Interface Control Document
Reference System add-on

Prepared (Author)

Signature : **Date :**

Name / Function **Pierre CUQ**
Reference System Engineer

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 [xExx]. ***(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 [xExxx] 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.

Dissemination is only allowed to legal or natural persons with right to know who are covered by an appropriate export license/authorization/exception.

SUMMARY

This ICD describes the Reference System add-on component format.

CHANGE LOG

Issue/Revision	Date	Change Requests	Observations
01 draft A	04 / 01 / 2022		First version
01 draft B	31 / 01 / 2022	SSN_002, FSI_001, FSI_005, FSI_006, FSI_009	Take into account RIDs from Reading_sheet-doc-COPRS-ICD-ADST-001133963 Update Artifactory directories.
01	09 /03 / 2022		Update after RIDS meeting. • Remove deploymentLabels property. • Namespace is no more immutable. • Add mongodb url. • Add sample chapter • Add rule for topic naming. • Update chapters number.
02	02 / 05 / 2022	§4.5.2	Update : Allow multi-line DSL definition.
03	05 / 05 / 2022	§3.4.3 §3.4.4 §4.5.2.5 §4.5	Set Kafka, secret and ES URL (cf #373) Add wiring chapter to define topic name. Remove “description” and “name” fields. Add comment format.
04	13 / 05 /2022	§3.4.4	Set single topic name for error-warning.
05	16 / 06 / 2022	§4.1, §4.2, §4.4, §5	Remove field [%DATE%] from filenames.

TABLE OF CONTENTS

1 GENERAL INFORMATION

1.1 DOCUMENT PURPOSE

This document describes the Reference System add-on format.

1.2 GLOSSARY AND ABREVIATIONS

Term	Meaning
CFI	Customer Furnished Item.
DPU	A Data Process Unit is a specific Process Unit that wraps a Sentinel Data Processor binaries and their static data.
DSL	Domain Specific Language
PU	A Process Unit is a micro-service, packaged as ready-to-be-deployed Docker, representing the simplest step that can be included into a Reference System workflow.
RS	Reference System
RS add-on	A RS add-on is an autonomous package ready for deployment on cloud on top of RS platform. It provide a Sentinel processing chain.
SCDF	Spring Cloud Data Flow is the workflow manager for RS platform.
SDP	A Sentinel Data Processors is a processing CFI provided by ESA.
Workflow manager	The workflow manager is responsible for creating a workflow instance from its source template. In particular, it gathers configuration values, negotiates execution resources and downloads the required Process Unit on the target nodes. SCDF is the workflow manager for RS platform.

2 APPLICABLES AND REFERENCE DOCUMENTS

2.1 APPLICABLE DOCUMENTS

Identifier	Document name	Reference	Version
[ICD1]	Sentinel Data Processor delivery ICD	ESA-EOPG-EOPGC-TN-55	1.1
[RD1]	Non functional requirements for Cloud Deployment & Configuration	COPRS-SP-ADST-001046261	1.0

