

Web ngữ nghĩa

Soạn bởi: Nguyễn Bá Ngọc

Chương 9

Hà Nội-2021

Chương 9.

SHACL

Nội dung

9.1. Các thành phần cơ bản

9.2. Khuôn hình và các ràng buộc

9.3. SHACL-SPARQL

9.4. Luật và các mở rộng

Nội dung

9.1. Các thành phần cơ bản

9.2. Khuôn hình và các ràng buộc

9.3. SHACL-SPARQL

9.4. Luật và các mở rộng

SHACL

- Quy chuẩn W3C: <https://www.w3.org/TR/shacl/> (T7/2017)
- Khởi nguồn từ SPIN, OSLC & và ShEx
- Bao gồm 2 phần: SHACL-Core, SHACL-SPARQL
- Sử dụng cú pháp RDF

Các triển khai SHACL

Tên triển khai	Các phần	Ngôn ngữ	Liên kết
Topbraid SHACL API	SHACL Core, SPARQL	Java (Jena)	TopBraid composer
SHACL playground	SHACL Core	JS (rdflib.js)	http://shacl.org/playground/
SHACL-S Part of SHacLEX	SHACL Core	Scala (Jena, RDF4j)	http://rdfshape.weso.es
pySHACL	SHACL Core, SPARQL	Python (rdflib)	https://github.com/RDFLib/pySHACL
Corese SHACL	SHACL Core, SPARQL	Java (STTL)	http://wimmics.inria.fr/corese
RDFUnit	SHACL Core, SPARQL	Java (Jena)	https://github.com/AKSW/RDFUnit
Jena SHACL	SHACL Core, SPARQL	Java (Jena)	https://jena.apache.org/
RDF4j SHACL	SHACL Core	Java (RDF4J)	https://rdf4j.org
Stardog	SHACL Core, SPARQL	Java	https://www.stardog.com
Zazuko SHACL	SHACL Core	JS	https://github.com/zazuko/rdf-validate-shacl

Ví dụ cơ bản

```
@prefix :      <http://semweb.edu.vn/people#> .  
@prefix sh:    <http://www.w3.org/ns/shacl#> .  
@prefix xsd:   <http://www.w3.org/2001/XMLSchema#> .
```

```
:UserShape a sh:NodeShape ;  
  sh:targetNode :alice, :bob, :carol ;  
  sh:nodeKind sh:IRI ;  
  sh:property :hasName,  
             :hasEmail .  
:hasName sh:path :name ;  
  sh:minCount 1;  
  sh:maxCount 1;  
  sh:datatype xsd:string .  
:hasEmail sh:path :email ;  
  sh:minCount 1;  
  sh:maxCount 1;  
  sh:nodeKind sh:IRI .
```

```
@prefix : <http://semweb.edu.vn/people#> .  
:alice :name "Alice Cooper" ;  
       :email <mailto:alice@mail.org> .
```

```
:bob :firstName "Bob" ;  
     :email <mailto:bob@mail.org> .
```



```
:carol :name "Carol" ;  
       :email "carol@mail.org" .
```



Đồ thị dữ liệu

Đồ thị khuôn hình

Ví dụ với các nút rỗng

```
@prefix :      <http://semweb.edu.vn/people#> .  
@prefix sh:    <http://www.w3.org/ns/shacl#> .  
@prefix xsd:   <http://www.w3.org/2001/XMLSchema#> .
```

```
:UserShape a sh:NodeShape ;  
  sh:targetNode :alice, :bob, :carol ;  
  sh:nodeKind sh:IRI ;  
  sh:property [sh:path :name ;  
    sh:minCount 1;  
    sh:maxCount 1;  
    sh:datatype xsd:string];  
  sh:property [sh:path :email ;  
    sh:minCount 1;  
    sh:maxCount 1;  
    sh:nodeKind sh:IRI] .
```

```
@prefix : <http://semweb.edu.vn/people#> .  
:alice :name "Alice Cooper" ;  
       :email <mailto:alice@mail.org> .
```

```
:bob :firstName "Bob" ;  
     :email <mailto:bob@mail.org> .
```



```
:carol :name "Carol" ;  
       :email "carol@mail.org" .
```

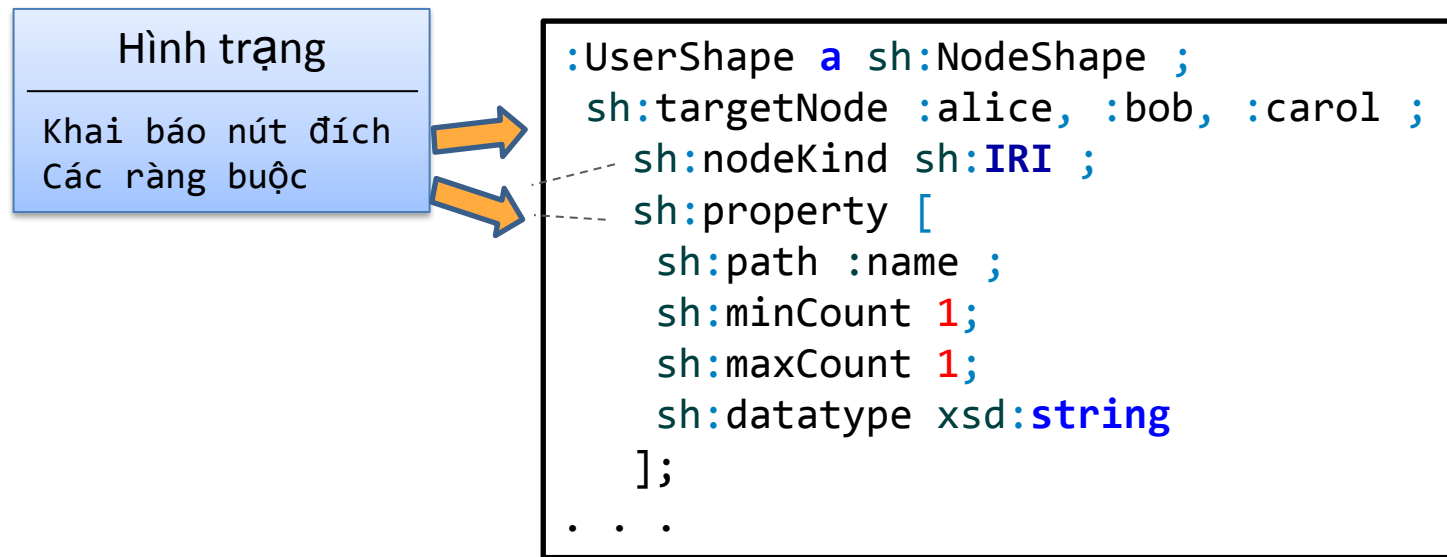


Đồ thị dữ liệu

Đồ thị khuôn hình

Các khái niệm cơ bản của SHACL

- Shape/Hình trạng - Tập các nút đích và các thành phần ràng buộc.
 - Nút đích: Các nút trong đồ thị được kiểm tra và phải tương thích với các yêu cầu.
 - Các thành phần ràng buộc: Các điều kiện mà nút phải đáp ứng.



