

オントロジー・セマンティクス・データモデル標準化 の基礎調査

古典的な研究の系譜から Open Semantic Interchange (Apache Ossie) まで

dobachi

2026-10-04

目次

1 はじめに :初めて読む人のために	2
1.1 何が問題なのか	2
1.2 四つの言葉をひとことで	3
1.3 なぜ今これが話題になるのか	4
1.4 歴史は四本の流れに分けられる	5
1.5 この報告書の読み方	5
2 エグゼクティブサマリ	5
3 調査の範囲	6
4 用語の整理	6
4.1 この報告書で使う整理の軸	6
4.2 出典による定義	8
4.3 先行する整理 :意味の指定の強さによる連続体	9
4.4 考察 :意味の 4 段階	10
5 歴史の系譜	14
5.1 四本の流れ	14
5.2 年表	14
5.3 知識表現からセマンティックウェブへ	17
5.4 意味の 4 段階と四本の流れの関係	18
6 技術要素の解説	19
6.1 データベース系	19
6.2 知識表現系	22
6.3 Web 標準系	25
6.4 分析系	35
7 データモデルとメタデータの標準化	38
8 ドメインオントロジーと業界標準	39
9 米国企業の近年の取り組み	41
9.1 知識グラフ	42
9.2 製品としてのオントロジー : Palantir	42
9.3 分析系のセマンティックレイヤ	44

9.4	Open Semantic Interchange から Apache Ossie へ	48
9.5	言語モデルに文脈を渡す形式：Open Knowledge Format	49
10	データスペースとの接点	52
10.1	国際：データスペースプロトコルとその標準化	52
10.2	欧州	54
10.3	日本	56
10.4	考察：データスペースプロトコルと ODS の違い	58
11	考察：体系化の試み	59
12	確認できなかった点と留意事項	64
13	付録 A：用語集	64
	参考文献	65
14	付録 B：改変履歴	73

i この文書について

オントロジー、セマンティクス、データモデル標準化という語が何を指すのかを整理した調査報告である。事実の記述には出典を付けた。筆者の解釈は「考察」と明示した節と段落に分けて書いた。

この文書は 2026-10-04.5 版である。本文と図は CC BY 4.0 で公開している([ライセンス](#))。

1 はじめに：初めて読む人のために

オントロジーやデータモデルに初めて触れる読者のために、専門用語をできるだけ使わずに、問題の全体像を説明する。内容に詳しい読者はセクション 2 から読んでよい。

1.1 何が問題なのか

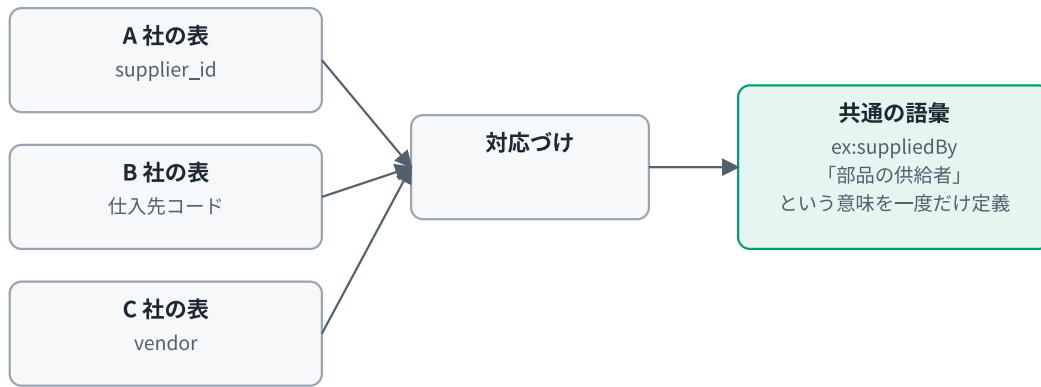
ある部品を、どの会社から仕入れているか。これを記録する表は、どの会社にもある。ところが、その列の名前は会社ごとに違う。A 社は `supplier_id`、B 社は「仕入先コード」、C 社は `vendor` と呼んでいるかもしれない。

人が見れば、三つとも同じものを指していそうだと見当がつく。ただし、名前だけでは断定できない。`vendor` には、部品の仕入先のほかに、保守や物流の委託先が含まれているかもしれない。人は、担当者に尋ねたり定義書を読んだりして確かめる。

最近の AI も、列の名前から同じものだと推測できる。しかし人の見当と同じく、それは推測である。そのうえ、同じデータを渡しても毎回同じ答えになるとは限らず、外れたときにそれと気づく手がかりも乏しい、と筆者は考える。Apache Ossie のプロジェクトも、当事者の立場から、人の分析者や AI エージェントがどの定義が正しいかを推測しなければならない状況は望ましくないと述べている [1]。業務で三社のデータを突き合わせるには、どの列とどの列が同じものを指すのかを、あとから確かめられる形で決めておく必要がある。

逆の問題もある。同じ名前で、違うものを指す場合である。

二つの会社の部品表に、どちらも「重量」という列があるとする。一方はグラム、もう一方はキログラムで記録しているかもしれない。梱包を含む重さか、部品そのものの重さかも、名前だけでは分からない。「価格」が税込か税抜か、「日付」が発注日か納品日も同じである。名前がそろっていても、意味がそろっていないとは限らない。



列の名前は会社ごとに違っても、指しているものが同じだと機械に分かるようにする。

図 1: 同じものを別の名前で呼ぶ問題と、共通の語彙による対応づけ。列の名前と例は筆者が作成した。

1.2 四つの言葉をひとことで

この報告書の題名にある三つの言葉と、その前提になる「データモデル」を、まず大づかみに押さえておく。正確な定義はセクション 4 にある。

四つの言葉の関係を、「意味の三角形」と呼ばれる古典的な図に重ねて示す。Ogden と Richards は 1923 年の著書で、記号(Symbol)、思考または指示(Thought or Reference)、指示対象(Referent)の三つを三角形の頂点に置いた。記号と指示対象は直接にはつながらず、思考を介してつながるという [2]。この報告書では、分かりやすさのために、頂点を「記号」「概念」「対象」と呼び替える。原典は「概念」という語を使っていない。

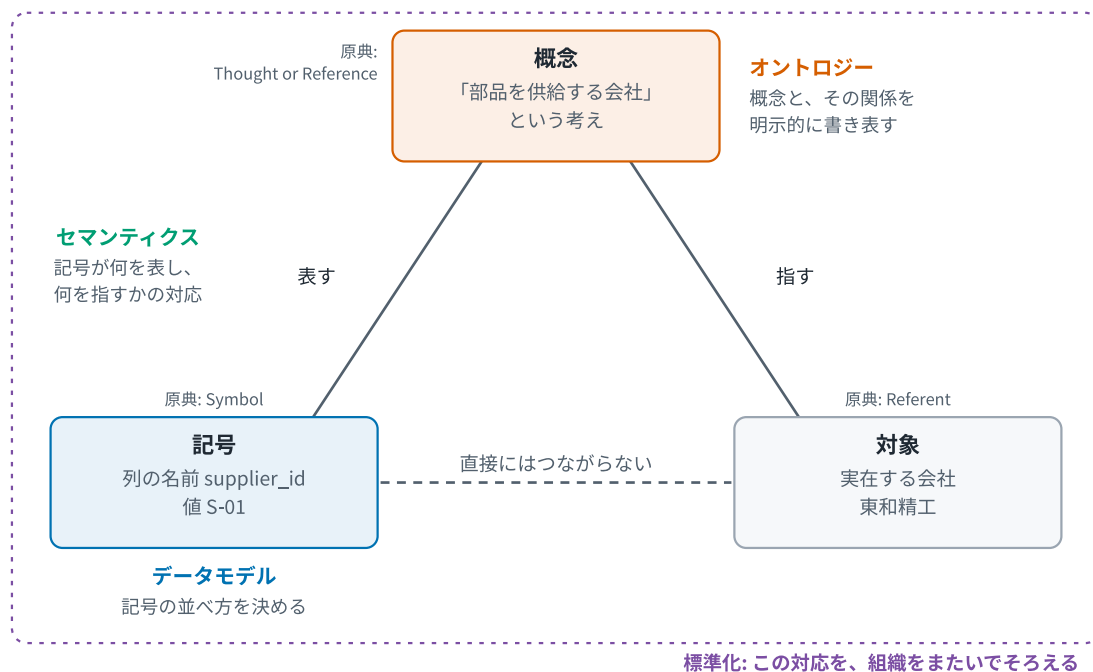


図 2: 意味の三角形と四つの言葉。三角形は Ogden と Richards (1923) の図をもとに筆者が作成した。頂点の日本語の呼び名と、四つの言葉の位置づけは筆者による。

supplier_id という列の名前は記号である。それが表す「部品を供給する会社」という考えが概念で、実在する会社が対象にあたる。四つの言葉は、この三角形のどこを扱うかが違う。機械に意味をどこま

で渡すかという段階との関係は、セクション 4.4 で示す。

データモデル

データの形を決めたものである。どんな項目があり、それぞれがどんな型の値を持ち、項目どうしがどうつながるかを定める。デジタル庁の政府相互運用性フレームワーク(GIF)は、データの構造や項目に加えて、データとデータの関係性を定義したものと説明している [3]。データの持ち方は表に限らない。JSON のような入れ子の文書も、点と線でつなぐグラフもある。どの持ち方にも、形の取り決めとしてのデータモデルがある(セクション 6)。

セマンティクス

データの意味のことである。「supplier_id は、その部品を供給している会社を指す」という取り決めがセマンティクスにあたる。データモデルが形を決めても、意味までは決まらない。

オントロジー

ある分野に出てくる概念と、その関係を、明示的に書き表したものである。知識工学では「概念化の明示的な仕様」と定義されてきた [4]。「部品」「供給者」「供給する」といった概念を定義する。OWL のような言語で書けば、計算機がその内容を処理できる [5]。

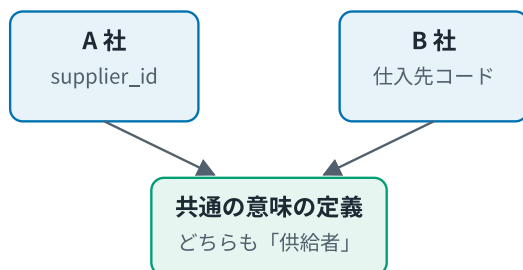
データモデル標準化

組織をまたいで、データモデルをそろえるための取り決めである。そろえるのは、まずデータの形である [3], [6]。ただし、形がそろっても意味がそろうとは限らない。そのため標準化は、語彙の取り決めに及ぶ [7], [8]。全員に同じものを使わせる方法だけでなく、別々の言葉を対応づける方法もある [7]。

1.3 なぜ今これが話題になるのか

理由は二つある。

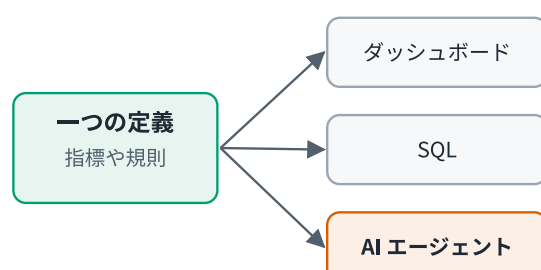
理由 1: 組織をまたいでデータを共有する



組織の間では、列の名前の意味が
その場ですり合わせられない。

どちらも、意味をあらかじめ定義して機械に渡すことを求める。

理由 2: AI がデータを読む



定義がなければ、AI は意味を推測する。
推測は外れることがあり、毎回同じとも限らない。

図 3: 意味の定義が話題になる二つの理由。筆者の整理である。

一つ目は、組織をまたいでデータを共有する仕組みである。欧州相互運用性フレームワークは、相互運用性を整理する 4 層の一つに意味の層を置く [9]。日本の ODS-RAM も、4 つのレイヤの一つにセマンティクスレイヤを置く [10]。詳しくはセクション 10 で扱う。

二つ目は、AI にデータを使わせる製品の動きである。Databricks は、指標や規則を一度定義すれば、ダッシュボードも SQL も AI エージェントも同じ定義で動くと説明している [11]。これは製品ベンダの説明である。最初の節で述べたとおり、AI の推測は安定しない。意味をあらかじめ定義して AI に渡しておく必要がある、と筆者は考える。

1.4 歴史は四本の流れに分けられる

この分野の取り組みは、1970 年代のデータベース研究から 2026 年の企業の仕様づくりまで、半世紀にわたる。筆者はこれを四本の流れに分けて整理した。流れごとに、何のための取り組みか、何が得られるか、そのために何をするかを分けて示す。

表 1: 四本の流れの目的と手段(筆者の整理)

流れ	何のために	得られるもの	そのためにすること	代表例
データベース系	データを保存し、複数のプログラムで共有する	同じデータを、別々のプログラムから同じ形で使える	データの形を、表や図で取り決める	関係モデル、実体関連モデル
知識表現系	人が持つ知識を、計算機に扱わせる	書かれていないことを、定義から導ける	概念とその関係を、論理として書き表す	記述論理、上位オントロジー
Web 標準系	組織をまたいで、データの意味を共有する	出どころの違うデータを、そのまま突き合わせられる	共通の書き方と語彙を、標準として定める	RDF、OWL、DCAT
分析系	指標の計算方法を、一つにそろえる	どのツールで集計しても、同じ数字になる	指標の定義を一か所に置き、各ツールから参照する	セマンティックレイヤ、Apache Ossie

四本は目的が違う。同じ「セマンティック」という言葉も、Web 標準系では組織をまたぐ意味の共有を、分析系では指標の計算方法の統一を指すと考えられる(セクション 11)。この報告書を読むときは、いまどの流れの話をしているかを意識すると混乱しにくい。流れごとの出来事は図 9 に年表としてまとめた。

1.5 この報告書の読み方

表 2: 目的別の読み方

知りたいこと	読む場所
結論だけ知りたい	セクション 2
用語の正確な定義	セクション 4
全体の流れを図で見たい	図 9
RDF や OWL などが何か知りたい	セクション 6
米国企業の最近の動き	セクション 9
データスペースとの関係	セクション 10
流れごとの比較	セクション 11

2 エグゼクティブサマリ

調査の問い

「オントロジー」「セマンティクス」「データモデル標準化」という語は、時代と業界によって指すものが違う。この調査では、その違いを、1970 年代の研究から近年の米国企業の取り組みまで並べて整理した。

主要な発見

1. 「オントロジー」には少なくとも三つの用法がある。

- ・ **知識工学の用法:** 概念化を明示的に書き表した仕様を指す。Gruber が「概念化の明示的な仕様」と定義し [4]、Studer らは、Gruber と Borst の定義を踏まえて「共有された概念化の、形式的で明示的な仕様」とした [12]。
- ・ **Web 標準の用法:** OWL のような、形式的に定義された意味を持つオントロジー言語で書いたものを指す [13]。日本の ODS のオープンソース実装も、この用法で「ODS オントロジー」を定義している [14]。
- ・ **製品の用法:** Palantir は自社の Ontology を、組織の「運用レイヤ」と定義している [15]。

2. 系譜は四本に整理できる。この分類は筆者の整理である。データベース系(関係モデル、実体関連モデル)、知識表現系(KL-ONE、Cyc、記述論理)、Web 標準系(RDF、OWL、SKOS、SHACL、JSON-LD、DCAT)、分析系(セマンティックレイヤ)である。

3. 米国企業の近年の動きは分析系から始まり、オントロジーへ範囲を広げている。Open Semantic Interchange (OSI)は 2025 年 9 月 23 日に Snowflake らが発表した [16]。その後 Apache Incubator に受け入れられ、Apache Ossie (Incubating) に改称された。Apache 側の記録ではインキュベーション開始は 2026 年 6 月 22 日である [1], [17]。プロジェクトのブログ記事は Ossie を「セマンティックレイヤとオントロジーの両方のためのオープン仕様」と説明している [1]。Google Cloud は 2026 年 6 月 12 日付の記事で、知識を Markdown のファイルで表す Open Knowledge Format を発表した [18]。型の語彙を固定せず、意味の多くを文章のまま渡して、解釈を読み手の言語モデルに任せる形式である(セクション 9.5)。

4. データスペースプロトコルは Web 標準系の語彙を再利用している。同プロトコルは DCAT と ODRL の語彙を再利用し、JSON-LD 1.1 を用いる [8], [19]。日本の ODS-RAM は 4 レイヤの一つとしてセマンティクスレイヤを置く [10]。設計思想として開世界仮説を前提に置き、オープンソース実装は OWL で書いたオントロジーと SAMM を使う(セクション 10.3)。

考察にもとづく示唆

- ・ 技術は「機械が意味について何をできるようになるか」という一本の軸で、4 段に並べると整理しやすい(セクション 4.4)。
- ・ Web 標準系と分析系は、同じ「セマンティック」という語を使うが、解く問題が違う。混ぜずに並べて示すほうがよいと考えられる(セクション 11)。

確認できなかった点

一部の事項は一次情報で裏づけられなかった。該当する日付は本文に書かず、セクション 12 にまとめた。

3 調査の範囲

調査は 2026 年 10 月 2 日時点の公開情報にもとづく。標準文書、学術論文、公的機関の文書を優先し、企業の動きは各社の公式発表と公式文書で確認した。企業の公式発表は当事者の主張であって、中立な評価ではない。

4 用語の整理

4.1 この報告書で使う整理の軸

この分野の用語が分かりにくいのは、一つの軸では整理しきれないためである。この報告書は、次の五つの軸を使い分ける。それぞれ、並べるものも、分かることも違う。

表 3: この報告書で使う整理の軸

軸	何を並べるか	意図	得られる効果	出どころ	扱う場所
意味の三角形	記号、概念、対象の三つ(呼び名は筆者の言い換え)	言葉が何を扱っているかを見分ける	データモデル、セマンティックス、オントロジーの違いが分かる	Ogden と Richards	セクション 1
意味の指定の強さによる連続体	成果物の種類。用語集からオントロジーまで	成果物がどれだけ形式的かを比べる	既存の議論の中での位置づけが分かる	Uschold と Gruninger	セクション 4.3
意味の 4 段階	機械にできること	意味をどこまで機械に渡すかを比べる	ある技術で何ができ、何ができないかが分かる。必要な技術を選ぶ手が必要になる	筆者	セクション 4.4
四本の流れ	取り組みの系譜。誰が何のために進めてきたか	目的の違いを見分ける	同じ言葉が流れごとに別の課題を指す、という混乱を解ける	筆者	セクション 5
共有の範囲	一つの組織の中か、組織をまたぐか	意味を誰と共有するかを比べる	製品のオントロジーと、データスペースの取り組みの違いが分かる	筆者	セクション 10、セクション 11

五つの軸は、互いに置き換えられるものではない。たとえば、意味の指定の強さによる連続体と意味の 4 段階は、どちらも四つに分かれているが、並べているものが違う(表 5)。意味の三角形と意味の 4 段階の関係は 図 6 に、意味の 4 段階と四本の流れの関係は セクション 5.4 に示す。

出どころが「筆者」の軸は、この報告書のための整理であり、確立された分類ではない。

考察：なぜ独自の軸を加えたのか

先行する二つの軸は、人が言葉や成果物を見分けるための整理である。意味の三角形は言葉が何を扱うかを示し、連続体は成果物がどれだけ形式的かを示す。連続体の原典は、右へ進むほど自動推論への支援が増すとも述べている [20]。ただし並べているのは成果物の種類であり、機械にできることを段階に分けてはいない。

近年は、機械に何ができるかという問いの重みが増していると考えられる。データを読むのが、人と分析ツールだけではなくなったためである。Snowflake らは OSI の狙いを、AI と BI のアプリケーションで一貫した業務ロジックを確保することと述べている [16]。Databricks は、一度定義すればダッシュボードも SQL も AI エージェントも同じ定義で動くと説明している [11]。Google Cloud は Open Knowledge Format を、エージェントと人の両方に読みやすい形式として発表した [18]。いずれも製品ベンダの説明だが、意味の定義の読み手として AI を挙げている点が共通する。

AI の推測は安定しない(セクション 1)。AI を業務に組み込むなら、どこまでを推測に任せ、どこからを定義として渡すかを決める必要がある。その判断には、成果物の種類で比べるよりも、機械にできることで比べるほうが役に立つ。

