

Sofia 

CONCETTI

SOFIA2

maggio 2014

versione 4



indra

1 INDICE

1	INDICE	2
2	INTRODUZIONE	3
2.1	OBIETTIVI ED AMBITO DEL DOCUMENTO	3
3	COS'È SOFIA2	4
4	CONCETTI DELLA PIATTAFORMA SOFIA2	6
4.1	SMART SPACE.....	6
4.2	SIB (SEMANTIC INFORMATION BROKER).....	6
4.3	KP (KNOWLEDGE PROCESSOR)	7
4.4	SSAP (SMART SPACE ACCESS PROTOCOL).....	7
4.5	ONTOLOGIE	8



2 Introduzione

2.1 Obiettivi ed ambito del documento

Questo documento descrive i concetti di base della piattaforma Sofia2.



3 Cos'è Sofia2

Sofia2 nasce da un progetto di ricerca europea che si chiama SOFIA.

SOFIA è l'acronimo di **SMART OBJECTS FOR INTELLIGENT APPLICATIONS**: è una piattaforma che nasce come progetto di ricerca Artemis di tre anni, terminato nel marzo 2012, cui hanno preso parte 19 partner di quattro paesi europei, come Nokia, Philips, Fiat, Acciona e Indra.

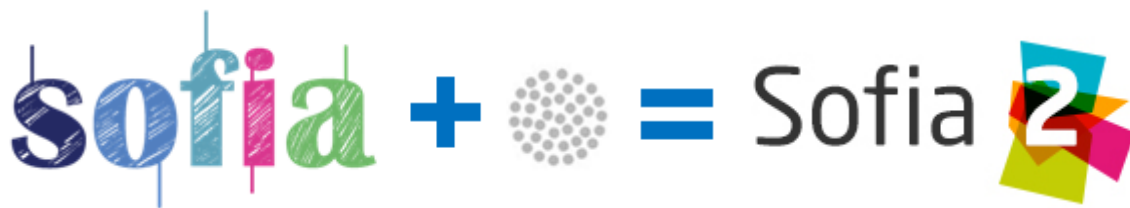
SOFIA è un'architettura "middleware", ossia permette l'interoperabilità fra sistemi e dispositivi diversi. Mette a disposizione di applicazioni intelligenti informazioni sul mondo reale (**Internet of Things**).

SOFIA è:

- Open-source
- Multiplatforma: disponibile per Windows, Android, Linux, iOS,...
- Multilinguaggio: portabile su Java, Javascript, C++, Arduino
- Indipendente dal protocollo di comunicazione: implementazioni TCP, MQTT, HTTP (REST e WebServices), Ajax Push,...

Il suo scopo è ottenere l'interoperabilità fra applicazioni che condividono **concetti semantici**.

Al termine del progetto Artemis, Indra ha continuato a sviluppare il progetto SOFIA trasformandolo in una piattaforma di carattere commerciale: **Sofia2**



Sofia2 punta su queste aree:

- **Adattamento alle esigenze di business:** funzionamento in alta disponibilità con datacenter distribuiti
- L'operatività sulla piattaforma è stata semplificata, specialmente per:
 - Sviluppo delle ontologie (rendendole più leggere)
 - Linguaggi di ricerca.
 - Protocollo SSAP: implementazione JSON oltre allo standard XML.

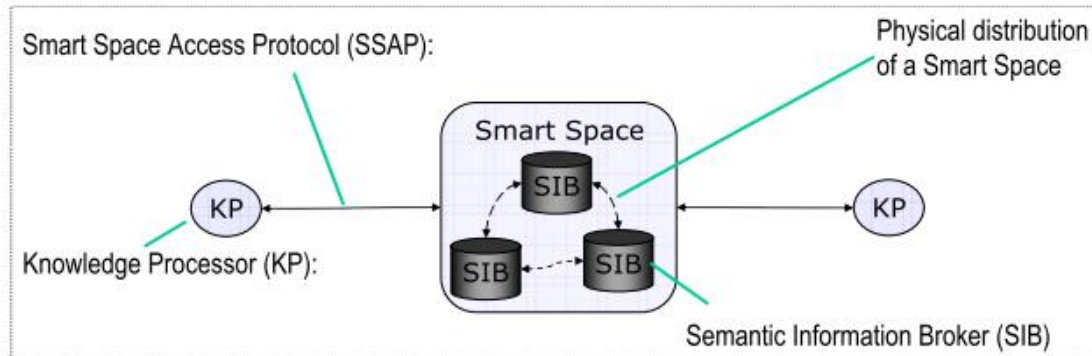
- Interfaccia Big Data (Hadoop) per l'immagazzinamento di grandi volumi di dati e datawarehousing.
- Capacità di integrazione con i back-end mediante protocolli standard come i web service
- Concetto di "plug-in" per ampliare il SIB (Semantic Information Broker)
- Memorizzazione e consultazione integrata di GIS (Geographic Information Services)
- Aggiunta come plug-in di meccanismi di sicurezza.
- Interfacce REST per connettersi semplicemente da smartphones, dispositivi, applicazioni RIA etc.



