

**AUV Common Control Language (CCL) –
A Proposed Standard Language and Framework for AUV
Monitoring & Control**

Layer 1 – CCL Vocabulary and Message Set Specification

Working Draft CCL Version 2.14
30 August 2007

Principal Authors: Rick J. Komerska and Steven G. Chappell
Autonomous Undersea Systems Institute (AUSI)



Copyright (c) 2007 Autonomous Undersea Systems Institute.

Permission is granted to copy, distribute and/or modify this document under the terms of the GNU Free Documentation License, Version 1.2 or any later version published by the Free Software Foundation; with no Invariant Sections, no Front-Cover Texts, and no Back-Cover Texts. A copy of the license is included in the section entitled "GNU Free Documentation License".

Change History

CCL Version: 2.14
Date: 30 August 2007
Author: Rick Komerska
Change: Clarified Monitor & Report inform messaging. The inform response is only sent for “completed”, non-acknowledgement type messages.

CCL Version: 2.13
Date: 9 July 2007
Author: Rick Komerska
Change: Fixed spelling errors in structure labels in text. Added change history page.

CCL Version: 2.13
Date: 15 May 2007
Author: Rick Komerska
Change: Baseline.

TABLE OF CONTENTS

SUMMARY	1
1 INTRODUCTION	1
2 CCL DESIGN	1
2.1 DESIGN REQUIREMENTS.....	3
2.2 PLATFORM REQUIREMENTS.....	3
2.3 VOCABULARY.....	3
2.3.1 <i>Generic Behaviors</i>	3
2.3.2 <i>Vocabulary Types</i>	3
2.3.3 <i>Data Structures and Basic Types</i>	4
2.3.4 <i>Parameters</i>	5
2.4 MESSAGE COMPOSITION.....	5
2.4.1 <i>Maneuver</i>	7
GoTo Waypoint Request.....	7
Follow Path Request.....	8
Speed and Bearing Request.....	8
Move Request.....	8
Station Keep Request.....	8
Maneuver Acknowledgment.....	9
2.4.2 <i>Navigate</i>	9
GPS Fix Request.....	9
Avoid Region Request.....	9
Set Path Request.....	10
Navigate Acknowledgement.....	11
2.4.3 <i>Communicate</i>	11
Message Inform.....	11
2.4.4 <i>Configure</i>	11
System Parameters Request and Inform.....	11
Sensor Request and Inform.....	12
Comms Interface Request and Inform.....	12
2.4.5 <i>Monitor and Report</i>	13
Status Request and Inform.....	14
Capabilities Request and Inform.....	15
File Request and Inform.....	16
Sensor Capabilities Request and Inform.....	16
Comms Interface Capabilities Request and Inform.....	17
2.4.6 <i>Exec Convention</i>	18
System Admin Request.....	18
Start Mission Request.....	18
Stop Mission Request.....	18
Abort Mission Request.....	18
Start Behavior Request.....	19
Stop Behavior Request.....	19
Update Behavior Request.....	19
Exec Convention Acknowledgment.....	19
3 IMPLEMENTATION	19
4 FUTURE WORK	20
REFERENCES	21
GNU FREE DOCUMENTATION LICENSE	22
APPENDIX A – PLATFORM PARAMETERS	27
APPENDIX B – VOCABULARY ENUMERATIONS	30
APPENDIX C – DATA STRUCTURES	46

Summary

The following document describes the message set and vocabulary of a “Common Control Language” (CCL) for autonomous undersea vehicle (AUV) monitoring and control. The CCL is a protocol that aims to provide a means for different autonomous vehicle and instrument platforms and their users to communicate with each other using a common set of commands and information. It is specifically tailored for autonomous platforms operating in the underwater domain.

This document focuses upon the message set and vocabulary of the language. These form the basis of a CCL layered framework which supports research and development efforts into technologies supporting multiple AUV cooperation. This framework approach provides a way for potential users within the community to adopt aspects of the language and its implementation best suited to their particular needs. Information pertaining to the other layers will be covered in separate documents.

This document is a significant update to the 1999 AUSI document “Generic Behaviors as an Interface for a Standard AUV Command & Monitoring Language” [1], which builds upon previous research work in AUV cooperative behaviors [2, 3]. It incorporates experience gained in integrating the specification onto the SAUV II [4] platform and lessons learned from subsequent field tests. This hardware integration and test effort began in 2003 and is currently ongoing.

This effort is part of a broader program to research and develop technology and systems to enable Multiple Cooperating AUVs (MCAUV) (ONR Award #N00014-04-1-0264). This is a team-based effort involving the Autonomous Undersea Systems Institute (AUSI), Naval Undersea Warfare Center – Newport (NUWC DIVNPT), Technology Systems Inc. (TSI), Falmouth Scientific Inc. (FSI), and others.

1 Introduction

An Autonomous Ocean Sampling Network (AOSN) [5] may be comprised of a number of diverse systems. Non-mobile instrument platforms may need minimal interaction to accomplish their tasks whereas mobile AUVs may need to coordinate their actions to undertake complex sampling tasks. These platforms will need to operate in a semi-autonomous or autonomous state over timeframes of days to months or possibly years. It is envisioned that within an AOSN there will exist a command and control hierarchy that will need to be capable of dynamically configuring and reconfiguring the fleet in response to events such as vehicles entering and leaving on a scheduled and unscheduled basis, and adapting mission parameters based on member platform sensor findings. Communication of capabilities, situation and other informational data between multiple heterogeneous platforms will be essential in making this paradigm work.

Toward this end, AUSI and its team members have been developing a standard language for AUV platforms for expressing high-level goals and intentions as well as distributing information. The underwater environment dictates that the language be optimized for very low bandwidth communications. It should be able to represent the various behaviors exhibited by operational AUVs. It should be capable of encapsulating domain specific information and situational context. It should be able to support different high-level control methodologies across varying hardware implementations and configurations. It should incorporate language design elements which enhance development of cooperative behaviors. Finally, it should be relatively simple and flexible to use and allow for gradual adoption by AUV developers.

2 CCL Design

The CCL design focuses on providing a standard vocabulary and grammar for inter-AUV and AUV-human communication. It draws heavily upon past work in other AUV command languages as well as AUSI work in Generic Behaviors. Early work on Generic Behaviors focused on the definition of a common interface to an AUV which would shield the low level vehicle implementation from the user and other AUVs [1, 2, 3, 6, 7]. This interface layer would provide the basis for allowing AUVs to easily communicate and exert control amongst themselves, as

well as form a basis for building higher level “aggregate behaviors” reflecting more complex missions. Out of this work came the idea of a set of “Generic Behaviors”. Fig. 1 lists the broad functional categories comprising the Generic Behaviors common to all AUVs. The level of capability for each behavior may be different for each AUV, and is defined by parameters associated with it. In different AUVs, those parameters may or may not be possible to change. The configuration of those parameters, once established, define a level of capability, relative to a specific behavior, that can be understood by all.

- | | | |
|--------------|----------------|--------------------------|
| 1. Maneuver | 4. Sense | 7. Execute (Conventions) |
| 2. Configure | 5. Communicate | 8. Acquire (Samples) |
| 3. Monitor | 6. Navigate | 9. Manipulate |

Figure 1. Generic Behavior Functional Categories

This document provides a working draft specification for the message set and vocabulary of the language. These form the foundational first layer of a CCL layered framework composed of five layers. Fig. 2 lists these layers in order of increasing platform capability and framework support. Each layer is designed to build upon the previous layer, providing ever-increasing system capabilities which leverage tools which already exist or are currently being developed. Potential users are free to use the software tools utilized by this team, or develop their own implementations, depending upon the user’s mission needs, vehicle capabilities and legacy tools.

Layer	Provides
1. Vocabulary and Message Set Specification	<ul style="list-style-type: none"> • AUV domain specific vocabulary • Message types and format • Expressive messaging context
2. CCL Support Library	<ul style="list-style-type: none"> • Implementation of vocabulary and message set • Bit-level encoding of messages optimized for low bandwidth channel • Serialization / deserialization of messages • ASCII string messaging API
3. Basic Behavior Process Set	<ul style="list-style-type: none"> • Distributed control environment (DICE) framework for managing behavior processes • Software processes to interpret CCL messages and which interface to both vehicle-specific and high-level problem solving processes
4. Mission Interpreter	<ul style="list-style-type: none"> • C language-like grammar for mission level development • Automatic generation of executable behaviors based on a mission file • Ability to update tasks in real-time • Sequential execution of behaviors • Libraries to support interpretation of mission grammar and task interface to basic behaviors
5. Cost-based Real-time Planning	<ul style="list-style-type: none"> • Adaptive re-planning which allows optimization of tasks to cope with dynamic aspects of the environment, working toward individual and potential group goals • Sequential or parallel execution of behaviors

Figure 2. CCL Framework Layers

2.1 Design Requirements

We have defined a set of top level requirements to help guide us in the development of the CCL:

1. One vehicle cannot look inside another vehicle; cooperation occurs through message-passing between the decision making levels among platforms
2. Users should be able to add their own messages if required but not expect that these new messages will be understood or achievable by all vehicles.
3. Allow for arbitrary execution of behaviors, e.g. repetition, sequential or parallel, as well as support “canned missions” and interactive tasking
4. Easily extensible to new vehicles and new missions
5. Build upon previous work on Generic Behaviors as well as other AUV and intelligent agent development efforts
6. Optimized for the AUV domain
7. Optimized to conserve transmission bandwidth

2.2 Platform Requirements

There is an implicit set of requirements for platforms that wish to implement this CCL. The most basic requirements include the following:

1. A file system
2. Capability to represent the specified set of monitorable and configurable state parameters
3. Ability to implement the functionality (to the best of platform capability) specified in the message set

The platform requirements increase as higher layers in the framework are implemented. Depending upon the tool set chosen, these requirements may include additional CPU capabilities and inclusion of an onboard C compiler.

2.3 Vocabulary

A vocabulary has been defined with data types and structures which characterize the CCL ontology. The vocabulary incorporates terms which support the use of Generic Behaviors. In addition, it also provides the handles, in the form of parameter structures, for accessing and configuring specific vehicle aspects.

2.3.1 Generic Behaviors

Generic behaviors provide a way of logically segmenting the expressed desired behavior of one platform to another platform, or group of platforms. It was expressly developed to support the types of behaviors associated with AUVs. As of this writing, the following generic behavior categories and their implied meanings are proposed:

Maneuver – “Move or relocate in this manner.”

Navigate – “Provide path constraints and positional information.”

Communicate – “Provide a conduit for command, control and information messages.”

Configure – “Change some pre-configured platform aspect.”

Monitor – “Watch some aspect of the platform or environment and notify another platform when this changes in some defined way.”

Sense – “Utilize on-board mission sensors to effectively collect environmental data.”

Exec Convention – “Carry out a well-known system-level action.”

2.3.2 Vocabulary Types

A set of vocabulary enumeration types has been defined which encapsulate many of the underwater domain-specific concepts of interest to AUV developers and users. Use of these vocabulary types helps to reduce communication bandwidth. Currently, 60 enumeration sets incorporating over 250 words have been defined; see Appendix B for a list of all types.

For development purposes, we have currently implemented the vocabulary types in the C programming language as enumeration sets. Fig. 3 shows a sample set, along with the equivalent “English gloss” text encoded as a

complimentary character string array variable. Note that this method allows for easy string parsing using the enumeration constants as input into the string array. See the document “Layer 2 – CCL Support Library” and associated software tools for an implementation of these concepts.

```
typedef enum CCL_VERT_LOCATION_ENUM {
    CCL_VERT_DEPTH,           // requires depth value [m]
    CCL_VERT_ALTITUDE,       // requires altitude value [m]
    CCL_VERT_CURRENT_DEPTH,
    CCL_VERT_CURRENT_ALTITUDE,
    CCL_VERT_CONFIGURED_DEPTH,
    CCL_VERT_CONFIGURED_ALTITUDE,
    CCL_VERT_OFFSET_PLATFORM, // requires offset value [m], +down/-up
    CCL_VERT_SURFACE,
    CCL_VERT_BOTTOM,
    CCL_VERT_LOCATION_ENUM_MAX // array bounds usage - not a language element
} CCL_VERT_LOCATION_ENUM;

const static ACHAR* CCL_VERT_LOCATION_STR[] = {
    "DEPTH",
    "ALTITUDE",
    "CURRENT_DEPTH",
    "CURRENT_ALTITUDE",
    "CONFIGURED_DEPTH",
    "CONFIGURED_ALTITUDE",
    "PLATFORM_OFFSET",
    "SURFACE",
    "BOTTOM"
};
```

Figure 3. Sample Vocabulary Enumeration Set

2.3.3 Data Structures and Basic Types

Data structures are used to provide logical segments for grouping, communicating and parsing of domain-specific commands and information. Information in the data structures is defined at the bit level. Currently, there are 61 structures which have been defined; see Appendix C for a complete list.

Communication devices used in the AUV domain such as acoustic and RF modems rely upon serial communications which have a 1 B transmission granularity. In addition, programming for most AUVs is done using higher level languages such as C and C++, whose functions also work at this 1 B level. These restrictions force us to pack the bit fields in the structures which have a granularity of 1 B. This can lead to a slight increase in transmission bandwidth, depending on how the bit fields are arranged and grouped. Although we have been careful to try and balance flexibility and reusability in the vocabulary and grammar with minimum transmission size, there is probably room for improvement here in the future.

CCL relies upon a mapping from the basic C language data types, as shown in Fig. 4. This lets us explicitly control the size of the basic data types across any compiler. This allows for very fine control over message sizes, which is important in bandwidth constrained communications links.

```
typedef unsigned char    ABYTE;      // 1 B
typedef char             ACHAR;      // 1 B
typedef short            AINT16;     // 2 B
typedef unsigned short  AUINT16;    // 2 B
typedef int              AINT32;     // 4 B
typedef unsigned int    AUINT32;    // 4 B
typedef float           AFLT32;     // 4 B
typedef double          AFLT64;     // 8 B
typedef long            ATIME32;    // 4 B
```

Figure 4. Basic Data Types

For development purposes, we have currently implemented the data structures in the C programming language as bit fields. We assume a 32-bit operating system, and define an unsigned character as 1 B, an unsigned short as 2 B and an unsigned integer as 4 B. We also assume structures can be packed along a 1 B boundary. Data type and structure names adhere to a 63 significant initial character limit. Fig. 5 shows an example of both a data type and data structure, and how they are used together. Note that in the `CCL_TIME_INTERVAL_STRUCT` structure, the **units:2** denotes that 2 bits are allocated to represent the `CCL_TIME_INTERVAL_ENUM` (defined above it), while the **time:14** denotes that 14 bits are used to store the actual time interval value. In this case, 14 bits provides a dynamic range from 0 to 16,383, which implies we can use this structure to represent time intervals from 0 – 4.5 hours at 1 second resolution, 0 – 273.1 hours at 1 minute resolution, 0 – 682.6 days at 1 hour resolution, or 0 – 44.9 years at 1 day resolution.

```
typedef enum CCL_TIME_INTERVAL_ENUM {
    CCL_TIME_SEC,           // "SECONDS"
    CCL_TIME_MIN,          // "MINUTES"
    CCL_TIME_HRS,          // "HOURS"
    CCL_TIME_DAYS          // "DAYS"
} CCL_TIME_INTERVAL_ENUM;

typedef struct {
    AUINT16 units:2;       // 2 B
                          // CCL_TIME_INTERVAL_ENUM
    AUINT16 time:14;      // 0...16K range
} CCL_TIME_INTERVAL_STRUCT; // Time interval in specified units
```

Figure 5. Example Data Type and Structure Sets

2.3.4 Parameters

Parameters are used as reference keys or handles to access platform information and configuration structures. The specification supports a maximum of 128 well-known parameters and 128 user-defined parameters. Parameters within both the well-known and user-defined sets are further divided into the set of monitorable parameters and the set of configurable parameters. Monitorable parameters can be thought of as platform state variables, and are represented using simple data types. These parameters are used in Monitor and Report messages when specifying triggered events. They are periodically updated within the platform system, and are not configurable outside of the system. Configurable parameters, conversely, are meant to be adjusted by the operator or other governing platforms to specifically alter the vehicle’s behavior. They can be composed of either simple or complex data structures.

Currently, 117 of the 128 well-known parameters are used. Appendix A lists these parameters, their associated data structure size, units, enumeration constant, support for derivative monitoring/reporting and whether the parameter is monitorable or configurable. See Appendix C for definition of these structures.

2.4 Message Composition

The exchange of commands and information among platforms borrows ideas for software agent communications documented in the FIPA Communicative Act Library Specification [8], as well as earlier work in AUV conceptual communication language development [9]. In the present design, the CCL is used to express message context in two broad ways:

- **Command** or imperative message (*Request, Urgent Request, Command*) directing the platform to perform a certain behavior, or
- **Inform** message (*Inform, Warn, Urgent Warn*), propagating data, information and knowledge throughout the system.

Each CCL message is composed of an ordered set of structures which include a header section, optional header support data structures, and a variable sized body section. The grammar is designed so a parser does not need to backtrack, a trait of popular LaLR grammars. Addressing and transmission framing of the messages is assumed to be handled separately and is outside the scope of this work.

