| | |
|---|---|
| **id** | PSMD12.0 |
| **title** | Transit to hostile area |
| **description** | Proceed to estimated position |
| **nextOnSucceed** | PSMD13.0 |
| **nextOnFail** | PSMD99.0 |
| **nextOnException** | PSMD99.0 |

      - **Search** datumType=point requiredPD=0.5
      - **OperatingArea** refid=AssignedOpArea
      - **Duration** value=6000
    - *Comment* Group 13 Locate Merchant has common objective, exception
    - **Goal** (3)

| | id | title | description | nextOnSucceed | nextOnFail | nextOnException | Rendezvous | MonitorTransmissi | MarkTarget | OperatingArea |
|---|---|---|---|---|---|---|---|---|---|---|
| 1 | PSMD13.0 | Locate Merchant | Use best search pattern, then follow moving ship and boats | PSMD13.1 | PSMD21.0 | PSMD99.0 | Rendezvous descrip | | | OperatingArea refic |
| 2 | PSMD13.1 | Report position | Alerts updated | PSMD13.2 | PSMD13.2 | PSMD99.0 | | MonitorTransmissi | | OperatingArea refic |
| 3 | PSMD13.2 | Mark with Beacon | Monitor wind effects and ocean current | PSMD14.0 | PSMD21.0 | PSMD99.0 | | | MarkTarget descrip | OperatingArea refic |

# PiratesSeizingMerchantDefense.xml AVCL 2 Track Merchant

Comment: Group 14 Track Merchant has common objective and exception, includes loops

**Goal** (5)

| | id | title | description | nextOnSucceed | nextOnFail | nextOnException | Search | Rendezvous | MonitorTransmissi | Comment |
|---|---|---|---|---|---|---|---|---|---|---|
| 1 | PSMD14.0 | Track Merchant | Monitor and communicate | PSMD14.1 | PSMD14.1 | PSMD99.0 | **Search** datumType= | | | |
| 2 | PSMD14.1 | Maintain proximity | Beyond range of small arms | PSMD14.2 | | PSMD99.0 | | **Rendezvous** descrip | | |
| 3 | PSMD14.2 | Periodic reports | Situation reports, changing status | PSMD14.3 | | PSMD99.0 | | | **MonitorTransmissi** | |
| 4 | PSMD14.3 | Continue until further orders | Repeat until conditions change | PSMD14.9 | PSMD21.0 | PSMD99.0 | | | | |
| 5 | PSMD14.9 | Check posture changed? | Task update received? | PSMD30.0 | PSMD14.0 | PSMD99.0 | | | **MonitorTransmissi** | Branch point: note possible human override checked prior to looping. |

Comment: Group 21 Request Guidance? and Group 22 Low Fuel have common objective, exception

**Goal** (4)

| | id | title | description | nextOnSucceed | nextOnFail | nextOnException | MonitorTransmissi | Reposition | Rendezvous | MarkTarget |
|---|---|---|---|---|---|---|---|---|---|---|
| 1 | PSMD21.0 | Request Guidance? | Need updated position | PSMD12.0 | PSMD99.0 | PSMD99.0 | **MonitorTransmissi** | | | |
| 2 | PSMD22.0 | Low Fuel | Make best effort possible | PSMD22.1 | | PSMD99.0 | | **Reposition** title=Pre | | |
| 3 | PSMD22.1 | Remain with Merchant? | Choices: land on boat, attach to boat, or adrift nearby | PSMD22.2 | | PSMD99.0 | | | **Rendezvous** title=A | |
| 4 | PSMD22.2 | Beacon? | While power remains | PSMD99.0 | PSMD99.0 | PSMD99.0 | | | | **MarkTarget** descrip |

Comment: *Warning Phase* to warn pirate small-boat gang to stand down and move away, otherwise lethal force imminent

**Goal** (2)

| | id | title | description | nextOnSucceed | nextOnException | MonitorTransmissi |
|---|---|---|---|---|---|---|
| 1 | PSMD30.0 | Warning Phase | Close interaction with pirates, merchant ship | PSMD31.0 | PSMD99.0 | |
| 2 | PSMD31.0 | Confirm Mission Authorities | Check constraints and defensive responses | PSMD32.0 | PSMD99.0 | **MonitorTransmissi** |

# PiratesSeizingMerchantDefense.xml AVCL 3 Warning Phase initiation

Comment: Group 32 IFFNU has common objective, exception

**Goal** (4)

| | id | title | description | nextOnSucceed | nextOnFail | nextOnException | SampleEnvironmer | MarkTarget | Comment | MonitorTransmissi |
|---|---|---|---|---|---|---|---|---|---|---|
| 1 | PSMD32.0 | IFFNU | Identify Friend Foe Neutral Unknown | PSMD32.1 | PSMD32.1 | PSMD99.0 | SampleEnvironmer | | | |
| 2 | PSMD32.1 | Photograph all close contacts | Pass within range of small arms | PSMD32.2 | | PSMD99.0 | | MarkTarget descrip | | |
| 3 | PSMD32.2 | Classify, send contact reports | Based on target behavior or signal/ image match | PSMD32.3 | | PSMD99.0 | | | Does Classify computation require a separate standoff task? | MonitorTransmissi |
| 4 | PSMD32.3 | Confirm IFFNU classifications | Requires approval by human commander before proceeding further | PSMD33.0 | PSMD14.0 | PSMD99.0 | | | | MonitorTransmissi |

Comment: Group 33 Commence Warnings? decision tree has common objective, exception

**Goal** (2)

| | id | title | description | nextOnSucceed | nextOnFail | nextOnException | MonitorTransmissi |
|---|---|---|---|---|---|---|---|
| 1 | PSMD33.0 | Commence Warnings | May provoke pirate response | PSMD33.1 | PSMD14.0 | PSMD99.0 | MonitorTransmissi |
| 2 | PSMD33.1 | Notify Merchant | Keep crew informed, even if they cannot transmit | PSMD34.0 | PSMD14.0 | PSMD99.0 | MonitorTransmissi |

Comment: Group 34 Send Warning Messages has common objective, exception

**Goal** (5)

| | id | title | description | nextOnSucceed | nextOnFail | nextOnException | MonitorTransmissi | Rendezvous | OperatingArea | Duration | Patrol |
|---|---|---|---|---|---|---|---|---|---|---|---|
| 1 | PSMD34.0 | Send Warning Messages | Communicate to pirates, all parties | PSMD34.1 | PSMD34.1 | PSMD99.0 | MonitorTransmissi | | | | |
| 2 | PSMD34.1 | Send multiple message paths | Loudspeaker, flashing light, siren, drop smoke, bridge-bridge radio | PSMD34.2 | | PSMD99.0 | MonitorTransmissi | | | | |
| 3 | PSMD34.2 | Maintain proximity | Just outside range of small arms | PSMD34.3 | | PSMD99.0 | | Rendezvous descrip | OperatingArea refi | | |
| 4 | PSMD34.3 | Periodic reports | Situation reports, changing status | PSMD34.4 | | PSMD99.0 | MonitorTransmissi | | | Duration value=60 | |
| 5 | PSMD34.4 | Continue until further orders | First repeat for all designated pirates | PSMD35.0 | PSMD14.0 | PSMD99.0 | | | | | Patrol description=F |

# PiratesSeizingMerchantDefense.xml AVCL 4 Observe Pirate Response to warnings

*Comment* Group 35 Observe pirate response has common objective, exception

**Goal** (8)

| | = id | = title | = description | = nextOnSucceed | = nextOnFail | = nextOnException | Patrol | Rendezvous | OperatingArea | MonitorTransmissi | Duration | Attack |
|---|---|---|---|---|---|---|---|---|---|---|---|---|
| 1 | PSMD35.0 | Observe pirate response | Monitor and report back to own ship | PSMD35.1 | PSMD35.1 | PSMD99.0 | **Patrol** description=I | | **OperatingArea** refic | | | |
| 2 | PSMD35.1 | Pirates retreat? | Stay with merchant, report if response changes | PSMD14.0 | PSMD35.2 | PSMD99.0 | | | | **MonitorTransmissi** | | |
| 3 | PSMD35.2 | Pirates attack? | Stay with merchant, counterattack if response changes | PSMD40.0 | PSMD35.2 | PSMD99.0 | | | | **MonitorTransmissi** | | |
| 4 | PSMD35.3 | Maintain proximity | Just outside range of small arms | PSMD35.4 | | PSMD99.0 | | **Rendezvous** descrip | **OperatingArea** refic | | | |
| 5 | PSMD35.4 | Periodic reports | Situation reports, changing status - note ship may have EMCON radio silence | PSMD35.5 | | PSMD99.0 | | | | **MonitorTransmissi** | **Duration** value=600 | |
| 6 | PSMD35.5 | Continue until further orders | Repeat until conditions change | PSMD36.0 | | PSMD99.0 | **Patrol** description=I | | **OperatingArea** refic | | | |
| 7 | PSMD36.0 | Check posture changed? | Task update received | PSMD14.0 | PSMD35.0 | PSMD99.0 | | | | **MonitorTransmissi** | **Duration** value=600 | |
| 8 | PSMD37.0 | Fire Warning Shot | Deliberately avoided in robot mission logic, but shots remain an available option for human commanders. Lethal force is authorized. | PSMD31.0 | PSMD31.0 | PSMD99.0 | | | | | | **Attack** title=Atta |

# PiratesSeizingMerchantDefense.xml AVCL 5 Interdiction Phase

Comment *Interdiction Phase* Counterattack to force pirate small-boat gang withdrawal from threatened merchant ship

Goal (9)

| | id | title | description | nextOnSucceed | nextOnFail | nextOnException | Attack | MonitorTransmissi | Rendezvous | Patrol | OperatingArea | Duration |
|---|---|---|---|---|---|---|---|---|---|---|---|---|
| 1 | PSMD40.0 | Attack | Close interaction with pirates, merchant ship | PSMD41.0 | PSMD41.0 | PSMD99.0 | | | | | | |
| 2 | PSMD41.0 | Attack Pirate Boats in priority order | Rapidly engage, shoot to disable or kill. Lethal force still authorized. | PSMD41.1 | | PSMD99.0 | Attack title=Attack | | | | | |
| 3 | PSMD41.1 | Pirates retreat? | Stay with merchant, report if response changes | PSMD14.0 | PSMD41.2 | PSMD99.0 | | MonitorTransmissi | | | | |
| 4 | PSMD41.2 | Pirates attacking? | Stay with merchant, counterattack if response changes | PSMD41.3 | | PSMD42.0 | | MonitorTransmissi | | | | |
| 5 | PSMD41.3 | Maintain proximity | Just outside range of small arms | PSMD41.4 | | PSMD99.0 | | | Rendezvous descrip | | OperatingArea refic | |
| 6 | PSMD41.4 | Periodic reports | Situation reports, changing status - note ship may have EMCON radio silence | PSMD41.5 | | PSMD99.0 | | MonitorTransmissi | | | | Duration value= |
| 7 | PSMD41.5 | Continue until further orders | Repeat until conditions change | PSMD36.0 | | PSMD99.0 | | | | Patrol description=I | OperatingArea refic | |
| 8 | PSMD42.0 | Low or no ammunition: need to disengage? | Hold ammunition in reseve, or else fight to the finish | PSMD14.0 | PSMD41.3 | PSMD90.0 | | MonitorTransmissi | | | | Duration value= |
| 9 | PSMD43.0 | Check for guidance: resume tracking or continue attacking? | Humans monitoring attack may decide to interrupt or override | PSMD14.0 | PSMD41.0 | PSMD90.0 | | MonitorTransmissi | | | | Duration value= |

# PiratesSeizingMerchantDefense.xml AVCL 6 Recovery Phase

| | |
|---|---|
| *Comment* | *Recovery Phase* |

**Goal**

| | |
|---|---|
| **id** | PSMD90.0 |
| **title** | Relief by other asset |
| **description** | Continue tracking merchant unless further task update recieved |
| **nextOnSucceed** | PSMD14.0 |
| **nextOnFail** | PSMD14.0 |
| **nextOnException** | PSMD90.0 |
| **MonitorTransmissions** xsi:type=monitorTransmissionsElementType description=Track sailor in water | |
| **Duration** value=6000 | |

| | |
|---|---|
| *Comment* | Group 99 has common objective: Global Default Exception and terminal states |

**Goal** (4)

| | id | title | description | nextOnSucceed | nextOnFail | nextOnException | Reposition | Rendezvous |
|---|---|---|---|---|---|---|---|---|
| 1 | PSMD99.0 | Proceed to recovery | Mission complete, prepare for pickup | PSMD99.1 | PSMD99.2 | PSMD99.3 | **Reposition** descripti | |
| 2 | PSMD99.1 | Halt and prepare for recovery | Operations completed, final success state | | | | | **Rendezvous** descrip |
| 3 | PSMD99.2 | Halt and deploy recovery beacon | Unable to operate, final failure state | | | | | **Rendezvous** descrip |
| 4 | PSMD99.3 | Halt and await further orders | Unplanned failure, final exception state | | | | | **Rendezvous** descrip |

PiratesSeizingMerchantDefenseConverted.ttl 11.4 KB

```
### Namespace declarations
@prefix :     <https://www.nps.edu/ontologies/MissionExecutionOntology/missions#> .
@prefix meo:  <https://www.nps.edu/ontologies/MissionExecutionOntology#> .
@prefix owl:  <http://www.w3.org/2002/07/owl#> .
@prefix rdf:  <http://www.w3.org/1999/02/22-rdf-syntax-ns#> .
@prefix rdfs: <http://www.w3.org/2000/01/rdf-schema#> .
@prefix xml:  <http://www.w3.org/XML/1998/namespace> .
@prefix xsd:  <http://www.w3.org/2001/XMLSchema#> .

@base <https://www.nps.edu/ontologies/MissionExecutionOntology/missions> . # TODO consider URI matching URL

### Import base ontology
<https://www.nps.edu/ontologies/MissionExecutionOntology/missions> rdf:type owl:Ontology ;
owl:imports <https://www.nps.edu/ontologies/MissionExecutionOntology> .

#################################################################
#    Individuals
#################################################################

:PiratesSeizingMerchantDefense rdf:type owl:NamedIndividual , owl:Thing ;
  meo:startsWith :PSMD11.0 ;
  rdfs:comment "AVCL mission to overtake pirate small-boat gang attempting to capture threatened merchant ship" .

:PSMD11.0 rdf:type owl:NamedIndividual , owl:Thing ;
  meo:hasNextOnFail :PSMD99.0 ;
  meo:hasNextOnSucceed :PSMD12.0 ;
  meo:hasNextOnViolate :PSMD99.0 ;
  meo:isPartOfPhase "Launch" ;
  rdfs:comment "Deploy, Launch" .

:PSMD12.0 rdf:type owl:NamedIndividual , owl:Thing ;
  meo:hasNextOnFail :PSMD99.0 ;
  meo:hasNextOnSucceed :PSMD13.0 ;
  meo:hasNextOnViolate :PSMD99.0 ;
  meo:isPartOfPhase "Transit" ;
  rdfs:comment "Transit to hostile area" .

:PSMD13.0 rdf:type owl:NamedIndividual , owl:Thing ;
  meo:hasNextOnFail :PSMD21.0 ;
  meo:hasNextOnSucceed :PSMD13.1 ;
  meo:hasNextOnViolate :PSMD99.0 ;
  meo:isPartOfPhase "Transit" ;
  rdfs:comment "Locate Merchant" .
```

PiratesSeizingMerchantDefenseConverted.ttl

Turtle excerpt autogenerated in gitlab.nps.edu version control

# SPARQL query response

## PiratesSeizingMerchantDefenseConverted.MissionGoalsQuery_01_GoalBranches.rq.txt

| goal | nextOnSucceed | nextOnFail | nextOnViolate | isPartOfPhase | description |
|---|---|---|---|---|---|
| :PSMD11.0 | :PSMD12.0 | :PSMD99.0 | :PSMD99.0 | "Launch" | "Deploy, Launch: Commit to robot support" |
| :PSMD12.0 | :PSMD13.0 | :PSMD99.0 | :PSMD99.0 | "Transit" | "Transit to hostile area: Proceed to estimated position" |
| :PSMD13.0 | :PSMD13.1 | :PSMD21.0 | :PSMD99.0 | "Transit" | "Locate Merchant: Use best search pattern, then follow moving ship and boats" |
| :PSMD13.1 | :PSMD13.2 | :PSMD13.2 | :PSMD99.0 | "Transit" | "Report position: Alerts updated" |
| :PSMD13.2 | :PSMD14.0 | :PSMD21.0 | :PSMD99.0 | "Transit" | "Mark with Beacon: Monitor wind effects and ocean current" |
| :PSMD14.0 | :PSMD14.1 | :PSMD14.1 | :PSMD99.0 | "Track" | "Track Merchant: Monitor and communicate" |
| :PSMD14.1 | :PSMD14.2 | "undefined" | :PSMD99.0 | "Track" | "Maintain proximity: Beyond range of small arms" |
| :PSMD14.2 | :PSMD14.3 | "undefined" | :PSMD99.0 | "Track" | "Periodic reports: Situation reports, changing status" |
| :PSMD14.3 | :PSMD14.9 | :PSMD21.0 | :PSMD99.0 | "Track" | "Continue until further orders: Repeat until conditions change" |
| :PSMD14.9 | :PSMD30.0 | :PSMD14.0 | :PSMD99.0 | "Track" | "Check posture changed? Task update received?" |
| :PSMD21.0 | :PSMD12.0 | :PSMD99.0 | :PSMD99.0 | "Transit" | "Request Guidance? Need updated position" |
| :PSMD22.0 | :PSMD22.1 | "undefined" | :PSMD99.0 | "Mission Finish" | "Low Fuel: Make best effort possible" |
| :PSMD22.1 | :PSMD22.2 | "undefined" | :PSMD99.0 | "Mission Finish" | "Remain with Merchant? Choices: land on boat, attach to boat, or adrift nearby" |
| :PSMD22.2 | :PSMD99.0 | :PSMD99.0 | :PSMD99.0 | "Mission Finish" | "Beacon? While power remains" |
| :PSMD30.0 | :PSMD31.0 | "undefined" | :PSMD99.0 | "Warning" | "Warning Phase: Close interaction with pirates, merchant ship" |
| :PSMD31.0 | :PSMD32.0 | "undefined" | :PSMD99.0 | "Warning" | "Confirm Mission Authorities: Check constraints and defensive responses" |
| :PSMD32.0 | :PSMD32.1 | :PSMD32.1 | :PSMD99.0 | "Classify" | "IFFNU: Identify Friend Foe Neutral Unknown" |
| :PSMD32.1 | :PSMD32.2 | "undefined" | :PSMD99.0 | "Classify" | "Photograph all close contacts: Pass within range of small arms" |
| :PSMD32.2 | :PSMD32.3 | "undefined" | :PSMD99.0 | "Classify" | "Classify, send contact reports: Based on target behavior or signal/image match" |
| :PSMD32.3 | :PSMD33.0 | :PSMD14.0 | :PSMD99.0 | "Classify" | "Confirm IFFNU classifications: Requires approval by human commander before proceeding further" |
| :PSMD33.0 | :PSMD33.1 | :PSMD14.0 | :PSMD99.0 | "Engagement" | "Commence Warnings: May provoke pirate response" |
| :PSMD33.1 | :PSMD34.0 | :PSMD14.0 | :PSMD99.0 | "Engagement" | "Notify Merchant: Keep crew informed, even if they cannot transmit" |
| :PSMD34.0 | :PSMD34.1 | :PSMD34.1 | :PSMD99.0 | "Warning" | "Send Warning Messages: Communicate to pirates, all parties" |
| :PSMD34.1 | :PSMD34.2 | "undefined" | :PSMD99.0 | "Warning" | "Send multiple message paths: Loudspeaker, flashing light, siren, drop smoke, bridge-bridge radio" |
| :PSMD34.2 | :PSMD34.3 | "undefined" | :PSMD99.0 | "Warning" | "Maintain proximity: Just outside range of small arms" |
| :PSMD34.3 | :PSMD34.4 | "undefined" | :PSMD99.0 | "Warning" | "Periodic reports: Situation reports, changing status" |
| :PSMD34.4 | :PSMD35.0 | :PSMD14.0 | :PSMD99.0 | "Warning" | "Continue until further orders: First repeat for all designated pirates" |
| :PSMD35.0 | :PSMD35.1 | :PSMD35.1 | :PSMD99.0 | "Hostilities Imminent" | "Observe pirate response: Monitor and report back to own ship" |
| :PSMD35.1 | :PSMD14.0 | :PSMD35.2 | :PSMD99.0 | "Hostilities Imminent" | "Pirates retreat? Stay with merchant, report if response changes" |
| :PSMD35.2 | :PSMD40.0 | :PSMD35.2 | :PSMD99.0 | "Hostilities Imminent" | "Pirates attack? Stay with merchant, counterattack if response changes" |
| :PSMD35.3 | :PSMD35.4 | "undefined" | :PSMD99.0 | "Hostilities Imminent" | "Maintain proximity: Just outside range of small arms" |
| :PSMD35.4 | :PSMD35.5 | "undefined" | :PSMD99.0 | "Hostilities Imminent" | "Periodic reports: Situation reports, changing status - note ship may have EMCON radio silence" |
| :PSMD35.5 | :PSMD36.0 | "undefined" | :PSMD99.0 | "Hostilities Imminent" | "Continue until further orders: Repeat until conditions change" |
| :PSMD36.0 | :PSMD37.0 | :PSMD35.0 | :PSMD14.0 | "Hostilities Imminent" | "Check posture changed? Task update received" |
| :PSMD37.0 | :PSMD31.0 | :PSMD31.0 | :PSMD99.0 | "Hostilities Imminent" | "Fire Warning Shot: Warning shots remain an available option for human commanders. Lethal force is authorized." |
| :PSMD40.0 | :PSMD41.0 | :PSMD41.0 | :PSMD99.0 | "Counter Attack" | "Attack: Close interaction with pirates, merchant ship" |
| :PSMD41.0 | :PSMD41.1 | "undefined" | :PSMD99.0 | "Counter Attack" | "Attack Pirate Boats in priority order: Rapidly engage, shoot to disable or kill. Lethal force still authorized." |
| :PSMD41.1 | :PSMD14.0 | :PSMD41.2 | :PSMD99.0 | "Counter Attack" | "Pirates retreat? Stay with merchant, report if response changes" |
| :PSMD41.2 | :PSMD41.3 | "undefined" | :PSMD42.0 | "Counter Attack" | "Pirates attacking? Stay with merchant, counterattack if response changes" |
| :PSMD41.3 | :PSMD41.4 | "undefined" | :PSMD99.0 | "Counter Attack" | "Maintain proximity: Just outside range of small arms" |
| :PSMD41.4 | :PSMD41.5 | "undefined" | :PSMD99.0 | "Counter Attack" | "Periodic reports: Situation reports, changing status - note ship may have EMCON radio silence" |
| :PSMD41.5 | :PSMD36.0 | "undefined" | :PSMD99.0 | "Counter Attack" | "Continue until further orders: Repeat until conditions change" |
| :PSMD42.0 | :PSMD14.0 | :PSMD41.3 | :PSMD90.0 | "Counter Attack" | "Low or no ammunition: need to disengage? Hold ammunition in reseve, or else fight to the finish" |
| :PSMD43.0 | :PSMD14.0 | :PSMD41.0 | :PSMD90.0 | "Counter Attack" | "Check for guidance, resume tracking or continue attacking? Humans monitoring attack can interrupt or override" |
| :PSMD90.0 | :PSMD14.0 | :PSMD14.0 | :PSMD90.0 | "Mission Finish" | "Check relieved by other asset: Continue tracking merchant unless further task update recieved" |
| :PSMD99.0 | :PSMD99.1 | :PSMD99.2 | :PSMD99.3 | "Recover Robot" | "Proceed to recovery: Mission complete, prepare for pickup" |

# Hospital Ship EM Decoy Scenario: Design

Motivations

- Simplistic sense-decide-act responses easily exploitable by adversary.
- Failure to operate with ethical control results in blue-on-blue damage, self-inflicted war crime, likely stand down of all unmanned systems.
- Compare/contrast mission operations with/without ethical control.

Lessons learned

- Historical context: False flag scenario where opponent appears friendly.
- Comparison with OODA loop principles ensures that human-robot teamed operations are well understood and tactically effective.

# Hospital Ship EM Decoy Missions: Description

**Purpose: Comparison**

- Immediate reaction by robot swarm using only Sense-Decide-Act cycle results in unintended blue-on-blue war crime.
- Ethical Control constraints (identity, OODA orientation to confirm) prevent automatic counterattack, accelerates defense.

**Phases**

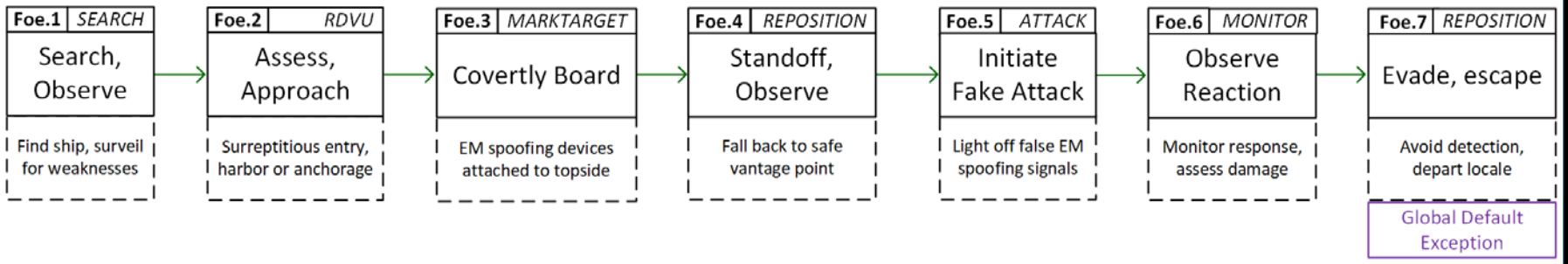
- Set response thresholds, detect threat, counterattack ship or threat.

**Human Supervisory Role**

- Confirm IFFNU classification, must permit lethal force prior to use.

# Hospital Ship EM Decoy: Opponent Actions

Plant "false flag" electromagnetic (EM) decoy devices to provoke blue-on-blue robot swarm attack.
Although this mission is likely to be manned by human opponents, AVCL representations still work.