4 Concetti della Piattaforma Sofia2

La Piattaforma **Sofia2** si basa su questi quattro concetti:

- **Smart Space**
- **SIB**
- **KP**
- **SSAP**



4.1 Smart Space

- E' l'ambito virtuale in cui più applicazioni interagiscono per offrire una funzionalità completa.
- Il nucleo di uno Smart Space è il **SIB**.
- In uno Smart Space in genere esiste un solo SIB (può essere un cluster), anche se in alcuni casi esistono federazioni di SIB.
- Gli Smart Spaces possono comunicare fra loro stabilendo relazioni di trust.

4.2 SIB (Semantic Information Broker)

- E' il nucleo della Piattaforma.
- Riceve, calcola e immagazzina tutte le informazioni delle applicazioni connesse alla piattaforma SOFIA2, funzionando come Bus di Interoperabilità.
- In esso si rispecchiano tutti i concetti del dominio (ossia l'ontologia) e il loro stato attuale (ossia le istanze delle ontologie).
- SOFIA2 usa JSON per lo scambio di informazioni (SSAP) e per la definizione delle ontologie.

```
{
  "body": {
    "query": "{SensorHumedad.medida:{>18}}",
    "direction": "REQUEST",
  }
}
```

```
"ontology":"SensorHumedad",  
"messageType":"QUERY",  
"messageId":121,  
"sessionKey":"88bf5ee7-84d4-4956-98a3-ff290222fd64"  
}
```

- Esistono implementazioni in vari linguaggi e su varie piattaforme. Indra fornisce un SIB JEE che gira su qualunque Web Server (Tomcat, JBoss,...).
- Il Gateway supporta i protocolli di comunicazione TCP/IP, HTTP, REST, Bluetooth e ZigBee.
- **Offre connettori** per comunicazioni fra clients:
 - REST: per clients Javascript, smartphones, etc.
 - MQTT per comunicazioni bidirezionali e dispositivi limitati
 - Web Services/JMS per applicazioni industriali
 - Altri protocolli come Bluetooth, ZigBee etc.
- SIB è estensibile mediante plugin.

4.3 KP (Knowledge Processor)

- Modella ognuna delle applicazioni che interagiscono nello Smart Space mediante il SIB.
- Ogni applicazione lavora con le istanze dei concetti rilevanti del dominio (ontologie) per cui sono progettate.
- Implementazioni in vari linguaggi come Java, Javascript, Arduino, etc.
- Ci sono tre tipi di KP:
 - **Producer**: KP che inserisce informazione nel SIB.
 - **Consumer**: KP che recupera informazione dal SIB.
 - **Prosumer**: KP che inserisce e recupera informazioni dal SIB.
- In SOFIA2 l'invio dei messaggi SSAP è implementato in JSON, perché più leggero e quindi adeguato ai dispositivi embedded.

4.4 SSAP (Smart Space Access Protocol)

- È il linguaggio di messaggeria standard per le comunicazioni fra il SIB e i KPs.
- Il linguaggio è indipendente dalla rete sottostante (GPRS, 3G, WIFI, BlueTooth, HFC, ZigBee)
- Ne esistono due implementazioni:
 - **SSAP-XML**: formato XML (migliore ampiezza di banda)