そこで筆者は、意味の 4 段階を加えた。四本の流れと共有の範囲の二つは、AI とは別の理由で加えたものである。同じ言葉が取り組みごとに別の課題を指すという混乱を解くために、誰が何のために、誰と共有するのかを分けて見る必要があった。

4.2 出典による定義

表 4: 主要用語の出典による説明

用語	出典による説明	出典
オントロジー(知識工学)	“An ontology is an explicit specification of a conceptualization.” (訳: オントロジーとは、概念化の明示的な仕様である。)	[4]
オントロジー(改訂版の定義)	“An ontology is a formal, explicit specification of a shared conceptualisation.” (訳: オントロジーとは、共有された概念化の、形式的で明示的な仕様である。)	[12]
オントロジー(工学的成果物)	語彙と、その意味に関する明示的な前提からなる工学的な成果物	[21]
セマンティックウェブ	境界を越えてデータを共有し再利用できる共通の枠組み	[22]
意味的相互運用性	交換されるデータと情報の正確な形式と意味が、やり取りを通じて保たれ理解されること。意味の側面と構文の側面の両方を含む	[9]
データモデル(欧州のデータスペース)	意味的相互運用性を促すために、データ要素と関係を構造化して表したもの。語彙、オントロジー、アプリケーションプロファイル、スキーマ仕様を含む	[23]
意味に関するメタデータ(日本の ODS)	データやサービスを利用するために必要な情報の意味を、ユニークかつアクセス可能な形式で定義したもの。ODS-RAM がセマンティクスレイヤの要求として定める	[10]
セマンティックレイヤ	データウェアハウスとそれを使うツールの間に位置するソフトウェア層。ベンダの用語集による説明である	[24]
知識グラフ	多様で動的かつ大規模なデータ集合を活用する場面で注目されるデータ表現	[25]

英語の引用に添えた日本語訳は筆者による。

日本の ODS-RAM には、欧州の「データモデル」に当たる用語がない。対応する概念は「意味に関するメタデータ」と呼ばれる [10]。ODS のプロトコル仕様にも「データモデル」という語はあるが、「情報モデルとして RDF を採用し、データモデルとして JSON-LD 形式や Turtle 形式を」採用するという文脈で、直列化の形式を指す [26]。同じ語が欧州と日本で別のものを指している。

Studer らの定義は、先行する定義を踏まえたものである [12]。Guarino は、上位オントロジーを特定の問題や領域に依存しない一般的な概念を記述するものとした [21]。同じ論文は、領域オントロジー、タスクオントロジー、アプリケーションオントロジーという区分も示している [21]。

4.3 先行する整理：意味の指定の強さによる連続体

用語集からオントロジーまでを一本の軸に並べる整理は、以前からある。

Uschold と Gruninger は、用語だけの軽いものから、厳密に形式化した論理の理論までを連続体として示し、連続体に沿って進むほど、指定される意味の量と形式性の度合いが増すと述べた [20]。McGuinness も、同様の図をオントロジー・スペクトラムと呼んでいる [27]。

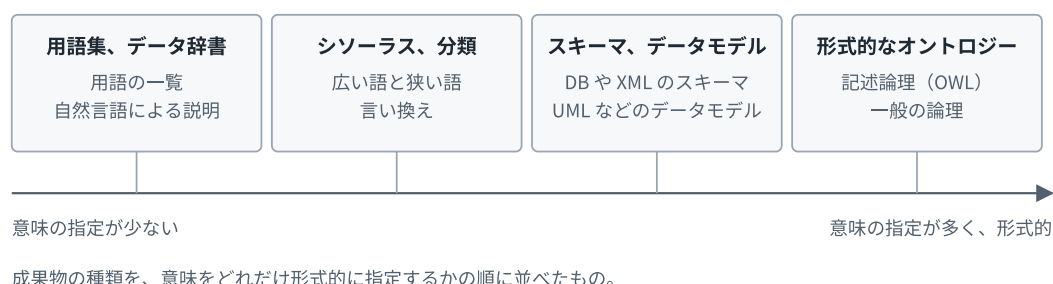


図 4: 意味の指定の強さによる連続体。Uschold と Gruninger (2004) の図をもとに筆者が作成した。四つの群は原典の図の区分による。

関連する区別を、ほかに二つ挙げておく。

- ・ **情報モデルとデータモデル**: IETF の RFC 3444 は、情報モデルを実装やプロトコルに依存しない概念の水準のモデル、データモデルをそれより低い抽象度で多くの詳細を含むモデルとし、一つの情報モデルから複数のデータモデルを導けるとしている [28]。日本の ODS も、データモデルから情報モデルを分離する(セクション 10.3.1)。
- ・ **語彙とオントロジー**: W3C は、語彙と呼ばれるものとオントロジーと呼ばれるものの間に明確な区分はないとしている。複雑で形式的なものをオントロジーと呼ぶ傾向がある、という説明がある [29]。

次の節で示す意味の 4 段階は、この連続体とは別のものである。どちらも四つに分かれているが、並べているものと並べる基準が違う。

表 5: 連続体と意味の 4 段階の違い

	意味の指定の強さによる連続体	意味の 4 段階
誰の整理か 並べているもの	Uschold と Gruninger 成果物の種類。用語集、シソーラス、スキーマ、形式的なオントロジー	筆者 機械にできること
並べる基準	意味の指定の量と、形式性の度合い。進むほど自動推論への支援も増すとされる	機械に渡す意味の深さ
四つである理由	原典の図が四つの群に分けている	筆者が四つに整理した

二つは一对一には対応しない。連続体ではシソーラスがスキーマより手前にある。意味の4段階では、スキーマにあたる1段目が、シソーラスにあたる3段目より手前にある。連続体は成果物がどれだけ形式的かを見ており、4段階は機械が何をできるようになるかを見ているためである。

4.4 考察：意味の4段階

以下は筆者による整理であり、特定の出典の主張ではない。この軸を加えた理由はセクション4.1に述べた。

技術は「機械が意味について何をできるようになるか」という一本の軸で、段に並べられると考えられる。2段目から4段目は積み上がる。3段目の用語の関係も、4段目の推論も、2段目の共通の識別子がなければ成り立たない。1段目の検査は、これらとは独立している。RDFで書くのにJSON Schemaは要らない。検査は、どの段のデータに対しても別に行える。



図 5: 意味の4段階。段を上がるごとに、機械が意味についてできることが増える。ただし1段目は、2段目から4段目の前提ではない。

表 6: 意味の4段階(筆者の整理)

段	機械ができるようになること	部品と供給者の例	代表的な技術
1	形を検査できる	重量の欄に数値が入っていると分かる	JSON Schema、XML Schema、関係モデル
2	同じものだと分かる	A社の列とB社の列が、どちらも供給者を指すと分かる	RDFと共有の語彙
3	用語の関係をたどれる	「ボルト」は「締結部品」より狭い語だとたどれる	SKOS
4	定義から導ける	供給者を持つものは調達品だと、書かれていなくても導ける	RDF Schema、OWL

4段階と、セクション5で示す四本の流れとの関係は、セクション5.4で説明する。セクション2で挙げたオントロジーの用法のうち、知識工学とWeb標準の用法は4段目にあたる。製品の用法である

Palantir の Ontology は、一つの段に収まらず、複数の段にまたがる(図 8)。各技術の説明はセクション 6 にある。

意味の三角形との対応

4 段階は、図 2 の意味の三角形と対応づけられる。三角形は、四つの言葉が何を扱うかを示す。4 段階は、その三角形のうち、どこまでを機械に書いて渡すかを示す。

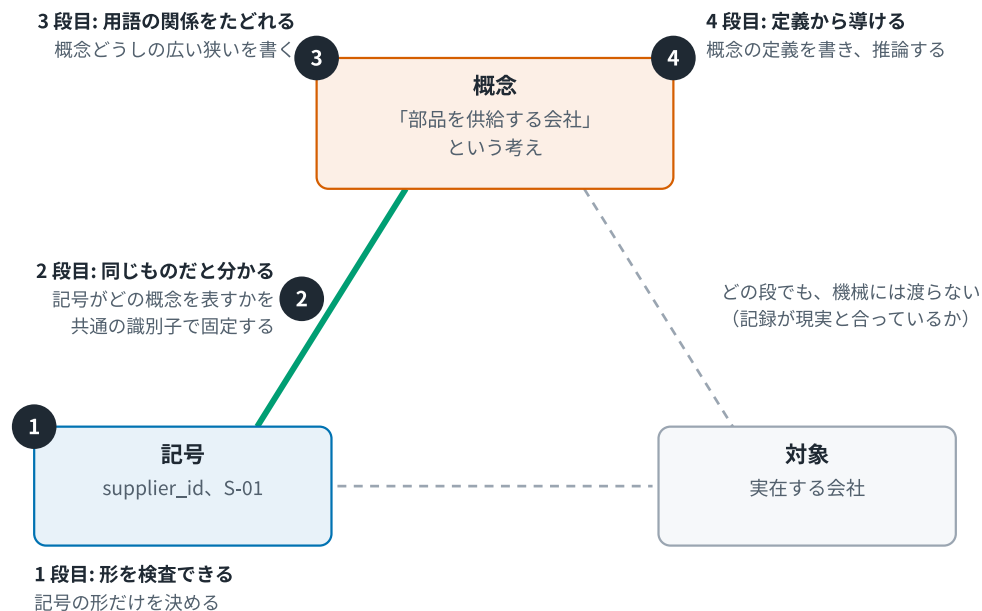


図 6: 意味の三角形と意味の 4 段階の対応。段を上がるとは、記号の側だけを決めていた状態から、概念の側まで機械に書いて渡す状態へ進むことである。筆者の整理である。

表 7: 意味の 4 段階と意味の三角形の対応(筆者の整理)

段	三角形のどこを機械に渡すか	四つの言葉との対応
1 段階目	記号の頂点だけ。列の名前や値の形を決める	データモデル
2 段階目	記号から概念への辺。この記号がどの概念を表すかを、共通の識別子で固定する	セマンティクス
3 段階目	概念の頂点の中身。用語どうしの広い狭い関係を書く	統制語彙、シソーラス。広い意味ではオントロジーと呼ばれることもある
4 段階目	概念の頂点の論理。概念の定義を書き、そこから推論する	オントロジー

標準化は段ではない。どの段であれ、その取り決めを組織の間でそろえることを指す。

三角形のもう一つの辺、概念から対象への対応は、どの段でも機械には渡らない。記録が現実の会社と合っているかは、技術の外で確かめる必要がある。

この対応づけは筆者によるものである。Ogden と Richards の図にも、Uschold と Gruninger の連続体にも、こうした対応は書かれていない。

一つの段に収まらない技術

4段階は、機械にできることを一つずつ切り出したものである。現実の技術や製品は、その一つだけを担うとは限らない。一つの段にきれいに収まらない技術には、三つの種類がある。

- **またがる**: 複数の段を、それぞれ一部ずつ備える。一つの製品や形式が、検査も同定も少しずつ受け持つ場合である。
- **同じ役割を担う**: ある段と同じ役割を、別の対象に対して果たす。新しい段を足すわけではない。
- **別の軸にある**: 段の仕組みを使ったり、その上に載ったりするが、目的が意味の深さとは別のところにある。

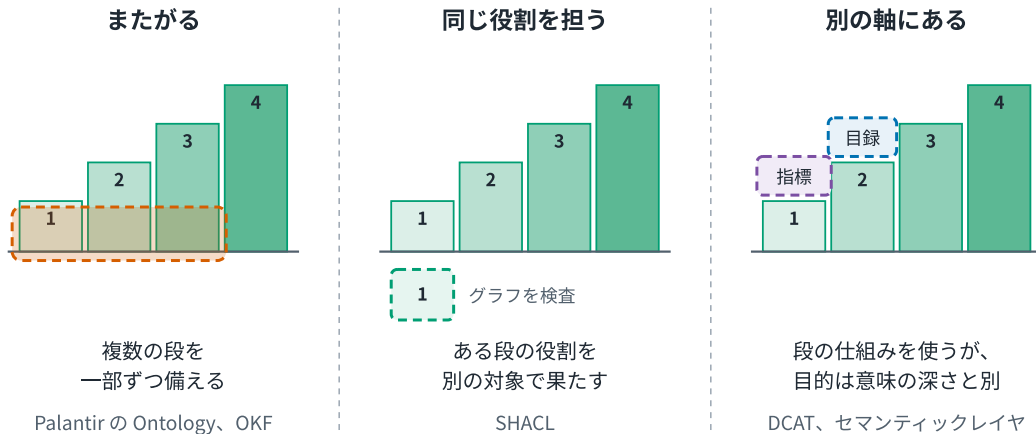


図 7: 一つの段に収まらない技術の三つの種類。緑の階段が意味の 4 段階、破線の枠が収まらない技術を表す。筆者の整理である。

どの種類も、いずれかの段と関係がある。図 8 に、それぞれが関係する段を示す。

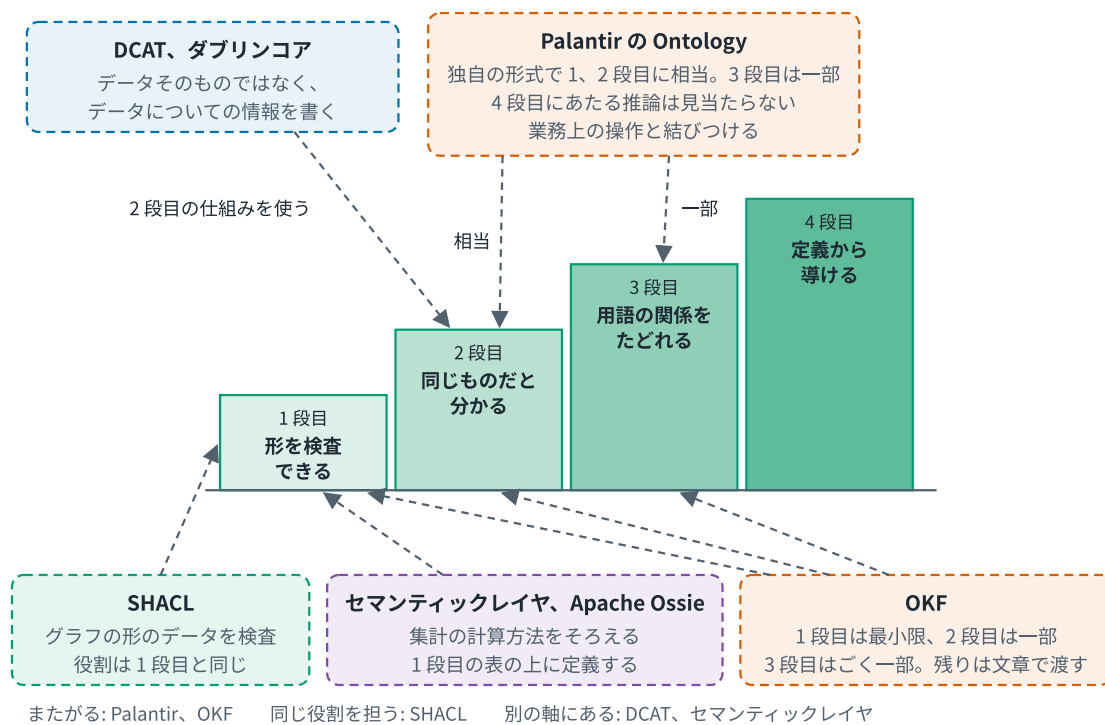


図 8: 一つの段に収まらない技術と、各段との関係。破線の箱がその技術で、矢印は関係する段を指す。Palantir の Ontology は、1、2 段目に相当する機能と、3 段目の一部にあたる機能を持つ。図の矢印は 2 段目と 3 段目だけに引いた。Open Knowledge Format は、1 段目を最小限に、2 段目を一部、3 段目をごく一部だけ備える。

表 8: 一つの段に収まらない技術(筆者の整理)

技術	種類	一つの段に収まらない理由	関係する段
Palantir の Ontology	またがる	W3C の標準とは別の、製品独自の形式である。意味の定義を業務上の操作と結びつける	1、2 段目に相当する機能を持つ。3 段目は型の継承という形で一部を備える。4 段目にあたる推論は、公式文書に見当たらない(表 14)
Open Knowledge Format	またがる	意味の多くを文章のまま言語モデルに渡し、解釈を読み手に任せる	1 段目を最小限に備える。2 段目は資産を指す URI に限って一部を備える。3 段目は種類のないリンクにとどまり、ごく一部である。4 段目は見当たらない(表 18)
SHACL	同じ役割を担う	新しい段ではなく、検査という 1 段目と同じ役割を担う	1 段目。検査の対象が、グラフの形のデータに変わる
DCAT、ダブリンコア	別の軸にある	記述する対象が違う。データそのものではなく、データについての情報を扱う	2 段目。共有の語彙として、2 段目の仕組みで書かれる
セマンティックレイヤ、Apache Ossie	別の軸にある	用途が違う。集計の計算方法をそろえる	1 段目。表のデータの上に、指標の定義を置く

5 歴史の系譜

5.1 四本の流れ

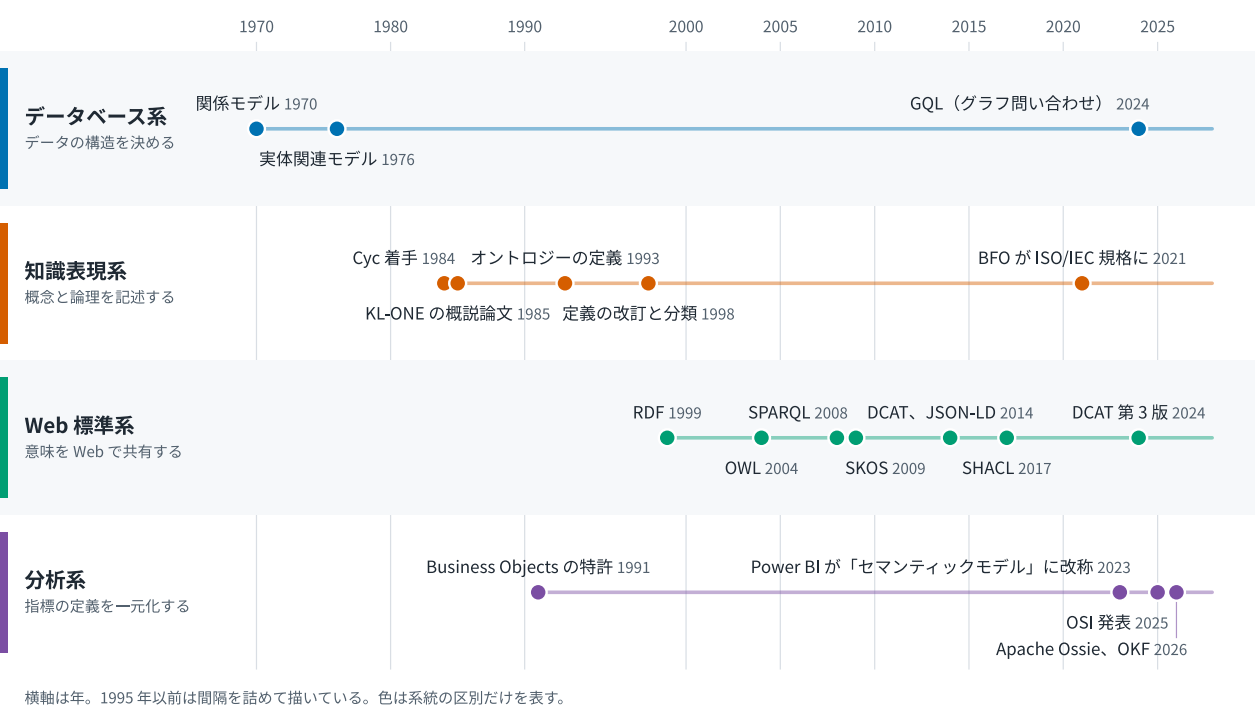


図 9: 四本の系譜の年表。系統ごとに主な出来事を年の順に並べた。

図 9 は筆者による整理である。どの出来事をどの系統に置くかは筆者の分類で、系統の間の影響関係は図に描いていない。年は表 9 の出典による。

5.2 年表

表 9: 年表

年月	系統	出来事	要点	出典
1970 年 6 月	データベース	Codd の論文が Communications of the ACM に掲載	データの関係モデルを提案した	[30]
1976 年 3 月	データベース	Chen の論文が ACM TODS に掲載	実体関連モデルを提案し、データの統一的な見方を目指した	[31]
1984 年	知識表現	Lenat らが Cyc の構築に着手	人間の常識にあたる一般的な概念を、大規模に記述する試みが始まった(筆者の要約)	[32]