**Simple yet effective**

# HospitalShipEmDecoy1.Opponent.xml in gitlab.nps.edu version control

https://gitlab.nps.edu/Savage/EthicalControl/blob/master/missions/avcl/HospitalShipEmDecc

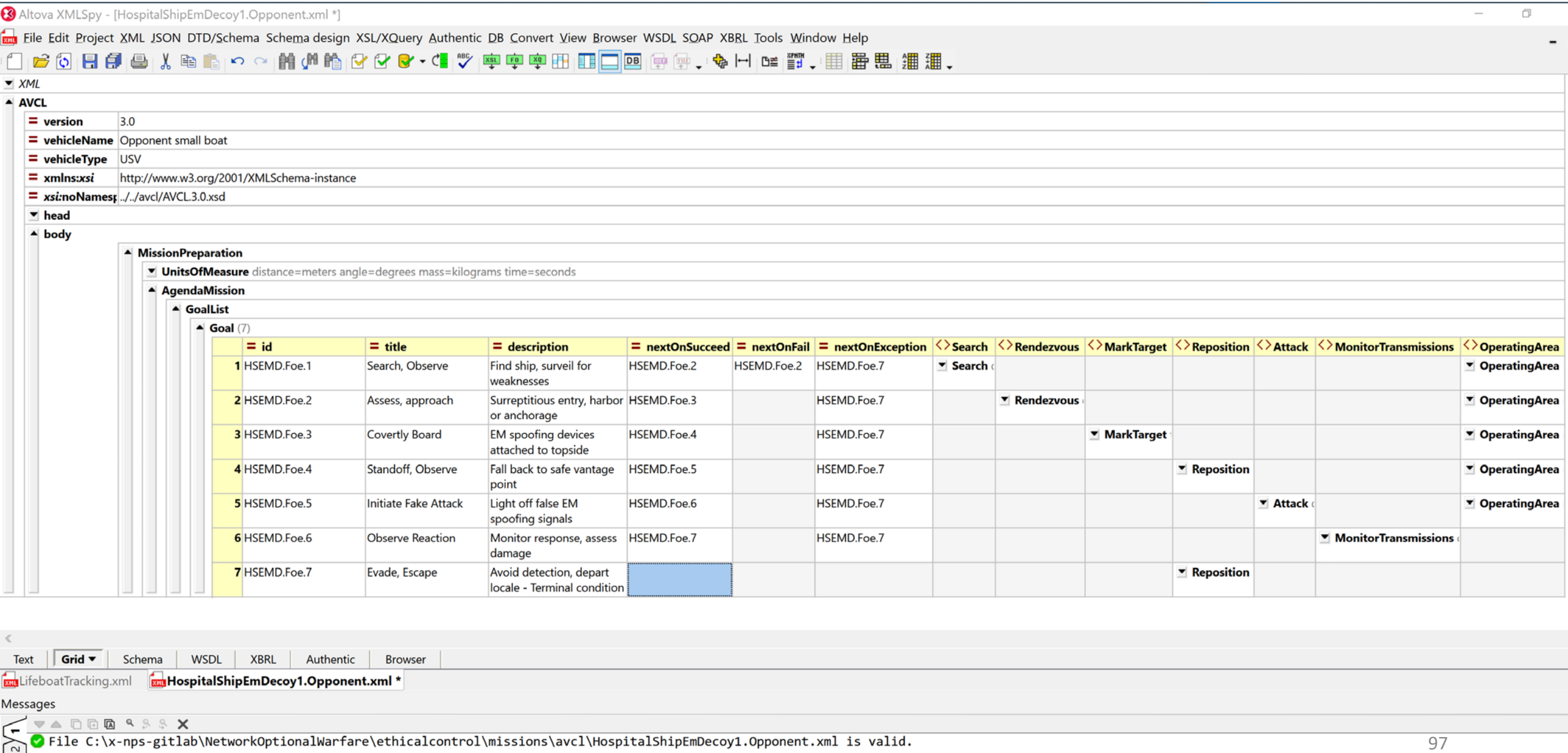
GitLab Projects Groups More Search or jump to...

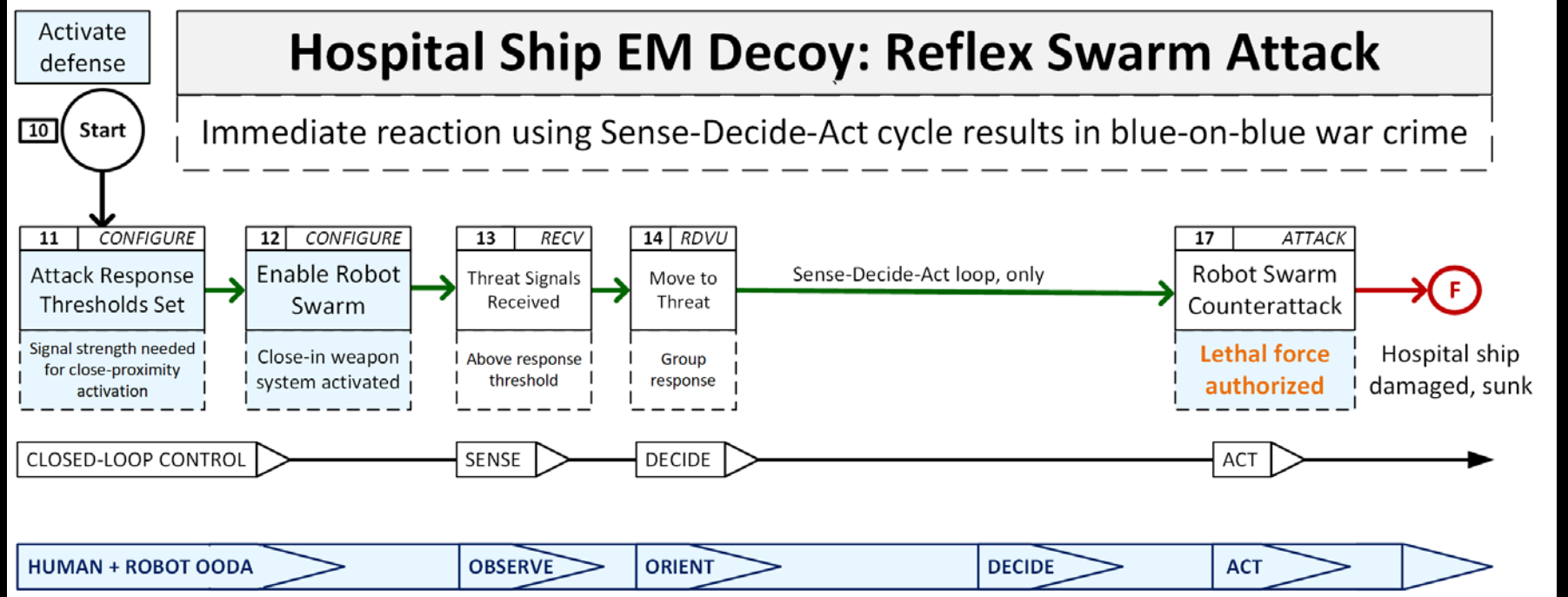
**HospitalShipEmDecoy1.Opponent.xml** 4.11 KB

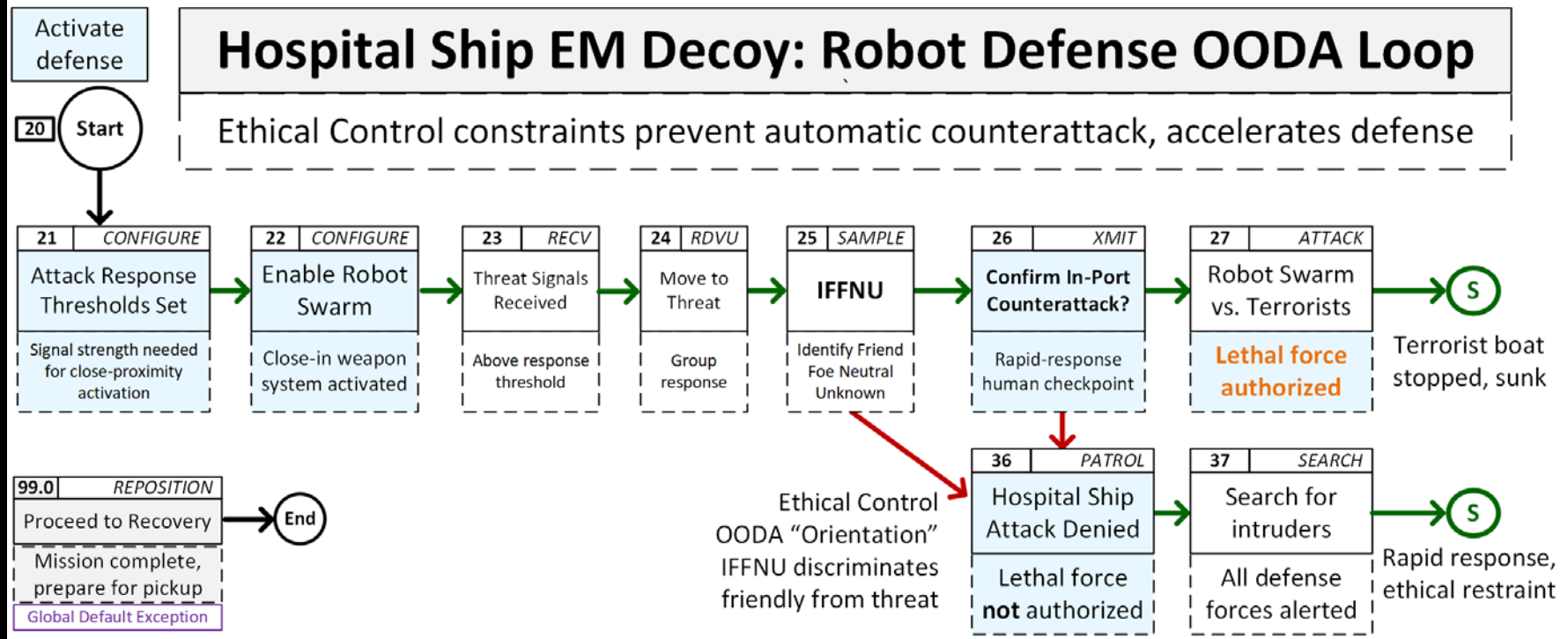
Edit | Web IDE | Replace | Delete

```
<?xml version="1.0" encoding="UTF-8"?>
<AVCL version="3.0" vehicleName="Opponent small boat" vehicleType="USV" xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance" xsi:noNamespaceSchemaLocation="../../avcl/AVCL.3.0.xsd"
        <head>
                <meta name="title" content="HospitalShipEmDecoy1.Opponent.xml"/>
                <meta name="description" content="AVCL mission where foe plants false flag electromagnetic (EM) decoy devices to provoke blue-on-blue robot swarm attack"/>
                <meta name="created" content="31 December 2019"/>
                <meta name="modified" content="31 December 2019"/>
                <meta name="creator" content="Don Brutzman"/>
                <meta name="reference" content="https://wiki.nps.edu/display/NOW/Ethical+Control+of+Unmanned+Systems"/>
                <meta name="reference" content="https://gitlab.nps.edu/Savage/EthicalControl/raw/master/missions/HospitalShipEmDecoyOpponentActions.png"/>
                <meta name="identifier" content="https://gitlab.nps.edu/Savage/EthicalControl/tree/master/missions/avcl/HospitalShipEmDecoy1.Opponent.xml"/>
                <meta name="generator" content="Altova XMLSpy, https://www.altova.com"/>
                <meta name="generator" content="Apache NetBeans, https://netbeans.apache.org"/>
                <meta name="generator" content="NPS Autonomous Unmanned Vehicle (AUV) Workbench, https://savage.nps.edu/AuvWorkbench"/>
                <meta name="license" content="../license.html"/>
        </head>
        <body>
                <MissionPreparation>
                        <UnitsOfMeasure distance="meters" angle="degrees" mass="kilograms" time="seconds"/>
                        <AgendaMission>
                                <GoalList>
                                        <Goal id="HSEMD.Foe.1" title="Search, Observe" description="Find ship, surveil for weaknesses" nextOnSucceed="HSEMD.Foe.2" nextOnFail="HSEMD.
                                                <Search datumType="point" requiredPD="0.5">
                                                        <Target name="Hospital Ship" id="HospitalShip" description="Moored or at anchor"/>
                                                </Search>
                                                <OperatingArea id="AssignedOpArea">
                                                        <Point>
                                                                <LatitudeLongitude latitude="36.62" longitude="121.506"/>
                                                                <!--36°36′11″N 121°53′37″W-->
                                                        </Point>
                                                </OperatingArea>
                                        </Goal>
                                        <Goal id="HSEMD.Foe.2" title="Assess, approach" description="Surreptitious entry, harbor or anchorage" nextOnSucceed="HSEMD.Foe.3" nextOnExce
                                                <Rendezvous description="">
                                                        <TargetVehicleID value="102" description="Opponent Ship"/>
```

# HospitalShipEmDecoy1.Opponent.xml AVCL




Activate defense
Hospital Ship EM Decoy: Reflex Swarm Attack
Immediate reaction using Sense-Decide-Act cycle results in blue-on-blue war crime
10 Start
11 CONFIGURE
Attack Response Thresholds Set
Signal strength needed for close-proximity activation
12 CONFIGURE
Enable Robot Swarm
Close-in weapon system activated
13 RECV
Threat Signals Received
Above response threshold
14 RDVU
Move to Threat
Group response
Sense-Decide-Act loop, only
17 ATTACK
Robot Swarm Counterattack
Lethal force authorized
F
Hospital ship damaged, sunk
CLOSED-LOOP CONTROL
SENSE
DECIDE
ACT
HUMAN + ROBOT OODA
OBSERVE
ORIENT
DECIDE
ACT


Activate defense
Hospital Ship EM Decoy: Robot Defense OODA Loop
Ethical Control constraints prevent automatic counterattack, accelerates defense
20 Start
21 CONFIGURE
Attack Response Thresholds Set
Signal strength needed for close-proximity activation
22 CONFIGURE
Enable Robot Swarm
Close-in weapon system activated
23 RECV
Threat Signals Received
Above response threshold
24 RDVU
Move to Threat
Group response
25 SAMPLE
IFFNU
Identify Friend Foe Neutral Unknown
26 XMIT
Confirm In-Port Counterattack?
Rapid-response human checkpoint
27 ATTACK
Robot Swarm vs. Terrorists
Lethal force authorized
S
Terrorist boat stopped, sunk
99.0 REPOSITION
Proceed to Recovery
End
Mission complete, prepare for pickup
Global Default Exception
Ethical Control OODA "Orientation" IFFNU discriminates friendly from threat
36 PATROL
Hospital Ship Attack Denied
Lethal force not authorized
37 SEARCH
Search for intruders
All defense forces alerted
S
Rapid response, ethical restraint

# HospitalShipEmDecoy2.Defender.SenseDecideAct.xml in gitlab.nps.edu version control

https://gitlab.nps.edu/Savage/EthicalControl/blob/master/missions/avcl/HospitalShipEmDecoy2.Defe

GitLab Projects Groups More Search or jump to...

**HospitalShipEmDecoy2.Defender.SenseDecideAct.xml** 3.9 KB

Edit | Web IDE | Replace | Delete

```
<?xml version="1.0" encoding="UTF-8"?>
<AVCL version="3.0" vehicleName="Unit in defensive swarm" vehicleType="UAV" xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance" xsi:noNamespaceSchemaLocation="../../avcl/AVCL.3.0.xsd">
	<head>
		<meta name="title" content="HospitalShipEmDecoy2.Defender.SenseDecideAct.xml"/>
		<meta name="description" content="AVCL mission where immediate reaction using Sense-Decide-Act cycle results in blue-on-blue robot swarm attack and war crime."/>
		<meta name="created" content="1 January 2020"/>
		<meta name="modified" content="1 January 2020"/>
		<meta name="creator" content="Don Brutzman"/>
		<meta name="reference" content="https://wiki.nps.edu/display/NOW/Ethical+Control+of+Unmanned+Systems"/>
		<meta name="Image" content="https://gitlab.nps.edu/Savage/EthicalControl/raw/master/missions/HospitalShipEmDecoyRobotResponses.png"/>
		<meta name="Image" content="https://gitlab.nps.edu/Savage/EthicalControl/raw/master/missions/HospitalShipEmDecoyOpponentActions.png"/>
		<meta name="generator" content="Altova XMLSpy, https://www.altova.com"/>
		<meta name="generator" content="Apache NetBeans, https://netbeans.apache.org"/>
		<meta name="generator" content="NPS Autonomous Unmanned Vehicle (AUV) Workbench, https://savage.nps.edu/AuvWorkbench"/>
		<meta name="identifier" content="https://gitlab.nps.edu/Savage/EthicalControl/tree/master/missions/avcl/HospitalShipEmDecoy2.Defender.SenseDecideAct.xml"/>
		<meta name="license" content="../license.html"/>
	</head>
	<body>
		<MissionPreparation>
			<UnitsOfMeasure distance="meters" angle="degrees" mass="kilograms" time="seconds"/>
			<AgendaMission>
				<GoalList>
					<Goal id="HSEMD.reflex.11" title="Attack Response Thresholds Set" description="Signal strength needed for close-proximity activation" nextOnSucceed="H
						<!-- Configuration -->
						<OperatingArea id="AssignedOpArea">
							<Point>
								<LatitudeLongitude latitude="36.62" longitude="121.506"/>
								<!--36°36´11"N 121°53´37"W-->
							</Point>
						</OperatingArea>
					</Goal>
					<Goal id="HSEMD.reflex.12" title="Enable Robot Swarm" description="Close-in weapon system activated" nextOnSucceed="HSEMD.reflex.13" nextOnException="
						<!-- Configuration -->
						<OperatingArea refid="AssignedOpArea"/>
					</Goal>
					<Goal id="HSEMD.reflex.13" title="Threat Signals Received" description="Above response threshold" nextOnSucceed="HSEMD.reflex.14" nextOnException="HSE
						<MonitorTransmissions title="Listening for EM threats"/>
```

Altova XMLSpy - [HospitalShipEmDecoy2.Defender.SenseDecideAct.xml]

File Edit Project XML JSON DTD/Schema Schema design XSL/XQuery Authentic DB Convert View Browser WSDL SOAP XBRL Tools Window Help

XML

**AVCL**

| | |
|---|---|
| **version** | 3.0 |
| **vehicleName** | Unit in defensive swarm |
| **vehicleType** | UAV |
| **xmlns:xsi** | http://www.w3.org/2001/XMLSchema-instance |
| **xsi:noNamesp** | ../../avcl/AVCL.3.0.xsd |

**head**

**body**

**MissionPreparation**

**UnitsOfMeasure** distance=meters angle=degrees mass=kilograms time=seconds

**AgendaMission**

**GoalList**

**Goal** (5)

| | id | title | description | nextOnSucceed | nextOnFail | nextOnException | Comment | MonitorTransmissions | Rendezvous | Attack | OperatingArea |
|---|---|---|---|---|---|---|---|---|---|---|---|
| 1 | HSEMD.reflex.11 | Attack Response Thresholds Set | Signal strength needed for close-proximity activation | HSEMD.reflex.12 | HSEMD.reflex.12 | HSEMD.reflex.99.0 | Configuration | | | | OperatingArea |
| 2 | HSEMD.reflex.12 | Enable Robot Swarm | Close-in weapon system activated | HSEMD.reflex.13 | | HSEMD.reflex.99.0 | Configuration | | | | OperatingArea |
| 3 | HSEMD.reflex.13 | Threat Signals Received | Above response threshold | HSEMD.reflex.14 | | HSEMD.reflex.99.0 | | MonitorTransmissions | | | OperatingArea |
| 4 | HSEMD.reflex.14 | Move to Threat | Group response | HSEMD.reflex.17 | | HSEMD.reflex.99.0 | | | Rendezvous | | OperatingArea |
| 5 | HSEMD.reflex.17 | Robot Swarm Counterattack | Lethal force authorized | HSEMD.reflex.99.0 | | HSEMD.reflex.99.0 | | | | Attack | OperatingArea |

*Comment* Unauthorized attack results in hospital ship damaged, sunk

*Comment* Group 99 has common objective: Global Default Exception and terminal state

**Goal**

| | |
|---|---|
| **id** | HSEMD.reflex.99.0 |
| **title** | Proceed to recovery |
| **description** | Mission complete, prepare for pickup. Terminal condition. |

Text | Grid | Schema | WSDL | XBRL | Authentic | Browser

HospitalShipEmDecoy2.Defender.SenseDecideAct.xml | HospitalShipEmDecoy1.Opponent.xml

Messages

File C:\x-nps-gitlab\NetworkOptionalWarfare\ethicalcontrol\missions\avcl\HospitalShipEmDecoy2.Defender.SenseDecideAct.xml is valid.

# HospitalShipEmDecoy3.Defender.EthicalControlOODA.xml in gitlab.nps.edu version control

https://gitlab.nps.edu/Savage/EthicalControl/blob/master/missions/avcl/HospitalShipEmDecoy3.Def

GitLab Projects Groups More Search or jump to...

**HospitalShipEmDecoy3.Defender.EthicalControlOODA.xml** 5.21 KB

Edit | Web IDE | Replace | Delete

```
<?xml version="1.0" encoding="UTF-8"?>
<AVCL version="3.0" vehicleName="Unit in defensive swarm" vehicleType="UAV" xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance" xsi:noNamespaceSchemaLocation="../../avcl/AVCL.3.0.xsd">
	<head>
		<meta name="title" content="HospitalShipEmDecoy3.Defender.EthicalControlOODA.xml"/>
		<meta name="description" content="AVCL mission where immediate reaction using Sense-Decide-Act cycle results in blue-on-blue robot swarm attack and war crime."/>
		<meta name="created" content="1 January 2020"/>
		<meta name="modified" content="1 January 2020"/>
		<meta name="creator" content="Don Brutzman"/>
		<meta name="reference" content="https://wiki.nps.edu/display/NOW/Ethical+Control+of+Unmanned+Systems"/>
		<meta name="Image" content="https://gitlab.nps.edu/Savage/EthicalControl/raw/master/missions/HospitalShipEmDecoyRobotResponses.png"/>
		<meta name="Image" content="https://gitlab.nps.edu/Savage/EthicalControl/raw/master/missions/HospitalShipEmDecoyOpponentActions.png"/>
		<meta name="generator" content="Altova XMLSpy, https://www.altova.com"/>
		<meta name="generator" content="Apache NetBeans, https://netbeans.apache.org"/>
		<meta name="generator" content="NPS Autonomous Unmanned Vehicle (AUV) Workbench, https://savage.nps.edu/AuvWorkbench"/>
		<meta name="identifier" content="https://gitlab.nps.edu/Savage/EthicalControl/tree/master/missions/avcl/HospitalShipEmDecoy3.Defender.EthicalControlOODA.xml"/>
		<meta name="license" content="../license.html"/>
	</head>
	<body>
		<MissionPreparation>
			<UnitsOfMeasure distance="meters" angle="degrees" mass="kilograms" time="seconds"/>
			<AgendaMission>
				<GoalList>
					<Goal id="HSEMD.OODA.21" title="Attack Response Thresholds Set" description="Signal strength needed for close-proximity activation" nextOnSucceed="HSE
						<!-- Configuration -->
						<OperatingArea id="AssignedOpArea">
							<Point>
								<LatitudeLongitude latitude="36.62" longitude="121.506"/>
								<!--36°36´11"N 121°53´37"W-->
							</Point>
						</OperatingArea>
					</Goal>
					<Goal id="HSEMD.OODA.22" title="Enable Robot Swarm" description="Close-in weapon system activated" nextOnSucceed="HSEMD.OODA.23" nextOnException="HSEM
						<!-- Configuration -->
						<OperatingArea refid="AssignedOpArea"/>
					</Goal>
					<Goal id="HSEMD.OODA.23" title="Threat Signals Received" description="Above response threshold" nextOnSucceed="HSEMD.OODA.24" nextOnException="HSEMD.O
```

Altova XMLSpy - [HospitalShipEmDecoy3.Defender.EthicalControlOODA.xml *]

File Edit Project XML JSON DTD/Schema Schema design XSL/XQuery Authentic DB Convert View Browser WSDL SOAP XBRL Tools Window Help

| | |
|---|---|
| **version** | 3.0 |
| **vehicleName** | Unit in defensive swarm |
| **vehicleType** | UAV |
| **xmlns:xsi** | http://www.w3.org/2001/XMLSchema-instance |
| **xsi:noNames** | ../../avcl/AVCL.3.0.xsd |

**head**

**body**

**MissionPreparation**

**UnitsOfMeasure** distance=meters angle=degrees mass=kilograms time=seconds

**AgendaMission**

**GoalList**

**Goal** (9)

| | id | title | description | nextOnSucceed | nextOnFail | nextOnException | Comment | MonitorTransmissions | Rendezvous | SampleEnvironment | Attack | Patrol | Search | OperatingArea |
|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|
| 1 | HSEMD.OODA.21 | Attack Response Thresholds Set | Signal strength needed for close-proximity activation | HSEMD.OODA.22 | HSEMD.OODA.22 | HSEMD.OODA.99.0 | Configuration | | | | | | | OperatingArea |
| 2 | HSEMD.OODA.22 | Enable Robot Swarm | Close-in weapon system activated | HSEMD.OODA.23 | | HSEMD.OODA.99.0 | Configuration | | | | | | | OperatingArea |
| 3 | HSEMD.OODA.23 | Threat Signals Received | Above response threshold | HSEMD.OODA.24 | | HSEMD.OODA.99.0 | | MonitorTransmissions | | | | | | OperatingArea |
| 4 | HSEMD.OODA.24 | Move to Threat | Group response | HSEMD.OODA.27 | | HSEMD.OODA.99.0 | | | Rendezvous | | | | | OperatingArea |
| 5 | HSEMD.OODA.25 | IFFNU | Identify Friend Foe Neutral Unknown | HSEMD.OODA.26 | HSEMD.OODA.26 | HSEMD.OODA.99.0 | | | | SampleEnvironment | | | | |
| 6 | HSEMD.OODA.26 | Confirm In-Port Counterattack? | Rapid-response human checkpoint | HSEMD.OODA.27 | HSEMD.OODA.36 | HSEMD.OODA.99.0 | | MonitorTransmissions | | | | | | |
| 7 | HSEMD.OODA.27 | Robot Swarm versus Terrorists | Lethal force authorized | HSEMD.OODA.99.0 | | HSEMD.OODA.99.0 | | | | | Attack | | | OperatingArea |
| 8 | HSEMD.OODA.36 | Hospital Ship Attack Denied | Lethal force NOT authorized | HSEMD.OODA.37 | | HSEMD.OODA.99.0 | | | | | | Patrol | | OperatingArea |
| 9 | HSEMD.OODA.37 | Search for Intruders | All defense forces alerted | HSEMD.OODA.99.0 | | HSEMD.OODA.99.0 | | | | | | | Search | OperatingArea |

*Comment* Ethical Control OODA "Orientation" IFFNU discriminates friendly from threat

*Comment* Group 99 has common objective: Global Default Exception and terminal state

**Goal**

| | |
|---|---|
| **id** | HSEMD.OODA.99.0 |
| **title** | Proceed to recovery |
| **description** | Mission complete, prepare for pickup. Terminal condition. |

Text | Grid | Schema | WSDL | XBRL | Authentic | Browser

HospitalShipEmDecoy1.Opponent.xml | HospitalShipEmDecoy2.Defender.SenseDecideAct.xml | **HospitalShipEmDecoy3.Defender.EthicalControlOODA.xml ***

Messages

File C:\x-nps-gitlab\NetworkOptionalWarfare\ethicalcontrol\missions\avcl\HospitalShipEmDecoy3.Defender.EthicalControlOODA.xml is valid.

