NATO STANDARD

ANEP-82

MESSAGE SPECIFICATION FOR DATA LINK BETWEEN COMBAT MANAGEMENT SYSTEMS AND IDATS

Edition A Version 3

OCTOBER 2019

NORTH ATLANTIC TREATY ORGANIZATION

ALLIED NAVAL ENGINEERING PUBLICATION

Published by the
NATO STANDARDIZATION OFFICE (NSO)

INTENTIONALLY BLANK

NORTH ATLANTIC TREATY ORGANIZATION (NATO)

NATO STANDARDIZATION OFFICE (NSO)

NATO LETTER OF PROMULGATION

14 October 2019

1. The enclosed Allied Naval Engineering Publication ANEP-82, Edition A, Version 3, MESSAGE SPECIFICATION FOR DATA LINK BETWEEN COMBAT MANAGEMENT SYSTEMS AND IDATS, which has been approved by the nations in the NNAG, is promulgated herewith. The agreement of nations to use this publication is recorded in STANAG 4710.

2. ANEP-82, Edition A, Version 3, is effective upon receipt and supersedes ANEP-82, Edition A, Version 2, which shall be destroyed in accordance with the local procedure for the destruction of documents.

3. No part of this publication may be reproduced, stored in a retrieval system, used commercially, adapted, or transmitted in any form or by any means, electronic, mechanical, photo-copying, recording or otherwise, without the prior permission of the publisher. With the exception of commercial sales, this does not apply to member or partner nations, or NATO commands and bodies.

4. This publication shall be handled in accordance with C-M(2002)60.

Zoltán GULYÁS
Brigadier General, HUNAF
Director, NATO Standardization Office

INTENTIONALLY BLANK

RESERVED FOR NATIONAL LETTER OF PROMULGATION

INTENTIONALLY BLANK

RECORD OF RESERVATIONS

CHAPTER	RECORD OF RESERVATION BY NATIONS
Note: The reservations listed on this page include only those that were recorded at time of promulgation and may not be complete. Refer to the NATO Standardization Document Database for the complete list of existing reservations.	

INTENTIONALLY BLANK

RECORD OF SPECIFIC RESERVATIONS

[nation]	[detail of reservation]
Note: The reservations listed on this page include only those that were recorded at time of promulgation and may not be complete. Refer to the NATO Standardization Document Database for the complete list of existing reservations.	

INTENTIONALLY BLANK

TABLE OF CONTENTS

INTENTIONALLY BLANK

CHAPTER 1 INTRODUCTION

1.1. PURPOSE

The purpose of this interface specification is to provide a common interface between Allied ships Combat Management Systems (CMS) and the NATO FORACS Integrated Data Acquisition Test System (IDATS). The number of vessels equipped with data highways is constantly increasing because of new vessels coming into service and from retrofitting existing vessels. There are also user-nation requirements for more sophisticated system tests. Providing a standardized output from the CMS to the IDATS increases the quality and the quantity of the data acquired.

Lack of a standardized output from CMS to IDATS will increase maintenance, operational and training costs, and endangers the quality of the data acquired.

1.2 SCOPE

The IDATS is used by the NATO FORACS Project[1] to collect and process sensor data during sensor accuracy tests. Sensor data are collected either by observing the measurements and entering them to IDATS through dedicated hand-held devices or by transmission from the ship's Combat Management System (CMS) to an IDATS Interface Computer (IDATS-IC). This document describes the protocol and the message syntax for transmission of CMS sensor data to the IDATS-IC.

The implementation of the filtering system (how it acquires the data from the CMS, how it filters them, etc.) is the CMS vendor's responsibility in accordance with the nation's interests and requirements.

1.3. ACCESS AND RELEASE OF PROCESSED SENSOR DATA

Actual national data obtained from a NATO FORACS test is normally classified by the nation concerned. Access to this national classified data in the NATO FORACS Data Bank is restricted to the NATO nation whose unit underwent the FORACS test and the NATO FORACS Office (NFO). The NFO will release data held in the NATO FORACS Data Bank only to the Government which provided the test information, unless the providing Government grants its express permission in writing for further release.

1.4 TERMS AND DEFINITIONS

FORACS	NATO Naval **For**ces Sensors and Weapons **Ac**curacy **C**heck **S**ites
GPS	**G**lobal **P**ositioning **S**ystem
IDATS	**I**ntegrated **D**ata **A**cquisition **T**est **S**ystem
IDATS-IC	IDATS **I**nterface **C**omputer
INS	**I**nertial **N**avigation **S**ystem
TPP	**T**hird **P**arty **P**rograms

1 Reference AC/141-D/252(final) FORACS Memorandum of Understanding.

UDP	**U**ser **D**atagram **P**rotocol
UTC	**U**niversal **C**oordinated Time

CHAPTER 2 DESCRIPTION

2.1. SYSTEM OVERVIEW

Data from the CMS are acquired by a dedicated filtering system which formats the data according to the specification described in this document and transmits the formatted data to an IDATS-IC.

The filtering system can be an application embedded in the CMS or an application running in a PC or workstation which has a connection between a CMS and (either serial or Ethernet) to IDATS-IC.