2.2 REFERENCE DOCUMENTS

N/A

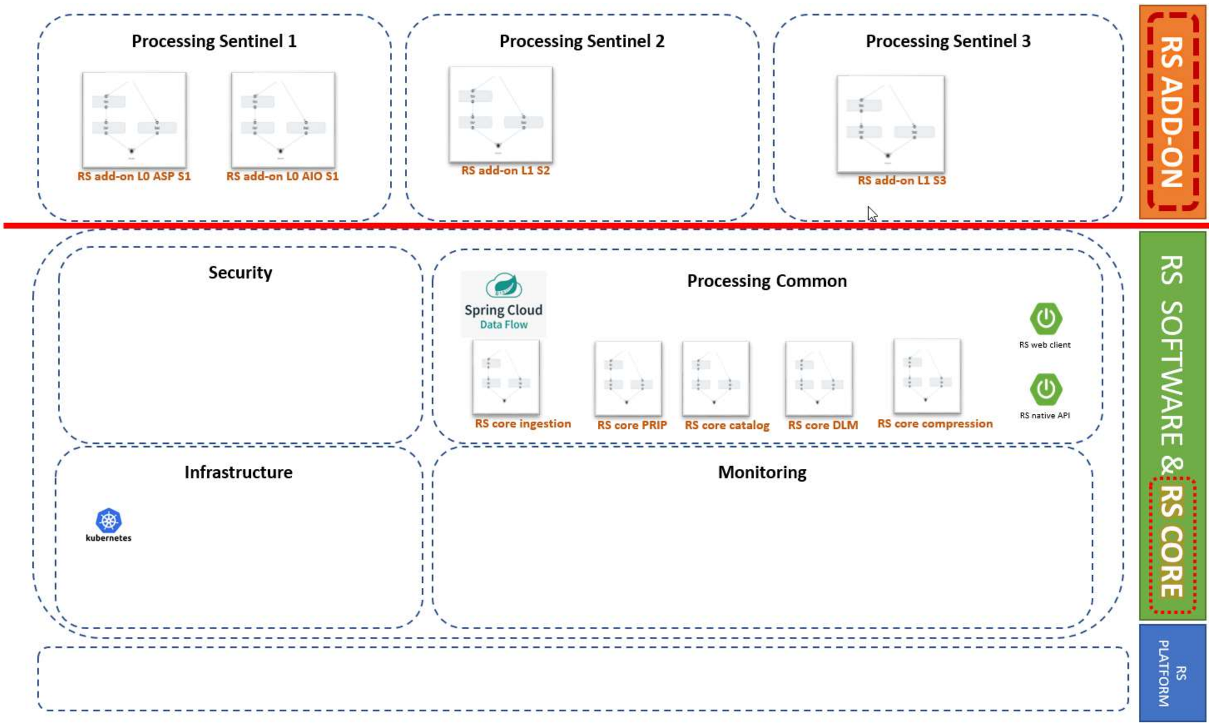
3 RS ADD-ON INTO REFERENCE SYSTEM

3.1 REFERENCE SYSTEM

Copernicus Reference System run Sentinel 1, Sentinel 2 and Sentinel 3 workflows. A workflow is provided as a RS add-on. Workflows on RS platform are managed by Spring Cloud Data Flow.

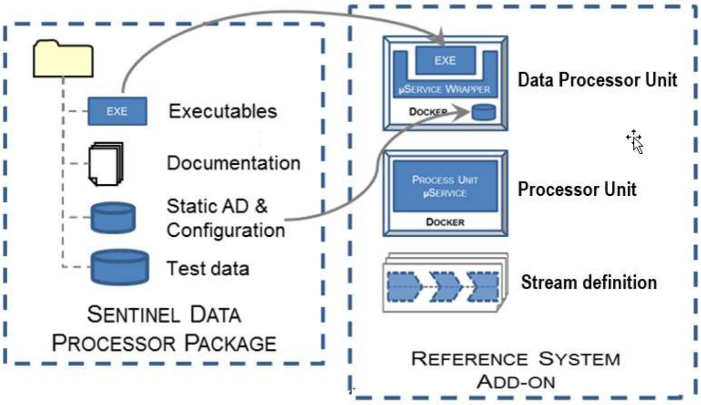
3.2 RS ADD-ON

A RS add-on provides a Sentinel processing chain. It is deployed on top of the RS platform.



A RS add-on will embed two kinds of processors :

- Processor Unit : mainly trigger and prepare worker.
- Data Process Unit : mainly execution worker. The execution worker wraps a Sentinel Data Processor binaries and their static data.



⚠ The RS add-on is provided with factory settings. The chain need to be instanciated with operational setting for deployment.

3.3 RS ADD-ON STORAGE

- ESA delivers Sentinel Data Packs into COPRS Function 1.
- COPRS Fonction 3 transforms SDP into DPU containers and push it to Function 2 Artifactory into a protected area (credentials are needed to retrieve them).
- COPRS Fonction 3 builds RS add-on package that point to both DPU & PU containers stored in Function 2 Artifactory.
- Source code RS add-on package is versioned on Function 2 GitHub.
  RS add-on payload is also stored into COPRS Function 1 storage. The procedure to store a data pack into Function 1 is described on [ICD1], §4.5.
  Only ESA authorized customers can download RS add-on package from COPRS Function 1.

