



SDF and WoT Conversion

Jan Romann, University of Bremen, Germany

T2TRG Virtual Summary Meeting, December 12, 2022



Current state of WoT specification process

- Current charter period lasts until January 31, 2023
 - Will probably be extended by three months
- Updated documents:
 - WoT TD 1.1
 - WoT Architecture 1.1
- Newly standardized document:
 - WoT Discovery 1.0
- Transitions to Candidate Recommendations this week
- Transition to Proposed Recommendations in March



Current state of WoT specification process

- New charter begins in May
 - Publishing of postponed Profile specification
 - Major updates to existing documents (e.g., TD 2.0)
 - New topics, such as a stronger focus on protocol bindings
- Current schedule: <https://github.com/w3c/wot/blob/main/charters/wg-2021-extension-plan.md>

Mapping between SDF and WoT



Motivation for Conversion between SDF and WoT

- WoT as an interesting conversion target for SDF
 - SDF focuses on interoperability between ecosystems/data models
 - WoT TD focuses on interoperability between instances/devices
 - → Both approaches complement each other
- However: Lack of a “canonical” mapping between the two specifications



Similarities between SDF and WoT TD

- Serialization format: JSON
- Interaction affordances
 - Properties
 - Actions
 - Events
- JSO-inspired data schemas and data qualities



Simple Conversion Example

SDF Model

```
{
  "sdfObject": {
    "Lamp": {
      "label": "Smart Lamp"
      "sdfProperty": {
        "status": {
          "type": "string",
          "description":
            "Status of the lamp."
        }
      }
    }
  }
}
```

WoT Thing Description

```
{
  "@context": "...",
  "title": "Smart Lamp",
  "security": [...],
  "properties":
    "status": {
      "type": "string",
      "description": "Status of the
        lamp.",
      "forms": [...]
    }
}
```

How can we map WoT-specific vocabulary to SDF?



SDF Mapping Files (with WoT TD terms)

SDF Model

```
{
  "sdfObject": {
    "Lamp": {
      "label": "Smart Lamp"
      "sdfProperty": {
        "status": {
          "type": "string",
          "description": "Status of
the lamp."
        }
      }
    }
  }
}
```

SDF Mapping File

```
{
  "map": {
    "#/sdfObject/Lamp": {
      "@context": "...",
      "security": [...]
    },
    "#/sdfObject/Lamp/sdfProperty/status": {
      "forms": [...]
    }
  }
}
```

How can we map (abstract) SDF models to WoT?

SDF \rightleftharpoons Thing Model Conversion

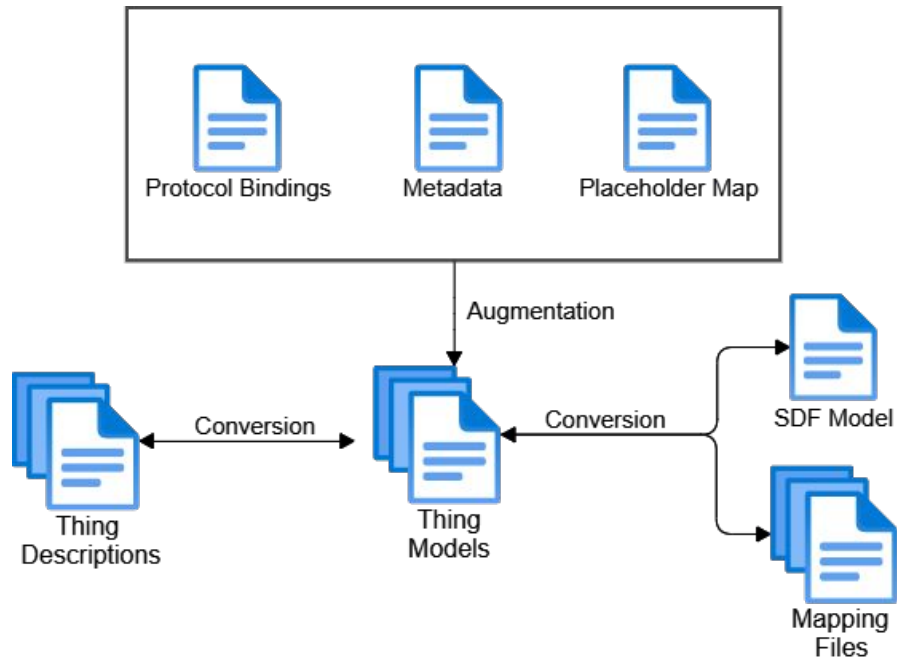
SDF Model

```
{
  "sdfObject": {
    "Lamp": {
      "label": "Smart Lamp"
      "sdfProperty": {
        "status": {
          "type": "string",
          "description":
            "Status of the lamp."
        }
      }
    }
  }
}
```