Each message is constructed using the structures shown in Fig. 6. Note that Backus-Nauer conventions are used to describe the grammar:

- [] denotes an optional data structure
- | denotes a logical ‘OR’
- * denotes repetition

	Structures	Size	
Headers	CCL_HDR_STRUCT.....	3 B	
	CCL_REQ_HDR_STRUCT.....	1 B	<i>Request Options</i>
	or CCL_INF_HDR_STRUCT.....	1 B	<i>Inform Options</i>
Request Options Support Data	[CCL_AUTHORITY_STRUCT].....	1 B	<i>Authority</i>
	[CCL_LOOP_STRUCT]..... [ATIME32 CCL_TIME_INTERVAL_STRUCT].....	1 B 4 or 2 B	<i>Looping</i>
Inform Options Support Data	[CCL_ADDRESS_STRUCT].....	1 B	<i>Solicitor</i>
Body	[additional support structures].....	1-n B	

Figure 6. CCL Message Composition and Size

Every CCL message contains a set of header structures which appear at the front of the message, followed by a set of optional header support data structures. An imperative message will contain a CCL_HDR_STRUCT followed by a CCL_REQ_HDR_STRUCT structure. An informative message will contain a CCL_HDR_STRUCT followed by a CCL_INF_HDR_STRUCT structure. The header structures include the following attributes:

CCL_HDR_STRUCT

- Message context (Request, Urgent Request, Command, Inform, Warn, Urgent Warn)
- Generic Behavior type (Maneuver, Navigate, Communicate, Configure, Monitor, Exec Convention)
- Generic Behavior subtype (GoTo Waypoint, Speed and Bearing, Station Keep, GPS Fix, ...)
- Conversation thread ID
- String API support indicator

CCL_REQ_HDR_STRUCT

- Request for acknowledgment
- Authority indicator
- Multiple CCL grouping indicators
- Looping indicators
- Behavior stacking

CCL_INF_HDR_STRUCT

- Informational / acknowledgment type and reasons

The request and inform options support data structures are used to provide additional information depending on what is specified in the CCL_REQ_HDR_STRUCT and CCL_INF_HDR_STRUCT structures, respectively.

The authority structure is used to indicate the level of authority of the sender. It is composed of two fields; an ‘authority level’ with a range of 0-7, and an ‘intra-level pecking order’ with a range from 0-31. Using this ranking structure, a platform having a higher authority level may choose to reject a request from a platform with a lower level.

The looping parameters provide a simple means of grouping and repeating a set of CCL behaviors. The number of times as well as time interval between repetitions can also be specified. In addition, scheduling parameters for a group are included only in the first message of the group, thereby reducing required transmission bandwidth.

Behavior stacking allows for a message request to be queued for execution after the last similar request, or to replace all similar queued and/or currently executing requests. For example, requesting that a maneuver be queued will ensure that it will be executed after all other queued maneuvers complete. Conversely, requesting a maneuver be replaced will stop the execution of any current maneuver, and clear any maneuvers in the queue, then execute the requested maneuver.

A single CCL or set of CCL messages must be read and parsed all at once. Behaviors are triggered sequentially upon receipt of a command message.

The solicitor address may be specified in an inform message and provides the address of the agent which requested the information contained in the message.

Following a request or inform header and support structures, the CCL message contains specific message content. The message content conforms to the following layout: a required options data structure followed by a set of optional support structures. The information contained within the options data structure indicates which support structures are to follow. Note that units for data values are defined at the supporting structure level.

The following sections detail these message types.

2.4.1 Maneuver

This is the primary interface for commanding the platform to move or relocate itself to a new position. It contains provisions to express one of five specific message types: *GoTo Waypoint*, *Follow Path*, *Speed and Bearing*, *Move*, and *Station Keep*.

When the CCL message is specified in the imperative mode (i.e. *Command*, *Request* or *Urgent Request*), the following behaviors and structures are specified. For each of these requests, the requestor may ask for an acknowledgment that the recipient has received and understands the request.

GoTo Waypoint Request

In this behavior, the platform will attempt to maneuver to the specified latitude and longitude or positional offset from the current position or preconfigured origin, and depth or altitude along the specified path at the specified speed. Transition to the next maneuver behavior occurs when the platform believes it has attained the target destination (within the pre-configured waypoint radius and waypoint depth tolerance settings), when the specified or pre-configured timeout occurs, or if explicitly stopped by some other command.

CCL_GOTO_OPT_STRUCT.....	2 B	options
[CCL_LAT_LNG_STRUCT CCL_NE_STRUCT].....	8 B	surface location
[AFLT32].....	4 B	depth/altitude [m]
[CCL_POS_TOL_STRUCT].....	2 B	radius tolerance
[CCL_POS_TOL_STRUCT].....	2 B	depth tolerance
[AUINT16].....	2 B	speed [cm/s]
[ATIME32 CCL_TIME_INTERVAL_STRUCT].....	4 or 2 B	timeout

Figure 7. GoTo Waypoint Message Composition and Size

Follow Path Request

In this behavior, the platform will attempt to maneuver to follow a preconfigured multiple segment waypoint path. The maneuver is finished after the last path segment is complete, when the specified or pre-configured timeout occurs, or if explicitly stopped by some other command. A path can be defined through the Navigate:SetPath message or from a pre-existing path file and key.

CCL_FOLLOW_PATH_OPT_STRUCT.....	1 B	options
[ACHAR array].....	1-15 B	path key label
[ATIME32 CCL_TIME_INTERVAL_STRUCT].....	4 or 2 B	timeout

Figure 8. Follow Path Message Composition and Size

Speed and Bearing Request

In this behavior, the platform will attempt to move at the specified speed and compass bearing for a given duration. The vehicle will end this behavior when the specified or pre-configured timeout duration occurs, or if explicitly stopped by some other command. Note that this behavior is particularly suited for torpedo-shaped AUVs.

CCL_SPD_AND_BRNG_OPT_STRUCT.....	2 B	options
AUINT16.....	2 B	true bearing [tenths of deg]
[AUINT16].....	2 B	speed [cm/s]
[AFLT32].....	4 B	depth/altitude [m]
[CCL_TIME_INTERVAL_STRUCT].....	2 B	time duration

Figure 9. Speed and Bearing Message Composition and Size

Move Request

The platform will attempt to attain a specific velocity (speed and direction) while maintaining a fixed orientation. If speed is set to zero, the platform will just attempt to orient itself in place. These behavior parameters need to be carefully specified to prevent under constrained, over constrained or nonsensical intended motion. The vehicle will end this behavior when the specified or pre-configured timeout duration occurs, or if explicitly stopped by some other command. Note that this behavior is particularly suited for platforms possessing 6-DOF control capabilities.

CCL_MOVE_OPT_STRUCT.....	1 B	options
[AUINT16].....	2 B	compass heading [tenths of deg]
[AINT16].....	2 B	pitch [+/- tenths of deg]
[AINT16].....	2 B	roll [+/- tenths of deg]
[AINT16].....	2 B	n/s [+/- cm/s]
[AINT16].....	2 B	e/w [+/- cm/s]
[AINT16].....	2 B	bottom/surf [+/- cm/s]
[CCL_TIME_INTERVAL_STRUCT].....	2 B	time duration

Figure 10. Move Message Composition and Size

Station Keep Request

The platform will attempt to remain within a circle at constant depth while minimizing energy usage. If the platform drifts beyond the circle radius or +/- depth bounds, it will attempt to move back to the circle center at the pre-

configured station keep speed. If position parameters are specified, the platform will consider these to be the circle center; otherwise the center is set at the current platform position. If orientation parameters are specified, the platform shall attempt to maintain the indicated orientation during this behavior. The vehicle will end this behavior when the specified or pre-configured timeout occurs, or if explicitly stopped by some other command.

CCL_STATION_KEEP_OPT_STRUCT.....	2 B	options
[CCL_LAT_LNG_STRUCT CCL_NE_STRUCT].....	8 B	surface location
[AFLT32].....	4 B	depth/altitude [m]
[CCL_POS_TOL_STRUCT].....	2 B	drift radius tolerance
[CCL_POS_TOL_STRUCT].....	2 B	drift depth tolerance
[AUINT16].....	2 B	compass heading fix [tenths of deg]
[AINT16].....	2 B	pitch fix [+/- tenths of deg]
[AINT16].....	2 B	roll fix [+/- tenths of deg]
[ATIME32 CCL_TIME_INTERVAL_STRUCT].....	4 or 2 B	timeout

Figure 11. Station Keep Message Composition and Size

Maneuver Acknowledgment

When acknowledging a request to maneuver, a platform may send a message using the informative mode (i.e. *Inform*, *Warn* or *Urgent Warn*). The relevant information is contained in the CCL_HDR_STRUCT, CCL_INF_HDR_STRUCT and inform options support structures.

2.4.2 Navigate

This is the primary interface for specifying path constraints and position updates for the platform as it maneuvers. It contains provisions to express one of three message types: *GPS Fix*, *Avoid Region* and *Set Path*. For each of these requests, the requestor may ask for an acknowledgment that the recipient has received and understands the request.

GPS Fix Request

The platform is requested to update its geo-referenced position using GPS navigation sensor. The platform will wait the specified period to acquire the GPS fix. The activity is complete when the position fix has been acquired, the timeout occurs, or if explicitly stopped by some other command. Note that this behavior does not generate a supporting maneuver behavior. The following structures are specified for a GPS Fix request:

CCL_GPS_FIX_REQ_OPT_STRUCT.....	1 B	options
[CCL_TIME_INTERVAL_STRUCT].....	2 B	time duration

Figure 12. GPS Fix Command Message Composition and Size

Avoid Region Request

The platform is requested to start avoiding or stop avoiding a designated region, specified as a sphere, a vertically-aligned cylinder, or a rectangular volume. Once set, avoidance regions remain in effect until explicitly cleared, using the “stop” or “stop all” behavior messages. Multiple regions can be superimposed, providing the ability to specify more elaborate avoidance regions.

Note that rectangular regions are aligned along north/south (x direction) and east/west (y direction) lines, before any angle transformation is applied. If height is specified, it refers to total z distance (e.g. height/2 above and height/2

below z location). Rotation takes place about the reference xy point. In addition, the region and location structures are only used when the “start” or “stop” behavior actions are specified; they are not needed for the “stop all” action.

CCL_AVOID_REG_REQ_OPT_STRUCT.....	3 B	options
[CCL_LAT_LNG_STRUCT CCL_NE_STRUCT].....	8 B	ref center surface location
[AFLT32].....	4 B	ref center depth [m]
[CCL_POS_TOL_STRUCT].....	2 B	x extent or diameter
[CCL_POS_TOL_STRUCT].....	2 B	y extent
[CCL_POS_TOL_STRUCT].....	2 B	z extent

Figure 13. Avoid Region Request Message Composition and Size

Set Path Request

The platform is requested to establish a preconfigured multiple segment path. The path can be specified directly with a path segment set or through reference to an onboard file. Attributes including xy location, z location, and speed can be specified explicitly or using configured parameters and on a per-path or per path segment basis. Radius and depth tolerances are set as configured parameters.

If the path key label is used, that path must be saved on the file system. A separate file is required for each path key reference. (This requirement is meant to reduce the impact on memory resources and shift it to disk resources.) The path is executed through the Maneuver:FollowPath behavior by specifying either the default configured path or through a path key label.

CCL_SET_PATH_OPT_STRUCT.....	5 B	options
[CCL_LAT_LNG_STRUCT CCL_NE_STRUCT].....	8 B	fixed surface location, if not specifying per-segment xy
[AFLT32].....	4 B	fixed depth/altitude [m], if not specifying per-segment z
[AUINT16].....	2 B	fixed speed [cm/s], if not specifying per-segment speed
{		
CCL_LOC_SPEC_STRUCT.....	1 B	specify xy and/or z location of end waypoint in path segment
[CCL_LAT_LNG_STRUCT CCL_NE_STRUCT].....	8 B	surface location, if specifying per-segment xy and as needed
[AFLT32].....	4 B	depth/altitude [m], if specifying per-segment z and as needed
[AUINT16].....	2 B	speed [cm/s], if specifying per-segment speed
}*	path segment set (31 segments max)	
[ACHAR array].....	1-15 B	file name
[ACHAR array].....	1-31 B	directory name
[ACHAR array].....	1-15 B	path key label
[ATIME32 CCL_TIME_INTERVAL_STRUCT].....	4 or 2 B	timeout

Figure 14. Set Path Request Message Composition and Size

Navigate Acknowledgement

When acknowledging a navigate request, a platform may send a message using the informative mode. The relevant information is contained in the CCL_HDR_STRUCT, CCL_INF_HDR_STRUCT and inform options support structures.

2.4.3 Communicate

This is the primary interface for implementing the sending and receiving of messages between platforms. There are currently no message request types specified. As for message inform types, *Message* is used as a container for unsolicited inform type communications.

This may be used in the future to specify enabling/disabling gateway mode and/or acomms ranging mode.

Message Inform

A message is used to convey unsolicited information between platforms. A message may contain user-specific information or CCL messages, in ASCII or binary format. The message content size is limited to 255 B. In addition to the message contents, this message contains information as to why it was generated and who the sender was.

If the message is ASCII text, the NULL terminator does not need to be included.

CCL_MSG_INF_OPT_STRUCT.....	2 B	options
[ABYTE array].....	1-255 B	message

Figure 15. Message Inform Message Composition and Size

2.4.4 Configure

This is the primary interface for specifying how the platform should reconfigure aspects of itself. It contains provisions to express one of three message types: *System Parameters*, *Sensor* and *Comms Interface*, reflecting the notion that key internal aspects of the platform can be accessed and modified. Not all parameters are configurable; Appendix A lists all parameters and the subset of that which is configurable. For each of these requests, the requestor may ask for an acknowledgment that the recipient has received and was able to carry out the configure action.

System Parameters Request and Inform

The platform is requested to modify the requested parameter values defined in the parameter list configurable subset. Note that parameters can be configured from either the well-known CCL list or a user-defined custom list. Up to 15 separate parameters may be modified through a single request message.

CCL_CFG_PARAMS_REQ_OPT_STRUCT.....	1 B	options
{		
CCL_PARAM_ID_STRUCT.....	1 B	parameter ID
[ABYTE].....	1 B	for custom parameters, size of following data structure
parameter structure.....	1-32 B	data
}*	15 configurable parameters max	

Figure 16. System Parameters Configuration Request Message Composition and Size

A platform uses the following structures to provide extended failure information from a request to configure one or more parameters. This message contains information as to why it was generated and who the requestor was. It also contains the identifiers for those parameters which were requested to be configured, as well as an error code for each parameter that was unable to be configured. Up to 15 parameters may be reported on in a single inform message. Note that these structures are only sent if a configuration attempt failure occurs.

```

[
    CCL_CFG_PARAMS_INF_OPT_STRUCT.....1 B      for failure only
                                                options
    {
        CCL_PARAM_ID_STRUCT.....1 B          parameter ID
        CCL_PARAM_ACCESS_INFO_STRUCT.....1 B   status
    }*
                                                15 configurable parameters max
]

```

Figure 17. System Parameters Configuration Inform Message Composition and Size

Sensor Request and Inform

The platform is requested to modify the specified attributes of the indicated mission sensor. Examples of these attributes include device control state (on/off/sleep/standby), sample rate, and sensitivity threshold. Mission sensors are categorized into *types*, of which up to three instances of each type may be addressed. Valid types are defined in the CCL_MISSION_SENSOR_ENUM set.

```

    CCL_SENSOR_OPT_STRUCT.....1 B          sensor type/instance
    CCL_SENSOR_CFG_STRUCT.....1 B          sensor configuration
                                                data

```

Figure 18. Sensor Configuration Request Message Composition and Size

This message is used for providing extended failure information from a request to change attributes of a mission sensor. These data structures are only sent if the request fails.

```

[
    CCL_SENSOR_OPT_STRUCT.....1 B          for failure only
                                                sensor type/instance
    CCL_CFG_SENSOR_FAIL_STRUCT.....1 B     sensor configuration
                                                failure information
]

```

Figure 19. Sensor Configuration Inform Message Composition and Size

Comms Interface Request and Inform

The platform is requested to modify the specified attributes of the indicated communications interface. Examples of these attributes include device control state (on/off/sleep/standby), power level and baud rate. Interfaces are categorized into *types*, of which up to three instances of each type may be addressed. Valid types are defined in the CCL_COMM_INTERFACE_ENUM set.

```

    CCL_COMMS_IF_OPT_STRUCT.....1 B        comms interface
                                                type/instance
    CCL_COMMS_IF_CFG_STRUCT.....1 B        comms interface
                                                configuration data

```

Figure 20. Comms Interface Configuration Request Message Composition and Size

This message is used for providing extended failure information from a request to change attributes of a communications interface. These data structures are only sent if the request fails.

[for failure only
	CCL_COMMS_IF_OPT_STRUCT.....1 B	comms interface type/instance
	CCL_CFG_COMMS_IF_FAIL_STRUCT.....1 B	comms interface configuration failure information
]		

Figure 21. Comms Interface Configuration Inform Message Composition and Size

2.4.5 Monitor and Report

This is the primary interface for specifying how the platform should watch and record some aspect of itself or environment and report these aspects to others. It contains provisions to express one of five message types: *Status*, *Capabilities*, *File*, *Sensor Capabilities* and *Comms Interface Capabilities*. For each of these requests, the requestor may ask for an acknowledgment that the recipient has received and was able to carry out the monitor action.

Each monitor behavior request begins with a schedule block and supporting data structures, indicating the trigger to use to initiate the report generation. The requestor can also provide a “report to” address for delivery of the resulting report.

	CCL_MON_SCHEDULE_STRUCT.....2 B	options
	[ATIME32].....4 B	start time
	[ATIME32].....4 B	end time
	[CCL_REGION_STRUCT].....20 B	“space”
	or	
[
	CCL_PARAM_TRIGGER_HDR_STRUCT.....1 B	trigger settings
	CCL_PARAM_ID_STRUCT.....1 B	event trigger
	[ABYTE].....1 B	monitorable parameter for custom parameter only: byte size of parameter structure, 1...32 B
	parameter structure.....1-32 B	threshold value event trigger
	[CCL_TIME_INTERVAL_STRUCT].....2 B	timebase average for derivative calculations
]*	up to 3 monitorable parameter-based	“events”
	[CCL_TIME_INTERVAL_STRUCT].....2 B	periodic update interval
	[CCL_ADDRESS_STRUCT].....1 B	report to address

Figure 22. Monitor and Report Request Message Trigger Header Composition and Size

Monitoring and reporting begins at the requested start time and remains active until the behavior is explicitly stopped by this or some other command, or if this is a one-time report and a report has already been generated, or (if specified) the behavior end time has been reached.

