Interface Control Document
Reference System Trace format

Prepared (Author)

Signature : **Date :**

Name / Function

Pierre CUQ
Reference System Engineer

Fabien CRAHEIX
Reference System Engineer

on behalf of SII

Approved

Signature : **Date :**

Name / Function

Jean-Luc PASCAL
Reference System Quality Manager

Pierre CUQ
Reference System Engineer

Released

Signature : **Date :**

Name / Function

Benjamin BRARD

Reference System project manager

Document type	Nb WBS	Keywords
ICD	XXX	

CMMI-SVC	CM

Export Control Information

This document contains EU or/and US Export Controlled technology (data) :

YES ☐ NO ☒

If **YES** :

1/ European/French regulation controlled content

☐ Technology contained in this document is controlled by the European Union in accordance with dual-use regulation 428/2009 under Export Control Classification Number [x**E**xx]. ***(1)***

☐ Technology contained in this document is controlled by Export Control regulations of French Munitions List under Export Control Classification Number [MLXX or AMAXX]. ***(1)***

2/ US Regulation controlled content

☐ Technology contained in this document is controlled under Export Control Classification Number [x**E**xxx] by the U.S. Department of Commerce - Export Administration Regulations (EAR). ***(1)***

☐ Technology contained in this document is controlled by the U.S. Department of State - Directorate of Defense Trade Controls - International Traffic in Arms Regulations (ITAR). ***(1)***

(1) See applicable export control license/authorization/exception in Delivery Dispatch Note.

SUMMARY

This ICD explains how an application must format its traces for the Reference System platform, so all needed information for monitoring and performance analysis can be computed.

CHANGE LOG

Issue/Revision	Date	Change Requests	Observations
01	30 Aug 2021	N/A	Derived from S1PDGS cloud system ICD S1PD-ICD-ADST-1000585123 issue 4.0. Add "mission identifier" field on header Remove intermediate step (only BEGIN & END are remaining) remove "step" field
02	07 Jun 2022	N/A	No change.
3.0 draft A	28/06/2022		• Remove Kubernetes section • Set BEGIN and END with uppercase. • Add missing_output optional section • Add rs_chain_name & rs_chain_version on the header • Provide sample with status NOK and missing_output section
3.0 draft B	30/06/2022		• Increase message maximum size to 10 Kbytes • Update wording for “missing output”
3.0	05/09/2022	§2.3	• Increase message maximum size to 250 Kbytes • **Remove draft.** Release for RS system v1.1.

TABLE OF CONTENTS

LIST OF FIGURES

No table of figures entries found.

LIST OF TABLES

No table of figures entries found.

1 GENERAL

1.1 SOURCE

Derived from S1PD-ICD-ADST-1000585123 issue 4.0

1.2 PURPOSE

The TRACE ICD explains how an application must format its traces, so all needed information for monitoring and performance analysis can be computed.

2 FORMAT

The application must write all traces to the standard outputs **stdout** and **stderr**.

2.1 SECTIONS

Each message is in **JSON** format to facilitate integration and analysis. The content of the message should be as follows :

section	mandatory	comment
header	YES	common information about the trace
message	YES	the message itself
task	NO	Section to be filled if the trace is linked to a task. There are two kinds of TRACE: "**On the fly**" TRACE which refers to an isolated event, TRACE linked to a **task** , described on an additional "task" section.
custom	NO	custom fields

2.2 HEADER

key	description	mandatory	value sample	format
type	message type	YES	"REPORT"	string "REPORT"
timestamp	message date and hours (UTC)	YES	"2019-01-21T05:24:40.000000Z"	<YYYY>-<MM>-<DD>T<hh>:<mm>:<ss>.<microseconds>Z
level	message level	YES	"WARNING"	string "INFO" | "WARNING" | "ERROR" | "DEBUG" | "FATAL"
mission	mission identifier	YES	"S1", "S2", "S3"	string "S1" | "S2" | "S3"
rs_chain_name	RS add-on or RS core name	NO	“s1_aio”	string maximum size of 64 UTF-8 character format

key	description	mandatory	value sample	format
rs_chain_version	RS add-on or RS core version	NO	"2.0.1"	string X.Y.Z with X is the major version id Y is the minor version id Z is the patch version id
workflow	processing workflow follow-up	NO	"NOMINAL"	string "NOMINAL" | "EXTERNAL_DEMAND" | "EXTERNAL_CUSTOM_DEMAND" | "OPERATOR_DEMAND" By default, the workflow is equal to "NOMINAL".
debug_mode	debug mode	NO	"false"	boolean true or false by default debug_mode is false
tag_list	list of tags	NO	["TAG1" , "TAG2"]	array of string