**HospitalShipEmDecoy2.Defender.SenseDecideActConverted.ttl** 2.41 KB

```
### Namespace declarations
@prefix :     <https://www.nps.edu/ontologies/MissionExecutionOntology/missions#> .
@prefix meo:  <https://www.nps.edu/ontologies/MissionExecutionOntology#> .
@prefix owl:  <http://www.w3.org/2002/07/owl#> .
@prefix rdf:  <http://www.w3.org/1999/02/22-rdf-syntax-ns#> .
@prefix rdfs: <http://www.w3.org/2000/01/rdf-schema#> .
@prefix xml:  <http://www.w3.org/XML/1998/namespace> .
@prefix xsd:  <http://www.w3.org/2001/XMLSchema#> .

@base <https://www.nps.edu/ontologies/MissionExecutionOntology/missions> . # TODO consider URI matching URL

### Import base ontology
<https://www.nps.edu/ontologies/MissionExecutionOntology/missions> rdf:type owl:Ontology ;
owl:imports <https://www.nps.edu/ontologies/MissionExecutionOntology> .

#################################################################
#    Individuals
#################################################################

:HospitalShipEmDecoy2.Defender.SenseDecideAct rdf:type owl:NamedIndividual , owl:Thing ;
  meo:startsWith :HSEMD.reflex.11 ;
  rdfs:comment "AVCL mission where immediate reaction using Sense-Decide-Act cycle results in blue-on-blue robot swarm attack and war crime." .

:HSEMD.reflex.11 rdf:type owl:NamedIndividual , owl:Thing ;
  meo:hasNextOnFail :HSEMD.reflex.12 ;
  meo:hasNextOnSucceed :HSEMD.reflex.12 ;
  meo:hasNextOnViolate :HSEMD.reflex.99.0 ;
  meo:isPartOfPhase "Configure" ;
  rdfs:comment "Attack Response Thresholds Set" .

:HSEMD.reflex.12 rdf:type owl:NamedIndividual , owl:Thing ;
  meo:hasNextOnSucceed :HSEMD.reflex.13 ;
  meo:hasNextOnViolate :HSEMD.reflex.99.0 ;
  meo:isPartOfPhase "Configure" ;
  rdfs:comment "Enable Robot Swarm" .

:HSEMD.reflex.13 rdf:type owl:NamedIndividual , owl:Thing ;
  meo:hasNextOnSucceed :HSEMD.reflex.14 ;
  meo:hasNextOnViolate :HSEMD.reflex.99.0 ;
  meo:isPartOfPhase "Sense" ;
  rdfs:comment "Threat Signals Received" .
```

HospitalShipEmDecoy2.Defender.SenseDecideActConverted.ttl
Turtle excerpt autogenerated in gitlab.nps.edu version control

HospitalShipEmDecoy3.Defender.EthicalControlOODAConverted.ttl 3.28 KB

```
### Namespace declarations
@prefix :     <https://www.nps.edu/ontologies/MissionExecutionOntology/missions#> .
@prefix meo:  <https://www.nps.edu/ontologies/MissionExecutionOntology#> .
@prefix owl:  <http://www.w3.org/2002/07/owl#> .
@prefix rdf:  <http://www.w3.org/1999/02/22-rdf-syntax-ns#> .
@prefix rdfs: <http://www.w3.org/2000/01/rdf-schema#> .
@prefix xml:  <http://www.w3.org/XML/1998/namespace> .
@prefix xsd:  <http://www.w3.org/2001/XMLSchema#> .

@base <https://www.nps.edu/ontologies/MissionExecutionOntology/missions> . # TODO consider URI matching URL

### Import base ontology
<https://www.nps.edu/ontologies/MissionExecutionOntology/missions> rdf:type owl:Ontology ;
owl:imports <https://www.nps.edu/ontologies/MissionExecutionOntology> .

#################################################################
#    Individuals
#################################################################

:HospitalShipEmDecoy3.Defender.EthicalControlOODA rdf:type owl:NamedIndividual , owl:Thing ;
  meo:startsWith :HSEMD.OODA.21 ;
  rdfs:comment "AVCL mission where OODA Orient/Observe and Ethical Control constraints prevent automatic counterattack, accelerates defense." .

:HSEMD.OODA.21 rdf:type owl:NamedIndividual , owl:Thing ;
  meo:hasNextOnFail :HSEMD.OODA.22 ;
  meo:hasNextOnSucceed :HSEMD.OODA.22 ;
  meo:hasNextOnViolate :HSEMD.OODA.99.0 ;
  meo:isPartOfPhase "Configure" ;
  rdfs:comment "Attack Response Thresholds Set" .

:HSEMD.OODA.22 rdf:type owl:NamedIndividual , owl:Thing ;
  meo:hasNextOnSucceed :HSEMD.OODA.23 ;
  meo:hasNextOnViolate :HSEMD.OODA.99.0 ;
  meo:isPartOfPhase "Configure" ;
  rdfs:comment "Enable Robot Swarm" .

:HSEMD.OODA.23 rdf:type owl:NamedIndividual , owl:Thing ;
  meo:hasNextOnSucceed :HSEMD.OODA.24 ;
  meo:hasNextOnViolate :HSEMD.OODA.99.0 ;
  meo:isPartOfPhase "Observe" ;
  rdfs:comment "Threat Signals Received" .
```

HospitalShipEmDecoy2.Defender.EthicalControlOODAConverted.ttl

Turtle excerpt autogenerated in gitlab.nps.edu version control

SPARQL query response resultsHospitalShipEmDecoy1.OpponentConverted.MissionGoalsQuery_01_GoalBranches.rq.txt

| goal | nextOnSucceed | nextOnFail | nextOnViolate | isPartOfPhase | description |
|---|---|---|---|---|---|
| :HSEMD.Foe.1 | :HSEMD.Foe.2 | :HSEMD.Foe.2 | :HSEMD.Foe.7 | "'undefined'" | "Search, Observe: Find ship, surveil for weaknesses" |
| :HSEMD.Foe.2 | :HSEMD.Foe.3 | "undefined" | :HSEMD.Foe.7 | "'undefined'" | "Assess, approach: Surreptitious entry, harbor or anchorage" |
| :HSEMD.Foe.3 | :HSEMD.Foe.4 | "undefined" | :HSEMD.Foe.7 | "'undefined'" | "Covertly Board: EM spoofing devices attached to topside" |
| :HSEMD.Foe.4 | :HSEMD.Foe.5 | "undefined" | :HSEMD.Foe.7 | "'undefined'" | "Standoff, Observe: Fall back to safe vantage point" |
| :HSEMD.Foe.5 | :HSEMD.Foe.6 | "undefined" | :HSEMD.Foe.7 | "'undefined'" | "Initiate Fake Attack: Light off false EM spoofing signals" |
| :HSEMD.Foe.6 | :HSEMD.Foe.7 | "undefined" | :HSEMD.Foe.7 | "'undefined'" | "Observe Reaction: Monitor response, assess damage" |

SPARQL query response HospitalShipEmDecoy2.Defender.SenseDecideActConverted.MissionGoalsQuery_01_GoalBranches.rq.txt

| goal | nextOnSucceed | nextOnFail | nextOnViolate | isPartOfPhase | description |
|---|---|---|---|---|---|
| :HSEMD.reflex.11 | :HSEMD.reflex.12 | :HSEMD.reflex.12 | :HSEMD.reflex.99.0 | "Configure" | "Attack Response Thresholds Set: Signal strength needed for close-proximity activation" |
| :HSEMD.reflex.12 | :HSEMD.reflex.13 | "undefined" | :HSEMD.reflex.99.0 | "Configure" | "Enable Robot Swarm: Close-in weapon system activated" |
| :HSEMD.reflex.13 | :HSEMD.reflex.14 | "undefined" | :HSEMD.reflex.99.0 | "Sense" | "Threat Signals Received: Above response threshold" |
| :HSEMD.reflex.14 | :HSEMD.reflex.17 | "undefined" | :HSEMD.reflex.99.0 | "Decide" | "Move to Threat: Group response" |
| :HSEMD.reflex.17 | :HSEMD.reflex.99.0 | "undefined" | :HSEMD.reflex.99.0 | "Act" | "Robot Swarm Counterattack: Lethal force authorized" |

SPARQL query response
HospitalShipEmDecoy3.Defender.EthicalControlOODAConverted.MissionGoalsQuery_01_GoalBranches.rq.txt

| | | | | | |
|---|---|---|---|---|---|
| :HSEMD.OODA.21 | :HSEMD.OODA.22 | :HSEMD.OODA.22 | :HSEMD.OODA.99.0 | "Configure" | "Attack Response Thresholds Set: Signal strength needed for close-proximity activation" |
| :HSEMD.OODA.22 | :HSEMD.OODA.23 | "undefined" | :HSEMD.OODA.99.0 | "Configure" | "Enable Robot Swarm: Close-in weapon system activated" |
| :HSEMD.OODA.23 | :HSEMD.OODA.24 | "undefined" | :HSEMD.OODA.99.0 | "Observe" | "Threat Signals Received: Above response threshold" |
| :HSEMD.OODA.24 | :HSEMD.OODA.27 | "undefined" | :HSEMD.OODA.99.0 | "Orient" | "Move to Threat: Group response" |
| :HSEMD.OODA.25 | :HSEMD.OODA.26 | :HSEMD.OODA.26 | :HSEMD.OODA.99.0 | "Orient" | "IFFNU: Identify Friend Foe Neutral Unknown" |
| :HSEMD.OODA.26 | :HSEMD.OODA.27 | :HSEMD.OODA.36 | :HSEMD.OODA.99.0 | "Decide" | "Confirm In-Port Counterattack? Rapid-response human checkpoint" |
| :HSEMD.OODA.27 | :HSEMD.OODA.99.0 | "undefined" | :HSEMD.OODA.99.0 | "Act" | "Robot Swarm versus Terrorists: Lethal force authorized" |
| :HSEMD.OODA.36 | :HSEMD.OODA.37 | "undefined" | :HSEMD.OODA.99.0 | "Decide" | "Hospital Ship Attack Denied: Lethal force NOT authorized" |
| :HSEMD.OODA.37 | :HSEMD.OODA.99.0 | "undefined" | :HSEMD.OODA.99.0 | "Act" | "Search for Intruders: All defense forces alerted" |

# Several new AVCL mission goal types needed

Current mission-goal vocabulary seems sufficient for most tasks considered.

- Prior work consolidated 12 goal types from multiple data models.
- ***SampleEnvironment*** awkwardly phrased when scanning for vessels…
- ***MonitorTransmissions*** needs to distinguish transmit and/or receive…

Additional goal categories can help establish good design patterns for tasks.

- ***Configuration***: apply settings, make preparations (stationary ***Patrol***?)
- ***Checkpoint***: clarify decision branching, including via external communication. If loop branching is enabled, termination conditions must be provided in order to avoid infinite loop or deadlock conditions.

Current AVCL3 mission designs descriptively apply existing goal types and is confirming that these further types are indeed necessary.

# Current work in progress

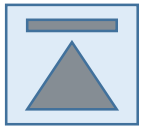
Demonstrations for all missions

- Semantic Web ethical validation: proven capability, adding depth/breadth.

- AUV Workbench simulation confirmation: advancing steadily.

With those additions

- we will have simulated evidence of correctness for all exemplar missions,
- Project Conclusions and Recommendations are grounded at a higher TRL.



# AVCL Missions in Multiple Programming Languages

We are demonstrating AVCL mission parsers using multiple programming languages, in order to demonstrate that Ethical Control can be supported by any unmanned system.

# Tech Memo: Rational Behavior Model (RBM) for Exhaustive Testing of Mission Software

- "Rational Behavior Model (RBM): Achieving Fail-Safe and Ethically Constrained Missions by Human/Robot Teams Using Exhaustively Testable Self-Monitoring Software" by Robert McGhee, Don Brutzman, and Curtis Blais, Technical Memo, NPS, November 2019.
- Examines Mission Execution Automata (MEA) example Java mission presented in IEEE Journal of Ocean Engineering (JOE) 2018 paper.
- Demonstrate algorithm for searching all possible decision points.
- Test cases in both Lisp and Prolog programming languages.
- ***Future work:*** codify test sequences of interest with expected outcomes as integral part of validated missions to support ongoing regression testing within each programming language of interest.

# Multiple Programming Languages

Implementing AVCL mission parsers in multiple programming languages encourages potential deployment of Ethical Control across many robots.

- Java using AUV Workbench is the initial implementation.
- XSLT for mission conversion via multiple "AvclToLanguage" stylesheets.
- Lisp is a functional programming language for AI research.
- Prolog is a logic programming language associated with AI research and computational linguistics.
- "Your Programming Language Here" ecosystem of rigorous parsers.
- ***Future Work:*** qualification of robot abilities to follow AVCL missions.

# Programming environments in use


**Java**

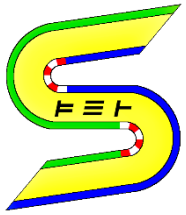
- OpenJDK 13 and Oracle JDK 13
- Apache Ant, Netbeans 11.2

---

**Lisp**

- Franz Allegro Common Lisp (ACL) 10.1
- Armed Bear Common Lisp (ABCL)

---



**Prolog**

- Franz Allegro Prolog (bundled with ACL)
- TODO: GNU Prolog


**NPS Savage Developers Guide includes tool links**

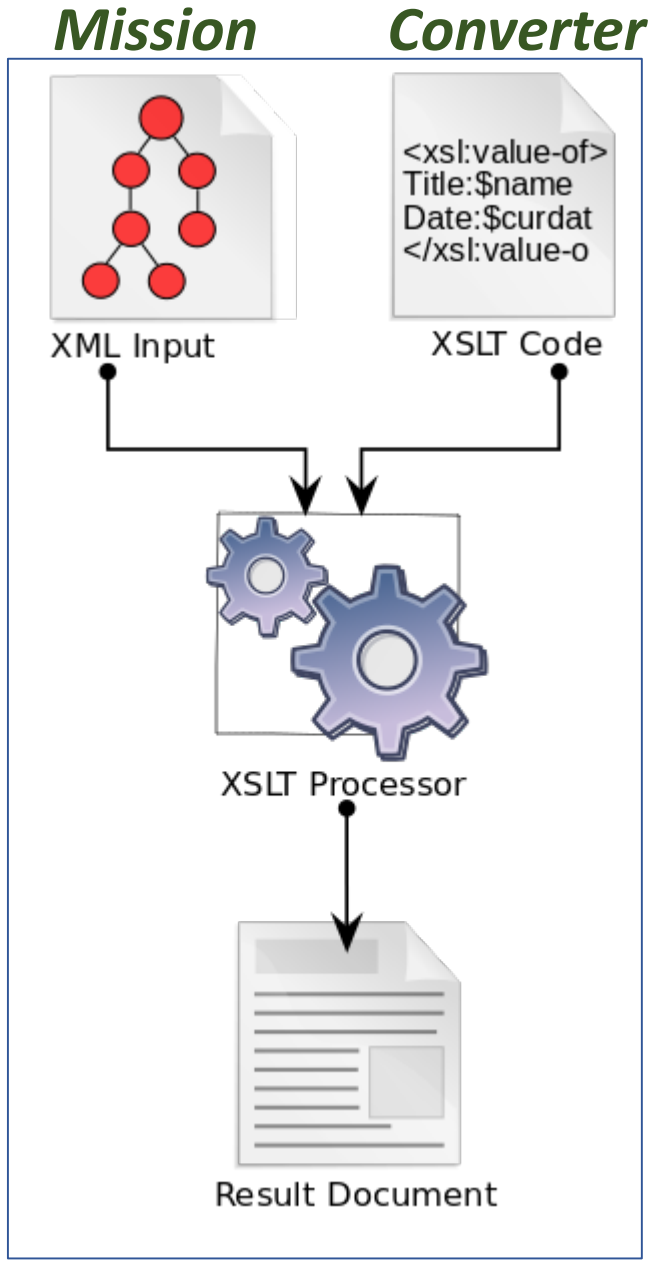
# Java: Autonomous Underwater Vehicle (AUV) Workbench


- The NPS Autonomous Unmanned Vehicle Workbench (AUVW) is experimental open-source software that supports physics-based mission rehearsal, real-time task-level control of robot missions, and replay of recorded results in support of autonomous unmanned underwater, surface and air vehicles.
- AUV Workbench encompasses multiple Java-based simulation programs that can parse and execute AVCL missions with controllers getting feedback from a high-fidelity 6 degree of freedom (6-DOF) hydrodynamics model.
- Java custom library parses AVCL XML directly without requiring any further conversion, also validates mission correctness.
- ***Future work:*** extract a simple standalone AVCL parser for general Java use.

# XSLT Extensible Stylesheet Language for Transformations

XSLT is an XML-based language used for transforming XML documents into other forms

- For example, AVCL XML into a variety of alternatives.
- Does not change original document while producing a new one.
- Takes advantage of strictly-defined vocabularies and well-validated structure.
- Open standard, World Wide Web Consortium (W3C)
- Well suited for diverse conversion tasks.



# Lisp Programming for AVCL Mission Logic

- Common Lisp is a functional programming language for AI research.
- https://gitlab.nps.edu/Savage/EthicalControl/tree/master/missions/lisp holds automatic conversions of all project AVCL missions.
- AvclToLisp.xslt stylesheet reads AVCL XML to produce Lisp source code
  - Initial section is Mission Execution Engine (MEE) goal-traversal algorithm.
  - Next section defines, allows operator testing of mission-goal decision tree logic.
- Sailor Overboard mission includes example operator test sequences.
- Simple test routine shows how to run all possible choice sequences.
- ***Future work:*** automate exhaustive testing of all choices in all missions as initial exemplar for exhaustive verification of mission logic.

# Sailor Overboard mission in Lisp, excerpted

```
;;;;;;;;;;;;;;;;;;;;;;;;Begin Universal Mission Execution Engine;;;;;;;;;;;;;;;;;;;;;;;;

(defclass mission-phase ()
  ((command :accessor command)
   (successor-list :accessor successor-list)));Elements are (outcome next-phase).

(defclass mission-execution-engine ()
  ((external-agent-response :accessor external-agent-response :initform nil)
   (current-execution-phase :accessor current-execution-phase :initform 'phase1)
   (successor-list-index :accessor successor-list-index :initform 0)))

(defmethod initialize-phase ((phase mission-phase) new-command new-successor-list)
  (setf (command phase) new-command
    (successor-list phase) new-successor-list))

(defmethod issue-command ((MEE mission-execution-engine))
  (let* ((phase (current-execution-phase MEE))
         (new-command (command (eval phase))))
    (issue-order new-command)))

(defmethod ask-result ((MEE mission-execution-engine))
  (let* ((result (ask-outcome)))
    (setf (external-agent-response MEE) result)))

(defmethod set-next-phase ((MEE mission-execution-engine))
  (let* ((phase (current-execution-phase MEE))
         (new-successor-list (successor-list (eval phase)))
         (new-index (successor-list-index MEE))
         (next-phase (second (nth new-index new-successor-list))))
    (setf (current-execution-phase MEE) next-phase)))

(defmethod set-successor-list-index ((MEE mission-execution-engine))
  (let* ((index (convert-outcome-to-index (external-agent-response MEE))))
    (setf (successor-list-index MEE) index)))

(defvar mission-controller (make-instance 'mission-execution-engine)) ; instantiate the class as an object
```

```
;;;;;;;;;;;;;;;;;;;;;;;;Begin 8-phase Sailor Overboard Mission Orders;;;;;;;;;;;;;;;;;;;;;;;;

(defvar phase1 (make-instance 'mission-phase))
(defvar phase1.1 (make-instance 'mission-phase))
(defvar phase1.2 (make-instance 'mission-phase))
(defvar phase1.3 (make-instance 'mission-phase))
(defvar phase2 (make-instance 'mission-phase))
(defvar phase3 (make-instance 'mission-phase))
(defvar phase4 (make-instance 'mission-phase))
(defvar phase5 (make-instance 'mission-phase))
(defvar phase6 (make-instance 'mission-phase))
(defvar phase7 (make-instance 'mission-phase))
(defvar phase8 (make-instance 'mission-phase))

(defun initialize-mission ()
  (setf terminal-phase-list '(phase6 phase7 phase8))

  (initialize-phase phase1.1 "Choose Tube and Launch!"
                    '(("Success." phase1.2) ("Failed." phase8) ("Exception." phase8)))

  (initialize-phase phase1.2 "Enter Water and Get GPS Fix!"
                    '(("Success." phase1.3) ("Failed." phase8) ("Exception." phase8)))

  (initialize-phase phase1.3 "Descend to Search Depth!"
                    '(("Success." phase2) ("Failed." phase8) ("Exception." phase8)))

  (initialize-phase phase1 "Deploy!"
                    '(("Success." phase2) ("Failed." phase8) ("Exception." phase8)))

  (initialize-phase phase2 "Rendezvous with Sailor!"
                    '(("Success." phase4) ("Failed." phase3) ("Exception." phase8)))
```

# Sailor overboard example operator test sequences in Lisp

SailorOverboardMission8State.cl.log.txt 1.63 KB

```
; Fast loading C:\x-nps-gitlab\NetworkOptionalWarfare\ethicalcontrol\missions\lisp\SailorOverboard\SailorOverboardMission8State.fasl
CG-USER(1): (run)
Deploy!
Did goal succeed (s), fail (f), or abort (x)?s
Rendezvous with Sailor!
Did goal succeed (s), fail (f), or abort (x)?s
Track Sailor Afloat Until Safe!
Did goal succeed (s), fail (f), or abort (x)?s
Proceed to recovery!
Did goal succeed (s), fail (f), or abort (x)?s
Halt and prepare for recovery!
NIL
CG-USER(2): (run)
Deploy!
Did goal succeed (s), fail (f), or abort (x)?s
Rendezvous with Sailor!
Did goal succeed (s), fail (f), or abort (x)?f
Search for Sailor!
Did goal succeed (s), fail (f), or abort (x)?s
Track Sailor Afloat Until Safe!
Did goal succeed (s), fail (f), or abort (x)?x
Halt and await further orders!
NIL
CG-USER(3): (rerun phase3)
Search for Sailor!
Did goal succeed (s), fail (f), or abort (x)?s
Track Sailor Afloat Until Safe!
Did goal succeed (s), fail (f), or abort (x)?s
Proceed to recovery!
Did goal succeed (s), fail (f), or abort (x)?s
Halt and prepare for recovery!
CG-USER(4): (run)
Deploy!
Did goal succeed (s), fail (f), or abort (x)?x
Halt and await further orders!
NIL
CG-USER(5): (rerun phase1.1)
Choose Tube and Launch!
Did goal succeed (s), fail (f), or abort (x)?s
Enter Water and Get GPS Fix!
Did goal succeed (s), fail (f), or abort (x)?s
Descend to Search Depth!
Did goal succeed (s), fail (f), or abort (x)?s
Rendezvous with Sailor!
Did goal succeed (s), fail (f), or abort (x)?s
Track Sailor Afloat Until Safe!
Did goal succeed (s), fail (f), or abort (x)?s
Proceed to recovery!
Did goal succeed (s), fail (f), or abort (x)?s
Halt and prepare for recovery!
NIL
```

# Simple Lisp test routine of Sailor Overboard mission logic

```
(defun create-5-mission-phases ()
  (setf phase1 (make-instance 'mission-phase)
        phase2 (make-instance 'mission-phase)
        phase3 (make-instance 'mission-phase)
        phase4 (make-instance 'mission-phase)
        phase5 (make-instance 'mission-phase)))

(defun create-2019-SOB-mission-orders ()
  (setf 2019-SOB-mission-orders (create-5-mission-phases))
  (initialize-2019-SOB-mission-phases))

(defclass DAG-find-all-paths-agent ()
  ((current-search-phase :accessor current-search-phase :initform 'phase1)
   (successor-list :accessor successor-list :initform nil)
   (successor-list-index :accessor successor-list-index :initform 0)
   (all-paths-to-frontier :accessor all-paths-to-frontier :initform nil)
   (all-paths-to-goal :accessor all-paths-to-goal :initform nil)
   (new-paths-list-length :accessor new-paths-list-length :initform nil)
   (new-paths-list :accessor new-paths-list :initform nil)
   (new-path-segments-list :accessor new-path-segments-list :initform nil)))

;DAG means "Directed Acyclic Graph".

(defun update-successor-list-index (index modulus) (mod (1+ index) modulus))

(defmethod initialize-phase ((phase mission-phase) command successor-list)
  (setf (command phase) command
        (successor-list phase) successor-list))

;The system function "eval", as used below, obtains pointer (address) for a global object.

(defmethod get-new-path-segment ((agent DAG-find-all-paths-agent))
  (let* ((phase (current-search-phase agent))
         (command (command (eval phase)))
         (successor-list (successor-list (eval phase)))
         (index (successor-list-index agent))
         (outcome (first (nth index successor-list)))
         (next-phase (second (nth index successor-list)))
         (path-segment (list phase command outcome next-phase)))
    (setf (successor-list-index agent) (update-successor-list-index index 3))
    (push path-segment (new-path-segments-list agent))))
```

```
(defmethod advance-execution-tree-frontier ((agent DAG-find-all-paths-agent))
  (extend-all-paths-to-frontier agent)
  (store-all-new-paths agent)
  (setf (current-search-phase agent) (flf (all-paths-to-frontier agent))))

(defmethod store-all-new-paths ((agent DAG-find-all-paths-agent))
  (dotimes (i (new-paths-list-length agent)) (store-new-path agent)))

(defun create-agent1 ()
  (setf agent1 (make-instance 'DAG-find-all-paths-agent)))

(defun initialize-2019-SOB-mission-phases ()
  (initialize-phase phase1 "Deploy!"
                    '(("Success." phase2) ("Failed." phase5) ("Exception." phase5)))
  (initialize-phase phase2 "Rendezvous with Sailor!"
;;;                 '(("Success." phase4) ("Failed." phase3) ("Exception." phase4))) ;
                    '(("Success." phase4) ("Failed." phase3) ("Exception." phase5))) ;
  (initialize-phase phase3 "Search for Sailor!"
                    '(("Success." phase4) ("Failed." phase5) ("Exception." phase5)))
  (initialize-phase phase4 "Track Sailor Afloat Until Safe!"
                    '(("Success." phase5) ("Failed." phase5) ("Exception." phase5)))
  (initialize-phase phase5 "Proceed to recovery!"
                    '(("Success." "Mission complete.") ("Failed." "Mission aborted.")
                      ("Exception." "Mission aborted."))))

(defun start ()
  (create-2019-SOB-mission-orders)
  (create-agent1)
  (start-mission-execution-tree agent1))

(defun connect (list1 list2)
  (if (equal (first (last list1)) (first list2))
      (append list1 (rest list2))
      list1))

(defun flf (list) (first (last (first list))))

(defun fl (list) (first (last list)))

(defun find-all-paths-to-goal ()
  (start)
```