For time-based scheduling, a report is generated once at the requested start time then periodically at the specified or configured time interval. For location-based scheduling, a report is generated when the platform is inside (or outside, depending on the included CCL_REGION_STRUCT setting) the specified region. Additional reports are generated periodically as specified. For parameter-based events, a report is generated when the specified event set evaluates to “true”. Specified relational operators are used to combine the parameter events for this evaluation. The parameter structure threshold value should match the parameter CCL_PARAM_ID_STRUCT data type for 0th order derivative; for 1st and 2nd derivative types an AFLT32 type should be used.

Status Request and Inform

The platform is requested to (for ‘start behavior’) transmit its current core status block and up to 15 additional monitorable or configurable parameters, based on a monitored trigger or (for ‘stop behavior’) explicitly stop reporting. Note that reporting of 1st or 2nd derivative values can only occur on monitorable parameters.

“monitor”:

Message trigger header structures shown in Fig. 22. These are required for *all* Monitor and Report request context messages.

“and report”:

```

[
    for 'start behavior' requests only
    CCL_STATUS_REQ_OPT_STRUCT.....1 B    options
    [
        ABYTE.....1 B    CCL_VALUE_DERIVATIVE
                           _ENUM, for reporting
                           of extended parameter
                           value or derivative -
                           requires CCL TIME
                           _INTERVAL_STRUCT for
                           1st or 2nd derivative
                           types
        CCL_PARAM_ID_STRUCT.....1 B    report parameter
        [CCL_TIME_INTERVAL_STRUCT].....2 B    timebase average for
                                                report parameter
                                                derivative
                                                calculations
    ]*
    up to 15 monitorable/configurable parameters
]

```

Figure 23. Monitor and Report Status Request Message Composition and Size

A platform responding to a status request uses the format shown in Fig. 24. This message contains information as to why it was generated and who the requestor was, as well as the platform status. The core status structure includes high-level indicators of subsystem status, current behavior, platform orientation, position, speed, navigation strategy, GPS signal type, and available energy. Information from up to 15 parameters is also returned, if requested and available. The inform response is only sent for “completed”, non-acknowledgement type messages.

```

[
    CCL_STATUS_INF_OPT_STRUCT.....1 B      for completed requests only
                                         options
    CCL_STATUS_STRUCT.....31 B           core status

    [
        CCL_PARAM_ID_STRUCT.....1 B      parameter ID
        CCL_PARAM_ACCESS_INFO_STRUCT.....1 B      status
        ABYTE.....1 B                    CCL_VALUE_DERIVATIVE
                                         _ENUM, for reporting
                                         of extended parameter
                                         value or derivative -
                                         requires CCL_TIME
                                         _INTERVAL_STRUCT for
                                         1st or 2nd derivative
                                         types
        [
            [ABYTE].....1 B              if access success:
                                         for custom parameter,
                                         size of following data
                                         structure
            parameter structure.....1-32 B      parameter value
            [CCL_TIME_INTERVAL_STRUCT].....2 B  timebase average for
                                         report derivative
                                         calculations
        ]
    ]*
]
                                         repeat for each extended parameter

```

Figure 24. Monitor and Report Status Inform Message Composition and Size

Capabilities Request and Inform

The platform is requested to (for ‘start behavior’) transmit its current system capabilities, based on a monitored trigger, or (for ‘stop behavior’) explicitly stop reporting.

“monitor”:

Message trigger header structures shown in Fig. 22.

“and report”:

No additional structures required.

Figure 25. Monitor and Report Capabilities Request Message Composition and Size

A platform responding to a capabilities request uses the following format. This message contains information as to why it was generated and who the requestor was, as well as the platform capabilities. The capabilities information structure includes platform identity, operational role and range, platform type and size, control properties, maximum/minimum/cruise speeds, energy system type, maximum/available/reserve energy, communications and navigation capabilities, networking protocol, and on-board mission sensors. The inform response is only sent for “completed”, non-acknowledgement type messages.

```

[
    for completed requests only
    CCL_CAPS_STRUCT.....63 B      capabilities
]

```

Figure 26. Monitor and Report Capabilities Inform Message Composition and Size

File Request and Inform

The platform is requested to (for ‘start behavior’) transmit the requested file, based on a monitored trigger, or (for ‘stop behavior’) explicitly stop reporting. The file name is provided and, optionally, a directory path where to search for the file. File and path names do *not* include a terminating NULL character.

“monitor”:

Message trigger header structures shown in Fig. 22.

“and report”:

```

[
    for 'start behavior' requests only
    CCL_FILE_REQ_OPT_STRUCT.....2 B      options
    ACHAR array.....1-15 B      file name
    [ACHAR array].....1-31 B      path
]

```

Figure 27. Monitor and Report File Request Message Composition and Size

A platform responding to a request for a file uses the following format. This message contains information as to why it was generated and who the requestor was, and if found, the requested file. Currently, the maximum file size permitted is 511 B. A reason code is provided if the file is not returned. File name does not include terminating NULL character. The inform response is only sent for “completed”, non-acknowledgement type messages.

```

[
    for completed requests only
    CCL_FILE_INF_OPT_STRUCT.....3 B      options
    ACHAR array.....1-15 B      file name
    [ABYTE array].....1-511 B      file contents
]

```

Figure 28. Monitor and Report File Inform Message Composition and Size

Sensor Capabilities Request and Inform

The platform is requested to (for ‘start behavior’) report on a specific sensor or sensors capabilities, based on a monitored trigger, or (for ‘stop behavior’) explicitly stop reporting.

“monitor”:

Message trigger header structures shown in Fig. 22.

“and report”:

```

[
    for 'start behavior' requests only
    CCL_SENSOR_CAPS_REQ_OPT_STRUCT.....1 B      sensor type/instances
]

```

Figure 29. Monitor and Report Sensor Caps Request Message Composition and Size

A platform responding to a request for mission sensor or sensors capability uses the following format. This message contains information as to why it was generated and who the requestor was, as well as the mission sensor capability. This message can contain up to 7 sets of capabilities; if more were requested, they can follow in additional inform messages (using same message context thread id). The inform response is only sent for “completed”, non-acknowledgement type messages.

```

[
    CCL_SENSOR_CAPS_INF_OPT_STRUCT.....1 B      for completed requests only
                                                options
    [
        CCL_SENSOR_CAPS_STRUCT.....8 B          sensor capability
        [ACHAR array].....1-15 B              sensor label
    ]*
]
up to 7 sets

```

Figure 30. Monitor and Report Sensor Caps Inform Message Composition and Size

Comms Interface Capabilities Request and Inform

The platform is requested to (for ‘start behavior’) report on a specific communication interface capabilities, based on a monitored trigger, or (for ‘stop behavior’) explicitly stop reporting.

“monitor”:

Message trigger header structures shown in Fig. 22.

“and report”:

```

[
    CCL_COMMS_CAPS_REQ_OPT_STRUCT.....1 B      for 'start behavior' requests only
                                                communication
                                                interface type
                                                /instances
]

```

Figure 31. Monitor and Report Communications Interface Capabilities Request Message Composition and Size

A platform responding to a request for communications interface capabilities uses the following format. This message contains information as to why it was generated and who the requestor was, as well as the interface capability. This message can contain up to 7 sets of capabilities; if more were requested, they can follow in additional inform messages (using same message context thread id). The inform response is only sent for “completed”, non-acknowledgement type messages.

```

[
    CCL_COMMS_CAPS_INF_OPT_STRUCT.....1 B      for completed requests only
                                                options
    [
        CCL_COMMS_CAPS_STRUCT.....7 B          comms interface
                                                capability
        [ACHAR array].....1-15 B              interface label
    ]*
]
up to 7 sets

```

Figure 32. Monitor and Report Communications Interface Capabilities Inform Message Composition and Size

2.4.6 Exec Convention

This is the primary interface for specifying how the platform should carry out a well-known action. It contains provisions to express one of seven message types: *System Admin*, *Start Mission*, *Stop Mission*, *Abort Mission*, *Start Behavior*, *Stop Behavior*, and *Update Behavior*. The platform may also send an acknowledgment message.

System Admin Request

The platform is requested to alter the specified system administrative aspects. This includes shutting down and restarting the high-level control applications, and shutting down and restarting the platform.

```
CCL_SYS_ADM_REQ_OPT_STRUCT.....1 B      options
[ATIME32 | CCL_TIME_INTERVAL_STRUCT].....4 or 2 B      restart time
```

Figure 33. System Admin Request Message Composition and Size

Start Mission Request

The platform is requested to begin executing the indicated mission.

```
CCL_START_MISS_REQ_OPT_STRUCT.....1 B      options
ACHAR array.....1-15 B      mission name
```

Figure 34. Start Mission Request Message Composition and Size

Stop Mission Request

The platform is requested to stop executing the current mission.

```
No extra fields required.
```

Figure 35. Stop Mission Request Message Composition and Size

Abort Mission Request

The platform is requested to stop executing the current mission and begin executing either the pre-configured or indicated abort mission.

```
CCL_ABORT_MISS_REQ_OPT_STRUCT.....1 B      options
[ACHAR array].....1-15 B      abort mission name
```

Figure 36. Abort Mission Request Message Composition and Size

Start Behavior Request

The platform is requested to begin executing the indicated behavior.

CCL_START_BEH_REQ_OPT_STRUCT.....	1 B	options
ACHAR array.....	1-15 B	behavior name

Figure 37. Start Behavior Request Message Composition and Size

Stop Behavior Request

The platform is requested to stop executing the indicated behavior.

CCL_STOP_BEH_REQ_OPT_STRUCT.....	1 B	options
ACHAR array.....	1-15 B	behavior name

Figure 38. Stop Behavior Request Message Composition and Size

Update Behavior Request

The platform is requested to modify the indicated behavior. This can be done using modifications contained in the attached byte string or through the indicated on-board file. This message is useful for when the sender knows enough about the named behaviors on the receiving platform to be able to safely control them.

CCL_UPDATE_BEH_REQ_OPT_STRUCT.....	3 B	options
ACHAR array.....	1-15 B	behavior name
[
ACHAR array.....	1-15 B	file name
[ACHAR array].....	1-31 B	path
]		
or		
[BYTE array].....	1-511 B	update code

Figure 39. Update Behavior Request Message Composition and Size

Exec Convention Acknowledgment

When acknowledging a request to execute a convention, a platform may send a message using the informative mode (i.e. *Inform*, *Warn* or *Urgent Warn*). The relevant information is contained in the CCL_HDR_STRUCT, CCL_INF_HDR_STRUCT and inform options support structures.

3 Implementation

A support library has been developed which comprises an implementation of the enumerated vocabulary and bit-mapped message data structures in the C language. In addition, this library also supports serialization and deserialization of messages for transmission through low bandwidth media. This library is currently available for Windows or Linux platform operating systems. See the related CCL framework architecture document “Layer 2 – CCL Support Library” for more detail.

4 Future Work

Some ideas for future specification modifications include:

- Explore more aspects of cooperative agent communications, such as reply-by time, developed within FIPA.
- Explore command authority in more detail. At present, commands issued with a given authority can only be overridden by a message with a higher authority.
- Look at specifying relative reference frames; subsequent positional commands may be able to be made smaller if both north and east do not need to be specified.

In addition, we plan on taking another look at the present state of AUV command languages to see how enhanced technology and greater operational experience has altered the way which we interact with these vehicles.

More work needs to be done in examining research conducted outside of the AUV community, particularly the Joint Architecture for Unmanned Systems (JAUS) framework and message set. Much has been done in the study of interoperability among heterogeneous UAV platforms and operational tools have been developed to meet their needs. Knowledge from these efforts should be integrated into the development of the CCL.

References

- [1] R.J. Komerska, S.G. Chappell, L. Peng, and D.R. Blidberg, "Generic Behaviors as an Interface for a Standard AUV Command and Monitoring Language, Working Draft Version A3," AUSI Technical Report 9904-01, Sept. 1999.
- [2] D.R. Blidberg, "Generic Behaviors: Definition and Structure, An Approach to Modularity in Intelligent System Control Architectures – Volume 1: Technical Proposal," Proposal for SOL BAA #94-19, July 21, 1994.
- [3] D.R. Blidberg, "Cooperative Distributed Problem Solving for Controlling Semi-autonomous and Autonomous Oceanographic Sampling Networks - Phase II," ONR BAA #97-021, July 1, 1997.
- [4] J. Jalbert, J. Baker, J. Duchesney, P. Pietryka, W. Dalton, D.R. Blidberg, S.G. Chappell, R. Nitzel, and K. Holappa, "Solar-Powered Autonomous Underwater Vehicle Development," in *Proceedings of the 13th International Symposium on Unmanned Untethered Submersible Technology*, Aug. 2003.
- [5] T.B. Curtin, J.G. Bellingham, J. Catipovic, and D. Webb, "Autonomous Oceanographic Sampling Networks," *Oceanography*, vol. 6, num 3, pp. 86-94, 1993.
- [6] R.J. Komerska, D. R. Blidberg, S.G. Chappell, and L. Peng, "Progress in the Development and Evaluation of a Standard AUV Command and Monitoring Language," in *Proceedings of the 11th International Symposium on Unmanned Untethered Submersible Technology*, Aug. 1999.
- [7] C.N. Duarte, G.R. Martel, C. Buzzell, D. Crimmins, R.J. Komerska, S.S. Mupparapu, S.G. Chappell, D.R. Blidberg, and R. Nitzel, "A Common Control Language to Support Multiple Cooperating AUVs," in *Proceedings of the 14th International Symposium on Unmanned Untethered Submersible Technology*, Aug. 2005.
- [8] Foundation for Intelligent Physical Agents. See <http://www.fipa.org/specs/fipa00061/SC00061G.html>.
- [9] E.H. Turner and S.G. Chappell, "Conceptual Communications for Multi-vehicle Systems," University of New Hampshire Technical Report #95-08, May 30, 1995.

GNU Free Documentation License

GNU Free Documentation License
Version 1.2, November 2002

Copyright (C) 2000,2001,2002 Free Software Foundation, Inc.
51 Franklin St, Fifth Floor, Boston, MA 02110-1301 USA

Everyone is permitted to copy and distribute verbatim copies of this license document, but changing it is not allowed.

0. PREAMBLE

The purpose of this License is to make a manual, textbook, or other functional and useful document "free" in the sense of freedom: to assure everyone the effective freedom to copy and redistribute it, with or without modifying it, either commercially or noncommercially. Secondly, this License preserves for the author and publisher a way to get credit for their work, while not being considered responsible for modifications made by others.

This License is a kind of "copyleft", which means that derivative works of the document must themselves be free in the same sense. It complements the GNU General Public License, which is a copyleft license designed for free software.

We have designed this License in order to use it for manuals for free software, because free software needs free documentation: a free program should come with manuals providing the same freedoms that the software does. But this License is not limited to software manuals; it can be used for any textual work, regardless of subject matter or whether it is published as a printed book. We recommend this License principally for works whose purpose is instruction or reference.

1. APPLICABILITY AND DEFINITIONS

This License applies to any manual or other work, in any medium, that contains a notice placed by the copyright holder saying it can be distributed under the terms of this License. Such a notice grants a world-wide, royalty-free license, unlimited in duration, to use that work under the conditions stated herein. The "Document", below, refers to any such manual or work. Any member of the public is a licensee, and is addressed as "you". You accept the license if you copy, modify or distribute the work in a way requiring permission under copyright law.

A "Modified Version" of the Document means any work containing the Document or a portion of it, either copied verbatim, or with modifications and/or translated into another language.

A "Secondary Section" is a named appendix or a front-matter section of the Document that deals exclusively with the relationship of the publishers or authors of the Document to the Document's overall subject (or to related matters) and contains nothing that could fall directly within that overall subject. (Thus, if the Document is in part a textbook of mathematics, a Secondary Section may not explain any mathematics.) The relationship could be a matter of historical connection with the subject or with related matters, or of legal, commercial, philosophical, ethical or political position regarding them.

The "Invariant Sections" are certain Secondary Sections whose titles are designated, as being those of Invariant Sections, in the notice that says that the Document is released under this License. If a section does not fit the above definition of Secondary then it is not allowed to be designated as Invariant. The Document may contain zero Invariant Sections. If the Document does not identify any Invariant Sections then there are none.

The "Cover Texts" are certain short passages of text that are listed, as Front-Cover Texts or Back-Cover Texts, in the notice that says that the Document is released under this License. A Front-Cover Text may be at most 5 words, and a Back-Cover Text may be at most 25 words.

A "Transparent" copy of the Document means a machine-readable copy, represented in a format whose specification is available to the general public, that is suitable for revising the document straightforwardly with generic text editors or (for images composed of pixels) generic paint programs or (for drawings) some widely available drawing editor, and that is suitable for input to text formatters or for automatic translation to a variety of formats suitable for input to text formatters. A copy made in an otherwise Transparent file format whose markup, or absence of markup, has been arranged to thwart or discourage subsequent modification by readers is not Transparent. An image format is not Transparent if used for any substantial amount of text. A copy that is not "Transparent" is called "Opaque".

Examples of suitable formats for Transparent copies include plain ASCII without markup, Texinfo input format, LaTeX input format, SGML or XML using a publicly available DTD, and standard-conforming simple HTML, PostScript or PDF designed for human modification. Examples of transparent image formats include PNG, XCF and JPG. Opaque formats include proprietary formats that can be read and edited only by proprietary word processors, SGML or XML for which the DTD and/or processing tools are not generally available, and the machine-generated HTML, PostScript or PDF produced by some word processors for output purposes only.

The "Title Page" means, for a printed book, the title page itself, plus such following pages as are needed to hold, legibly, the material this License requires to appear in the title page. For works in formats which do not have any title page as such, "Title Page" means the text near the most prominent appearance of the work's title, preceding the beginning of the body of the text.

