Sentinel-2 Metadata Indexes

DHuS Open Source Framework

Role/Title	Name	Signature	Date
Authors	Calogera Tona- OSF Manager		02/5/2016
	Barbara Borgia- Dissemination Engineer		
	Raffaele Bua- DHuS Developer		
Reviewed	Guido Vingione		
Approved	Andrea Tesseri –Contract Manager		

Contract: 40000113153

Change register

Version/Rev.	Date	Reason for Change	Pages modified
1.0 Draft		First issue	
1.1	11/18/2015	Update considering the introduction of new indexes	7-18
1.2	02/05/2016	Update considering the introduction of new indexes	7-10, 13-17 , 21-26

Contents

List of Tables

Applicable Document

Id	Title	Reference	Issue
AD-1	OSF ICD	SPA-COPE-OSF-TN-005	1.1

Table 1 Applicable Documents

Reference Documents

Id	Title	Reference	Issue
RD-1	OData	http://www.odata.org/documentation/odata-version-2-0/	
RD-2	Apache Solr Reference Guide Covering Apache Solr 4.7	https://www.apache.org/dyn/closer.cgi/lucene/solr/ref-guide/	
RD-3	Sentinel-2 Product Specification	S2-PDGS-TAS-DI-PSD	n.13, 05/06/2015

Table 2 Reference Documents

1 Open Data Indexes

1.1 Introducing the Sentinel-2 products formatting

SENTINEL-2 data products, as described in the Sentinel-2 Product Specification (see RD-3), are distributed using a SENTINEL-specific variation of the Standard Archive Format for Europe (SAFE) format specification. The SAFE format has been designed to act as a common format for archiving and conveying data within ESA Earth Observation archiving facilities.

The SENTINEL-SAFE format wraps a folder containing image data in a binary data format and product metadata in XML. This flexibility allows the format to be scalable enough to represent all levels of SENTINEL products.

A SENTINEL product refers to a directory folder that contains a collection of information. It includes:

- a 'manifest.safe' file which holds the general product information in XML
- subfolders for measurement datasets containing image data in various binary formats
- a preview folder containing 'quicklooks' in PNG format, Google Earth overlays in KML format and HTML preview files
- an annotation folder containing the product metadata in XML as well as calibration data
- a support folder containing the XML schemes describing the product XML.

The data delivered is packaged as a file structure containing a manifest file in XML format listing general product metadata and subfolders for measurement data, annotations, previews and support files.

1.2 Inspection of Product Nodes

The DHuS recognises the Sentinel-2 products at ingestion time and makes products nodes accessible through the OData Protocol. The following odata query returns the list of nodes within the document root (first level). Note that both UUID and product name are needed.

```
/odata/v1/Products['UUID']/Nodes('PRODUCT_NAME.SAFE')/Nodes
```

```
https://scihub-test.esa.int/odata/v1/Products('b5eee74c-47c4-4774-
9bbb-
2ad96f68326a')/Nodes('S2A_OPER_PRD_MSIL1C_PDMC_20150720T103101_R137_V2
0150712T112303_20150712T112303.SAFE')/Nodes
```