Đồ thị hình trạng và đồ thị dữ liệu

- Về mặt lô-gic các đồ thị được phân chia thành:
 - Đồ thị khuôn hình: Chứa các khuôn hình
 - Đồ thị dữ liệu: Chứa dữ liệu cần được kiểm tra
 - !Các đồ thị đều là những đồ thị RDF

```
:UserShape a sh:NodeShape ;  
    sh:targetNode :alice, :bob,  
:carol ;  
    sh:nodeKind sh:IRI ;  
    sh:property [  
    sh:path :name ;  
    sh:minCount 1;  
    sh:maxCount 1;  
    sh:datatype xsd:string] ;  
. . .
```

Đồ thị khuôn hình

```
:alice :name "Alice Cooper" ;  
    :email <mailto:alice@mail.org> .  
  
:bob :firstName "Bob" ;  
    :email <mailto:bob@mail.org> .  
  
:carol :name "Carol" ;  
    :email "carol@mail.org" .
```

Đồ thị dữ liệu

Kết quả kiểm tra

- Đầu ra của tiến trình kiểm tra là 1 danh sách lỗi vi phạm
- Không có lỗi \Rightarrow RDF tương thích với đồ thị hình trạng

```
[ a          sh:ValidationReport ;
  sh:conforms false ;
  sh:result  [
    a          sh:ValidationResult ;
    sh:focusNode :bob ;
    sh:message  "MinCount violation. Expected 1, obtained: 0" ;
    sh:resultPath :name ;
    sh:resultSeverity sh:Violation ;
    sh:sourceConstraintComponent
      sh:MinCountConstraintComponent ;
    sh:sourceShape :hasName
  ] ;
  ...
```

```
[ a sh:ValidationReport ;
  sh:conforms true
].
```

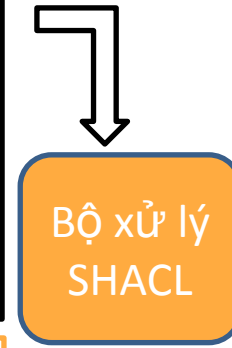
Sơ đồ nguyên lý

Đồ thị
khuôn hình

```
:UserShape a sh:NodeShape ;  
  sh:targetNode :alice,  
  :bob, :carol ;  
  sh:nodeKind sh:IRI ;  
  sh:property :hasName,  
              :hasEmail .  
:hasName sh:path :name ;  
  sh:minCount 1;  
  sh:maxCount 1;  
  sh:datatype xsd:string .  
. . .
```

Đồ thị
dữ liệu

```
:alice :name "Alice Cooper" ;  
       :email  
<mailto:alice@mail.org> .  
:bob   :name "Bob" ;  
       :email  
<mailto:bob@mail.org> .  
:carol :name "Carol" ;  
       :email  
<mailto:carol@mail.org> .
```



Kết quả kiểm tra

```
[ a  
  sh:ValidationReport ;  
  sh:conforms true  
].
```

Chèn các đồ thị hình trạng

Bộ xử lý SHACL dò theo các khai báo `owl:imports` và mở rộng đồ thị khuôn hình đang được xử lý bằng các đồ thị được chèn vào.

```
:UserShape a sh:NodeShape ;  
  sh:targetNode :alice,  
:bob, :carol ;  
  sh:nodeKind sh:IRI ;  
  sh:property :hasName .  
:hasName sh:path :name ;  
  sh:minCount 1;  
  sh:maxCount 1;  
  sh:datatype xsd:string .
```

```
<> owl:imports  
<http://example.org/UserShapes> .  
  
:TeacherShape a sh:NodeShape;  
  sh:targetClass :Teacher ;  
  sh:node :UserShape ;  
  sh:property [  
    sh:path :teaches ;  
    sh:minCount 1;  
    sh:datatype xsd:string;  
  ] .
```

Nội dung

9.1. Các thành phần cơ bản

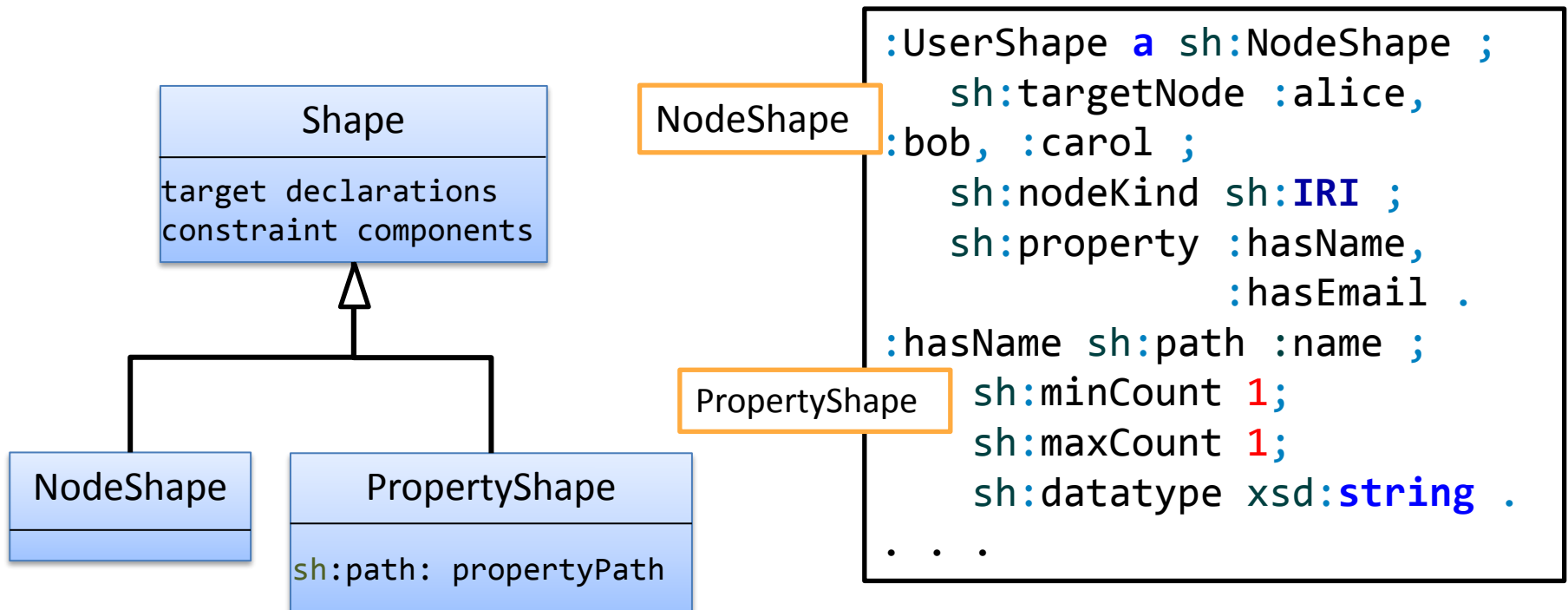
9.2. Khuôn hình và các ràng buộc

9.3. SHACL-SPARQL

9.4. Luật và các mở rộng

Phân loại khuôn hình

- Có 2 loại khuôn hình:
 - NodeShape: Khuôn hình cho các nút
 - PropertyShape: Khuôn hình cho các giá trị trên các đường dẫn thuộc tính của nút.



Khuôn hình nút

- Các ràng buộc đối với nút được kiểm tra

```
:UserShape a sh:NodeShape ;  
    sh:nodeKind      sh:IRI ;  
    sh:targetClass   :User .
```

```
:alice a :User .
```

```
<http://example.org/bob> a :User .
```

```
_:1 a :User .
```



Khuôn hình thuộc tính

- Các ràng buộc gắn với thuộc tính và các giá trị của nó đối với nút được kiểm tra
 - `sh:property` Gắn kết 1 khuôn hình với 1 khuôn hình thuộc tính.
 - `sh:path` Xác định đường dẫn.

```
:User a sh:NodeShape ;  
  sh:property [  
    sh:path      :email ;  
    sh:nodeKind  sh:IRI  
  ] .
```

```
:alice a :User ; :email <mailto:alice@mail.org> .
```

```
:bob a :User; :email <mailto:bob@mail.org> .
```

```
:carol a :User; :email "carol@mail.org" .
```



Đường dẫn trong khuôn hình thuộc tính

- Tập con của đường dẫn thuộc tính SPARQL
 - `inversePath`
 - `alternativePath`
 - `zeroOrMorePath`
 - `oneOrMorePath`
 - `zeroOrOnePath`

```
:alice a :User;  
       :follows :bob .  
  
:bob   a :User .  
  
:carol a :User;  
       :follows :alice .  
  
_:1 :follows :bob .
```



```
:User a sh:NodeShape, rdfs:Class ;  
      sh:property [  
        sh:path [sh:inversePath :follows ];  
        sh:nodeKind sh:IRI ;  
      ] .
```

Các thành phần ràng buộc

- Các nút khai báo ràng buộc được gắn với khuôn hình
- Chúng có các tham số với các giá trị đặc tả ràng buộc
- SHACL-core cung cấp 1 danh sách các thành phần ràng buộc đã được định nghĩa sẵn
 - Hầu hết đều có 1 tham số đại diện.

