Transforming Network Data to RDF using RML (RDF Mapping Language)

YANG Mapping Example

- Yet Another Next Generation (YANG) is a data modeling language for the definition of data sent over network management protocols
- RDF (Resource Description Framework) is a data model created by W3C to describe and exchange graph data
- This tool allows to transform XML network configuration data to RDF using declarative scripting language to map structured and semi-structured data (e.g. relational databases, CSV files, and JSON) into RDF format

RML Mapping Language

Source Descriptions

```
:Triplesmap a rr:TriplesMap;

    rml:logicalSource [
        a rml:LogicalSource;
        rml:source "P1_config.xml";
        rml:referenceFormulation ql:XPath;
        rml:iterator "/*[local-name()='rpc-reply']/*[local-name()='data']/*[local-name()='ifm']/*[local-name()='interfaces']/*[local-name()='interface']"
    ];
```

File Path

File Format

Data Path

```
    rr:subjectMap [
        rr:template "http://huawei.com/1.1.1.1/{string-join(ancestor-or-self::*[local-name()!='rpc-reply']/name(), '/')}/{*[local-name()='name']/text()}" ; ];

    rr:predicateObjectMap [
        rr:predicate huawei-yang:interfaceName ;
        rr:objectMap [ rml:reference "*[local-name()='name']/text()" ];
    rr:graphMap [ rml:reference "string-join(ancestor-or-self::*/name(), '/') || '/name'" ] ;

    ] ;
.
```