年月	系統	出来事	要点	出典
1985 年 4 月	知識表現	KL-ONE の概説論文が Cognitive Science に掲載	知識表現システム KL-ONE の全体像が示された	[33]
1991 年	分析	Business Objects が、関係データベースへの問い合わせ技術の特許を出願(優先日は 11 月 27 日)	利用者が関係構造や SQL を知らなくても問い合わせられる仕組みが示された	[34]
1993 年	知識表現	Gruber がオントロジーを定義	オントロジーを「概念化の明示的な仕様」として定義	[4]
1998 年	知識表現	Studer らの定義、Guarino の分類	定義に「共有」と「形式的」が加わり、上位、領域、タスク、アプリケーションという分類が示された	[12], [21]
1999 年 2 月 22 日	Web 標準	RDF の仕様が W3C 勧告	RDF をメタデータ処理の基盤と位置づけた	[35]
2000 年 5 月	知識表現(生命科学)	Gene Ontology の論文が Nature Genetics に掲載	生物学の知識を統一するための道具としてオントロジーを示した	[36]
2001 年 5 月	Web 標準	「The Semantic Web」が Scientific American に掲載	セマンティックウェブの構想を広く紹介した	[37]
2001 年 5 月 2 日	Web 標準	XML Schema (Part 1: Structures) が W3C 勧告	XML 文書の構造を記述し、内容を制約する言語が定まった	[38]
2003 年 3 月	分析	OMG が CWM 1.1 版を発行	データウェアハウスのメタデータを交換するための共通のメタモデルが示された(筆者の要約)	[39]
2004 年 2 月 10 日	Web 標準	OWL が W3C 勧告	情報の内容进行处理するアプリケーション向けの、オントロジー言語が定まった	[5]
2008 年 1 月 15 日	Web 標準	SPARQL が W3C 勧告	RDF の問い合わせ言語が定まった	[40]
2009 年 8 月 18 日	Web 標準	SKOS が W3C 勧告	知識組織化体系を共有しリンクする、共通データモデルが定まった	[41]

年月	系統	出来事	要点	出典
2011 年 6 月 2 日	Web 標準(語彙)	Google、Bing、Yahoo! が schema.org を発表	Web ページの構造化データに使う、共通のスキーマ群が示された	[42]
2012 年 5 月 16 日	知識表現(知識グラフ)	Google がナレッジグラフを紹介	検索の対象を、文字列ではなく実世界の事物とその関係とした	[43]
2012 年 12 月 11 日	Web 標準	OWL 2 第 2 版が W3C 勧告	OWL の改訂版。形式的に定義された意味を持つオントロジー言語	[13]
2014 年 1 月 16 日	Web 標準	DCAT と JSON-LD 1.0 が W3C 勧告	データカタログのための語彙と、Linked Data を JSON で書く形式が定まった	[44], [45]
2014 年 1 月 16 日	Web 標準	RDF Data Cube 語彙が W3C 勧告	ディメンションとメジャーを RDF で書く語彙が勧告になった	[46]
2014 年 2 月 25 日	Web 標準	RDF 1.1 が W3C 勧告	RDF の改訂版	[47]
2017 年 7 月 20 日	Web 標準	SHACL が W3C 勧告	RDF グラフを条件に照らして検証する言語が定まった	[48]
2020 年 7 月 16 日	Web 標準	JSON-LD 1.1 が W3C 勧告	JSON-LD の改訂版	[49]
2021 年	知識表現	BFO が ISO/IEC 21838-2:2021 に	上位オントロジーの一つが国際規格になった	[50]
2023 年	業界標準(製造)	アセット管理シェルの構造が IEC 63278-1:2023 に	アセットの標準化されたデジタル表現の構造が国際規格になった	[6]
2023 年 11 月	分析	Power BI の「データセット」が「セマンティックモデル」に改称	分析用のデータ定義を「セマンティック」と呼ぶ用語が、主要製品に入った(筆者の要約)	[51], [52]
2024 年 2 月 13 日	知識表現(知識グラフ)	Microsoft Research が GraphRAG を紹介	言語モデルと知識グラフを組み合わせた手法が示された(筆者の要約)	[53]
2024 年	データベース	GQL が ISO/IEC 39075:2024 に	プロパティグラフのデータ構造と基本操作が国際規格になった	[54]
2024 年 8 月 22 日	Web 標準	DCAT 第 3 版が W3C 勧告	DCAT の改訂版	[55]

年月	系統	出来事	要点	出典
2025 年 9 月 23 日	分析	Snowflake らが OSI を発表	ベンダ中立のセマンティックモデル仕様を作る取り組みが始まった	[16]
2025 年 10 月 14 日	分析	dbt Labs が MetricFlow のオープンソース化を発表	指標を定義するエンジンが Apache 2.0 ライセンスになった	[56]
2025 年 12 月 2 日	データスペース	Eclipse Foundation が 2 仕様の ISO/IEC 提出を発表	データスペースプロトコルが国際標準化の手続きに進んだ	[57]
2026 年 1 月 27 日	分析	OSI 仕様の最初の版を公開	仕様が Apache 2 ライセンスのリポジトリで読めるようになった	[58]
2026 年 4 月 2 日	分析	Databricks が Unity Catalog Business Semantics の一般提供とオープンソース化を告知	指標とディメンションの定義を、データ層で一度だけ行う機能が提供された	[11]
2026 年 4 月 7 日	Web 標準	RDF 1.2 が勧告候補スナップショット	RDF の次の改訂版。2026-10-02 時点で勧告ではない	[59]
2026 年 6 月 12 日	分析	Google Cloud が Open Knowledge Format を発表	言語モデルに渡す文脈を、Markdown のファイルで表す形式が示された(筆者の要約)	[18]
2026 年 6 月 22 日	分析	Ossie が Apache Incubator でインキュベーション開始	仕様づくりの場が Apache Software Foundation に移った	[17]
2026 年 7 月 10 日	分析	プロジェクトがブログ記事で改称とインキュベーション入りを公表	名称が OSI から Apache Ossie (Incubating) になった	[1]

各技術要素の説明は セクション 6 にある。「系統」の列は図 9 の四本に対応する。業界標準とデータスペースの行は、四本の外にある関連の出来事である。「要点」の列は出典の記述にもとづく。出典に直接の記述がなく筆者がまとめた行には「筆者の要約」と記した。

5.3 知識表現からセマンティックウェブへ

OWL DL という名は記述論理との対応に由来する。記述論理は OWL の形式的な基礎となる論理を研究する分野である [5]。KL-ONE が記述論理の源流の一つであるという見方は広く語られるが、本調

査ではそれを直接述べる一次情報を確認していない。

5.4 意味の 4 段階と四本の流れの関係

セクション 4.4 の意味の 4 段階と、図 9 の四本の流れは、別の軸である。どちらも筆者による整理である。

- **四本の流れ**は、誰が何のために取り組んできたかという、歴史の分類である。
- **意味の 4 段階**は、機械に意味をどこまで渡すかという、深さの分類である。

二つを重ねると 図 10 のようになる。

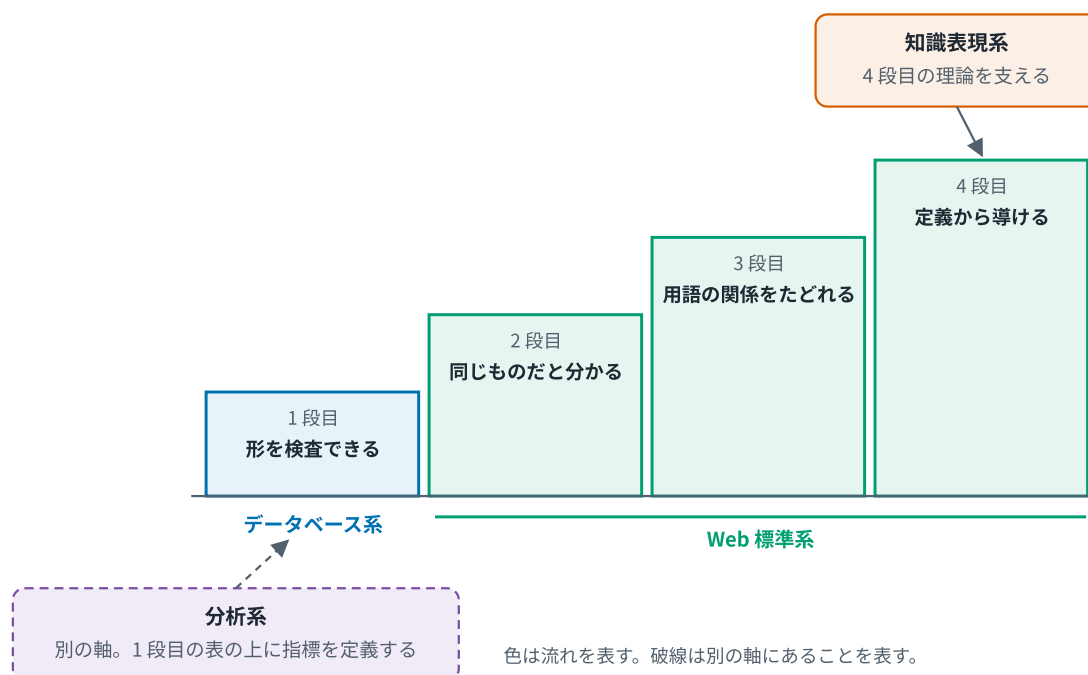


図 10: 意味の 4 段階と四本の流れの対応。段の色は、その段を主に担う流れを表す。

表 10: 四本の流れと意味の 4 段階の対応(筆者の整理)

流れ	対応する段	説明
データベース系	1 段目	データの形を決める。関係モデル、実体関連モデル、プロパティグラフがここにあたる
Web 標準系	2、3、4 段目	2 段目が RDF、3 段目が SKOS、4 段目が RDF Schema と OWL である。1 段目の XML Schema も含む。SHACL と DCAT は一つの段に収まらない(表 8)が、これも Web 標準系に属する

流れ	対応する段	説明
知識表現系	4 段目	4 段目の理論の出どころである。記述論理が OWL の形式的な基礎になった。上位オントロジーは 4 段目で書かれる中身にあたる
分析系	別の軸にある	1 段目の表の上に、指標の定義を置く

筆者はこの対応から、次の三点を読み取る。

1. 2 段目から上を担うのは Web 標準系である。データベース系は 1 段目にとどまる。表 6 に挙げた 2 段目から 4 段目の代表的な技術(RDF、SKOS、RDF Schema、OWL)は、いずれも Web 標準系の仕様である。
2. 4 段目には二つの流れが関わる。知識表現系が理論を、Web 標準系がその標準化を担った。記述論理が OWL の形式的な基礎であることは、出典で確認できる [5]。
3. 分析系は別の軸にある。解いているのは、指標の計算方法という別の課題である。

6 技術要素の解説

セクション 5 の図と年表に出てくる技術要素を、系統ごとに一つずつ説明する。

各項目は次の順に書く。

- ・ **概要:** 何であるか。出典の記述にもとづく。
- ・ **解く問題:** これがないと何に困るか。
- ・ **具体例:** 部品と供給者という共通の題材で書いた例。
- ・ **似たものとの違い:** 混同しやすい相手との比較。
- ・ **使われている場所:** 採用例。出典で確認できたものだけを書き、確認できなかった項目では省いた。
- ・ **限界と注意点:** 向かない用途と、よくある誤解。

「解く問題」「似たものとの違い」「限界と注意点」は筆者による解説である。具体例は筆者が作成したもので、仕様書からの転載ではない。関係モデル、RDF 系の各言語、JSON Schema、Apache Ossi の例は、処理系または公式のスキーマで検証した。GQL の例だけは検証していない。

題材は一貫して次のものを使う。「六角ボルト M8」という部品(番号 P-100、重量 12 グラム)を、「東和精工」という供給者(番号 S-01)が供給している。

6.1 データベース系

6.1.1 関係モデル

概要

データを表の形で扱うモデルである。1970 年に Codd が論文で提案した [30]。

解く問題

プログラムごとにばらばらだったデータの持ち方を、表という共通の形にそろえる。データの持ち方を変えても、利用する側のプログラムを書き直さずに済むようにする。

具体例

```

CREATE TABLE supplier (
  supplier_id TEXT PRIMARY KEY,
  name        TEXT NOT NULL
);
CREATE TABLE part (
  part_id      TEXT PRIMARY KEY,
  name         TEXT NOT NULL,
  weight_g     REAL,
  supplier_id  TEXT NOT NULL REFERENCES supplier(supplier_id)
);
INSERT INTO supplier VALUES ('S-01', '東和精工');
INSERT INTO part VALUES ('P-100', '六角ボルト M8', 12, 'S-01');

SELECT part.name, supplier.name
FROM part JOIN supplier ON part.supplier_id = supplier.supplier_id;

```

部品 (part)

part_id	name	weight_g	supplier_id
P-100	六角ボルト M8	12	S-01

外部キーで参照

供給者 (supplier)

supplier_id	name
S-01	東和精工

データは表で持ち、表どうしは値の一致（キー）で結びつける。

図 11: 関係モデルの例。部品の表が供給者の表を参照する。

似たものとの違い

RDF との違いは、意味の置き場所にある。関係モデルでは `supplier_id` という列名が何を意味するかは設計書に書かれ、データ自体には含まれない。RDF では述語が識別子を持ち、その意味を別の文で記述できる。

使われている場所

Apache Ossie のセマンティックモデルは、物理的な表やビューを参照するデータセットを単位とする [60]。

限界と注意点

表の設計は組織ごとに異なる。組織をまたいでデータを突き合わせるには、列の意味についての合意が別に必要になる。

6.1.2 実体関連モデル

概要

データを、実体とその間の関連として捉えるモデルである。1976 年に Chen が論文で提案し、データの統一的な見方を目指した [31]。

解く問題

表を作る前に、何を管理し、それらがどう結びつくかを、利用者と設計者が同じ図で確認する。

具体例



設計の段階で、実体（四角）と関連（ひし形）、数の対応を図に描く。

図 12: 実体関連モデルの例。部品と供給者を実体、供給を関連として描く。

似たものとの違い

関係モデルが実装の形であるのに対し、実体関連モデルは設計の段階の図である。オントロジーと似ているが、推論のための論理的な定義は持たない。

限界と注意点

図は人が読むためのものである。図そのものを機械が解釈して推論することは想定していない。

6.1.3 プロパティグラフと GQL

概要

プロパティグラフは、ノードと辺にプロパティを持たせるグラフのデータモデルである。GQL はその問い合わせ言語で、ISO/IEC 39075:2024 がプロパティグラフのデータ構造と基本操作を定義する [54]。

解く問題

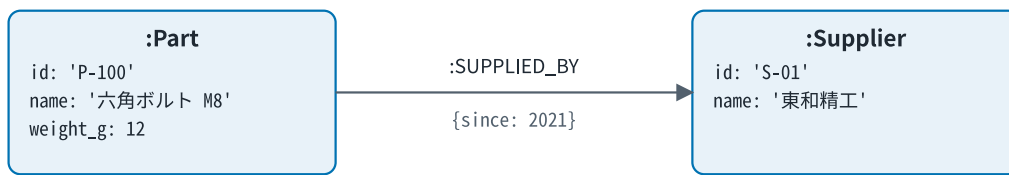
つながりをたどる問い合わせを、表の結合を何段も重ねずに書く。

具体例

```
// ノードと辺に、ラベルとプロパティを直接持たせる
INSERT (:Part {id: 'P-100', name: '六角ボルト M8', weight_g: 12})
    -[:SUPPLIED_BY {since: 2021}]->
    (:Supplier {id: 'S-01', name: '東和精工'})

// 供給者ごとの部品を調べる
MATCH (p:Part)-[:SUPPLIED_BY]->(s:Supplier)
RETURN p.name, s.name
```

この例は処理系で検証していない。



ノードにも辺にも、ラベルとプロパティを直接持たせる。

図 13: プロパティグラフの例。辺にもプロパティを持たせられる。

似たものとの違い

RDF もグラフだが、設計の目的が違う。Neo4j は、プロパティグラフはデータ交換形式ではなくデータベースモデルとして設計されたと説明している [61]。これは製品ベンダの説明である。RDF は識別子を Web 全体で一意にし、組織をまたぐ交換を重視する。

限界と注意点

ノードや辺の識別子は、ふつう一つのデータベースの中で閉じている。組織をまたいで同じものを指すには、別の取り決めが要る。

6.2 知識表現系

6.2.1 Cyc

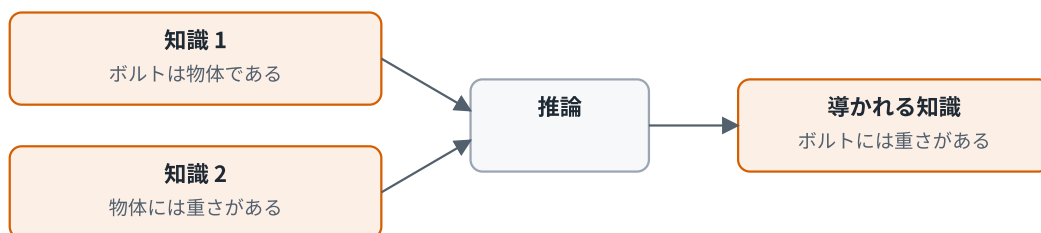
概要

人間の常識にあたる一般的な概念を、大規模に記述しようとした試みである。Lenat らが 1984 年に着手した [32]。

解く問題

プログラムが、人なら当然知っている背景知識を持たないために誤るという問題に、知識を書き下すことで対処しようとした。

具体例



筆者が作成した例であり、Cyc の実際の記述ではない。人には当然の知識を明示的に書いておく。

図 14: 常識の知識を書き下して推論する考え方。筆者が作成した例で、Cyc の実際の記述ではない。

限界と注意点

本調査では、Cyc の成果や現在の利用状況は確認していない。歴史上の起点の一つとして挙げた。

6.2.2 KL-ONE と記述論理

概要

KL-ONE は知識表現システムの一つで、1985 年に概説論文が出た [33]。記述論理は、OWL の形式的な基礎となる論理を研究する分野である [5]。

解く問題

概念の定義を、計算機が矛盾の検出や分類に使える厳密な形で書く。

具体例

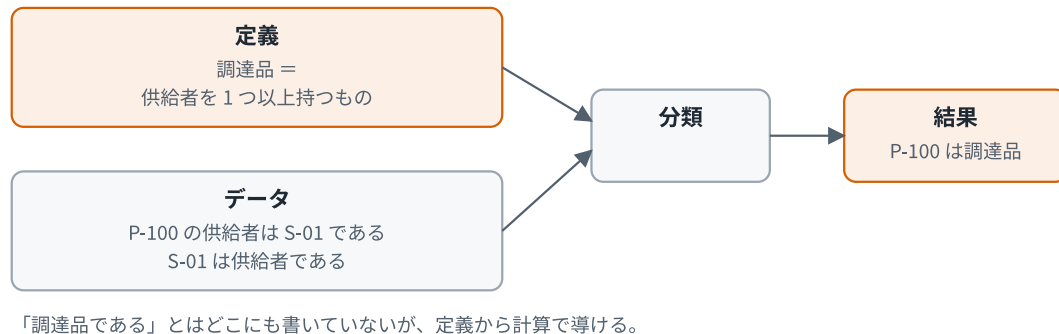


図 15: 記述論理による自動分類。定義とデータから、書かれていない分類を計算で導く。

OWL の項に、同じ定義を OWL で書いた例がある。

限界と注意点

KL-ONE が記述論理の源流の一つであるという見方は広く語られる。本調査では、それを直接述べる一次情報を確認していない。

6.2.3 上位オントロジー

概要

空間、時間、物、出来事のような、特定の領域に依存しない一般的な概念を記述するオントロジーである [21]。Basic Formal Ontology (BFO) は、ISO/IEC 21838-2:2021 に収められた上位オントロジーである [50]。

