

DH ・ 技術要素 ・ SHACL / ShEx

SHACL入門

RDFの「形」を検証する

DH入門 / 技術要素シリーズ

中村

※実験的な取り組みです（構成・図・AI音声合成を含む）。内容をご確認・ご注意のうえご利用ください

この動画について

- ✓ **クリエイティブ・コモンズ**方針のもと、独自に構成した解説です（今回は仕様の事実確認に基づく新規作成）
- ✓ スライド・図は新規作成、ナレーションは**AI音声合成**（この回は本人クローン声ではない合成音声）
- ✓ これは**実験的な取り組み**です。内容は**ご確認・ご注意のうえ**ご利用ください
- ✓ 誤りに気づいたら概要欄からご指摘ください。出典・ライセンスは末尾と概要欄に記載しています

この回のゴール

RDFが「期待した形」になっているかを検証する考え方をつかむ

- ✓ **SHACL** が、RDFを**シェイプ（期待する形）** で検証する仕組みだと説明できる
- ✓ **ノード/プロパティシェイプ**と主な制約（個数・型・クラス）を説明できる
- ✓ 検証が**適合/違反のレポート**を返すことを説明できる
- ✓ 別の道 **ShEx** と、DHでの使いどころを知る

RDF・SPARQL入門を見ているとつながりますが、必須ではありません。

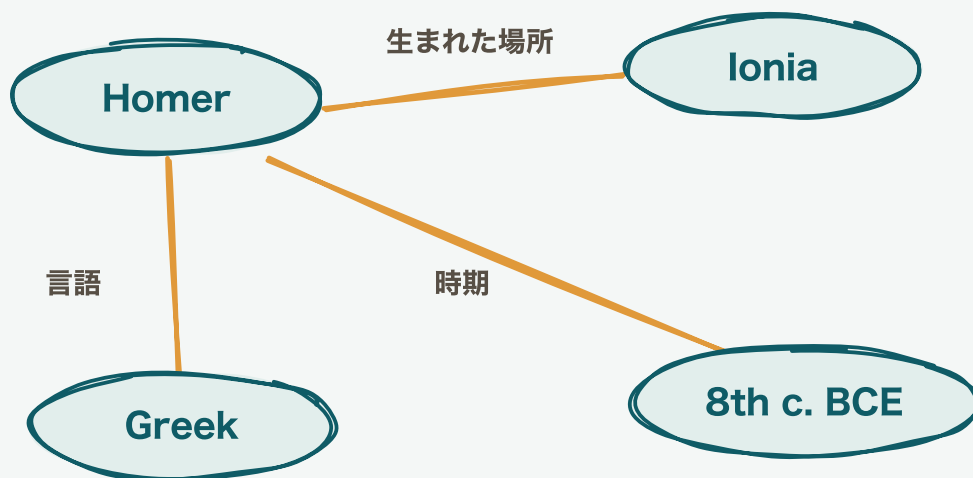
今日の流れ

- ✓ なぜ「検証」が要るのか（形のバラつき）
- ✓ シェイプで検証する — SHACL
- ✓ もう一つの道 — ShEx
- ✓ 位置づけ（問い合わせ・XMLスキーマとの関係）
- ✓ 使いどころ・始め方