2.2. INTERFACE DESCRIPTION AND DIAGRAMS

Data from the CMS are acquired by a dedicated filtering system which formats the data according to the specification described in this document and transmits the formatted data to an IDATS interface PC. The transmission to the IDATS-IC can be either serial (RS-232) or Ethernet (UDP). The IDATS-IC is part of the IDATS network. It formats the data it receives from the filtering system to IDATS network packets and transmits them to the IDATS network for further processing.

The filtering system may act as a firewall to the CMS preventing any data transmission to the CMS and/or filter out any data that are not cleared for transmission to IDATS. In the former case, an approved data diode or the elimination of the receiving (Rx) part in case of RS-232 or an IP router in case of an Ethernet connection may serve that purpose.

The filtering system should also provide a mechanism to select from the numerous sensor contacts that are processed in the CMS. The selected contacts will be output to IDATS. It could also provide a mechanism to map CMS data (tracker identification numbers, the source of the data, etc.) to either predefined or user defined sensor identification names. Appendix C of this document provides a description of the basic modules needed. An example of a User Interface for these modules is also provided.

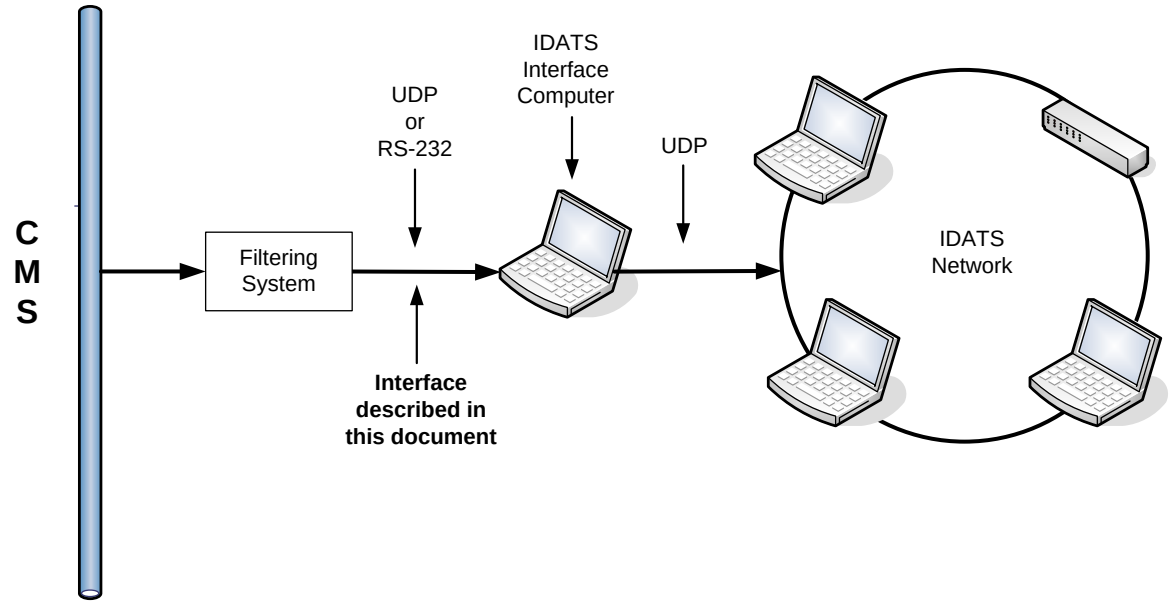


Fig. 1

FIGURE 1: INTERFACE DIAGRAM

2.3. MESSAGE PROTOCOL

There are two types of messages:

- Sensor Data Message
- Time Synchronization Message

The Sensor Data Message contains one observation from one data source.
The Time Synchronization Message contains one observation which is the current time of the CMS clock. This message is used to ensure in the CMS does not deviate substantially from the IDATS time of Day. If not, IDATS may use the time difference to adjust the time of validity in the messages that follow to align them with the IDATS time of day.

2.4. MESSAGE FREQUENCIES

The maximum frequency for the Time Synchronization Message is 0.2 Hz, while the recommended frequency is 1/60 Hz (once per minute).
The minimum frequency for a heading reference sensor message is 2 Hz, while the recommended frequency is the native rate.

2.5. ETHERNET IMPLIMENTATION

For network (Ethernet) connections, messages are sent to IDATS in UDP packets that are directed to a specific Port (default **4100)** and can be either sent to the IDATS domain broadcast address, (current default is **213.213.213.255)** or to a specific IDATS IP address defined during the system interfacing setup. The latter is the preferred method because it reduces IDATS network traffic. The port 4100 is the default port. Other port numbers should be allowed and defined during the system interfacing setup.

Each UDP packet contains one message.

2.6 SERIAL RS-232 IMPLIMENTATION

For serial (RS-232) connections, the RS-232 signal from the CMS will contain serial asynchronous characters transmitted at minimum 9600 Baud and containing 8 data bits, no parity, and 1 stop bit. Each message starts with the six characters string $SIIS, [2] (including the trailing comma), and terminates by the ASCII Line Feed (0x0A) character.