解く問題

分野ごとに作られたオントロジーを、共通の土台の上でつなぐ。

具体例

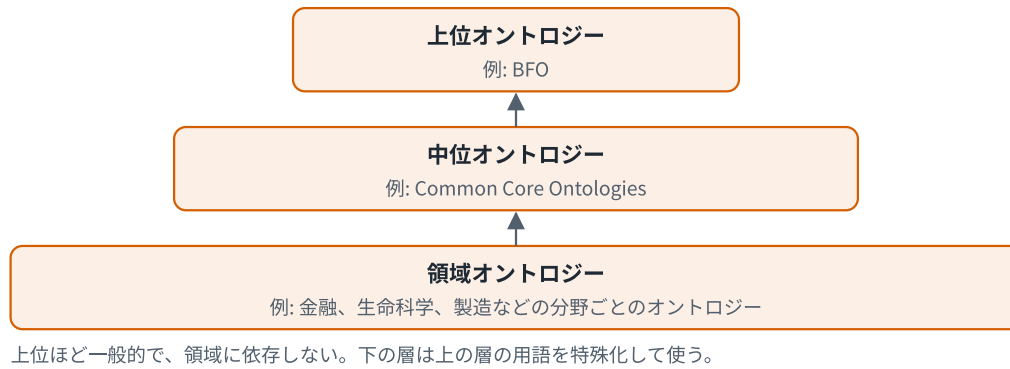


図 16: 上位、中位、領域オントロジーの重なり。

Common Core Ontologies は、あらゆる領域に共通する汎用的な用語と関係を定義する中位のオントロジー群である [62]。

似たものとの違い

領域オントロジーは特定の分野の語彙を扱う。上位オントロジーはその分野を問わない枠組みだけを与える。

使われている場所

金融の FIBO、生命科学の Gene Ontology、製造の Industrial Ontologies Foundry など、領域ごとのオントロジーは セクション 8 にまとめた。

限界と注意点

すべての領域オントロジーが、特定の上位オントロジーにもとづいて作られているわけではない。上位オントロジーの採用には、抽象的な区分を理解するための学習が必要になる。

6.2.4 知識グラフと GraphRAG

概要

知識グラフは、多様で動かつ大規模なデータ集合を活用する場面で、産業界と学术界の両方から注目されているデータの表現である [25]。Google は 2012 年に、検索の対象を文字列ではなく実世界の事物とその関係とするものとしてナレッジグラフを紹介した [43]。GraphRAG は、Microsoft Research が 2024 年に紹介した手法である [53]。

解く問題

ばらばらの情報源にある事物を、関係でつないで一つの網として扱う。

具体例



知識グラフは、事物を点、関係を線として持つデータの網である。

図 17: 知識グラフと GraphRAG の流れ。

似たものとの違い

オントロジーが概念と関係の定義であるのに対し、知識グラフは個々の事物とそのつながりを大量に持つデータである。知識グラフの基盤には、RDF もプロパティグラフも使われる [25]。

限界と注意点

「知識グラフ」は厳密に定義された規格の名前ではない。製品や文献によって指す範囲が異なる。

6.3 Web 標準系

W3C が勧告した標準を中心に説明する。全体の一覧は次のとおりである。

表 11: W3C の主要標準

標準	役割	勧告日	出典
RDF	Web 上で情報を表現する枠組み	1999-02-22、1.1 は 2014-02-25	[35], [47]
RDF Schema	RDF データのためのデータモデリング語彙	1.1 は 2014-02-25	[63]
OWL	形式的に定義された意味を持つオントロジー言語	2004-02-10、OWL 2 第 2 版は 2012-12-11	[5], [13]
SKOS	知識組織化体系を共有しリンクする共通データモデル	2009-08-18	[41]
SPARQL	RDF の問い合わせ言語	2008-01-15、1.1 は 2013-03-21	[40], [64]
SHACL	RDF グラフを条件の集合に照らして検証する言語	2017-07-20	[48]
JSON-LD	Linked Data を直列化する JSON ベースの形式	1.0 は 2014-01-16、1.1 は 2020-07-16	[45], [49]
DCAT	データカタログ間の相互運用性のための RDF 語彙	2014-01-16、第 2 版は 2020-02-04、第 3 版は 2024-08-22	[44], [55], [65]

考察

これらは役割で分けると理解しやすいと考えられる。RDF は「書き方」、RDF Schema と OWL は「語彙と論理」、SKOS は「用語の階層」、SHACL は「検査」、SPARQL は「問い合わせ」、JSON-LD は「開発者になじむ直列化」、DCAT は「データの目録」である。

6.3.1 RDF

概要

Web 上で情報を表現するための枠組みである [47]。1999 年の仕様は、RDF をメタデータ処理の基盤と位置づけた [35]。

解く問題

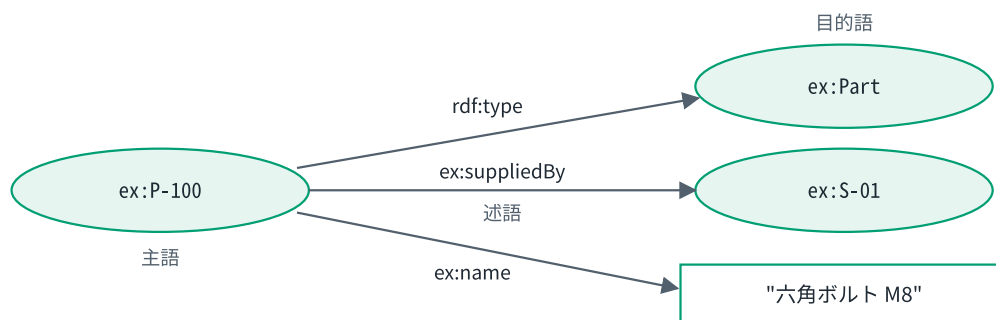
出どころの違う情報を、変換の取り決めなしに一つに合わせる。そのために、すべてを同じ 3 つ組の形で書き、識別子を Web 全体で一意にする。

具体例

```
@prefix ex: <https://example.org/> .
@prefix xsd: <http://www.w3.org/2001/XMLSchema#> .

ex:P-100 a          ex:Part ;
        ex:name      "六角ボルト M8" ;
        ex:weightGram 12 ;
        ex:suppliedBy ex:S-01 .

ex:S-01 a          ex:Supplier ;
        ex:name      "東和精工" .
```



すべての情報を「主語、述語、目的語」の3つ組で表す。楕円は識別子、四角は値。

図 18: RDF のトリプルの例。

似たものとの違い

JSON や XML は入れ子の木の形でデータを持つ。RDF は3つ組の集まりであり、順序や入れ子に意味がない。そのため、別々に書かれた RDF をそのまま足し合わせられる。

使われている場所

Gaia-X のクレデンシャルはクレームを RDF で表現する [66]。日本の ODS のメタデータエクステンション仕様は、メタデータを RDF で表現することを求める [26]。

限界と注意点

RDF は書き方を決めるだけである。どの述語を使うかという語彙は、別に決めなければならない。

6.3.2 RDF Schema

概要

RDF データのためのデータモデリング語彙を提供する [63]。

解く問題

RDF で使うクラスとプロパティについて、上下関係と、主語や目的語になるものの種類を宣言する。

具体例

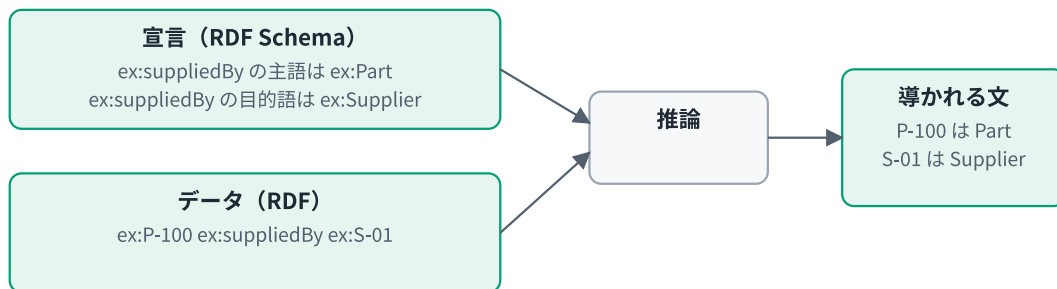
```

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

ex:Part        a rdfs:Class ; rdfs:subClassOf ex:Product .
ex:Supplier    a rdfs:Class ; rdfs:subClassOf ex:Organization .

ex:suppliedBy  a <http://www.w3.org/1999/02/22-rdf-syntax-ns#Property> ;
               rdfs:label    "供給者"@ja ;
               rdfs:domain   ex:Part ;
               rdfs:range    ex:Supplier .

```



宣言はデータを拒否するためではなく、書かれていない事実を補うために使われる。

図 19: RDF Schema による推論。宣言から、書かれていない型を補う。

似たものとの違い

データベースのスキーマは、合わないデータを拒否する。RDF Schema の宣言は拒否のためのものではなく、推論の手がかりである。上の例では、`ex:suppliedBy` の主語になったものは部品だと推論される。

使われている場所

IDS インフォメーションモデルは RDFS と OWL で書かれたオントロジーである [67]。

限界と注意点

データの誤りを検査する用途には向かない。検査には SHACL を使う。

6.3.3 OWL

概要

形式的に定義された意味を持つ、セマンティックウェブのためのオントロジー言語である [13]。情報を人に見せるだけでなく、その内容を処理するアプリケーションのために設計された [5]。

解く問題

「供給者を持つものを調達品と呼ぶ」のような定義を、計算機が推論に使える形で書けるようにする。

具体例

```

@prefix ex:    <https://example.org/> .
@prefix owl: <http://www.w3.org/2002/07/owl#> .
@prefix rdfs:  <http://www.w3.org/2000/01/rdf-schema#> .

ex:suppliedBy a owl:ObjectProperty ;
              owl:inverseOf ex:supplies .

# 「供給者が 1 つ以上あるもの」を「調達品」と定義する
ex:ProcuredItem a owl:Class ;
  owl:equivalentClass [
    a owl:Restriction ;
    owl:onProperty      ex:suppliedBy ;
    owl:someValuesFrom  ex:Supplier
  ] .

# 部品と供給者は重ならない
ex:Part owl:disjointWith ex:Supplier .

```



図 20: RDF Schema と OWL の例。

似たものとの違い

SHACL との違いは、前提にある。OWL は、書かれていないことを「偽」ではなく「未知」と扱う。この考え方を開世界仮説と呼ぶ [68]。供給者の記載がない部品を見ても、OWL は誤りとは判定しない。

使われている場所

金融業界の FIBO は OWL で開発されている [69]。

限界と注意点

OWL で「部品には供給者が必ずある」と書いても、供給者のない部品のデータは誤りとして検出されない。

6.3.4 SPARQL

概要

RDF のための問い合わせ言語である [40]。SPARQL 1.1 は、RDF グラフの内容を問い合わせ、操作するための仕様群である [64]。

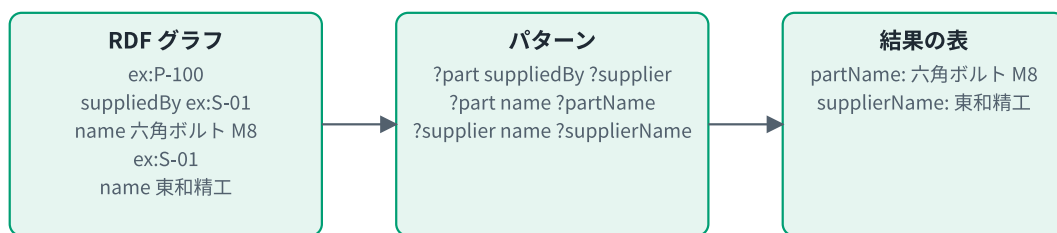
解く問題

3 つ組の集まりから、条件に合う部分を取り出す。

具体例

```
PREFIX ex: <https://example.org/>

SELECT ?partName ?supplierName
WHERE {
  ?part      a          ex:Part ;
            ex:name     ?partName ;
            ex:suppliedBy ?supplier .
  ?supplier  ex:name     ?supplierName .
}
```



「?」で始まる変数を含むパターンを書き、グラフの中で当てはまる部分を取り出す。

図 21: SPARQL の問い合わせの流れ。

似たものとの違い

SQL が表の行を対象にするのに対し、SPARQL はグラフのパターンを対象にする。

使われている場所

日本の ODS のメタデータエクスチェンジ仕様は、メタデータの取得手段の一つに SPARQL を挙げる [26]。

限界と注意点

問い合わせを書くには、対象のデータがどの語彙で書かれているかを知っている必要がある。

6.3.5 SKOS

概要

知識組織化体系を Web 上で共有し、リンクするための共通データモデルである [41]。

解く問題

分類表や用語集のような、用語の階層と言い換えを、機械が読める形で共有する。

具体例

```
@prefix ex:    <https://example.org/term/> .
@prefix skos:  <http://www.w3.org/2004/02/skos/core#> .

ex:fastener    a skos:Concept ;
               skos:prefLabel "締結部品"@ja , "fastener"@en .
```

```

ex:bolt      a skos:Concept ;
             skos:prefLabel "ボルト"@ja , "bolt"@en ;
             skos:altLabel  "ボルト類"@ja ;
             skos:broader   ex:fastener .

```

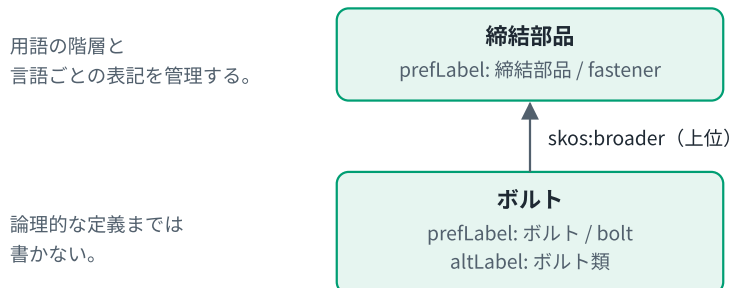


図 22: SKOS の例。用語の上下関係と表記を持つ。

似たものとの違い

OWL のクラスの階層は論理的な包含関係を表す。SKOS の上位と下位は、用語の広い狭いというゆるやかな関係である。

限界と注意点

用語の整理には手軽だが、推論の根拠となる厳密な定義は書けない。

6.3.6 SHACL

概要

RDF グラフを、条件の集合に照らして検証するための言語である [48]。

解く問題

受け取ったデータが、必須の項目を備え、値の型や範囲が正しいかを機械的に検査する。

具体例

次のシェイプは、部品に名称と供給者が必ずあり、重量が 0 より大きいことを求める。

```

@prefix ex:  <https://example.org/> .
@prefix sh:  <http://www.w3.org/ns/shacl#> .
@prefix xsd: <http://www.w3.org/2001/XMLSchema#> .

ex:PartShape a sh:NodeShape ;
  sh:targetClass ex:Part ;
  sh:property [
    sh:path      ex:name ;
    sh:datatype  xsd:string ;
    sh:minCount  1
  ] ;
  sh:property [
    sh:path      ex:suppliedBy ;
    sh:class     ex:Supplier ;
    sh:minCount  1
  ] .

```

```

] ;
sh:property [
  sh:path          ex:weightGram ;
  sh:minExclusive  0
] .

```

次のデータは、供給者がなく重量が負なので、2 件の違反として報告される。

```

@prefix ex: <https://example.org/> .

# 供給者がなく、重量が負の部品
ex:P-200 a          ex:Part ;
         ex:name      "平ワッシャー M8" ;
         ex:weightGram -1 .

```

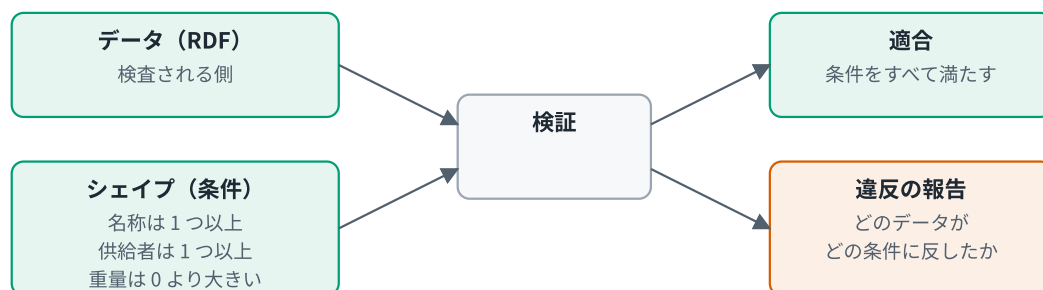


図 23: SHACL による検証の流れ。

似たものとの違い

OWL は書かれていないことを未知と扱う。SHACL は、手元のデータに書かれていなければ違反と判定する。

使われている場所

Gaia-X はクレデンシャルを SHACL のシェイプに照らして検証する [66]。Catena-X が使う SAMM は、SHACL の検証規則を伴う [70]。

限界と注意点

検査できるのは形と値の条件である。値が現実と合っているかは検査できない。

6.3.7 JSON-LD

概要

Linked Data を直列化するための、JSON ベースの形式である [45]。

解く問題

開発者が慣れた JSON の形のまま、RDF として解釈できるデータを書けるようにする。

具体例

```

{
  "@context": {
    "ex": "https://example.org/",
    "name": "ex:name",
    "weightGram": "ex:weightGram",
    "suppliedBy": { "@id": "ex:suppliedBy", "@type": "@id" }
  },
  "@id": "ex:P-100",
  "@type": "ex:Part",
  "name": "六角ボルト M8",
  "weightGram": 12,
  "suppliedBy": "ex:S-01"
}

```

この文書は、RDF の項で示した部品の 4 つの文と同じ内容に解釈される。

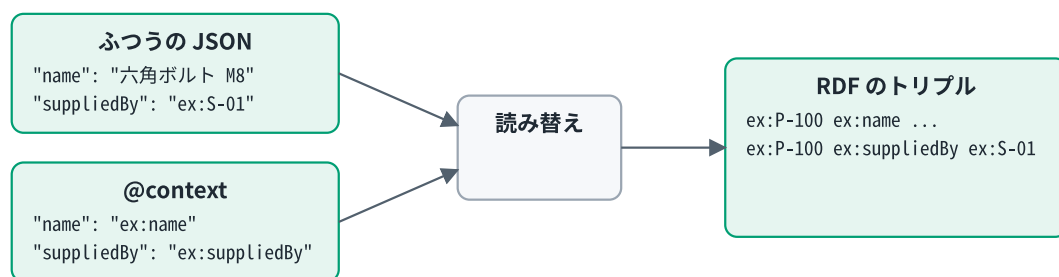


図 24: JSON-LD の仕組み。@context が JSON の項目名を識別子に読み替える。

似たものとの違い

ふつうの JSON では、`name` という項目名の意味は送り手と受け手の取り決めに任される。JSON-LD では @context が項目名を識別子に対応づける。

使われている場所

データスペースプロトコルは JSON-LD 1.1 を用いる [19]。日本の ODS のメタデータエクスチェンジ仕様は、メタデータを JSON-LD 形式で送受信することを原則とする [26]。

限界と注意点

数値の型には注意が要る。JSON の小数は倍精度の浮動小数点数として解釈され、Turtle で書いた小数とは別の型になる。本章の例は、この食い違いを避けるために重量を整数にした。

6.3.8 DCAT

概要

Web 上で公開されるデータカタログの間の相互運用性を高めるための RDF 語彙である [55]。

解く問題

どこにどんなデータがあり、どうすれば入手できるかを、カタログをまたいで同じ形で記述する。

具体例

```

@prefix ex:      <https://example.org/> .
@prefix dcat:    <http://www.w3.org/ns/dcat#> .
@prefix dcterms: <http://purl.org/dc/terms/> .

ex:catalog a dcat:Catalog ;
  dcterms:title "東和精工 データカタログ"@ja ;
  dcat:dataset ex:parts-master .

ex:parts-master a dcat:Dataset ;
  dcterms:title "部品マスタ"@ja ;
  dcterms:conformsTo ex:PartShape ;
  dcat:keyword "部品"@ja , "締結部品"@ja ;
  dcat:distribution ex:parts-master-csv .

ex:parts-master-csv a dcat:Distribution ;
  dcat:mediaType <https://www.iana.org/assignments/media-types/text/csv> ;
  dcat:accessURL <https://example.org/data/parts> .