なぜ「検証」が要るのか

まず、データの形のバラつきから

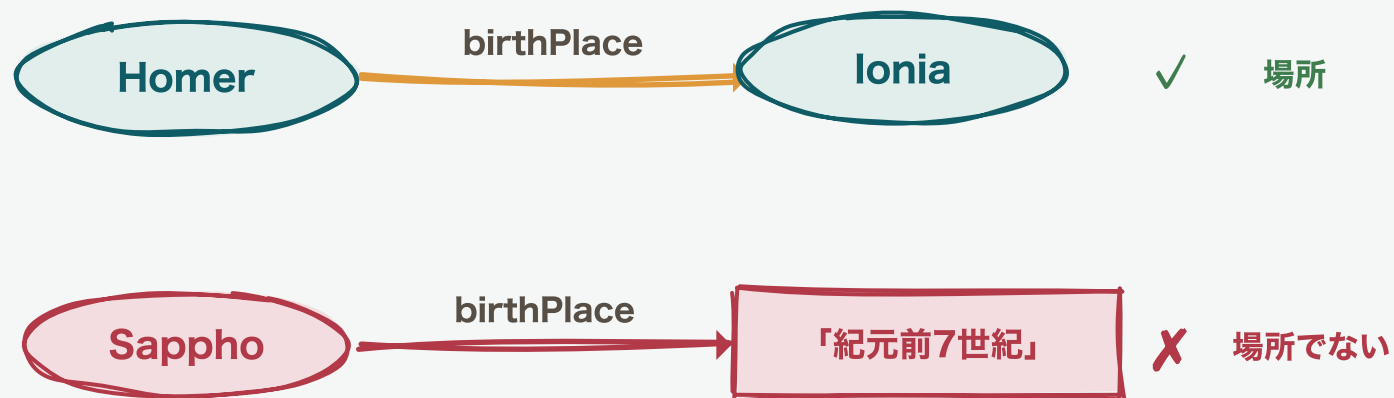
RDF=三つ組が重なった「網」



三つ組が重なってできた「網」(RDFグラフ)

RDFは、主語・述語・目的語の**三つ組**を重ねたグラフ。まずはここが出発点

データの「形」はバラつく



同じ Person でも、生地の有無や型がバラつくことがある

必須のはずの項目が**無い**、場所のはずが**日付**…。大量データでは気づきにくい

「期待する形」を決めて照らしたい

- ✓ 人手で一件ずつ確かめるのは、大量データでは**無理**
- ✓ そこで、データに期待する**形（ルール）**をあらかじめ決めておく
- ✓ その形に**合っているかを機械的に照合**したい
- ✓ それを担うのが **SHACL**（シャクル） — シェイプによる検証

シェイプで検証する — SHACL

シェイプ=期待する「形」の宣言

PersonShape = Person に期待する「形」

対象 : Person = ノードシェイプ

生まれた場所 (birthPlace) = プロパティシェイプ

- 1つ以上ある (minCount 1)
- 場所 (Place) であること (class)

データに期待する「形」を、シェイプとして宣言する

対象を決める **ノードシェイプ**、各項目を縛る **プロパティシェイプ** で書く

書き方 (SHACL/Turtle)

```
ex:PersonShape a sh:NodeShape ;  
  sh:targetClass ex:Person ;  
  sh:property [  
    sh:path ex:birthPlace ; sh:minCount 1 ; sh:class ex:Place ;  
  ] .
```

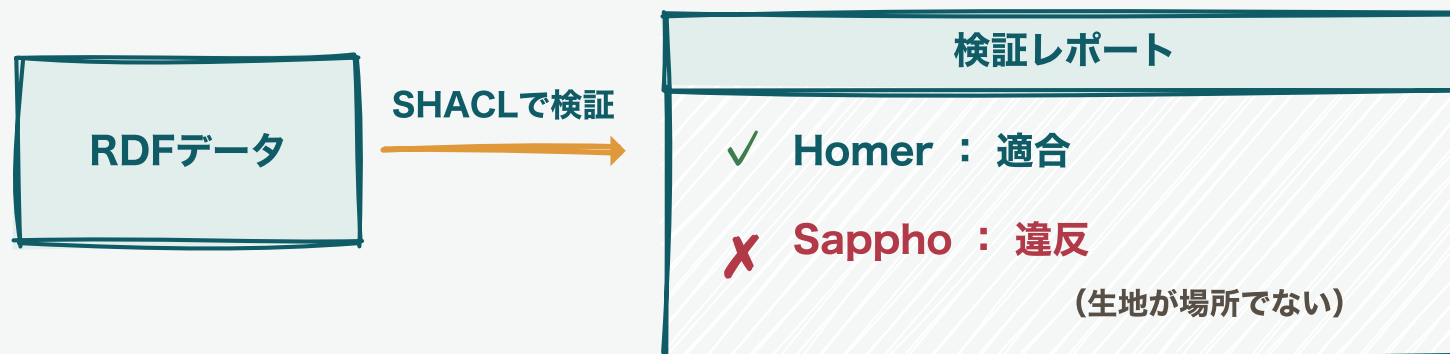
「Person は **生地を1つ以上**持ち、それは **場所**であること」を宣言