2.3 MESSAGE

key	description	mandatory	value sample	format
content	message itself	YES	"free text..."	string maximum size of 250 Kbytes UTF-8 character format

2.4 TASK

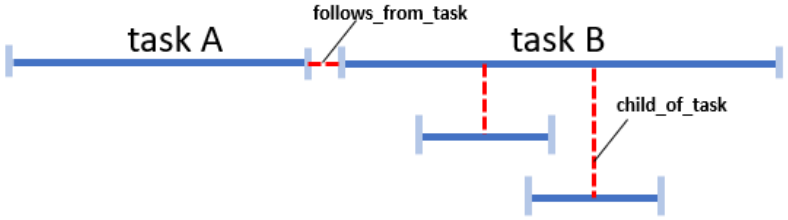
When a trace is linked to a task, it embeds a nested "**task**" tag.

2.4.1 Task event

Each task is linked to “BEGIN” and “END” events.

2.4.2 Link between tasks

The optional fields "**follows_from_task**" and "**child_of_task**" provide information about links between tasks.



"**child_of_task**" field deals with child/father relationships. The child task is executed inside the father task time frame.

"**follows_from_task**" deals with tasks that have a time execution dependency. The task B starts after task A in the example above.

2.4.3 Task content

As a consequence, the "**task**" nested tag will host following keys:

key	description	mandatory	value sample	format
uid	unique task identifier	YES	"34995e37-196c-4332-8075-c6c6891a69cc"	string length 36 characters pattern: [0-9a-fA-F]{8}\-[0-9a-fA-F]{4}\-[0-9a-fA-F]{4}\-[0-9a-fA-F]{4}\-[0-9a-fA-F]{12} randomly generated for each new task
name	task name	YES	"job_processing"	string UTF-8 character format maximum length 256 characters
event	task event	YES	"BEGIN"	string "BEGIN" | "END"
data_rate_mebibytes_sec	total speed of data handled by the task step	NO	"783740.123"	double >= 0 Mebibytes / seconds (1024 * 1024 bytes per seconds) Precision millisecond. i.e. 0,001
data_volume_mebibytes	total size of data handled by an the task step	NO	"783740.123"	double >= 0 Mebibytes (1024 * 1024 bytes) Precision 3 digits (0,001 Mebibyte)
satellite	satellite identifier	NO	"S1C"	String 3 characters Example: "S1A", "S1B", "S1C"

2.4.4 Task "BEGIN"

The task with event "**BEGIN**" embeds the nested tag "**input**", with all input useful for task starting.
The two optional fields "**child_of_task**" and "**follows_from_task**" provide information about links between tasks.
Either "**child_of_task**" and "**follows_from_task**" shall be used. Not both at the same time.

key	description	mandatory	value sample	format

input	inputs used
	by the task
		YES	{
			"key1" : "value1" ,
			"key2" : "value2"
			}
				Each {"key" : "value" pair complies with the format described
				on §"**Custom key/value format".**
				
				The array can be empty if no input is used by the task.

child_of_task	UID of the task that has created the current task	NO	"34978e37-296c-4332-8075-c6c6891a69ee"	string length 36 characters pattern: [0-9a-fA-F]{8}\-[0-9a-fA-F]{4}\-[0-9a-fA-F]{4}\-[0-9a-fA-F]{4}\-[0-9a-fA-F]{12} randomly generated for each new task
follows_from_task	UID of the predecessor task	NO	"24478f38-207c-0098-9275-c6c6781b69ac"	string length 36 characters pattern: [0-9a-fA-F]{8}\-[0-9a-fA-F]{4}\-[0-9a-fA-F]{4}\-[0-9a-fA-F]{4}\-[0-9a-fA-F]{12} randomly generated for each new task

2.4.5 Task "END"

The task with event "**END**" embeds the nested tag "**output**", "**input**" and "**quality**", with all available information at the end of the task.