2.7. DATA MESSAGE SYNTAX

Messages from CMS to IDATS are ASCII text.
The messages consist of one or more data item segments separated by commas.
A data item segment consists of tokens and values separated by colons ':'.
Tokens are not case sensitive, except if it is specifically stated otherwise.

The tokens are:

a. A **data item descriptor** token which indicates the content of the data item segment. It is always the first token of a data item segment.
Duplicate **data item descriptors** are not allowed within the same data message.
No overload[3] of a **data item descriptor** is allowed.
A **data item descriptor** must be used only within the specific scope the ANEP indicates.

b. A token that follows a data value to qualify it. In the case of a real number the qualifier is the **units of measure** of the data value. Absence of the **units of measure** token indicates that the data item value is without unit (unit-less). If the data item value has units then the **units of measure** token should be always included[4].

c. An optional token that follows the qualifier and serves as an extra item descriptor. This token is reserved in case the data item descriptor and the qualifier tokens are not sufficient to define the measurement. This field should be thirty-two (32) or fewer characters in length. The extra item descriptor must always follow the qualifier token. Therefore in case of unit-less data values, there should be two consecutive colons (i.e. ::) between the data value and the extra item descriptor token.

---

[2] In capital letters.

[3] Overload is to assign different semantics on the same descriptor.

[4] For example, although the specification defines default units for time and angle, still the **units of measurement** token should be present when the message contains angle or time.

A **data value** is either a real number represented in ASCII by ordinary decimal notation or a character string. This field should be thirty-two (32) or fewer characters in length. In case that represents a real number it is either an integer or a decimal; no other form is allowed. If it is a decimal at least one fractional digit is required after a decimal point and one leading zero is required for proper fractional numbers. Only the ASCII characters 0-9, the plus (+), minus (-) and period (.) characters are allowed in number representation. Therefore no leading or trailing spaces are allowed. There is no actual limit in the decimal digits other than the field's maximum length of 32-characters and the limit imposed by the fact that the numerical data values are parsed (in the IDATS-IC) in the data type of the maximum precision.

Commas - which are reserved as data item segment delimiters - and colons - which are reserved as data item delimiters - cannot be part of a character string.

ASCII control characters (characters numbered below 0x20) and the NULL (\0) character cannot be part of a character string.

Leading and trailing space characters in a character string are not considered part of the string[5].

The first token of a valid data message is always either the ***time*** or the ***sensorid*** data item descriptor. The first token in the data message serves as the indicator of the type of the message[6].

When the data item descriptor '***time***' is the first token, it indicates a time synchronization message. This message is used to ensure that the time of day in the CMS does not deviate substantially from the IDATS time of Day [7]. Therefore the **time** message must carry the most recent CMS reference time and be transmitted to IDATS-IC with minimum total (acquisition + transmission) latency not exceeding 20 ms[8].

If different reference time sources are used within the CMS, then time synchronisation messages must be output for each time source and the **extra item descriptor** must be used to indicate the CMS time reference system that provided the time.

When the data item descriptor '***sensorid***' is the first token, it indicates a sensor data message. This message can have various data segments depending of the nature of the sensor (see "**Message Examples**").

When the data item descriptor '***time***' is part of a sensor data message, it indicates the time of validity of that sensor data message.

---

5 They may be present –e.g. for visual reasons- but will be ignored when the string gets parsed.

6 See **2.3.** Message Protocol.

7 IDATS records the difference between IDATS current time and CMS's time. In data processing IDATS may use the time difference to adjust the time of validity in the messages that follow to align them with the IDATS time.

8 The latency should be less than 20 ms since IDATS assumes that the message carries the actual CMS time.

A sensor data message should always contain a time of validity data segment (i.e. a "time-stamp")[9]. There must be only one time segment per message.

If different reference time sources are used within the CMS, the time of validity segment must indicate the reference time system that provided the time of validity. The **extra item descriptor** of the time of validity data segment must be used for that purpose.

A **data item descriptor** token followed by a value is the shortest data item segment allowed.

2.8. CHECKSUM

Including a checksum in the message is optional and recommended only for serial RS-232 implementations. If a checksum is included it should appear only once per message and be the last data item segment (see **2.7**). The data item descriptor for the checksum is an asterisk (*). Checksum is unit-less. The checksum number is in decimal form and it is provided by an 8 bit exclusive OR (XOR) of all characters between, including the message delimiters (i.e. commas and colons). In a serial implementation the '$' of the '$SIIS,' and the '*:' are not included in the checksum.

2.9. USER-DEFINED DATA ITEM DESCRIPTOR

A **data item descriptor** that does not match any of the approved tokens (see **2.10.)** indicates a user defined data item segment. A **user-defined data item descriptor** must be part of a valid message[10]. User defined data item segments are parsed and recorded but no further process takes place. Their purpose is to provide additional data specific to each CMS for review and further -non-FORACS- analysis. In a user defined data item segment the **extra item descriptor** is parsed but not recorded. The **user-defined data item descriptor** should be thirty-two (32) or fewer characters in length. The recommended length - to conform to the FORACS descriptors - is fewer than eight (8) and more than two (2) characters. See ANNEX B for limitations on user-defined data item descriptors nomenclature.

