

DH ・ 技術要素 ・ SPARQL

スパークル入門

グラフに問いを投げる

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**のオープン教材を参照し、独自に構成した解説です
- ✓ スライド・図は新規作成、ナレーションは**本人声のAI音声合成**
- ✓ 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

つないだデータに「問い」を投げて、答えを引き出せるようになる

- ✓ **スパークル**が、RDFのグラフに問い合わせる言語だと説明できる
- ✓ 基本の形 **SELECT ... WHERE** と **?変数**の考え方を説明できる
- ✓ 条件を**重ねて絞り込む**イメージを持てる
- ✓ よく使う句と、公開エンドポイントで**試す**流れを知る

RDF・リンクトデータの回を見ているとつながりますが、必須ではありません。

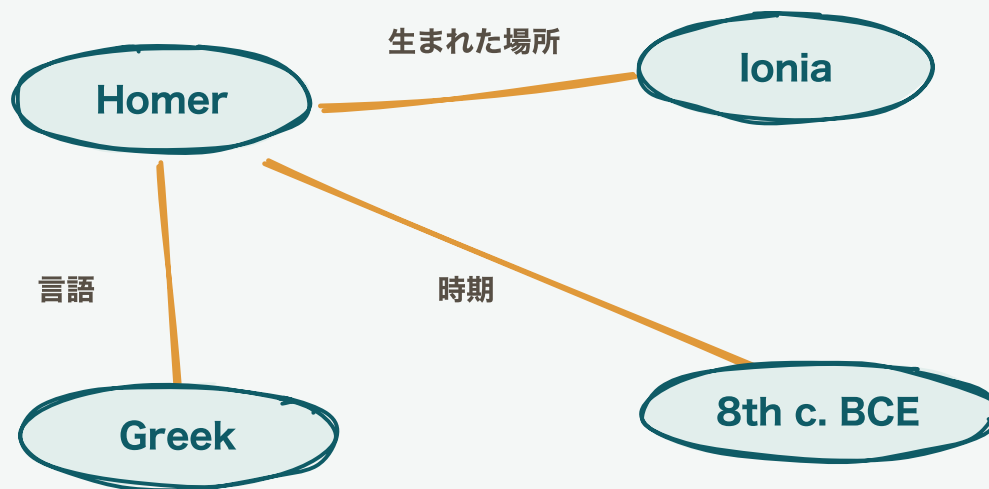
今日の流れ

- ✓ おさらい — RDFという「網」と、問い合わせの必要
- ✓ 基本の形 — トリプルパターン (SELECT ... WHERE)
- ✓ つなげて絞る — パターンを重ねる
- ✓ 結果を整える — FILTER・OPTIONAL・並べ替え
- ✓ 実際に試す・学ぶには

おさらい — RDFという「網」

まずは、何に問い合わせるのかを確認します

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)

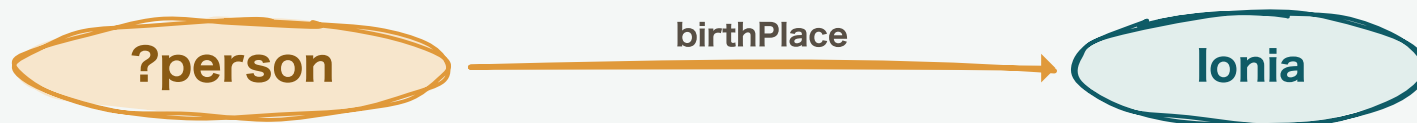
データは、主語・述語・目的語の**三つ組**を重ねた**グラフ (網)** でした

大きな網は「手で探す」のが難しい

- ✓ 網（グラフ）が大きくなると、目で追って答えを探すのは**大変**
- ✓ 「イオニアで生まれた人を、全部あげて」を、手作業では拾いきれない
- ✓ そこで、グラフに**問いを書いて投げる**仕組みが要る
- ✓ それが **スパークル**（SPARQL） — グラフへの問い合わせ言語