The "**input**" tag is **a strict copy** of the "input" tag from task "**BEGIN**".

key	description	mandatory	value sample	format
status	status of the achieved task	YES	"OK"	string "OK" | "NOK" | "TIMEOUT"
error_code	error code at the end of the task processing.	YES	"0"	integer "0" for status "OK" ">0" for status "NOK"
duration_in_seconds	task duration in seconds	YES	"342.1231"	double Precision microsecond. i.e. 0.000001 (10-6 sec)
input	inputs used by the task (copy of input from task "BEGIN")	YES	{ "key1" : "value1", "key2" : "value2"}	Each {"key" : "value" pair complies with the format described on §"Custom key/value format". The array can be empty if no input is used by the task.

output	available output information
	generated by the task
		YES	{
			"key1" : "value1" ,
			"key2" : "value2"
			}
				Each {"key" : "value"} pair complies
				with the format described on
				§"Custom key/value format".
				The array can be empty if no
				input is used by the task.

quality	quality information available
	at the end of the task
		YES	{
			"key1" : "value1" ,
			"key2" : "value2"
			}
				Each {"key" : "value"} pair complies
				with the format described on
				§"Custom key/value format".
				The array can be empty if no
				input is used by the task.

key	description	mandatory	value sample	format

missing_output	Estimate of missing output elements	NO	(*)	If some outputs are missing, this section shall be filled. The format is a JSON array of elements
				{ "product_metadata_custom_object":
				{
				"key1" : "value1",
				"key2" : "value2" …
				},
				"end_to_end_product_boolean": <boolean>,
				"estimated_count_integer": <integer>
				}
				• **key / value** pair complies with the format described on §"Custom key/value format".
				• **end_to_end_product_boolean** identifies if the missing products are for the end user or not.
				• **estimated_count_integer** is an estimate of missing products that corresponds to the description form **product_metadata_custom_object**

(*) Example :

```
"missing_output": [
    {
      "product_metadata_custom_object": {
        "instrument_mode_string": "EW",
        "product_type_string": "SLC",
        "resolution_class_string": "",
        "polarisation_string": "DH",
        "product_class_string": "S",
        "processing_level_integer": 1,
      },
      "end_to_end_product_boolean": true,
      "estimated_count_integer": 14
    },
    {
     "product_metadata_custom_object": {
        "instrument_mode_string": "EW",
        "product_type_string": "GRD",
        "resolution_class_string": "",
        "polarisation_string": "DH",
        "product_class_string": "S",
        "processing_level_integer": 1,
      },
      "end_to_end_product_boolean": true,
      "estimated_count_integer": 2
    }
  ]
```

2.5 APPLICATION CUSTOM FIELDS

The application can add custom fields. These fields are nested on a tag named "custom" using JSON format.

This additional section is facultative. It respects the following format :

```
"custom": {
    "key1" : "value1",
    "key2" : "value2",
      ...
    "keyN" : "valueN"
}
```

Each {"key" : "value"} pair complies with the format described on §**"Custom key/value format".**

2.6 CUSTOM KEY/VALUE FORMAT

This chapter describes the custom {"**key**":"**value**"} format.

2.6.1 Key

"**key"** is formatted following these rules:

rule	description
snake_notation	Each key is a string field. 256 characters maximum length. It uses a snake_notation : all word lowercase written and separated with underscore character (_) and no space.
single key suffix	The suffix provides information about key value type: - numeric type: "_long" | "_integer" | "_short" | "_byte" | "_double" | "_float" - other type: "_boolean" | "_string" | "_date" | "_object"
array key suffix	The suffix provides information about the key value type for **all the array elements.** The array key suffix is identical to the single key suffix with the extra '**s**' at the end. - array of numeric type : "_long**s**" | "_integer**s**" | "_short**s**" | "_byte**s**" | "_double**s**" | "_float**s**" - array of other type: "_boolean**s**" | "_string**s**" | "_date**s**" | "_object**s**"

2.6.2 Value

One single "**value"** can have one of the following types:

type	description
numeric value	long, integer, short, byte, double, float
boolean	true | false
string	UTF-8 character format / maximum size of 2048

type	description
date	UTC only <YYYY>-<MM>-<DD>T<hh>:<mm>:<ss>.<microseconds>Z
object	{"key1" : "value1", "key2" : "value2", ... "keyN" : "valueN"} Each "key" embedded on an object complies to the same rules as a single key.

