istituto per il rilevamento elettromagnetico dell'ambiente

GeoNode

11 - 13 June, 2019

GET-IT: Integrating sensor information and semantics-aware metadata in GeoNode

Cristiano Fugazza, Paolo Tagliolato,
Simone Lanucara, Alessandro Oggioni

{fugazza.c, tagliolato.p, lanucara.s, oggioni.a}[at]irea[dot]cnr[dot]it

Context

TRAINING FOR EDUCATION, LEARNING AND LEADERSHIP
TOWARDS A NEW METROPOLITAN DISCIPLINE

ICPSM

Description (1)

Software tools:

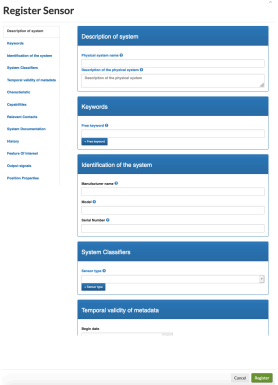
- GeoNode (now v2.7, soon v2.10)
  - Improved **geoext** library for SOS resources
  - JavaScript application (***upload_observations***) for insertion of SOS observations
- EDI (v1.2)
  - Integrated via module **geosk.mdtools**
- 52°North SOS (v4.4.2)
  - Integrated via module **geosk.osk**

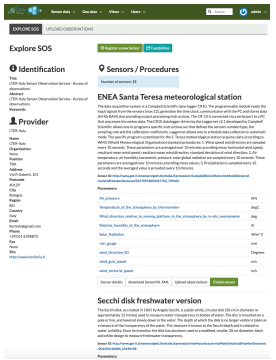
Relevant formats:

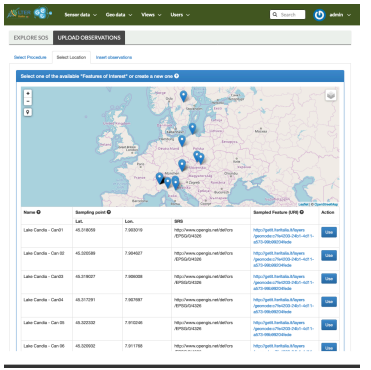
- XML
  - Template language for definition of the metadata profile (and of the editing interface)
  - Storage format for metadata based on the template language
  - XPath simple paths specifying the XML nodes to be created
  - XSLT for:
    - modifying the target metadata output prior to insertion in the catalog
    - composing transactional SOS operations operated by JavaScript
    - HTML representation of XML SensorML
- RDF
  - Triple store
  - SPARQL

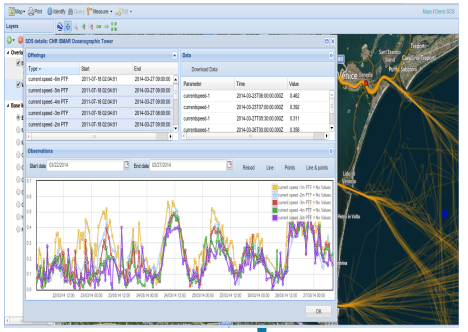
Sensor management WF

	Insert Sensor	Sensor list	Sensor details	Insert observation	Maps with observations
Integration module	**geosk.osk**	**geosk.osk**	**geosk.osk**	***upload_observations***	**geoext** extended
Programming language	py	py	py XSLT	js	js
Software involved	EDI SOS	SOS	SOS GeoServer	SOS GeoServer	SOS GeoServer