Quy cách:

Tham số: `sh:xx`

Thành phần ràng buộc: `sh:xxConstraintComponent`

```
:UserShape a sh:NodeShape ;  
  sh:nodeKind sh:IRI .
```

Tham số
`sh:nodeKind`

Giá trị của tham số
`sh:IRI` ;

Thành phần ràng buộc
`sh:nodeKindConstraintComponent`

Lưu ý: Có thể tự định nghĩa các thành phần ràng buộc với SHACL-SPARQL

Lặp tham số

Mỗi giá trị tham số khai báo 1 ràng buộc khác

```
:UserShape a sh:NodeShape;  
  sh:class foaf:Person ;  
  sh:class :Person .
```

```
:alice a :Person, foaf:Person .
```

```
:bob a :Person .
```



Các thành phần ràng buộc trong SHACL Core

Type	Ràng buộc
Cơ số	minCount, maxCount
Kiểu của giá trị	class, datatype, nodeKind
Giá trị	node, in, hasValue, property
Khoảng giá trị	minInclusive, maxInclusive minExclusive, maxExclusive
Thuộc tính của chuỗi	minLength, maxLength, pattern
Ngôn ngữ	languageIn, uniqueLang
Liên kết lô-gic	not, and, or, xone
Hình trạng đóng	closed, ignoredProperties
Ràng buộc theo cặp	equals, disjoint, lessThan, lessThanOrEquals
Thành phần bổ trợ	name, description, order, group
Hình trạng có định kiểu	qualifiedValueShape, qualifiedValueShapesDisjoint qualifiedMinCount, qualifiedMaxCount

Thông điệp thân thiện với người dùng

- Khai báo thông điệp được đưa ra trong báo cáo trong trường hợp có vi phạm.

```
:bob a :User; :alias "Bob" .
```



```
:UserShape a sh:NodeShape ;  
  sh:targetClass :User ;  
  sh:property [  
    sh:path      :name ; sh:minCount 1 ;  
    sh:message   "Where is the name?"  
  ] .
```

```
:report a :ValidationReport ;  
sh:conforms false ;  
sh:result [ a sh:ValidationResult ;  
  sh:resultSeverity      sh:Violation ;  
  sh:sourceConstraintComponent sh:MinCountConstraintComponent ;  
  sh:sourceShape         ... ;  
  sh:focusNode           :bob ;  
  sh:resultPath          :name ;  
  sh:resultMessage       "Where is the name?" ;  
] .
```

Các mức cảnh báo

- Xác định mức độ vi phạm, đã định nghĩa sẵn 3 mức độ: Violation (Mặc định), Warning, Info

```
:bob a :User ; :alias "Bob" .
```

```
:UserShape a sh:NodeShape ;  
sh:targetClass :User ;  
sh:property [  
  sh:path :name ;  
  sh:datatype xsd:string ;  
  sh:severity sh:Warning  
] .
```

```
:report a :ValidationReport ;  
sh:conforms false ;  
sh:result [ a sh:ValidationResult ;  
  sh:resultSeverity sh:Warning ;  
  sh:sourceConstraintComponent sh:MinCountConstraintComponent ;  
  sh:sourceShape ... ;  
  sh:focusNode :bob ;  
  sh:resultPath :name ;  
  sh:resultMessage "MinCount Error" ;  
] .
```

Vô hiệu khuôn hình

- Có thể vô hiệu khuôn hình trong đồ thị, ví dụ khi nhập vào

UserShapes

```
:UserShape a sh:NodeShape;  
  sh:targetClass :User ;  
  sh:property :HasName, :HasEmail .  
:HasName sh:path :name ; sh:datatype xsd:string .  
:HasEmail sh:path :email ; sh:minCount 1; sh:nodeKind sh:IRI .
```

```
<> owl:imports <UserShapes> .  
  
:TeacherShape a sh:NodeShape;  
  sh:targetClass :Teacher ;  
  sh:node :UserShape ;  
  sh:property [ sh:path :teaches ;  
    sh:minCount 1;  
    sh:datatype xsd:string;  
  ] .  
  
:HasEmail sh:deactivated true .
```


Xác định các đối tượng được kiểm tra

- Xác định các nút cụ thể cần được kiểm tra theo khuôn hình
- Các lựa chọn:

Bộ lọc	Mô tả
targetNode	Liệt kê các nút cụ thể
targetClass	Các phần tử của lớp
targetSubjectsOf	Các phần tử là chủ thể của thuộc tính
targetObjectsOf	Các phần tử là giá trị của thuộc tính

Liệt kê nút

- Trực tiếp khai báo các nút cần được kiểm tra theo khuôn hình

```
:alice :name "Alice Cooper" ;  
       :email <mailto:alice@mail.org> .  
  
:bob   :givenName "Bob" ;  
       :email <mailto:bob@mail.org> .  
  
:carol :name "Carol" ; :email "carol@mail.org" .
```

```
:UserShape a sh:NodeShape ;  
  sh:targetNode :alice, :bob, :carol ;  
  sh:property [ sh:path :name ;  
    sh:minCount 1; sh:maxCount 1;  
    sh:datatype xsd:string ;  
  ] ;  
  sh:property [ sh:path :email ;  
    sh:minCount 1; sh:maxCount 1;  
    sh:nodeKind sh:IRI ;  
  ] .
```

Lựa chọn theo lớp

Lựa chọn các phần tử của 1 lớp, tìm các khai báo rdf:type:

```
:UserShape a sh:NodeShape ;  
  sh:targetClass :User ;  
  sh:property [  
    sh:path :name ;  
    sh:minCount 1;  
    sh:maxCount 1;  
    sh:datatype xsd:string ;  
  ] ;  
  sh:property [  
    sh:path :email ;  
    sh:minCount 1;  
    sh:maxCount 1;  
    sh:nodeKind sh:IRI ;  
  ] .
```

```
:alice a :User;  
      :name "Alice Cooper" ;  
      :email  
<mailto:alice@mail.org> .  
  
:bob a :User;  
     :givenName "Bob" ;  
     :email  
<mailto:bob@mail.org> .  
  
:carol a :User;  
       :name "Carol" ;  
       :email "carol@mail.org" .
```

*!Ngoài ra còn suy diễn các phần tử theo
rdfs:subClassOf**

Ngầm chọn lớp đích

- Khuôn hình kiểu sh:Shape và rdfs:Class là phạm vi kiểm tra của chính nó. Khai báo targetClass được ngầm định.

```
:alice a :User;  
      :name "Alice Cooper" ;  
      :email <mailto:alice@mail.org> .  
:bob   a :User;  
      :givenName "Bob" ;  
      :email <mailto:bob@mail.org> .  
:carol a :User;  
      :name "Carol" ; :email "carol@mail.org" .
```

```
:User a sh:NodeShape, rdfs:Class ;  
  sh:property [ sh:path :name ;  
                sh:minCount 1; sh:maxCount 1;  
                sh:datatype xsd:string ;  
              ] ;  
  sh:property [ sh:path :email ;  
                sh:minCount 1; sh:maxCount 1;  
                sh:nodeKind sh:IRI ;  
              ] .
```