The table bellows summarizes storage location for both Sentinel Data Pack and RS add-on.

	Function 1 storage	Function 2 Artifactory	Function 2 GitHub
Sentinel Data Pack	Full package	Credendial protected CFI containers	
RS add-on	RS add-on payload only	Credendial protected DPU containers	source code template properties files

3.4 RS PLATFORM

3.4.1 Compatibility

RS add-on components shall be compliant with :

FOSS	Version
SCDF	>= 2.9.1

3.4.2 Container

All container that are part of RS add-on shall fulfill requirements from [RD1], applicable to container.

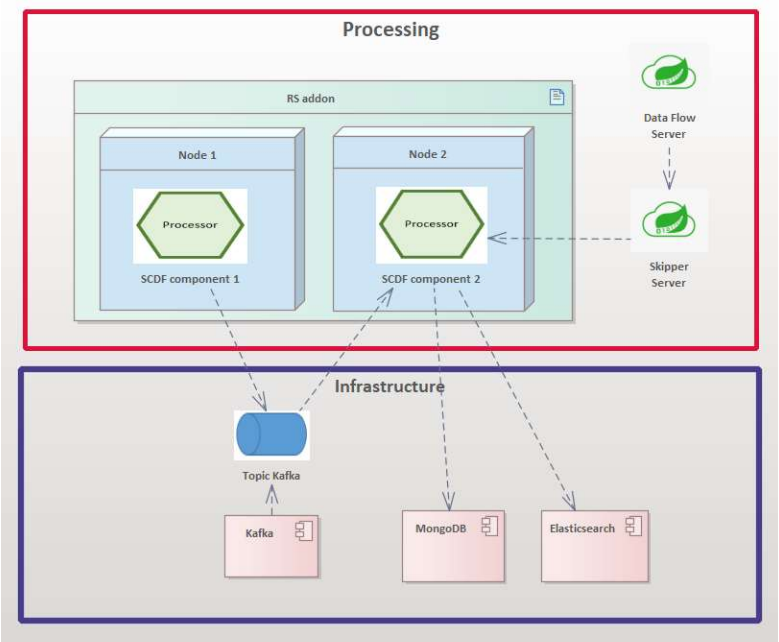
All container that are part of RS add-on shall be stored on the following registry root path.

Registry name	Registry path
Jfrog Artifactory	https://artifactory.coprs.esa-copernicus.eu

For each RS add-on, we propose such sub directories :

Mission	Sub path
S1	https://artifactory.coprs.esa-copernicus.eu/rs-docker-private/rs-addon/S1
S2	https://artifactory.coprs.esa-copernicus.eu/rs-docker-private/rs-addon/S2
S3	https://artifactory.coprs.esa-copernicus.eu/rs-docker-private/rs-addon/S3

3.4.3 Services



RS components can access storage service through:

Service name	Version	Namespace	URL	Secret
MongoDB	v5.0.3	infra	mongodb-0.mongodb-headless.database.svc.cluster.local, mongodb-1.mongodb-headless.database.svc.cluster.local, mongodb-2.mongodb-headless.database.svc.cluster.local	Yes
ElasticSearch	v7.15.2	infra	elasticsearch-processing-es-coordinating.database.svc.cluster.local	No

To access secret and Kafka messaging services, these properties shall be set :

- app.*.spring.kafka.bootstrap-servers: kafka-cluster-kafka-bootstrap.infra.svc.cluster.local:9092
- app deployer.*.kubernetes.imagePullSecrets: spring-cloud-dataflow-registry-dockersecret

3.4.4 Wiring

A RS add-on shall read input on topic : **catalog-event.**

A RS add-on shall write output on topics : **catalog-job** and/or **error-warning.**

4 RS ADD-ON STRUCTURE

4.1 COMPATIBILITY WITH SDP PUBLISHING SERVICE

The RS add-on format has been created from the Sentinel Data Processors Publishing service format described on the document [ICD1]. There are some discrepencies that will need some adjustement before publication to Sentinel Data Processors Publishing service. In particular the field [%DATE%] has been removed.