いろいろな制約を組み合わせる

- ✓ **個数** : minCount / maxCount (必須・上限)
- ✓ **型** : datatype (文字列・日付・数 など)
- ✓ **クラス** : class (「場所」であること など)
- ✓ **値域・書式** : 範囲 (minInclusive 等) や pattern (正規表現)

これらを積み重ねて、データに期待する「形」を細かく表現できる

検証すると「レポート」が返る



検証すると、適合/違反を「理由つき」で返す

適合か違反か、違反なら**どのノードが・なぜ**かを、機械可読のレポートで返す

ここまでのポイント

- ✓ **SHACL** は、RDFを**シェイプ（期待する形）** で検証する
- ✓ **ノードシェイプ**で対象を、**プロパティシェイプ**で各項目を縛る
- ✓ 個数・型・クラスなどの**制約**を組み合わせる
- ✓ 検証は**適合／違反**を理由つきのレポートで返す

SHACL は W3C の標準。もう一つ、よく使われるシェイプ言語が ShEx

もう一つの道 — ShEx

ShEx = 別の書き味のシェイプ言語

```
ex:PersonShape {  
  ex:birthPlace @ex:PlaceShape + ;  
}
```

チェックス。同じ「形」を、文法のように簡潔に書く流儀（+ = 1つ以上）

例：Wikidata の EntitySchemas

- ✓ **ウィキデータ**では、ShEx で「このクラスはこういう形」という**スキーマ**を共有
- ✓ これを **エンティティスキーマ** (EntitySchemas) と呼ぶ
- ✓ 「人物なら 生年・職業 を持つ」のような**期待形をみんなで合意**できる
- ✓ コミュニティでデータの質をそろえる、実践的な使い方

SHACL と ShEx の使い分け

- ✓ どちらも「**シェイプで形を検証する**」という目的は同じ
- ✓ **SHACL** : W3C 標準。RDFそのもので書け、ツールや拡張が豊富
- ✓ **ShEx** : 文法的で簡潔。Wikidata など**コミュニティでの採用**が厚い
- ✓ 迷ったら、まわりの**エコシステムに合わせて**選べばよい

