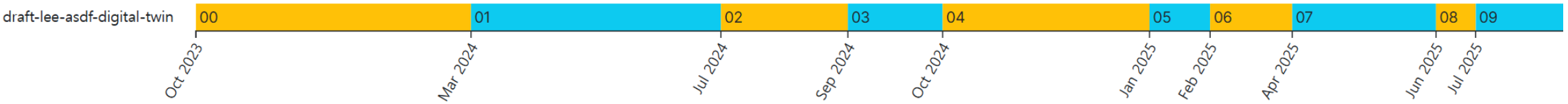
Semantic Definition Format (SDF) modeling for Digital Twin

draft-lee-asdf-digital-twin-09

July 2025

Hyunjeong Lee, hjlee294@etri.re.kr
Jungha Hong, jhong@etri.re.kr
ETRI

Status



- draft-lee-asdf-digital-twin-00 (IETF 118, Prague, Nov. 2023)
  - Introducing SDF extensions for digital twin
- draft-lee-asdf-digital-twin-01 (ASDF interim, online, Mar. 2024)
  - Adding the examples of a boat's location description
- draft-lee-asdf-digital-twin-02 (ASDF interim, online, Jul. 2024)
  - Adding sdfLocation to SDF model (Figure 1)
- draft-lee-asdf-digital-twin-03 (ASDF interim, online, Sep. 2024)
  - Adding requirements for digital twin
- draft-lee-asdf-digital-twin-04 (IETF 121, Dublin, Nov. 2024)
  - Adding the basic architecture of digital twin with SDF objects
- draft-lee-asdf-digital-twin-05 (ASDF interim, online, January 2025)
  - Adding the structure of SDF location in 5.1
- draft-lee-asdf-digital-twin-06 (ASDF interim, online, February 2025)
  - Split digital twin from the previous version 05
- draft-lee-asdf-digital-twin-07 (ASDF interim, online, April 2025)
  - Mapping information attributes of digital twin to qualities of sdfThing
- draft-lee-asdf-digital-twin-08 (ASDF interim, online, June 2025)
  - Complement SDF mapping and an example modeling in Chapter 4
- draft-lee-asdf-digital-twin-09 (ASDF interim, online, July 2025)
  - Example of relationship modeling and procedure for digital twin implementation
  - Candidate for WG Adoption

Table of contents

- Example of relationship modeling
- Procedure for digital twin implementation

Table of Contents

1. Introduction 2
2. Terminology 3
3. SDF structure for digital twin 3
4. Motivation and design rationale 4
4.1. Introduction of sdfNonAffordance 5
4.2. Digital Twin-Centric Modeling within sdfThing 5
4.3. Example modeling 6
5. Requirements for digital twin 8
5.1. Overview 8
5.2. Data acquisition 8
5.3. Data analysis 8
5.4. Identification 9
5.5. Accuracy 9
5.6. Synchronization 9
6. Security Considerations 9
7. IANA Considerations 9
8. References 9
8.1. Normative References 9
8.2. Informative References 10
Authors' Addresses 10

Table of Contents

1. Introduction 2
2. Terminology 3
3. SDF structure for digital twin 3
4. Motivation and design rationale 4
4.1. Introduction of sdfNonAffordance 5
4.2. Digital Twin-Centric Modeling within sdfThing 5
5. Examples 6
5.1. Boat modeling 6
5.2. Relationship modeling 7
6. Requirements for digital twin 9
7. Procedure for digital twin implementation 10
7.1. Overview 10
7.2. Identifying and scoping physical assets 11
7.3. Defining a digital twin 11
7.4. Metadata and contextualization 11
7.5. Binding Interfaces and Communications 11
7.6. Verification and compliance 11
7.7. Deployment and registration 11
7.8. Runtime monitoring and updating 12
7.9. Lifecycle and governance management 12
8. Security Considerations 12
9. IANA Considerations 12
10. References 12
10.1. Normative References 12
10.2. Informative References 13
Authors' Addresses 13

Digital Twin Modeling within sdfThing

- Mapping information attributes of Digital twin (source: ISO 23247-3) to SDF qualities