A section "Entitled XYZ" means a named subunit of the Document whose title either is precisely XYZ or contains XYZ in parentheses following text that translates XYZ in another language. (Here XYZ stands for a specific section name mentioned below, such as "Acknowledgements", "Dedications", "Endorsements", or "History".) To "Preserve the Title" of such a section when you modify the Document means that it remains a section "Entitled XYZ" according to this definition.

The Document may include Warranty Disclaimers next to the notice which states that this License applies to the Document. These Warranty Disclaimers are considered to be included by reference in this License, but only as regards disclaiming warranties: any other implication that these Warranty Disclaimers may have is void and has no effect on the meaning of this License.

2. VERBATIM COPYING

You may copy and distribute the Document in any medium, either commercially or noncommercially, provided that this License, the copyright notices, and the license notice saying this License applies to the Document are reproduced in all copies, and that you add no other conditions whatsoever to those of this License. You may not use technical measures to obstruct or control the reading or further copying of the copies you make or distribute. However, you may accept compensation in exchange for copies. If you distribute a large enough number of copies you must also follow the conditions in section 3.

You may also lend copies, under the same conditions stated above, and you may publicly display copies.

3. COPYING IN QUANTITY

If you publish printed copies (or copies in media that commonly have printed covers) of the Document, numbering more than 100, and the Document's license notice requires Cover Texts, you must enclose the copies in covers that carry, clearly and legibly, all these Cover Texts: Front-Cover Texts on the front cover, and Back-Cover Texts on the back cover. Both covers must also clearly and legibly identify you as the publisher of these copies. The front cover must present the full title with all words of the title equally prominent and visible. You may add other material on the covers in addition. Copying with changes limited to the covers, as long as they preserve the title of the Document and satisfy these conditions, can be treated as verbatim copying in other respects.

If the required texts for either cover are too voluminous to fit legibly, you should put the first ones listed (as many as fit reasonably) on the actual cover, and continue the rest onto adjacent pages.

If you publish or distribute Opaque copies of the Document numbering more than 100, you must either include a machine-readable Transparent copy along with each Opaque copy, or state in or with each Opaque copy a computer-network location from which the general network-using public has access to download using public-standard network protocols a complete Transparent copy of the Document, free of added material. If you use the latter option, you must take reasonably prudent steps, when you begin distribution of Opaque copies in quantity, to ensure that this Transparent copy will remain thus accessible at the stated location until at least one year after the last time you distribute an Opaque copy (directly or through your agents or retailers) of that edition to the public.

It is requested, but not required, that you contact the authors of the Document well before redistributing any large number of copies, to give them a chance to provide you with an updated version of the Document.

4. MODIFICATIONS

You may copy and distribute a Modified Version of the Document under the conditions of sections 2 and 3 above, provided that you release the Modified Version under precisely this License, with the Modified Version filling the role of the Document, thus licensing distribution and modification of the Modified Version to whoever possesses a copy of it. In addition, you must do these things in the Modified Version:

- A. Use in the Title Page (and on the covers, if any) a title distinct from that of the Document, and from those of previous versions (which should, if there were any, be listed in the History section of the Document). You may use the same title as a previous version if the original publisher of that version gives permission.
- B. List on the Title Page, as authors, one or more persons or entities responsible for authorship of the modifications in the Modified Version, together with at least five of the principal authors of the Document (all of its principal authors, if it has fewer than five), unless they release you from this requirement.
- C. State on the Title page the name of the publisher of the Modified Version, as the publisher.
- D. Preserve all the copyright notices of the Document.
- E. Add an appropriate copyright notice for your modifications adjacent to the other copyright notices.
- F. Include, immediately after the copyright notices, a license notice giving the public permission to use the Modified Version under the terms of this License, in the form shown in the Addendum below.
- G. Preserve in that license notice the full lists of Invariant Sections and required Cover Texts given in the Document's license notice.
- H. Include an unaltered copy of this License.
- I. Preserve the section Entitled "History", Preserve its Title, and add to it an item stating at least the title, year, new authors, and publisher of the Modified Version as given on the Title Page. If there is no section Entitled "History" in the Document, create one stating the title, year, authors, and publisher of the Document as given on its Title Page, then add an item describing the Modified Version as stated in the previous sentence.
- J. Preserve the network location, if any, given in the Document for public access to a Transparent copy of the Document, and likewise the network locations given in the Document for previous versions it was based on. These may be placed in the "History" section. You may omit a network location for a work that was published at least four years before the Document itself, or if the original publisher of the version it refers to gives permission.
- K. For any section Entitled "Acknowledgements" or "Dedications", Preserve the Title of the section, and preserve in the section all the substance and tone of each of the contributor acknowledgements and/or dedications given therein.

- L. Preserve all the Invariant Sections of the Document, unaltered in their text and in their titles. Section numbers or the equivalent are not considered part of the section titles.
- M. Delete any section Entitled "Endorsements". Such a section may not be included in the Modified Version.
- N. Do not retitle any existing section to be Entitled "Endorsements" or to conflict in title with any Invariant Section.
- O. Preserve any Warranty Disclaimers.

If the Modified Version includes new front-matter sections or appendices that qualify as Secondary Sections and contain no material copied from the Document, you may at your option designate some or all of these sections as invariant. To do this, add their titles to the list of Invariant Sections in the Modified Version's license notice. These titles must be distinct from any other section titles.

You may add a section Entitled "Endorsements", provided it contains nothing but endorsements of your Modified Version by various parties--for example, statements of peer review or that the text has been approved by an organization as the authoritative definition of a standard.

You may add a passage of up to five words as a Front-Cover Text, and a passage of up to 25 words as a Back-Cover Text, to the end of the list of Cover Texts in the Modified Version. Only one passage of Front-Cover Text and one of Back-Cover Text may be added by (or through arrangements made by) any one entity. If the Document already includes a cover text for the same cover, previously added by you or by arrangement made by the same entity you are acting on behalf of, you may not add another; but you may replace the old one, on explicit permission from the previous publisher that added the old one.

The author(s) and publisher(s) of the Document do not by this License give permission to use their names for publicity for or to assert or imply endorsement of any Modified Version.

5. COMBINING DOCUMENTS

You may combine the Document with other documents released under this License, under the terms defined in section 4 above for modified versions, provided that you include in the combination all of the Invariant Sections of all of the original documents, unmodified, and list them all as Invariant Sections of your combined work in its license notice, and that you preserve all their Warranty Disclaimers.

The combined work need only contain one copy of this License, and multiple identical Invariant Sections may be replaced with a single copy. If there are multiple Invariant Sections with the same name but different contents, make the title of each such section unique by adding at the end of it, in parentheses, the name of the original author or publisher of that section if known, or else a unique number. Make the same adjustment to the section titles in the list of Invariant Sections in the license notice of the combined work.

In the combination, you must combine any sections Entitled "History" in the various original documents, forming one section Entitled "History"; likewise combine any sections Entitled "Acknowledgements", and any sections Entitled "Dedications". You must delete all sections Entitled "Endorsements".

6. COLLECTIONS OF DOCUMENTS

You may make a collection consisting of the Document and other documents released under this License, and replace the individual copies of this License in the various documents with a single copy that is included in the collection, provided that you follow the rules of this License for verbatim copying of each of the documents in all other respects.

You may extract a single document from such a collection, and distribute it individually under this License, provided you insert a copy of this License into the extracted document, and follow this License in all other respects regarding verbatim copying of that document.

7. AGGREGATION WITH INDEPENDENT WORKS

A compilation of the Document or its derivatives with other separate and independent documents or works, in or on a volume of a storage or distribution medium, is called an "aggregate" if the copyright resulting from the compilation is not used to limit the legal rights of the compilation's users beyond what the individual works permit. When the Document is included in an aggregate, this License does not apply to the other works in the aggregate which are not themselves derivative works of the Document.

If the Cover Text requirement of section 3 is applicable to these copies of the Document, then if the Document is less than one half of the entire aggregate, the Document's Cover Texts may be placed on covers that bracket the Document within the aggregate, or the electronic equivalent of covers if the Document is in electronic form. Otherwise they must appear on printed covers that bracket the whole aggregate.

8. TRANSLATION

Translation is considered a kind of modification, so you may distribute translations of the Document under the terms of section 4. Replacing Invariant Sections with translations requires special permission from their copyright holders, but you may include translations of some or all Invariant Sections in addition to the original versions of these Invariant Sections. You may include a translation of this License, and all the license notices in the Document, and any Warranty Disclaimers, provided that you also include the original English version of this License and the original versions of those notices and disclaimers. In case of a disagreement between the translation and the original version of this License or a notice or disclaimer, the original version will prevail.

If a section in the Document is Entitled "Acknowledgements", "Dedications", or "History", the requirement (section 4) to Preserve its Title (section 1) will typically require changing the actual title.

9. TERMINATION

You may not copy, modify, sublicense, or distribute the Document except as expressly provided for under this License. Any other attempt to copy, modify, sublicense or distribute the Document is void, and will automatically terminate your rights under this License. However, parties who have received copies, or rights, from you under this License will not have their licenses terminated so long as such parties remain in full compliance.

10. FUTURE REVISIONS OF THIS LICENSE

The Free Software Foundation may publish new, revised versions of the GNU Free Documentation License from time to time. Such new versions will be similar in spirit to the present version, but may differ in detail to address new problems or concerns. See <http://www.gnu.org/copyleft/>.

Each version of the License is given a distinguishing version number. If the Document specifies that a particular numbered version of this License "or any later version" applies to it, you have the option of following the terms and conditions either of that specified version or of any later version that has been published (not as a draft) by the Free Software Foundation. If the Document does not specify a version number of this License, you may choose any version ever published (not as a draft) by the Free Software Foundation.

Appendix A – Platform Parameters

Parameter Name	Enum ID	Data Type	Size (B)	1 st /2 nd Deriv?	Units
Monitorable Parameters					
up_time	0	ATIME32	4		sec
time	1	ATIME32	4		sec, UTC
mission_time	2	ATIME32	4		sec
control_ss_status	3	ABYTE	1		CCL_SUBSYSTEM_STATUS_ENUM
navigation_ss_status	4	ABYTE	1		CCL_SUBSYSTEM_STATUS_ENUM
communications_ss_status	5	ABYTE	1		CCL_SUBSYSTEM_STATUS_ENUM
propulsion_ss_status	6	ABYTE	1		CCL_SUBSYSTEM_STATUS_ENUM
energy_ss_status	7	ABYTE	1		CCL_SUBSYSTEM_STATUS_ENUM
data_logger_ss_status	8	ABYTE	1		CCL_SUBSYSTEM_STATUS_ENUM
payload_ss_status	9	ABYTE	1		CCL_SUBSYSTEM_STATUS_ENUM
internal_temperature	10	AINT16	2	x	deg C
internal_pressure	11	AINT16	2	x	mbar
battery_voltage	12	AUINT16	2	x	mV
health	13	ABYTE	1		CCL_PLATFORM_HEALTH_ENUM
available_cpu	14	ABYTE	1	x	percent
available_data_storage	15	AUINT32	4	x	kB
maneuver_type	16	ABYTE	1		CCL_MANEUVER_CONTROL_ENUM
platform_mode	17	ABYTE	1		CCL_PLATFORM_CONTROL_ENUM
mission	18	ACHAR[15]	15		text, current mission name
num_tasks	19	ABYTE	1		0...15, # of running tasks
task_set	20	ABYTE[15]	15		0...63 taskID/slot, 15 max
maneuver_index	21	ABYTE	1		executing maneuver index
path_segment	22	ABYTE	1		executing path segment index
path_type	23	ABYTE	1		CCL_MANEUVER_PATH_ENUM
path	24	ACHAR[15]	15		text, current path label
avoiding_obstacle	25	ABYTE	1		CCL_BOOLEAN_ENUM
latitude	26	AFLT32	4	x	deg, +n/-s
longitude	27	AFLT32	4	x	deg, +e/-w
origin_offset_ns	28	AFLT32	4	x	m, +n/-s
origin_offset_ew	29	AFLT32	4	x	m, +e/-w
navigation	30	ABYTE	1		CCL_NAVIGATION_ENUM
track_error	31	AFLT32	4	x	m, perpendicular to track
speed_over_ground	32	AUINT16	2	x	cm/s
speed_over_water	33	AUINT16	2	x	cm/s
commanded_latitude	34	AFLT32	4		deg, +n/-s
commanded_longitude	35	AFLT32	4		deg, +e/-w
commanded_depth	36	AFLT32	4		m
commanded_altitude	37	AFLT32	4		m
commanded_speed	38	AUINT16	2		cm/s
commanded_bearing	39	AUINT16	2		tenths deg, true bearing
commanded_heading	40	AUINT16	2		tenths deg

commanded_pitch	41	AINT16							+/- tenths deg
commanded_roll	42	AINT16							+/- tenths deg
commanded_maneuver_timeout	43	ATIME32							sec
maneuver_remaining_time	44	ATIME32							sec
available_energy	45	AUINT16					x		wattHrs
used_energy_since_uptime	46	AUINT16					x		wattHrs
used_energy_since_midnite	47	AUINT16					x		wattHrs, UTC reference
used_energy_since_mission	48	AUINT16					x		wattHrs, from mission start
depth	49	AFLT32					x		m
altitude	50	AFLT32					x		m
heading	51	AUINT16					x		tenths deg
pitch	52	AINT16					x		+/- tenths deg
roll	53	AINT16					x		+/- tenths deg
bearing	54	AUINT16					x		tenths deg, true bearing
water_temperature	55	AINT32					x		milli-deg C
salinity	56	AUINT16					x		milli-PSU
conductivity	57	AUINT32					x		uS/cm
dissolved_oxygen	58	AUINT16					x		ug/l
fluorescence	59	AUINT16					x		ug/l
turbidity	60	AUINT16					x		NTU
transmissivity	61	AFLT32					x		percent
optical_backscatter	62	AUINT32					x		(TBD)
current_northerly	63	AINT16					x		cm/s, +n/-s
current_easterly	64	AINT16					x		cm/s, +e/-w
current_downward	65	AINT16					x		cm/s, +down/-up
current	66	AUINT16					x		cm/s
sound_velocity	67	AFLT32					x		m/s
magnetic_flux	68	AUINT32							(TBD)
subbottom_profiler	69	AUINT32							(TBD)
sidescan_image	70	AUINT32							(TBD)
multibeam_image	71	AUINT32							(TBD)
still_camera_image	72	AUINT32							(TBD)
video_camera_clip	73	AUINT32							(TBD)
audio_clip	74	AUINT32							(TBD)
Configurable Parameters									
authority	75	CCL_AUTHORITY_STRUCT							platform authority
abort_convention	76	ACHAR[15]						1	text, abort mission label
other_failure_convention	77	ACHAR[15]						15	text, failure mission label
loss_of_position_convention	78	ACHAR[15]						15	text, failure mission label
loss_of_depth_convention	79	ACHAR[15]						15	text, failure mission label
loss_of_altitude_convention	80	ACHAR[15]						15	text, failure mission label
loss_of_compass_convention	81	ACHAR[15]						15	text, failure mission label
loss_of_power_convention	82	ACHAR[15]						15	text, failure mission label
loss_of_thrust_convention	83	ACHAR[15]						15	text, failure mission label
low_energy_convention	84	ACHAR[15]						15	text, failure mission label
collision_convention	85	ACHAR[15]						15	text, failure mission label
leak_detected_convention	86	ACHAR[15]						15	text, failure mission label
pressure_oob_convention	87	ACHAR[15]						15	text, failure mission label

mission_role	88	ABYTE	1	CCL_MISSION_ROLE_ENUM
operations_mode	89	ABYTE	1	CCL_OPERATIONS_MODE_ENUM
operations_region	90	CCL_REGION_STRUCT	20	region spec
max_depth	91	AFLT32	4	m
min_altitude	92	AFLT32	4	m
configured_latitude	93	AFLT32	4	deg, +n/-s
configured_longitude	94	AFLT32	4	deg, +e/-w
configured_depth	95	AFLT32	4	m
configured_altitude	96	AFLT32	4	m
configured_speed	97	AUINT16	2	cm/s
configured_path	98	ACHAR[15]	15	text, path key label
configured_maneuver_timeout	99	ATIME32	4	sec, UTC
configured_monitor_timeout	100	ATIME32	4	sec, UTC
location_radius_tolerance	101	CCL_POS_TOL_STRUCT	2	xy tolerance radius
location_depth_tolerance	102	CCL_POS_TOL_STRUCT	2	z tolerance distance
navigation_origin	103	CCL_LAT_LNG_STRUCT	8	navigation origin
gps_configuration	104	CCL_GPS_CFG_STRUCT	2	GPS aspects
network_configuration	105	CCL_NETWORK_CFG_STRUCT	1	network aspects
vip_address	106	CCL_ADDRESS_STRUCT	1	platform address
report_update_period	107	CCL_TIME_INTERVAL_STRUCT	2	monitored report updates
no_contact_interval	108	CCL_TIME_INTERVAL_STRUCT	2	trigger for abort mission
energy_strategy	109	ABYTE	1	CCL_ENERGY_MANAGEMENT_ENUM
low_energy_limit	110	AUINT16	2	watchHrs
magnetic_deviation	111	AINT16	2	tenths deg offset true N
station_keep_speed	112	AUINT16	2	cm/s
timezone_offset	113	ACHAR	1	ops area time zone offset
max_speed	114	AUINT16	2	cm/s, max platform speed
cruise_speed	115	AUINT16	2	cm/s, platform cruise speed
min_speed	116	AUINT16	2	cm/s, min platform speed