WoT Thing Model

```
{
  "@context": "...",
  "@type": "tm:ThingModel",
  "title": "Smart Lamp",
  "properties": {
    "status": {
      "type": "string",
      "description": "Status of
the lamp."
    }
  }
}
```

Conversion Process





Challenges

- Nested Models/composition
 - WoT uses linking approach for creating hierarchies
 - Slight adjustments to the `sdfThing` class were needed
 - Use of TM/TD “Collections”
- Roundtripping
 - Use of keys prefixed with `sdf:` in WoT documents
 - Not possible when resolving external references

Converter Implementation

sdf-wot-converter 1.9.1

```
pip install sdf-wot-converter
```



Latest version

Released: Jul 25, 2022

🐍 Support Python in 2022! End of year fundraiser and membership drive are live now!

[Join in today!](#)

Navigation

Project description

Release history

Download files

Project links

Homepage

Bug Tracker

Statistics

GitHub statistics:

★ Stars: 3

🍴 Forks: 0

🔖 Open issues/PRs: 2

[View statistics for this project via](#)

Project description

Build Status passing pypi package 1.9.1 codecov 100% code style: black

SDF-WoT-Converter

This repository provides a Python-based converter from [SDF](#) to [WoT TD](#) including Thing Models.

The converter is both usable as a library and a command line tool. It provides conversion functions between WoT Thing Descriptions, WoT Thing Models and SDF Models (one for each combination). You can find a number of examples for the usage of the converter down below as well as overviews for the conversion between SDF and WoT TMs.

The CI pipeline is set up to automatically convert all (valid) models from the [oneDM playground](#) to WoT Thing Models and upload to the results to [this repository](#).

Installation

You can install the converter using pip:

```
pip install sdf-wot-converter
```

SDF WoT converter

SDF

» «

WoT TM

```
{
  "info": {
    "title": "Example file for OneDM Semantic Definition Format",
    "version": "2019-04-24",
    "copyright": "Copyright 2019 Example Corp. All rights reserved.",
    "license": "https://example.com/license"
  },
  "namespace": {
    "cap": "https://example.com/capability/cap"
  },
  "defaultNamespace": "cap",
  "sdfObject": {
    "Switch": {
      "sdfProperty": {
        "value": {
          "description": "The state of the switch; false for off and true for on.",
          "type": "boolean"
        }
      },
      "sdfAction": {
        "on": {
          "description": "Turn the switch on; equivalent to setting value to true."
        },
        "off": {
          "description": "Turn the switch off; equivalent to setting value to false."
        },
        "toggle": {
          "description": "Toggle the switch; equivalent to setting value to its complement."
        }
      }
    }
  }
}
```

Clear

Insert Example

Format

```
{
  "@context": [
    "https://www.w3.org/2022/wot/td/v1.1",
    {
      "cap": "https://example.com/capability/cap",
      "sdf": "https://example.com/sdf"
    }
  ],
  "@type": "tm:ThingModel",
  "sdf:objectKey": "Switch",
  "sdf:title": "Example file for OneDM Semantic Definition Format",
  "sdf:copyright": "Copyright 2019 Example Corp. All rights reserved.",
  "links": [
    {
      "href": "https://example.com/license",
      "rel": "license"
    }
  ],
  "version": {
    "model": "2019-04-24"
  },
  "sdf:defaultNamespace": "cap",
  "actions": {
    "on": {
      "description": "Turn the switch on; equivalent to setting value to true."
    },
    "off": {
      "description": "Turn the switch off; equivalent to setting value to false."
    }
  }
}
```