An example of the xml returned by the previous query is shown below. The nodes are provided in the `<entry>` blocks.


```
<feed xmlns="http://www.w3.org/2005/Atom" xmlns:m="http://schemas.microsoft.com/ado/2007/08/dataservices/metadata" xmlns:d="http://schemas.microsoft.com/ado/2007/08/dataservices"
 xml:base="https://scihub-test.esa.int/dhus/odata/v1/Products('b5eee74c-47c4-4774-9bbb-2ad96f68326a')/Nodes('S2A_OPER_PRD_MSIL1C_PDMC_20150720T103101_R137_V20150712T112303_20150712T112303.SAFE')/">
  <id>
    https://scihub-test.esa.int/dhus/odata/v1/Products('b5eee74c-47c4-4774-9bbb-2ad96f68326a')/Nodes('S2A_OPER_PRD_MSIL1C_PDMC_20150720T103101_R137_V20150712T112303_20150712T112303.SAFE')/Nodes
  </id>
  <title type="text">Nodes</title>
  <updated>2015-07-21T10:49:26.369Z</updated>
  <author>
    <name/>
  </author>
  <link href="Nodes" rel="self" title="Nodes"/>
  <entry>
    <id>
      https://scihub-test.esa.int/dhus/odata/v1/Products('b5eee74c-47c4-4774-9bbb-
      2ad96f68326a')/Nodes('S2A_OPER_PRD_MSIL1C_PDMC_20150720T103101_R137_V20150712T112303_20150712T112303.SAFE')/Nodes('AUX_DATA')
    </id>
    <title type="text">AUX_DATA</title>
    <updated>2015-07-21T10:49:26.369Z</updated>
    <category term="DHuS.Node" scheme="http://schemas.microsoft.com/ado/2007/08/dataservices/scheme"/>
    <link href="Nodes('AUX_DATA')" rel="edit" title="Node"/>
    <link href="Nodes('AUX_DATA')/$value" rel="edit-media" type="application/octet-stream"/>
    <link href="Nodes('AUX_DATA')/Nodes" rel="http://schemas.microsoft.com/ado/2007/08/dataservices/related/Nodes" title="Nodes" type="application/atom+xml;type=feed"/>
    <link href="Nodes('AUX_DATA')/Attributes" rel="http://schemas.microsoft.com/ado/2007/08/dataservices/related/Attributes" title="Attributes" type="application/atom+xml;type=feed"/>
    <content type="application/octet-stream" src="Nodes('AUX_DATA')/$value"/>
    <m:properties>
      <d:Id>AUX_DATA</d:Id>
      <d:Name>AUX_DATA</d:Name>
      <d:ContentType>User Product Auxiliary Data Folder</d:ContentType>
      <d:ContentLength>0</d:ContentLength>
      <d:ChildrenNumber>0</d:ChildrenNumber>
      <d:Value m:null="true"/>
    </m:properties>
  </entry>
  <entry>...</entry>
  <entry>...</entry>
  <entry>...</entry>
  <entry>...</entry>
  <entry>...</entry>
  <entry>...</entry>
  <entry>...</entry>
</feed>
```


To inspect the nodes of lower levels,

```
/odata/v1/Products['UUID']/Nodes('PRODUCT_NAME.SAFE')/Nodes('nodename')/Nodes
```

For example, in order to get all children of the Node 'DATASTRIP' of a Product

```
https://scihub-test.esa.int/dhus/odata/v1/Products('b5eee74c-47c4-
4774-9bbb-
2ad96f68326a')/Nodes('S2A_OPER_PRD_MSIL1C_PDMC_20150720T103101_R137_V2
0150712T112303_20150712T112303.SAFE')/Nodes('DATASTRIP')/Nodes
```

The Content Type property reveals the type of the Node content. `<d:ContentType>`

If the content type of a node is '**`Item`**' and the number of its children is 0, the node is actually a leaf and it has a value. The user might get this value by appending the string `/Value/$value` to the leaf path.

The following example will return the value of the constant in an xml file:

Getting the metadata included in the leaf '/'Level-1C_User_Product/ 'General_Info'/…' of the Product (XML response)

```
/odata/v1/Products('b5eee74c-47c4-4774-9bbb-
2ad96f68326a')/Nodes('S2A_OPER_PRD_MSIL1C_PDMC_20150720T103101_R137_V2
0150712T112303_20150712T112303.SAFE')/Nodes('S2A_OPER_MTD_SAFL1C_PDMC_
20150720T103101_R137_V20150712T112303_20150712T112303.xml')/Nodes('Lev
el-
1C_User_Product')/Nodes('General_Info')/Nodes('Product_Info')/Nodes('P
RODUCT_TYPE')/Value/$value
```

If the node has content type different from 'item' (e.g `XML Document (eXtensible Markup Language)`, `SAFE Manifest`, etc..), the content download is allowed by appending the string `/$value` to the node path.

The following example will download the manifest.safe of a product:

Getting the manifest of a product

```
/odata/v1/Products('b5eee74c-47c4-4774-9bbb-
2ad96f68326a')/Nodes('S2A_OPER_PRD_MSIL1C_PDMC_20150720T103101_R137_V20150712
T112303_20150712T112303.SAFE')/Nodes('manifest.safe')/$value
```

2 Open Search Indexes -Sentinel-2 Metadata Indexes table

The following table contains the list of Sentinel-2 metadata indexed for Open Search.