2.10. DATA ITEM DESCRIPTION TOKENS

If a CMS outputs data needed for FORACS data processing and those data are not covered by the current data item descriptor tokens, the list may be expanded. However, any additional data not needed for the actual FORACS data processing, but nevertheless required by the user, should be provided through the '*user-defined*' mechanism.

---

[9] If a data message lacks time of validity it cannot be processed. In extreme and well justified cases IDATS could stamp it with reception time but this mechanism will lead to incorrect (time biased) data processing, the amount of which should be calculated and established in advance of any interface deployment.

[10] See **2.7.** DATA MESSAGE SYNTAX

Descriptor	Definition
sensorid	Name of the data source. It is the first token in a sensor data message. The data source name field can be any valid string, but it has to be unique per data source and may not be changed throughout the test[11]. This field should be thirty-two (32) or fewer characters in length and the relationship between this name and the official name of the data source that produced the observation should be made known to FORACS personnel.
time	Used for both Time Of Day expressed in decimal seconds past midnight and for the UTC time expressed in decimal seconds (past 1/1/1970). It is the first token in a time synchronization message. When it is part of a sensor data message it is the sensor time of validity.
rbre	Relative Bearing to the target. Default unit is decimal degrees.
tbre	True Bearing Reading. When coming from a heading sensor, tbre is the heading information; from a non-heading sensor, tbre is the True Bearing to the target. Default unit is decimal degrees.
rnre	Range to the target
rnxre	Range to the target on X-Axis: 1. On an own ship reference 2D or 3D Cartesian system (Local Cartesian Coordinates, or LCC). 2. On an own ship reference topocentric system East-North-Up (ENU). 3. On an own ship reference topocentric system North-East-Down (NED). The **extra item descriptor** values are LCC, ENU, NED respectively. Absence of **extra item descriptor** indicates own ship reference topocentric system East-North-Up. Negative values are permitted.
rnyre	Range to the target on Y-Axis: 1. On an own ship reference 2D or 3D Cartesian system. 2. On an own ship reference topocentric system East-North-Up. 3. On an own ship reference topocentric system North-East-Down. The **extra item descriptor** values are LCC, ENU, NED respectively. Absence of **extra item descriptor** indicates own ship reference topocentric system East-North-Up (ENU). Negative values are permitted.

[11] The data source id is mapped to a FORACS sensor id; therefore it must not change throughout the test to avoid repeated re-mapping requests. Failing a re-map request could lead to data loss.

Descriptor	Definition
rnzre	Range to the target on Z-Axis: 1. On an own ship reference 2D or 3D Cartesian system. 2. On an own ship reference topocentric system East-North-Up 3. On an own ship reference topocentric system North-East-Down. The **extra item descriptor** values are LCC, ENU, NED respectively. Absence of **extra item descriptor** indicates own ship reference topocentric system East-North-Up. Negative values are permitted and indicate “DOWN” on East-North-Up and “UP” on North-East-Down.
delre	Depression/Elevation Angle Reading. Negative values indicate a depression angle reading. Default unit is decimal degrees.
htre	Height Reading. "Height" is the height above or the depth below (positive or negative values respectively): 1. The z=0 plane of an own ship reference Cartesian coordinate system. 2. The ellipsoid (ELL). 3. The mean sea level (MSL) The **extra item descriptor** values are **LCC, ELL, MSL** accordingly. Absence of **extra item descriptor** indicates height above ellipsoid. It is used only when the other location parameters are **rbre/rnre** or **latre/lonre**. Negative values to indicate depth are allowed.
latre	Latitude Reading. The default datum is WGS-84. Therefore absence of **extra item descriptor** indicates WGS-84. The following datums are allowed as **extra item descriptors**: WGS-84 ETRS89 ED79 ED50 NAD83 WGS72 OSGB36

Descriptor	Definition
lonre	Longitude Reading. The default datum is WGS-84. Therefore absence of **extra item descriptor** indicates WGS-84. The following datums are allowed as **extra item descriptors**: WGS-84 ETRS89 ED79 ED50 NAD83 WGS72 OSGB36
snrre	Sound to Noise Ratio Reading
hdre	Heading Reading. It is used whenever a message from a non-heading sensor contains a heading reference[12]. Default unit is decimal degrees.
pitch	Pitch. Default unit is decimal degrees.
roll	Roll. Default unit is decimal degrees.
scxre	Data Source offset on X-Axis on an own ship reference 2D or 3D Cartesian system.
scyre	Data Source offset on Y-Axis on an own ship reference 2D or 3D Cartesian system.
sczre	Data Source offset: 1. On the Z-Axis of an own ship reference 2D or 3D Cartesian system. 2. As height or depth from the ellipsoid 3. As height or depth from the mean sea level. The **extra item descriptor** values are **LCC, ELL, MSL** accordingly. Absence of **extra item descriptor** indicates offset on Z-Axis of a local Cartesian system. Negative values are permitted.
sentrkr	Sensor Tracker. The sensor tracker field is optional and may have any valid string as value. This field should be thirty-two (32) or fewer characters in length. It is used whenever is needed to distinguish the specific sensor internal operations[13]. It is used independently of and it must not be confused with the **systrkr.**

[12] For example, in a towed array message 'hdre' represents the heading of the array at the time of the measurement.