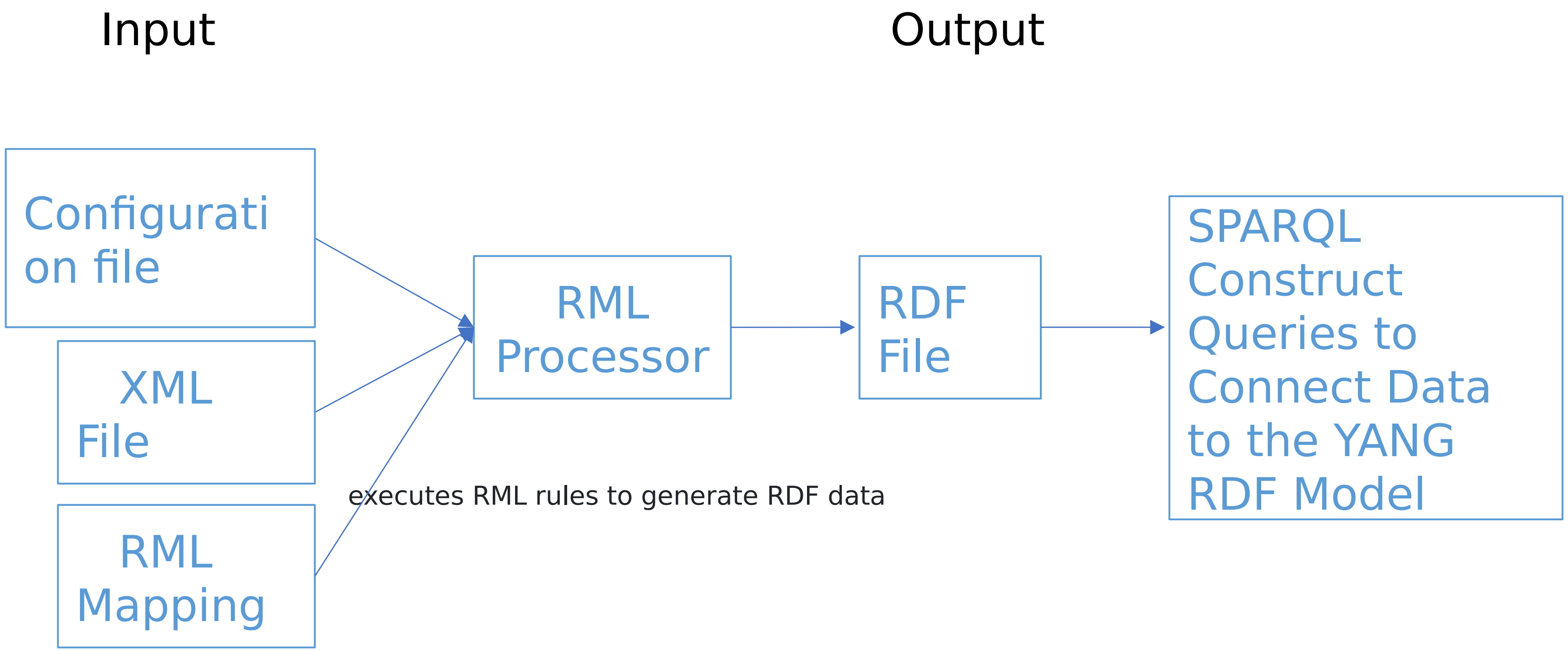
Subject

Predicate

Object

Graph

Triple Descriptions



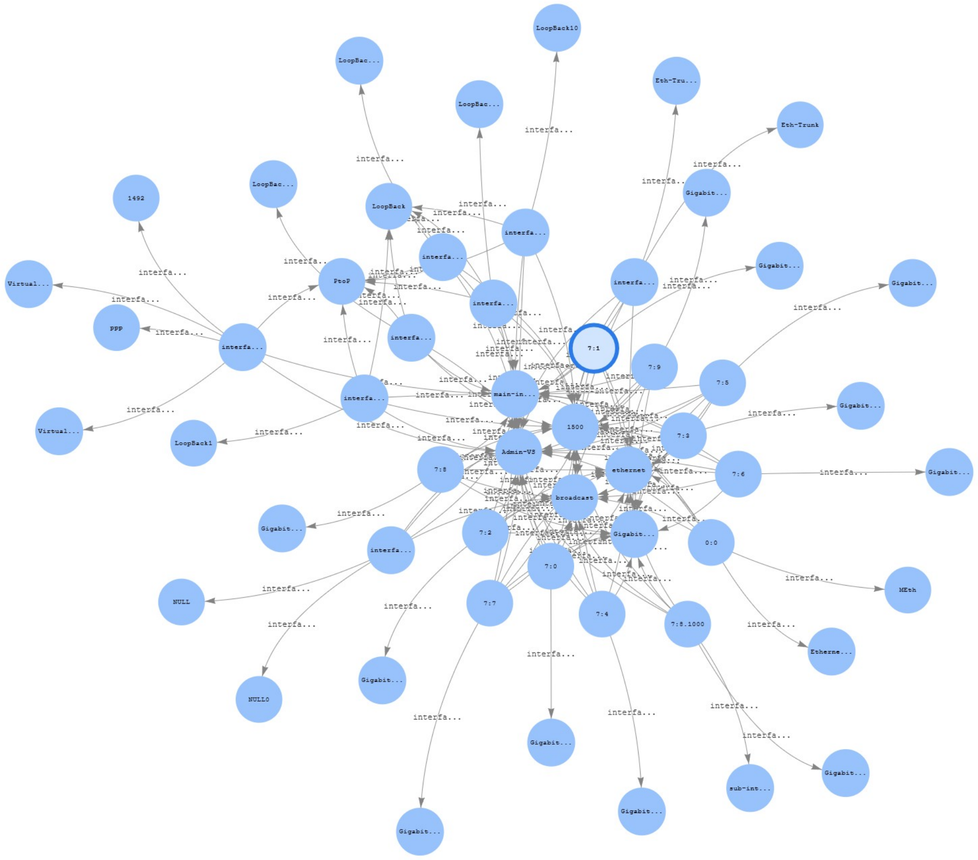
Conversion From Logical Source (XML, JSON, CSV etc) to RDF data

Summary

- **Transforms data modelled in YANG to RDF using RML**
  - **Data is produced using Named Graph model which adds context to the data**
  - **Context represents the path to the data**
- **Processes produced RDF triples, and for each triple**
  - Runs a SPARQL query to find contexts which are described as types in **YANG model**
  - **extracts and filters types and appends the found element back to the original triple**
  - **Writes the modified RDF dataset to a new file.**