The RS add-on will be a single ZIP file named : RS_ADDON_[%PROCESSORID%]_[%REL%].ZIP

By opening the ZIP file, we will found the following items:

- RS_ADDON_[%PROCESSORID%]_[%REL%]_Manifest.json
- RS_ADDON_[%PROCESSORID%]_[%REL%]_Release_Note.pdf
- **/RS_ADDON_[%PROCESSORID%]_[%REL%]_Executables**
    - stream-application-list.properties
    - stream-definition.properties
    - stream-parameters.properties
    - **/additional_resources/** *(optional)*
        - KafkaTopic1.yml
        - KafkaTopic2.yml

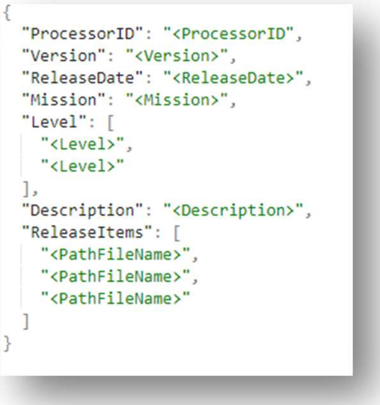
The items in bold are directories.

- [%PROCESSORID%] is the identifier of the delivered processor. We extend section 4.1 of [ICD1] with following values (non-exhausted list):
    - For RS add-on S1, we can have : **S1_L0ASP, S1_L0AIO, S1_L1, S1_L2**
    - For RS add-on S2, we can have : **S2_L0, S2_L1, S2_L2**
    - For RS add-on S3, we can have : **S3_L0, S3_MW1, S3_OL1, S3_OL1_RAC, S3_SL1, S3_SR1, S3_OL2, S3_SL2, S3_SL2_FRP, S3_SY2, S3_ SY2_VGS, S3_SY2_AOD, S3_PUG, S3_SM2**
- [%REL%] is the unique identifier of the current release (e.g. 3.2.1).

(source : §4.3 of [ICD1])

4.2 MANIFEST FORMAT

The delivery's Manifest contains general information about the RS add-on.
The JSON structure of the manifest is described below. All nodes are mandatory. All values are strings.



```
{
  "ProcessorID": "<ProcessorID",
  "Version": "<Version>",
  "ReleaseDate": "<ReleaseDate>",
  "Mission": "<Mission>",
  "Level": [
    "<Level>",
    "<Level>"
  ],
  "Description": "<Description>",
  "ReleaseItems": [
    "<PathFileName>",
    "<PathFileName>",
    "<PathFileName>"
  ]
}
```

- Mission can be : **S1, S2, S3**.
- Level can be : **L0, L1, L2, L3**
- Decription : free text providing a description of the RS add-on.
- PathFileName : fully qualified file name of an item being part of the RS add-on structure.

Example :

"ReleaseItems": [
"RS_ADDON_S1_L0ASP_1.3.8_*Manifest.json",*
"RS_ADDON_S1_L0ASP_1.3.8_*Release_Note.pdf",*
" RS_ADDON_S1_L0ASP_1.3.8_*Executables/stream-application-list.properties",*
" RS_ADDON_S1_L0ASP_1.3.8_*Executables/*stream-definition.properties",
" RS_ADDON_S1_L0ASP_1.3.8_*Executables/*stream-parameters.properties"*]*

(updated from source : §4.4 of [ICD1])

4.3 RELEASE NOTE

The release note is a single PDF file.It describes briefly the product and details about specific changes from previous release. It provided information about the RS add-on resource needed for execution.

At the minimum, the following elements shall be provided:

Ressource	Value
CPU	Min number of vCore
Memory	Min number of GigaBytes
Disk volume needed	YES / NO
Disk access	ReadWriteOnce / ReadOnlyMany / ReadWriteMany / ReadWriteOncePod
Disk storage capacity	Min number of GigaBytes
Affinity between POD/Node	To be define if needed.