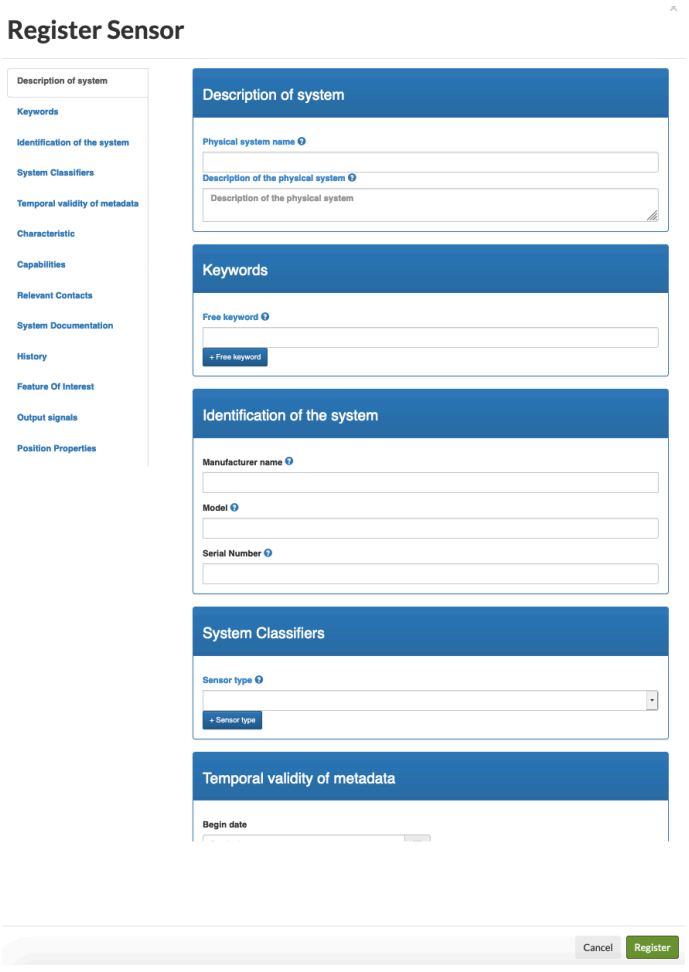


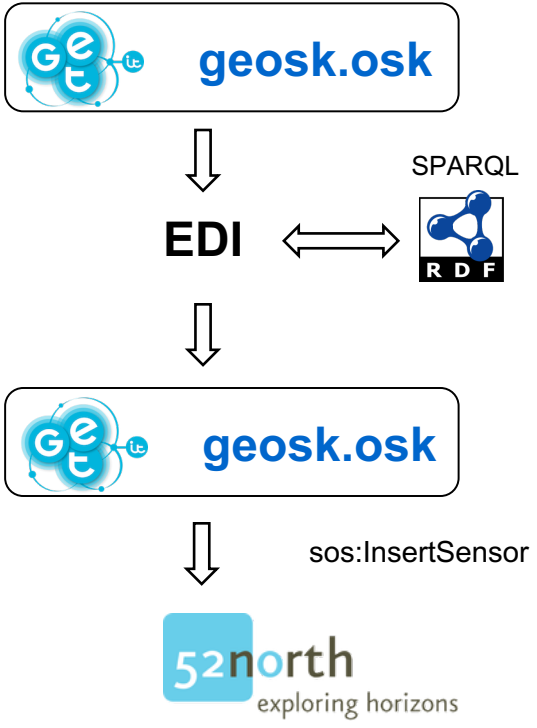
istituto per il rilevamento elettromagnetico dell'ambiente

GeoNode

1. Insert sensor

	Insert Sensor
Integration module	geosk.osk
Programming language	py
Involved software	EDI SOS



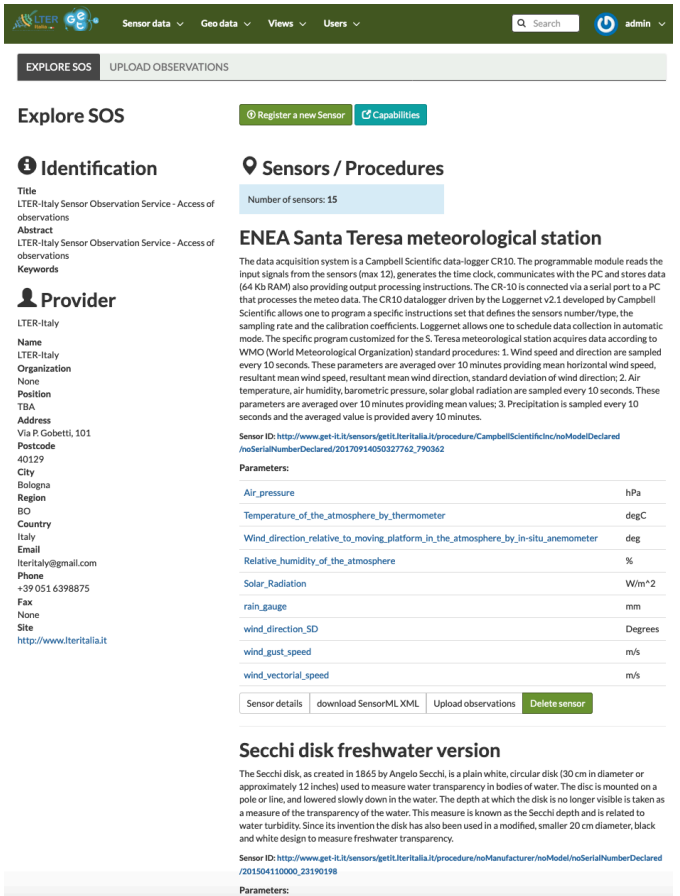


istituto per il rilevamento elettromagnetico dell'ambiente

GeoNode

2. Sensor list

	Sensor list
Integration module	geosk.osk
Programming language	py
Involved software	SOS