[13] For example, in a sonar test you can locate the same target (same **systrkr**) while you use two different beam forming algorithms (conventional and adaptive). The **sentrkr** may be used to distinguish the two algorithms.

Descriptor	Definition
spd	Speed. “Speed” may be: 1. Speed Over Ground 2. Speed Through Water The **extra item descriptor** values are **SOG**, **STW** accordingly. Absence of **extra item descriptor** indicates Speed Over Ground.
tgcrsre	Target Course Reading
tgspdre	Target Speed Reading
systrkr	System Tracker Number. It is the system wide name of the track that generated the sensor data presented in the message. If the system tracker field is not present[14], then IDATS-IC defaults it to 1. This field should be thirty-two (32) or fewer characters in length and must contain –if present- a valid string.
freq	Frequency
source	Source that generated the sensor data message. It is used whenever an observation may derive from multiple sources although within the CMS is treated as single data source observation. Typical uses are in cases of “sensor fusion” or different antennas (port, starboard, forward, aft). It should not be confused with the sensor tracker (sentrkr) which distinguishes the specific sensor internal operations. This field should be thirty-two (32) or fewer characters in length and must contain –if present- a valid string.
svmsrd	The sound velocity measured by the sensor (e.g. a sound speed probe). The default value is meters per second (i.e. m sec -1). The unit must be always present (see 2.7.)
svset	The sound velocity set in the sensor (e.g. a sonar). The default value is meters per second (i.e. m sec -1). The unit must be always present (see 2.7.)

[14] For example, a gyro message has no track to associate with, so the **systrkr** can be either a fictitious one (recommended value is ‘1’) or be omitted in which case IDATS-IC defaults it to 1.

2.11. DATA UNIT TOKENS

If in a data item segment the **data unit** token does not match an existing FORACS data unit token, the parser defaults it to “num”.
If a CMS outputs data needed for FORACS data processing in data units not covered by the current Data Unit Tokens, the list can be expanded. The full list of the Data Units Tokens used in FORACS test data products is in ANNEX B.

The default unit for angle is decimal degrees.
The default unit for time is seconds.

The following type of units are defined:
1. Basic units (e.g. seconds, degrees, meters, yards, feet etc.)
2. Coherent units (e.g. Kilometres, Kiloyards, KHz, MHz, GHz etc.)
3. Widely-used Non-Coherent units (Nautical mile, Statue mile, Knots, Hours etc.)

Unit Token	Definition
sec	Seconds
deg	Degrees
Dm	Data Miles
Ft	Feet
yd	Yards
kyd	Kiloyards
M	Meters
km	Kilometers
nm	Nautical Miles
sm	Statute Miles
hz	Hertz
khz	Kilohertz
mhz	Megahertz
ghz	Gigahertz
kn	Knots
db	Decibels
num	Number

2.12. DERIVED UNITS NOTATION

Derived units are formed by using the basic, coherent and non-coherent units defined in this document, separated by a space[15].
Exponents are allowed.
Only numbers 1-9 and the minus sign are allowed in an exponent.
The positive exponent has no sign while the negative has a minus.
A negative exponent is used to form units by division[16]. If a number follows a basic unit it is considered an exponent.

[15] Knots (Nautical miles per hour) and Hertz (cycles per second) although derived units have their own notation in this document due to their established and wide-use as knots and Hz and also because hour and cycle are not defined in this specification.
[16] For example, meters per second is defined as “m sec -1”

ANNEX A SENSOR DATA MESSAGE EXAMPLES

The following sections provide examples of messages that could be generated by the CMS based on the syntax that was described above. ***This list is indicative and does not cover all the different messages that can be output by a CMS. Nor does it define the only acceptable way to form a message from the relevant sensor. For example the systrkr token and value are optional and could have been omitted; checksum values have been omitted, etc.***

The following examples provide only the message body (valid in both Ethernet and Serial implementations).
In the case of serial RS-232 implementation, a valid serial message should include both the leading $SIIS, and the terminating LF character.