Lựa chọn theo chủ thể

```
:UserShape a sh:NodeShape;  
  sh:targetSubjectsOf :teaches ;  
  sh:property [sh:path :name ;  
    sh:minCount 1;  
    sh:maxCount 1;  
    sh:datatype xsd:string ;  
  ] .
```

```
:alice :teaches :Algebra ; # Tương thích :UserShape  
  :name "Alice" .  
  
:bob :teaches :Logic ; # Không tương thích :UserShape  
  foaf:name "Robert" .  
  
:carol foaf:name 23 . # Bỏ qua
```

Lựa chọn theo giá trị

```
:UserShape a      sh:NodeShape ;  
sh:targetObjectsOf :isTaughtBy ;  
sh:property [  
  sh:path      :name ;  
  sh:minCount  1;  
  sh:maxCount  1;  
  sh:datatype  xsd:string ;  
] .
```

```
:alice :name "Alice" . # Tương thích với :UserShape  
:bob foaf:name "Robert" . # Không tương thích :UserShape  
:carol foaf:name 23 .      # Bỏ qua  
:algebra :isTaughtBy :alice, :bob .
```

Các thành phần ràng buộc

Ràng buộc cơ sở

Ràng buộc	Mô tả
minCount	Giới hạn số lượng bộ-3 tối thiểu có chủ thể là nút được kiểm tra và thuộc tính được cho. Giá trị mặc định: 0
maxCount	Giới hạn số lượng bộ-3 tối đa có chủ thể là nút được kiểm tra và thuộc tính được cho. Nếu không định nghĩa = không giới hạn.

```
:User a sh:NodeShape ;  
  sh:property [  
    sh:path      :follows ;  
    sh:minCount  2 ;  
    sh:maxCount  3 ;  
  ] .
```

```
:alice :follows :bob, :carol .  
:bob    :follows :alice .  
:carol  :follows :alice,  
                                     :bob,  
                                     :carol,  
                                     :dave .
```



Kiểu dữ liệu của giá trị

Ràng buộc	Mô tả
datatype	Giới hạn kiểu của tất cả các giá trị của thuộc tính được cho đối với nút được kiểm tra.

```
:User a sh:NodeShape ;  
  sh:property [  
    sh:path      :birthDate ;  
    sh:datatype  xsd:date ;  
  ] .
```

```
:alice :birthDate "1985-08-20"^^xsd:date .
```

```
:bob    :birthDate "Unknown"^^xsd:date . 😞
```

```
:carol  :birthDate 1990 . 😞
```

Lớp của giá trị

Ràng buộc	Mô tả
class	Kiểm tra nút là phần tử của lớp, đồng thời nó cũng cho phép là phần tử của các lớp khác theo quan hệ <code>rdfs:subClassOf*</code> .

```
:User a sh:NodeShape, rdfs:Class ;  
  sh:property [ sh:path :follows ;  
    sh:class :User ] .
```

```
:Manager rdfs:subClassOf :User .  
:alice a :User;  
      :follows :bob .  
:bob   a :Manager ;  
      :follows :alice .  
:carol a :User; :follows :alice,  
                      :dave .  
:dave  a :Employee .
```



Phân loại giá trị

Ràng buộc

Mô tả

nodeKind

Các giá trị:

BlankNode, IRI, Literal,

BlankNodeOrIRI, BlankNodeOrLiteral, IRIOrLiteral

```
:User a sh:NodeShape, rdfs:Class ;  
  sh:property [ sh:path      :name ;  
                sh:nodeKind sh:Literal ;  
              ] ;  
  sh:property [ sh:path      :follows ;  
                sh:nodeKind sh:BlankNodeOrIRI ;  
              ] ;  
sh:nodeKind sh:IRI .
```

```
:alice a :User; :name      _:1 ;  
        :follows :bob .  
:bob    a :User; :name      "Robert";  
        :follows [ :name      "Dave" ] .  
:carol  a :User; :name      "Carol" ;  
        :follows "Dave" .  
_:1 a :User .
```



Ràng buộc đối với giá trị

Ràng buộc	Mô tả
hasValue	Kiểm tra nút đang được xử lý có giá trị được cho
in	Liệt kê các giá trị được phép của thuộc tính được cho

```
:User a sh:NodeShape, rdfs:Class ;  
  sh:property [ sh:path      :affiliation ;  
                sh:hasValue :OurCompany ;  
              ] ;  
  sh:property [ sh:path :gender ;  
                sh:in   (:Male :Female)  
              ] .
```

```
:alice a :User; :affiliation :OurCompany ;  
        :gender :Female .  
:bob    a :User; :affiliation :AnotherCompany ;  
        :gender :Male .  
:carol  a :User; :affiliation :OurCompany ;  
        :gender :Unknown .
```



Ràng buộc giá trị với khuôn hình khác

Ràng buộc	Mô tả
node	Tất cả giá trị của thuộc tính được cho phải tương thích với khuôn hình. Đề quy không được hỗ trợ trong phiên bản SHACL hiện tại.

```
:User a sh:NodeShape, rdfs:Class ;  
  sh:property [ sh:path :worksFor ;  
    sh:node :Company ;  
  ] .  
:Company a sh:Shape ;  
  sh:property [ sh:path :name ;  
    sh:datatype xsd:string ;  
  ] .
```

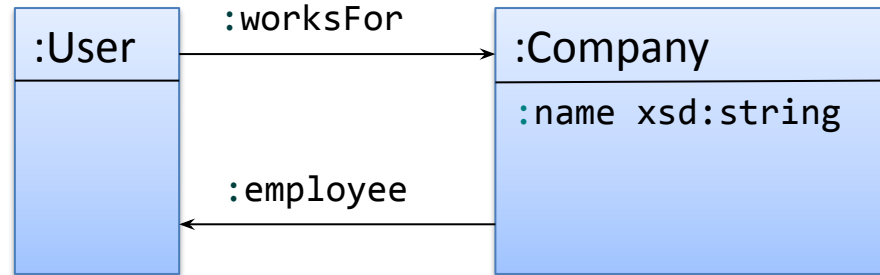
```
:alice a :User; :worksFor :OurCompany .  
:bob    a :User; :worksFor  
    :Another .  
:OurCompany  
    :name "OurCompany" .  
:Another :name 23 .
```



Khuôn hình giá trị và đệ quy

Chúng ta có thể tạo mô hình với chu trình như sau không?

```
:User a sh:NodeShape ;
  sh:property [
    sh:path :worksFor ;
    sh:node :Company ;
  ] .
:Company a sh:Shape ;
  sh:property [
    sh:path :name ;
    sh:datatype xsd:string ;
  ] ;
  sh:property [
    sh:path :employee ;
    sh:node :User ;
  ] .
```



```
:alice :worksFor :OneCompany .
:bob :worksFor :OneCompany .
:carol :worksFor :OneCompany .

:OneCompany :name "One" ;
:OneCompany :employee :alice, :bob, :carol .
```

Quy chuẩn SHACL hiện tại không hỗ trợ

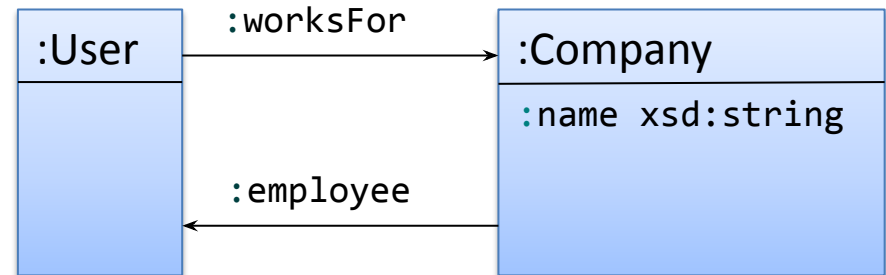
Phụ thuộc vào triển khai 😞

Bài tập 9.1

- Biểu diễn các khuôn hình với đường dẫn thuộc tính, không sử dụng đệ quy.