geosk.osk

geosk.osk

sos:GetCapabilities

52north exploring horizons

SPARQL

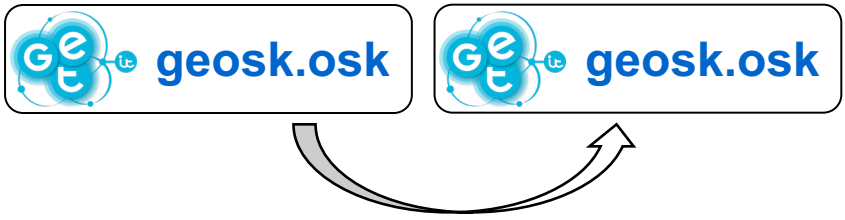
RDF

istituto per il rilevamento elettromagnetico dell'ambiente

GeoNode

3. Sensor details

	Sensors list
Integration module	geosk.osk
Programming language	py XSLT
Involved software	SOS GeoServer



sos:describeSensor

WFS GetFeature

SPARQL

52north exploring horizons

GeoServer

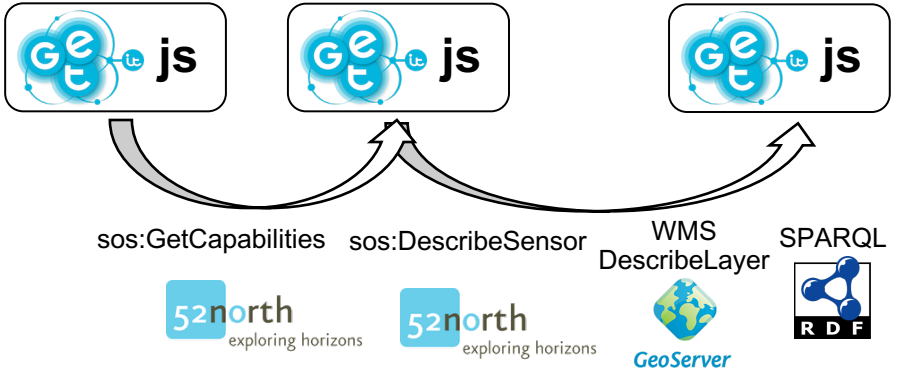
RDF

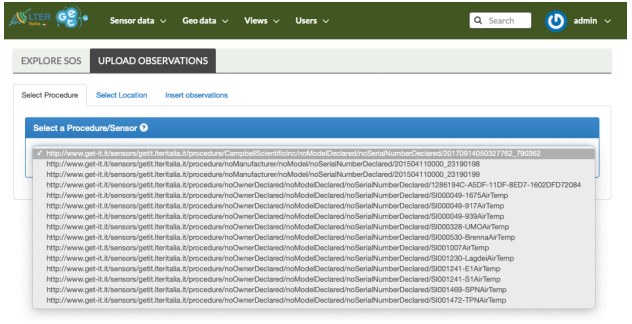
istituto per il rilevamento elettromagnetico dell'ambiente