```

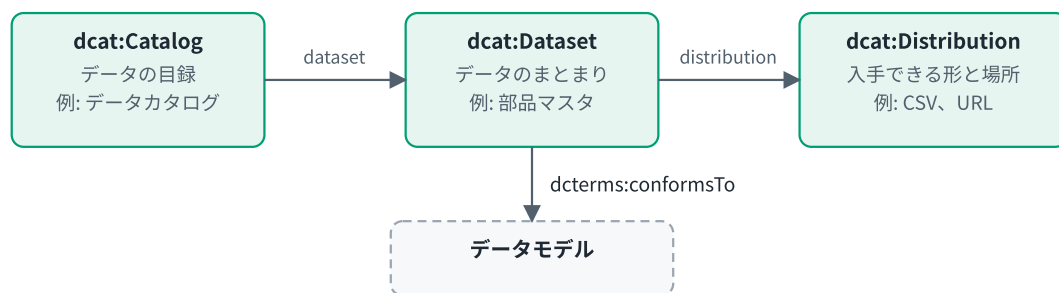


図 25: DCAT の主なクラス。

似たものとの違い

DCAT が記述するのは、データそのものではなくデータについての情報である。データの中身の意味は、別のデータモデルが担う。DSSC のブループリントは、データセットが `dcterms:conformsTo` でデータモデルを参照できるとしている [23]。

使われている場所

データスペースプロトコルのカタログは DCAT の語彙を再利用する [8]。欧州の DCAT-AP は DCAT のプロファイルである [71]。

限界と注意点

DCAT で目録を共有しても、データの中身の解釈が一致するとは限らない。

6.3.9 XML Schema と JSON Schema

概要

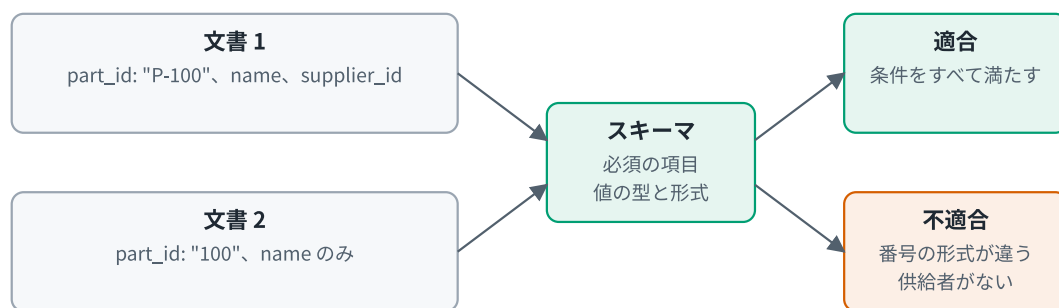
XML Schema は、XML 文書の構造を記述し、内容を制約する言語である [38]。JSON Schema は、JSON データの構造と制約を定義する宣言的な言語である [72]。

解く問題

受け取った文書が、決められた項目と型を備えているか进行检查する。

具体例

```
{
  "$schema": "https://json-schema.org/draft/2020-12/schema",
  "type": "object",
  "required": ["part_id", "name", "supplier_id"],
  "properties": {
    "part_id": { "type": "string", "pattern": "^P-[0-9]+$" },
    "name": { "type": "string" },
    "weight_g": { "type": "number", "exclusiveMinimum": 0 },
    "supplier_id": { "type": "string" }
  }
}
```



項目と型がそろっているかは検査できるが、項目の意味は扱わない。

図 26: スキーマによる文書の検査。

似たものとの違い

SHACL がグラフを検査するのに対し、これらは木の形の文書を検査する。項目の意味は扱わない。

使われている場所

データスペースプロトコルのメッセージは、JSON Schema で規範的に定義される [19]。

限界と注意点

形が合っていても、送り手と受け手が項目の意味を同じに解釈している保証はない。

6.3.10 schema.org

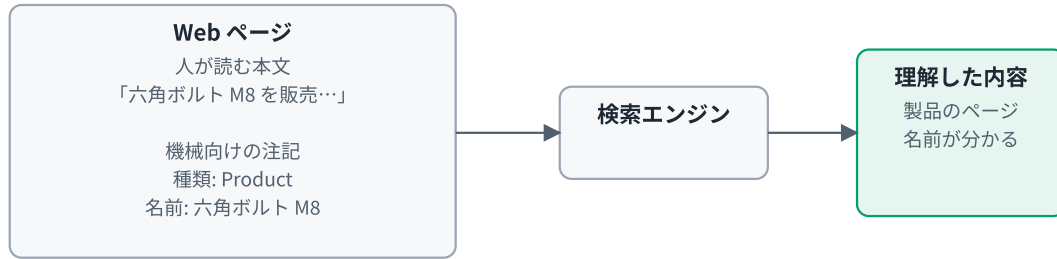
概要

Web ページの構造化データのマークアップに使う、共通のスキーマ群である。2011 年に Google、Bing、Yahoo! の取り組みとして発表された [42]。

解く問題

Web ページに書かれた事物の種類と属性を、検索エンジンが共通の語彙で読み取れるようにする。

具体例



ページに共通の語彙で注記を添えると、検索エンジンが内容の種類と属性を読み取れる。

図 27: schema.org による構造化データの読み取り。

限界と注意点

W3C の勧告ではなく、検索エンジン各社が始めた語彙である。本調査では、現在の採用状況は確認していない。

6.4 分析系

6.4.1 セマンティックレイヤ

概要

データウェアハウスと、それを使うツールの間に位置するソフトウェア層である。これはベンダの用語集による説明である [24]。Google は Looker を、指標を一度定義してどこでも使えるセマンティックモデルを中核とする設計だと説明している [73]。

解く問題

「売上」や「部品点数」のような指標の計算方法が、部署やツールごとに食い違うことを防ぐ。

i 補足 : 指標とディメンションとは

分析の世界で使う言葉である。データを集計するときの「何を数えるか」と「どの単位で分けるか」にあたる。

- **指標** (メトリクス、メジャーとも呼ぶ) は、合計、平均、比率のように、データを集計して得る数値である [60]。例は、部品点数、平均重量、売上である。
- **ディメンション** は、集計を分ける切り口である。行ごとの属性のうち、グループ分けや絞り込みに使うものを指す [60]。例は、供給者、月、地域である。
- **結合** は、別々の表をどの列でつなぐかの取り決めである。

「供給者ごとの部品点数」という集計では、部品点数が指標、供給者がディメンションにあたる。部品の表と供給者の表を、供給者番号で結合して数える。

```
-- 指標: 部品点数(部品の数を数える)
-- ディメンション: 供給者の名称(この単位で分けて数える)
SELECT supplier.name AS supplier_name,
       COUNT(DISTINCT part.part_id) AS part_count
FROM part JOIN supplier ON part.supplier_id = supplier.supplier_id
GROUP BY supplier.name;
```

指標の定義は、書く人によって食い違いやすい。「部品点数」に廃番の部品を含めるのか、同じ部

品を二重に数えていないか、といった判断が、ツールや部署ごとに別々に行われるためである。セマンティックレイヤは、この定義を一か所を書いて共有する。

具体例

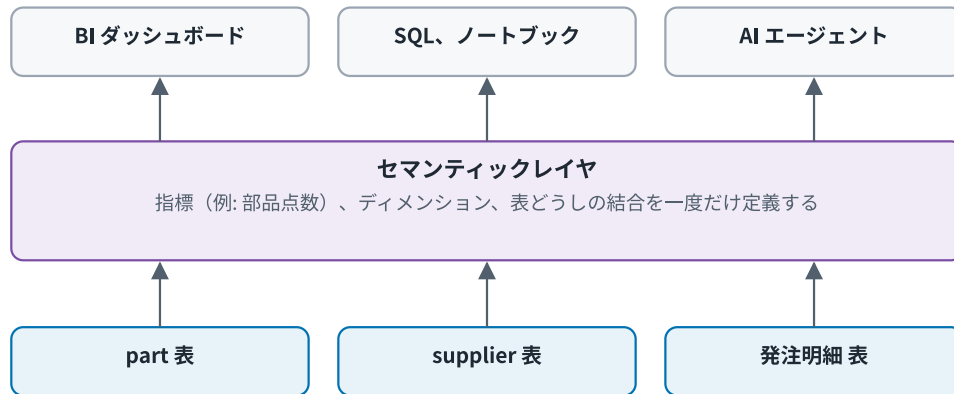


図 28: セマンティックレイヤの位置。

似たものとの違い

オントロジーが概念の意味を定義するのに対し、セマンティックレイヤは集計の計算方法を定義する。同じ「セマンティック」という語を使うが、対象が違う。

使われている場所

Databricks の Unity Catalog Business Semantics、dbt Labs の MetricFlow、Google の Looker がある [11], [56], [73]。Microsoft は Power BI のデータセットをセマンティックモデルに改称した [51]。

限界と注意点

定義の形式は製品ごとに異なる。この違いを埋めようとするのが、次の Apache Ossie である。

6.4.2 OSI と Apache Ossie

概要

セマンティックモデルの定義を、ツールの間で交換するための仕様である。Open Semantic Interchange (OSI) として 2025 年に発表され、現在は Apache Ossie (Incubating) という名称である [16], [17]。仕様は JSON と YAML にもとづき、1つの文書が1つのセマンティックモデルを表す [60], [74]。

解く問題

ある製品で定義した指標を、別の製品や AI エージェントでも同じ定義のまま使えるようにする。

具体例

```
version: 0.2.0.dev0
name: procurement
description: 部品の調達に関する分析モデル

datasets:
  - name: part
    source: erp.public.part
```

```

primary_key: [part_id]
fields:
  - name: part_id
    expression:
      dialects:
        - dialect: ANSI_SQL
          expression: part_id
  - name: weight_g
    expression:
      dialects:
        - dialect: ANSI_SQL
          expression: weight_g
- name: supplier
  source: erp.public.supplier
  primary_key: [supplier_id]
  fields:
    - name: supplier_name
      expression:
        dialects:
          - dialect: ANSI_SQL
            expression: name

relationships:
  - name: part_to_supplier
    from: part
    to: supplier
    from_columns: [supplier_id]
    to_columns: [supplier_id]

metrics:
  - name: part_count
    description: 部品の点数
    expression:
      dialects:
        - dialect: ANSI_SQL
          expression: COUNT(DISTINCT part.part_id)
    ai_context:
      synonyms:
        - "部品点数"
        - "品目数"

```

この例は、仕様のリポジトリにある JSON Schema に適合することを確認した。対象とした仕様の版は 0.2.0.dev0 である [60]。



各ツールが独自形式で持つ定義を、共通の形式に書き出して受け渡す。

図 29: Apache Ossie による定義の交換。

似たものとの違い

W3C の標準群が組織をまたぐ意味の共有を狙うのに対し、Ossie は分析ツールの間で指標の定義を持ち運ぶことを狙う。

リポジトリには、セマンティックモデルの仕様とは別に、オントロジーの仕様もある。同仕様はオントロジーを、概念、関係、業務規則によって企業を記述する概念モデルと説明する [75]。OWL とは別の、独自の YAML の書式である。

使われている場所

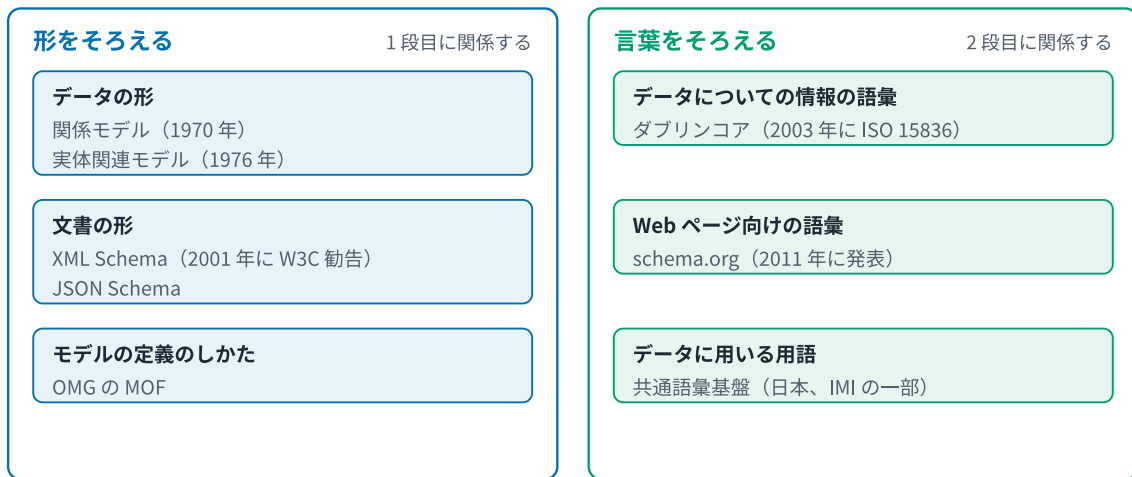
経過と参加企業は セクション 9.4 にまとめた。

限界と注意点

仕様の版番号は開発版を示しており、Apache のインキュベーション段階にある。本調査では、各製品での実装状況は確認していない。

7 データモデルとメタデータの標準化

この章の標準は、そろえる対象で二つに分けられる。データや文書の形をそろえるものと、項目の名前に使う言葉をそろえるものである。



左は項目の並びや型を決める。右は項目の名前が何を指すかを定める。
左右の分け方と、意味の 4 段階との対応は筆者の整理である。

図 30: データモデルとメタデータの標準が、何をそろえるか。左右の分け方と、意味の 4 段階との対応は筆者の整理である。各標準の出典は本文に示す。

関係モデルは 1970 年、実体関連モデルは 1976 年に論文として発表された [30], [31]。モデルそのものを定義する側では、OMG の MOF が、モデル駆動アーキテクチャの言語群におけるメタモデル定義の基礎となっている [76]。

文書の構造を記述する言語には、XML Schema と JSON Schema がある。XML Schema は 2001 年 5 月 2 日に W3C 勧告となった [38]。JSON Schema は、JSON データの構造と制約を定義する宣言的な言語である [72]。

メタデータの語彙としては、ダブリンコアがある。その歴史は、Web 資源のための中核的な意味の集合を議論した OCLC/NCSA メタデータワークショップに始まる [77]。ISO 15836 として 2003 年に初めて発行され、最後の改訂は 2017 年の ISO 15836-1:2017 である [78]。

Web ページ向けの語彙である schema.org は、2011 年 6 月 2 日付の記事で、Google、Bing、Yahoo! の取り組みとして発表された [42]。公式ページは、創設企業として Google、Microsoft、Yahoo、Yandex を挙げている [79]。

日本では、IMI (情報共有基盤) の一部である共通語彙基盤が、データに用いる用語の表記・意味・構造の統一を掲げている [80]。

8 ドメインオントロジーと業界標準

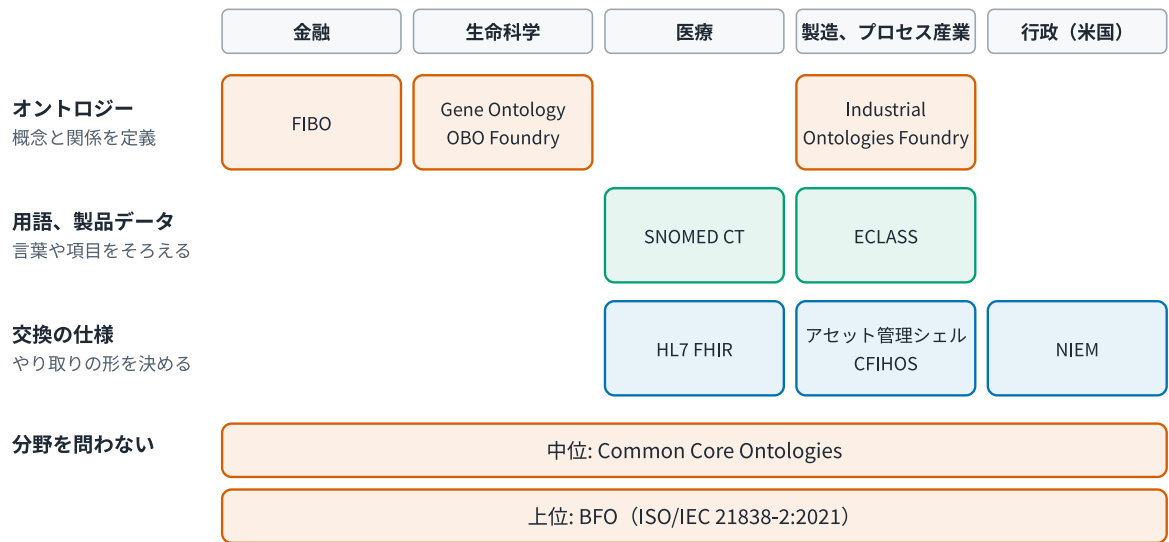
表 12: ドメインオントロジーと業界標準

分野	名称	公式サイト等による説明	出典
金融	FIBO	OWL で開発されたオントロジー。OMG でも標準化	[69]
生命科学	Gene Ontology	生物学的知識の構造化され標準化された表現	[36], [81]

分野	名称	公式サイト等による説明	出典
生命科学	OBO Foundry	生物学のための相互運用可能なオントロジーのコミュニティ開発	[82]
医療	SNOMED CT	多言語の臨床医療用語集	[83]
医療	HL7 FHIR	HL7 が発行する医療データ交換の標準	[84]
プロセス産業	CFIHOS	産業施設の情報引き渡しのための標準化された仕様	[85]
製造	アセット管理シェル (AAS)	各アセットをデジタルに表現する。IDTA が推進し、構造は IEC 63278-1:2023 が定義する	[6], [86]
製造	ECLASS	製品マスタデータを業種・国・言語・組織を越えて交換可能にする	[87]
製造	Industrial Ontologies Foundry	デジタル製造の全領域にわたるオープンな参照オントロジー群の作成	[88]
上位	BFO	ISO/IEC 21838-2:2021 に収められた上位オントロジー	[50]
中位	Common Core Ontologies	あらゆる領域に共通する汎用的な用語と関係を定義する 11 のオントロジー群 (2026-10-02 時点の README の記載)	[62]
行政(米国)	NIEM	公共部門と民間部門の組織間で情報を交換する枠組み	[89]

表の説明は各団体の自己記述にもとづくもので、中立な評価ではない。

表 12 の標準は、どれもオントロジーというわけではない。概念と関係を定義するもの、言葉や項目をそろえるもの、やり取りの形を決めるものが混ざっている。分野と種類で並べ直すと図 31 のようになる。



種類の分け方は、各団体の自己記述をもとにした筆者の整理である。
SNOMED CT は概念と関係も持ち、オントロジーの行にもまたがる。

図 31: ドメインオントロジーと業界標準の見取り図。横は分野、縦は標準が主に定めるものの種類を表す。種類の分け方は、表 12 の自己記述をもとにした筆者の整理である。

この配置は、自己記述の文言だけから決めたものである。たとえば SNOMED CT は臨床の用語集と説明されるが、中核の要素に概念と関係を持ち、オントロジーの行にもまたがる [83]。各標準の中身を比べた結果ではなく、種類の境目もはっきりしたものではない。上位と中位のオントロジーの関係は図 16 に示した。

9 米国企業の近年の取り組み

この章では、米国企業の動きを五つに分けて見る。以下、表の順に説明する。

表 13: 米国企業の動きの見取り（筆者の整理）

順	動き	代表	狙い	意味の 4 段階との関係	節
1	知識グラフ	Google のナレッジグラフ、Wikidata	事物とその関係を、グラフで持つ	データモデルによる。RDF で書けば 2 段階目の仕組みを使う	セクション 9.1
2	製品のオントロジー	Palantir	データを業務の操作と結ぶ	またがる。1 段階目から 3 段階目の軸にある	セクション 9.2
3	セマンティッククレイヤ	Looker、Power BI、dbt	指標の定義を一か所に置く	別の軸から、意味の定義へ広がる	セクション 9.3
4	交換のための仕様	OSI、Apache Ossie	定義をツールの間でそろえる	またがる。下の段を一部ずつ備える	セクション 9.4
5	文脈を渡す形式	Open Knowledge Format	知識を文章のまま AI に渡す		セクション 9.5

3 から 5 は、分析系の流れに属する。1 と 2 は、四本の流れのどれか一つには収まらない。知識グラフは知識表現系と Web 標準系の両方に関わり、製品のオントロジーは四本の外にある(表 26)。「意味の 4 段階との関係」の列は筆者の整理である。

9.1 知識グラフ

Google は 2012 年 5 月 16 日付の記事でナレッジグラフを紹介し、「things, not strings」(文字列ではなく事物)という表現を用いた [43]。Wikidata は自らを、自由で協働的な多言語の二次的な知識ベースと説明している [90]。