N.B.: every open search is triggered by adding to the dhus path the string "/search?q=" followed by the example provided in the last column below.

2.1 L1B Granule indexes

The following table contains the list of Sentinel-2 Metadata.

Metadata Name	Index Name	Description	Example
acquisitionPeriod >startTime	beginPosition	Sensing start time	`beginposition:[2015-07-08T11:41:39.000Z TO NOW]` `beginposition:"2015-07-08T11:41:39.000Z"`
Sensing stop	endPosition	Sensing stop time	`endposition:[2015-07-04T10:24:27.000Z TO NOW`

Metadata Name	Index Name	Description	Example
			endposition:"2015-07-04T10:24:27.000Z"
PROCESSING_BASELINE	processingbaseline	The Processing Baseline completely defines the processing environment baseline used at the time of the product generation in terms of: · Processors version number; · Static Auxiliary Data (e.g. DEM, GRI) each one with a version number; · Dynamic Auxiliary Data (e.g. ECMWF data or POD data), each one with its associated version number, · Processing Configuration files versions. Processing Baseline = xx.yy where x,y = {0;9} An increase of the Processing Baseline code is generated by	processingBaseline:01.00

Metadata Name	Index Name	Description	Example
		a change of the elements listed above. A major change is traced by the "xx" digits, a minor change is traced by the "yy" digits.	
Satellite name	platformName	Satellite name (e.g. S2A or S2B)	`platformname:Sentinel-2`
instrument>familyName	instrumentName	The instrument name used for acquiring the product data	`instrumentName: Multi-Spectral Instrument`
Instrument abbreviation	instrumentShortName	Instrument name abbreviation	`instrumentshortname:MSI`
general_info>DATATAKE_TYPE>	sensorOperationalMode	The mode of the instrument: -Nominal_Observation -Dark_Signal_Calibration -Extended_Observation -Absolute_Radiometry_Calibration -Vicarious_Calibration	`sensoroperationalmode:INS-NOBS`

Metadata Name	Index Name	Description	Example
		-Raw_Measurement -Test_Mode	
Cloud_Coverage_ Assessment	cloudcoverpercentage	Percentage of cloud coverage of the product for each area covered bya reference band.	`cloudcoverpercentage:0.0`

Metadata Name	Index Name	Description	Example
Mission datatake id	S2datatakeid	Datatake Id. The Sentinel-2 User Products will always refer to a given Datatake. Datatake definition refers to a continuous acquisition of an image from one Sentinel-2 satellite in a given MSI imaging mode. The maximum length of an imaging Datatake is 15000 km (continuous observation from Northern Russia to Southern Africa) and this is the longest possible product that a user can ask for.	`s2datatakeid:GS2A_20150704T101006_000162_N77.00`
Orbit number (start)	orbitNumber	Absolute orbit number.	`orbitNumber:162` `orbitnumber:[2000 TO 2700]`
PRODUCT_TYPE	productType	Product type identifier: - S2MSI0	`productType:S2MSI1C`

Metadata Name	Index Name	Description	Example
		- S2MSI1A - S2MSI1B - S2MSI1C - S2MSI2Ap	
Product_Footprint>Global_Footprint><EXT_POS_LIST>	gmlfootprint	Product footprint using Geography markup language coordinates (http://en.wikipedia.org/wiki/Geography_Markup_Language).	`gmlfootprint=POLYGON((-4.53 29.85, 26.75 29.85, 26.75 46.80,-4.53 46.80,-4.53 29.85))`
footprint>coordinates	footprint	Product footprint using Java topology suite coordinates (http://en.wikipedia.org/wiki/JTS_Topology_Suite).	`footprint:"Intersects(POLYGON((-13.115927734375%2027.752507427949,37.509072265625%2027.752507427949,37.509072265625%2061.475999093721,-13.115927734375%2061.4759990`