The primary target audience is the IVV team and Production Service development team.

4.4 EXECUTABLES

This is the RS add-on payload.

Payload files are stored into a directory named : /RS_ADDON_[%PROCESSORID%]_[%REL%]_Executables

The payload is composed of 3 main files and an optional directory:

- ○ stream-application-list.properties
- ○ stream-definition.properties
- ○ stream-parameters.properties
- ○ **/additional_resources/** *(optional)*

Directory name sample:

- /RS_ADDON_S1_L0ASP_1.3.8_Executables
- /RS_ADDON_S2_L1_2.3.3c_Executables
- /RS_ADDON_S3_L0_2.3.3c_Executables

4.5 PAYLOAD FORMAT

The RS add-on payload is composed of:

1. SCDF stream containers list: PU and DPU container images.
2. SCDF stream definition : description of the workflow between the containers.
3. SCDF stream properties: properties for the POD build and the application dedicated
4. Services settings : directory with operators to set each service. This is an optional section.

On each properties files (item 1 to 3), comments can be added to the file with prefix #.

```
Any text
# my comment here
# another comment here
Any text
```

4.5.1 SCDF stream containers list

Filename : stream-application-list.properties

Content file format :
<type>.<name>=docker:<docker-image-path>/<imageName>:<version> (on line per application)
where ***<type>*** is equal to ***sink*** , ***source*** or ***processor***.

Sample RS add-on (with 2 PU and 1 DPU):

```
processor.ipf-preparation-trigger=docker:artifactory.coprs.esa-copernicus.eu/docker-name10:1.5.0
processor.ipf-preparation-worker=docker:artifactory.coprs.esa-copernicus.eu/docker-name20:3.6.1
processor.s1-l1-execution-worker=docker:artifactory.coprs.esa-copernicus.eu/docker-name30:0.1.0
```

4.5.2 SCDF stream definition

4.5.2.1 Application

A stream is defined by using a Unix-inspired Pipeline syntax. The syntax uses vertical bars, known as "pipes", to connect multiple commands. In Data Flow, the Unix command is replaced by a Spring Cloud Stream application and each pipe symbol represents connecting the input and output of applications over messaging middleware Apache Kafka. RS add-on can be composed by several lines.

Sample :

```
applicationName1 | applicationName2 | applicationName3
applicationName2 | applicationName4
applicationName3 | applicationName4 | applicationName5
```

4.5.2.2 Named destination

Instead of referencing a source or sink application, you can use a named destination. A named destination corresponds to a specific destination name in the middleware broker Kafka. When using the | symbol, applications are connected to each other with messaging middleware destination names created by the Data Flow server. In keeping with the Unix analogy, you can redirect standard input and output using the less-than (<) and greater-than (>) characters. To specify the name of the destination, prefix it with a colon (:).

Sample 1: `:myDestination > applicationName1`

Sample 2: `applicationName2 > :myDestination`

See : https://dataflow.spring.io/docs/feature-guides/streams/named-destinations/ for details.

4.5.2.3 Fan-In and Fan-Out

By using named destinations, you can support fan-in and fan-out use cases. Fan-in use cases are when multiple sources all send data to the same named destination.

The fan-out use case is when you determine the destination of a stream based on some information that is known only at runtime.

See : https://dataflow.spring.io/docs/feature-guides/streams/fanin-fanout/ for details.

4.5.2.4 Wiring

In cases with multiple input and output bindings, Data Flow cannot make any assumptions about the flow of data from one application to another. Therefore, you need to set the binding properties to "wire up" the application. The *Stream Application DSL* uses a "double pipe" (instead of the "pipe symbol") to indicate that Data Flow should not configure the binding properties of the application. Think of || as meaning "in parallel".

Sample : `applicationName1 || applicationName2 || applicationName3`

See : https://dataflow.spring.io/docs/feature-guides/streams/stream-application-dsl/ for details.