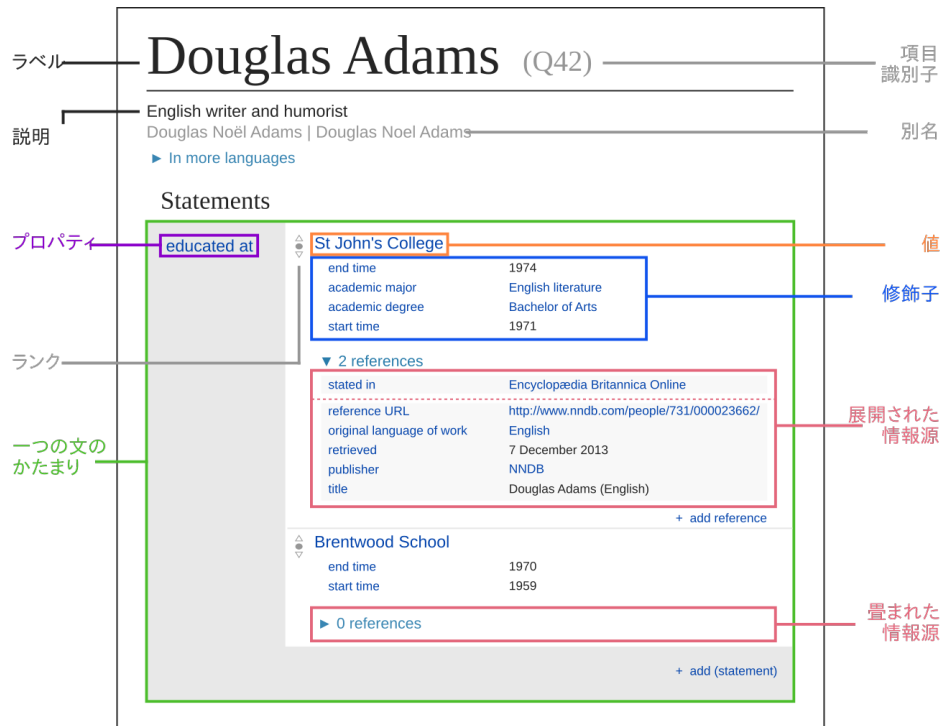


図 32: Wikidata のデータモデル。一つの項目が、プロパティと値の組からなる文のかたまりを持ち、それぞれに修飾子と情報源が付く。出典: Charlie Kritschmar (WMDE), Datamodel in Wikidata, Wikimedia Commons, CC0 [91]。

図 32 は知識グラフの一つの例である。「ダグラス・アダムズは St John's College で教育を受けた」という事実が、期間などの修飾子と、根拠となる情報源とともに記録されている。

グラフのデータモデルの代表的なものに、RDF とプロパティグラフがある [25]。後者については、ISO/IEC 39075:2024 (GQL) がデータ構造と基本操作を定義する [54]。Neo4j は、プロパティグラフはデータ交換形式ではなくデータベースモデルとして設計されたと説明しているが、これは製品ベンダ自身の説明である [61]。

言語モデルとの組み合わせでは、Microsoft Research が 2024 年 2 月 13 日付のブログ記事で GraphRAG を紹介した [53]。

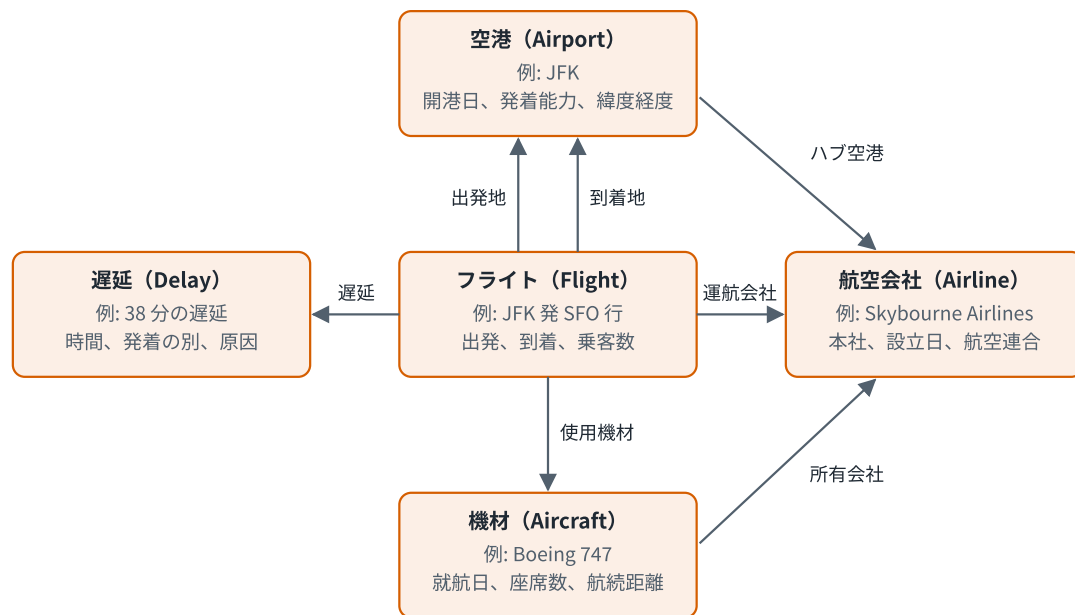
9.2 製品としてのオントロジー：Palantir

Palantir は自社の Ontology を次のように定義している [15]。

“The Palantir Ontology is an operational layer for the organization.”

(訳：Palantir Ontology は、組織のための運用レイヤである。)

同社の説明では、Ontology は意味要素(semantic elements :オブジェクト、プロパティ、リンク)と動的要素(kinetic elements :アクション、関数、動的セキュリティ)の両方を含む [15]。



箱がオブジェクト型、矢印がリンク型である。各箱の 3 行目はプロパティの例。

図 33: Palantir の Ontology の例。航空業界の 5 つのオブジェクト型と、それらを結ぶリンク型を示す。Palantir の公式文書の図 [92] をもとに筆者が作成した。

図 33 は意味要素の例である。空港、フライト、航空会社といったオブジェクト型が、それぞれプロパティを持ち、リンク型で結ばれている。

公式文書をもとに、Palantir の Ontology が セクション 4.4 の意味の 4 段階のどこまでを備えるかを整理すると、次のようになる。

表 14: Palantir の Ontology と意味の 4 段階の対応

段	Palantir の Ontology での対応	判定	出典
1 段目(形を検査できる)	必須プロパティは、値を持たなければならないプロパティである。値型は、検証の制約を伴う。主キーは一意でなければならない	備える	[93], [94], [95]
2 段目(同じものだと分かる)	既存のデータ源を、共通のオブジェクト、プロパティ、リンクに対応づける。一つのオブジェクト型は複数のデータソースを背後に持てる。ただし突き合わせは主キーの一致による	備える。一つの基盤の中に限る	[15], [96]

段	Palantir の Ontology での対応	判定	出典
3 段目(用語の関係をたどれる)	インターフェースは別のインターフェースを拡張でき、子はプロパティを継承する。これは型の継承であり、用語の上位と下位や言い換えを扱う仕組みではない	一部。型の継承はある	[97]
4 段目(定義から導ける)	派生プロパティは、他のプロパティやリンクの値から実行時に計算される。関数は TypeScript か Python で書く。定義からの論理的な推論にあたる記述は見当たらない	見当たらない	[98], [99]

「見当たらない」は、Ontology 関連の公式文書およそ 180 ページを検索した結果である。存在しないことの証明ではない。

W3C の標準との関係について、公式文書には次の記述がある。

- Foundry のデータ型は、RDF、OWL、XSD の類似の概念に着想を得ている [100]。調査した範囲で RDF と OWL に触れるのはこの一文だけで、対象はデータ型に限られる。
- Ontology のスキーマ定義は JSON ファイルに保存され、書き出しと取り込みができる。ただし、書き出した JSON のスキーマは変わりうるので依存すべきでないとしている [101]。
- 共有プロパティが共有するのはメタデータであり、背後のデータではない [102]。

Palantir は、自社の Ontology はセマンティックレイヤではないとも述べている [103]。

考察

Palantir の Ontology は、意味の 4 段階のうち 1 段目と 2 段目を、一つの基盤の中で、独自の形式で実現している。3 段目は型の継承という形で一部を備える。4 段目にあたる推論は、宣言された定義から導くのではなく、コードで書いた計算として実現している。

したがって同社の用法は、OWL のような論理的な推論よりも、データと業務上の操作を結びつけることを重視していると考えられる。

標準化された交換形式を持たない点は、組織をまたぐ意味の共有という観点では制約になると考えられる。書き出しの形式が安定していないことは、公式文書が自ら述べている。

9.3 分析系のセマンティックレイヤ

指標やディメンションという言葉になじみのない読者は、先にセクション 6.4.1 の補足を読むとよい。

Google は Looker を、指標を一度定義してどこでも使えるセマンティックモデルを中核とする設計だと説明している [73]。Microsoft は Power BI の「データセット」を「セマンティックモデル」に改称した [51]。時期は 2023 年 11 月である [52]。時期の出典は二次情報である。Databricks は 2026 年 4 月 2 日付の記事で、Unity Catalog Business Semantics の一般提供とオープンソース化を告知した [11]。dbt Labs は MetricFlow を Apache 2.0 ライセンスでオープンソース化すると発表した [56]。

i コラム：分析系はなぜ別の流れとして現れたのか(考察)

意味を扱う技術には、RDF や OWL という標準がすでにある。それなのに、分析系のセマンティクレイヤは別の流れとして育った。

一つの仮説はこうである。既存の手法や道具は、高度すぎるか複雑すぎた。一方、データ基盤の事業者が必要だったのは、分析ツールや BI から安定して使ってもらうことだった。それに足る機能が備わっていれば十分だった。

追加の調査で、この仮説の後半は裏づけられた。前半は、直接の裏づけが見つからなかった。順序にも修正が要る。

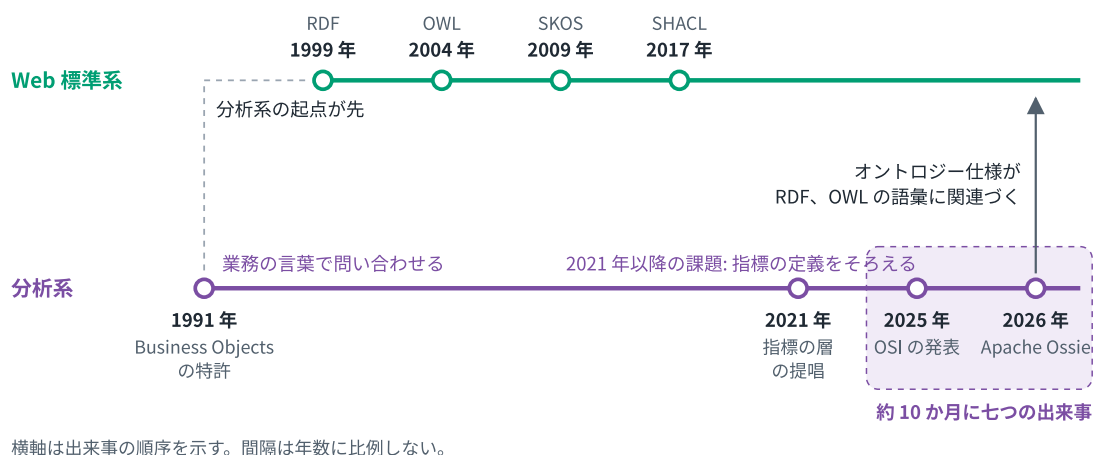


図 34: 分析系と Web 標準系の順序と接近。分析系の起点は Web 標準系より古く、近年になって両者が近づいている。出来事と年は表 9 による。筆者の整理である。

分析系の起点は、Web 標準系より古い

分析系の源流は、RDF より前にさかのぼれる。Business Objects の特許は 1991 年に出願され、利用者が関係構造や SQL を知らなくても、なじみのある業務の言葉で関係データベースに問い合わせられるようにする技術を記述している [34]。特許は「セマンティクレイヤ」という語を使っていない。これを分析系の起点と見るのは、筆者の位置づけである。RDF の最初の仕様は 1999 年である [35]。分析系は、既存のセマンティック技術の代わりとして作られたのではなく、別の課題から先に始まっていた。

2021 年以降の課題は、指標の定義をそろえることである

2021 年には、指標の計算方法をツールごとに定義するのをやめ、一か所で決めるという考え方が提唱された [104]。dbt Labs は、指標の定義を BI の層からモデリングの層へ移せば、どのツールでも同じ定義を使えると説明している [105]。OSI の発表文は、AI と BI のアプリケーションで一貫した業務ロジックを確保することを狙いに挙げた [16]。Apache Ossie の README も、同じ KPI がツールごとに違う定義になっていることを問題に挙げる [74]。

どれも当事者の説明だが、求めているものは共通している。指標とディメンションと結合の定義であり、定義からの推論ではない。仮説の後半は、これらの説明と合う。

複雑すぎたから避けた、という直接の証拠はない

分析系の事業者や Ossie の文書の中に、RDF や OWL を採らない理由を述べたものは見つからなかった。

一般論としては、複雑さを指摘する議論がある。Hogan は、セマンティックウェブの標準は長く複雑で理解しにくく、新しい利用者を引きつける障壁になっているという批判を紹介している。ただし同じ箇所、SQL の標準も大部だが中核は広く使われている、という反論も併記している [106]。複雑さだけで説明できるとは、この論文も結論していない。

実態としては、指標のための語彙やオントロジーは存在するが、セマンティクレイヤの界限で

は使われていないようだ、という署名記事の観察がある [107]。

いまは近づいている

Apache Ossie のオントロジー仕様は、概念や関係を、既存の RDF や OWL の語彙の定義に関連づけられるようにしている [75]。ロードマップは、概念的な相互運用性が未解決だと認めている [108]。

まとめ

分析系は、複雑な既存技術を避けて生まれたというより、別の課題を、それに足る機能で解いてきたと見るほうが、証拠に合う。複雑さが普及の妨げになったという見方には一定の支持があるが、分析系の当事者がそれを理由に挙げた証拠は見つかっていない。

いま両者が近づいているのは、AI に意味を渡すには指標の定義だけでは足りなくなったためだと推測される。ツールごとに名前や構造が違う概念を対応づけるという課題は、Web 標準系が取り組んできた課題と重なる。

これをどう読むか

別の課題を解いてきた流れが、いま意味の定義も扱い始めた、と読むのがよいと考えられる。

動きは活発である。2025 年 9 月から 2026 年 7 月の約 10 か月に、七つの出来事が続いた(表 9、表 17)。OSI の発表、MetricFlow のオープンソース化、OSI 仕様の公開、Databricks の一般提供とオープンソース化、Open Knowledge Format の発表、Ossie の Apache Incubator 入り、Open Knowledge Format の 0.2 版である。それ以前の分析系の出来事は、年表では 1991 年、2003 年、2023 年の三つである。

成熟度は低い。Ossie はインキュベーションの段階にあり、Open Knowledge Format は 0.2 版である。それでも注目に値すると筆者が考える理由は、三つある。

- ・ **誰が動かしているか:** すでに広く使われているデータ基盤と BI の事業者である。競合する事業者が、同じ仕様のワーキンググループに参加している [58], [109]。
- ・ **どこへ向かっているか:** 指標の定義から、概念や文脈といった意味の定義へ、範囲を広げている(セクション 9.4、セクション 9.5)。これは Web 標準系が取り組んできた課題と重なる。
- ・ **何が後押ししているか:** 読み手に AI が加わったことである。各社の発表が、そろって AI を読み手に挙げている(セクション 4.1)。

ただし、言い過ぎには注意が要る。根拠はすべて当事者の発表であり、採用の実績は評価していない。仕様が一本にまとまるとも限らない。Ossie、Open Knowledge Format、各社の仕組みが並んでいる。

共通化の試みは、過去にもあった

定義をツールや事業者の間でそろえようとする試みは、今回が初めてではない。確認できたものを挙げる。

表 15: 過去の共通化の試み

時期	試み	内容	その後	出典
2003 年 3 月	OMG の CWM (Common Warehouse Metamodel) 1.1 版	データウェアハウスのメタデータを、共通のメタモデルにもとづいて XMI で交換する。 Hyperion Solutions、IBM、NCR、Oracle、Unisys などが開発に貢献した	OMG のページに挙がる正式な版は、この 1.1 版だけである	[39]
2000 年から 2012 年	JOLAP (JSR 69)	OLAP のデータとメタデータを、ベンダに依存しない形で扱う Java の API。CWM を設計の参照に挙げた	2012 年 4 月 16 日に、仕様の責任者の求めで取り下げられた	[110]
2014 年 1 月 16 日	W3C の RDF Data Cube 語彙	データセットのディメンション、属性、メジャーを RDF で書く。統計データの交換の標準である SDMX のモデルと互換である	W3C 勧告になった	[46]
2024 年 9 月 10 日	AtScale の SML (Semantic Modeling Language)	セマンティックモデルを書く言語を、Apache ライセンスのオープンソースとして公開すると発表した	AtScale は OSI のワーキンググループに参加した	[58], [111], [112]

ここから言えることは二つある。

一つは、事業者が集まって共通の仕様を作るだけでは、続くとは限らないことである。CWM と JOLAP には、当時の主要な事業者が加わっていた。それでも、CWM の正式な版は 2003 年のものにとどまり、JOLAP は取り下げられた。今回の動きも続かない可能性は、否定できない。

もう一つは、指標やディメンションのための語彙が、Web 標準系にもすでにあることである。RDF Data Cube 語彙は 2014 年に勧告になっている。指標のための語彙は存在するがセマンティックレイヤの界限では使われていないようだ、という観察 [107] と合う。

今回が過去と違う点は、三つあると考えられる。

- **書き方:** CWM は、メタモデルを MOF で定義し、XMI で交換した。Ossie は JSON と YAML、Open Knowledge Format は Markdown である。道具をそろえなくても読み書きできる。
- **作る場:** 今回は、公開されたリポジトリで、オープンソースのライセンスのもとで仕様を作っている(セクション 9.4、セクション 9.5)。
- **読み手:** 当時の読み手は、ツールだった。今回は、読み手に AI が加わっている。

ただし、これらの違いが定着につながるかどうかは、まだ分らない。過去の試みがどれだけ使われたかも、本調査では確認していない。上の表の「その後」は、版や手続きの記録から言えることに限っている。

9.4 Open Semantic Interchange から Apache Ossie へ

表 16: OSI と Apache Ossie の経過

時点	出来事	出典
2025-09-23	Snowflake が Salesforce、BlackRock、dbt Labs、RelationalAI 等とともに OSI を発表	[16]
2026-01-27	仕様の最初の版を Apache 2 ライセンスのリポジトリで公開。Databricks、AtScale、Qlik 等がワーキンググループに参加	[58]
2026-04-28	コミュニティ更新で新たな参加者を列举	[109]
2026-06-22	Apache Incubator でインキュベーション開始(Apache 側の記録)	[17]
2026-07-10	プロジェクトがブログ記事で、Apache Incubator への受け入れと Apache Ossie (Incubating) への改称を公表	[1]

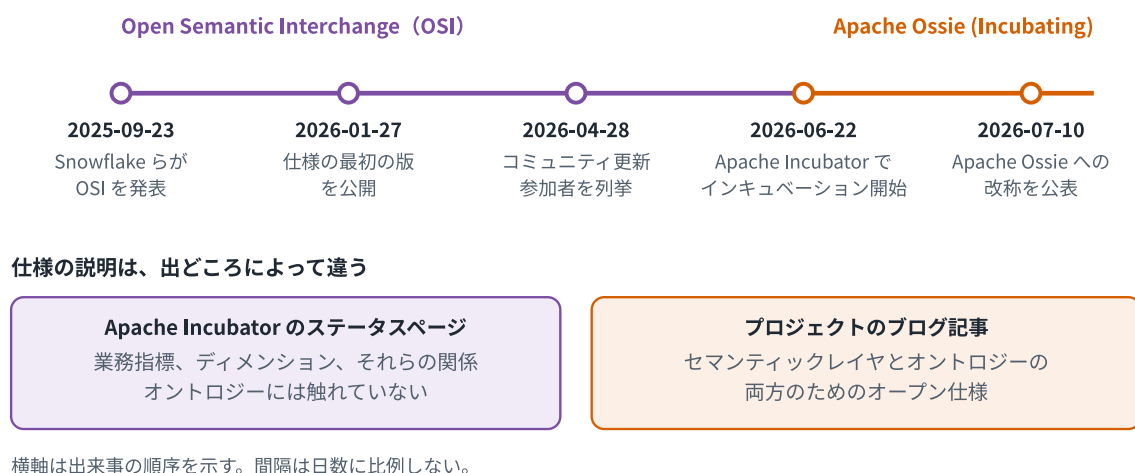


図 35: OSI から Apache Ossie への経過と、仕様の説明の違い。表 16 と本文の出典をもとに筆者が作成した。

2026 年 4 月の更新が列挙する参加者には、Oracle、Cloudera、Dremio が含まれる [109]。

プロジェクトのブログ記事は Ossie を次のように説明している [1]。

“Ossie is an open specification for both semantic layer and ontology.”