- **SSAP-JSON**: messaggi adattati a questo protocollo, concepito per le comunicazioni con i dispositivi mobili, i navigatori, etc.
- I messaggi sono di tre tipi:
 - **REQUEST**: ossia una richiesta inviata dal KP al SIB.
 - **RESPONSE**: ossia la risposta del SIB verso il KP corrispondente all'arrivo di un messaggio REQUEST.
 - **INDICATION**: ossia una notifica inviata dal SIB al KP prima del verificarsi di un evento al quale il KP è iscritto.
- Le operazioni mediante le quali il SIB e i KPs interagiscono sono:
 - **JOIN**: connessione di un KP a un SIB (implica autenticazione, autorizzazione e la creazione di una sessione nello Smart Space)
 - **LEAVE**: disconnessione di un KP dal SIB
 - **INSERT/UPDATE/DELETE**: permette ai KPs l' /inserzione/modifica/cancellazione di informazione nel SIB
 - **QUERY**: permette ai KPs di recuperare informazione dal SIB dalle basi dati relative al Tempo Reale o dello Storico.
 - **SUBSCRIBE**: permette ai KPs di iscriversi all'esecuzione di una consultazione riguardo lo stato del SIB ogni tot secondi o al verificarsi di un evento nel SIB
 - **INDICATION**: segnalazione di riscontro in seguito ad un'operazione di "subscribe", è inviata dal SIB ai KPs.
 - **UNSUBSCRIBE**: cancella un'operazione di "subscribe".
 - **CONFIG**: permette al KP di richiedere i parametri di configurazione relativi alla sua istanza.
 - Notifiche di cambiamento di stato dal SIB ai KP che si sono "iscritti" (operazione SUBSCRIBE) ad una data notifica di cambio di stato.

4.5 Ontologie

Le **ontologie** sono descrizioni semantiche di un insieme di classi, esse rappresentano le entità del sistema.

In SOFIA2 le ontologie sono rappresentate nel formato JSON-Schema, per esempio un'ontologia che usa un KP per rappresentare un sensore di temperatura si rappresenta così:

```
"SensorTemperatura": {  
  "coordenadaGps": {  
    "altitud": 0,  
    "latitud": 40.512274,
```



```

    "longitud": -3.675679
  },
  "identificador": "S_Temperatura_00001",
  "medida": 19,
  "timestamp": 1373887443001,
  "unidad": "C"
}
},

```

Le ontologie JSON vengono scaricate nella piattaforma ed esiste un JSON Schema che serve a validare le informazioni semantiche che arrivano dalle KP secondo le regole dell'ontologia:

Per tornare all'esempio precedente dell'ontologia SensorTemperatura, il JSON Schema che la valida è:

```

{
  "$schema": "http://json-schema.org/draft-03/schema#",
  "title": "SensorTemperatura Schema",
  "type": "object",
  "properties": {
    "_id": {
      "type": "object",
      "$ref": "#/identificador"
    },
    "SensorTemperatura": {
      "type": "string",
      "$ref": "#/datos"
    }
  },
  "identificador": {
    "title": "id",
    "description": "Id insertado del SensorTemperatura",
    "type": "object",
    "properties": {
      "$oid": {
        "type": "string",
        "required": false
      }
    }
  },
  "datos": {
    "title": "datos",
    "description": "Info SensorTemperatura",
    "type": "object",
    "properties": {
      "identificador": {
        "type": "string",
        "required": true
      },
      "timestamp": {
        "type": "integer",
        "minimum": 0,
        "required": true
      },
      "medida": {
        "type": "number",
        "required": true
      }
    }
  }
}

```

```

    },
    "unidad": {
      "type": "string",
      "required": true
    },
    "coordenadaGps": {
      "required": true,
      "$ref": "#/gps"
    }
  }
},
"gps": {
  "title": "gps",
  "description": "Gps SensorTemperatura",
  "type": "object",
  "properties": {
    "altitud": {
      "type": "number",
      "required": false
    },
    "latitud": {
      "type": "number",
      "required": true
    },
    "longitud": {
      "type": "number",
      "required": true
    }
  }
},
"additionalItems": false
}

```

Quando un'ontologia è conservata nella BDTR (Base Dati in Tempo Reale) la piattaforma l'arricchisce con le meta-informazioni relative al contesto d'uso dell'ontologia. Nell'esempio questa informazione è in giallo:

```

{
  "_id": {
    "$oid": "51e3dbd465701fd8e0f69828"
  },
  "contextData": {
    "session_key": "08bf50c8-6ea6-41dc-99ac-5d12a6f517a3",
    "user_id": 1,
    "kp_id": 9,
    "kp_identificador": "gatewaysensores",
    "timestamp": "1373887444356"
  },
  "SensorTemperatura": {
    "coordenadaGps": {
      "altitud": 0,
      "latitud": 40.512274,
      "longitud": -3.675679
    },
    "identificador": "S_Temperatura_00001",
    "medida": 19,
    "timestamp": 1373887443001,
    "unidad": "C"
  }
}

```

```
}  
,
```

Nel contextData sono riportati la chiave della sessione stabilita fra KP e SIB, l'identificatore dell'utente che usa il KP, l'identificazione del KP, l'identificatore dell'istanza di KP che è connessa e il timestamp dell'inserzione dell'informazione.