Thiết lập các ràng buộc:

- + Giá trị của thuộc tính worksFor đối với các phần tử thuộc lớp :User phải thuộc lớp :Company.
- + Đối với các phần tử thuộc lớp :Company thì:
 - + Giá trị của thuộc tính :name phải có kiểu xsd:string.
 - + Giá trị của thuộc tính employee phải thuộc lớp :User.



Bài tập 9.1₍₂₎

@prefix : <http://example.org/> .

@prefix rdfs: <http://www.w3.org/2000/01/rdf-schema#> .

@prefix sh: <http://www.w3.org/ns/shacl#> .

@prefix xsd: <http://www.w3.org/2001/XMLSchema#> .

:User a sh:NodeShape, rdfs:Class ;

sh:property [
sh:path :worksFor;
sh:class :Company
].

:Company a sh:NodeShape, rdfs:Class ;

sh:property [
sh:path :name;
sh:datatype xsd:string
];
sh:property [
sh:path :employee ;
sh:class :User
].

@prefix : <http://example.org/>.

:alice a :User ;
:worksFor :ABC .

:anton a :User ;
:worksFor :Something .

:alexa a :User ;
:worksFor "Pencil ltd".
:maria :worksFor :CMC.

:ABC a :Company ;
:name "ABC corp." ;
:employee :alice, :anton.

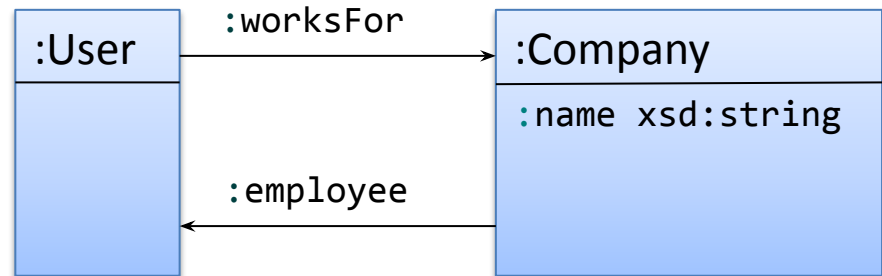
:CMC a :Company ;
:employee :maria.

Bài tập 9.1₍₃₎

- Một giải pháp khả thi: Định kiểu cho tài nguyên sau đó sử dụng ràng buộc lớp (`sh:class`)

```
:User a sh:NodeShape ;
  sh:property [
    sh:path :worksFor ;
    sh:class :Company ;
  ] .

:Company a sh:Shape ;
  sh:property [
    sh:path :name ;
    sh:datatype xsd:string ;
  ] ;
  sh:property [
    sh:path :employee ;
    sh:class :User ;
  ] .
```



```
:alice a :User ;
      :worksFor :OneCompany .
:bob a :User ;
    :worksFor :OneCompany .
:carol a :User ;
      :worksFor :Something .

:OneCompany a :Company ;
      :name "One" ;
      :employee :alice, :bob, :carol .
```



So sánh 2 đồ thị khuôn hình

```
:User a sh:NodeShape ;
  sh:property [
    sh:path :worksFor ;
    sh:node :Company ;
  ] .
:Company a sh:Shape ;
  sh:property [
    sh:path :name ;
    sh:datatype xsd:string ;
  ] ;
sh:property [
  sh:path :employee ;
  sh:node :User ;
] .
```

Phụ thuộc đệ quy

```
:User a sh:NodeShape ;
  sh:property [
    sh:path :worksFor ;
    sh:class :Company ;
  ] .
:Company a sh:Shape ;
  sh:property [
    sh:path :name ;
    sh:datatype xsd:string ;
  ] ;
sh:property [
  sh:path :employee ;
  sh:class :User ;
] .
```

*Phụ thuộc thường -
Không đệ quy.*

Bài tập 9.2

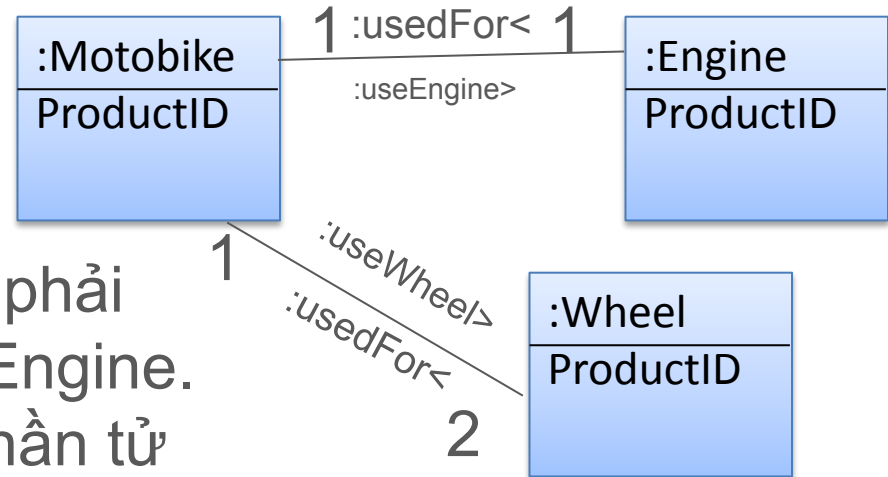
Mô tả các ràng buộc cơ sở trên sơ đồ bằng SHACL?

+ Các phần tử thuộc các lớp

:Motobike, :Engine, :Wheel

đều phải có ProductID.

- + Thuộc tính useEngine cho phần tử thuộc lớp Motobike phải có đúng 1 giá trị thuộc lớp :Engine.
- + Thuộc tính useWheel cho phần tử thuộc lớp Motobike phải có đúng 2 giá trị khác nhau thuộc lớp :Wheel.
- + Thuộc tính usedFor cho phần tử thuộc lớp :Engine phải có đúng 1 giá trị thuộc lớp Motobike.
- + Thuộc tính usedFor cho phần tử thuộc lớp :Wheel phải có đúng 1 giá trị thuộc lớp Motobike.

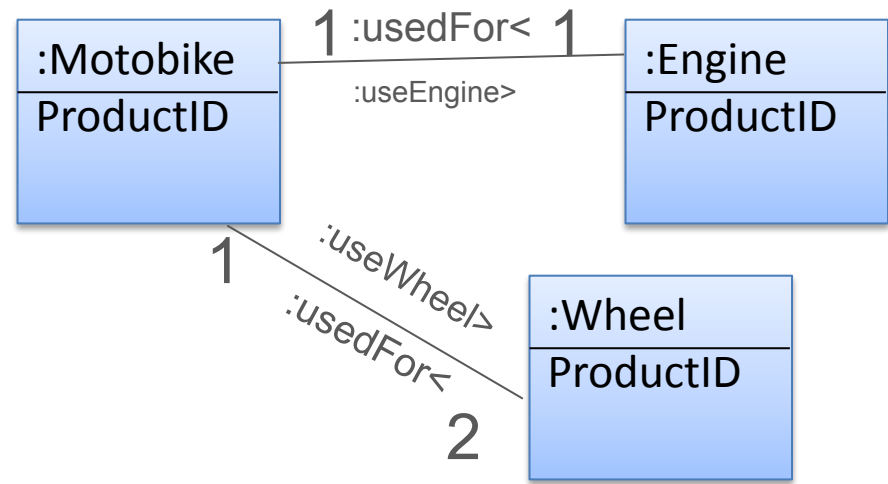


Bài tập 9.2₍₂₎

@prefix : <http://semweb.edu.vn/vehicle#>.

```
:cub1 a :Motobike;  
      :ProductID "11-22-33";  
      :useEngine :engine1;  
      :useWheel :w1, :w2.  
:engine1 :ProductID "21-22-23-55";  
         :usedFor :cub1; a :Engine .  
:w1 :ProductID "w111"; a :Wheel;  
    :usedFor :cub1.  
:w2 :ProductID "w222"; a :Wheel;  
    :usedFor :cub1.
```

```
:engine2 :usedFor :cub1; a :Engine. # Fail ???  
:w3 :usedFor :cub1, :cub2; a :Wheel;  
    :ProductID "W333". #Failed ????  
:cub2 a :Motobike . # Failed ???
```