基本の形 — トリプルパターン

三つ組の一部を「?変数」にする



主語を ?変数 にした三つ組 = 「トリプルパターン」

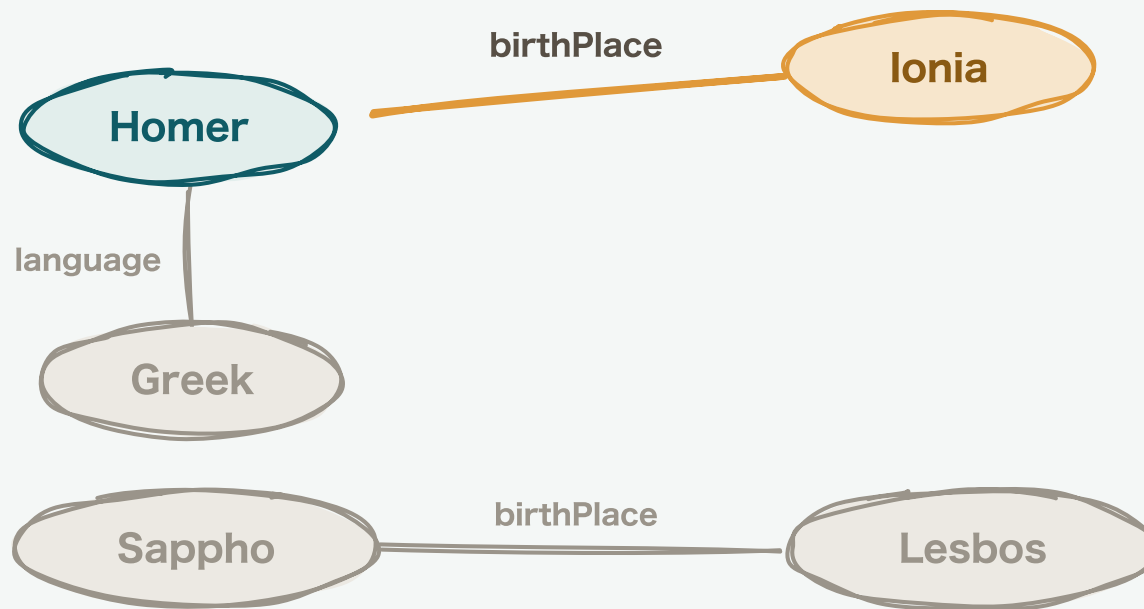
知りたいところを **?変数** にした三つ組を、**トリプルパターン**と呼びます

基本の形：SELECT … WHERE

```
SELECT ?person WHERE {  
  ?person birthPlace Ionia .  
}
```

WHERE に探すパターンを書き、**SELECT** で受け取りたい変数を選ぶ

パターンに「合う三つ組」を集める



「?person birthPlace Ionia」に当てはまるのは Homer

グラフ全体から、パターンに**当てはまる三つ組**を機械的に拾い出す

ここまでのポイント

- ✓ スパークルは、RDFの**グラフに問い合わせる**言語
- ✓ 知りたいところを **?変数** にした三つ組 = **トリプルパターン**
- ✓ **SELECT ... WHERE** で、合う三つ組を**まとめて拾う**

パターンが一つだと、まだ条件はゆるい。 → 重ねて絞り込む

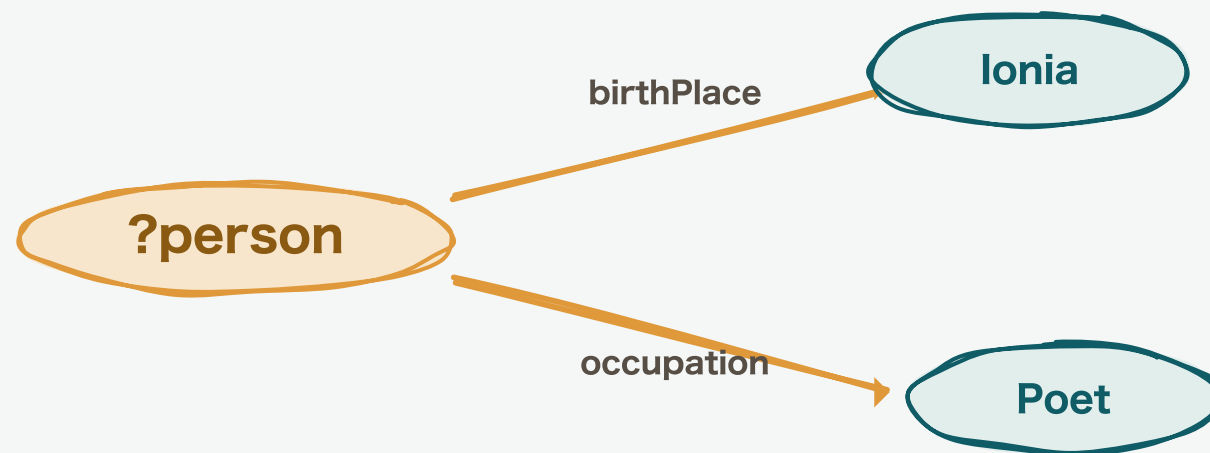
つなげて絞る — パターンを重ねる

パターンを二つ書くと「かつ」になる

```
SELECT ?person WHERE {  
  ?person birthPlace Ionia .  
  ?person occupation Poet .  
}
```

同じ **?person** を二度使うと、「イオニア生まれ **かつ** 詩人」に絞れる

同じ変数が「つなぎ目」になる



同じ `?person` が、両方を満たすものだけが残る

同じ名前の **?変数** でパターンをつなぐ。これが絞り込みの基本

長いURIは PREFIX で短く書く

PREFIX schema: <http://schema.org/>

```
SELECT ?person WHERE {  
  ?person schema:birthPlace ?place .  
}
```