An array has a format : **"key" : ["value1", "value2", "value3"]** where all values have the same type.

For example: "my_key_string**s**" : ["a1", "abcd", "Test123"]


Array type must be avoided as much as possible because It prevents introspection from monitoring tools.


When the array only contains one element, it shall be replaced by a "**single**" value.

For example, filename_string**s** shall contain more than one element.

2.6.3 Example

```
{
"custom": {
     "my_param0_float": "12.34",
     "my_filename_strings": [ "temp1.txt", "tmp.csv","tmp.xml"],
     "my_param1_object": {
       "item1_long": "123",
       "item2_boolean": "false
       },
     "sample_date": "2019-01-21T05:24:40.000000Z"
     }
}
```

2.7 FULL SAMPLES

Task BEGIN


```
{
  "header": {
    "type": "REPORT",
    "timestamp": "2021-08-30T15:02:24.125000Z",
    "level": "INFO",
    "mission": "S3",
    "rs_chain_name": "s3_acq",
    "rs_chain_version": "1.2.3",
    "workflow": "NOMINAL"
  },
  "message": {
    "content": "Start compression processing"
  },
  "task": {
    "uid": "4cb9fa49-2c0a-4363-82c3-ea9ab223c53a",
    "name": "CompressionProcessing",
    "event": "BEGIN",
    "input": {},
    "follows_from_task": "a66d3ac2-2483-4891-8151-1bc77e4296e8"
  }
}
```


Trace END


```
{
  "header": {
    "type": "REPORT",
    "timestamp": "2021-08-30T15:10:41.504000Z",
    "level": "INFO",
    "rs_chain_name": "s2_L0u",
    "rs_chain_version": "1.4.5",
    "mission": "S2",
    "workflow": "NOMINAL"
  },
  "message": {
    "content": "End compression processing"
  },
  "task": {
    "uid": "43c95d84-2709-4aca-98a9-d056ba97fc2e",
    "name": "CompressionProcessing",
    "event": "END",
    "status": "OK",
    "output": {
      "filename_string":
"S1B_OPER_REP_STNACQ_SGS__20210830T150224_V20210830T145043_20210830T145217.EOF"
    },
    "input": {},
    "quality": {},
    "error_code": 0,
    "duration_in_seconds": 0.061999
  }
}
```

Same example with a NOK status.


```
{
  "header": {
    "type": "REPORT",
    "timestamp": "2021-08-30T15:10:41.504000Z",
    "level": "INFO",
    "rs_chain_name": "s2_L0u",
    "rs_chain_version": "1.4.5",
    "mission": "S2",
    "workflow": "NOMINAL"
  },
  "message": {
    "content": "End compression processing"
  },
  "task": {
    "uid": "43c95d84-2709-4aca-98a9-d056ba97fc2e",
    "name": "CompressionProcessing",
    "event": "END",
    "status": "NOK",
    "output": {},
    "input": {},
    "quality": {},
    "output": {},
    "missing_output": [
            {
              "product_metadata_custom_object": {
                "instrument_mode_string": "EW",
                "product_type_string": "SLC",
                "resolution_class_string": "",
                "polarisation_string": "DH",
                "product_class_string": "S",
                "processing_level_integer": 1,
              },
              "end_to_end_product_boolean": true,
              "estimated_count_integer": 14
            },
            {
              "product_metadata_custom_object": {
                "instrument_mode_string": "EW",
                "product_type_string": "GRD",
                "resolution_class_string": "",
                "polarisation_string": "DH",
                "product_class_string": "S",
                "processing_level_integer": 1,
              },
              "end_to_end_product_boolean": true,
              "estimated_count_integer": 2
            }
    ],
    "error_code": 0,
    "duration_in_seconds": 0.061999
  }
}
```

DISTRIBUTION LIST

	Overall document		Summary
	Action	Information	
ESA project team		X	
ADS project team	X		
CS-FR project team	X		
Werum project team	X		