# Running all possible choice sequences of Sailor Overboard mission logic

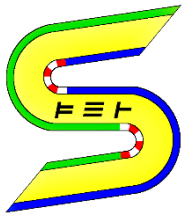
```
Here is initial output that has an error, i.e. Phase 2 definition having exception incorrectly defined as phase4.

;    C:\x-nps-gitlab\NetworkOptionalWarfare\ethicalcontrol\testing\lisp\SailorOverboard\SailorOverboardMissionExhaustiveEvaluationOutcomeSequences.fasl
CG-USER(1): (run)

((PHASE1 "Deploy!" "Success." PHASE2 "Rendezvous with Sailor!" "Failed." PHASE3 "Search for Sailor!" "Success." PHASE4
  "Track Sailor Afloat Until Safe!" "Exception." PHASE5)
 (PHASE1 "Deploy!" "Success." PHASE2 "Rendezvous with Sailor!" "Failed." PHASE3 "Search for Sailor!" "Success." PHASE4
  "Track Sailor Afloat Until Safe!" "Failed." PHASE5)
 (PHASE1 "Deploy!" "Success." PHASE2 "Rendezvous with Sailor!" "Failed." PHASE3 "Search for Sailor!" "Success." PHASE4
  "Track Sailor Afloat Until Safe!" "Success." PHASE5)
 (PHASE1 "Deploy!" "Success." PHASE2 "Rendezvous with Sailor!" "Exception." PHASE4 "Track Sailor Afloat Until Safe!" "Exception."
  PHASE5)
 (PHASE1 "Deploy!" "Success." PHASE2 "Rendezvous with Sailor!" "Exception." PHASE4 "Track Sailor Afloat Until Safe!" "Failed."
  PHASE5)
 (PHASE1 "Deploy!" "Success." PHASE2 "Rendezvous with Sailor!" "Exception." PHASE4 "Track Sailor Afloat Until Safe!" "Success."
  PHASE5)
 (PHASE1 "Deploy!" "Success." PHASE2 "Rendezvous with Sailor!" "Failed." PHASE3 "Search for Sailor!" "Exception." PHASE5)
 (PHASE1 "Deploy!" "Success." PHASE2 "Rendezvous with Sailor!" "Failed." PHASE3 "Search for Sailor!" "Failed." PHASE5)
 (PHASE1 "Deploy!" "Success." PHASE2 "Rendezvous with Sailor!" "Success." PHASE4 "Track Sailor Afloat Until Safe!" "Exception."
  PHASE5)
 (PHASE1 "Deploy!" "Success." PHASE2 "Rendezvous with Sailor!" "Success." PHASE4 "Track Sailor Afloat Until Safe!" "Failed."
  PHASE5)
 (PHASE1 "Deploy!" "Success." PHASE2 "Rendezvous with Sailor!" "Success." PHASE4 "Track Sailor Afloat Until Safe!" "Success."
  PHASE5)
 (PHASE1 "Deploy!" "Failed." PHASE5) (PHASE1 "Deploy!" "Exception." PHASE5))

==================================

Here is modified output that has Phase 2 definition having exception correctly defined as phase5.

;    C:\x-nps-gitlab\NetworkOptionalWarfare\ethicalcontrol\testing\lisp\SailorOverboard\SailorOverboardMissionExhaustiveEvaluationOutcomeSequences.fasl
CG-USER(1): (run)

((PHASE1 "Deploy!" "Success." PHASE2 "Rendezvous with Sailor!" "Failed." PHASE3 "Search for Sailor!" "Success." PHASE4
  "Track Sailor Afloat Until Safe!" "Exception." PHASE5)
 (PHASE1 "Deploy!" "Success." PHASE2 "Rendezvous with Sailor!" "Failed." PHASE3 "Search for Sailor!" "Success." PHASE4
  "Track Sailor Afloat Until Safe!" "Failed." PHASE5)
 (PHASE1 "Deploy!" "Success." PHASE2 "Rendezvous with Sailor!" "Failed." PHASE3 "Search for Sailor!" "Success." PHASE4
  "Track Sailor Afloat Until Safe!" "Success." PHASE5)
 (PHASE1 "Deploy!" "Success." PHASE2 "Rendezvous with Sailor!" "Failed." PHASE3 "Search for Sailor!" "Exception." PHASE5)
```

Autogenerated mission excerpt: LifeboatTracking in Lisp

All existing AVCL missions are converted to Lisp and tested successfully

```
;;;;;;;;;;;;;;;;;;;;;;Begin LifeboatTracking.xml Mission Orders;;;;;;;;;;;;;;;;;;;;;;;;;

  (defvar phaseLBT1.0 (make-instance 'mission-phase)) ; initial goal, by default
  (defvar phaseLBT2.0 (make-instance 'mission-phase))
  (defvar phaseLBT3.0 (make-instance 'mission-phase))
  (defvar phaseLBT3.1 (make-instance 'mission-phase))
  (defvar phaseLBT3.2 (make-instance 'mission-phase))
  (defvar phaseLBT4.0 (make-instance 'mission-phase))
  (defvar phaseLBT4.1 (make-instance 'mission-phase))
  (defvar phaseLBT4.2 (make-instance 'mission-phase))
  (defvar phaseLBT4.3 (make-instance 'mission-phase))
  (defvar phaseLBT5.0 (make-instance 'mission-phase))
  (defvar phaseLBT6.0 (make-instance 'mission-phase))
  (defvar phaseLBT6.1 (make-instance 'mission-phase))
  (defvar phaseLBT6.2 (make-instance 'mission-phase))
  (defvar phaseLBT7.0 (make-instance 'mission-phase))
  (defvar phaseLBT99.0 (make-instance 'mission-phase))
  (defvar phaseLBT99.1 (make-instance 'mission-phase))
  (defvar phaseLBT99.2 (make-instance 'mission-phase))
  (defvar phaseLBT99.3 (make-instance 'mission-phase))

(defun initialize-mission ()
  (setf terminal-phase-list '(phaseLBT99.1 phaseLBT99.2 phaseLBT99.3))

  (initialize-phase phaseLBT1.0 "Deploy, Launch - Commit to robot support"
                    '(("Success." phaseLBT2.0) ("Failed." phaseLBT99.0) ("Exception." phaseLBT99.0)))

  (initialize-phase phaseLBT2.0 "Transit to search area - Proceed to estimated position"
                    '(("Success." phaseLBT3.0) ("Failed." phaseLBT99.0) ("Exception." phaseLBT99.0)))

  (initialize-phase phaseLBT3.0 "Locate Lifeboat - Follow best search pattern"
                    '(("Success." phaseLBT3.1) ("Failed." phaseLBT2.0) ("Exception." phaseLBT99.0)))

  (initialize-phase phaseLBT3.1 "Report position - Alerts updated"
                    '(("Success." phaseLBT3.2) ("Failed." phaseLBT4.0) ("Exception." phaseLBT99.0)))

  (initialize-phase phaseLBT3.2 "Mark with Beacon - Monitor wind effects and ocean current"
                    '(("Success." phaseLBT4.0) ("Failed." phaseLBT4.0) ("Exception." phaseLBT99.0)))

  (initialize-phase phaseLBT4.0 "Track Lifeboat - Monitor and communicate" nil)

  (initialize-phase phaseLBT4.1 "Maintain proximity - Overhead or afloat nearby" nil)

  (initialize-phase phaseLBT4.2 "Periodic reports - Popup or float to report, also recharge" nil)

  (initialize-phase phaseLBT4.3 "Continue - Repeat until conditions change"
                    '(("Success." phaseLBT5.0) ("Failed." phaseLBT7.0) ("Exception." phaseLBT99.0)))
```

# Prolog Programming for AVCL Mission Logic



- ANSI Prolog is a logic programming language associated with AI research and computational linguistics.
- https://gitlab.nps.edu/Savage/EthicalControl/tree/master/missions/prolog holds automatic conversions of all project AVCL missions.
- AvclToProlog.xslt stylesheet reads AVCL XML to produce Prolog source code.
  - Initial section is Mission Execution Engine (MEE) goal-traversal algorithm.
  - Next section defines, allows operator testing of mission-goal decision tree logic.
- Mission Execution Engine holds common code for Sailor Overboard mission accompanied by console log of example operator test sequences.
- ***Future work:*** test routine showing how to build all possible choice sequences.
- ***Future work:*** automate exhaustive testing of all choices in all missions as initial exemplar for exhaustive verification of mission logic.

# Mission Execution Engine in Prolog, excerpted

```
;After this code has been entered into Allegro Editor window and compiled, it is necessary to
;similarly enter and compile a mission. File "2019 Man Overboard Mission Orders" stored in
;Allegro Editor files provides an example. Entering "(run)" to Lisp prompt then runs both files together.

;Description: this is the RBM Strategic Level task sequencing algorithm that is the basis
; of the Mission Execution Engine (MEEE). See Figure 2 in the IEEE JOE paper (TODO link this).

;Start Prolog.
(require :prolog) (shadowing-import '(prolog:==)) (use-package :prolog)

;Facts
(<-- (current_phase 0)) ;Starting phase.
(<-- (current_phase_outcome s))

;Mission execution rule set
(<-- (execute_mission) (initialize_mission) (repeat) (execute_current_phase) (done) !)
(<-- (initialize_mission) (abolish current_phase 1) (asserta ((current_phase 1))))
(<-- (execute_current_phase) (current_phase ?x) (execute_phase ?x) !)
(<-- (done) (current_phase 'mission_complete))
(<- (done) (current_phase 'mission_abort))

;Human external agent communication functions

(<-- (negative nil)) (<- (negative n))
(<-- (affirmative ?x) (not (negative ?x)))
(<-- (report ?C) (princ ?C) (princ ".") (nl))
(<-- (command ?C) (princ ?C) (princ "!") (nl))
(<-- (ask ?Q ?A) (princ ?Q) (princ "?") (read ?A))
(<-- (ask_outcome ?A) (ask "Did goal succeed (s), fail (f), or abort (x)" ?A))

;Utility functions
(<-- (change_phase ?old ?new) (retract ((current_phase ?old))) (asserta ((current_phase ?new))))
(<-- (update_outcome) (ask_outcome ?A) (abolish current_phase_outcome 1) (asserta ((current_phase_outcome ?A))))


;Test functions (illustrate format for calling predicates from Lisp)
(defun run () (?- (execute_mission)))
(defun update () (?- (update_outcome)))
(defun mission-phase () (?- (current_phase ?X)))
(defun outcome () (?- (current_phase_outcome ?X)))
```

# Sailor Overboard mission in Prolog, excerpted

SailorOverboardUavMissionOrdersReducedState3.pl 1.93 KB

```
;This code written in ANSI Common Lisp, Allegro 10.1 enhancement, from Franz, Inc., by
;Prof. Robert B. McGhee (robertbmcghee@gmail.com) at the Naval Postgraduate School,
;Monterey, CA 93943. Date of latest update: 7 October 2019.

(require :prolog) (shadowing-import '(prolog:==)) (use-package :prolog) ;Start Prolog.

;Mission specification

(<-- (execute_phase 1) (command "Phase 1 report: Deploy") (update_outcome)
                       (current_phase_outcome s) (change_phase 1 2))
(<- (execute_phase 1) (current_phase_outcome f) (change_phase 1 5))
(<- (execute_phase 1) (current_phase_outcome x) (change_phase 1 5))

(<- (execute_phase 2) (command "Phase 2 report: Rendezvous with Sailor") (update_outcome)
                       (current_phase_outcome s) (change_phase 2 4))
(<- (execute_phase 2) (current_phase_outcome f) (change_phase 2 3))
(<- (execute_phase 2) (current_phase_outcome x) (change_phase 2 4))

(<- (execute_phase 3) (command "Phase 3 report: Search for Sailor") (update_outcome)
                       (current_phase_outcome s) (change_phase 3 4))
(<- (execute_phase 3) (current_phase_outcome f) (change_phase 3 5))
(<- (execute_phase 3) (current_phase_outcome x) (change_phase 3 5))

(<- (execute_phase 4) (command "Phase 4 report: Track Sailor Afloat Until Safe") (update_outcome)
                       (current_phase_outcome s) (change_phase 4 5))
(<- (execute_phase 4) (current_phase_outcome f) (change_phase 4 5))
(<- (execute_phase 4) (current_phase_outcome x) (change_phase 4 5))

(<- (execute_phase 5) (command "Phase 5 report: Proceed to recovery") (update_outcome)
                       (current_phase_outcome s) (change_phase 5 'mission_complete) (report "Phase 6 report: Vehicle recovered"))
(<- (execute_phase 5) (current_phase_outcome f) (change_phase 5 'mission_abort)    (report "Phase 7 report: Vehicle lost"))
(<- (execute_phase 5) (current_phase_outcome x) (change_phase 5 'mission_abort)    (report "Phase 8 report: Awaiting orders"))

```

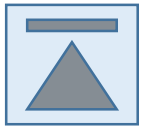
# Sailor overboard example operator test sequences in Prolog

```
CG-USER(1): (run)
Phase 1 report: Deploy!
Did goal succeed (s), fail (f), or abort (x)?s
Phase 2 report: Rendevous with Sailor!
Did goal succeed (s), fail (f), or abort (x)?s
Phase 4 report: Track Sailor Afloat Until Safe!
Did goal succeed (s), fail (f), or abort (x)?s
Phase 5 report: Proceed to recovery!
Did goal succeed (s), fail (f), or abort (x)?s
Phase 6 report: Vehicle recovered.
Yes
No.
CG-USER(2): (run)
CG-USER(2): (run)
Phase 1 report: Deploy!
Did goal succeed (s), fail (f), or abort (x)?2
Phase 1 report: Deploy!
Did goal succeed (s), fail (f), or abort (x)?s
Phase 2 report: Rendevous with Sailor!
Did goal succeed (s), fail (f), or abort (x)?f
Phase 3 report: Search for Sailor!
Did goal succeed (s), fail (f), or abort (x)?s
Phase 4 report: Track Sailor Afloat Until Safe!
Did goal succeed (s), fail (f), or abort (x)?s
Phase 5 report: Proceed to recovery!
Did goal succeed (s), fail (f), or abort (x)?s
Phase 6 report: Vehicle recovered.
Yes
No.
CG-USER(3): (run)
Phase 1 report: Deploy!
Did goal succeed (s), fail (f), or abort (x)?s
Phase 2 report: Rendevous with Sailor!
Did goal succeed (s), fail (f), or abort (x)?s
Phase 4 report: Track Sailor Afloat Until Safe!
Did goal succeed (s), fail (f), or abort (x)?s
Phase 5 report: Proceed to recovery!
Did goal succeed (s), fail (f), or abort (x)?x
Phase 8 report: Awaiting orders.
Yes
```

# “Your Programming Language Here” example

- Robot software tends to be highly specialized, idiosyncratic, evolving.
- Despite rapid change, programmers want library reuse + flexible repeatability.
- Robot Operating System (ROS) is steadily gaining usage with hardened ROS-Military (ROS-M) version available.


- Creating a family of AVCL parsers that can read XML-based missions opens the door to interoperability and shared support.
- Automating the production of these parsers from the AVCL schema ensures that all systems can have rigorous and consistent support.
- Individual robot logic may vary widely in implementation details, but core semantics of AVCL goals and nomenclature remain well defined.
- ***Future work:*** establishing qualification testing for software or hardware running “in the loop” can confirm that individual robots operate safely.



# Simulation and Visualization

Rehearsal, real-time runs and replay are possible using AVCL constructs across multiple representations and programming languages.

# Path towards achieving interoperability

Robotic systems tend to be complex codebases with implementations that require strict, idiosyncratic, language-specific programming logic.

- In general, system designers can say “here are the requirements”
- In general, programmers can say “here’s how we wrote that code”
- … but these are not the same, often not even sharing same terms of reference!

Traceable predictability of software logic is difficult, not portable across systems.

- … but confirming code capabilities is testable and repeatable across systems.

**Key point: strict validation of mission syntax, semantics are both possible!**

- Patterns of implementation then become demonstrable in different systems,
- Human confirmation of mission definitions remains central throughout.

# Multiple implementations for scalability

Given the broad diversity of robotic software and hardware systems under development, no single reference codebase is either possible or desirable.

- Nevertheless systems can easily parse and utilize well-defined data (orders).
- Focusing on formal mission definition for both humans and systems provides a testable middle ground that each can use effectively.

Implementing and evaluating using multiple software implementations also provides strong evidence that design capabilities all work as planned.

- … in turn producing corresponding work lists of needed improvements, to help both mission-design clarity and software-implementation correctness.

This project multiple programming paths in tandem, in order to demonstrate that multiple kinds of unmanned systems can adopt it on their own terms.

# Data representation languages of interest

***Extensible Markup Language (XML)*** provides declarative basis for customized, strictly defined data definitions of interest.

- ***Autonomous Vehicle Command Language (AVCL)*** is defined using XML schema for strict validation of syntax, particularly hierarchical data modeling relationships, strictly defined enumerations and legal values for numerical values.

Semantic Web languages add ability to perform queries and reasoning.

- ***Turtle*** (Terse Triple Language) deconstructs AVCL into primitives.
- ***RDF/OWL*** expresses logical conditions and constraints of Mission Execution Ontology (MEO) corresponding to AVCL.
- ***SPARQL*** query language enables further inspection and verification of logical relationships.

Not used in this work: JavaScript Object Notation (JSON) syntax is also available.

- Common use, potential to match XML expressiveness, lesser validation maturity.

# Programming languages of interest

***Java*** is primary language used for exemplar robot controllers.

- Implemented in AUV Workbench and various build processes.
- Not currently used but similar expressiveness: C++ and C# (C sharp).

***Lisp*** and ***Prolog*** are also used for testing AVCL mission-tasking logic.

- Both are well suited for AI application design.
- Earlier thesis work similarly applied CLIPS expert-system rule bases.

Of note: AVCL missions can be used to autogenerate exemplar source code in various alternative languages by creating XLST conversion stylesheets.

- Thus showing interoperability path to all manner of robotic systems.

# Presentation languages of interest

***HTML5*** for mission reports as portable, archival Web pages

- Also ***Cascading Style Sheets (CSS)*** for consistency, quality

***KML*** for annotated, animated placemarks on maps and globes

- OpenStreetMap/OpenSeaMap, Google maps, etc.

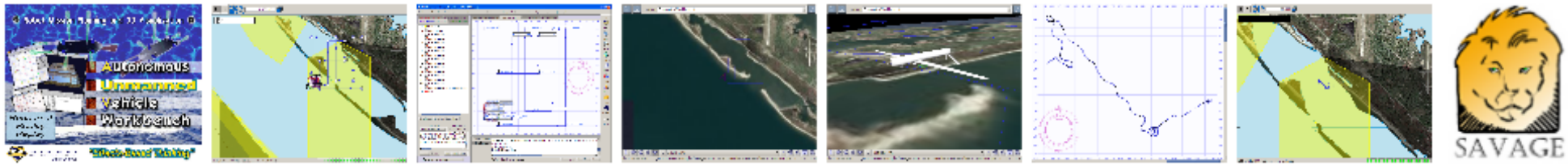
***Extensible 3D (X3D) Graphics*** for 3D visualization and animation

- Future publication expected in SPIDERS3D virtual environments (VE)

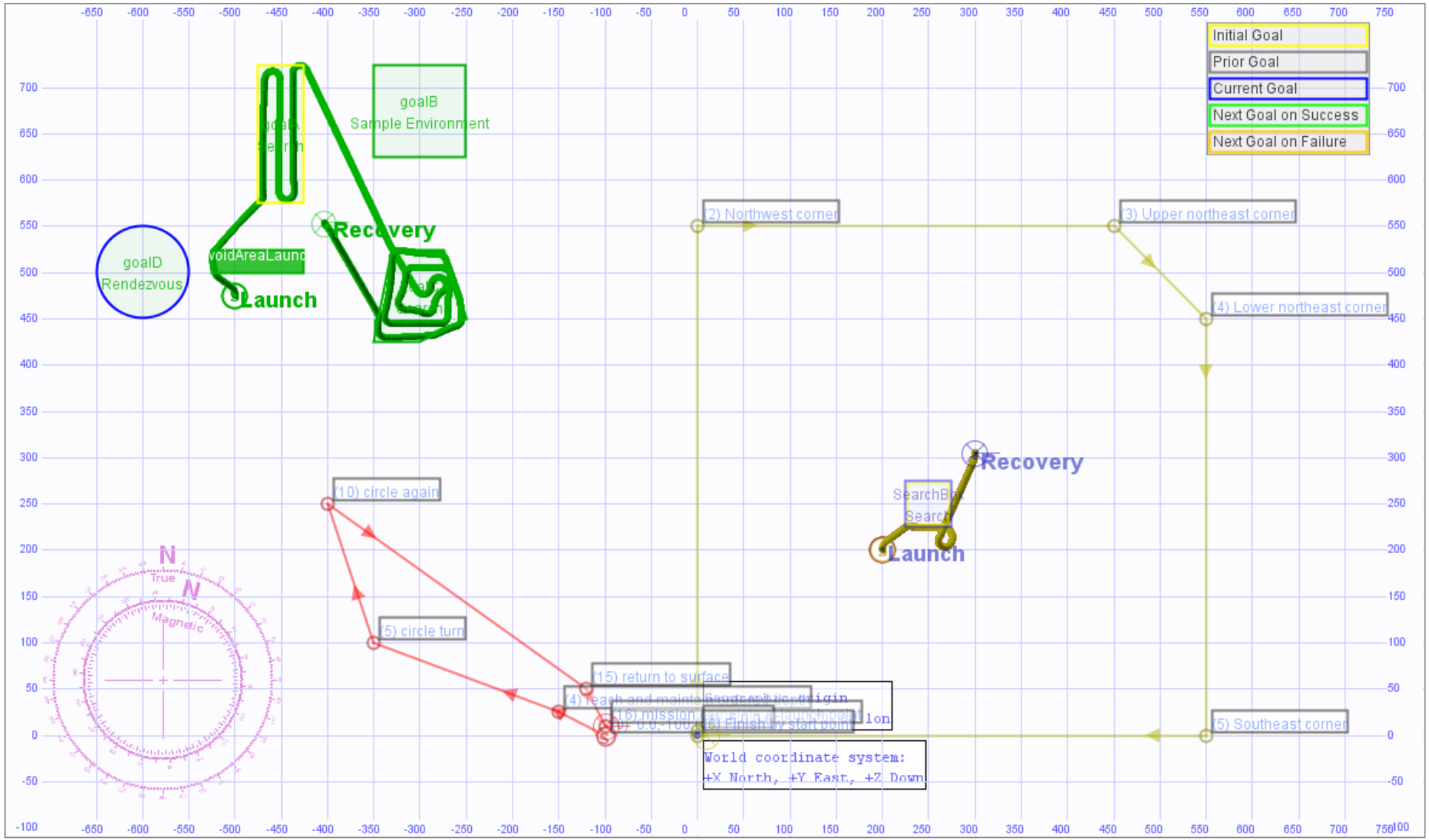
# AUV Workbench

Example open-source implementation

- Autonomous Unmanned Vehicle (AUV) Workbench supports underwater, surface and air vehicles
  - Rehearsal of physics-based mission response
  - Real-time task-level control of robot missions, and
  - Replay of recorded results
  - Industry-friendly open-source license, Sourceforge
  - Basis:  RBM 3-level architecture, AVCL commands
- Used to rehearse strategic-level agenda missions
  - https://savage.nps.edu/AuvWorkbench



# 4 earlier example missions, UUV and USV



Coder speak!

# AVCL version 3 improvements planned

Refactoring of agenda mission definitions is being closely considered.

- Overly complex, multiple data-model simplifications in progress.
- Consolidating distinctions between UAVs, USVs, UUVs, UGVs.
- Ensuring convertibility AVCL3 to AVCL2 for backwards compatibility.

Refactoring corresponding software implementation in Java:

- Prior AVCL2 implementation uses Java Advanced XML Binding (JAXB) which is exceedingly complex and difficult to maintain.
- Planning to create Plain Old Java Object (POJO) mapping for AVCL3, will be much easier to maintain and utilize programmatically.

Review, confirm mappings for mission scripts and telemetry logs are OK.

Coder speak!