Metadata Name	Index Name	Description	Example
			93721,-platformidentifier 13.115927734375%2027.752507427949)))"
-	collection	Collection to whuch the products belong.Note that the collection is not set in the native product, it is defined when the product is ingested in th DHuS	collection:name_collection

2.2 L1B User Product indexes

Metadata Name	Index Name	Description	Example
acquisitionPeriod>startTime	beginPosition	Sensing start time	beginposition:[2015-07-08T11:41:39.000Z TO NOW] beginposition:"2015-07-08T11:41:39.000Z"
Sensing stop	endPosition	Sensing stop time	endposition:[2015-07-04T10:24:27.000Z TO NOW

Metadata Name	Index Name	Description	Example
			endposition:"2015-07-04T10:24:27.000Z"
PROCESSING_BASELINE	processingbaseline	The Processing Baseline completely defines the processing environment baseline used at the time of the product generation in terms of: · Processors version number; · Static Auxiliary Data (e.g. DEM, GRI) each one with a version number; · Dynamic Auxiliary Data (e.g. ECMWF data or POD data), each one with its associated version number, · Processing Configuration files versions. Processing Baseline = xx.yy where x,y = {0;9} An increase of the Processing Baseline code is	processingBaseline:01.00

Metadata Name	Index Name	Description	Example
		generated by a change of the elements listed above. A major change is traced by the “xx” digits, a minor change is traced by the “yy” digits.	
Satellite name	platformName	Satellite name (e.g. S2A or S2B)	`platformname:Sentinel-2`
instrument>familyName	instrumentName	The instrument name used for acquiring the product data	`instrumentName: Multi-Spectral Instrument`
Instrument abbreviation	instrumentShortName	Instrument name abbreviation	`instrumentshortname:MSI`
general_info>DATATAKE_TYPE>	sensorOperationalMode	The mode of the instrument: -Nominal_Observation -Dark_Signal_Calibration -Extended_Observation -Absolute_Radiometry_Calibration -Vicarious_Calibration	`sensoroperationalmode:INS-NOBS`

Metadata Name	Index Name	Description	Example
		-Raw_Measurement -Test_Mode	
Cloud_Coverage_ Assessment	cloudcoverpercentage	Percentage of cloud coverage of the product for each area covered bya reference band.	`cloudcoverpercentage:0.0`

Metadata Name	Index Name	Description	Example
Mission datatake id	S2datatakeid	Datatake Id. The Sentinel-2 User Products will always refer to a given Datatake. Datatake definition refers to a continuous acquisition of an image from one Sentinel-2 satellite in a given MSI imaging mode. The maximum length of an imaging Datatake is 15000 km (continuous observation from Northern Russia to Southern Africa) and this is the longest possible product that a user can ask for.	`s2datatakeid:GS2A_20150704T101006_000162_N77.00`
Orbit number (start)	orbitNumber	Absolute orbit number.	`orbitNumber:162` `orbitnumber:[2000 TO 2700]`
PRODUCT_TYPE	productType	Product type identifier: - S2MSI0	`productType:S2MSI1C`

Metadata Name	Index Name	Description	Example
		- S2MSI1A - S2MSI1B - S2MSI1C - S2MSI2Ap	
Product_Footprint>Global_Footprint><EXT_POS_LIST>	gmlfootprint	Product footprint using Geography markup language coordinates (http://en.wikipedia.org/wiki/Geography_Markup_Language).	gmlfootprint=POLYGON((-4.53 29.85, 26.75 29.85, 26.75 46.80,-4.53 46.80,-4.53 29.85))
footprint>coordinates	footprint	Product footprint using Java topology suite coordinates (http://en.wikipedia.org/wiki/JTS_Topology_Suite).	footprint:"Intersects(POLYGON((-13.115927734375%2027.752507427949,37.509072265625%2027.752507427949,37.509072265625%2061.475999093721,-13.115927734375%2061.4759990