(訳：Ossie は、セマンティックレイヤとオントロジーの両方のためのオープン仕様である。)

一方、Apache Incubator のステータスページは、業務指標、ディメンション、それらの関係を表すベンダ中立の形式を定義するオープン仕様と説明しており、オントロジーには触れていない [17]。

リポジトリの README は、仕様を JSON と YAML にもとづくものと記している [74]。

ロードマップは、オントロジーを加える理由を説明している。現在の Ossie が解決するのは構造的な相互運用性であり、別々のモデルが同じ業務概念を違う名前や構造で書いている場合の、概念的な相互運用性はまだ解決していないという [108]。オントロジーで意味を定義する先例として、Palantir と Goldman Sachs Legend を挙げている [108]。

考察

OSI は、指標とディメンションの交換という分析系の課題から始まった。オントロジーは、改称より前から範囲に入っていた。2026 年 4 月の更新は、オントロジーの表現を扱うワーキンググループを挙げている [109]。7 月のブログ記事では、仕様の説明そのものに「ontology」が入った。分析系が意味の定義へ範囲を広げている兆候と考えられる。近づいている相手は二つある。ロードマップが先例として挙げるのは、Palantir のような製品のオントロジーである [108]。一方、オントロジー仕様は、概念や関係を既存の RDF や OWL の語彙に関連づけられるようにしている [75]。製品のオントロジーと Web 標準系の両方に近づいていると考えられる。ただし、Apache 側の公式な説明にこの語はない。プロジェクトの総意としての位置づけかどうかは確認できていない。仕様は Apache のインキュベーション段階にあり、成熟度や採用実績も本調査では評価していない。

9.5 言語モデルに文脈を渡す形式：Open Knowledge Format

Google Cloud は 2026 年 6 月 12 日付のブログ記事で、Open Knowledge Format (OKF) を発表した [18]。記事は OKF を次のように説明している。

“an open specification that formalizes the LLM-wiki pattern into a portable, interoperable format”

(訳：LLM ウィキのパターンを、持ち運べて相互運用できる形式として定めたオープン仕様)

記事の著者は、Google Cloud のデータ分析と BigQuery の技術リードである [18]。

表 17: OKF の経過

時点	出来事	出典
2026-06-12	0.1 版を発表。知識を、YAML の前書きつき Markdown ファイルのディレクトリで表す	[18]
2026-07-24	0.2 版を発表。来歴、信頼、ライフサイクル、検証を、前書きの項目として加えた	[113], [114]

仕様と README の要点は次のとおりである。

- 一つ概念を、一つの Markdown ファイルで表す。前書きで常に必須なのは `type` だけである [114]。
- `type` の値は、中央に登録されない。読む側は、知らない型を受け入れなければならない [114]。
- 概念の間は、通常の Markdown のリンクで結ぶ。関係の種類は、リンクそのものではなく、周りの文章が伝える [114]。
- 仕様は、概念の型の固定の分類を定めることを、目的としないものに挙げている [114]。
- 仕様は、組織をまたいで交換されることも想定している [114]。
- README は、OKF が表すものを「カタログの知識」と呼び、既存のデータカタログからの書き出しを作り手の一つに挙げている [115]。

0.2 版の記事は、0.1 版と同じく最小限の決まりだけを定める形式のままだと述べている [113]。同じ版で加わった「検証つきの計算」は、数値が決めたとおりに計算されたかを確認するための仕組みである。記事は、これが Looker や AtScale などのセマンティックモデルの呼び出しにも対応しようと述べる [113]。仕様は、今後の課題にセマンティックレイヤのテンプレートを挙げている [114]。

調査した範囲では、仕様、README、二つの記事のいずれにも、RDF、OWL、JSON-LD、SKOS への言及は見つからなかった。

考察

OKF は、セクション 5 の四本の流れのうち、分析系の延長にあると考えられる。著者はデータ分析の担当であり、対象はデータカタログの知識である。型の例にはテーブルや指標が並び、検証つきの計算は「どのツールでも同じ数字」という分析系の課題に答えるものである。

狙いは、知識表現系と重なる。知識表現系は、プログラムが持たない背景知識を書き下して渡そうとした(セクション 6.2.1)。OKF も、言語モデルが持たない組織の文脈を書き下して渡す。違うのは手段である。知識表現系は論理で書いた。OKF は文章で書き、解釈を読み手の言語モデルに任せる。

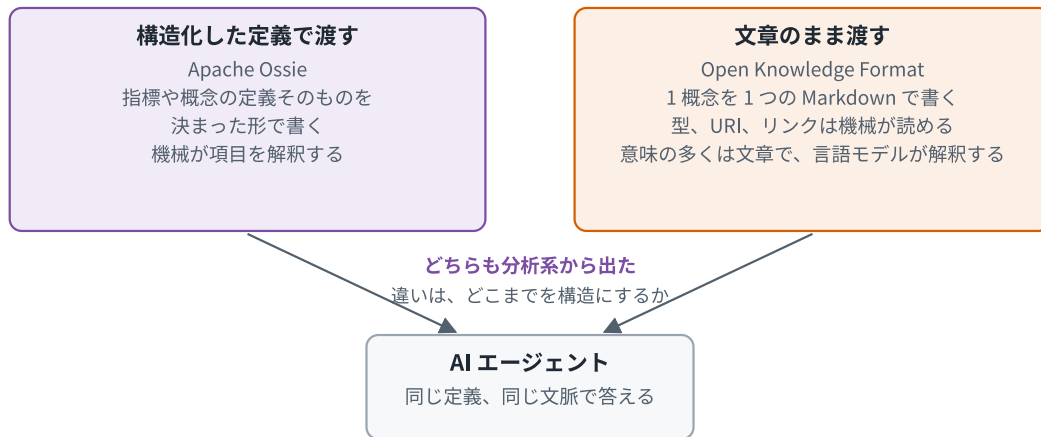
セクション 4.4 の意味の 4 段階に照らすと、OKF は下の段を一部ずつ備えている。Palantir の Ontology と同じく、段ごとに見ると次のようになる。

表 18: OKF と意味の 4 段階の対応(筆者の整理)

段	OKF での対応	判定	出典
1 段目(形を検査できる)	適合の条件は、前書きが YAML として読めることと、 <code>type</code> が空でないことである。0.2 版は、来歴や状態の項目も前書きに置く	備える。ただし最小限	[114]
2 段目(同じものだと分かる)	<code>resource</code> は、概念が記述する資産を一意に識別する URI である。抽象的な概念には付かない。 <code>type</code> の値は中央に登録されない	一部。資産を指す場合に限る。概念や用語を同定する仕組みはない	[114]
3 段目(用語の関係をたどれる)	リンクは関係があることを表し、ディレクトリは暗黙の親子関係を表す。ただし関係の種類は周りの文章が伝える。グラフを作る側は、種類のない有向の辺として扱うのが普通とされる	ごく一部。つながりはたどれるが、何の関係かは機械に分からない。広い語と狭い語のような用語の関係は表せない	[114]
4 段目(定義から導ける)	論理的な定義や推論にあたる記述は見当たらない。検証つきの計算は、数値の計算を確かめる仕組みであり、定義からの推論ではない	見当たらない	[114]

つまり OKF は、1 段目を最小限に備え、2 段目と 3 段目はごく一部にとどまる。資産は URI で同定でき、概念の間のつながりもたどれる。しかし、型の語彙は共有されず、何の関係でつながっているかは機械に分からない。意味の多くは本文の文章にあり、その解釈は読み手の言語モデルに任されている。

表 8 では、OKF を「またがる」に分類した。Palantir の Ontology と同じく、一つの段の代表的な技術ではなく、複数の段を一部ずつ備えるためである。



Ossie は 2025 年 9 月に OSI として発表、OKF は 2026 年 6 月に発表。
二つに分けて見るのは筆者の整理である。

図 36: AI に意味を渡す二つの方法。構造化した定義で渡す Apache Ossie と、文章のまま渡す Open Knowledge Format。二つに分けて見るのは筆者の整理である。

Apache Ossie と並べると、AI に意味を渡す方法は二つに分かれる。Ossie は、指標や概念を構造化した定義として渡す。OKF は、文章のまま渡す。どちらも分析系から出て、読み手に AI を挙げている。

ただし、二つの違いは、構造を持つか持たないかではない。どこまでを構造にするかの、程度の差である。OKF も、型、資産の URI、リンクという最小限の構造を持つ。Ossie は、指標や概念の定義そのものを構造にする。OKF は、定義の中身を文章に残す。

OKF は 0.2 版の段階にある。ここに書いた内容は当事者の説明にもとづくもので、採用の実績は本調査では評価していない。

10 データスペースとの接点

データスペースが意味をどう扱っているかを、国際、欧州、日本の順に見る。米国については、データスペースにあたる取り組みを本調査では扱っていない。

10.1 国際：データスペースプロトコルとその標準化

データスペースプロトコルは、Eclipse Foundation の作業グループ(Eclipse Dataspace Working Group)が公開している仕様である [57]。特定の地域のものでないので、最初に扱う。

表 19: データスペースプロトコルにおける意味の扱い

対象	意味の扱い	出典
データスペースプロトコル 2025-1	メッセージは JSON Schema で規範的に定義。JSON-LD 1.1 を使用。独自の名前空間 <code>dspace</code> を持つ。カタログは DCAT と ODRL の語彙を再利用し、その再利用は JSON Schema と JSON-LD コンテキストを通じて暗黙に行われる	[8], [19], [116]

Eclipse Foundation は 2025 年 12 月 2 日、ISO/IEC JTC1 の PAS 手続きで国際標準化に提出される 2 つのプロトコル仕様の公開を発表した [57]。2 つの仕様は Eclipse Dataspace Protocol と Eclipse Dataspace Decentralised Claims Protocol である [57]。セルビアの標準化機関が掲載する ISO の案件記録によれば、ISO/IEC DIS 26450 (Eclipse Dataspace Protocol) は 2026 年 9 月 15 日付で FDIS としての登録が承認された段階にある [117]。この記録は ISO 本体のページではなく、各国機関による掲載である。

10.1.1 データスペースプロトコルの成り立ち

データスペースプロトコルは、International Data Spaces Association (IDSA) の中で作られた。0.8 版と 2024-1 版は IDSA の管理のもとで開発され、2024-1 版が Eclipse のプロジェクトへの最初の寄贈となった [118]。

IDSA にはそれ以前から、IDS インフォメーションモデルというオントロジーがあった(セクション 10.2)。ただし、データスペースプロトコルがこのオントロジーを基にしているという記述は、調査した範囲では見つからなかった。最初期の 0.8 版は、データ資産を DCAT のカタログとして、利用制御を ODRL のポリシーとして表すとしている [119]。プロトコルは `dspace` という独自の名前空間も定義している [116]。

何を規範とするかは、先行する IDS-RAM と、データスペースプロトコルの各版とで違う。IDS-RAM はプロトコルの版ではなく、IDSA の別の仕様である。

表 20: 規範の置き方の移り変わり

段階	規範の置き方	出典
IDS-RAM 4.0	オントロジーである IDS 語彙が、インフォメーションモデルの唯一の規範的な仕様	[120]
データスペースプロトコル 2024-1	メッセージは JSON-LD のコンパクト形式で直列化しなければならない。JSON Schema と SHACL のシェイプが添えられているが、メッセージの型を規範的に定めるものではなかった。版の情報を返すデータだけは、両方に従うとされた	[121], [122], [123]

段階	規範の置き方	出典
データスペースプロトコル 2025-1	メッセージは JSON Schema で規範的に定義する。 JSON-LD のコンテキストも提 供するが、JSON として処理す るか JSON-LD として処理する かは実装が選べる	[19]

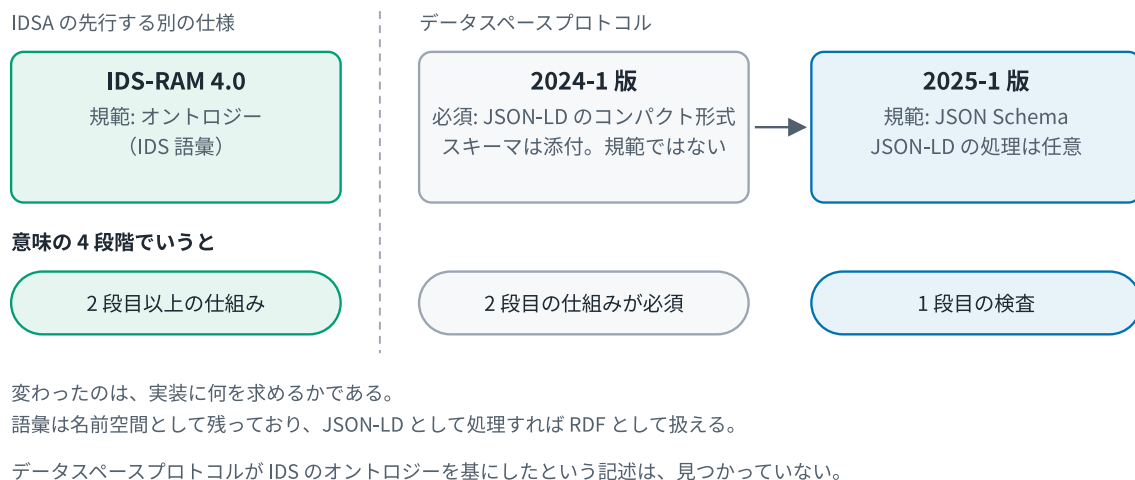


図 37: 規範の置き方の移り変わり。上の段は各仕様が規範に置くもの、下の段は意味の 4 段階での位置を表す。表 20 の出典をもとに筆者が作成した。下の段は筆者の整理である。

JSON Schema を規範とする変更は、2024 年 9 月の提案に由来する。提案は、当時の仕様がメッセージの型を規範的に定めるスキーマ技術を使っておらず、実装に JSON-LD の処理を強いていることを問題に挙げ、JSON-LD の処理を任意にすることを求めた [123]。

考察：検査の規範が、グラフから文書の形へ移った

この移り変わりは、意味を捨てたということではない。語彙は名前空間として残っており、JSON-LD として処理すれば RDF として扱える。変わったのは、実装に何を求めるかである。IDS-RAM では、オントロジーがインフォメーションモデルの規範であり、記述の検証には SHACL のシェイプを使うとされた [120]。2024-1 版は、メッセージを JSON-LD として扱うことを求めた。2025-1 版は、JSON Schema に合うことを求め、JSON-LD としての処理は任意にした。

セクション 4.4 の意味の 4 段階でいえば、規範が 2 段目以上の仕組みから 1 段目の検査へ移ったことになる。実装する側は、RDF やオントロジーを解釈できなくても、JSON の形が合っていれば接続できる。接続のしやすさを優先した選択だと考えられる。提案が相互運用性の妨げとして複雑さを挙げていることは、この解釈と合う。

10.2 欧州

欧州相互運用性フレームワークは、相互運用性を法的、組織的、意味的、技術的の 4 層で整理する [9]。この報告書が扱うのは、このうち意味的な層である。

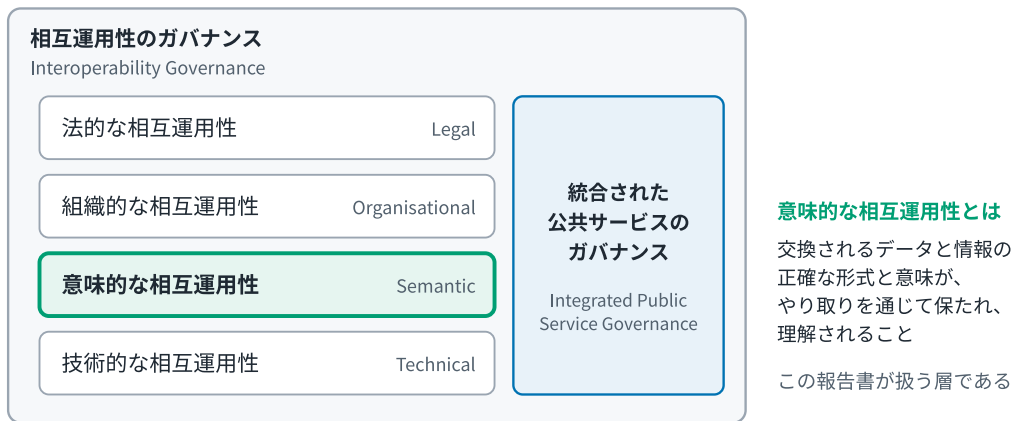


図 38: 欧州相互運用性フレームワークの相互運用性モデル。同文書の図 3 [9] をもとに筆者が作成した。意味的な層の強調と右側の説明は筆者が加えた。

表 21: 欧州のデータスペースにおける意味の扱い

対象	意味の扱い	出典
IDS インフォメーションモデル	IDS の基本概念を扱う RDFS/OWL オントロジー。契約クラスは ODRL の Policy のサブクラスで、Resource は DCAT-AP 1.1 の属性にもとづく。GitHub リポジトリはアーカイブ済み	[67], [124], [125], [126]
Gaia-X	クレデンシャルは、クレームを RDF で表現した W3C 検証可能クレデンシャル	[66]
Catena-X	アスペクトモデルは SAMM に準拠しなければならない。SAMM は Eclipse ESMF のメタモデルで、RDF と Turtle で規定され、SHACL の検証規則を伴う	[70], [127]
DCAT-AP	欧州でカタログ情報を共有するための DCAT プロファイル。3.0.1 は 2025-10-27 付の SEMIC 勧告	[71]
DSSC ブループリント	データモデルを、語彙、オントロジー、アプリケーションプロファイル、スキーマ仕様を含むものと定義	[23]

IDS-RAM 4.0 は、宣言的表現である IDS 語彙を、インフォメーションモデルの唯一の規範的な仕様としている [120]。このオントロジーは、契約には ODRL、データ資源には DCAT-AP という既存の語彙を取り込んでいた [125], [126]。その後の規範の移り変わりはセクション 10.1.1 で述べた。

10.3 日本

表 22: 日本の ODS-RAM における意味の扱い

対象	意味の扱い	出典
ODS-RAM（日本）	データ、トランザクション、アイデンティティ、セマンティクスの 4 レイヤ。セマンティクスレイヤはメタデータの提供による意味とオントロジーの相互運用性を要求	[10], [128]

10.3.1 ODS の設計思想

IPA は ODS の設計思想を説明する文書を公開している [129]。意味の扱いについて、次の考え方が示されている。

表 23: ODS の設計思想における意味の扱い

考え方	内容
データモデルと情報モデルの分離	観測された事実を記録するデータモデルから、事実の解釈や制約を表す情報モデルを明確に分離する。前者を Data Product、後者を Ontology Product と呼ぶ
開世界仮説を前提にする	Ontology Product では、判断できないことを偽とみなさない開世界仮説を前提とする。Data Product では、閉世界仮説を選択的に導入する。二層の構造である
意味を後から付与する	先にデータがあり、そこに意味を後から付与する。これをスキーマフレキシブルと呼ぶ
RDF、RDF Schema、OWL を使う	RDF で事実を知識の単位として保持し、RDF Schema で語彙と構造を与え、OWL で妥当性や排他性といった制約を定義する
閉世界的な検証を強制しない	SHACL のような閉世界的な検証を強制することは、自己宣言とゆるやかな統合をめざす ODS の発想と相いれないとする。代わりに、二段階の問い合わせを導入する

同文書は、この設計を、意味の所在をコードから論理体系へ移すものと説明している。強みとして挙げるのは、均質な完全性を強制することではなく、開放性を維持しながら選択的な厳格性を実現できる点である [129]。