Các toán tử lô-gic

Ràng buộc	Mô tả
and	Hội của 1 danh sách khuôn hình
or	Tuyển của 1 danh sách khuôn hình
not	Phủ định 1 khuôn hình
xone	Đúng 1 (Tương tự XOR với 2 đối số)

Hội (và/and)

Hội là liên kết mặc định

```
:User a sh:NodeShape ;
  sh:and (
    [ sh:property [
      sh:path      :name;
      sh:minCount  1;
    ]
    [ sh:property [
      sh:path      :affiliation;
      sh:minCount  1;
    ]
  ]
) .
```

=

```
:User a sh:Shape ;
  sh:property [
    sh:path      :name;
    sh:minCount  1;
  ];
  sh:property [
    sh:path      :affiliation;
    sh:minCount  1;
  ]
.
```

Tuyển (hoặc/or)

```
:User a sh:NodeShape ;  
  sh:or (  
    [ sh:property [  
      sh:predicate foaf:name;  
      sh:minCount 1;  
    ]  
    [  
      sh:property [  
        sh:predicate :name;  
        sh:minCount 1;  
      ]  
    ]  
  ) .
```

```
:alice :name "Alice" .  
:bob   foaf:name "Robert" .  
:carol rdfs:label "Carol" .
```



Phủ định (not)

```
:NotFoaf a sh:NodeShape ;  
  sh:not [ a sh:Shape ;  
    sh:property [  
      sh:predicate foaf:name ;  
      sh:minCount 1 ;  
    ] ;  
  ] .
```

```
:alice :name "Alice" . # ???  
:bob   foaf:name "Robert" . # ???  
:carol rdfs:label "Carol" . # ???
```


Phủ định (not)₍₂₎

```
:NotFoaf a sh:NodeShape ;  
  sh:not [ a sh:Shape ;  
    sh:property [  
      sh:predicate foaf:name ;  
      sh:minCount 1 ;  
    ] ;  
  ] .
```

```
:alice :name "Alice" . # Pass  
:bob   foaf:name "Robert" . # Fail  
:carol rdfs:label "Carol" . # Pass
```

Đúng 1 (xone)

```
:UserShape a sh:NodeShape ;  
  sh:targetClass :User ;  
  sh:xone (  
    [ sh:property [  
      sh:path      foaf:name;  
      sh:minCount 1;  
    ]  
    ]  
    [ sh:property [  
      sh:path      :name;  
      sh:minCount 1;  
    ]  
  ]  
) .
```

```
:alice a :User ;                # ???  
      :name "Alice" .  
  
:bob   a :User ;                # ???  
      foaf:name "Robert" .  
  
:carol a :User ;                # ???  
      foaf:name "Carol";  
      :name "Carol" .  
  
:dave  a :User ;                # ???  
      rdfs:label "Dave" .
```

Đúng 1 (xone)₍₂₎

```
:UserShape a sh:NodeShape ;  
  sh:targetClass :User ;  
  sh:xone (  
    [ sh:property [  
      sh:path      foaf:name;  
      sh:minCount 1;  
    ]  
    ]  
    [ sh:property [  
      sh:path      :name;  
      sh:minCount 1;  
    ]  
  ]  
) .
```

```
:alice a :User ;                # Pass  
      :name "Alice" .  
  
:bob   a :User ;                # Pass  
      foaf:name "Robert" .  
  
:carol a :User ;                # Fail  
      foaf:name "Carol";  
      :name "Carol" .  
  
:dave  a :User ;                #Fail  
      rdfs:label "Dave" .
```

Bài tập 9.3

Mẫu IF-THEN: IF X Then Y = (NOT X) OR Y

Tạo hình trạng kiểm tra: Tất cả các sản phẩm đều có đúng 1 productID, bên cạnh đó nếu sản phẩm thuộc lớp Vehicle thì các thuộc tính vehicleEngine và fuelType phải được cung cấp.

```
:p1 a :Book;           # Tương thích
  :productID    "P1" .

:p2 a :Vehicle ;       # Tương thích
  :productID    "P2" ;
  :fuelType     "Gasoline" ;
  :vehicleEngine "X2" .

:p3 a :Vehicle ;       # Không tương thích
  :productID    "P3" .
```

Bài tập 9.3₍₂₎

Mẫu IF-THEN: IF X Then Y = (NOT X) OR Y

Tạo hình trạng kiểm tra: Tất cả các sản phẩm đều có đúng 1 productID, bên cạnh đó nếu sản phẩm thuộc lớp Vehicle thì các thuộc tính vehicleEngine và fuelType phải được cung cấp.

```
:p1 a :Book;      # Tương thích
  :productID      "P1" .
:p2 a :Vehicle ;  # Tương thích
  :productID      "P2" ; :fuelType      "Gasoline" ; :vehicleEngine "X2" .

:p3 a :Vehicle ;  # Không
  :productID      "P3" .
```

```
@prefix :      <http://semweb.edu.vn/people#> .
@prefix sh:    <http://www.w3.org/ns/shacl#> .
@prefix xsd:   <http://www.w3.org/2001/XMLSchema#> .
:ProductShape a sh:NodeShape ;
  sh:targetNode :p1, :p2, :p3, :p4, :p5, :p6 ;
  sh:property [ sh:path :productID ; sh:minCount 1; sh:maxCount 1 ] ;
  sh:or ([sh:not [sh:class :Vehicle]]
        [sh:property [sh:path :fuelType; sh:minCount 1],
                  [sh:path :vehicleEngine; sh:minCount 1]]).
```

Bài tập 9.4

Mẫu IF-THEN-ELSE

Tất cả sản phẩm đều phải có :productID và, nếu 1 sản phẩm có kiểu :Vehicle thì nó phải có thuộc tính :vehicleEngine và :fuelType ngược lại nó phải có thuộc tính :category với giá trị là chuỗi.

Khoảng giá trị

Ràng buộc	Mô tả
minInclusive	\leq
maxInclusive	\geq
minExclusive	$<$
maxExclusive	$>$

```
:Rating a sh:NodeShape ;
  sh:property [
    sh:path
  :ratingValue ;
    sh:minInclusive 1 ;
    sh:maxInclusive 5 ;
    sh:datatype xsd:integer
  ] .
```

```
:bad      :ratingValue 1 .
:average  :ratingValue 3 .
:veryGood :ratingValue 5 .
:zero     :ratingValue 0 .
```



Các ràng buộc dựa trên chuỗi

Ràng buộc	Mô tả
minLength	Giới hạn độ dài tối thiểu trên giá trị chuỗi của nút
maxLength	Giới hạn độ dài tối đa trên giá trị chuỗi của nút
pattern	Kiểm tra giá trị chuỗi khớp với biểu thức chính quy

Độ dài tối thiểu/tối đa

- Kiểm tra biểu diễn chuỗi của giá trị:
 - Không áp dụng được cho các nút rỗng
 - Nếu minLength = 0, không có giới hạn đối với độ dài chuỗi.

```
:User a sh:NodeShape ;  
  sh:property [  
    sh:path      :name ;  
    sh:minLength 4 ;  
    sh:maxLength 10 ;  
  ] .
```

```
:alice :name "Alice" .  
  
:bob :name "Bob" .  
  
:carol :name :Carol .  
  
:strange :name _:strange .
```



?