4.5.2.5 Stream Format

Filename : stream-definition.properties

Content format :

```
:myDestination1 > applicationName1 | applicationName2 || applicationName3  > :myDestination2
applicationName1 | applicationName3
```

Sample :

```
ipf-preparation-trigger | ipf-preparation-worker | s1-l1-execution-worker > :product-element
ipf-preparation-trigger | other-application
```

4.5.2.6 Stream name

When the “RS add-on” is deployed, it receives a name.

In a general case, the stream is defined by several lines. As a consequence, each line of the RS add-on is identifies as follow :

```
<RS-ADD-ON-NAME>-partX   where X>0. Start from 1 and incremented.
```

Sample :

To deploy the RS add-on “S1_L0ASP” with the name “**sentinel1-l0-asp**”.

```
sentinel1-L0-asp-part1
sentinel1-L0-asp-part2
sentinel1-L0-asp-part3
```

4.5.3 SCDF stream properties

All properties of the stream are grouped on a single file named : `stream-parameters.properties` .

Propertes fall into three groups:

- Deployer Properties: These properties control how the apps are deployed to the target platform and use a deployer prefix.
- Application binding properties (optional section) : binding properties (if use || on DSL definition, see §2.6.2.4).
- Application custom properties: These properties control or override how the application behaves and are set during stream creation.

4.5.3.1 Deployer properties

(source : https://docs.spring.io/spring-cloud-dataflow/docs/current/reference/htmlsingle/#configuration-kubernetes-deployer)

Kubernetes properties shall be draft as follow :

```
deployer.<application>.kubernetes.<property>=<value>
```

Mandatory properties to be set

Property	Description	Immutable	Example
namespace	Namespace to use	NO	myNs
livenessProbeDelay	Delay in seconds when the Kubernetes liveness check of the app container should start checking its health status.	YES	1234
livenessProbePeriod	Period in seconds for performing the Kubernetes liveness check of the app container.	YES	1234
livenessProbeTimeout	Timeout in seconds for the Kubernetes liveness check of the app container. If the health check takes longer than this value to return it is assumed as 'unavailable'.	YES	1234
livenessProbePath	Path that app container has to respond to for liveness check.	YES	/myProbe
livenessProbePort	Port that app container has to respond on for liveness check.	YES	1234
readinessProbeDelay	Delay in seconds when the readiness check of the app container should start checking if the module is fully up and running.	YES	1234
readinessProbePeriod	Period in seconds to perform the readiness check of the app container.	YES	1234

Property	Description	Immutable	Example
readinessProbeTimeout	Timeout in seconds that the app container has to respond to its health status during the readiness check.	YES	1234
readinessProbePath	Path that app container has to respond to for readiness check.	YES	/myProbe
readinessProbePort	Port that app container has to respond on for readiness check.	YES	1234
limits.memory	The memory limit, maximum needed value to allocate a pod, Default unit is mebibytes, 'M' and 'G" suffixes supported	YES	1234
limits.cpu	The CPU limit, maximum needed value to allocate a pod	YES	1234
requests.memory	The memory request, guaranteed needed value to allocate a pod.	YES	1234
requests.cpu	The CPU request, guaranteed needed value to allocate a pod.	YES	1234
maxTerminatedErrorRestarts	Maximum allowed restarts for app that fails due to an error or excessive resource use.	YES	1234