Metadata Name	Index Name	Description	Example
			93721,-platformidentifier 13.115927734375%2027.752507427949)))"
nssdcIdentifier	platformidentifier	Platform NSSDC identifier	platformidentifier:2015-000A
SENSING_ORBIT_DIRECTION	orbitdirection	orbit direction	orbitdirection:ASCENDING
relativeOrbitNumber	relativeOrbitNumber	Indicates if the orbit number refers to the oldest or the most recent data unit.	relativeorbitnumber:[10 TO 30]
platform>number	platformSerialIdentifier	Identifier of the mission satellite	platformSerialIdentifier: 2A
-	collection	Collection to whuch the products belong.Note that the collection is not set in the native product, it is defined when the product is ingested in th DHuS	collection:name_collection

2.3 L1C User Product indexes

Metadata Name	Index Name	Description	Example
Sensing start	beginPosition	Sensing start time	`beginposition:[2015-07-08T11:41:39.000Z TO NOW]` `beginposition:"2015-07-08T11:41:39.000Z"`
Sensing stop	endPosition	Sensing stop time	`endposition:[2015-07-04T10:24:27.000Z TO NOW` `endposition:"2015-07-04T10:24:27.000Z"`
Processing baseline	processingBaseline	The Processing Baseline completely defines the processing environment baseline used at the time of the product generation in terms of: · Processors version number;	`processingBaseline:01.00`

		· Static Auxiliary Data (e.g. DEM, GRI) each one with a version number; · Dynamic Auxiliary Data (e.g. ECMWF data or POD data), each one with its associated version number, · Processing Configuration files versions. Processing Baseline = xx.yy where x,y = {0;9} An increase of the Processing Baseline code is generated by a change of the elements listed above. A major change is traced by the "xx" digits, a minor change is traced by the "yy" digits.	
Satellite name	platformName	Satellite name (e.g. S2A or S2B)	`platformName:S2A`
Instrument name	instrumentName	The instrument name used for acquiring the product data	`instrumentName:"Multi-Spectral Instrument"`
Instrument	instrumentShortName	Instrument abbreviation	`instrumentshortname:"MSI"`

abbreviation			
Instrument mode	sensorOperationalMode	The mode of the instrument: -Nominal_Observation -Dark_Signal_Calibration -Extended_Observation -Absolute_Radiometry_Calibration -Vicarious_Calibration -Raw_Measurement -Test_Mode	`sensorOperationalMode:"INS-NOBS"`
Cloud coverage percentage	cloudcoverpercentage	Percentage of cloud coverage of the product for each area covered by a reference band	`cloudcoverpercentage:0.0`
Mission datatake id	S2datatakeid	Datatake Id. The Sentinel-2 User Products will always refer to a given Datatake.	`s2datatakeid:GS2A_20150704T101006_000162_N77.00`

		Datatake definition refers to a continuous acquisition of an image from one Sentinel-2 satellite in a given MSI imaging mode. The maximum length of an imaging Datatake is 15000 km (continuous observation from Northern Russia to Southern Africa) and this is the longest possible product that a user can ask for.	
Orbit number (start)	orbitNumber	Absolute orbit number	`orbitNumber:162`
Relative orbit (start)	relativeOrbitNumber	Relative orbit number	`relativeorbitnumber:19`
Product type	productType	Product type identifier: - S2MSI0 - S2MSI1A	

		- S2MSI1B - S2MSI1C - S2MSI2Ap	`productType:S2MSI1C`
Footprint	GMLfootprint	Product footprint using Geography markup language coordinates (http://en.wikipedia.org/wiki/Geography_Markup_Language)	
JTS footprint	footprint	Product footprint using Java topology suite coordinates (http://en.wikipedia.org/wiki/JTS_Topology_Suite)	`footprint:"Intersects(POLYGON((-13.115927734375%2027.752507427949,37.509072265625%2027.752507427949,37.509072265625%2061.475999093721,-13.115927734375%2061.475999093721,-13.115927734375%2027.752507427949)))"`

nssdcIdentifier	platformidentifier	Platform NSSDC identifier	`platformidentifier:2015-000A`
SENSING_ORBIT_DIRECTION	orbitdirection	orbit direction	`orbitdirection:ASCENDING`
platform>number	platformSerialIdentifier	Identifier of the mission satellite	`platformSerialIdentifier: 2A`
-	collection	Collection to whuch the products belong.Note that the collection is not set in the native product, it is defined when the product is ingested in th DHuS	`collection:name_collection`