Mẫu (pattern)

- Kiểm tra xem giá trị có khớp với biểu thức chính quy
- Có thể kết hợp với sh:flags

```
:Product a sh:NodeShape ;  
sh:property [  
  sh:path      :productID ;  
  sh:pattern    "^P\\d{3,4}" ;  
  sh:flags      "i" ;  
] .
```

```
:car      :productID "P2345" .  
:bus      :productID "p567" .  
:truck    :productID "P12" .  
:bike     :productID "B123" .
```



Ràng buộc dựa trên ngôn ngữ

Ràng buộc	Mô tả
languageIn	Khai báo các ngôn ngữ được phép của hằng giá trị
uniqueLang	Quy định không có cặp nút nào có cùng thẻ ngôn ngữ

languageIn

- Giới hạn các thẻ ngôn ngữ có thể được sử dụng cho hằng giá trị

```
:ProductShape a sh:NodeShape;  
  sh:targetClass :Product ;  
  sh:property [ sh:path      rdfs:label ;  
                sh:languageIn ("es" "en" "fr")  
  ] .
```

```
:p234 a :Product ;  
  rdfs:label "jamón"@es, "ham"@en .  
:p235 a :Product ;  
  rdfs:label "milk"@en .  
:p236 a :Product ;  
  rdfs:label "Käse"@de .  
  
:p237 a :Product ;  
  rdfs:label "patatas"@es ,  
            "kartofeln"@de .
```



uniqueLang

- Đảm bảo không cặp bộ-3 nào của cùng 1 chủ thể có cùng thẻ ngôn ngữ

```
:CountryShape a sh:NodeShape ;  
  sh:targetClass :Country ;  
  sh:property [  
    sh:path      skos:prefLabel ;  
    sh:uniqueLang true  
  ] .
```

```
:spain a :Country;  
  skos:prefLabel "Spain"@en,  
                 "España"@es .  
:france a :Country;  
  skos:prefLabel "France",  
                 "France"@en,  
                 "Francia"@es .  
:italy a :Country .  
:usa a :Country;  
  skos:prefLabel "USA"@en,  
                 "United States"@en.
```



Bài tập 9.5

- Nút phải có đúng 1 giá trị chuỗi tiếng anh và 1 giá trị chuỗi tiếng tây ban nha cho thuộc tính skos:prefLabel.

```
:spain a :Country;  
      skos:prefLabel "Spain"@en,  
                     "España"@es .
```

```
:france a :Country;  
        skos:prefLabel "France",  
                       "Francia"@es .
```



```
:italy a :Country .
```



```
:usa a :Country;  
     skos:prefLabel "USA"@en,  
                   "United States"@en.
```



Ràng buộc cặp thuộc tính

Ràng buộc	Mô tả
equals	Tập giá trị của 2 thuộc tính được cho đối với nút được kiểm tra phải bằng nhau
disjoint	Tập giá trị của 2 thuộc tính được cho đối với nút được kiểm tra phải khác nhau
lessThan	Các giá trị phải nhỏ hơn giá trị của thuộc tính kia
lessThanOrEquals	Các giá trị phải nhỏ hơn hoặc bằng giá trị của thuộc tính kia

```
:User a sh:NodeShape ;  
  sh:property [ sh:path :givenName ; sh>equals foaf:firstName ] ;  
  sh:property [ sh:path :givenName ; sh:disjoint :lastName ] .
```

```
:alice :givenName "Alice"; :lastName "Cooper";  
      foaf:firstName "Alice" .  
:bob   :givenName "Bob"; :lastName "Smith" ;  
      foaf:firstName "Robert" .  
:carol :givenName "Carol"; :lastName "Carol" ;  
      foaf:firstName "Carol" .
```



Các khuôn hình đóng

Ràng buộc	Mô tả
closed	Tài nguyên hợp lệ chỉ được có các thuộc tính xuất hiện trong <code>sh:property</code>
ignoredProperties	Danh sách thuộc tính được phép sử dụng nhưng không bắt buộc.

```
:User a sh:NodeShape ;  
  sh:closed true ;  
  sh:ignoredProperties ( rdf:type ) ;  
  sh:property [ sh:path :givenName ; ] ;  
  sh:property [ sh:path :lastName ; ] .
```

```
:alice :givenName "Alice";  
       :lastName "Cooper" .  
  
:bob   a :Employee ; :givenName "Bob"; :lastName "Smith" .  
  
:carol :givenName "Carol";  
       :lastName "King" ;  
       rdfs:label "Carol" .
```



Khuôn hình giá trị đã được xác nhận

- Vấn đề với thuộc tính lặp
- Ví dụ: Sách có 2 mã định danh (isbn và mã nội bộ)

```
:Book a sh:NodeShape ;  
  sh:property [  
    sh:path      :productID ;  
    sh:minCount  1;  
    sh:datatype  xsd:string ;  
    sh:pattern    "^isbn"  
  ];  
  sh:property [  
    sh:path      :productID ;  
    sh:minCount  1;  
    sh:datatype  xsd:string ;  
    sh:pattern    "^code"  
  ] .
```

```
:b1 :productID  
  "isbn:123-456-789" ;  
  :productID "code234" .
```

*!Hợp lệ nhưng kết quả kiểm tra sẽ
là không hợp lệ!*

Khuôn hình giá trị đã được xác thực

- Khuôn hình giá trị đã được xác định kiểm duyệt số lượng xác định các giá trị của 1 thuộc tính được cho tương ứng với 1 khuôn hình được cho.

```
:Book a sh:NodeShape;  
sh:property [  
  sh:path :productID ; sh:minCount 2; sh:maxCount 2; ];  
sh:property [ sh:path :productID ;  
  sh:qualifiedMinCount 1 ;  
  sh:qualifiedValueShape [  
    sh:pattern "^isbn"  
  ]];  
sh:property [ sh:path :productID ;  
  sh:qualifiedMinCount 1 ;  
  sh:qualifiedValueShape [  
    sh:pattern "^code"  
  ] ] .
```

```
:b1 :productID "isbn:123-456-789" ;  
    :productID "code234" .
```

Các ràng buộc phi hành vi

- Có thể hữu ích để tạo ghi chú cho các khuôn hình, sử dụng cho các giao diện đồ họa

Ràng buộc	Mô tả
name	Cung cấp nhãn cho người đọc
description	Cung cấp mô tả cho 1 thuộc tính
order	Thứ tự tương đối của thuộc tính
group	Gom nhiều ràng buộc thành nhóm

```
:User a sh:NodeShape ;
  sh:property [ sh:path :url ; sh:name "URL";
    sh:description "User URL";
    sh:order 1
  ];
  sh:property [ sh:path :name ; sh:name "Name";
    sh:description "User name";
    sh:order 2
  ] .
```

Các ràng buộc phi hành vi₍₂₎

```
:User a sh:NodeShape ;
  sh:property [ sh:path :url ;
    sh:name "URL"; sh:group :userDetails
  ];
  sh:property [ sh:path :name ;
    sh:name "Name"; sh:group :userDetails
  ];
  sh:property [ sh:path :address ;
    sh:name "Address"; sh:group :location
  ];
  sh:property [ sh:path :country ;
    sh:name "Country"; sh:group :location
  ] .
```

```
:userDetails a sh:PropertyGroup ;
  sh:order 0; rdfs:label "User details".

:location a sh:PropertyGroup ;
  sh:order 1 ; rdfs:label "Location" .
```