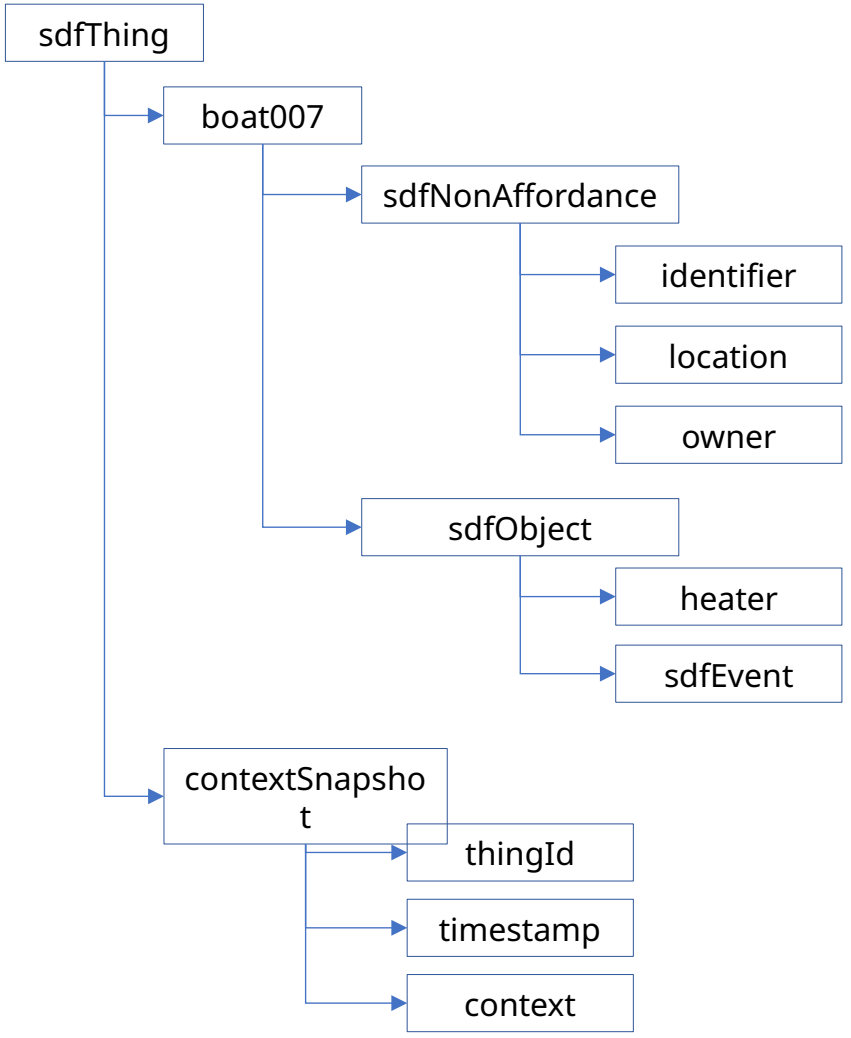
Attribute	Recommended Mapping	Description
Identifier	sdfNonAffordance	Globally unique digital twin ID (e.g., URN)
Characteristic	sdfProperty or sdfData	General description or domain properties
Schedule	sdfEvent or sdfData	Time-based actions, availability, or maintenance
Status	sdfAction or sdfProperty	Actual or calculated operating conditions
Location	sdfNonAffordance	Physical or logical location information
Report	sdfData	Measurement summaries, analytics, or logs
owner	sdfNonAffordance	Organization or entity responsible for the digital twin
Relationship	sdfRelation	Inter-object/inter-twin relationships

Table 1: Digital twin modeling within sdfThing

An example of SDF mapping for digital twin (1/2)

```
{
  "sdfThing": {
    "boat007": {
      "label": "Digital Twin of Boat #007",
      "sdfRequired": {
        "heater1": "#/sdfObject/heater"
      },
      "sdfNonAffordance": {
        "identifier": { "type": "string", "const": "urn:boat:007:heater:1" },
        "location": { "type": "string", "default": "Deck A, Port Side" },
        "owner": { "type": "string", "default": "OceanTech Ltd." }
      }
      "sdfObject": {
        "heater": {
          "label": "Cabin Heater",
          "sdfProperty": {
            "status": { "type": "string", "enum": ["on", "off", "error"], "default": "off" },
            "characteristic": { "type": "string", "default": "12V electric heater, 800W" }
          },
          "sdfEvent": {
            "maintenanceSchedule": { "type": "string", "format": "date-time" }
          },
          "nipcProtocol": "zigbee"
        }
      },
    }
  },
  "contextSnapshot": {
    "thingId": "boat-007",
    "timestamp": "2025-06-20T09:00:00Z",
    "context": {
      "location": { "lat": "35.2988", "lon": "129.2547", "alt": "0.0" },
      "installationInfo": { "floor": 1, "mountType": "wall" }
    }
  }
}
```