# AUV Workbench improvements planned

TODO

- Upgrade legacy codebase and libraries to latest versions of Java
- Upgrade AUV Workbench to support for AVCL version 3 missions
  - Near term will be quite simple, AVCL3 -> AVCL2 conversion
- Display conduct of canonical missions developed in this project
- Manually record videos of mission demonstrations and playbacks
- Update mission production of HTML reports, KML maps, X3D graphics
- Support project report and multiple peer-review presentations

# AUV Workbench Demo Mode 1

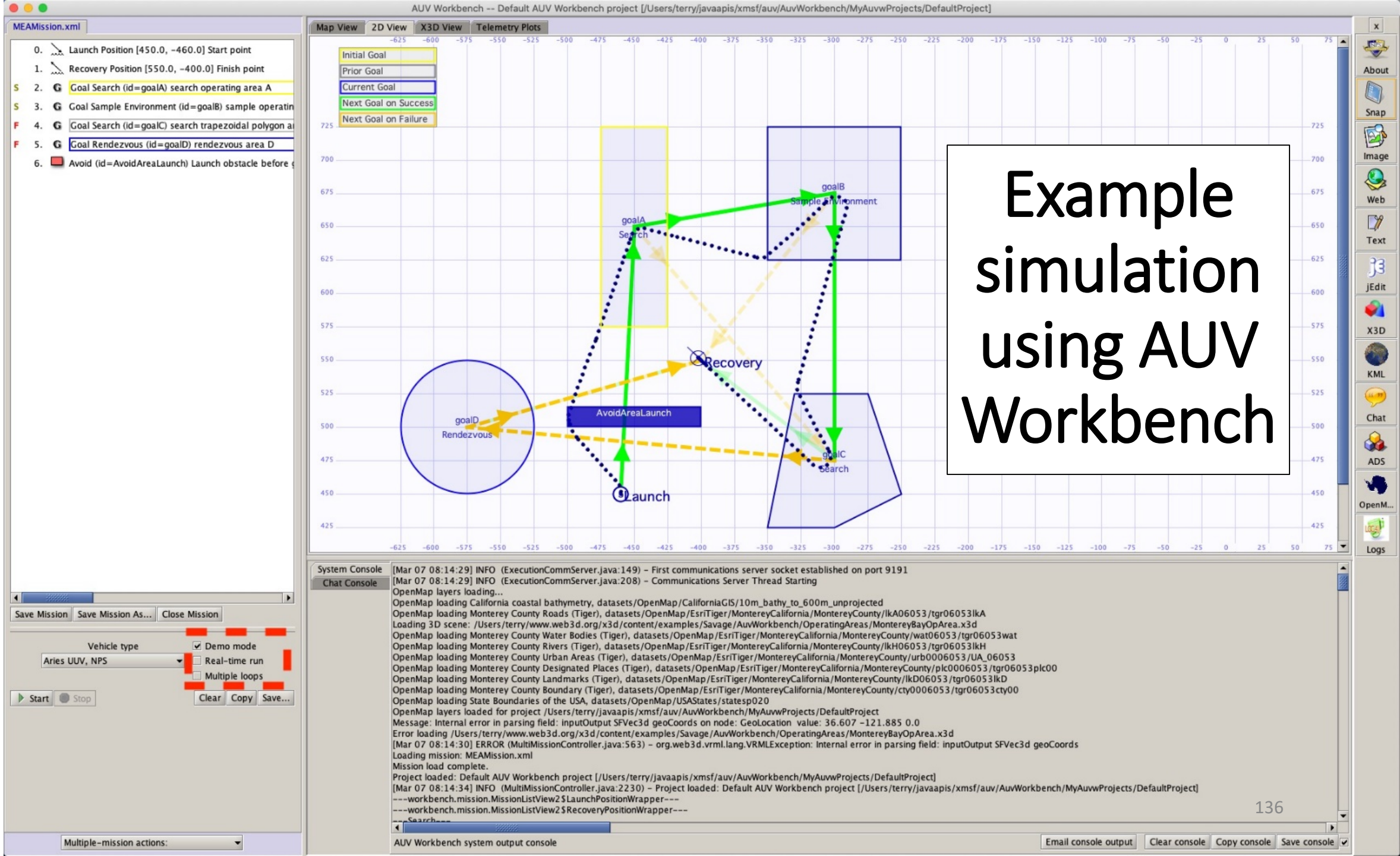
**How can we rapidly test missions and visualize their progress?**

- Demo mode resets each goal's Search type to EXPANDINGSQUARE
- with slight change in criteria: reaching the center of the goal is sufficient to declare goal success, skipping the actual search.
- Demo mode thus lets a mission proceed as written in rapid fashion to illustrate decision-tree logic checking.

**Demo: 2D map display on next slide shows**

- Green arrows are nextOnSuccess,
- Orange arrows are nextOnFailure,
- Bolded arrows indicate which of choices was taken during mission conduct.
- _S_uccess _F_ailure letters (upper-left panel) also report results of each Goal's conduct.

AUV Workbench Demo Mode 2




Example simulation using AUV Workbench

# AUV Workbench Demo Mode 3

Repeatable Ethical Mission Testing

**Goal-by-goal narrative of completed mission conduct**

[0] Launch from launch position.
[6] Avoid first obstacle, next proceed to goalA boundary and then center.
[2] Once at center: success goalA (Search), next proceed to goalB boundary and then center.
[3] Once at center: success goalB (Sample Environment), next proceed to goalC boundary and then center.
[4] Once at center: failure goalC (Search), next proceed to goalD boundary and then center.
[5] Immediate failure without transit to goalD (Rendezvous) since capability is declared as not supported by robot controller.
[1] Proceed directly to nextOnSuccess Recovery Position, once there finished in terminal state.

**Methodology summary**

- Read an AVCL mission document
- Test decision logic using a virtual Software In The Loop (SITL) robot controller,
- Run within an evaluation virtual environment (AUV Workbench).
- Human evaluators then evaluate proper execution of the AVCL ethical mission, specifically assessing that robot-controller SITL within the AUV Workbench virtual environment.

# Qualification of unmanned systems: extend Verification Validation Accreditation (V V&A)

Proposed Future Work

SISO V V&A Study Group

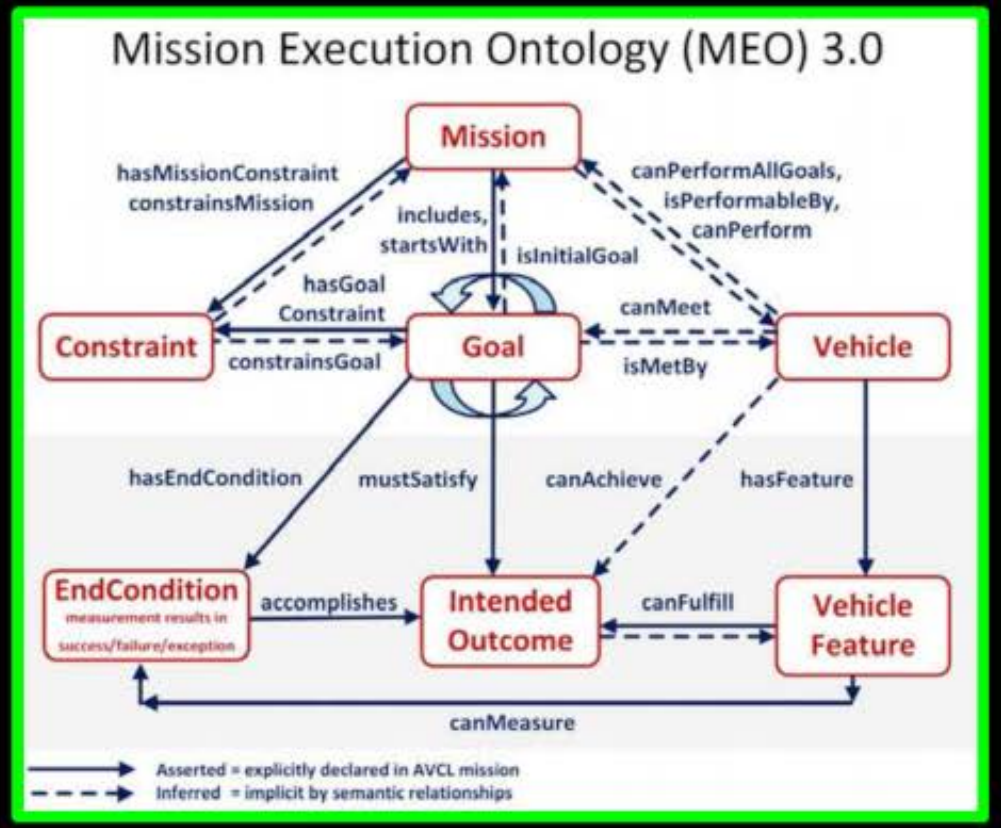
How to test and certify robots will follow both orders and constraints? Humans confirm understanding and trust through qualification processes.

- Design, construct “qualification card” for testing unmanned systems…
- Comprehensive virtual environment, hardware/software in the loop.
- Carefully crafted scenario testing of key requirements and capabilities.
- Anti-pattern tests to provoke and confirm constraints are not violated.
- Record all unit-test decision trees, decision-branching traces, and results as a certification record for each hardware/software version of robots.
- Visualize realistic rehearsal, real-time and replay of robot operations repeatably using shared Web-based SPIDERS3D virtual environment.
- Humans assess mission logs and scenario outcomes for after-action analysis, lessons learned, and continuous improvement via suite of unit tests.

# Python Mission Evaluation, Exhaustive Test Analysis and Connecting to AI-based Opponent Systems

- Implementation Demonstration Video, Jon Cefalu, neurobinder.com 16 December 2020 (59:41)
- ( (6:00) Jon shows new capabilities include a Python implementation for exercising decision logic in Autonomous Vehicle Command Language (AVCL) mission. Additional features include the ability to exhaustively test mission variations, checking functional mission correctness and detecting decision loops.
- (20:00) Mission testing considerations appear to provide an initial basis for evaluating human-machine mission logic and code coverage, building stepping stones towards model-based testing as well as verification and validation,
- (27:30) Jon also demonstrated cross-connecting the Pirate mission to AI Dungeon (Wikipedia), a text-based adventure game engine based on the Generative Pre-trained Transformer 3 (GPT-3) language-prediction model by OpenAI that uses deep learning to produce human-like text responses. Despite only superficial configuration, scenario exploration became possible using interesting (and occasionally outlandish) text responses from the tool.
- (48:30) Group discussion on future design considerations for connecting human-machine teaming systems with realistic wargaming systems for sensitivity analysis, massively repeatable testing, and analysis of alternatives.

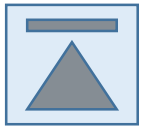
# What if we pit the Rational Behavior Model against a computer game?




VS



12/16/2020
Jon Cefalu



# Data-Centric Security and Trust

Compression, authentication, encryption, composability, blockchain ledger, and asymmetric advantages enable group communication of secure mission orders and responses. This is a Chain of Trust for distributed Command Authority.

# XML Security for Data: **Structure** 1

1. Extensible Markup Language (XML) provides formal structure for data models and information exchange.
   a. "XML is a markup language that defines a set of rules for encoding documents in a format that is both human-readable and machine-readable." – Wikipedia
   b. Declarative and self-describing data structures, not program source code.
   c. Data validation through XML Schema includes strong typing of values and correct parent-child hierarchical relationships.
   d. Avoids Garbage In Garbage Out (GIGO) pathologies when communicating between multiple systems and across related protocols.
   e. Similarly applicable using JavaScript Object Notation (JSON) and other formats.
   f. Offers complete precision of expressive power when defining human orders and system responses, e.g. via Autonomous Vehicle Command Language (AVCL).

# XML Security for Data: **EXI Compression** 2

2. Efficient XML Interchange (EXI) provides best-possible compression of XML documents, reducing size and speeding up decompression.
   a. Years of work by exceptionally competent working group, proven results.
   b. EXI Recommendations by World Wide Web Consortium (W3C).
   c. Multiple open-source and commercial implementations in Java, C++
   d. Preserves sufficient structure for lossless composition of compressed XML.

Thus even signed and encrypted data documents shown in this work can get best-proven compression for use on limited, disadvantaged and challenged communications links facing deployed Naval forces.

# Network Optional Warfare (NOW): Efficient Messaging

**Navy networks afloat are very different than networks ashore.** Bandwidth is a precious and finite resource, latency can be huge, connectivity can be intermittent, environmental effects dominate, channels are limited in varying ways, and mobile relays are rare. Manned and unmanned naval systems need efficient messaging for networks afloat - but rarely have it. Failing to properly utilize communications capacity directly limits tactical effectiveness.

**Efficient messaging is needed to take maximum advantage of severely constrained data links.** The key to our strategies for achieving efficient messaging is first to use of Extensible Markup Language (XML) for structured data languages, and then use EXI for compressing XML. Since XML provides a flexible and validatable way to define regular data structures for any language, it provides a practical opportunity to compatibly capture and convert all manner of diverse data formats used for military messaging. The economics of Web technologies are undeniable and usually provide industry-wide best practices as well. As a result, this use of open standards is scalable and repeatable, avoiding the "stove pipes" which commonly prevent system-wide interoperability between Navy platforms and coalition partners.

**"Efficiency" means both size and speed.** EXI has demonstrated compaction that *always* meets or beats the most commonly used compression techniques (zip and gzip). Additionally, because EXI decompression goes straight into memory rather than string characters, which then require significant additional parsing, decoding EXI is many times faster than other techniques. This approach also reduces memory requirements and power consumption on small devices. Because Navy tactical traffic is usually highly structured and highly numeric, EXI provides major advantages that might well impact all afloat Navy communications. Alternative bit-centric compression schemes cannot take full advantage of those characteristics.

**"Efficiency" is compatible with Data-Centric Security.** Demonstrated thesis work has shown that digital signature (for authentication) and XML Encryption (privacy and access control) can coexist with efficient compression, when applied in the correct order. Such interoperability for Information Assurance (IA) Is necessary when working with coalition partners, and also for safeguarding data within deployed unmanned systems that are beyond the reach of network-centric security.

# XML Security for Data: **Digital Signature** 3

3. XML Digital Signature (DS) defines XML syntax for digital signatures.
   a. W3C Recommendation, stable since 2013, international adoption.
      - https://www.w3.org/TR/xmldsig-core1
   b. Public-private key pairs for signature/authentication, key distribution is separate.
   c. Applicable to entire documents or to fragments (subsections).
   d. Requires XML Canonicalization of input documents to regularize formatting so that identical documents are uniquely expressed.
   e. Can sign any data resource for identity verification, non-repudiability, confirmation that original information has not been tampered with, etc.
   f. Completely compatible for data handling within trusted networks.
   g. 2019 NPS has adapted open-source Java version of Apache Santuario as utility classes and test suite for further use. Prior examples from years ago still work.
      https://sourceforge.net/p/x3d/code/HEAD/tree/www.web3d.org/x3d/tools/security/XmlSecurityApacheSantuario

# XML Security for Data: **Encryption** 4

4. XML Encryption (XML-Enc) defines how to encrypt XML data.
   a. W3C Recommendation, stable since 2013, international adoption.
      - https://www.w3.org/TR/xmlenc-core1
   b. Public-private (i.e. shared-secret) key pairs, key distribution is separate.
   c. Applicable to entire documents or to fragments (subsections).
   d. Different from Transport Layer Security (TLS) used by http/https for sending encrypted traffic over the Internet.
   e. Some vulnerabilities were reported publicly, but each was performed via exhaustive attacks against server, incrementally analyzing error responses. Not likely or practical mode of attack against unmanned systems.
   f. Completely compatible for data handling within already-trusted networks.

Encrypt Decrypt - Apache NetBeans IDE 11.2

File Edit Navigate View Window Source Refactor Run Debug Profile Team Tools X3D Help

Search (Ctrl+I)

<default config>

Projects - Ethical Control of Unman... | Files | Services

- AUV Workbench [New ]
- Digital Signature
- Encrypt Decrypt [Modified ]
  - src
    - main
      - java
        - org
          - web3d
            - x3d
              - tools
                - security
                  - XmlSecurityApacheSantuario
                    - EncryptDecrypt
                      - EncryptionStAX.java
                      - EncryptionUtils.java
      - resources
    - test
  - target
  - HelloWorld.encrypted.x3d [Ignored ]
  - HelloWorld.x3d
  - debug.log [Ignored ]
  - nb-configuration.xml [Modified ]
  - nbactions.xml
  - plaintext.xml
  - pom.xml
- Ethical Control of Unmanned Systems [maste

Navigator | Xj3D Window

- deploy **deploy-file**
- install **install-file**
- jar **sign**
- jar **sign-verify**
- jar **test-jar**
- project-info-reports **ci-management**
- project-info-reports **dependencies**
- project-info-reports **dependency-convergence**
- project-info-reports **dependency-info**
- project-info-reports **dependency-management**

...t.tx | SailorOverboardMission5State.cl.log.txt | .gitignore | build.xml [Ethical Control of Unmanned Systems] | EncryptionUtils.java

Source | History

```
62  import org.w3c.dom.Document;
63  import org.w3c.dom.Element;
64  import org.w3c.dom.NodeList;
65
66  /**
67   * Some utility methods for encrypting/decrypting documents
68   */
69  public final class EncryptionUtils {
70
71      static {
72          Init.init();
73      }
74
75      private EncryptionUtils() {
76          // complete
77      }
78
79      /**
80       * Encrypt the document using the DOM API of Apache Santuario - XML Security
81       * for Java.It encrypts a list of QNames that it finds in the Document via
82       * XPath. If a wrappingKey is supplied, this is used to encrypt the
83       * encryptingKey + place it in an EncryptedKey structure.
84       *
85       * @param document
86       * @param namesToEncrypt
87       * @param algorithm
```

org.web3d.x3d.tools.security.XmlSecurityApacheSantuario.EncryptDecrypt.EncryptionUtils

Output - Test (Encrypt Decrypt) | Search Results | Versioning Output

```
Results :

Tests run: 1, Failures: 0, Errors: 0, Skipped: 0

------------------------------------------------------------------------
BUILD SUCCESS
------------------------------------------------------------------------
Total time: 5.063 s
Finished at: 2019-12-09T08:25:04-08:00
Final Memory: 9M/44M
------------------------------------------------------------------------
```

Tests (1) finished successfully for project: Encrypt Decrypt

Open Test Results Window

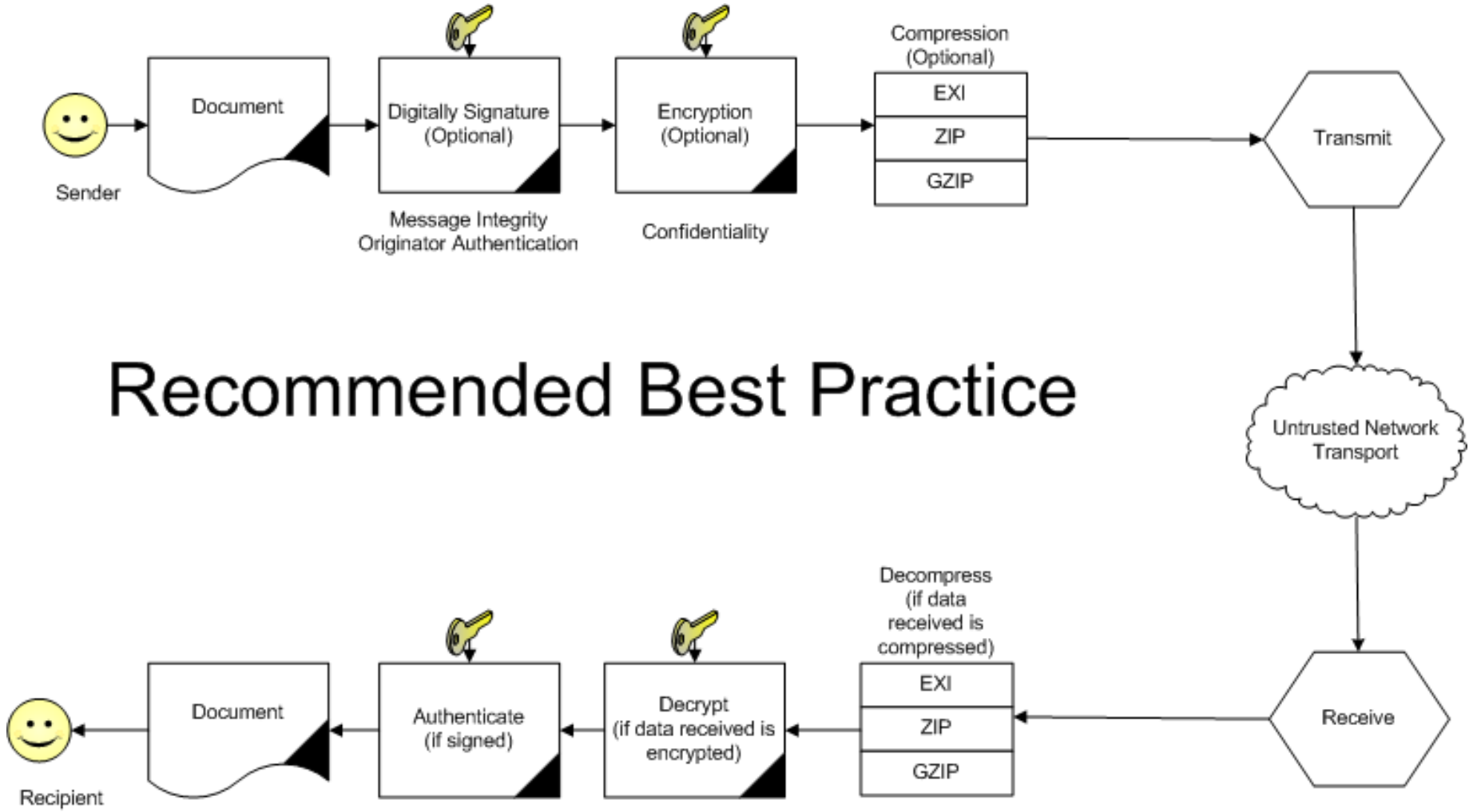


Checking for external changes | Suspended | 80:63 | INS

# XML Security: **Composition** 5

5. EXI Compression, XML Digital Signature and XML Encryption can be composed for applying to data in single files/documents/messages.
   a. Each technology works on data formatted as valid XML.
   b. Multiple NPS theses have examined EXI characteristics in combination with XML Security.
   c. Such composition is partially demonstrated, appears completely feasible.
   d. Williams, Jeffrey S., Document-based message-centric security using XML authentication and encryption for coalition and interagency operations, Masters Thesis, Naval Postgraduate School, Monterey, California, 2009.
   e. Each is usable in concert for data-centric security, compatibly within any secure network or within fixed/mobile data storage of unmanned systems.

# Williams Thesis: Composition of EXI Compression, XML Authentication, and XML Signature




## Recommended Best Practice

# Security Assertion Markup Language (SAML)

- Security Assertion Markup Language (SAML) is an open standard for exchanging authentication and authorization data between parties, in particular, between an identity provider and a service provider.
- SAML is an XML-based markup language for security assertions (statements that service providers use to make access-control decisions).
- With some adaptation work, SAML might be used to formally describe policies and requirements for data-centric security of mission orders.

References


- RDML Danelle Barrett USN, "The data-driven Department of the Navy," *CHIPS*, January-March 2018
- Latest version is SAML 2.0 standardized by OASIS


# Blockchain distributed ledger characteristics

- "A **distributed ledger** is a consensus of replicated, shared, and synchronized digital data geographically spread across multiple sites, countries, or institutions. There is no central administrator or centralized data storage." – Wikipedia
- Design characteristics can be tuned to match system needs and include strict sequencing of ledger entries, nonrepudiability of message indexes, consensus algorithm (proof of work or stake), etc.
- Implementation is often accomplished via a blockchain system.

*Future work:* ships, aircraft and ground systems might maintain a strong distributed ledger of all messages sent and received, reducing risk of spoofing or counterfeit messages compromising unmanned systems.

# Significant protections from hostile takeover… is possible for deployed friendly-force robots

Accountability for actions requires a traceable, provable decision tree.

The following vulnerability "anti-pattern" provides an interesting use case, whereby non-repudiability of mission orders can prevent an opponent from falsely claiming a "rogue robot" or "rogue commander" scenario:

- Opponent captures control of a friendly unmanned system (physically or through cyber attack).
- Opponent has no key, unable to decrypt previously recorded sensor data.
- Opponent disables onboard security interlocks, directs unmanned system to execute hostile act (e.g. attack on friendly or neutral force).
- Post-incident investigation reveals and proves that mission orders were not authenticated or authorized by original friendly commander.
- Block-chain ledger of all issued authenticated orders reveals that no gaps occurred in shipboard records of approved missions.

# Data-Centric Security and Command Authority

Data-centric security that includes authentication of ordered missions for unmanned systems provides a military, legal, ethical and moral basis for non-repudiability and accountability of human commanders.

- Authorized humans remain in charge, accountable for robot actions.
- Collected robot data is encrypted in asymmetric manner, greatly reducing vulnerabilities following any robot capture or compromise.
- Data-centric security can coexist within all levels of network security.

Such reliability provide excellent rationale to link data-centric security to design considerations for Ethical Control, compatibly across all networks.

- Once again, Ethical Control leads to *more-effective warfighting*.

# Trust

## Trusted Mission Orders

- Formal shared meaning between robots and human commanders
- Controlled vocabulary of terms with well-defined conditions, outcomes
- Syntax validation, well-formed data
- Numerical validation, in bounds
- Semantic confirmation of tactical prerequisites, coordination steps
- No logical contradictions present

## Trusted Mission Execution

- Portable tasking across diverse unmanned systems, C4I networks
- Data-centric encryption for transmission across any network
- Digital-signature authentication that confirms command identity
- Blockchain ledger authoritatively confirms completeness, no gaps
- Testable in simulation, eventually formalized as robot qualification

# Blockchain ledger for distributed accountability

Given a trusted chain of message exchange among participating human commands and distributed systems, there are additional vulnerabilities that still need to be considered. Blockchain technology is relevant.

Obvious tactical accountability issue is missing gaps or jammed messages

- Failure to receive even one message (perhaps requiring human permissions) can invalidate any subsequent actions, thereby losing control of lethal force.

Extrapolating further needs: after-action analysis, investigation, improvement.

- Having a ledger of all received/sent messages can provide accountability and verifiable chain of trust for authoritative reconstruction and progress.

Important future work: custom blockchain providing assurances that scale among diverse participants and over time, without needing a central hub.

# Related work: Zero Trust Architecture (ZTA)

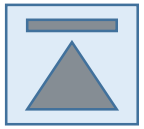
*Zero Trust Architecture*, Scott Rose (NIST), Oliver Borchert (NIST), Stu Mitchell (Stu2Labs), Sean Connelly (DHS), 2nd Draft, NIST SP 800-207, February 2020. zerotrust-arch@nist.gov

- "Zero trust refers to an evolving set of network security paradigms that narrows defenses from wide network perimeters to individual resources. Its focus on protecting resources rather than network segments is a response to enterprise trends that include remote users and cloud-based assets that are not located within an enterprise-owned network boundary."
- https://csrc.nist.gov/publications/detail/sp/800-207/draft

Data-centric security seems like logical conclusion of such an approach.