Công cụ phần mềm có thể sinh biểu mẫu, ví dụ:

User details

URL:

Name:

Location

Address:

Country:

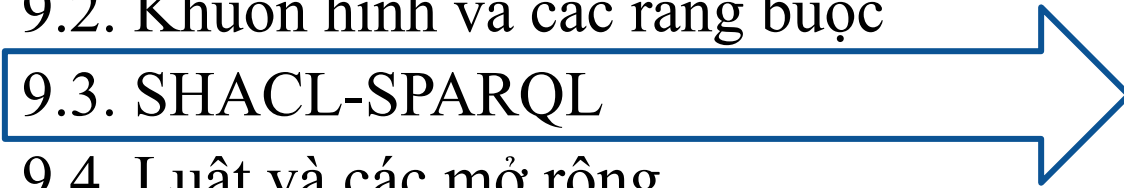
Nội dung

9.1. Các thành phần cơ bản

9.2. Khuôn hình và các ràng buộc

9.3. SHACL-SPARQL

9.4. Luật và các mở rộng



Các ràng buộc SPARQL

- Báo cáo vi phạm sẽ được tạo trong trường hợp truy vấn SPARQL trả về lỗi kiểm tra.
- Ràng buộc SPARQL có kiểu sh:SPARQLConstraint

Ràng buộc	Mô tả
message	Thông điệp trong trường hợp lỗi
sparql	Truy vấn SPARQL được thực hiện
prefixes	Mô tả các khai báo tiền tố không gian tên với sh:declare Mỗi nút có: sh:prefix: Tiền tố sh:namespace: IRI của không gian tên

Các ràng buộc SPARQL₍₂₎

- Bộ xử lý SHACL-SPARQL thiết lập giá trị cho các biến đặc biệt

Ràng buộc	Mô tả
<code>\$this</code>	Nút được kiểm tra
<code>\$shapesGraph</code>	Có thể được sử dụng để truy cập đồ thị khuôn hình, ví dụ: <code>GRAPH \$shapesGraph { ... }</code>
<code>\$currentShape</code>	Khuôn hình hiện tại

Các ràng buộc SPARQL₍₃₎

- Ánh xạ giữa dòng kết quả và thông tin lỗi kiểm tra.

Thuộc tính	Truy vấn
sh:focusNode	Giá trị của biến <code>\$this</code>
sh:resultPath	Biến <code>\$path</code>
sh:value	Giá trị của biến <code>value</code>
sh:resultMessage	Giá trị của biến <code>?message</code>
sh:sourceConstraint	Ràng buộc đang được áp dụng, ví dụ giá trị của <code>sh:sparql</code>

Các ràng buộc SPARQL₍₄₎

- Ví dụ name phải là kết quả kết hợp của givenName và familyName

```
:UserShape a sh:NodeShape ;
sh:targetClass :User ;
sh:sparql [ a sh:SPARQLConstraint ;
  sh:message ":name must equal :givenName+:familyName";
  sh:prefixes [ sh:declare [ sh:prefix "" ;
    sh:namespace "http://semweb.edu.vn/user#"^^xsd:anyURI;
  ] ] ;
sh:select
  """SELECT $this (:name AS ?path)
    (?name as ?value)

  WHERE {
    $this :name ?name .
    $this :givenName ?givenName .
    $this :familyName ?familyName .
    FILTER (!isLiteral(?value) ||
      !isLiteral(?givenName) || !isLiteral(?familyName) ||
      concat(str(?givenName), ' ', str(?familyName))!=?name )
  }""" ;
] .
```

```
:alice a :User ;
  :givenName "Alice" ;
  :familyName "Cooper" ;
  :name "Alice Cooper" .
:bob a :User ;
  :givenName "Bob" ;
  :familyName "Smith" ;
  :name "Robert Smith" .
```



Các thành phần ràng buộc SPARQL

- SHACL-SPARQL cho phép tự định nghĩa các thành phần ràng buộc. Sau khi được định nghĩa, chúng có thể được sử dụng như các thành phần ràng buộc có sẵn.

```
:ProductShape a sh:NodeShape ;  
  sh:targetClass :Product ;  
  sh:property [  
    sh:path      :color ;  
    :size        3 ;  
    sh:minCount  1 ;  
  ] .
```

```
:c1 a :Product; :color (255 0 255) .
```

```
:c2 a :Product; :color (255 0 210 345) .
```



```
:c3 a :Product; :color (255 0) .
```



Các thành phần ràng buộc SPARQL₍₂₎

```
:FixedListConstraintComponent a sh:ConstraintComponent ;
sh:parameter [ sh:path          :size ;
               sh:name          "Size of list" ;
               sh:description   "The size of the list" ;
             ] ;
sh:labelTemplate "Size of values: \"{$size}\"" ;
sh:propertyValidator :fixedLengthValidator .
```

```
:fixedLengthValidator a sh:SPARQLSelectValidator ;
sh:message "{ $PATH } must have length { ?size }, not { ?count }" ;
sh:prefixes [ sh:declare [ sh:prefix "rdf" ;
                           sh:namespace "http://www.w3.org/1999/02/22-rdf-syntax-ns#"] ] ;
sh:select """SELECT $this ?value $count WHERE {
    $this $PATH ?value .
    { SELECT $this ?value (COUNT(?member) AS ?count) $size WHERE {
        ?value rdf:rest*/rdf:first ?member
      } GROUP BY $this ?value $size }
    FILTER (!isBlank(?value) || ?count != $size) }""" .
```

Có 2 kiểu ràng buộc:

SPARQLSelectValidator và SPARQLASKValidator

Các thành phần ràng buộc SPARQL₍₃₎

Thuộc tính	Mô tả
sh:parameter	Mô tả tham số của thành phần ràng buộc. Các giá trị là các lớp con của khuôn hình thuộc tính. sh:path Mô tả tên tham số. sh:optional khai báo tính chất bắt buộc của tham số.
sh:labelTemplate	Mô tả ràng buộc, có thể nhắc đến các tham số bằng biến: \$varName
sh:nodeValidator	Gắn kết với 1 bộ xử lý khuôn hình nút.
sh:propertyValidator	Gắn kết với 1 bộ xử lý khuôn hình thuộc tính.

Bài tập 9.6

@prefix : <http://semweb.edu.vn/family#> .

```
:nva :ho "Nguyen";  
      :ten "Van A".  
:btc :ho "Bui";  
      :ten "Thi C".  
:nvd :coBo :nva; :coMe :btc;  
      :ho "Nguyen";  
      :ten "Van D".  
:nth :coBo :nva; :coMe :btc;  
      :ho "Tran";  
      :ten "Thi H".
```

Thử thiết lập ràng buộc họ của 1 người phải giống với họ của bố hoặc mẹ với 2 nút :nvd, :nth?

Bài tập 9.6₍₂₎

@prefix : <http://semweb.edu.vn/family#> .

@prefix sh: <http://www.w3.org/ns/shacl#> .

```
:PersonShape a sh:NodeShape ;
  sh:targetNode :nvd, :nth;
  sh:sparql [ a sh:SPARQLConstraint ;
    sh:message "Họ phải giống họ của bố hoặc mẹ";
    sh:prefixes [ sh:declare [ sh:prefix "" ;
      sh:namespace "http://semweb.edu.vn/family#";
    ] ] ;
    sh:select
      """SELECT *
      WHERE {
        $this :ho ?ho ; :coBo ?bo; :coMe ?me.
        ?bo :ho ?ho_bo.
        ?me :ho ?ho_me.
        FILTER (?ho != ?ho_bo && ?ho != ?ho_me)
      }""" ;
  ].
```

Các hạn chế cần khắc phục??