Appendix B – Vocabulary Enumerations

Enumeration Set Context	Set Size (Bits)	Words (English gloss)	Description
CCL_MANEUVER_ENUM	3	GOTO FOLLOW_PATH	<p>The platform will attempt to maneuver to the specified latitude and longitude or positional offset from the current position or preconfigured origin, and depth or altitude along the specified path at the specified speed. Transition to the next maneuver behavior occurs when the platform believes it has attained the target destination (within the pre-configured waypoint radius and waypoint depth tolerance settings), when the specified or pre-configured timeout occurs, or if explicitly stopped.</p> <p>The platform will attempt to maneuver to follow a preconfigured multiple segment waypoint path. The maneuver is finished after the last path segment is complete, when the specified or pre-configured timeout occurs, or if explicitly stopped. A path can be defined through the Navigate:SetPath message or from a pre-existing path file and key.</p>
		SPEED_AND_BEARING	<p>The platform will attempt to move at the specified speed and compass bearing for a given duration. The vehicle will end this behavior when the specified or pre-configured timeout duration occurs, or if explicitly stopped.</p>
		MOVE	<p>The platform will attempt to attain a specific velocity (speed and direction) while maintaining a fixed orientation. If speed is set to zero, the platform will just attempt to orient itself in place. These behavior parameters need to be carefully specified to prevent under constrained, over constrained or nonsensical intended motion. The vehicle will end this behavior when the specified or pre-configured timeout duration occurs, or if explicitly stopped.</p>
		STATION_KEEP	<p>The platform will attempt to remain within a circle at constant depth while minimizing energy usage. If the platform drifts beyond the circle radius or +/- depth bounds, it will attempt to move back to the circle center at the pre-configured station keep speed. If position parameters are specified, the platform will consider these to be the circle center; otherwise the center is set at the current platform position. If orientation parameters are specified, the platform shall attempt to maintain the indicated orientation during this behavior. The vehicle will end this behavior when the specified or pre-configured timeout occurs, or if explicitly stopped.</p>

CCL_NAVIGATE_ENUM	2	GPS_FIX	The platform will attempt to update its geo-referenced position using its GPS navigation sensor. The platform will drive to the surface, acquire the GPS fix, and then resume its maneuver schedule. The activity is complete when the position fix has been acquired, a timeout occurs, or if explicitly stopped.
		AVOID_REGION	The platform is requested to avoid (for "start" behavior action) or stop avoiding (for "stop" behavior action) the designated regions, specified as a spherical or vertically-aligned cylindrical volume with the specified radius. Once set, avoidance regions remain in effect until explicitly cleared, using the "stop" or "stop all" behaviors actions.
	SET_PATH	The platform is requested to establish a preconfigured multiple link path. The path can be specified directly with a path segment set or through reference to an onboard file. Attributes including xy location, z location, and speed can be specified explicitly or using configured parameters and on a per-path or per path segment basis. Radius and depth tolerances are set as configured parameters. The path is executed through the Maneuver:FollowPath behavior by specifying either the default configured path or through a path key label.	

CCL_COMMUNICATE_ENUM	1	MESSAGE	A message is used to convey unsolicited information between platforms. A message may contain platform-specific information or CCL behaviors, in ASCII or binary format. The message content size is limited to 255 B. In addition to the message contents, this message contains information as to why it was generated and who the sender was.
----------------------	---	---------	---

CCL_CONFIGURE_ENUM	2	SYSTEM	The platform is requested to modify the requested parameter values defined in the parameter list configurable subset. Note that parameters can be configured from either the well-known CCL list or a user-defined custom list. Up to 15 separate parameters may be modified through a single request message.
		SENSOR	The platform is requested to modify the requested attributes of the indicated mission sensor. Mission sensors are categorized into types, of which up to three instances of each type may be addressed. Valid types are defined in the CCL_MISSION_SENSOR_ENUM set.
		COMMS	The platform is requested to modify the requested attributes of the indicated communications interface. Interfaces are categorized into types, of which up to three instances of each type may be addressed. Valid types are defined in the CCL_COMM_INTERFACE_ENUM set.

CCL_MONITOR_ENUM	3	STATUS	The platform is requested to transmit its current core status block and/or up to 15 additional monitorable or configurable parameters, based on a monitored trigger. Note that reporting of 1st or 2nd derivative values can only occur on monitorable parameters. The requestor can also provide a "report to" address for delivery of the resulting report.
		SYSTEM_CAPS	The platform is requested to provide a list of its capabilities, based on a monitored trigger. In addition, the requestor can provide a "report to" address.
		FILE	The platform is requested to transfer a specific file, based on a monitored trigger. The file name is provided and, optionally, a directory path where to search for the file. In addition, the requestor can provide a "report to" address.
		SENSOR_CAPS	The platform is requested to report on specific mission sensor capabilities, based on a monitored trigger. In addition, the requestor can provide a "report to" address.
		COMMS_CAPS	The platform is requested to report on specific communication interface capabilities, based on a monitored trigger. In addition, the requestor can provide a "report to" address.

CCL_EXEC_CONV_ENUM	3	SYSTEM_ADMIN	The platform is requested to alter the specified system administrative aspects. This includes shutting down and restarting the high-level control applications, and shutting down and restarting the platform.
		START_MISSION	The platform is requested to begin executing the indicated mission.
		STOP_MISSION	The platform is requested to stop executing the current mission.
		ABORT_MISSION	The platform is requested to stop executing the current mission and begin executing either the pre-configured or indicated abort mission.
		START_BEHAVIOR	The platform is requested to begin executing the indicated behavior.
		STOP_BEHAVIOR	The platform is requested to stop executing the indicated behavior.
		UPDATE_BEHAVIOR	The platform is requested to modify the indicated behavior. This can be done using modifications contained in an attached byte string or through the indicated on-board file.

CCL_CONTEXT_ENUM	3	COMMAND	Message context denoting a high-level request that should be performed immediately.
		URGENT_REQUEST	Message context denoting a request that should be performed immediately.
		REQUEST_INFORM	Message context denoting a request for an action to be performed.
		WARN	Message context denoting information to be added to the knowledge base.
		URGENT_WARN	Message context denoting a situation that is a threat to the platform.
			Message context denoting a situation that requires immediate attention.

CCL_BEH_ENUM	4	MANEUVER	This is the primary behavior class for commanding the platform to move or relocate itself to a new position.
		NAVIGATE	This is the primary behavior class for specifying path constraints and position updates for the platform as it maneuvers.
		COMMUNICATE	This is the primary behavior class for sending and receiving messages.
		CONFIGURE	This is the primary behavior class for specifying how the platform should reconfigure aspects of itself.
		MONITOR	This is the primary behavior class for specifying how the platform should watch and record some aspect of its self and report these aspects to others.
		EXEC_CONVENTION	This is the primary behavior class for specifying how the platform should carry out a universally understood action.
		SENSE	This is the primary behavior class for specifying how the platform senses its environment, using mission payload instruments.
		MANIPULATE	This is the primary behavior class for specifying how the platform physically manipulates its environment, using specialized actuators and sensors.
		SAMPLE	This is the primary behavior class for specifying how the platform collects samples of its environment, using specialized instruments.

CCL_BOOLEAN_ENUM	1	FALSE	Indicates a negative response to a situation or command.
		TRUE	Indicates an affirmative response to a situation or command.

CCL_BEHAVIOR_STACKING_ENUM	1	QUEUE	Queue the message for execution after the last similar request.
		REPLACE	Replace all similar queued and/or currently executing requests.

CCL_BEHAVIOR_ACTION_ENUM	2	START	Indicates that a new behavior action is being requested.
		STOP	Indicates that a behavior matching the specified input parameters should be stopped.
		STOP_ALL	Indicates that all behaviors within the context of this request should be stopped.

CCL_SCHEDULE_ENUM	2	SCHEDULE_NONE	Schedule this behavior to begin as soon as possible, given message context priority and system resource constraints.
		SCHEDULE_ON_TIME	Schedule this behavior to begin or end at the specified time.
		SCHEDULE_ON_LOCATION	Schedule this behavior to begin or end when the platform reaches the specified location.
		SCHEDULE_ON_EVENT	Schedule this behavior to begin or end when the specified event is triggered.
CCL_LOOP_ENUM	2	LOOP_N_TIMES	Repeat this behavior the specified number of times.
		LOOP_OVER_TIME_INTERVAL	Repeat this behavior over the time interval specified.
		LOOP_UNTIL_TIME	Repeat this behavior until the specified time is reached.
		LOOP_INDEFINITELY	Repeat this behavior indefinitely.
CCL_TIME_ENUM	2	NOW	Indicates the time when the platform receives the information or command message.
		CONFIGURED_TIME	Indicates that the pre-configured time value be used.
		ABSOLUTE_TIME	Specified time is an absolute time.
		RELATIVE_TIME	Specified time is relative to when the platform receives the information or command message.
CCL_TIME_INTERVAL_ENUM	2	SECONDS	Specified time is in units of seconds.
		MINUTES	Specified time is in units of minutes.
		HOURS	Specified time is in units of hours.
		DAYS	Specified time is in units of days.
CCL_HORZ_LOCATION_ENUM	3	CURRENT_LAT_LNG	Specifies the current latitude
		LAT_LNG	Latitude and longitude are specified.
		CONFIGURED_LAT_LNG	Specifies the configured latitude and longitude parameter values.
		PLATFORM_OFFSET	Specifies the north/east offset from current platform location.
		ORIGIN_OFFSET	Specifies the north/east offset from the currently configured origin parameter value.
CCL_VERT_LOCATION_ENUM	4	DEPTH	Specifies a given depth value.
		ALTITUDE	Specifies a given altitude value.
		CURRENT_DEPTH	Specifies the current platform depth.
		CURRENT_ALTITUDE	Specifies the current platform altitude.
		CONFIGURED_DEPTH	Specifies using the configured platform depth parameter.
		CONFIGURED_ALTITUDE	Specifies using the configured platform altitude parameter.
		PLATFORM_OFFSET	Specifies using a vertical offset from the current platform depth (+down/-up).
		SURFACE	Specifies a depth of zero; i.e. the water surface.
BOTTOM	Specifies the bottom of the water column.		

CCL_MANEUVER_PATH_ENUM	2	SHORTEST_PATH	In the maneuver behaviors, this indicates the platform follow a shortest path between two waypoints.
		CONSTANT_DEPTH_PATH	In the maneuver behaviors, this indicates the platform follow a constant depth path between two waypoints. If the waypoints are at different depths, the platform will change depth to attain the waypoints at the ends of the path.
		CONSTANT_ALTITUDE_PATH	In the maneuver behaviors, this indicates the platform follow a constant altitude path between two waypoints. If the waypoints are at different altitudes, the platform will change altitude to attain the waypoints at the ends of the path.
		USER_CONFIGURED_PATH	In the maneuver behaviors, this indicates the platform follow the path specified in the configurable parameter.
CCL_MANEUVER_CONTROL_TYPE	3	NOT MANEUVERING	The platform is currently not executing any maneuver behaviors.
		GOTO	The platform is currently executing the GoTo Waypoint behavior.
		FOLLOW_PATH	The platform is currently executing the Follow Path behavior.
		SPEED_AND_BEARING	The platform is currently executing the Speed and Bearing behavior.
		MOVE	The platform is currently executing the Move behavior.
		STATION_KEEP	The platform is currently executing the Station Keep behavior.
CCL_SPEED_ENUM	3	MAX SPEED	Indicates the platform travel at its maximum speed.
		CRUISE_SPEED	Indicates the platform travel at its configured cruise speed.
		MIN_SPEED	Indicates the platform travel at its minimum controlled speed.
		CONFIGURED_SPEED	Indicates the platform travel at configured parameter speed.
		SPEED	Indicates the platform travel at the specified speed.
CCL_MSG_THREAD_ENUM	2	MSG_THREAD_NOT_USED	Indicates the message conversation ID is not used (ID set to zero).
		MSG_THREAD_USE_ID	Indicates the message conversation ID is set by the user.
CCL_RDE_ENUM	3	UNSOLICITED	Denotes that this informative message was unsolicited.
		POLLED	Denotes that an informative message was solicited by another platform.
		TIMED_EVENT	Denotes that an informative message was due to a time monitored event on the platform.
		LOCATION_MONITORED_EVENT	Denotes that an informative message was due to a location monitored event on the platform.
		PARAMETER_MONITORED_EVENT	Denotes that an informative message was due to a parameter monitored event on the platform.
		FAILURE_EVENT	Denotes that an informative message was due to a failure on the platform.
		OTHER_EVENT	Denotes that an informative message was due to an (unspecified) event on the platform.
		ACK	Denotes that an informative message was an acknowledgement to a previous request message from an external agent.

CCL_REQUEST_ACK_ENUM	2	REQUEST NO ACK	No acknowledgment is requested for a given message.
		REQUEST ACK RECEIVED	An acknowledgment of receipt is requested for a given message.
		REQUEST_ACK_COMPLETED	An acknowledgment of task completion is requested for a given message.
		REQUEST_ACK_RECEIVED_AND_COMPLETED	Both an acknowledgment of initial receipt and then task completion is requested for a given message.
CCL_ACK_TYPE_ENUM	1	RECEIVED	The request has been received (with or without problems).
		COMPLETED	The request has completed (with or without problems).
CCL_ACK_STATUS_ENUM	4	SUCCESS	The request was successfully received and/or executed.
		OTHER_FAILURE	The request was not successfully received and/or executed, due to an unspecified reason.
		BEYOND_CAPS	The request was not successfully received and/or executed, as it is beyond the platform's capabilities.
		INSUFFICIENT_INFORMATION	The request was not successfully received and/or executed due to missing required information.
		NOT_UNDERSTOOD	The request was not successfully received and/or executed as it was not understood.
		INSUFFICIENT_AUTHORITY	The request was not successfully received and/or executed, as it is being requested by an agent without the proper authority.
		RESOURCE_NOT_FOUND	The request was not successfully received and/or executed, as a required resource needed to complete the behavior is missing.
		TIMED_OUT	The request was not successfully received and/or executed due to a behavior activity timing out.
CCL_SUBSYSTEM_STATUS_ENUM	2	NO_INFO	There is no information available on the indicated subsystem status.
		FAILURE	The indicated subsystem has failed.
		MARGINAL	The indicated subsystem is registering a problem.
		NORMAL	The indicated subsystem is functioning properly.

CCL_PLATFORM_HEALTH_ENUM	4	HEALTH_OK OTHER_FAILURE LOSS_OF_POSITION LOSS_OF_DEPTH LOSS_OF_ALTITUDE LOSS_OF_COMPASS LOSS_OF_POWER LOSS_OF_THRUST LOW_ENERGY COLLISION LEAK_DETECTED PRESSURE_OOB	There are no platform failures currently detected. An unspecified platform failure exists. The platform has indicated a failure due to uncertainty in its current estimated latitude and longitude. The platform has indicated a failure due to uncertainty in its current estimated depth. The platform has indicated a failure due to uncertainty in its current estimated altitude (if applicable). The platform has indicated a failure due to uncertainty in its current compass estimate. The platform has indicated a failure due to a loss in system power. The platform has indicated a failure due to a loss in platform thrust capability. The platform has indicated a failure due to on-board energy dropping below the configured parameter safety threshold. The platform has indicated a failure due to a sensed collision. The platform has indicated a failure due to a detected water leak. The platform has indicated a failure due to a detected out-of-bounds internal pressure reading.
--------------------------	---	---	---

CCL_NAVIGATION_ENUM	4	OTHER DEAD_RECKONING ACOUSTIC_LBL ACOUSTIC_SBL ACOUSTIC_USBL ACOUSTIC_MODEM_RANGING DOPPLER_SONAR INERTIAL GPS GPS_WAAS INTEGRATED MAGNETIC_DIPOLE GEOPHYSICAL	The indicated navigation subsystem is not specified. The indicated navigation subsystem is using dead reckoning. The indicated navigation subsystem is using acoustic long baseline. The indicated navigation subsystem is using acoustic short baseline. The indicated navigation subsystem is using acoustic ultra-short baseline. The indicated navigation subsystem is using acoustic modem ranging. The indicated navigation subsystem is using Doppler sonar. The indicated navigation subsystem is using an inertial navigation system (INS). The indicated navigation subsystem is using global positioning satellite system. The indicated navigation subsystem is using global positioning satellite system in WAAS mode. The indicated navigation subsystem is using an integrated navigation system. The indicated navigation subsystem is using a magnetic dipole navigation system. The indicated navigation subsystem is using geophysical navigation system.
---------------------	---	--	--

CCL_MISSION_ROLE_ENUM	4	OTHER	The current mission role of the platform is unspecified.
		ISR	The current mission role is as an intelligence, surveillance and reconnaissance platform.
		MCM	The current mission role is as a mine counter-measures platform.
		ASW	The current mission role is as an anti-submarine warfare platform.
		INSPECT_AND_IDENTIFY	The current mission role is as an inspection/identification platform.
		OCEANOGRAPHY	The current mission role is as an oceanographic sensing platform.
		CN3	The current mission role is as a communications and navigation network platform.
		PAYLOAD_DELIVERY	The current mission role is as a payload delivery platform.
		INFORMATION_OPERATIONS	The current mission role is as an information operations platform.
		TIME_CRITICAL_STRIKE	The current mission role is as a time critical strike platform.