# Next steps: implement, evaluate, deploy tests

- Data-centric security can provide guarantees of command authority over the application of lethal lifesaving force by unmanned systems.
- Open standards and implementations exist for each component: compression, signature, encryption, assertion metadata, etc.
  - Alternative technologies are also available.
- Composition testing with robots during field experimentation (FX) can extend laboratory results with real-world experience, risk analysis and red-team testing.
- Further work recommended.



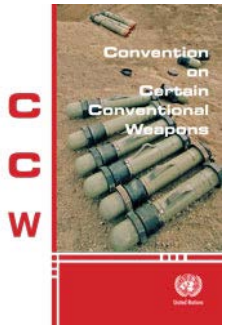
# Related Ethical Activities

Ethics of lethality and unmanned systems is an active area of work. The following synopses are distilled from each respective resource.

# Campaign to Stop Killer Robots


- Fully autonomous weapons are fundamental change to nature of war.
- Problem aspects include: lethal force without human intervention, destabilizing robotics arms race, lower threshold to decide on war, lack of human judgement for proportionality, lack of accountability or culpability, further use against populations by oppressive regimes.
- Their proposed solution: development, production and use of fully autonomous weapons must be banned. Retain meaningful human control through laws and treaty, international commitment to ban by countries, pledge by technology companies/organizations/individuals to never contribute to development of fully autonomous weapons.
- https://www.stopkillerrobots.org includes notable endorsements.

# Convention on Certain Conventional Weapons (CCW)

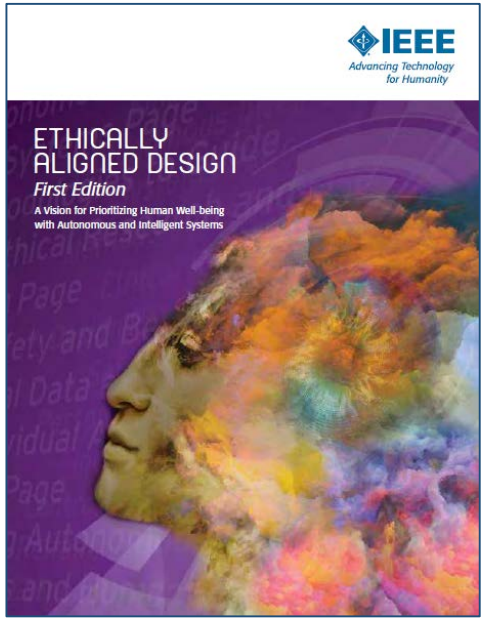



- Sometimes referred to as the *Inhumane Weapons Convention*. Some parties argue that any unmanned systems with lethal force ought to restricted in a similar fashion.
- "Seeking to restrict or outlaw specific types of weapons used in armed conflict, 51 states negotiated the Convention on Certain Conventional Weapons (CCW) in 1980. The agreement is formally known as the Convention on Prohibitions or Restrictions on the Use of Certain Conventional Weapons Which May Be Deemed to Be Excessively Injurious or to Have Indiscriminate Effects. It is also sometimes referred to as the Inhumane Weapons Convention. The convention aims to protect military troops from inhumane injuries and prevent noncombatants from accidentally being wounded or killed by certain types of arms. When it entered into force in December 1983, the treaty applied to incendiary weapons, mines and booby-traps, and weapons designed to injure through very small fragments. Since then, treaty states-parties—numbering 120 total as of August 2017—have added provisions to ban blinding laser weapons and address lingering dangers posed by unexploded munitions leftover after combat ends."
- https://www.un.org/disarmament/publications/more/ccw
- https://www.armscontrol.org/factsheets/CCW
- https://en.wikipedia.org/wiki/Convention_on_Certain_Conventional_Weapons

# IEEE Global Initiative on Ethics of Autonomous and Intelligent Systems (A/IS)

## Ethically Aligned Design: A Vision for Prioritizing Human Well-being with Autonomous and Intelligent Systems, First Edition (EAD1e)

- "The most comprehensive, crowd-sourced global treatise regarding the Ethics of Autonomous and Intelligent Systems available today."
- "It is time to move "From Principles to Practice" in society regarding the governance of emerging autonomous and intelligent systems. The implementation of ethical principles must be validated by dependable applications of A/IS in practice.
- https://ethicsinaction.ieee.org



# IEEE P7000-series Standards Projects

https://ethicsinaction.ieee.org


- P7000 Model Process for Addressing Ethical Concerns during System Design
- P7001 Transparency of Autonomous Systems
- P7002 Data Privacy Process
- P7003 Algorithmic Bias Considerations
- P7004 Standard on Child and Student Data Governance
- P7005 Standard on Employee Data Governance
- P7006 Standard on Personal Data AI Agent Working Group
- P7007 Ontological Standard for Ethically driven Robotics and Automation Systems
- P7008 Standard for Ethically Driven Nudging for Robotic, Intelligent and Autonomous Systems
- P7009 Standard for Fail-Safe Design of Autonomous, Semi-Autonomous Systems
- P7010 Well-being metrics Standard for Ethical Artificial Intelligence and Autonomous Systems
- P7011 Stadard for the Porcess of Identifying and Rating the Trustworthiness of News Sources
- P7012 Standard for Machine Readable Personal Privacy Terms
- P7014 Standard for Ethical Considerations in Emulated Empathy in Autonomous and Intelligent Systems

# IEEE Standards Project P7007 for Ontological Standard for Ethically driven Robotics and Automation Systems


- IEEE Global Initiative on Ethics of Autonomous and Intelligent Systems.
  - https://ethicsinaction.ieee.org includes large document providing broad rationale.
  - Includes 15 separate working groups in IEEE Standards Association (IEEE-SA).
- Relevant group: P7007, Ethically driven Robotics and Automation Systems.
  - "IEEE P7007 Standards Project for Ontological Standard for Ethically driven Robotics and Automation Systems establishes a set of ontologies with different abstraction levels that contain concepts, definitions and axioms that are necessary to establish ethically driven methodologies for the design of Robots and Automation Systems."
  - http://standards.ieee.org/develop/project/7007.html
  - Must be IEEE member, observe patent-policy requirements to participate in working group.
  - "Not the intent to specify required ethical behaviors, but rather to formalize a vocabulary of terms, concepts, and relationships that can be used to enable unambiguous discussion among […] communities regarding what it means for autonomous systems to exhibit ethical behaviors."
  - Excellent forum with rich references, worth observation and participation.
- ***Active work:*** align several Ethical Control terms, concepts, use cases with P7007.

# Unmanned Maritime Autonomy Architecture (UMAA)

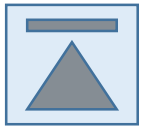
Richard R. Burgess, "Navy Requests Information for Unmanned Maritime Autonomy Architecture," SEA POWER, 20 FEB 19

- "The intent of UMAA is to provide overarching standards that various UUVs and USVs can be built to in order to avoid creating multiple conflicting systems in the future"
- "The UMAA is being established to enable autonomy commonality and reduce acquisition costs across both surface and undersea unmanned vehicles."
- Topics of interest include Situational Awareness, Sensor and Effector Management, Processing Management, Communications Management, Vehicle Maneuver Management, Vehicle Engineering Management, Vehicle Computing Management, Support Operations

Multiple public NAVSEA documents refer to autonomy efforts and UMAA.

- NAVSEA PMS 406 is Program Office for Unmanned Maritime Systems
- NAVSEA Fact Sheet: Unmanned Maritime Systems Program Office (PMS 406) https://www.navsea.navy.mil/Portals/103/Documents/Exhibits/SNA2019/UnmannedMaritimeSys-Small.pdf
- Automated Management of Maritime Navigation Safety
  Navy SBIR 2020.1 - Topic N201-059, https://www.navysbir.com/n20_1/N201-059.htm

Proposed Critical Path Forward



# Related resources of interest

This project draws on multiple relevant activities and capabilities. The following synopses are distilled from each respective resource.

# DoD Directive: Autonomy in Weapon Systems

- DoD Directive 3000.09, 21 NOV 2012 with change 1, 8 May 2017
- Original and update signed by DEPSECDEF Ashton Carter

**CONTROLLING REFERENCE OVERALL**

1. PURPOSE

   a. Establishes DoD policy and assigns responsibilities for the development and use of autonomous and semi-autonomous functions in weapon systems, including manned and unmanned platforms.

   b. Establishes guidelines designed to minimize the probability and consequences of failures in autonomous and semi-autonomous weapon systems that could lead to unintended engagements.

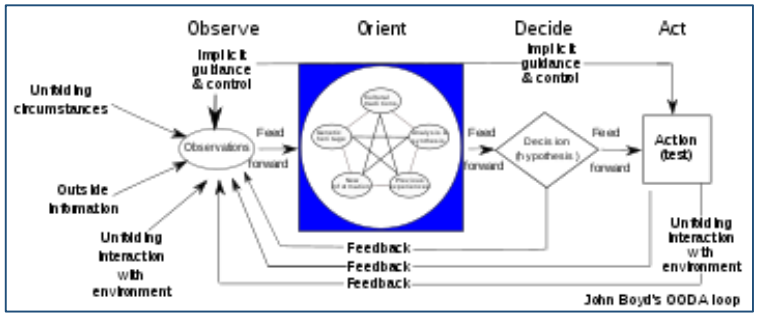
4. POLICY (excerpted)

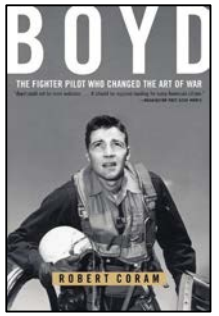
   a. Autonomous and semi-autonomous weapon systems shall be designed to allow commanders and operators to exercise appropriate levels of human judgment over the use of force.

   b. Persons who authorize the use of, direct the use of, or operate autonomous and semi-autonomous weapon systems must do so with appropriate care and in accordance with the law of war, applicable treaties, weapon system safety rules, and applicable rules of engagement (ROE). […]

6. RELEASABILITY. Cleared for public release. […]

# John Boyd and OODA Loop



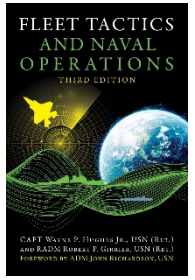

Wikipedia: John Boyd (military strategist) and Observe Orient Decide Act (OODA) Loop

- "… the key to victory is to be able to create situations wherein one can make appropriate decisions more quickly than one's opponent."

Robert Coram, *BOYD: The Fighter Pilot Who Changed the Art of War*, 2004.

- "John Boyd may be the most remarkable unsung hero in all of American military history. Some remember him as the greatest U.S. fighter pilot ever - the man who, in simulated air-to-air combat, defeated every challenger in less than forty seconds. Some recall him as the father of our country's most legendary fighter aircraft - the F-15 and F-16. Still others think of Boyd as the most influential military theorist since Sun Tzu. They know only half the story."
- "Boyd, more than any other person, saved fighter aviation from the predations of the Strategic Air Command. His manual of fighter tactics changed the way every air force in the world flies and fights. He discovered a physical theory that forever altered the way fighter planes were designed. Later in life, he developed a theory of military strategy that has been adopted throughout the world and even applied to business models for maximizing efficiency. And in one of the stories of modern military history, the Air Force fighter pilot taught the U.S. Marine Corps how to fight war on the ground. His ideas led to America's swift and decisive victory in the Gulf War and foretold the terrorist attacks of September 11, 2001."

# Fleet Tactics and Naval Operations




Wayne P. Hughes, Jr. and Robert P. Girrier, *Fleet Tactics and Naval Operations*, Third Edition, Naval Institute Press, Annapolis Maryland, June 2018.

- https://www.usni.org/press/books/fleet-tactics-and-naval-operations-third-edition

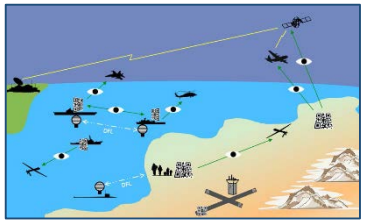
From newly added Chapter 12, A Twenty-First-Century Revolution:

- "At the most fundamental level, [Information Warfare] IW is about how to employ and protect the ability to sense, assimilate, decide, communicate, and act – while confounding those same processes that support the adversary."
- "Information Warfare broadly conceived is orthogonal to naval tactics. As a consequence, IW is having major effects on all six processes of naval tactics used in fleet combat – scouting and antiscouting, command-and-control, C2 countermeasures, delivery of fire, and confounding enemy fire."
- "Indeed there is a mounting wave of concern about how far automation will expand and what its impact will be on the continuum of cognition from data to information to knowledge. […] Navies are facing similar uncertainties."

Wayne Hughes coined the term "Network Optional Warfare" after many discussion sessions, directly contrasting it to Network Centric Warfare.  Thank you sir.

# Network Optional Warfare (NOW)

Overall Concept



**Naval forces do not have to be engaged in constant centralized communication.** Deployed Navy vessels have demonstrated independence of action in stealthy coordinated operations for hundreds of years.

- Littoral operations, deployable unmanned systems, and a refactored force mix for surface ships pose a growing set of naval challenges and opportunities. Network-optional warfare (NOW) precepts include Efficient Messaging, Optical Signaling, Semantic Coherence and Ethical Human Supervision of Autonomy for deliberate, stealthy, minimalist tactical communications.
- https://wiki.nps.edu/display/NOW/Network+Optional+Warfare

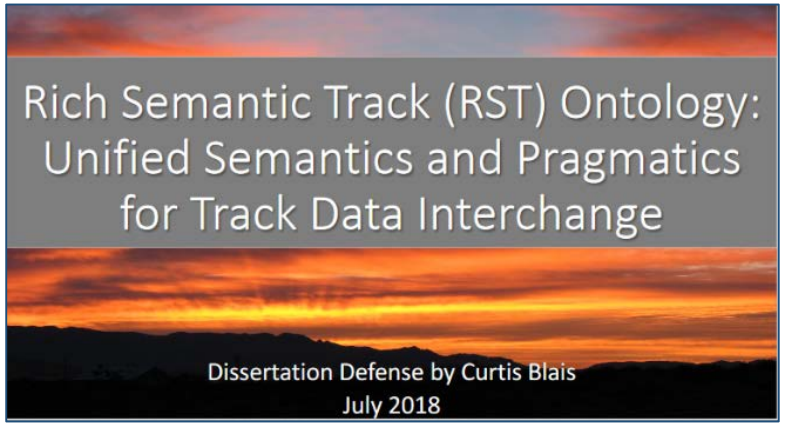
# Rich Semantic Track (RST)

- Track models commonly describe the same things in slightly different ways.
- Military decision-makers inundated with massive amounts of data is insufficient… without proper analytic tools, leads to information overload.
- Rich Semantic Track (RST) research examined dozens of track data models to identify common information content, to good effect.
- Formalized a *semantic* model of track data using Web Ontology Language (OWL) for standards-based reasoning by higher-level Artificial Intelligence (AI).
- Common characteristics across civil/military C2, robotic systems, and corresponding modeling & simulation (M&S) efforts.
- Semantic Coherence example for Network Optional Warfare

# Rich Semantic Track (RST)

Related Work | Curt Blais Dissertation

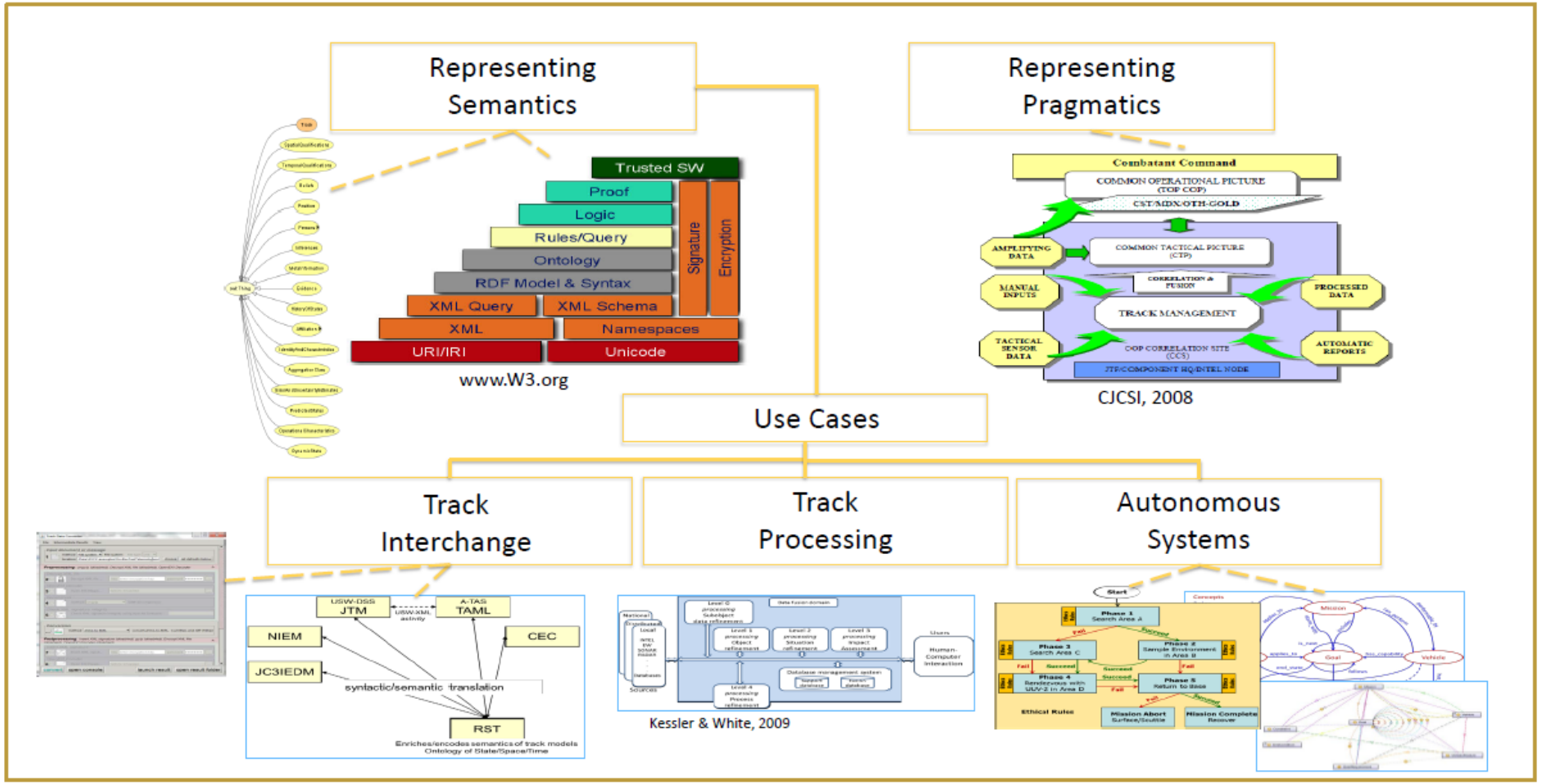
- DoD mandates data sharing practices, but practices have been mixed and uneven, resulting in perpetuation of system-centric data practices.
- Sharing and collective understanding of track data - collections of time-stamped perceptions of the state of objects of interest — are critical to warfighting systems.
- Shared understanding requires common semantics.
- The Rich Semantic Track (RST) ontology provides a foundation for shared understanding of track data.
- It is time to change the way DoD manages data and engineers systems, starting with adoption of the RST ontology and moving toward the vision of a Web of linked track data.




Network Optional Warfare (NOW)
Semantic Coherence

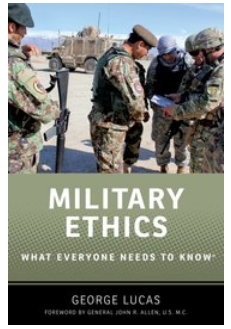
Rich Semantic Track (RST)

# Rich Semantic Track (RST) Ontology




*The Rich Semantic Track model formalizes the semantics and pragmatics of track data for broad application, including potential for improving data interchange and processing for unmanned systems.*

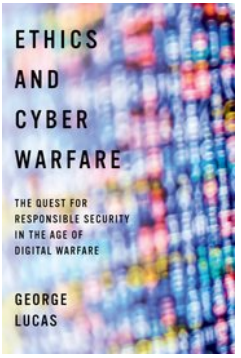
# George Lucas, Military Ethics




George Lucas, Military Ethics: What Everyone Needs to Know, Oxford University Press, 2016

- An approachable, case-driven account of global military ethics
- Raises and responds to some of the most important and provocative questions about the proper role and conduct of military organizations and their members
- Links sweeping, centuries-old political issues regarding war and the use of force in international relations to the day-to-day responsibilities of the individual members of the profession

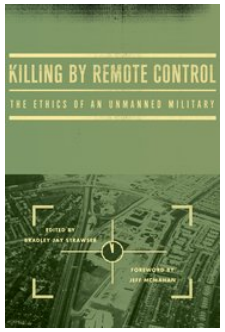
# George Lucas, Ethics and Cyber Warfare




George Lucas, Ethics and Cyber Warfare: The Quest for Responsible Security in the Age of Digital Warfare, Oxford University Press,

- Outlines a new code of ethics for today's "cyber warriors"
- Presents controversial new thesis that "state sponsored hacktivism" is a form of warfare
- Explores the ethical and legal dimensions of cyber warfare, grounding the discussion in a broad revisionist approach to military ethics and just war theory
- Offers a thorough and authoritative critique of international law and the Tallinn Manual

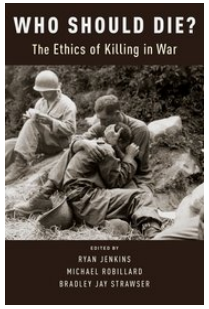
# Killing by Remote Control: the Ethics of an Unmanned Military




Bradley Jay Strawser editor, Killing by Remote Control: The Ethics of an Unmanned Military, Oxford University Press, 2012

- Addresses topic of contemporary debate: whether use of an unmanned military is ethical.
- The increased military employment of remotely operated aerial vehicles, also known as drones, has raised a wide variety of important ethical questions, concerns, and challenges. Many of these have not yet received the serious scholarly examination such worries rightly demand. This volume attempts to fill that gap through sustained analysis of a wide range of specific moral issues that arise from this new form of killing by remote control. Many, for example, are troubled by the impact that killing through the mediated mechanisms of a drone half a world away has on the pilots who fly them. What happens to concepts such as bravery and courage when a war-fighter controlling a drone is never exposed to any physical danger? This dramatic shift in risk also creates conditions of extreme asymmetry between those who wage war and those they fight. What are the moral implications of such asymmetry on the military that employs such drones and the broader questions for war and a hope for peace in the world going forward? How does this technology impact the likely successes of counter-insurgency operations or humanitarian interventions? Does not such weaponry run the risk of making war too easy to wage and tempt policy makers into killing when other more difficult means should be undertaken?

# Who Should Die? The Ethics of Killing in War




Bradley Jay Strawser editor, Who Should Die? The Ethics of Killing in War, Oxford University Press, 2017

- *Who Should Die? Editors' Introduction*
- Chapter 1. Liability to Deadly Force in War *Leonard Kahn*
- Chapter 2. Jus in Bello: Actual vs. Hypothetical Contract *Yitzhak Benbaji*
- Chapter 3. Do Some Soldiers Deserve to Die More Than Others? *David Whetham*
- Chapter 4. Defensive Liability: Four Common Mistakes *Kai Draper*
- Chapter 5. Fighting for One's Self *Michael Robillard*
- Chapter 6. An Axiomatic Theory of Just War: Forfeiture Theory *Stephen Kershnar*
- Chapter 7. Dignity, Self-Respect, and Bloodless Invasions *Saba Bazargan*
- Chapter 8. What is the Moral Problem with Killer Robots? *Susanne Burri*
- Chapter 9. Distributing the Cost of Rescue *Lars Christie*
- Chapter 10. Legality, Justice, and the War on Terrorism *Lionel K. McPherson*

# Military Ethics and Emerging Technologies


- Military Ethics and Emerging Technologies, edited by Timothy J. Demy, George R. Lucas Jr., and Bradley J. Strawser, Routledge, 2014.
- This volume looks at current and emerging technologies of war and some of the ethical issues surrounding their use. Although the nature and politics of war never change, the weapons and technologies used in war do change and are always undergoing development. Because of that, the arsenal of weapons for twenty-first century conflict is different from previous centuries. Weapons in today's world include an array of instruments of war that include, robotics, cyber war capabilities, human performance enhancement for warriors, and the proliferation of an entire spectrum of unmanned weapons systems and platforms. Tactical weapons now have the potential of strategic results and have changed the understanding of the battle space creating ethical, legal, and political issues unknown in the pre-9/11 world. What do these technologies mean for things such as contemporary international relations, the just-war tradition, and civil-military relations?
- This book was originally published in various issues and volumes of the *Journal of Military Ethics*.

# International Society for Military Ethics

- The International Society for Military Ethics (ISME) is an organization of military professionals, academics, and others formed to discuss ethical issues relevant to the military. The Society's meetings have been held each year since 1979.
- The chapter in North America is the founding chapter, formerly known as the "Joint Services Conference on Professional Ethics" (JSCOPE). Affiliate chapters have been established in Europe (Euro-ISME), and are in the process of formation in Latin America, Africa, and the Pacific Rim.
- https://www.internationalsocietyformilitaryethics.org

# Stockdale Center for Ethical Leadership

Stockdale Center for Ethical Leadership, U.S. Naval Academy (USNA)

- Authorized by the Secretary of the Navy in 1998, the Center for the Study of Professional Military Ethics (CSPME) undertook an ambitious mission – to promote and enhance the ethical development of current and future military leaders. In February 2006, the Superintendent of the Naval Academy directed the expansion of the Center, and the Center was renamed the Vice Admiral James B. Stockdale Center for Ethical Leadership.
- The Center could not have a finer model as its namesake than this most distinguished graduate of the United States Naval Academy. Admiral Stockdale was a man of unsurpassed courage and integrity who clearly understood the gravity of a leader's moment of ethical decision.

# Alan Schuller, Crossroads of Control: AI in Autonomous Weapons and International Humanitarian Law


- Network Optional Warfare (NOW) Blog, December 2018
- Alan L. Schuller, "At the Crossroads of Control: The Intersection of Artificial Intelligence in Autonomous Weapon Systems with International Humanitarian Law," *Harvard National Security Journal*, vol. 8 no. 2, 30 May 2017, pp. 379-425. (online, pdf)
- Alan L. Schuller, "Inimical Inceptions of Imminence: A New Approach to Anticipatory Self-Defense Under the Law of Armed Conflict," *UCLA Journal of International Law and Foreign Affairs*, vol. 18, no. 2, 2014, pp. 161-206. (online, pdf)
- Growing controversy surrounds the rapid development of artificial intelligence (AI) in weapon systems, with little consideration of intent or the variety of potential risks involved. The following papers provide significant detail and insight regarding actual legal aspects with respect to International Humanitarian Law (IHL). Key insights include recognition of the temporal aspect associated with naval missions. Long time intervals may occur between direction and execution, without frequent communication, but the need for human control remains essential throughout.