Query

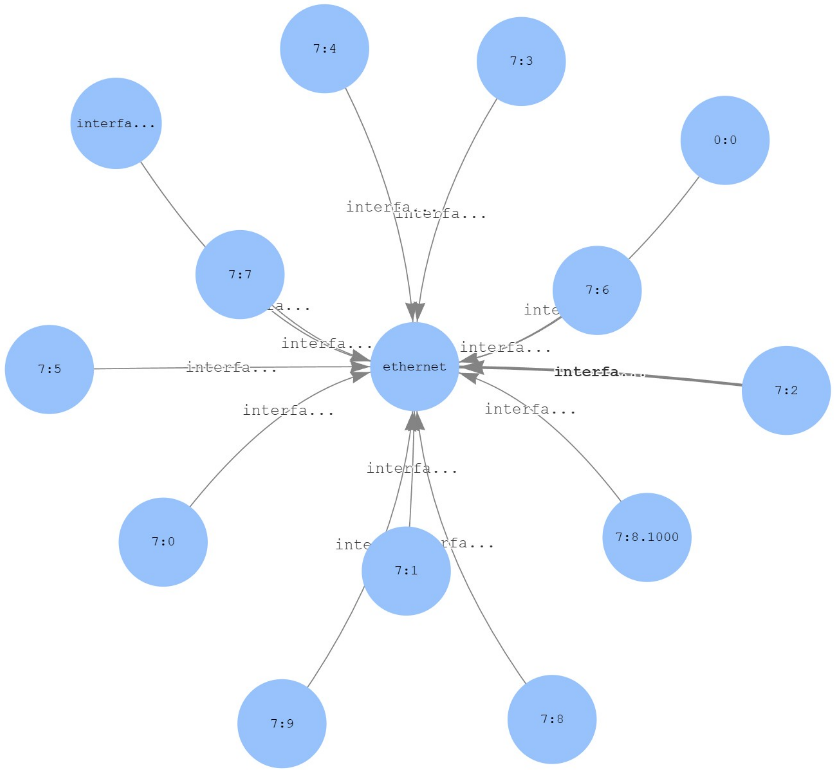
- All the instances of interfaces in the data (for a small amount of example data)



Query

- Query for specific broadcast router type at a leaf level

- SELECT * WHERE {?s ?p ?o. ?p a <http://www.typefox.io/yang/Yang#
- FILTER (?o="broadcast")}



Git: https://github.com/Huawei-IOAM/yang2rdf
Folder: yang-mapping-example

Conclusions

- **RML is powerful** because it follows a **declarative approach**, making it easier to describe mappings
  - However, **network data** is often highly structured, hierarchical, and dynamic, making it **challenging to model** with a static declarative approach
  - **Alternative?** A hybrid approach using **RML with preprocessing scripts**
- **Scalability is a Major Issue: Tools Are Memory Dependent & Files Are Too Big**
  - **RML tools (e.g., RMLMapper, SDM-RDFizer) work well for small datasets**.
    However, **network data dumps (e.g., YANG models, XML-based configurations) can be huge**, leading to:
    - **High memory consumption**.
    - **Slow processing times**.
    - **Scalability issues when handling large datasets**.
- **Too Many Elements to Define Manually in Both Data & YANG Model**
  - **Network models like YANG define thousands of elements**, and mapping them manually to RDF **is impractical**.
    The **lack of an automated schema mapping tool** means that users must:
  - Define each **YANG element manually** in the schema.
- **At Least 3 Layers Are Required to Fully Model Network Data**
  - A network configuration typically involves **three layers**: **Device-level configurations** (Physical interfaces, hardware details).

  Logical configurations (VLANs, IP routing, protocols). **Service-level data** (QoS, monitoring, telemetry).
  - **RML struggles with cross-layer relationships**, as it lacks built-in inference for **network dependencies**.