Except if it is specifically stated otherwise, tokens and string values are not case sensitive but the implementation convention follows these rules regarding character case:

a. The data item descriptors are in lower case
b. The sensorid values are in CAPITAL letters
c. The qualifiers are in lower case.
d. The user-defined descriptors are in lower case.
e: The extra item descriptors are in CAPITAL case

A.1. TIME SYNCHRONIZATION MESSAGE

time:29893.312:sec

A.2. INERTIAL NAVIGATION SYSTEM

sensorid:INS_1,time:12113.456:sec,tbre:213.949:deg

A.3 GLOBAL POSITIONING SYSTEM

sensorid:GPS3,time:12224.512:sec,latre:59.988273:deg,lonre:-17.623959:deg

A.4 ACTIVE SONAR

sensorid:SNR_1,systrkr:128a32,time:23224.543:sec,rbre:123.456:deg,rnre:12345.67:yd

A.5 PASSIVE SONAR

sensorid:SQR_19_P,systrkr:128a32,time:34865.220:sec,tbre:358.106:deg,freq:12.334:khz

A.6 PASSIVE RANGING SONAR

sensorid:PUFS,systrkr:128a32,time:31127.365:sec,rbre:256.391:deg,rnre:2301.33:yd

A.7 NAVIGATION RADAR

sensorid:NAV_RAD_1,systrkr:128a32,time:24219.111:sec,tbre:312.950:deg,rnre:1520.20:yd

A.8 HEIGHT FINDING RADAR

sensorid:HFR_SP8219,systrkr:128a32,time:1328.454:sec,tbre:12.455:deg,rnre:23113.166:yd

A.9 ESM

sensorid:8291,systrkr:128a32,time:28902.328:sec,rbre:296.2:deg,freq:8.8865:ghz

A.10 PASSIVE SONAR WITH UNIT-LESS USER-DEFINED DATA ITEM DESCRIPTOR (THREAT LEVEL –thrlvl-)

sensorid:SQR_19_P,systrkr:128,time:34865.22:sec,tbre:358.10:deg,freq:12.334:khz,thrlvl:5

ANNEX B DATA DESCRIPTORS USED IN FORACS TEST DATA PRODUCTS

B.1. DATA ITEM DESCRIPTORS

The nomenclature used in this specification for the data item descriptor tokens (see **0210**) is provided in the table below. In the column labelled "Status" a symbol is provided to delineate how it is used.
In particular:

"IN" indicates descriptors used in the specification.
"✓" indicates descriptors that may be used in future as data item descriptors tokens.
"☒" indicates descriptors reserved for specific use in the FORACS data products.
"□" indicates descriptors used in legacy data products.
Legacy descriptors shall not be used in any future revision of the specification.

User defined data item descriptors (see **0209**) shall not use the data descriptors provided in the table below either partially or as they are.
The list of the data item descriptors is amended when a new entry is required to cover a new data type.

Data Item Descriptor	Description	Status	Comments
Attr	DATA_ATTRIBUTE	☒	Reserved for plot attributes
corfa	CORRECTION_FACTOR	☒	Reserved
cumper	CUMULATIVE_%	☒	Reserved as FORACS data processing product
d_theo	DECK_THEODOLITE_READING	✓	
delay	DELAY	☒	Reserved as FORACS actual
delre	DEPRESSION_ELEVATION_READ	IN	
disper	DISPLACEMENT_FROM_ACTUAL_FIX	☒	Reserved as FORACS data processing product
doppac	DOPPLER_ACTUAL	☒	Reserved as FORACS actual
dopper	DOPPLER_ERROR	☒	Reserved as FORACS data processing product
doppre	DOPPLER_READ	✓	
Elac	ACTUAL_ELEVATION	☒	Reserved as FORACS actual
Eler	ELEVATION_ERROR	☒	Reserved as FORACS data processing product

Data Item Descriptor	Description	Status	Comments
Elre	ELEVATION_READ	☒	It has been superseded by the 'htre' descriptor.
event	FORACS TEST EVENT	☒	Reserved for data processing
g1ac	ACTUAL_GYRO_1	☒	Reserved as FORACS actual
g1er	GYRO_1_ERROR	☒	Reserved as FORACS data processing product
g1re	GYRO_1_READ	☐	
g2ac	ACTUAL_GYRO_2	☒	Reserved as FORACS actual
g2er	GYRO_2_ERROR	☒	Reserved as FORACS data processing product
g2re	GYRO_2_READ	☐	
Gcor	GYRO_CORRECTED	☒	Reserved as FORACS data processing product
gyrac	ACTUAL_GYRO	☒	Reserved as FORACS actual
gyrcrs	GYRO_COARSE	✓	
gyrer	GYRO_ERROR	☒	Reserved as FORACS data processing product
gyrfin	GYRO_FINE	✓	
gyrre	GYRO_READ	✓	
hdac	ACTUAL_HEADING	☒	Reserved as FORACS actual
hder	HEADING_ERROR	☒	Reserved as FORACS data processing product
hdop	HORIZONTAL_DILUTION_OF_POSITION	✓	
hrate	HEADING_RATE	☒	Reserved as FORACS data processing product
Htac	ACTUAL_HEIGHT	☒	Reserved as FORACS actual
Hter	HEIGHT_ERROR	☒	Reserved as FORACS data processing product
Htre	HEIGHT_READ	IN	
latac	ACTUAL_LATITUDE	☒	Reserved as FORACS actual
Latre	LATITUDE_READ	IN	
Later	LATITUDE_ERROR	☒	Reserved as FORACS data processing product
lonac	ACTUAL_LONGITUDE	☒	Reserved as FORACS actual
loner	LONGITUDE_ERROR	☒	Reserved as FORACS data processing product
lonre	LONGITUDE_READ	IN	
mark	MARK	✓	
noise	NOISE	✓	
nrber	NORMALIZED_RELATIVE_BEARING_ERROR	☒	Reserved as FORACS data processing product
ntber	NORMALIZED_TRUE_BEARING_ERROR	☒	Reserved as FORACS data processing product