Clear

Insert Example

Format

Settings


☒ Output SDF Mapping files ☒ Include additional fields for roundtripping


SDF Conversion Tool Collection

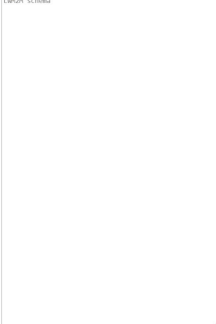
<http://wishi.nomadiclab.com/sdf-converter/>


Conversion tools for SDF


These are experimental tools for making conversions from data models in different IoT ecosystems to and from the Semantic Definition Format (SDF) models. Please check the [information page](#)

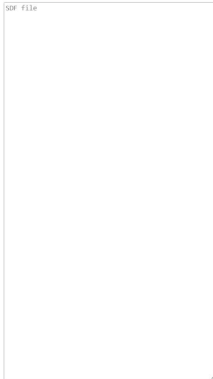
 LwM2M - IPSO
By Ericsson Research








 OneDM - SDF







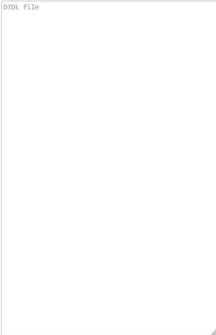
 Web of Things – Thing Model
By Jan Romann / Uni Bremen








 MS Azure DTDL
By Ericsson Research





 IETF YANG Data Model
By Jana Kiesewalter / Uni Bremen





Clear fields

Load Data Model

Select SDF model

LOAD SDF

Select IPSO model

LOAD IPSO

Select DTDL model

LOAD DTDL

Convert to other formats

CONVERT

Conclusion and Outlook



Conclusion and Outlook

- WoT and SDF can be mapped to each other
 - Additional concepts such as SDF mapping files are needed
 - WoT TMs as intermediaries
- Flexible converter implementation in Python
 - Library, CLI tool, and web interface
- However: More standardization work needed
 - “Canonical” mapping specification?
 - WoT Linking/sdfRelations
 - Nested TMs/TDs in a single document?

Backup



Roundtripping

SDF (sdfChoice)

```
{
  ...,
  "sdfChoice": {
    "foo": {
      "const": 2
    },
    "bar": {
      "const": 5
    }
  }
}
```

WoT (enum)

```
{
  ...,
  "enum": [
    {
      "sdf:choiceName": "foo",
      "const": 2
    },
    {
      "sdf:choiceName": "bar",
      "const": 5
    }
  ]
}
```



Composition: TM/TD Collections

```
{
  "model1": {
    "@context": "https://www.w3.org/2022/wot/td/v1.1",
    "links": [{ "href": "#/model2", "rel": "tm:submodel" }]
  },
  "model2": {
    "@context": "https://www.w3.org/2022/wot/td/v1.1",
    "title": "This is a submodel!"
  }
}
```



Conversion of Nested Models

SDF Model (old)

```
{
  "sdfThing": {
    "TopLevel": {
      "sdfObject": {
        "SecondLevel": {
          ...
        }
      }
    }
  }
}
```

WoT "TM Collection"

```
{
  "TopLevel": {
    ...,
    "properties": { ... },
    "links": [
      {
        "href": "#/SecondLevel",
        "rel": "tm:submodel"
      }
    ]
  },
  "SecondLevel": {
    ...
  }
}
```



Extension of sdfThing with Affordances

SDF Model (new)

```
{
  "sdfThing": {
    "TopLevel": {
      "sdfProperty": {
        ...
      },
      "sdfObject": {
        "SecondLevel": {
          ...
        }
      }
    }
  }
}
```

WoT "TM Collection"

```
{
  "TopLevel": {
    ...,
    "properties": { ... },
    "links": [
      {
        "href": "#/SecondLevel",
        "rel": "tm:submodel",
      }
    ],
  },
  "SecondLevel": {
    ...
  }
}
```