CCL_DATA_COLLECTION_ENUM	4	OTHER	The primary data collection role of the platform is unspecified.
		COMMUNICATIONS_INTEL	The primary data collection role of the platform is communications intelligence gathering.
		ELECTRONIC_INTEL	The primary data collection role of the platform is electronic intelligence gathering.
		IMAGERY_INTEL	The primary data collection role of the platform is imagery intelligence gathering.
		SIGNAL_INTEL	The primary data collection role of the platform is signal intelligence gathering.
		MEASUREMENT_AND_SIGNATURE_INTEL	The primary data collection role of the platform is measurement and signature intelligence gathering.
		BATHYMETRY_MAPPING	The primary data collection role of the platform is bathymetric mapping.
		ACOUSTIC_IMAGERY	The primary data collection role of the platform is acoustic imagery gathering.
		OPTICAL_IMAGERY	The primary data collection role of the platform is optical imagery gathering.
		SUBBOTTOM_PROFILING	The primary data collection role of the platform is sub-bottom profiling.
		WATER_COLUMN_CHARACTERIZATION	The primary data collection role of the platform is water column characterization.
		TEMPERATURE_PROFILING	The primary data collection role of the platform is temperature profiling.
		SALINITY_PROFILING	The primary data collection role of the platform is salinity profiling.
		WATER_CLARITY	The primary data collection role of the platform is water clarity measurement.
		BIOLUMINESCENCE	The primary data collection role of the platform is bioluminescence measurement.
		CBN_DETECT_AND_TRACK	The primary data collection role of the platform is chemical, biological, and nuclear detection and tracking.

CCL_OPERATIONS_MODE_ENUM	1	OVERT COVERT	The platform is specified as operating in an overt mode. The platform is specified as operating in a covert mode.
CCL_PLATFORM_CLASS_ENUM	2	PORTABLE LIGHT WEIGHT HEAVY WEIGHT LARGE DISPLACEMENT	The platform displacement is on order of 100 lbs or less. The platform displacement is on order of 500 lbs. The platform displacement is on order of 3000 lbs or less. The platform displacement is on order of 20,000 lbs or more.
CCL_BODY_ENUM	3	OTHER TETHERED_BUOY DRIFTING_BUOY OPEN_SPACE_FRAME TORPEDO GLIDER CRAWLER	The platform body type is unspecified. The platform is a tethered surface buoy, which is confined to a specified location. The platform body drifting surface buoy. The platform body type is an open-space frame (e.g. EAVE). The platform body type is a torpedo (e.g. REMUS or MARV). The platform body type is a glider. The platform body type is a crawler, which is confined to the sea bottom.
CCL_PROPULSION_ENUM	2	NONE OTHER THRUSTER GLIDER	The platform does not employ any means of propulsion. The platform is using an unspecified propulsion system. The platform uses a thruster system for propulsion. The platform uses a glider system for propulsion.
CCL_ENERGY_MANAGEMENT_ENUM	3	OTHER MINIMUM_RESERVE SURFACE_OPS DAY_TRIP LONG_ENDURANCE	The platform is employing an unspecified energy management strategy. The platform is specifying an energy strategy which emphasizes minimum energy reserves. The platform is specifying an energy strategy optimized for surface operations. The platform is specifying an energy strategy optimized for day trips. The platform is specifying an energy strategy optimized for long endurance missions.
CCL_COMM_INTERFACE_ENUM	3	ALL OTHER SATELLITE ACOUSTIC RF CELLULAR	All communication interfaces are specified. The indicated communication interface is not specified. The indicated communication interface is a satellite transceiver. The indicated communication interface is an acoustic modem. The indicated communication interface is an RF modem/transceiver. The indicated communication interface is a cellular phone.
CCL_NETWORK_PROTOCOL_ENUM	3	NONE OTHER SEAWEB COFSNET AUSNET	No network protocol is specified. An unspecified network protocol is specified. The Seaweb network protocol is specified. The Controlled Flooding for Small Networks (COFSNET) protocol is specified. The Autonomous Systems Network (AUSNET) protocol is specified.

CCL_ADDRESS_ENUM	1	SENDER	Indicates the sender of a message, or this platform for wrapped messages.
		VIPS	Indicates that the specified platform(s) is the recipient.
CCL_SENSOR_CLASS_ENUM	2	OTHER	The sensor measurement properties are unspecified.
		ACOUSTIC	The sensor measures acoustic parameters.
		PHYSICAL	The sensor measures physical parameters.
		OPTICAL	The sensor measures optical parameters.
CCL_MISSION_SENSOR_ENUM	5	ALL	All mission sensors are referenced.
		OTHER	The reference is to an unspecified sensor.
		CTD	The reference is to a CTD sensor.
		PRESSURE	The reference is to a pressure sensor.
		ALTIMETER	The reference is to an altimeter sensor.
		TEMPERATURE	The reference is to a temperature sensor.
		SALINITY	The reference is to a salinity sensor.
		CONDUCTIVITY	The reference is to a conductivity sensor.
		DISSOLVED OXYGEN	The reference is to a dissolved oxygen sensor.
		FLUOROMETER	The reference is to a fluorometer sensor.
		TURBIDITY	The reference is to a turbidity sensor.
		TRANSMISSOMETER	The reference is to a transmissometer sensor.
		OPTICAL BACKSCATTER	The reference is to an optical backscatter sensor.
		DVL	The reference is to a doppler velocity logger sensor.
		MAGNETIC FLUX	The reference is to a magnetic flux sensor.
		SUBBOTTOM PROFILER	The reference is to a sub-bottom profiler.
		SIDESCAN SONAR	The reference is to a side-scan sonar.
		MULTIBEAM SONAR	The reference is to a multi-beam sonar.
		STILL CAMERA	The reference is to a still camera.
		VIDEO CAMERA	The reference is to a video camera.
		AUDIO RECORDER	The reference is to an audio recorder.
CCL_SENSOR_ACTIVE_ENUM	2	TRIGGERED	The sensor is triggered into an active state.
		FREE RUNNING	The sensor is free running.
		PASSIVE	The sensor is passive.
CCL_SENSOR_DIRECTION_ENUM	1	BEAM	The sensor uses a beam configuration for sensing.
		OMNI	The sensor is an omni-directional sensor.
CCL_FILE_ACCESS_ENUM	2	ACCESS_ALLOWED	Access to the specified file is allowed.
		INADEQUATE_PERMISSIONS	Access to the file is denied due to inadequate permissions.
		FILE_TOO_BIG	Access to the file is denied because the file is too big to open.
		FILE_NOT_FOUND	Access to the file is denied because the file could not be found.
CCL_FILE_ENCODING_ENUM	1	ASCII	The file contents are in ASCII format.
		BINARY	The file contents are in binary format.

CCL_FILE_CONTENT_ENUM	3	OTHER_CONTENT LOG_DATA DATA_SNIP CCL_SET AGGREGATE_BEHAVIOR PATH	File contains other content. File contains log data. File contains a snippet of sensor data. File contains a group of CCL messages. File contains an aggregate behavior which is possibly understood by this platform only and which may or may not employ CCL. File contains maneuver path segments.
-----------------------	---	---	--

CCL_FILE_SIZE_ENUM	2	BYTE KILOBYTE MEGABYTE	File size units are in bytes. File size units are in kilobytes. File size units are in megabytes.
--------------------	---	------------------------------	---

CCL_ACCESS_PARAM_ENUM	3	OTHER_FAILURE ACCESS_ALLOWED INADEQUATE_PERMISSIONS NOT_UNDERSTOOD NOT_SUPPORTED NOT_MONITORABLE NOT_CONFIGURABLE	A request to access a parameter (for monitoring, configuring ...) has failed for unspecified reasons. A request to access a parameter (for monitoring, configuring ...) has been accepted. A request to access a parameter has been denied due to inadequate permissions. A request to access a parameter has been denied because the request syntax was not properly formed. A request to access a parameter was denied because the parameter is not supported. A request to access a parameter was denied because the parameter is not accessible for monitoring. A request to access a parameter was denied because the parameter is not configurable.
-----------------------	---	---	---

CCL_PARAM_STATE_ENUM	1	UNINITIALIZED INITIALIZED	The parameter has not yet been initialized. The parameter has been initialized.
----------------------	---	------------------------------	--

CCL_VOLUME_GEOMETRY_ENUM	2	SPHERE VERTICAL_ROUND_CYLINDER VERTICAL_SQUARE_CYLINDER VERTICAL_RECT_CYLINDER	Refers to a spherical geometry. Refers to a cylindrical geometry with circular cross-section oriented along the depth axis. Refers to a cylindrical geometry with square cross-section oriented along the depth axis. Refers to a cylindrical geometry with rectangular cross-section oriented along the depth axis.
--------------------------	---	---	---

CCL_LENGTH_UNIT_ENUM	3	MILLIMETER CENTIMETER DECIMETER METER DECAMETER HECTOMETER KILOMETER	The units of length are in millimeters. The units of length are in centimeters. The units of length are in decimeters. The units of length are in meters. The units of length are in decameters. The units of length are in hectometers. The units of length are in kilometers.
----------------------	---	--	---

CCL_POWER_UNIT_ENUM	1	MILLIWATT	The units of power are milliwatts.
		WATT	The units of power are watts.
CCL_VALUE_DERIVATIVE_ENUM	2	VALUE	Base monitoring behavior actions on actual value of parameter.
		1 ST DERIVATIVE	Base monitoring behavior actions on first derivative of parameter.
		2 ND DERIVATIVE	Base monitoring behavior actions on second derivative of parameter.
CCL_LOGICAL_OPERATOR_ENUM	1	AND	The logical operator "and".
		OR	The logical operator "or".
CCL_RELATION_OPERATOR_ENUM	3	EQUAL TO	The relational operator "=".
		NOT EQUAL TO	The relational operator "!=".
		GREATER THAN	The relational operator ">".
		GREATER THAN OR EQUAL TO	The relational operator ">=".
		LESS THAN	The relational operator "<".
		LESS THAN OR EQUAL TO	The relational operator "<=".
CCL_PARAMETER_ENUM (monitorable)	128	UP_TIME	Elapsed time since the platform powered on.
		TIME	Current time on platform, UTC reference.
		MISSION_TIME	Elapsed time since mission start.
		CONTROL_SS_STATUS	Status of control subsystem.
		NAV_SS_STATUS	Status of navigation subsystem.
		COMM_SS_STATUS	Communications subsystem status.
		PROPULSION_SS_STATUS	Status of propulsion subsystem.
		ENERGY_SS_STATUS	Status of energy subsystem.
		DATALOGGER_SS_STATUS	Status of data logging subsystem.
		PAYLOAD_SS_STATUS	Status of mission payload subsystem.
		INTERNAL_TEMPERATURE	Current internal temperature of subsystem.
		INTERNAL_PRESSURE	Current internal pressure of subsystem.
		BATTERY_VOLTAGE	Current voltage across battery subsystem.
		HEALTH	Indicator of vehicle health.
		AVAILABLE_CPU	Relative amount of platform processing power currently available.
		AVAILABLE_DATA_STORAGE	Amount of disk space currently available for data storage.
		MANEUVER_TYPE	The kind of maneuver currently being executed.
		PLATFORM_CONTROL_MODE	The current fundamental control mode of the platform (idle, supervisory, autonomous).
		MISSION	The name of the current mission.
		NUM_TASKS	The number of currently running (high level) behavior tasks.
		TASK_SET	The currently running (high level) task IDs.
		MANEUVER_INDEX	The counter index for the currently running maneuver.
		PATH_SEGMENT	If following a path, the current path counter index.
		PATH_TYPE	If following a path, the kind of path being run.
		PATH	If following a path, the name of the path.
		AVOIDING_OBSTACLE	An indicator if an obstacle is currently being avoided.
LATITUDE	The current estimated latitude.		
LONGITUDE	The current estimated longitude.		

ORIGIN_OFFSET_NS	The current estimated north/south offset from the origin.
ORIGIN_OFFSET_EW	The current estimated east/west offset from the origin.
NAVIGATION	Current navigation strategy.
TRACK_ERROR	Estimated perpendicular distance from desired track.
SPEED_OVER_GROUND	The estimated platform speed over the ground.
SPEED_OVER_WATER	The estimated platform speed relative to the water.
COMMANDED_LATITUDE	The last commanded maneuver latitude value.
COMMANDED_LONGITUDE	The last commanded maneuver longitude value.
COMMANDED_DEPTH	The last commanded maneuver depth value.
COMMANDED_ALTITUDE	The last commanded maneuver altitude value.
COMMANDED_SPEED	The last commanded maneuver speed value.
COMMANDED_BEARING	The last commanded maneuver bearing value.
COMMANDED_HEADING	The last commanded maneuver heading value.
COMMANDED_PITCH	The last commanded maneuver pitch value.
COMMANDED_ROLL	The last commanded maneuver roll value.
COMMANDED_MANEUVER_TIMEOUT	The last commanded maneuver timeout value.
MANEUVER_REMAINING_TIME	Estimate of time remaining for current maneuver to complete.
AVAILABLE_ENERGY	Remaining minus reserve energy.
USED_ENERGY_SINCE_UPTIME	Energy used since power up.
USED_ENERGY_SINCE_MIDNITE	Energy used since previous midnight UTC time.
USED_ENERGY_SINCE_MISSION	Energy used since current mission began.
DEPTH	Current platform depth.
ALTITUDE	Current platform altitude.
HEADING	Current platform heading.
PITCH	Current platform pitch angle.
ROLL	Current platform roll angle.
BEARING	Current platform bearing angle.
WATER_TEMPERATURE	Current sensed water temperature value.
SALINITY	Current sensed salinity value.
CONDUCTIVITY	Current sensed conductivity value.
DISSOLVED_OXYGEN	Current sensed dissolved oxygen value.
FLUORESCENCE	Current sensed fluorescence value.
TURBIDITY	Current sensed turbidity value.
TRANSMISSIVITY	Current sensed transmissivity value.
OPTICAL_BACKSCATTER	Current sensed optical backscatter value.
CURRENT_NORTHERLY	Current sensed north/south water current value.
CURRENT_EASTERLY	Current sensed east/west water current value.
CURRENT_DOWNWARD	Current sensed down/up water current value.
CURRENT	Estimate of sensed current magnitude.
SOUND_VELOCITY	Estimate of local sound velocity.
MAGNETIC_FLUX	Sensed magnetic flux value (TBD).
SUBBOTTOM_PROFILE	Sensed sub-bottom profiler value (TBD).
SIDECAN_IMAGE	Sensed side-scan sonar value (TBD).
MULTIBEAM_IMAGE	Sensed multi-beam sonar value (TBD).
STILL_CAMERA_IMAGE	Sensed still camera image value (TBD).
VIDEO_CAMERA_CLIP	Sensed video camera clip value (TBD).
AUDIO_CLIP	Sensed audio clip value (TBD).

CCL_PARAMETER_ENUM (configurable)	AUTHORITY	The authority level of this platform.
	ABORT_CONVENTION	The mission name to be executed in an abort.
	OTHER_FAILURE_CONVENTION	The mission name to be executed when an unspecified failure occurs.
	LOSS_OF_POSITION_CONVENTION	The mission name to be executed for failure due to uncertainty in its current estimated latitude and longitude.
	LOSS_OF_DEPTH_CONVENTION	The mission name to be executed for failure due to uncertainty in its current estimated depth.
	LOSS_OF_ALTITUDE_CONVENTION	The mission name to be executed for failure due to uncertainty in its current estimated altitude.
	LOSS_OF_COMPASS_CONVENTION	The mission name to be executed for failure due to uncertainty in its current compass measurement.
	LOSS_OF_POWER_CONVENTION	The mission name to be executed for a power failure event.
	LOSS_OF_THRUST_CONVENTION	The mission name to be executed for a thruster failure event.
	LOW_ENERGY_CONVENTION	The mission name to be executed when the energy available drops below the configured low energy limit.
	COLLISION_CONVENTION	The mission name to be executed in the event of a collision.
	LEAK_DETECTED_CONVENTION	The mission name to be executed if a leak failure is detected.
	PRESSURE_OOB_CONVENTION	The mission name to be executed if the internal pressure is detected to be out-of-bounds.
	MISSION_ROLE	The platform's primary mission role.
	OPERATIONS_MODE	The platform's operational mode.
	OPERATIONS_REGION	The platform's operational region.
	MAX_DEPTH	Maximum operating depth of platform.
	MIN_ALTITUDE	Minimum platform operating altitude.
	CONFIGURED_LATITUDE	The configured maneuver latitude value.
	CONFIGURED_LONGITUDE	The configured maneuver longitude value.
	CONFIGURED_DEPTH	The configured maneuver depth value.
	CONFIGURED_ALTITUDE	The configured maneuver altitude value.
	CONFIGURED_SPEED	The configured maneuver speed value.
	CONFIGURED_PATH	The configured maneuver path key label.
	CONFIGURED_MANEUVER_TIMEOUT	The configured maneuver behavior timeout value.
	CONFIGURED_MONITOR_TIMEOUT	The configured monitor behavior timeout value.
	LOCATION_RADIUS_TOLERANCE	Waypoint radius tolerance.
	LOCATION_DEPTH_TOLERANCE	Waypoint depth tolerance.
	NAVIGATION_ORIGIN	Navigation origin for relative location specification.
	GPS_CONFIGURATION	GPS configuration settings.
	NETWORK_CONFIGURATION	Network configuration settings.
	VIP_ADDRESS	Platform network address.
	REPORT_UPDATE_PERIOD	Update period for monitorable reports.
	NO_CONTACT_INTERVAL	Time interval within which platform must hear from operator else abort convention is executed.
	ENERGY_STRATEGY	Currently executing energy management strategy.
	LOW_ENERGY_LIMIT	Energy threshold, below which low energy convention is executed.
	MAGNETIC_DEVIATION	User configured magnetic deviation for operating region.
	STATION_KEEP_SPEED	The speed value used to return to the center of the station keep circle.

		TIMEZONE_OFFSET	The time zone offset for the platform operations area.
		MAX_SPEED	The maximum commanded platform speed.
		CRUISE_SPEED	A configurable platform cruise speed.
		MIN_SPEED	The minimum commanded platform speed.

CCL_DEVICE_STATE_ENUM	3		The device is on.
		OFF	The device is off.
		SLEEP	The device is in sleep mode.
		STANDBY	The device is in standby mode.
		FAILURE	The device has failed.

CCL_DEVICE_CONTROL_ENUM	2		Used to turn the device on.
		OFF	Used to turn the device off.
		SLEEP	Used to put the device in sleep mode.
		STANDBY	Used to put the device in standby mode.