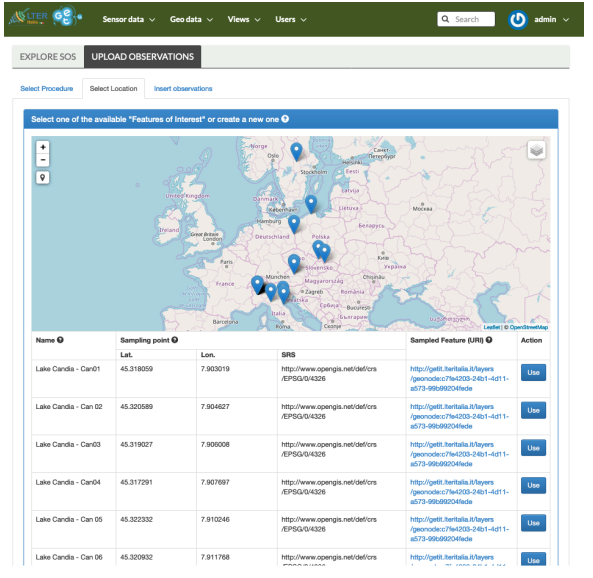
GeoNode

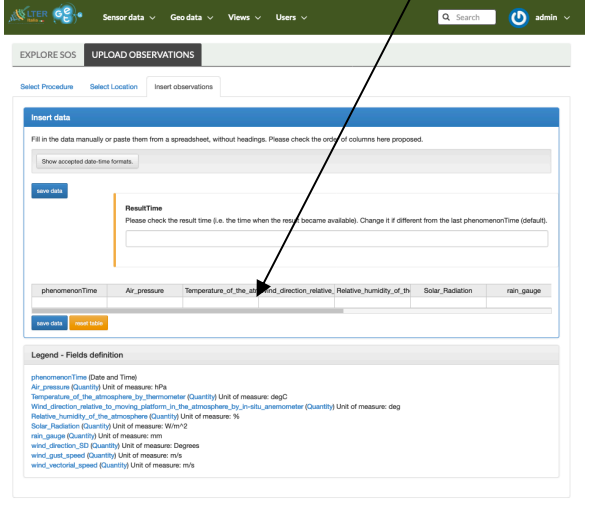
4. Insert observation

	Insert observation
Integration module	upload_ observations
Programming language	js
Involved software	SOS GeoServer