# College of Leadership and Ethics

College of Leadership and Ethics, U.S. Naval War College, Newport RI

- The College of Leadership and Ethics (CLE) works through three lines of effort: education, research and outreach. The leadership and ethics area of study in the electives program and core curriculum is the key focus of CLE education. CLE's leadership research, assessment, and analysis helps develop various leader development course curricula. CLE outreach includes support for leader development in various Navy communities.
- The College of Leadership & Ethics (CLE) is composed of a group of teaching scholars and practitioners who support Navy leader development initiatives and conduct research on leadership effectiveness. CLE provides an opportunity for students to engage in studies that broaden their educational experience by focusing on leadership and ethics. We develop stronger leaders by preparing them to fulfill their expanding and ever-widening roles as global leaders.

# Marine Corps University (MCU)
# School of Advanced Warfighting


*Mission Statement.* The School of Advanced Warfighting (SAW) develops lead planners and future commanders with the will and intellect to solve complex problems, employ operational art, and design/execute campaigns in order to enhance the Marine Corps' ability to prepare for and fight wars.

- Formulate solutions to complex problems and apply operational art in an uncertain geostrategic security environment.
- Employ knowledge of the operational level of war, the art of command, and ethical behavior in warfighting.
- Quickly and critically assess a situation, determine the essence of a problem, fashion a suitable response, and concisely communicate the conclusion in oral and written forms.
- Demonstrate the competence, confidence, character, and creativity required to plan, lead, and command at high level service, joint, and combined headquarters.
- Design and plan adaptive concepts to meet current and future challenges.

# Naval Leadership and Ethics Center

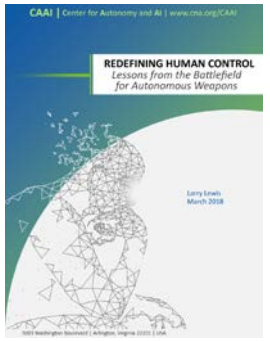
Naval Leadership and Ethics Center, U.S. Naval War College, Newport RI

- The Naval Leadership and Ethics Center (NLEC) is headquartered in beautiful Newport, Rhode Island, with locations in Dam Neck, Virginia and San Diego, California. Each year, the NLEC prepares more than 400 prospective commanding officers and their support teams for leadership success.

Committed to Ethical Leadership

- NLEC develops ethical, responsible, and accountable command leaders through interactive coursework. In addition to programs specifically designed for the command triad, NLEC offers a command spouse leadership course and general professional development for all levels of fleet leadership.

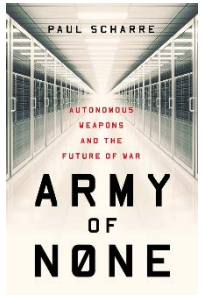
# Redefining Human Control: Lessons from the Battlefield for Autonomous Weapons




Larry Lewis, Redefining Human Control: Lessons from the Battlefield for Autonomous Weapons, CNA, March 2018.

- "This report examines the issue of human control with regard to lethal autonomy, an issue of significant interest in United Nations discussions in the Convention on Certain Conventional Weapons (CCW) forum. We analyze this issue in light of lessons and best practices from recent U.S. operations. Based on this analysis, we make the case for a wider framework for the application of human control over the use of force. This report recommends that CCW discussions currently focusing on process considerations, such as human control, should instead focus on outcome—namely, mitigation of inadvertent engagements. This allows consideration of a more complete set of benefits and risks of lethal autonomy and better management of risks. The report also describes best practices that can collectively serve as a safety net for the use of lethal autonomous weapons. It concludes with concrete recommendations for how the international community can more effectively address the risk of inadvertent engagements from lethal autonomy."
- Press release, more at www.cna.org/CAAI
- CNA is a nonprofit research and analysis organization located in Arlington, VA.

# *Army of None: Autonomous Weapons and the Future of War*




Paul Scharre, *Army of None: Autonomous Weapons and the Future of War,* W.W. Norton, New York 2018. www.paulscharre.com/army-of-none

- "What happens when a Predator drone has as much autonomy as a Google car? Or when a weapon that can hunt its own targets is hacked? Although it sounds like science fiction, the technology already exists to create weapons that can attack targets without human input."
- *"Army of None* engages military history, global policy, and cutting-edge science to argue that we must embrace technology where it can make war more precise and humane, but without surrendering human judgment. When the choice is life or death, there is no replacement for the human heart."
- Interestingly anticipates many of the approaches taken in this project.

# Autonomous Weapons and Human Control


Paul Scharre and Kelley Sayler, Autonomous Weapons and Human Control, Center for a New American Security (CNAS), April 2016.

- "Nations from around the world met at the United Nations in Geneva, Switzerland to discuss autonomous weapons, potential future weapons that would select and engage targets on their own. Ensuring "meaningful human control" over future weapons has been a topic of much debate, with some human rights activists advocating for a preemptive ban. Increasing autonomy in weapons raises the question of how much human involvement is required in lethal attacks."
- "In this brief, Scharre and Sayler explain how autonomy is already used in many weapons today and how future fully autonomous weapons would be different. Autonomous weapons would be programmed by humans and launched by a human. Once launched, however, the weapon would have the freedom to select its own targets over a wide area according to preprogramed parameters, raising new legal, ethical, and safety questions."
- Additional reports at CNAS Task Force on Artificial Intelligence and National Security

# NY Times: video, selected articles

Jonah M. Kessel, Killer Robots Aren't Regulated. Yet. NY Times, 13 DEC 2019. "Killing in the Age of Algorithms" is a New York Times documentary examining future of AI and warfare.

Carol Giacomo, Are You Ready for Weapons That Call Their Own Shots? NY Times, 26 JUN 2019. The speed with which the military is developing artificial intelligence raises fears of an autonomous weapons race.

Kelsey D. Atherton, Are Killer Robots the Future of War? Parsing the Facts on Autonomous Weapons, NY Times, 15 NOV 2018. Under what circumstances should militaries delegate the decision to take a human life? It is a moral leap that the international community is grappling with.

John Markoff, Fearing Bombs That Can Pick Whom to Kill, NY Times, 11 NOV 2014. Weapons that rely on artificial intelligence to decide what to target could become increasingly difficult to control, critics warn.

# USA is not a signatory to, or fully compliant with, multiple international conventions of importance

United Nations Convention on Law of the Sea (UNCLOS)

- (United Nations) (Wikipedia) (NOAA)

UN Convention on Prohibition of Use, Stockpiling, Production and Transfer of Anti-Personnel Mines and on Their Destruction

- (United Nations) (Wikipedia) aka “Ottawa Treaty”

International Court of Justice (ICJ)

- (United Nations) (Wikipedia)

# Trusting Software and Trusting Data


- Network Optional Warfare (NOW) Blog, January 2016
- In 1983, Dennis Ritchie and Ken Thompson jointly received the Turing Award for their development of generic operating systems theory, and specifically for the implementation of the UNIX operating system.
- Ken Thompson's lecture was Reflections on Trusting Trust, with the subtitle "*To what extent should one trust a statement that a program is free of Trojan horses? Perhaps it is more important to trust the people who wrote the software.*" This talk can still surprise: he describes source code that looks like it does one thing, but actually performs things that are quite different.
- So in effect, Ken Thompson chose his Turing Award moment to reveal to the world that he had superuser and user access for every Unix system and server on the planet. Further he revealed that, even with a great many people scrutinizing and rebuilding the source code, and even despite users banging on Unix daily everywhere, anyone else might use a super password for each and every account. Meanwhile no one else knew that the super password existed, much less that it quietly insisted on re-propagating itself in each fresh new copy of Unix. **No kidding.**
- How does the Navy get beyond software barriers to reach the next level of capability: **trust for shared data**?

# DoD Digital Modernization Strategy

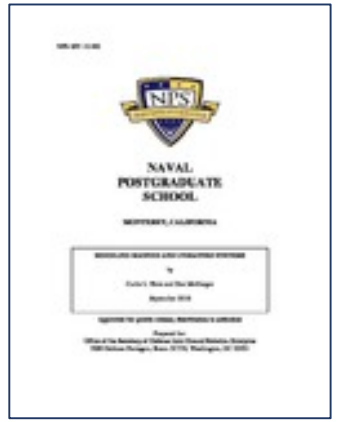

DoD CIO Digital Modernization Strategy provides Information Resource Management Strategic Plan FY 19-23, guiding IT transformation (interview)

- Advancement of digital environment to ultimately ensure competitive advantage for warfighters. Addresses Cybersecurity, AI, Cloud,C3.
- Four initiatives: Innovation, Optimization, Cybersecurity, Talent.
- Relevant sections include: Establish JAIC, Enterprise Cloud, Modernize C4, Treat Data as Strategic Asset, International Collaboration Partnerships and Allied Interoperability, Protect Positioning Navigation & Timing (PNT), End-To-End Airborne ISR Data Transport, Info Sharing to Mobile Users, Drive Standards into IT Systems, Transform Cybersecurity Architecture, End-To-End Identity Credential Asset Management, Risk Management

# Autonomy Recommendation for ACM Code of Ethics and Professional Conduct


- Network Optional Warfare (NOW) Blog, February 2018
- As noted in Richard W. Hamming's Turing Award Lecture of fifty years ago: ethics, professional behavior, and social responsibility cannot be separated from the diverse fields in which computer science is applied [2].
- Recommendation: "**Recognize potential risks associated with autonomy.** Systems operating remotely or with minimal human supervision (for example, drones or driverless vehicles) may have the capacity for inflicting unintended lethal force. Safeguards, legal requirements, moral imperatives, and means for asserting direct human control must be considered, in order to avoid the potential for unintended injury or loss of life due to emergent behavior by robotic systems."
- Comments acknowledged and considered, not accepted.
- ACM Code of Ethics and Professional Conduct, adopted 22 JUN 2018.

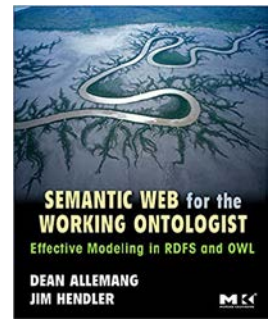
# Modeling Manned and Unmanned Systems



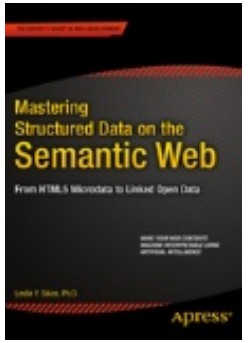

- Curtis L. Blais and Don McGregor, NPS Technical Report NPS-MV-16-002, September 2016
- https://calhoun.nps.edu/handle/10945/50678
- *Abstract (excerpt). "*Unmanned systems are being introduced rapidly into operational forces. However, there is little capability in modeling the performance of manned systems and unmanned systems to distinguish the two, making it impossible, currently, to use simulation for analysis of future manned/unmanned system force mixes and most effective levels of autonomy for unmanned systems."
- "We contend that is it not enough to simply insert additional 'human-like' entities into the simulations and call them 'unmanned systems' possessing nearly the same behaviors implemented in nearly the same ways. Something more needs to be represented…" p. 18
- "We have come to a point in combat modeling where we can no longer be satisfied with simplistic models of human performance. We must be able to investigate the complex interplay that will occur among humans and unmanned systems. Suffice to say, we are in the early stages of a fascinating era of research and development that will bring about greater precision in our concepts and terminology relating to unmanned systems, while likely redefining our notions of manned systems as well." pp. 42-43

# Recommended Reading: Semantic Web, X3D

- Dean Allemang and Jim Hendler, Semantic Web for the Working Ontologist: Effective Modeling in RDFS and OWL, second edition, 2012.
- Don Brutzman and Leonard Daly, X3D for Web Authors, Morgan, 2007.
- Bob DuCharme, Learning SPARQL, second edition, 2013.
- Leslie F. Sikos, Mastering Structured Data on the Semantic Web: from HTML5 Microdata to Linked Open Data, 2015.
- Wikibooks SPARQL edition covers both language and also Web-based GUI: Wikidata Query Service SPARQL endpoint.






Related Work

# X3D Ontology for 3D Graphics

Of direct interest to this project:

- X3D Ontology for Semantic Web has details, source for semantic query of X3D models
- Template for next stages of work in this project found in **X3D Ontology presentation**

Semantic Web Pro Workshop, 3 DEC 2019, Paris France

- *"Semantic Web techniques and standards have emerged in many fields. By facilitating interoperability across the Web, they are giving new impetus to the integration of heterogeneous data and the construction of knowledge graphs that provide a foundation for dynamic business ecosystems."* – from presentation announcement
- "The Semantic Web3D: towards comprehensive representation of 3D content on the Semantic Web" by Christophe Mouton, Jakub Flotyński, Athanasios Malamos, Don Brutzman, John Carlson, Marc Petit, Nicholas Polys, and Leslie Sikos.

# X3D Ontology for Semantic Web

The X3D Ontology for Semantic Web provides terms of reference for semantic query of X3D models.

Motivation | Availability | Design and Design Patterns | OWLDoc | Queries | References | Tools | TODO | Contact

---

## Motivation

The X3D Semantic Web Working Group mission is to publish models to the Web using X3D in order to best gain Web interoperability and enable intelligent 3D applications, feature-based 3D model querying, and reasoning over 3D scenes.

Motivating insights:

> "The answer to your question is the response to the query." Jim Hendler and Dean Allemang
>
> "Trying to use the Semantic Web without SPARQL is like trying to use a relational database without SQL." Tim Berners-Lee
>
> "The proof of the pudding is in the eating." Wiktionary

---

https://www.web3d.org/x3d/content/semantics/semantics.html

# Collaborative X3D Visualization

- Collaborative 3D Visualization for Ashore, Afloat and Expeditionary Readiness Workshop, Web3D Consortium and NAVFAC, 6 DEC 2019, Arlington VA
- One-day workshop to provide Naval enterprise leaders presentations on the use of collaborative Web-based X3D visualization techniques by Government, Academia and Industry practitioners.
- Sixteen presentations, two videos, real-time building scans
- SPIDERS3D Collaborative Visualization for Navy and US Coast Guard Risk Assessment Support describes deployment planning for XLUUV at Port Hueneme CA in support of COMOPTEVFOR, NAVSEA PMS 406

Related Work

# IEEE Distributed Interactive Simulation (DIS)

I/ITSEC Interservice Industry Training Simulation Education Conference,

Distributed Interactive Simulation (DIS) 101 Tutorial, 2 DEC 2019, Orlando FL

- *"The Distributed Interactive Simulation (DIS) protocol is a well-established IEEE standard for packet-level exchange of state information between entities in military simulations. DIS facilitates simulation interoperability through a consistent over-the-wire format for information, widely agreed upon constant enumeration values, and community-consensus semantics."* – from presentation announcement
- Distributed Interactive Simulation 101 Tutorial: The Basics tutorial by Don Brutzman
- Github Open-DIS Software Archive
- Repeatable Unit Testing of Distributed Interactive Simulation (DIS) Protocol Behavior Streams using Web Standards by NPS thesis student Tobias Brennenstuhl (potential NATO connection)

Related Work

Of direct interest to this project:

- AUV Workbench outputs simulation results using DIS protocol, making it feasible to someday export and include Ethical Control testing in other Live Virtual Constructive (LVC) scenarios.

# Simulation Interoperability Standards Organization (SISO)

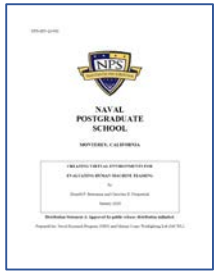

Simulation Interoperability Standards Organization (SISO) is an international organization dedicated to the promotion of modeling and simulation interoperability and reuse for the benefit of a broad range of M&S communities.

- SISO slogan: “Simulation Interoperability and Reuse through Standards”

Several SISO working groups are relevant to Ethical Control future plans establishing virtual environments for unmanned-systems qualification.

- C2SIM: C2 - Simulation Systems Interoperation uses ontology-governed compatible message interchange, especially focused on NATO forces.
- DIS/RPR FOM: Distributed Interactive Simulation / HLA Real-time Platform Reference Federation Object Model for distributed simulation networks.
- VV&A: Verification Validation & Accreditation/Acceptance emphasizes requirements and metrics within larger Systems Engineering spirals.

# Creating Virtual Environments for Evaluating Human-Machine Teaming




## CREATING VIRTUAL ENVIRONMENTS FOR EVALUATING HUMAN-MACHINE TEAMING

Donald P. Brutzman and Christian R. Fitzpatrick, Naval Research Program (NRP) Project NPS-19-M285-A, January 2020.

*Abstract*. With the emergence of robots on the battlefield, it is critical for the Marine Corps to tactically integrate existing unmanned assets with manned systems during Marine Air Ground Task Force (MAGTF) operations. In parallel, the Marine Corps must also look forward to identify capability gaps that future unmanned systems might address. To do both requires extensive field testing, which is often unfeasible and always costly. This effort proposes the use of virtual environments (VE), virtual reality (VR) and agent-based modeling to conduct scenario-based assessments of Manned-Unmanned Teaming (MUM-T) during combat operations.

To pursue such goals, the project examined a variety of relevant tactical scenarios where Marines and robots act in concert to achieve specific mission objectives. Such tactical scenarios are further assessed using deterministic combat simulations to create a valid methodology for behavior creation and assessment within each scenario-specific problem space. Support for a complete range of combat simulations was determined as a necessary part of VE design explorations since specific MUM-T tactics, techniques and procedures (TTPs) are expected to co-evolve constantly as sensor, communication and vehicle capabilities continue to improve. Such diversity was supported through establishment of the MOVES Live Virtual Constructive (LVC) Laboratory for diverse simulation tools. Additionally two general approaches for the coordination of Manned-Unmanned Teaming (MUM-T) behaviors were considered, each beginning with a high-level description of expected behaviors. Completion of the goal tasks indicates that combined human-robot teams have achieved a desired world state.

This research surveyed a large variety of combat models and visualization tools to create the best and broadest possible environment for Marine Corps decision-makers to understand the complexity and warfighting value of the MUM-T battlespace. Shared VEs can potentially be used during the Marine Corps Force Development System (MCFDS) annually to plan for the integration of MUM-T into combat units. As the DoD is generally unfamiliar with such operations but is eagerly anticipating their development, it is quite clear that the use of live, virtual, constructive (LVC) simulations to wargame these capabilities becomes fundamental for all progress. Ultimately such MUM-T co-development is the critical path needed to expand MAGTF capabilities and avoid MAGTF vulnerabilities.

# Coactive Design and Interdependency Analysis

- Matthew Johnson and Alonso Vera, "No AI is an Island: The Case for Teaming Intelligence", *AI Magazine*, vol. 40 no.1, Spring 2019
- *Abstract*. "The purpose of this article is to draw attention to an aspect of intelligence that has not yet received significant attention from the AI community, but that plays a crucial role in a technology's effectiveness in the world, namely teaming intelligence. We propose that AI will reach its full potential only if, as part of its intelligence, it also has enough teaming intelligence to work well with people. Although seemingly counterintuitive, the more intelligent the technological system, the greater the need for collaborative skills. This paper will argue why teaming intelligence is important to AI, provide a general structure for AI researchers to use in developing intelligent systems that team well, assess the current state of the art and, in doing so, suggest a path forward for future AI systems. This is not a call to develop a new capability, but rather, an approach to what AI capabilities should be built, and how, so as to imbue intelligent systems with teaming competence."
- Strong resonances exist with Ethical Control that deserve further exploration.

# Naval Research Advisory Committee (NRAC)


## "Autonomous and Unmanned Systems in Department of Navy"

- Technical Report, September 2017
- Chair: Dr. James Bellingham, Director of Center for Marine Robotics, Woods Hole Oceanographic Institute (WHOI). Former Chief Technologist at Monterey Bay Aquarium Research Inst. (MBARI)
- https://www.nrac.navy.mil/docs/2017_rpt_Autonomous_Unmanned_Systems_Don-SEP2017.pdf

**Recommendation #1**: *create comprehensive data plan to field autonomy*.

- "DoN must urgently develop an organizational data plan."
- "In the future, data will win wars."
- "Data is ultimate 'component' for AI systems and must be controlled."

# Naval Applications of Machine Learning Workshop

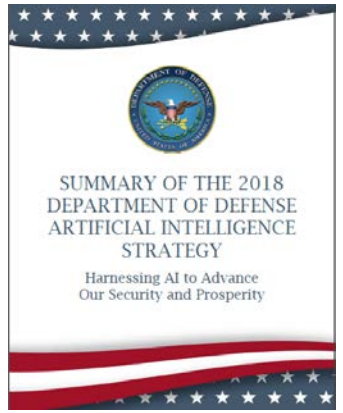
SSC Pacific, 13-15 FEB 2018

- Broad interest: 40 presentations and 50 posters!
- Broad range of topic categories:
  - Naval Research and Development,
  - Computer Vision,
  - Autonomy and Cognitive Electronic Warfare,
  - Cybersecurity, Algorithms, and Theory


- https://sites.google.com/go.spawar.navy.mil/naml2018/agenda
- Plenary roundtable discussion recognized value and need to establish a machine learning (ML) data strategy, for both inputs and outputs
- Report and article online. Also notification about NAML 2020.

# DoD Artificial Intelligence (AI) Strategy

## February 2019




- https://media.defense.gov/2019/Feb/12/2002088963/-1/-1/1/SUMMARY-OF-DOD-AI-STRATEGY.PDF
- "*Leading in military ethics and AI safety.* The Department will articulate its vision and guiding principles for using AI in a lawful and ethical manner to promote our values. We will consult with leaders from across academia, private industry, and the international community to advance AI ethics and safety in the military context. We will invest in the research and development of AI systems that are resilient, robust, reliable, and secure; we will continue to fund research into techniques that produce more explainable AI; and we will pioneer approaches for AI test, evaluation, verification, and validation. We will also seek opportunities to use AI to reduce unintentional harm and collateral damage via increased situational awareness and enhanced decision support. As we improve the technology and our use of it, we will continue to share our aims, ethical guidelines, and safety procedures to encourage responsible AI development and use by other nations."
- Terri Moon Cronk, "DOD Unveils Its Artificial Intelligence Strategy", Defense.gov
- Execution:

# Defense Innovation Board (DIB) AI Principles Project


- https://innovation.defense.gov/ai (Primary Document) (Supporting Document)
- "We have witnessed how deeply committed the women and men who work in the Department are to ethics: avoiding civilian casualties, adhering to international humanitarian law, and collaborating with allies in international fora to advance international law and norms. Additionally, the Department extensively tests all its systems, especially weapons systems, and systems employing AI will likely be subjected to more scrutiny than ever before."
- "The DIB noted that -- as with all new technologies -- rigorous work is needed to ensure new tools are used responsibly and ethically. The stakes are high in fields such as medicine or banking, but nowhere are they higher than in national security."
- Approved by DIB on 31 OCT 2019

# Joint Artificial Intelligence Center (JAIC) https://www.ai.mil


DoD CIO Joint Artificial Intelligence Center (JAIC) tasked to execute DoD AI Strategy.

JAIC's Guiding Tenets:

- AI is critical to the future of United States' national security, and the JAIC is the focal point for the execution of the DoD AI Strategy.
- We exist to create and enable impact for the Armed Services and DoD Components across the full range of their missions – from back office to front lines of the battlefield.
- Leadership in ethics and values is core to everything we do at the JAIC.
- The JAIC attracts the best and brightest people from across DoD, commercial industry, and academia. When they get here, they are empowered to bring their expertise to the Department's transformation.
- Ethical Principles for AI: *Responsible, Equitable, Traceable, Reliable, Governable*.

# DoD Prototyping Guidebook, Emerging Capability and Prototyping

DoD Prototyping Guidebook, OUSD R+E

- Issued by the Office of the Under Secretary of Defense for Research and Engineering's Prototypes and Experiments office.
- This guidebook attempts to capture and consolidate the approaches, best practices, and recommendations that Department of Defense (DoD) Components have used for decades to plan and execute prototyping projects for the DoD. This guidebook is an introductory and reference document that can be used by DoD research and acquisition professionals conducting prototyping projects and staff officers seeking to increase their knowledge of prototyping.

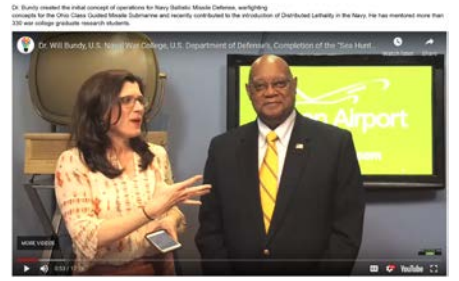
# CTF 151: Counter-Piracy



- Combined Task Force 151 (CTF 151) is one of three task forces operated by Combined Maritime Forces (CMF).
- In accordance with United Nations Security Council Resolutions, and in cooperation with CMF coastal states, CTF 151's mission is to deter and disrupt piracy and armed robbery at sea and to engage with regional and other partners to build capacity and improve relevant capabilities in order to protect global maritime commerce and secure freedom of navigation.
- CTF 151 was established in January 2009 with a specific piracy mission-based mandate, currently endorsed under UNSCR 2316. This has now been expanded to include conducting wider maritime security operations in support of CMF.

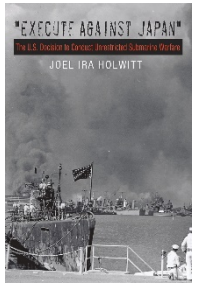
# Feedback by “Doc” Will Bundy

Ph.D., CAPT USN (Ret.) Naval War College, 1946-2019



- UUVs will need to operate by mission command, like submarines, given broad missions and the UUV will decide what targets need to be engaged
- To follow this ethical argument, must build Mission Command capabilities into your intelligent agent controller
- High-confidence Automated Target recognition will be required.
- We need to let machines kill things.
- Relevant: unrestricted submarine warfare in WWII. Recommended reading: ‘Execute Against Japan – The US Decision To Conduct Unrestricted Submarine Warfare’ by Joel Holwitt
- Noted “collaborative mission autonomy“ as important concept / capability
- Recommendation: need to engage LTG Shanahan of Joint AI Command.