CCL_DATA_TYPE_ENUM	4	CHAR_STR	The data type is a character string, need to specify length.
		BYTE_STR	The data type is a byte string, need to specify length.
		CHAR	The data type is a signed character.
		BYTE	The data type is an unsigned character.
		INT16	The data type is a 2 byte signed integer.
		UINT16	The data type is a 2 byte unsigned integer.
		INT32	The data type is a 4 byte signed integer.
		UINT32	The data type is a 4 byte unsigned integer.
		FLT32	The data type is a 4 byte floating point value.
		FLT64	The data type is an 8 byte floating point value.

CCL_SYS_ADM_CONTROL_ENUM	3	SHUTDOWN_APPS	The platform is instructed to shut down the high-level software applications.
		SHUTDOWN_PLATFORM	The platform is instructed to shut down the primary control computer.
		RESTART_APPS	The platform is instructed to restart the high-level software applications.
		RESTART_OS	The platform is instructed to restart the high-level operating system.
		RESTART_PLATFORM	The platform is instructed to restart the primary control computer.

CCL_PLATFORM_CONTROL_ENUM	2	IDLE	The platform is currently idling, running neither a mission nor a human-supervised behavior.
		REMOTELY_OPERATED	The platform is currently being supervised in an ROV-like fashion.
		RUNNING_MISSION	The platform is currently running an autonomous mission.

Appendix C – Data Structures

<u>Structure Label</u>	<u>Size (B)</u>	<u>Data Field</u>	<u>Field Bits</u>	<u>Data Type</u>	<u>Description</u>
CCL_HDR_STRUCT	3	context	3	CCL_CONTEXT_ENUM	Context of this message.
		beh_type	4	CCL_BEH_ENUM	Generic Behavior type.
		string_api_support	1	CCL_BOOLEAN_ENUM	If true, serialization and de-serialization using string API guaranteed to be safe.
		beh_subtype	3	CCL_MANEUVER_ENUM CCL_NAVIGATE_ENUM CCL_COMMUNICATE_ENUM CCL_CONFIGURE_ENUM CCL_MONITOR_ENUM CCL_EXEC_CONV_ENUM	Depending on beh_type field, the indicated subtype will be used.
		thread_id	13	integer, 0...8191	Unique conversational thread ID.
CCL_REQ_HDR_STRUCT	1	ack_type	2	CCL_REQUEST_ACK_ENUM	Request of acknowledgment type from recipient.
		use_authority	1	CCL_BOOLEAN_ENUM	If true, compare this platform's authority level to sender's level.
		group_head	1	CCL_BOOLEAN_ENUM	Is this the head of a CCL group? If yes, do not need to specify schedule and loop info for other CCLs between this and group tail.
		group_tail	1	CCL_BOOLEAN_ENUM	Group tail marker.
		group_loop	1	CCL_BOOLEAN_ENUM	If true, then repeating this behavior and CCL_LOOP_STRUCT required.
		beh_stacking	1	CCL_BEHAVIOR_STACKING_ENUM	Behavior stacking allows for a message request to be queued for execution after the last similar request, or to replace all similar queued and/or currently executing requests.
		-	1	-	Padding.
CCL_INF_HDR_STRUCT	1	raison_d_etre	3	CCL_RDE_ENUM	The reason why this inform message was generated.
		ack_type	1	CCL_ACK_TYPE_ENUM	Acknowledgment type.
		ack_status	4	CCL_ACK_STATUS_ENUM	Extended information on ack success or failure.

CCL_GOTO_OPT_STRUCT	2	target_xy	3	CCL_HORZ_LOCATION_ENUM	Specific waypoint reference type for GoTo maneuver.
		target_z	4	CCL_VERT_LOCATION_ENUM	Target depth for GoTo maneuver.
		use_config_radius_tol	1	CCL_BOOLEAN_ENUM	Waypoint size. If false: requires CCL_POS_TOL_STRUCT, if true: use configured radius tolerance.
		use_config_depth_tol	1	CCL_BOOLEAN_ENUM	Waypoint size. If false: requires CCL_POS_TOL_STRUCT, if true: use configured depth tolerance.
		path_type	2	CCL_MANEUVER_PATH_ENUM	Path type for GoTo maneuver.
		speed	3	CCL_SPEED_ENUM	Speed reference type for GoTo maneuver.
		timeout	2	CCL_TIME_ENUM	Time reference for indicating end of GoTo maneuver.

CCL_FOLLOW_PATH_OPT_STRUCT	1	pathkey_len	4	integer, 0...15	Path key label. If = 0: start default configured path, if > 0, start named path: 1...15 ACHAR max array.
		timeout	2	CCL_TIME_ENUM	Time reference for indicating end of Follow Path maneuver.
		-	2	-	Padding.

CCL_SPD_AND_BRNG_OPT_STRUCT	2	boundary_z	4	CCL_VERT_LOCATION_ENUM	Target depth for Speed and Bearing maneuver.
		speed	3	CCL_SPEED_ENUM	Speed for maneuver.
		duration	2	CCL_TIME_ENUM	Duration of maneuver.
		-	7	-	Padding.

CCL_MOVE_OPT_STRUCT	1	heading_constrained	1	CCL_BOOLEAN_ENUM	If true, specify compass heading.
		pitch_constrained	1	CCL_BOOLEAN_ENUM	If true, specify pitch angle.
		roll_constrained	1	CCL_BOOLEAN_ENUM	If true, specify roll angle.
		ns_vel_spec	1	CCL_BOOLEAN_ENUM	If true, specify n/s speed vector.
		ew_vel_spec	1	CCL_BOOLEAN_ENUM	If true, specify e/w speed vector.
		ds_vel_spec	1	CCL_BOOLEAN_ENUM	If true, specify down/up speed vector.
		duration	2	CCL_TIME_ENUM	Duration of maneuver.
		-	7	-	Padding.

CCL_STATION_KEEP_OPT _STRUCT	2	target_xy_constrained	1	CCL_BOOLEAN_ENUM	Constrain the lat/lon for this maneuver?
		target_xy	3	CCL_HORZ_LOCATION_ENUM	If constrained, target lat/lon for maneuver.
		target_z_constrained	1	CCL_BOOLEAN_ENUM	Constrain the depth for this maneuver?
		target_z	4	CCL_VERT_LOCATION_ENUM	If constrained, target depth for maneuver.
		drift_radius	1	CCL_BOOLEAN_ENUM	If true, use the specified drift radius for the watch circle. Otherwise, use preconfigured radius.
		drift_depth	1	CCL_BOOLEAN_ENUM	If true, use the specified drift depth for the watch circle. Otherwise, use preconfigured depth.
		heading_control	1	CCL_BOOLEAN_ENUM	If true, maintain the indicated heading.
		pitch_control	1	CCL_BOOLEAN_ENUM	If true, maintain the indicated pitch angle.
		roll_control	1	CCL_BOOLEAN_ENUM	If true, maintain the indicated roll angle.
		timeout	2	CCL_TIME_ENUM	Time reference for indicating end of maneuver.

CCL_GPS_FIX_REQ_OPT _STRUCT	1	use_config_timeout	1	CCL_BOOLEAN_ENUM	If true, use configured timeout value. Otherwise, use specified time interval.
		-	7	-	Padding.

CCL_AVOID_REG_REQ_OPT _STRUCT	3	action	2	CCL_BEHAVIOR_ACTION_ENUM	Use to set or clear avoidance regions.
		shape	2	CCL_VOLUME_GEOMETRY_ENUM	The shape of the region.
		x_constrained	1	CCL_BOOLEAN_ENUM	Is n/s direction constrained? If true, specify ref lat/lon or offsets and extent.
		y_constrained	1	CCL_BOOLEAN_ENUM	Is e/w direction constrained? If true, specify ref lat/lon or offsets and extent.
		z_constrained	1	CCL_BOOLEAN_ENUM	Is down/up direction constrained? If true, specify ref depth and extent.
		use_latlng	1	CCL_BOOLEAN_ENUM	Use lat/lon rather than offsets.
		within_region	1	CCL_BOOLEAN_ENUM	Use to denote within region (true) or outside region (false).
		angle	8	integer, 0...359	Clockwise rotation angle of region from north.
		-	7	-	Padding.

CCL_SET_PATH_OPT_STRUCT	5	target_xy	3	CCL_HORZ_LOCATION_ENUM	Use this lat/lon for entire path.	
		per_segment_xy	1	CCL_BOOLEAN_ENUM	If true, use the specified per-segment lat/lon in the path segment set or file. If false, use same lat/lon for entire path from previous target xy field.	
		target_z	4	CCL_VERT_LOCATION_ENUM	Use this depth/altitude for entire path.	
		per_segment_z	1	CCL_BOOLEAN_ENUM	If true, use the specified per-segment depth in the path segment set or file. If false, use same depth for entire path from previous target z field.	
		speed	3	CCL_SPEED_ENUM	Use this speed over entire path.	
	CCL_CFG_PARAMS_REQ_OPT_STRUCT	1	per_segment_spd	1	CCL_BOOLEAN_ENUM	If true, use the specified per-segment speed in the path segment set or file. If false, use same speed for entire path from previous speed field.
			num_segments	5	integer, 0...31	If = 0, reference an existing specified path file. If > 0, specify number of path segments in this message.
			path	2	CCL_MANEUVER_PATH_ENUM	The type of maneuver path to follow.
			save_path	1	CCL_BOOLEAN_ENUM	If true, save this path using the following filename and directory.
			filename_len	4	integer, 0...15	Name of file to save.
			dirname_len	5	integer, 0...31	Directory name, use a default directory if = 0.
			pathkey_len	4	integer, 0...15	Assign a path key label if > 0.
			set_as_config	1	CCL_BOOLEAN_ENUM	If true, make this path the default path.
			timeout	2	CCL_TIME_ENUM	Time reference for indicating end of maneuver.
			-	3	-	Padding.
	CCL_MSG_INF_OPT_STRUCT	2	message_size	8	integer, 0...255	Size of the message [B].
			encoding	1	CCL_FILE_ENCODING_ENUM	Message encoding type.
content_type			3	CCL_FILE_CONTENT_ENUM	Message content type.	
-			4	-	Padding.	
CCL_CFG_PARAMS_REQ_OPT_STRUCT	1	num_params	4	integer, 1...15	The total number of parameters requested for configuration.	
		-	4	-	Padding.	

CCL_SENSOR_OPT_STRUCT	1	type	5	CCL_MISSION_SENSOR_ENUM	The type or class of sensor to be configured.
		instance	2	integer, 0...3	Instance of the class: 0 = all instances, 1,2,3 = specific instance.
		-	1	-	Padding.
CCL_SENSOR_CFG_STRUCT	1	control	2	CCL_DEVICE_CONTROL_ENUM	Command to the sensor.
		sample rate	1	TBD	Set the sample rate.
		power level	1	TBD	Set the power level.
		sensitivity_threshold	1	TBD	Set the sensitivity threshold.
		gain	1	TBD	Set the gain.
		log rate	1	TBD	Set the logging rate.
-	1	-	Padding.		
CCL_COMMS_IF_OPT_STRUCT	1	type	3	CCL_COMM_INTERFACE_ENUM	The type or class of communications interface to be configured.
		instance	2	integer, 0...3	Instance of the class: 0 = all instances, 1,2,3 = specific instance.
		-	3	-	Padding.
CCL_COMMS_IF_CFG_STRUCT	1	control	2	CCL_DEVICE_CONTROL_ENUM	Command to the sensor.
		power level	1	TBD	Set the power level.
		sensitivity_threshold	1	TBD	Set the sensitivity threshold.
		gain	1	TBD	Set the gain.
		baud_rate	1	TBD	Set the baud rate.
		-	2	-	Padding.
CCL_CFG_PARAMS_INF_OPT_STRUCT	1	num_params	4	integer, 1...15	The total number of parameters which failed a request for configuration.
		-	4	-	Padding.
CCL_CFG_SENSOR_FAIL_STRUCT	1	control_set_fail	1	CCL_BOOLEAN_ENUM	The request to configure this item failed.
		sample_rate_set_fail	1	CCL_BOOLEAN_ENUM	The request to configure this item failed.
		power_level_set_fail	1	CCL_BOOLEAN_ENUM	The request to configure this item failed.
		sensitivity_thresh_set_fail	1	CCL_BOOLEAN_ENUM	The request to configure this item failed.
		gain_set_fail	1	CCL_BOOLEAN_ENUM	The request to configure this item failed.
		log_rate_set_fail	1	CCL_BOOLEAN_ENUM	The request to configure this item failed.
		-	2	-	Padding.

CCL_CFG_COMMS_IF_FAIL_STRUCT	1	control_set_fail	1	CCL_BOOLEAN_ENUM	The request to configure this item failed.
		power_level_set_fail	1	CCL_BOOLEAN_ENUM	The request to configure this item failed.
		sensitivity_thresh_set_fail	1	CCL_BOOLEAN_ENUM	The request to configure this item failed.
		gain_set_fail	1	CCL_BOOLEAN_ENUM	The request to configure this item failed.
		baud_rate_set_fail	1	CCL_BOOLEAN_ENUM	The request to configure this item failed.
		-	3	-	Padding.

CCL_MON_SCHEDULE_STRUCT	2	action	2	CCL_BEHAVIOR_ACTION_ENUM	Use to set or clear a monitor behavior.
		specify_start_time	1	CCL_BOOLEAN_ENUM	If true, start monitoring at indicated time. If false, start monitoring immediately.
		specify_end_time	1	CCL_BOOLEAN_ENUM	If true, end monitoring at indicated time. If false, continue until explicitly stopped.
		schedule	2	CCL_SCHEDULE_ENUM	If "SCHEDULE_NONE", execute immediately. If "SCHEDULE_ON_TIME", begins at start time. If "SCHEDULE_ON_LOCATION", include CCL_REGION_STRUCT. If "SCHEDULE_ON_EVENT", include 1-3 parameters.
		num_params	2	integer, 0...3	If "SCHEDULE_ON_EVENT", the number of parameters to watch.
		one_time_report	1	CCL_BOOLEAN_ENUM	If true, once a monitored event is triggered, monitor behavior ends. If false, continue to monitor and report at the configured CCL_PARAM_REPORT_UPDATE_PERIOD or indicated period.
		specify_update_period	1	CCL_BOOLEAN_ENUM	If true, include CCL_TIME_INTERVAL_STRUCT. If false, use configured CCL_PARAM_REPORT_UPDATE_PERIOD.
		report_to	1	CCL_ADDRESS_ENUM	The address of the platform for which to send the report.
		-	5	-	Padding.

CCL_PARAM_TRIGGER_HDR_STRUCT	1	deriv_type	2	CCL_VALUE_DERIVATIVE_ENUM	For triggering a report based on a parameter value or derivative change - requires CCL_TIME_INTERVAL_STRUCT for 1st or 2nd derivative types.
		rel_operator	3	CCL_RELATION_OPERATOR_ENUM	Specifies how a parameter and a value or derivative is relationally tested for true or false.
		log_operator	1	CCL_LOGICAL_OPERATOR_ENUM	Specifies how two parameters are logically test: only used if another parameter is to follow.
		-	2	-	Padding.
CCL_STATUS_REQ_OPT_STRUCT	1	extended_params	4	integer, 0...15	Number of additional parameters to report on ("extended status").
		-	4	-	Padding.
CCL_FILE_REQ_OPT_STRUCT	2	filename_len	4	integer, 1...15	The length of the filename.
		dirname_len	5	integer, 0...31	The length of the pathname. If 0, assume a default directory.
		-	7	-	Padding.
CCL_SENSOR_CAPS_REQ_OPT_STRUCT	1	type	5	CCL_MISSION_SENSOR_ENUM	The type or class of sensor to be reported.
		instance	2	integer, 0...3	Instance of the class: 0 = all instances, 1,2,3 = specific instance.
		-	1	-	Padding.

CCL_COMMS_CAPS_REQ_OPT_STRUCT	1	type	3	CCL_COMM_INTERFACE_ENUM	The type or class of communications interface to be reported.
		instance	2	integer, 0...3	Instance of the class: 0 = all instances, 1,2,3 = specific instance.
		-	3	-	Padding.

CCL_STATUS_INF_OPT_STRUCT	1	extended_params	4	integer, 0...15	Number of additional parameters requested included with status.
		-	4	-	Padding.

CCL_STATUS_STRUCT	31	time	32	4 B Unix time t	UTC time since Jan 1, 1970 [sec].
		agent_id	6	integer, 0...62	Unique platform identifier.
		group_id	2	integer, 0...2	Platform's group identifier.
		health	4	CCL_PLATFORM_HEALTH_ENUM	Indicator of overall vehicle health.
		heading	9	integer, 0...359	Vehicle compass heading [deg].
		speed	10	integer, 0...1023	Vehicle speed [cm/s].
		avoiding_obstacle	1	CCL_BOOLEAN_ENUM	Indicator if an obstacle is currently being avoided.
		current_nav	4	CCL_NAVIGATION_ENUM	Current navigation strategy in use.
		maneuver_type	3	CCL_MANEUVER_CONTROL_ENUM	Current maneuver strategy in use.
		maneuver_index	8	integer, 0...255	Currently executing maneuver identifier.
		ops_mode	1	CCL_OPERATIONS_MODE_ENUM	Covert/overt ops mode.
		altitude	16	integer, 0...65535	Vehicle altitude [cm]
		available_energy	16	integer, 0...65535	Available energy on vehicle [whrs].
		depth	32	4 B float	Vehicle depth [m].
		lat	32	4 B float	Vehicle latitude [deg].
		lng	32	4 B float	Vehicle longitude [deg].
		path_segment	8	integer, 0...255	Currently executing path segment id.
		num_tasks	3	integer, 0...7	Number of valid aggregate behavior task IDs.
		taskID_1	3	integer, 0...7	Aggregate behavior task ID 1.
		taskID_2	3	integer, 0...7	Aggregate behavior task ID 2.
		taskID_3	3	integer, 0...7	Aggregate behavior task ID 3.
		taskID_4	3	integer, 0...7	Aggregate behavior task ID 4.
		taskID_5	3	integer, 0...7	Aggregate behavior task ID 5.
		taskID_6	3	integer, 0...7	Aggregate behavior task ID 6.
		taskID_7	3	integer, 0...7	Aggregate behavior task ID 7.
		platform_mode	2	CCL_PLATFORM_CONTROL_ENUM	Current control mode of platform.
		-	6	-	Padding.