URIの述語は本来とても長い。**PREFIX** で略称を決めて、短く読みやすく

ここまでのポイント

- ✓ パターンを**重ねると**、条件を「かつ」で**絞り込める**
- ✓ 同じ名前の **?変数** が、パターンどうしの**つなぎ目**になる
- ✓ 長いURIは **PREFIX** で短く書ける

ここまでで「探す」はできた。あとは結果を整える句を見ていく

結果を整える — よく使う句

FILTER = 条件でしぼる

```
SELECT ?person ?year WHERE {  
  ?person birthYear ?year .  
  FILTER( ?year < 0 )  
}
```

取り出した値に条件を足す。例として「年代でしぼる」といった使い方ができる

OPTIONAL = あれば取る

```
SELECT ?person ?image WHERE {  
  ?person birthPlace Ionia .  
  OPTIONAL { ?person image ?image . }  
}
```

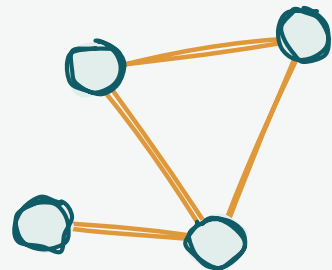
画像が**あれば**一緒に取り、**なくても**その人物を結果から落とさない

ORDER BY・LIMIT=並べる・件数をしぼる

```
SELECT ?person WHERE {  
  ?person birthPlace Ionia .  
}  
ORDER BY ?person LIMIT 10
```

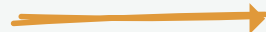
ORDER BY で並べ替え、**LIMIT** で件数を区切る。まず10件だけ、なども簡単

グラフに問うと「表」が返る



RDFグラフ

スパークル



?person
Homer
Mimnermus
Anacreon
...

グラフに問い合わせると、答えは「表」で返る

問い合わせの答えは、**?変数を列にした表**として返ってくる

ここまでのポイント

- ✓ **FILTER** で値に条件を足し、**OPTIONAL** で「あれば取る」
- ✓ **ORDER BY・LIMIT** で、並べ替え・件数の調整ができる
- ✓ 答えは **?変数を列にした表**で返ってくる

部品はそろった。あとは、実際に動く窓口へ問いを投げしてみる

実際に試す — エンドポイント

どこに問いを投げる？

- ✓ スパークルを受け付ける公開の窓口を、**エンドポイント**と呼ぶ
- ✓ データを持つ側が用意し、誰でも問い合わせできるものも多い
- ✓ 入門に向くのは **ウィキデータ・クエリ・サービス**
- ✓ ブラウザに問いを書いて実行し、その場で**結果の表**を見られる

例：Wikidata に問いを投げる

- ✓ ウィキデータでは、ものも述語も **URI** で表されている
- ✓ 「ある場所で生まれた人」「ある職業の人」を、パターンで書いて取り出せる
- ✓ 用意された**例文**を実行し、少しずつ書き換えるのが学びやすい
- ✓ 結果は表のほか、**地図・グラフ**などにも切り替えて見られる

本動画の例（Ionia など）は説明用です。実際の URI・語彙は各データの定義に従います

考えてみよう

あなたのデータに投げたい「問い」は、どんなものですか？

「○○な△△を、全部あげて」… ここで少し、動画を止めて考えてみてください。

問いの立て方にも「判断」が入る

- ✓ どの述語・語彙で書くかで、**拾えるもの**が変わる
- ✓ データに無い関係は、当然ながら答えに出てこない
- ✓ 結果は「データがそう書かれている範囲」での答え、と捉える
- ✓ 広すぎる問いは**重い**。**LIMIT** で小さく試してから広げると安全

始め方・学ぶには

- ✓ まず触れる：**ウィキデータ・クエリ・サービス**の例文を実行してみる
- ✓ 体系的に：**Programming Historian** 「Using SPARQL to access Linked Open Data」
- ✓ 背景を固める：同 「Intro to Linked Open Data」 / **Aalto SeCo** の教材
- ✓ コツ：小さな問いから始め、パターンを**一行ずつ足して**育てる

まとめ

- ✓ スパークルは、RDFの**グラフに問い合わせる**言語
- ✓ **?変数**を入れたトリプルパターンを、**SELECT … WHERE** で書く
- ✓ パターンを重ねて絞り、**FILTER ・ OPTIONAL ・ ORDER BY ・ LIMIT** で整える
- ✓ 答えは**表**で返る。公開エンドポイントで、すぐ試せる

データを「線でつなぐ」RDFから、その網に「問いを投げる」スパークルへ。地続きです

出典・ライセンス

本動画は、以下のオープンライセンス教材を参照して作成しました。

- ✓ Using SPARQL to access Linked Open Data / M. Lincoln (Programming Historian) — CC BY 4.0
- ✓ Introduction to the Principles of Linked Open Data / J. Blaney (Programming Historian) — CC BY 4.0
- ✓ Linked Data Technologies for Cultural Heritage and DH / SeCo, Aalto University — CC BY 4.0

スライド・図は中村による新規作成（概念を参照し、表現は新たに構成）。

ご清聴ありがとうございました