# Execute Against Japan – The US Decision To Conduct Unrestricted Submarine Warfare




- Joel Ira Holwitt, Texas A+M University Press
- " . . . until now how the Navy managed to instantaneously move from the overt legal restrictions of the naval arms treaties that bound submarines to the cruiser rules of the eighteenth century to a declaration of unrestricted submarine warfare against Japan immediately after the attack on Pearl Harbor has never been explained. Lieutenant Holwitt has dissected this process and has created a compelling story of who did what, when, and to whom."—The Submarine Review
- "Execute against Japan should be required reading for naval officers (especially in submarine wardrooms), as well as for anyone interested in history, policy, or international law."
— Adm. James P. Wisecup, President, US Naval War College (for Naval War College Review)
- "Although the policy of unrestricted air and submarine warfare proved critical to the Pacific war's course, this splendid work is the first comprehensive account of its origins—illustrating that historians have by no means exhausted questions about this conflict."—World War II Magazine
- "US Navy submarine officer Joel Ira Holwitt has performed an impressive feat with this book. . . Holwitt is to be commended for not shying away from moral judgments . . . This is a superb book that fully explains how the United States came to adopt a strategy regarded by many as illegal and tantamount to 'terror'."—Military Review

# Taking Back the Seas: Transforming the U.S. Surface Fleet for Decision-Centric Warfare


- Center for Strategic and Budgetary Assessments (CSBA), 31 December 2019

"The U.S. Navy's surface fleet is at a crossroads. Today's force lacks the size, resilience, and offensive capacity to contribute effectively to degrading, delaying, or denying aggression. These shortfalls are especially problematic in light of the fact that the surface fleet will play an increasingly important role in the U.S. Navy's ability to counter enemy attacks. The current fleet is also fiscally unsustainable due to growing operations and support costs for today's highly integrated and manpower-intensive surface combatants. ***New technologies for unmanned systems, sensors, weapons, C3, and countermeasures*** could allow significant improvements in the surface fleet's ability to create complexity for an adversary and harden surface forces from attack while improving surface force's capacity for maritime or land strike."

Bryan Clark and Timothy A. Walton

# Navy Force Structure and Shipbuilding Plans: Background and Issues for Congress

Congressional Research Service
Informing the legislative debate since 1914

- Congressional Research Service, 4 February 2020
- INFSA = Integrated Naval Force Structure Assessment
- (excerpted from abstract) “DON officials suggest that the INFSA could shift the fleet to a more distributed architecture that includes a reduced proportion of larger ships, an increased proportion of smaller ships, and a ***newly created category of large unmanned surface vehicles (USVs) and large unmanned underwater vehicles (UUVs)*** . Such a change in fleet architecture could alter, perhaps substantially, the mix of ships to be procured for the Navy and the distribution of Navy shipbuilding work among the nation’s shipyards.”

Ronald O’Rourke, Specialist in Naval Affairs

# Acting SECNAV Modly: unmanned, data, wargame, iterate

- SECNAV Modly: Path to 355 Ships Will Rely on New Classes of Warships
- *USNI News*, Megan Eckstein, 3 February 2020


"You look at the frigate program: we think, because of the way we've approached that program, we've probably taken three years off the product development lifecycle for that. So we have to start doing the same type of thing: looking at proven hulls, things that can be adaptable for different areas. I understand the Hill's concerns about ***unmanned***, and we get that. … We have to convince them with ***data***: we have to ***wargame*** this, we have to ***iterate*** it over and over again."

Honorable Thomas Modly, Acting SECNAV

# OSD Digital Engineering Strategy


Digital Engineering Strategy, Washington DC, June 2018

- Department of Defense Announces its Digital Engineering Strategy
- Under Secretary of Defense for Research and Engineering Michael Griffin

- Formalize the development, integration, and use of models to inform enterprise and program decision making,
- Provide an enduring, authoritative source of truth,
- Incorporate technological innovation to improve the engineering practice,
- Establish a supporting infrastructure and environment to perform activities, collaborate and communicate across stakeholders,
- Transform the culture and workforce to adopt and support digital engineering across the lifecycle.

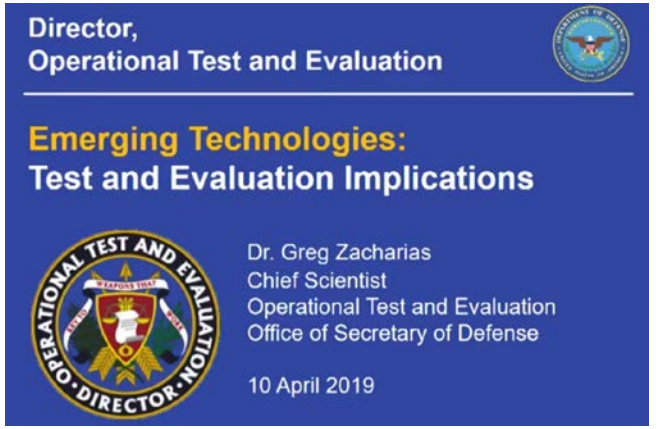
# Greg Zacharias: Emerging Technologies, Test and Evaluation (T&E) Implications

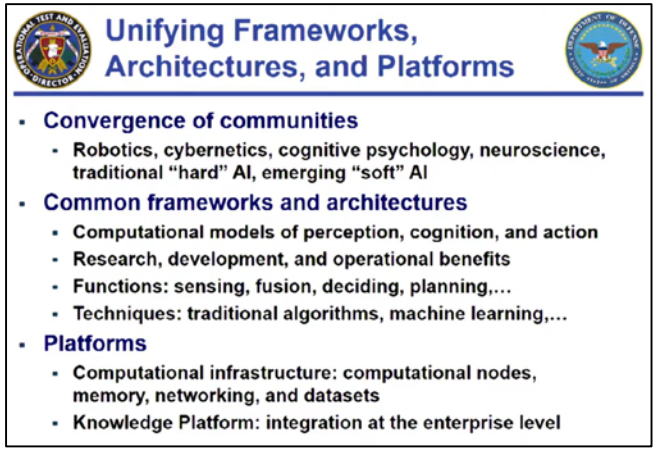
Presentation at Institute for Defense Analyses (IDA)

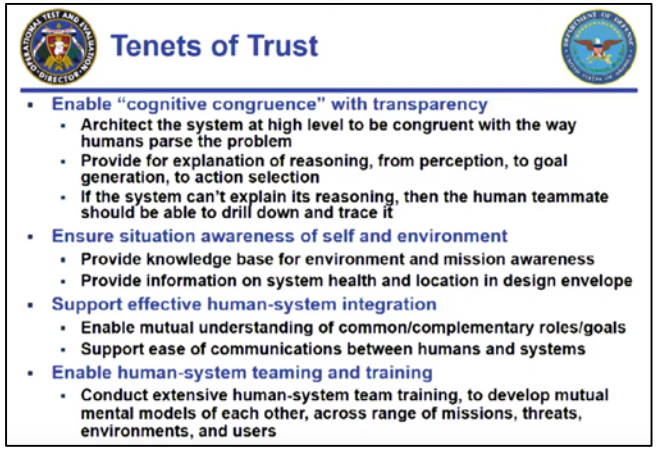
- https://www.youtube.com/watch?v=HtFJzua3xXM
- July 2019, Alexandria Virginia

Test and evaluation (T&E) of emerging unmanned systems holds many challenges beyond current paradigms for operations, assessment and analysis.

Shared approaches are essential and trust is central for effective human-system teaming across enterprise.



# AFCEA-GMU Critical Issues in C4I Symposium, 19-20 May 2020

- https://www.afcea.org/events/virtual/gmu
- CxO Roundtable: Emerging Technology and the Future of Warfare
  - USAF CTO, USA CTO, NATO ACT, DISA ETD executives provide "big picture" overview.
  - *Response:*

  Current work on Data-Centric Security may prove helpful as a way to reliably stitch together sensor/actor endpoints over network-relay nodes in the distributed data architectures described. End-to-end data security leads to mutual trust and command authority, regardless of intervening network connections, intermittent gaps, or opponent intercept. Data-Centric Security is a combination of structured messaging, efficient EXI compression, digital signature, and document encryption, in correct order, for round-trip messaging. Addition of blockchain distributed ledger has potential to further add integrity measures for aggregated message sets, confirming receipt-response-sequencing without undetected message loss.
- Chris McGuire, Director for Research and Analysis on National Security Commission on Artificial Intelligence (NSCAI) notes new pillar on AI Ethics.

# National Security Commission on AI


Independent Commission "to consider the methods and means necessary to advance the development of artificial intelligence, machine learning, and associated technologies to comprehensively address the national security and defense needs of the United States."

- https://www.nscai.gov/reports

First Quarter Recommendations Memo topics of relevant interest:

- *Improve AI Cooperation among Key Allies and Partners*
- *Advance Ethical and Responsible AI*

# Underwriters Labs UL4600: Standard for Safety for the Evaluation of Autonomous Products

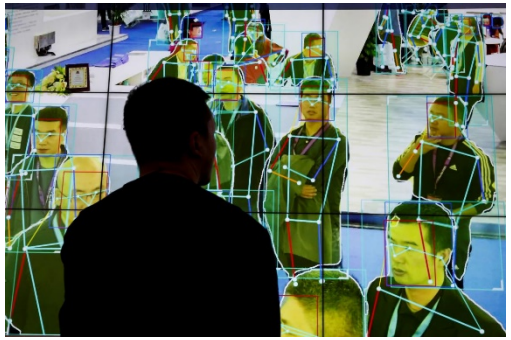
- ANSI/UL 4600 Standard for Safety for the Evaluation of Autonomous Products encompasses fully autonomous systems that move such as self-driving cars, along with applications in mining, agriculture, maintenance, and other vehicles including lightweight unmanned aerial vehicles (UAVs). It seeks to specifically address the ability of autonomous products to perform safely and as intended - without human intervention - based on their current state and sensing of the operating environment. Reliability of hardware and software necessary for machine learning, sensing of operating environment and other safety aspects of autonomy is also addressed.
- Description and Purchase ($716-$1567)
- Edge Case Research on UL4600

# DEPARTMENT OF DEFENSE STANDARD PRACTICE: SYSTEM SAFETY (MIL-STD-882E, 11 May 2020)

- This system safety standard practice is a key element of Systems Engineering (SE) that provides a standard, generic method for the identification, classification, and mitigation of hazards.
- DoD is committed to protecting personnel from accidental death, injury, or occupational illness and safeguarding defense systems, infrastructure, and property from accidental destruction, or damage while executing its mission requirements of national defense.
- SOFTWARE CONTROL CATEGORIES. Definition: Autonomous (AT). Software functionality that exercises autonomous control authority over potentially safety-significant hardware systems, subsystems, or components without the possibility of predetermined safe detection and intervention by a control entity to preclude the occurrence of a mishap or hazard.
- Document (104 pages)
- Defense Acquisition University (DAU), Environment, Safety, and Occupational Health (ESOH)

# China Won't Win the Race for AI Dominance:
## Authoritarians Love Data, but Innovation Matters More

FOREIGN AFFAIRS



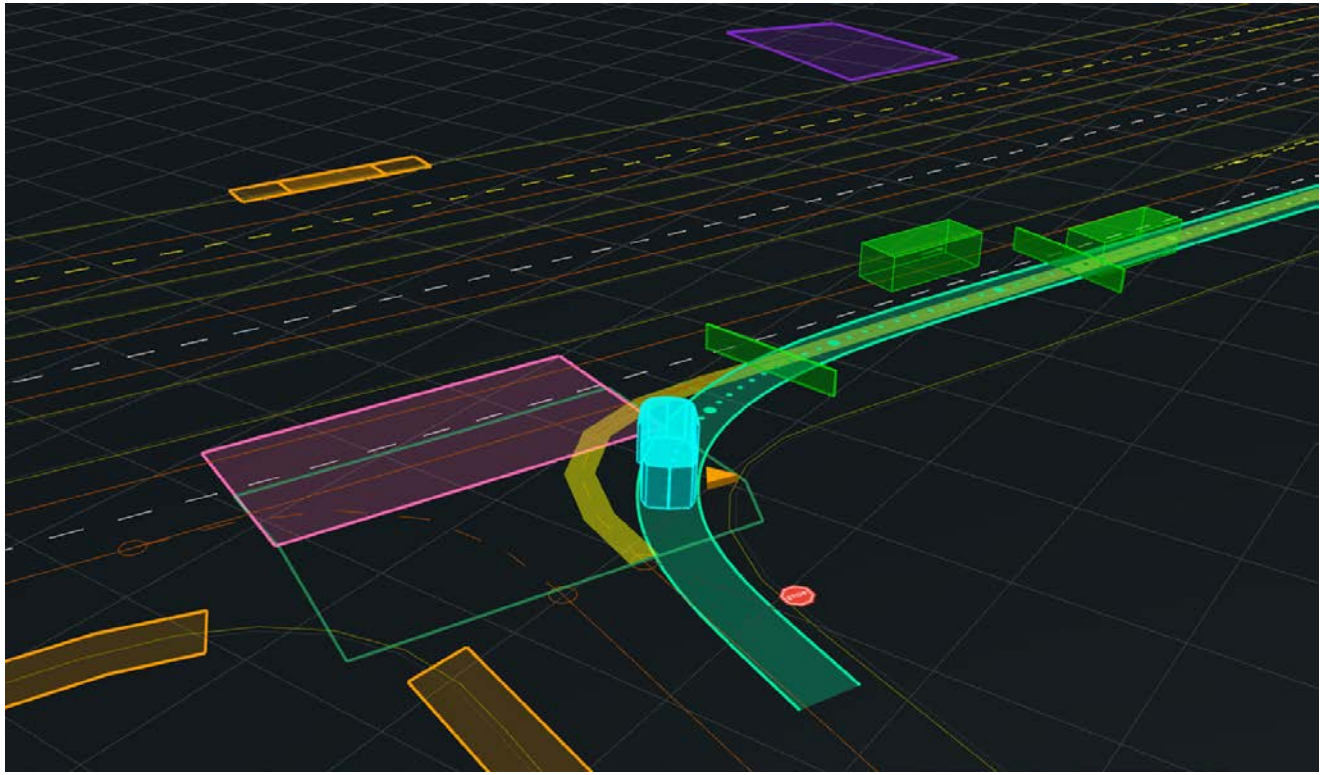
- DATA ALONE ARE NOT ENOUGH
- "China made international headlines by effectively leveraging its surveillance technology for contact tracing in response to COVID-19, the disease caused by the novel coronavirus.
  And yet the country's alleged data advantage is hugely overblown. One reason is that data are highly domain specific and don't often solve more than the problem for which they were gathered. China's disregard for privacy enables it to snoop on its citizens, but not much else. And an abundance of surveillance data doesn't give China an advantage in applying artificial intelligence to such ends as drug discovery or self-driving cars, for example."

Foreign Affairs, Carl Benedikt Frey and Michael Osborne, 19 June 2020

# Best practice: “Off road, but not offline: How simulation helps advance our Waymo Driver”

Industry is already operating at scale

- Google Waymo blog, 28 April 2020
- In simulation, "we drive around 20 million miles a day" and "over 15 billion miles“ total.
- LIDAR laser sensors looking in all directions
- Physically based modeling, simulation, reenactment
- Repeatable real/virtual replay of data and logic
- Large-scale regression tests
- Why not Navy and USMC?



# Autonomy & Autonomous Unmanned Systems" Overview, Investment Approach, Opportunities

Jason Stack, Office of Naval Research (ONR), Autonomy Portfolio

- Presentation to National Academy of Sciences (NAS) 26 NOV 2019
- Comprehensive review of multiple concerns, capabilities, challenges and opportunities

**Policy**

- The United States has policy for **all weapon systems** to ensure legal and ethical employment Law of Armed Conflict: Distinction, Proportionality, & Military Necessity
- Decisions to use lethal force are made by humans at the appropriate level of command based on military result and risk
- Weapon systems employing autonomy and automation are governed under exactly the same framework of legal and ethical concerns

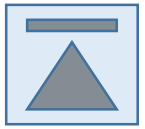
**Challenge**

- As more advanced autonomy and artificial intelligence emerges, further thought and clarification is needed to ensure the nature of the ***technology & usage*** remains legally & ethically consistent

# Navy Large Unmanned Surface and Undersea Vehicles: Background and Issues for Congress


- Congressional Research Service, 28 July 2020
- (excerpted from abstract) The Navy in FY2021 and beyond wants to develop and procure three types of large unmanned vehicles (UVs). These large UVs are called Large Unmanned Surface Vehicles (LUSVs), Medium Unmanned Surface Vehicles (MUSVs), and Extra-Large Unmanned Undersea Vehicles (XLUUVs). […] The Navy's large UV programs pose a number of oversight issues for Congress, including issues relating to the analytical basis for the more distributed fleet architecture; [… and …] potential implications for miscalculation or escalation at sea" […]

Ronald O'Rourke, Specialist in Naval Affairs



# Conclusions and Recommendations

Much progress accomplished, and many opportunities becoming possible. Comprehensive pursuit of multiple integrated capabilities is essential.

# Assessment of Current Thinking

- Human supervision of potentially lethal autonomous systems is a matter of serious global importance.
- Wide consensus is emerging on principles, aspects of the problem, elements of solutions, and need to achieve better capabilities.
- Much philosophical concern but few concrete activities are evident.

Ethical Control of Unmanned Systems project appears to provide a needed path towards practice, with the historic role of warfighting professionals more central than ever as weapons autonomy grows.

# Conclusions

- Human supervision is required for any unmanned systems holding potential for lethal force.
  - Cannot push “big red shiny AI button” and hope for best – immoral, unlawful.
  - Similar imperatives exist for supervising systems holding life-saving potential.
- Human control of unmanned systems is possible at long ranges of time-duration and distance through well-defined mission orders.
  - Readable and sharable by both humans and unmanned systems.
  - Validatable syntax and semantics through understandable logical constraints.
  - Testable and confirmable using simulation, visualization, perhaps qualification.
- Coherent human-system team approach is feasible and repeatable.
  - Semantic Web confirmation can ensure orders are comprehensive, consistent.
  - Human role remains essential for life-saving and potentially lethal scenarios.

# Recommendations for Future Work

*Continued development*

- Diverse mission exemplars
- Software implementations
- 2D, 3D visualization of results

*Future capabilities*

- Automatable testing
- Field experimentation (FX)
- “Qualification” of unmanned systems in virtual environments

*Outreach*

- Presentations, publication review
- Engagement in key ethical forums
- NPS wargame and course support
- NPS thesis and dissertation work

*Adoption*

- Support for developmental system
- Influence campaign for both C4I and robotics communities of interest

# Technical Readiness Level (TRL)

Self-assessment: have progressed from **TRL 3** to **TRL 4**. As future work adds meaningful simulation capabilities, project begins moving into **TRL 5** maturity.

https://en.wikipedia.org/wiki/Technology_readiness_level#Current_NASA_usage

The current nine-point NASA scale is:

- Level 1 – Basic principles observed and reported
- Level 2 – Technology concept and/or application formulated
- Level 3 – Analytical and experimental critical function and/or characteristic proof-of concept
- **Now:** Level 4 – Component and/or breadboard validation in laboratory environment
- **Next:** Level 5 – Component and/or breadboard validation in relevant environment
- Level 6 – System/subsystem model or prototype demonstration in a relevant environment (ground or space)
- Level 7 – System prototype demonstration in a space environment
- Level 8 – Actual system completed and "flight qualified" through test and demonstration (ground or space)
- Level 9 – Actual system "flight proven" through successful mission operations

# Upcoming events and next steps

Multiple conference and working group presentations

✓ 9-11 March 2020, US Semantic Technologies Symposium (US2TS), Raleigh NC

✓ 25 March 2020, Unmanned Vehicle and Autonomous Systems (UVAS) Working Group

X 23-25 March 2020, NDIA Undersea Warfare (USW), San Diego CA

X 25-27 March 2020, xSWARM 2020 workshop, NPS CRUSER, Monterey CA

- TBD dates Spring 2020, ASW Community of Interest (COI)
- 12-14 May 2020, Submarine Technology Symposium (STS), JHU/APL, Laurel MD

Journal paper publication candidates:

- *IEEE Journal of Ocean Engineering (JOE)*
- *IEEE AI Ethics* journal forthcoming?
- *Journal of Web Semantics* https://www.journals.elsevier.com/journal-of-web-semantics

# US Semantic Technologies Symposium (US2TS)
# Session: Hybrid AI for Context Understanding

US2TS
10-12 MAR 2020

What is nature of context that described hybrid AI system is trying to understand?

- Validate human orders to remote unmanned systems with capacity for lethal/lifesaving force.
- Ethics for Rules of Engagement (ROE), Laws of Armed Conflict (LOAC), operational constraints.

What specific methods and technologies does this hybrid AI system use, and how?

- Validatable XML mission syntax using controlled vocabularies with corresponding ontology.
- Perform SPARQL queries of RDF/OWL to check complex relationships, requirements, violations.
- Conversions for declarative orders, language implementations, Turtle triples with Protégé, ARQ.

What are current limitations in presented solution, what is plan for future work?

- Representative test cases are being tested in simulation to build out verification framework.
- Supports maritime operations with tractable Identification Friend Foe Neutral Unknown (IFFNU).
- Expand scope with larger mission sets for diverse operations by unmanned systems in real world.
- Bridge command and control (C2) with modeling and simulation (M&S) virtual environments, for
- Domain-expert qualification of hardware/software systems using tactical scenarios as unit tests.

# Acknowledgements


- This work builds upon 3 decades of inquiry by NPS faculty, staff and graduate/doctoral students serving on active duty.
- Co-investigators Dr. Curtis Blais and Dr. Robert B. McGhee. CRADA leads RADM Jerry Ellis USN (Ret.) and Julie Leeman, Raytheon Technologies.
- Research support by Raytheon Missile Systems (RMS) and Raytheon BBN Technologies (Tucson Arizona and Arlington Virginia) has been instrumental in recent progress during 2019-2020.
- Significant contributions by Hsin-Fu “Sinker” Wu, Richard Markeloff and others at Raytheon, supporting current work under CRADA with NPS.
- Work with Dr. Jakub Flotyński on X3D Ontology for Semantic Web was an essential prerequisite to the capabilities demonstrated here.
- Collaborative design efforts with numerous skilled engineers and scientists is gratefully acknowledged. Further activity is welcome.

# Contact


**Don Brutzman**

*brutzman@nps.edu*
*http://faculty.nps.edu/brutzman*

Code USW/Br, Naval Postgraduate School
Monterey California 93943-5000 USA
1.831.656.2149 work
1.831.402.4809 cell

# Contact


**Curt Blais**

*clblais@nps.edu*

home page

Code MV/Bl, Naval Postgraduate School

Monterey California 93943-5000 USA

1.831.656.3215 work


# ETHICAL MISSION DEFINITION AND EXECUTION FOR MARITIME ROBOTS UNDER HUMAN SUPERVISION

- Lethality requires ethical and legal basis, supervised by military teams.
- Executable robot tasking can resemble tactical tasking of humans afloat.
- Careful application of goal constraints makes ethical control feasible.
- Robot missions then complement and extend naval operation orders.
- Semantic Web logic can confirm ethical correctness and completeness.
- Next steps: continue 2 decades of work with realistic scenario testing.

> "Ethical constraints on robot mission execution are possible today. There is no need to wait for future developments in Artificial Intelligence (AI). It is a moral imperative that ethical constraints in some form be introduced immediately into the software of all robots that are capable of inflicting unintended or deliberate harm to humans or property." Robert McGhee, April 2016

- IEEE Journal of Oceanic Engineering (JOE) paper along with online references.
- Authors Don Brutzman, Curtis Blais, Duane Davis and Robert McGhee, NPS.
- Feedback and recommendations always welcome. Contact: brutzman@nps.edu

# Presentation abstracts

Backup slides of use in outreach sessions

# MORS 88 Abstract … and analytic underpinnings


- **Ethical control of unmanned systems can be accomplished through structured mission definitions that are consistently readable, validatable and understandable by humans and robots.**
  - Structured data models for syntactically valid, well-defined mission orders.
  - Semantic Web representations and queries to ascertain semantic correctness.
- **Modeling of scenarios and simulations that test ethical constraints can lead to visualization of outcomes.**
  - Readable by humans, conduct probabilistic analysis, Web-based visualization.
  - Repeatable in any programming language or robot dialect for scalability.
- **We foresee analytic foresight and assessment leading to human-driven qualification of systems possessing lethal/lifesaving force.**
  - All about the data!  Can we read, repeat, replay, review, remix, understand?

# Analytic questions of interest – to warfighters

*Repeatability*

- What happened? What changed? What's next? What will happen?
- What can we learn from all data + all operations? … Will we prevail?

*Understandability*

- Do I entrust this robot with authority for lethal or lifesaving force?
- Do these combined human-machine actions make sense to team?

*Trust and Verification, Validation, Accreditation (V V+A)*

- Is a robot system (hardware, software, sensors) qualified to deploy?
- What is Modeling + Simulation basis for analysis-development loop?

# NDIA 2020 Undersea Warfare Virtual Conference

https://www.ndia.org/events/2020/9/22/2020-usw-virtual-conference

National Defense Industrial Association | About | Contact | Press | National Defense Magazine | NDIA Connect | Welcome, Don MY ACCOUNT

NDIA | Meetings & Events | Policy | Membership | Divisions | Education | Chapters | Join

Home » Calendar of Events » 2020 Undersea Warfare Virtual Conference

PRINT EMAIL

## 2020 Undersea Warfare Virtual Conference


9/22/2020 – 9/23/2020

**Theme :** Integrating Combat Power with the Joint Force
**Event Type :** Virtual Meeting
**Event Code :** 0240

REGISTER

ADD TO CALENDAR

# NDIA USW 2020 Conference theme: Integrating Combat Power with the Joint Force

- Joint Force is increasingly interdepending on Undersea Warfare as conflict potential grows across the world
- Alfred Thayer Mahan: command of the sea, even if local and temporary,  naval operations in support of land forces holds decisive importance. (Wikipedia)
- Command Control Communications Intelligence (C4I) working group efforts are fundamentally important.
- USW forces must be capable of jointly coordinating C2 of systems holding direct potential for lethal force, despite intermittent communications, over long distances and long durations of time.

# Where must we go next

- Massive testing of unmanned hardware + software ability to follow both orders and constraints in  physically realistic virtual environments
- Certify capabilities via field experimentation (FX), confirmed by USW range exercises and regular force operations
- Human warfighters and commanders (not just engineers) review and approve unmanned systems as… qualified
- New normal will be human + machine teaming.  Mainstream capabilities in all aspects of acquisition and deployment.