Optional properties to be set

Property	Description	Immutable	default
probeCredentialsSecret	The secret name containing the credentials to use when accessing secured probe endpoints.	NO	<none>
statefulSet.volumeClaimTemplate.storageClassName	Name of the storage class for a stateful set	YES	<none>
statefulSet.volumeClaimTemplate.storage	The storage amount. Default unit is mebibytes, 'M' and 'G" suffixes supported	YES	<none>
environmentVariables	List of environment variables to set for any deployed app container	YES	<none>
volumeMounts	volume mounts expressed in YAML format. e.g. `[{name: 'testhostpath', mountPath: '/test/hostPath'}, {name: 'testpvc', mountPath: '/test/pvc'}, {name: 'testnfs', mountPath: '/test/nfs'}]`	YES	<none>
volumes	The volumes that a Kubernetes instance supports specifed in YAML format. e.g. `[{name: testhostpath, hostPath: { path: '/test/override/hostPath' }},{name: 'testpvc', persistentVolumeClaim: { claimName: 'testClaim', readOnly: 'true' }}, {name: 'testnfs', nfs: { server: '10.0.0.1:111', path: '/test/nfs' }}]`	YES	<none>
secretRefs	The name of the secret(s) to load the entire data contents into individual environment variables. Multiple secrets may be comma separated.	NO	<none>
secretKeyRefs.envVarName	The environment variable name to hold the secret data	NO	<none>
secretKeyRefs.secretName	The secret name to access	NO	FALSE
secretKeyRefs.dataKey	The key name to obtain secret data from	NO	<none>
configMapRefs	The name of the ConfigMap(s) to load the entire data contents into individual environment variables. Multiple ConfigMaps be comma separated.	YES	<none>
configMapKeyRefs.envVarName	The environment variable name to hold the ConfigMap data	YES	<none>
configMapKeyRefs.configMapName	The ConfigMap name to access	YES	<none>

Property	Description	Immutable	default
configMapKeyRefs.dataKey	The key name to obtain ConfigMap data from	YES	<none>
maximumConcurrentTasks	The maximum concurrent tasks allowed for this platform instance.	YES	<none>
statefulSetInitContainerImageName	A custom image name to use for the StatefulSet Init Container	YES	<none>
initContainer	An Init Container expressed in YAML format to be applied to a pod. e.g. `{containerName: 'test', imageName: 'busybox:latest', commands: ['sh', '-c', 'echo hello']}`	YES	<none>

4.5.3.2 Application binding properties

This is an optional section in case the DSL definition include “||“ (see §2.6.2.4).

You can use default binding method if there is a single input/output:

```
app.<application>.spring.cloud.stream.bindings.input.destination=<kafkaTopicName>
app.<application>.spring.cloud.stream.bindings.output.destination=<kafkaTopicName>
```

Or map your own method to complete the wiring:

```
app.<application>.spring.cloud.stream.bindings.<method>.destination=<kafkaTopicName>
```

4.5.3.3 Application custom properties

Custom application properties shall be draft as follow :

```
app.<application>.<property>=<value>
```

The RS-add will provide only factory settings.

These settings can be updated for each SCDF chain.

Sample :
For a RS add-on Sentinel1 Level1 processing, properties can be :

- app.ipf-preparation-worker. IpfPreparationJob=”XXX”
- app.ipf-preparation-worker.inputTopicLowPriorityName=”XXX”
- app.ipf-preparation-worker.inputTopicMediumPriorityName=”XXX”
- app.ipf-preparation-worker.inputTopicHighPriorityName=”XXX”
- app. ipf-execution-worker.taskTable=”XXX”
- app. s1-l1-execution-worker.obs.path=”/sentinel-2/l1/”

4.5.4 Services settings

Setting for Kafka service are stored on specific directory.

/additional_resources/

For the time being, only Kafka setting can be updated thanks to operator.

4.5.4.1 Service Kafka settings

One YAML file per queue setting located here : **/additional_resources/KafkaTopic**

The filename is free. Extension must be YAML. One file per configuration.

```
apiVersion: kafka.strimzi.io/v1beta2
kind: KafkaTopic
metadata:
  name: <my_topic_name>
  labels:
    strimzi.io/cluster: kafka-cluster
    app.kubernetes.io/instance: <application_name>
    app.kubernetes.io/managed-by: additional_resources
spec:
  partitions: <2>
  replicas: <4>
  config:
    retention.bytes: <89478485>
    retention.ms: <"-1">
    segment.bytes: <22369621>
    cleanup.policy: <delete>
    min.insync.replicas: <2>
```

🟩 To be replaced. Mandatory.
🟧 Default value. The change is optional.

4.5.4.2 Service ElasticSearch settings

The application is allowed to create its index directly.

4.5.4.3 Service MongoDB settings

The application is allowed to create its index directly.

5 SAMPLE

RS_ADDON_S1_L0ASP_3.2.1.zip