Data Item Descriptor	Description	Status	Comments
pcode	PLOT_CODE	☒	Reserved for data processing
pctrng	PERCENTAGE_RANGE_ERROR	☒	Reserved as FORACS data processing product
perac	ACTUAL_PERISCOPE	☒	Reserved as FORACS actual
perer	PERISCOPE_ERROR	☒	Reserved as FORACS data processing product
perre	PERISCOPE_READ	✓	
pitch	PITCH	IN	
rbac	ACTUAL_RELATIVE_BEARING	☒	Reserved as FORACS actual
rber	RELATIVE_BEARING_ERROR	☒	Reserved as FORACS data processing product
rbrate	RELATIVE_BEARING_RATE	☒	Reserved as FORACS data processing product
rbre	RELATIVE_BEARING_READ	IN	
rnact	ACTUAL_RANGE	☒	Reserved as FORACS actual
rnadj	RANGE_ADJUSTED	☒	Reserved as FORACS actual
rner	RANGE_ERROR	☒	Reserved as FORACS data processing product
rngscale	RANGE_SCALE	✓	
rnre	RANGE_READ	IN	
roll	ROLL	IN	
rrate	RANGE_RATE	☒	Reserved as FORACS data processing product
rnxac	ACTUAL_X_POSITION	☒	Reserved as FORACS actual
rnxer	X_POSITION_ERROR	☒	Reserved as FORACS data processing product
rnxre	X_POSITION_READ	✓	
rnyac	ACTUAL_Y_POSITION	☒	Reserved as FORACS actual
rnyer	Y_POSITION_ERROR	☒	Reserved as FORACS data processing product
rnyre	Y_POSITION_READ	✓	
rnzac	ACTUAL_Z_POSITION	☒	Reserved as FORACS actual
rnzer	Z_POSITION_ERROR	☒	Reserved as FORACS data processing product
rnzre	Z_POSITION_READ	✓	
s_theo	SHORE_THEODOLITE	☒	Reserved as FORACS data processing product
scert	TARGET_POSITION_ERROR	☒	Reserved as FORACS data processing product
scerx	DATA_SOURCE_X_POSITION_ERROR	☒	Reserved as FORACS data processing product
scery	DATA_SOURCE_Y_POSITION_ERROR	☒	Reserved as FORACS data processing product

Data Item Descriptor	Description	Status	Comments
scxac	DATA_SOURCE_X_POSITION_ACTUAL	☒	Reserved as FORACS actual
scxre	DATA_SOURCE_X_POSITION_READ	✓	
scyac	DATA_SOURCE_Y_POSITION_ACTUAL	☒	Reserved as FORACS actual
scyre	DATA_SOURCE_Y_POSITION_READ	✓	
sczac	DATA_SOURCE_Y_POSITION_ACTUAL	☒	Reserved as FORACS actual
sczer	DATA_SOURCE_Y_POSITION_ERROR	☒	Reserved as FORACS data processing product
sczre	DATA_SOURCE_Y_POSITION_READ	✓	
sensorid	SENSOR_ID	IN	
sentrkr	SENSOR_TRACKER	IN	
ship_h	SHIP_HEADING	✓	
ship_x	X_POSITION_OF_THE_SHIP	✓	
ship_xv	X_VELOCITY_OF_THE_SHIP	✓	
ship_y	Y_POSITION_OF_THE_SHIP	✓	
ship_yv	Y_VELOCITY_OF_THE_SHIP	✓	
ship_z	Z_POSITION_OF_THE_SHIP	✓	
ship_zv	Z_VELOCITY_OF_THE_SHIP	✓	
skip	NO_USEFULL_DATA	☒	Reserved for data exclusion
snrre	SIGNAL_NOISE_RATIO	IN	
spd	SPEED	IN	
svmsrd	SOUND_VEL_MSRD	✓	
svset	SOUND_VEL_SET	✓	
systrkr	SYSTEM_TRACKER	IN	
target	TARGET_NUMBER	☒	Reserved as FORACS data processing input
tbac	ACTUAL_TRUE_BEARING	☒	Reserved as FORACS actual
tber	TRUE_BEARING_ERROR	☒	Reserved as FORACS data processing product
tbrate	TRUE_BEARING_RATE	☒	Reserved as FORACS data processing product
tbre	TRUE_BEARING_READ	IN	
theo_1	THEODOLITE_1_READING	✓	
theo_2	THEODOLITE_2_READING	✓	
theo_3	THEODOLITE_3_READING	✓	
time	TIME	IN	
utc_time	UTC_TIME	☒	In the specification the ‘time’ descriptor covers both Time of Day and UTC Time.
validity	VALIDITY	✓	
*	CHECKSUM	IN	

B.1. DATA UNITS

The nomenclature used in this specification for the data units tokens (see **2.11.**) is provided in the table below. In the column labelled "Status" a symbol is provided to delineate how it is used.
In particular:

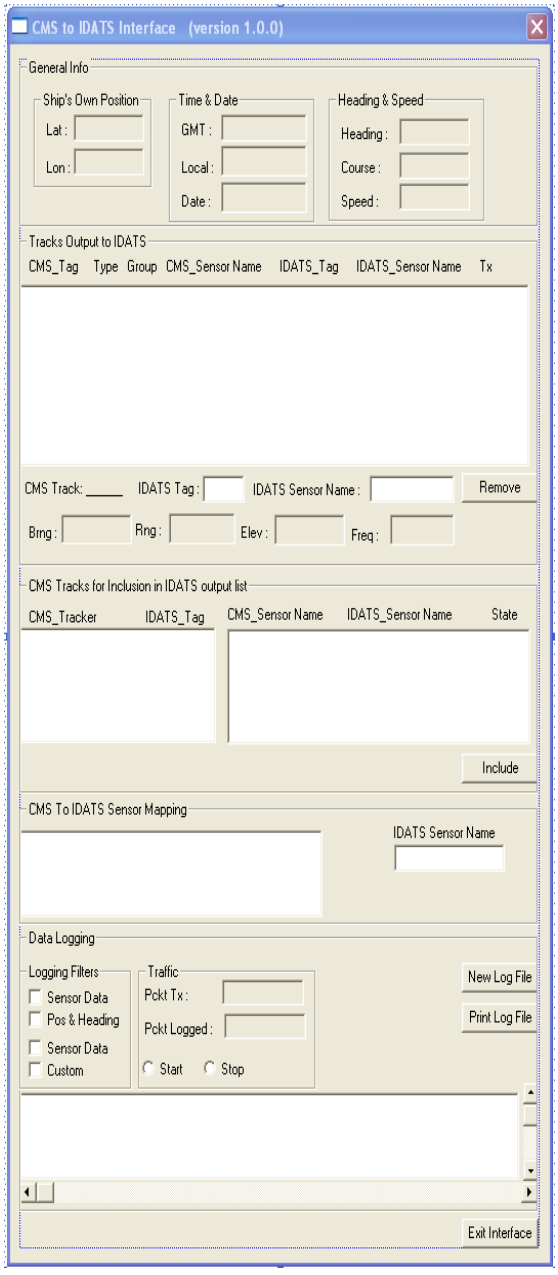
"IN" indicates units used in the specification.
"✓" indicates units that may be used in future as data item descriptors tokens.
"☒" indicates units reserved for specific use in the FORACS data products.

Unit	Description	Status	Comments
Deg	DEGREES	IN	
hour	HOURS	☒	The unit for time in the specification is seconds.
M	METERS	IN	
Km	KMETERS	IN	
Yd	YARDS	IN	
kyd	KYARDS	IN	
Nm	NAUTICAL MILES	IN	
Dm	DATA MILES	IN	
num	NUMBER	IN	
sec	SECONDS	IN	
Ft	FEET	IN	
Sm	STATUTE MILES	IN	
Db	DECIBEL	IN	
msec	MSECONDS	✓	
amin	ARC MINUTES	☒	The unit for angles in the specification is degrees.
asec	ARC SECONDS	☒	The unit for angles in the specification is degrees
Kn	KNOTS	IN	
Hz	HERTZ	IN	
khz	KILOHERTZ	IN	
mhz	MEGAHERTZ	IN	
ghz	GIGAHERTZ	IN	

INTENTIONALLY BLANK

ANNEX C FILTER PC USER INTERFACE EXAMPLE

The window below consists of five distinctive areas that cover the necessary functionality of the GUI of a CMS Interface to IDATS. These areas could be separate windows accessed from a main menu or any other GUI arrangement. The functionality described below serves explanatory purposes and does not necessarily dictate or recommend a specific GUI design.



General Info

Data like ship's position taken from the CMS are displayed for information and for verification of the operation of the interface.

Tracks Output to IDATS

Sensor contacts that are currently output to IDATS are monitored. Individual contacts selected through the list can be removed, assigned to a different IDATS Tag and/or IDATS Sensor Name. Individual contacts selected through the list present their quantities when applicable (Bearing, Range, Elevation, Frequency values).

CMS Tracks for inclusion in IDATS output

All the sensor contacts processed in the CMS and cleared for output to IDATS will be displayed on this (part of the) window. The user will select individual contacts to be output to IDATS. Sensor contacts selected in this list automatically appear in the above "**Tracks Output to IDATS**" list.

CMS to IDATS Sensor Mapping

IDATS sensor names are based on the agreed, full and detailed nomenclature each nation provides to NATO FORACS. Sensor names in the CMS differ from the ones that IDATS uses. This part of the window provides the mapping mechanism between IDATS and CMS sensor names. IDATS provides a similar mechanism; therefore, this part of the interface can be omitted.

Data Logging

As a minimum requirement, data output to IDATS should - at the user discretion - be logged to a file. The amount of data logged is chosen through an exclusion filter mechanism (i.e. by default all the data output is logged).

ANEP-82(A)(3)