対立ではなく「同じ問題への二つの道」。考え方が分かれば行き来できます

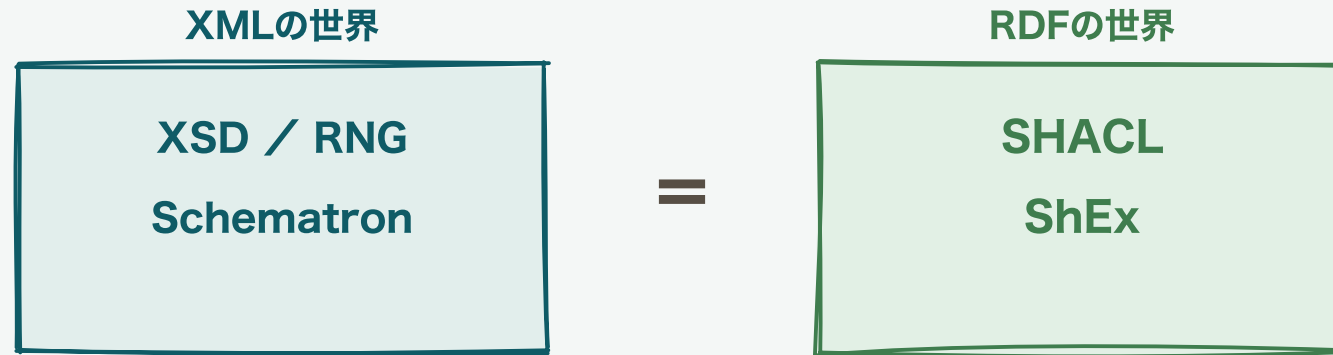
位置づけ・使いどころ

「検証」と「問い合わせ」は別もの

- ✓ **SPARQL** は、条件に合うデータを**取り出す**（問い合わせ）
- ✓ **SHACL** は、データが期待した**形か判定する**（検証）
- ✓ 目的が違う：**取り出す** / **形を保証する**
- ✓ なお SHACL は内部で SPARQL を使う書き方（SHACL-SPARQL）もある

RDFを「書く→問う→検証する」。SHACLはその最後の一手です

XMLのスキーマに当たる層



XMLにスキーマがあるように、RDFには「シェイプ言語」がある

XMLに**スキーマ**があるように、RDFには**シェイプ言語**がある、と捉えると掴みやすい

考えてみよう

あなたのデータに課したい「形（ルール）」は、何ですか？

「この項目は必須」「ここは日付」… ここで少し、動画を止めて考えてみてください。

ここまでのポイント

- ✓ **ShEx** も、同じ「シェイプで検証」を別の書き味で実現する
- ✓ SHACL は標準・拡張が豊富、ShEx はコミュニティ採用が厚い
- ✓ 検証 (SHACL) は、問い合わせ (SPARQL) とは**目的が違う**
- ✓ XMLの**スキーマ**に当たる、RDFの**形の保証**の層

DHでの活用

- ✓ **文化遺産LODの品質保証**：公開前に、データの形をまとめて点検
- ✓ **機関間の取り決め**：シェイプを「データの約束ごと」として共有
- ✓ **Wikidata**：EntitySchemas (ShEx) でクラスの期待形をそろえる
- ✓ **パイプラインに組み込み**、壊れたデータを早期に発見する

おさえておきたい前提

- ✓ RDFは**オープンワールド**前提 = 「書かれていない = 偽」ではない
- ✓ だから「**必須**」などの形は、シェイプで**明示**して初めて検証できる
- ✓ 検証はあくまで「**データに書かれている範囲**」での判定
- ✓ まずは**1つの必須項目**から、シェイプを少しずつ足していく

始め方・学ぶには

- ✓ まず触れる：**SHACL Playground / RDFShape**
(ブラウザで検証を試す)
- ✓ ツール：**pySHACL** (Python)、**Apache Jena** など
- ✓ 仕様：**W3C SHACL 勧告** (一次情報で確認)
- ✓ コツ：**1つの必須プロパティ**のシェイプから、小さく育てる

まとめ

- ✓ **SHACL** は、RDFを**シェイプ（期待する形）** で検証する仕組み
- ✓ **ノード/プロパティシェイプ**に、個数・型・クラスなどの制約を書く
- ✓ 検証は**適合/違反**を理由つきレポートで返す。別の道に **ShEx**
- ✓ RDFを「書く→問う→**検証する**」。データの質をそろえる最後の一手

RDFでつなぎ、SPARQLで引き出し、SHACLで形を保証する。地続きの考え方です

出典・ライセンス

本動画の**スライド・図・ナレーション原稿は CC BY 4.0** で公開します (© 2026 中村 覚)。出典表示のうえ自由に再利用いただけます。

- ✓ 事実確認：W3C 「Shapes Constraint Language (SHACL)」 勧告 (W3C Document License)
- ✓ 事実確認：Validating RDF Data / Labra Gayo ほか (参考。翻案はせず)

CCライセンスの教材が乏しい領域のため、概念のみ参照し図・文は新規作成。掛け合い版の音声・立ち絵は VOICEVOX / 坂本アヒル氏の各規約に従います。

ご清聴ありがとうございました