CCL_CAPS_STRUCT	63	lang_release	1	CCL_BOOLEAN_ENUM	CCL language version info. If true, "release". If false, "pre-release".
		lang_major_rev	4	integer, 0...15	CCL language major revision number.
		lang_minor_rev	11	integer, 0...2047	CCL language minor revision number.
		name	16 B	string, 1...15 characters, with trailing NULL	Platform name.
		agent_id	6	integer, 0...62	Unique platform identifier.
		group_id	2	integer, 0...2	Platform's group identifier.
		platform_class	2	CCL_PLATFORM_CLASS_ENUM	Relative size/displacement of platform.
		ops_range	14	integer, 0...16383	Operational range [km].
		mission_role	4	CCL_MISSION_ROLE_ENUM	Platform mission role.
		ops_mode	1	CCL_OPERATIONS_MODE_ENUM	Platform operational role.
		length	10	integer, 0...1023	Platform length [decimeter].
		longitudinal_control	1	CCL_BOOLEAN_ENUM	If true, longitudinal control available.
		lateral_control	1	CCL_BOOLEAN_ENUM	If true, lateral control available.
		body_type	3	CCL_BODY_ENUM	Body shape/type of platform.
		propulsion_type	2	CCL_PROPULSION_ENUM	Propulsion type of platform.
		displacement	16	integer, 0...65535	Displacement [kg].
		diameter	10	integer, 0...1023	Representative platform diameter [cm].
		vertical_control	1	CCL_BOOLEAN_ENUM	If true, platform can move vertically.
		max_depth	14	integer, 0...16383	Maximum depth of platform [m].
		min_altitude	14	integer, 0...16383	Minimum altitude (bottom safety) of platform [cm].
		heading_control	1	CCL_BOOLEAN_ENUM	If true, yaw control available.
		pitch_control	1	CCL_BOOLEAN_ENUM	If true, pitch control available.
		roll_control	1	CCL_BOOLEAN_ENUM	If true, roll control available.
		max_roll	8	integer, 0...180	Maximum symmetric roll angle [deg].
		min_turn_radius	10	integer, 0...1023	Minimum turn radius [decimeter].
		max_speed	10	integer, 0...1023	Maximum platform speed [cm/s].
		underwater_navigation	4	CCL_NAVIGATION_ENUM	Primary navigation strategy while underwater.
		surface_navigation	4	CCL_NAVIGATION_ENUM	Primary navigation strategy while surfaced.
		min_speed	10	integer, 0...1023	Minimum operational platform speed [cm/s].
		cruise_speed	10	integer, 0...1023	Pre-configured operational platform speed [cm/s].
		max_yaw_rate	8	integer, 0...255	Maximum yaw rate of platform [deg/s].
		min_pitch	7	integer, 0...90	Minimum controlled pitch angle, [deg] below horizontal.
		max_pitch	7	integer, 0...90	Maximum controlled pitch angle,

					[deg] above horizontal. If true, obstacle avoidance system available.
obstacle_avoid_ss	1	CCL_BOOLEAN_ENUM			
available_energy	16	integer, 0...65535			Remaining energy - reserve energy [kWhrs]
reserve_energy	16	integer, 0...65535			Reserve energy [kWhrs].
capacity_energy	16	integer, 0...65535			Maximum energy storage capacity [kWhrs].
endurance_units	2	CCL_TIME_INTERVAL_ENUM			Units for endurance under low and high energy consumption conditions.
endurance_low_load	10	integer, 0...1023			Estimate of remaining endurance under low energy consumption conditions [endurance units].
endurance_high_load	10	integer, 0...1023			Estimate of remaining endurance under high energy consumption conditions [endurance units].
energy_strategy	3	CCL_ENERGY_MANAGEMENT_ENUM			Current energy management strategy.
other_energy	1	CCL_BOOLEAN_ENUM			Platform employs an unspecified energy collection/storage mechanism.
solar_energy	1	CCL_BOOLEAN_ENUM			Platform employs a solar energy collection mechanism.
fuel_cell_energy	1	CCL_BOOLEAN_ENUM			Platform employs a fuel cell subsystem for energy storage.
diesel_energy	1	CCL_BOOLEAN_ENUM			Platform employs a diesel energy generation mechanism.
nuclear_energy	1	CCL_BOOLEAN_ENUM			Platform employs a nuclear energy generation mechanism.
data_collect_role	4	CCL_DATA_COLLECTION_ENUM			Primary data collection role (if relevant).
disk_capacity	20	integer, 0...1048575			Storage capacity of disk [MB].
disk_pct_available	7	integer, 0...100			Percent of disk capacity remaining.
network_protocol	3	CCL_NETWORK_PROTOCOL_ENUM			Current communications network protocol.
other_comms	2	integer, 0...3			Number of unspecified communications interfaces installed.
satellite_comms	2	integer, 0...3			Number of satellite communications interfaces installed.
acoustic_comms	2	integer, 0...3			Number of acoustic communications interfaces installed.
rf_comms	2	integer, 0...3			Number of RF communications interfaces installed.
cellular_comms	2	integer, 0...3			Number of cellular phone communications interfaces installed.
cpu_capability	8	integer, 0...255			CPU benchmark index [ref machine TBD].
other_sensors	2	integer, 0...3			Number of unspecified mission

					sensors installed. Number of CTD mission sensors installed.
	ctd_sensors	2	integer, 0...3		Number of CTD mission sensors installed.
	pressure_sensors	2	integer, 0...3		Number of pressure measurement mission sensors installed.
	altimeter_sensors	2	integer, 0...3		Number of altitude measurement mission sensors installed.
	temperature_sensors	2	integer, 0...3		Number of water temperature measurement mission sensors installed.
	salinity_sensors	2	integer, 0...3		Number of salinity measurement mission sensors installed.
	conductivity_sensors	2	integer, 0...3		Number of conductivity measurement mission sensors installed.
	dissolved_oxygen_sensors	2	integer, 0...3		Number of DO mission sensors installed.
	fluorometer_sensors	2	integer, 0...3		Number of fluorometer mission sensors installed.
	turbidity_sensors	2	integer, 0...3		Number of turbidity mission sensors installed.
	transmissometer_sensors	2	integer, 0...3		Number of transmissometer mission sensors installed.
	optical_backscatter_sensors	2	integer, 0...3		Number of optical backscatter mission sensors installed.
	dvl_sensors	2	integer, 0...3		Number of DVL mission sensors installed.
	magnetic_flux_sensors	2	integer, 0...3		Number of magnetic flux measurement mission sensors installed.
	subbottom_profiler_sensors	2	integer, 0...3		Number of sub-bottom profiler mission sensors installed.
	sidescan_sonar_sensors	2	integer, 0...3		Number of side-scan sonar mission sensors installed.
	multibeam_sonar_sensors	2	integer, 0...3		Number of multi-beam sonar mission sensors installed.
	still_cameras	2	integer, 0...3		Number of still cameras installed.
	video_cameras	2	integer, 0...3		Number of video cameras installed.
	audio_recorders	2	integer, 0...3		Number of audio recorders installed.
	-	7	-		Padding.

CCL_FILE_INF_OPT_STRUCT	3				Additional information regarding file access.
file_access_status	2	CCL_FILE_ACCESS_ENUM			Size of file [B].
file_size	9	integer, 0...511			Length of filename (single byte char).
filename_len	4	integer, 1...15			Content type of file.
content_type	3	CCL_FILE_CONTENT_ENUM			File encoding type.
encoding	1	CCL_FILE_ENCODING_ENUM			Padding.
-	5	-			

CCL_SENSOR_CAPS_INF_OPT_STRUCT	1	num_sensor_caps	3	integer, 0...7	The number of mission sensor capability descriptions in this message.
		-	5	-	Padding.

CCL_SENSOR_CAPS_STRUCT	8	type	5	CCL_MISSION_SENSOR_ENUM	The type or class of sensor in this capability description.
		instance	2	integer, 0...3	Instance of the class: 0 = all instances, 1,2,3 = specific instance.
		sensor_class	2	CCL_SENSOR_CLASS_ENUM	The class of this type of sensor.
		description_len	4	integer, 0...15	Sensor text description length.
		state	3	CCL_DEVICE_STATE_ENUM	Current state of the sensor.
		active	2	CCL_SENSOR_ACTIVE_ENUM	Trigger which activates sensor.
		directionality	1	CCL_SENSOR_DIRECTION_ENUM	Sensor directionality.
		sample_rate	1	TBD	Sample rate of sensor.
		power_level	1	TBD	Power level of sensor.
		sensitivity_threshold	1	TBD	Sensitivity threshold of sensor.
		gain	1	TBD	Gain of the sensor.
		log_rate	1	TBD	Data logging rate for the sensor.
		range	1	TBD	Effective range of sensor.
		sensitivity_at_range	1	TBD	Sensitivity at effective range.
		resolution	1	TBD	Resolution of sensor at indicated frequency.
		frequency_at_resolution	1	TBD	Current frequency of sensor.
		power_use_on_units	1	CCL_POWER_UNIT_ENUM	Units for power usage while on.
		power_use_on	10	integer, 0...1023	Power usage while on [power_use_on units].
		power_use_sleep_units	1	CCL_POWER_UNIT_ENUM	Units for power usage while sleeping.
		power_use_sleep	10	integer, 0...1023	Power usage while sleeping [power_use_sleep units].
		-	14	-	Padding.

CCL_COMMS_CAPS_INF_OPT_STRUCT	1	num_comms_caps	3	integer, 0...7	The number of communication interface capability descriptions in this message.
		-	5	-	Padding.

CCL_COMMS_CAPS_STRUCT	7	type	5	CCL_COMM_INTERFACE_ENUM	The type or class of communications interfaces in this capability description.
		instance	2	integer, 0...3	Instance of the class: 0 = all instances, 1,2,3 = specific instance.
		description_len	4	integer, 0...15	Interface text description length.
		state	3	CCL_DEVICE_STATE_ENUM	Current state of the interface.
		power_level	1	TBD	Power level of the interface.
		sensitivity_threshold	1	TBD	Sensitivity threshold of the interface.
		gain	1	TBD	Gain of the interface.
		baud_rate	1	TBD	Effective baud rate for the interface.
		range	1	TBD	Effective range of the interface.
		sensitivity_at_range	1	TBD	Sensitivity at effective range.
		resolution	1	TBD	Resolution of interface at indicated frequency.
		frequency_at_resolution	1	TBD	Current frequency used by interface.
		power_use_on_units	1	CCL_POWER_UNIT_ENUM	Units for power usage while on.
		power_use_on	10	integer, 0...1023	Power usage while on [power use on units].
		power_use_sleep_units	1	CCL_POWER_UNIT_ENUM	Units for power usage while sleeping.
		power_use_sleep	10	integer, 0...1023	Power usage while sleeping [power use sleep units].
		-	14	-	Padding.

CCL_SYS_ADM_REQ_OPT_STRUCT	1	action	3	CCL_SYS_ADM_CONTROL_ENUM	Specifies a particular system administration action to occur.
		restart_time	2	CCL_TIME_ENUM	Specify a time to restart the platform.
		-	6	-	Padding.

CCL_START_MISS_REQ_OPT_STRUCT	1	name_len	4	integer, 1...15	Number of characters in the mission name to start.
		-	4	-	Padding.

CCL_ABORT_MISS_REQ_OPT_STRUCT	1	name_len	4	integer, 0...15	Number of characters in the abort mission name to use. If = 0, use the pre-configured abort mission. If > 0, use the specified mission name.
		-	4	-	Padding.

CCL_START_BEH_REQ_OPT_STRUCT	1	name_len	4	integer, 1...15	Number of characters in the behavior name to start. Padding.
		-	4	-	
CCL_STOP_BEH_REQ_OPT_STRUCT	1	name_len	4	integer, 1...15	Number of characters in the behavior name to stop. Padding.
		-	4	-	
CCL_UPDATE_BEH_REQ_OPT_STRUCT	3	beh_name_len	4	integer, 1...15	Number of characters in the behavior name to update.
		filename_len	4	integer, 0...15	The length of the named file used to update the behavior.
		dirname_len	5	integer, 0...31	The length of the pathname for the file. If = 0, assume a default directory.
		update_attached	1	CCL_BOOLEAN_ENUM	If = true, update the behavior using the attached byte string. If = false, use indicated file.
		beh_len	9	integer, 0...511	Length of attached behavior update code.
		beh_code_type	1	CCL_FILE_ENCODING_ENUM	Format of code.
CCL_VERSION_STRUCT	2	release	1	CCL_BOOLEAN_ENUM	If true: release version, if false: pre-release version.
		major_rev	4	integer, 0...15	Major revision number.
		minor_rev	11	integer, 0...2047	Minor revision number.
CCL_AUTHORITY_STRUCT	1	level	3	integer, 0...7	Authority levels, low to high.
		pecking_order	5	integer, 0...31	Authority sublevels, low to high.
CCL_TIME_INTERVAL_STRUCT	2	units	2	CCL_TIME_INTERVAL_ENUM	Time interval unit indicator.
		time	14	integer, 0...16383	Time interval range.
CCL_LAT_LNG_STRUCT	8	lat	4 B	float	+n/-s latitude [deg].
		lng	4 B	float	+e/-w longitude [deg].
CCL_NE_STRUCT	8	north	4 B	float	+n/-s distance from navigation origin reference point [m].
		east	4 B	float	+e/-w distance from navigation origin reference point [m].

CCL_POS_TOL_STRUCT	2	units	3	CCL_LENGTH_UNIT_ENUM	Units of length.
		val	13	integer, 0...8191	Tolerance distance on radius or depth [units].
CCL_LOOP_STRUCT	1	options	2	CCL_LOOP_ENUM	Specify type of looping behavior.
		loop_n_times	6	integer, 0...63	The number of times to loop.
CCL_PARAM_ID_STRUCT	1	parameter_id	7	integer, 0...127	A CCL_PARAMETER_ENUM or custom identifier tag denoting a parameter of interest.
		custom_flag	1	CCL_BOOLEAN_ENUM	Indicates whether the "parameter_id" is a generic parameter (false) or platform custom parameter (true).
CCL_PARAM_ACCESS_INFO_STRUCT	1	access_status	3	CCL_ACCESS_PARAM_ENUM	Indicates success or failure of monitoring or configuring a particular parameter.
		init_state	1	CCL_PARAM_STATE_ENUM	Indicates if the parameter has previously been initialized.
		-	4	-	Padding.
		group_id	2	integer, 0...3	Unique group ID; 0-2. ID 3 reserved for broadcast to all groups.
CCL_ADDRESS_STRUCT	1	host_id	6	integer, 0...63	Unique platform ID; 0-62. ID 63 reserved for broadcast to all platforms in group.

CCL_REGION_STRUCT	20	shape	2	CCL_VOLUME_GEOMETRY_ENUM	Shape of the region.
		x_constrained	1	CCL_BOOLEAN_ENUM	If true, specify ref lat/lon or offsets and x extent.
		y_constrained	1	CCL_BOOLEAN_ENUM	If true, specify ref lat/lon or offsets and y extent.
		z_constrained	1	CCL_BOOLEAN_ENUM	If true, specify ref depth and z extent.
		use_latlng	1	CCL_BOOLEAN_ENUM	If true, specify lat/lon, else navigation origin offsets.
		angle	8	integer, 0...180	Rotation angle of rectangle, ccw from north datum.
		within_region	1	CCL_BOOLEAN_ENUM	Denote if reference is to within region (true) or outside region (false).
		lat_or_n	4 B	float	Reference center, latitude [deg] or +n/-s origin offset [m].
		lng_or_e	4 B	float	Reference center, longitude [deg] or +e/-w origin offset [m].
		depth	4 B	float	Reference center depth [m].
		x_extent_units	3	CCL_LENGTH_UNIT_ENUM	Units for x extent distance.
		x_extent_val	13	integer, 0...8191	Extent of region along n/s direction (@ 0 deg rotation), or diameter, centered on reference origin.
		y_extent_units	3	CCL_LENGTH_UNIT_ENUM	Units for y extent distance.
		y_extent_val	13	integer, 0...8191	Extent of region along e/w direction (@ 0 deg rotation) centered on reference origin.
		z_extent_units	3	CCL_LENGTH_UNIT_ENUM	Units for z extent distance.
		z_extent_val	13	integer, 0...8191	Extent of region along z direction centered on reference origin.
		-	1	-	Padding.

CCL_GPS_CFG_STRUCT	2	always_get_gps_on_surface	1	CCL_BOOLEAN_ENUM	Indicate that platform should always try for GPS while on the surface.
		gps_timeout	12	integer, 0...4095	GPS timeout value [sec].
		-	3	-	Padding.

CCL_NETWORK_CFG_STRUCT	1	network_protocol	3	CCL_NETWORK_PROTOCOL_ENUM	Platform should use this network protocol for communications.
		msg_forwarding	1	CCL_BOOLEAN_ENUM	Indicator as to whether platform should/should not forward messages.
		tll_hops	4	integer, 0...15	Number of intermediate hops outgoing messages should be configured for.

CCL_LOC_SPEC_STRUCT	1	xy_loc_spec	3	CCL_HORZ_LOCATION_ENUM	Specify xy (horizontal) location type.
		z_loc_spec	4	CCL_VERT_LOCATION_ENUM	Specify z (vertical) location type.
		-	1	-	Padding.