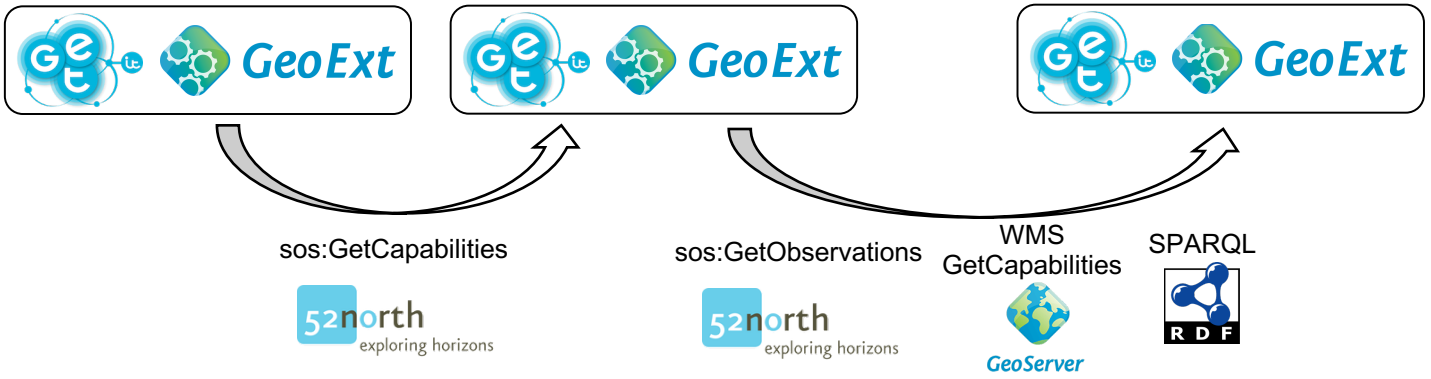


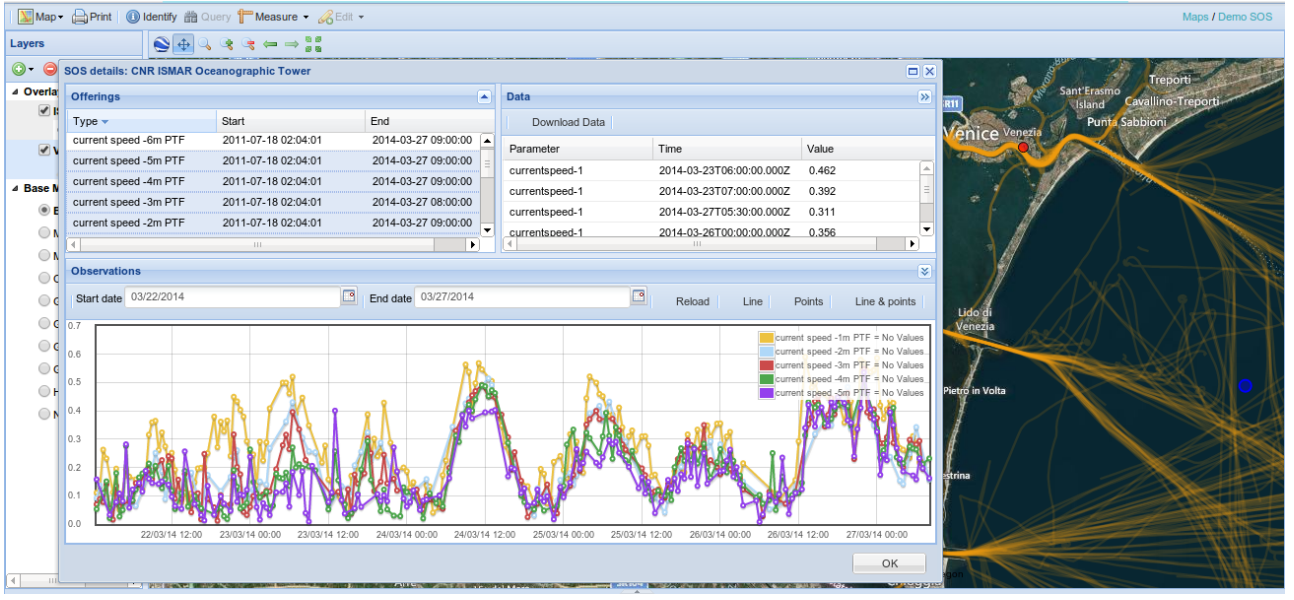


istituto per il rilevamento elettromagnetico dell'ambiente

5. Maps with observation

	Maps with observation
Integration module	**geoext** extended
Programming language	js
Involved software	SOS GeoServer





istituto per il rilevamento elettromagnetico dell'ambiente

GeoNode

Formats: XML

```
<element xml:id="resp">
   <label xml:lang="en">Responsible party</label>
   <label xml:lang="it">Responsabile</label>
   <produces>
      <item datasource="person"...>
         <label xml:lang="en">Email</label>
         <label xml:lang="it">Email</label>
         <hasPath>/.../gmd:electronicMailAddress/...</hasPath>
      </item>
      <item ...>
         <label xml:lang="en">Institute</label>
         <label xml:lang="it">Istituto</label>
         <hasPath>/.../gmd:organisationName/...</hasPath>
      </item>
      <item ...>
         <label xml:lang="en">Role</label>
         <label xml:lang="it">Ruolo</label>
         <hasPath>/.../gmd:CI_RoleCode/...</hasPath>
      </item>
   </produces>
</element>
```

Formats: RDF

```
<datasources>
 <sparql xml:id="person">
  <query>
  <![CDATA[
   SELECT ?contact ?label
   WHERE {
    ?contact rdf:type foaf:Person .
    ?contact vcard:email ?label .
    FILTER(
     REGEX( STR(?label) ,"$search_param","i") )
   }
   ORDER BY ASC(?label)
  ]]>
  </query>
  <url>http://url.to.endpoint/</url>
 </sparql>
</datasources>
```

GET-IT specific endpoints

URL / Endpoint	Type	Title	Software
/layers/[layername]#ediclient_container	GUI	Semantic metadata editor	GET-IT (EDI Client)
/maps/	GUI	Explore view/map	GeoNode/GET-IT (SOS Client)
/observations/	GUI	SOS 52°North Home page	SOS 52°North
/observations/sos/	Service	SOS	SOS 52°North
/sensors/	GUI	Explore sensor	GET-IT (SOS Manager)
/sensors/#ediclient_container	GUI	Semantic metadata editor	GET-IT (EDI Client)
/sensors/sensor/ds/?[sensor_id]	GUI	Sensor details	GET-IT (SOS Manager)
/sensors/upload?[sensor_id]	GUI	Upload observations	GET-IT (SOS Manager)
/sensors/deletesensor?[sensor_id]	GUI	Delete sensor	GET-IT (SOS Manager)
/whoami	API	Who am I (GET-IT info)	GET-IT
/mdtools/api/listediml	API	List of EDIML resources	GET-IT
/layers/[layername]/ediml	API	EDIML resource metadata	GET-IT
/layers/[layername]/rndt	API	RNDT resource metadata	GET-IT
/mdtools/api/importediml	GUI/API	Import EDIML metadata	GET-IT
/mdtools/api/importrndt	GUI/API	Import RNDT metadata	GET-IT
/static/EDI-NG_client/	GUI	EDI Client	GET-IT (EDI Client)

Outcome

- Integration of geographic and sensor data
  - Both for authentication and visualization
- Semantic characterization of metadata
  - Allow for advanced assisted editing of metadata
  - Integrated with the GeoNode structures (e.g., for facets)
  - Enables query expansion in discovery tasks

Outlook

- Internal triple store (e.g., Virtuoso, Jena) for self-contained customization of the metadata editing facilities
- SDI-wide federated authentication / authorization among the nodes (and selected third-party applications)
  - An internal triple store would allow for RDF-based representation of users and re-use of this information for editing metadata
- Semantic lift of existing metadata for re-EDIting
- Community support

Question time

https://github.com/SP7-Ritmare/starterkit

http://get-it.it/

https://getit.readthedocs.io/en/latest/index.html