Figure 2: An example of SDF mapping for digital twin

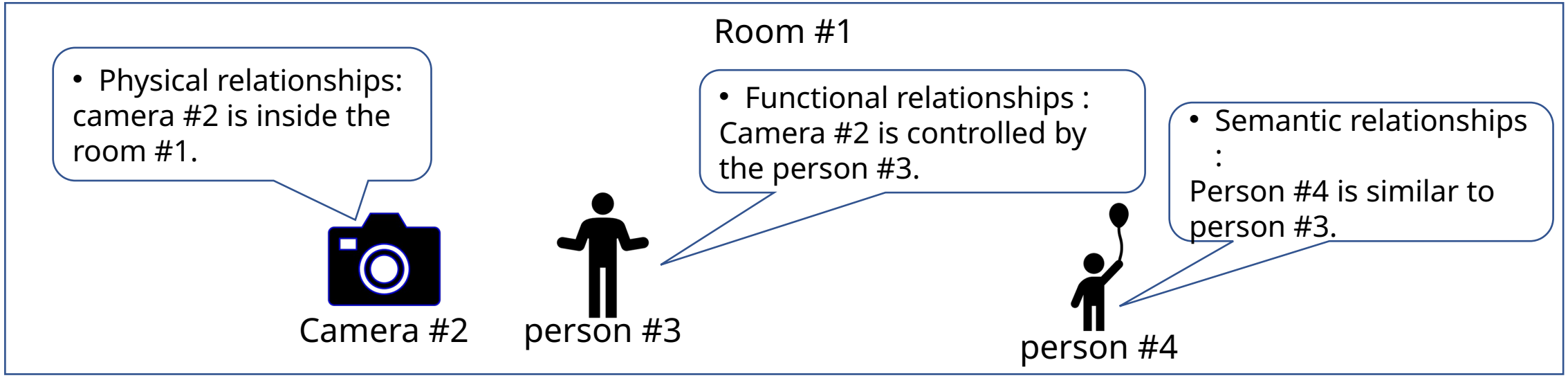


An example of SDF mapping for digital twin (2/2)

- Figure 2
  - Slightly modified so that sdfObject is within boat007
- Boat007 includes
  - sdfNonAffordance for the static information, e.g., ID, location and owner
  - sdfProperty for dynamic information, e.g., status, events and maintenance schedule
- ContextSnapshot
  - captures a full metadata state for a digital twin at a point in time

Relationship modeling for digital twin

- Relationships can include:
  - Physical relationships (e.g., "inside", "next to")
  - Functional relationships (e.g., "controls", "is controlled by")
  - Semantic relationships (e.g., "similar to", "same as")



Example of sdfRelation

```
{
  "namespace": {
    "saref": "https://saref.etsi.org/saref4bldg/v1.1.2/"
  },
  "sdfObject": {
    "lightbulb": {
      "description": "A smart lightbulb",
      "sdfProperty": {
        "adjacent-node": {
          "type": "object",
          "sdfType": "link"
        }
      },
      "sdfRelation": {
        "sameRoomAsThermostat": {
          "relType": "saref:isLocatedIn",
          "target": "#/sdfObject/thermostat",
          "description": "This lightbulb is located in the
same room as the thermostat.",
          "label": "Located together"
        }
      }
    },
    "thermostat": {
      "description": "A thermostat in the same room",
      "sdfProperty": {
        "adjacent-node": {
          "type": "object",
          "sdfType": "link"
        }
      }
    }
  }
}
```

Figure 3: An example of sdfRelation

- We can recognize that the lightbulb and the thermostat are in the same room using sdfRelation.
- The physical relationship between humans and robots in the factory is very important information for safety.

Procedure for digital twin implementation

1. **Identify** the physical assets.
2. **Define** digital twins using affordances and metadata.
3. **Contextualize** with non-affordance info.
4. **Bind** to protocols like MQTT, CoAP, or HTTP.
5. **Validate** with schema and semantic rules.
6. **Deploy** in registries or clouds.
7. **Monitor** and **update** with real-time data.
8. **Manage** versions and compliance lifecycle.

Next steps

- Add more examples for various domains

- Provide the protocols and data formats to represent and use an sdfThing as a digital twin

Thank you!