なお、同文書は第一の柱を「Ontology and Semantic Interoperability (OSI)」と呼んでいる [129]。セクション 9.4 の Open Semantic Interchange と略称が同じだが、別のものである。

10.3.2 ODS のオープンソース実装

ODS は GitHub で実装を公開している。主導する中立的な組織は、IPA のデジタルアーキテクチャ・デザインセンター (DADC) である [130]。セマンティクスレイヤにあたるリポジトリを見ると、ODS がオントロジーを実際にどう使っているかが分かる。

表 24: ODS のオープンソース実装における意味の扱い

対象	内容	出典
ODS オントロジー	SDK for Semantics に含まれる。OWL のオントロジーとして宣言され、題は“ODS (Open Data Space) Ontology”。データスペースにおけるデータ交換サービスとエンドポイントを定義する。 ods:DataExchangeService などのクラスを持つ	[14]
Web サービスの意味定義	Eclipse ESMF の SAMM をベースとする。SAMM には REST API の呼び出し方を定義する手段がないため、W3C 標準の語彙を組み合わせて API 定義を拡張している	[131]
領域のオントロジー	サンプルに、ドローン航路の概念を OWL のクラスとして定義したものがある。UASL は Unmanned Aircraft Systems Lines (ドローン航路)の略である	[130], [132]
RDF の格納と問い合わせ	サンプルの RDF サーバーは、Apache Jena Fuseki に対して SPARQL クエリを実行する	[133]
分散カタログ	複数のドメインアプリケーションが公開する SPARQL エンドポイントから、RDF データを収集して格納する	[134]
AI との接続	チャットサーバは、SAMM アスペクトモデルと API 定義から MCP ツールを生成する	[135]

MCP (Model Context Protocol)は、AI アプリケーションを外部のシステムにつなぐためのオープンな標準である [136]。

二層のオントロジー（OWL）



分散カタログ



AI との接続



サンプルの RDF サーバーは、Apache Jena Fuseki に対して SPARQL クエリを実行する。
各要素の出典は、本文の表に示した。

図 39: ODS のオープンソース実装における意味の扱い。表 24 の内容をもとに筆者が作成した。

考察

ODS の実装でいうオントロジーは、セクション 2 の三つの用法のうち Web 標準の用法にあたる。ただし対象は二層に分かれている。一つは ODS オントロジーで、扱うのは業務データの概念ではなく、サービスと接続先である。もう一つは領域のオントロジーで、ドローン航路のような分野ごとの概念を扱う。

意味定義から MCP ツールを生成する実装は、意味をあらかじめ定義して AI に渡すという考え方の具体例と考えられる。

確認した範囲のリポジトリには、SHACL のシェイプは見つからなかった。ただし、全ファイルを確認したわけではない。閉世界的な検証を強制しないという設計思想(表 23)と合う結果である。

10.4 考察：データスペースプロトコルと ODS の違い

国際の節のデータスペースプロトコルと、日本の節の ODS とでは、何を規範に置くかが違う。

表 25: データスペースプロトコルと ODS の対比(筆者の整理)

観点	データスペースプロトコル 2025-1	ODS（日本）
規範に置くもの	JSON Schema。メッセージの形を定める	RDF。メタデータは RDF で表現しなければならない
オントロジー	独自の名前空間を持ち、DCAT と ODRL の語彙を再利用する。オントロジーを規範としない	情報モデルをオントロジーとして分離し、OWL で制約を定義する。これは設計思想の文書の記述で、規範的な要件は確認できていない
検査の考え方	文書の形が JSON Schema に合うかを検査する	開世界仮説を前提にし、閉世界的な検証は強制しない。選択的に導入する
問い合わせ	仕様の対象外	SPARQL を取得手段の一つに挙げる

セクション 4.4 の意味の 4 段階の上に両者を置くと、図 40 のようになる。

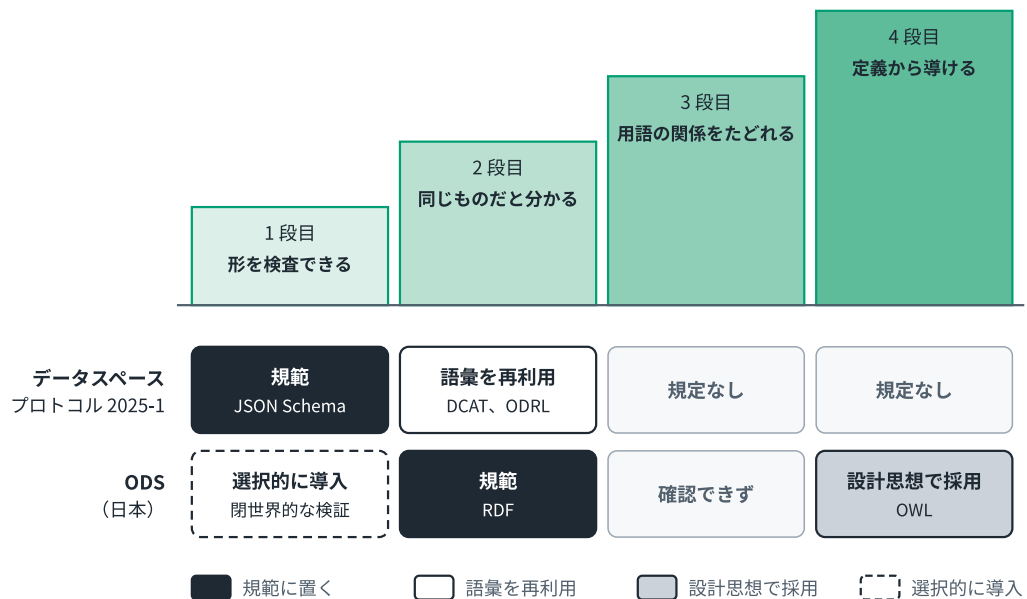


図 40: データスペースプロトコルと ODS が、意味の 4 段階のどこを規範に置くか。黒く塗った枠が規範、白い枠が語彙の再利用、灰色の枠が設計思想の文書での採用、破線の枠が選択的な導入を表す。筆者の整理である。

データスペースプロトコルは、1 段階目の検査を規範とする [19]。2 段階目の語彙である DCAT と ODRL は再利用するが、その規範は JSON Schema に置かれている [8]。3 段階目と 4 段階目にあたる規定はない。ODS は、2 段階目の RDF を規範に置いている [26]。4 段階目の OWL は、設計思想の文書が制約の定義に使うと述べている [129]。ただし、OWL を求める規範的な要件は、調査した仕様の文書では確認できなかった。1 段階目にあたる閉世界的な検証は、強制せず、選択的に導入する。3 段階目にあたる記述は、調査した文書では確認できなかった。

これは優劣ではなく、設計の選択である。データスペースプロトコルは、接続のしやすさを優先したと考えられる(セクション 10.1.1)。ODS は、提供者が不完全な記述のまま参加できる開放性を保ちつつ、意味の矛盾は論理的に検出するという考え方を、設計思想として明示している [129]。

この対比は、仕様と設計思想の文書、および公開されている実装の記述にもとづく。実際の運用での違いは確認していない。

11 考察：体系化の試み

以下は筆者による整理である。セクション 5 の四本の流れを、解く問題、中心となる要素、形式、標準化の場の四つの観点で比べる。製品としてのオントロジーは四本のどれにも属さないのので、別の行として加えた。

表 26: 四本の流れと製品のオントロジーの比較(筆者の整理)

流れ	解く問題	中心となる要素	代表的な形式	標準化の場
データベース系	データの保存と共有	表、列、キー	関係モデル、プロパティグラフ	ISO/IEC (GQL など)

流れ	解く問題	中心となる要素	代表的な形式	標準化の場
知識表現系	知識を計算機に扱わせる	概念、関係、公理	記述論理、上位オントロジー	学術界、ISO/IEC (BFO)
Web 標準系	組織や領域をまたぐ意味の共有	クラス、プロパティ、語彙	RDF、OWL、SHACL、DCAT	W3C
分析系	指標の定義の一元化	指標、ディメンション、結合	YAML や SQL による定義	Apache Software Foundation (Ossie)
製品のオントロジー(四本の外)	データと業務操作の結合	オブジェクト、リンク、アクション	製品独自の形式	単一の事業者

表 26 は流れごとの違いを示す。これに、セクション 4.1 の二つの軸を重ねると、主な取り組みの位置が一枚で見える。横に意味の 4 段階、縦に共有の範囲を取ったものが図 41 である。

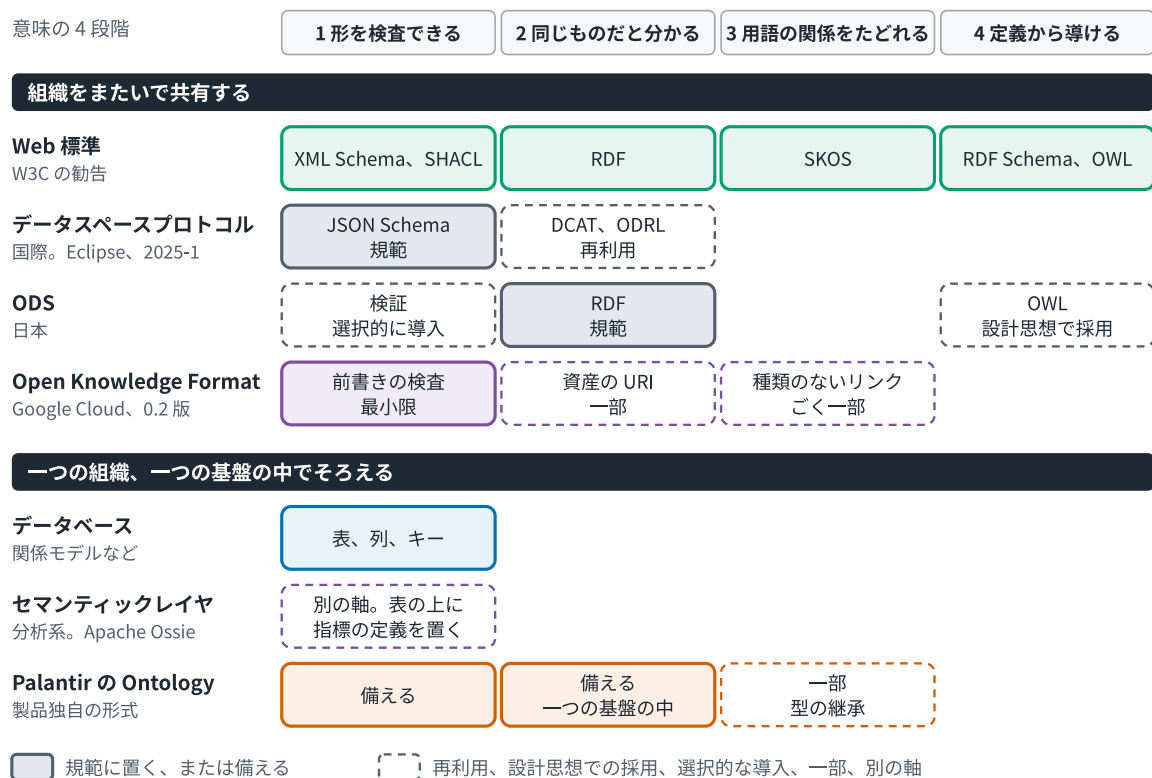


図 41: 意味の 4 段階と共有の範囲による見取り図。実線の枠は規範に置くか備える段、破線の枠は語彙の再利用、設計思想での採用、選択的な導入、一部、別の軸にあるものを表す。各取り組みの配置は筆者の整理であり、根拠は表 10、表 14、表 25 にある。

筆者はこの図から、次のことを読み取る。

- 2 段階以上を規範に置くか、設計思想に掲げる取り組みは、組織をまたぐ側に集まっている。Open Knowledge Format は、仕様が組織をまたぐ交換を想定しているので、こちらに置いた。ただし 2 段階と 3 段階は一部にとどまる。組織の間では、列の名前の意味を口頭ですり合わせる手段がないためと考えられる。
- 一つの組織や基盤の中でそろえる取り組みは、1 段階とその周辺が中心である。Palantir の Ontology は 2 段階に相当する機能を備えるが、範囲は一つの基盤の中に限られる。

- ・組織をまたぐ取り組みの間でも、規範に置く段は違う。データスペースプロトコルは1段目、ODSは2段目である。ODSは4段目のOWLも設計思想に掲げるが、規範的な要件は確認できていない(セクション 10.4)。

Web 標準の技術そのものは、組織の中でも使える。図の上の段に置いたのは、組織や領域をまたぐ共有を主な狙いとするためである。

課題から選ぶ

ここまでの整理を、選ぶ側の立場から並べ直す。何をしたいかから始め、条件で枝分かれして、候補にたどり着く。

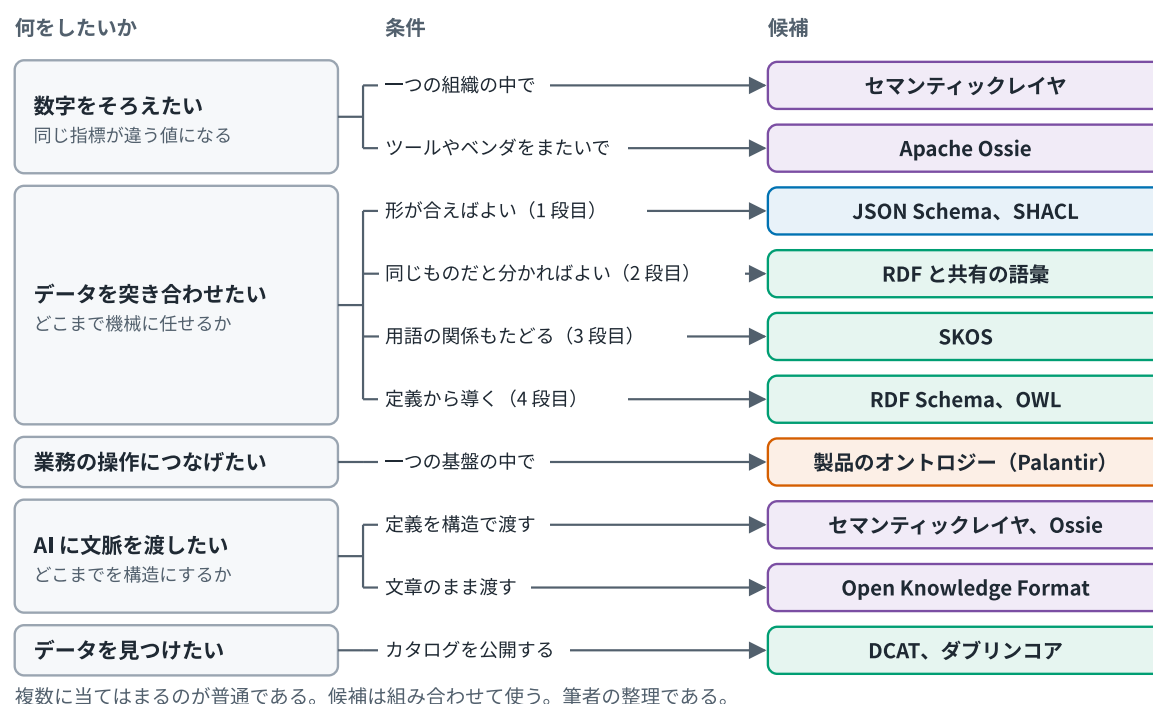


図 42: 課題から候補を選ぶ木。左から、したいこと、条件、候補の順に読む。筆者の整理である。

表 27: 課題から候補を選ぶ木の内容(筆者の整理)

したいこと	条件	候補	本報告書の該当箇所
数字をそろえたい	一つの組織の中で ツールやベンダをまたいで	セマンティックレイヤ Apache Ossie	セクション 6.4.1 セクション 9.4
別々のデータを突き合わせたい	形が合えばよい(1 段目)	JSON Schema、XML Schema。グラフのデータなら SHACL	セクション 6.3.9、セクション 6.3.6
	同じものだと分かればよい(2 段目)	RDF と共有の語彙	セクション 6.3.1
	用語の関係もたどる(3 段目)	SKOS	セクション 6.3.5
	定義から導く(4 段目)	RDF Schema、OWL。領域や上位のオントロジー	セクション 6.3.3、セクション 8
業務の操作につなげたい	一つの基盤の中で	製品のオントロジー (Palantir)	表 14

したいこと	条件	候補	本報告書の該当箇所
AI に文脈を渡したい	定義を構造で渡す	セマンティックレイ	セクション 9.4
	文章のまま渡す	や、Apache Ossie Open Knowledge Format	セクション 9.5
データを見つけない	カタログを公開する	DCAT、ダブリンコア	セクション 6.3.8

公の指針を、どう組み替えたか

選び方についての公の指針に、W3C の勧告がある。勧告は、形式的な意味の水準を、データと、最もありそうな用途の両方に合うように選ぶよう勧めている。複雑な語彙は、作るのにも理解するのにも手間がかかり、再利用の妨げになりうるとも述べる。語彙については、共有された語彙、できれば標準化された語彙の用語を使うよう勧めている [137]。

この木は、その指針を出発点にして、本調査で分かったことを三つ加えたものである。

表 28: W3C の勧告と、この木の関係(筆者の整理)

問い	W3C の勧告	この木での組み替え	加えた理由
どの水準を選ぶか	形式的な意味の水準を、データと用途に合わせる	形式性の高さではなく、機械に何をさせた いかで選ぶ。条件を意 味の 4 段階にした	形式性の順と、機械に できることの順は一致 しない(表 5)
何をしたいか	問いとして立てていない	最初の枝分かれにし た。数字、突き合わせ、 業務の操作、AI への文 脈、発見の五つ	同じ「セマンティック」 でも、流れが違えば解 く課題が違う(表 26)
誰と共有するか	データを Web に公開 する場面を前提にして いる	一つの組織や基盤の中 か、組織をまたぐかを 条件にした	製品のオントロジーは 一つの基盤の中に限ら れ、データスペースは 組織をまたぐ(図 41)
誰が読むか	人とプログラム	AI を読み手に加え、構 造で渡すか文章で渡す かを条件にした	近年の仕様は、読み手 に AI を挙げている(セ クション 9.5)

迷ったときの四つの問い

木の枝で迷ったら、次の四つの問いに戻るとよい。どれも、本調査で見た事例から引き出したものである。例は、部品と供給者の題材で作った説明用のものである。

表 29: 迷ったときの四つの問い(筆者の整理)

問い	目安	例	もともになった知見
機械に、どこまでさせたいか	課題を解くのに足りるところで止める。深く作り込むほどよい、とは限らない	重量の欄に数値が入っているかを確かめたいだけなら、形の検査(JSON Schema)で足りる。定義から導く仕組み(OWL)まで作る必要はない	集計の数字をそろえる仕組み(セマンティッククレーヤ)は、機械に推論させずに課題を解いてきた(セクション 9.3 のコラム)。データスペースプロトコルは、つなぎやすさを優先して、形の検査だけを必須にした(セクション 10.1.1)
書かれていないことを、どう扱いたい	誤りにしたいなら検査を、まだ分からないものとして扱いたいなら推論を選ぶ	供給者が書かれていない部品を、入力の流れとして止めたいなら検査(SHACL)。漏れではなく不明として残したいなら、定義の言語(OWL)	SHACL は書かれていなければ違反とし、OWL は分からないだけと扱う(セクション 6.3.6、セクション 6.3.3)。ODS は後者の考え方(開世界仮説)を前提にし、検査は必要なところだけに入れる(表 23)
AI に、何を任せるか	間違えると困るものは、決まった形の定義で渡す。背景の説明は、文章で渡す	「売上」の計算方法は、定義(セマンティッククレーヤ)で渡す。その数字を見るときに注意や経緯は、文章(Open Knowledge Format)で渡す	定義を決まった形で書く仕様(Apache Ossie)と、文章のまま渡す形式(Open Knowledge Format)がある(セクション 9.5)
誰と、やり取りするか	一つの製品の中で足りるうちは、製品の機能でよい。別の組織とやり取りするなら、標準の形式を選ぶ	社内の一つの基盤で部品と供給者を結ぶだけなら、製品のオントロジー(Palantir)でもできる。取引先と突き合わせるなら、共通の識別子と語彙(RDF)が要る	Palantir は、書き出しの形式が変わりうると公式文書で述べている(セクション 9.2)。ODS は、標準の形式(RDF)を必須にしている(セクション 10.4)

選び始める前に、確かめておくことが三つある。

- **相手の言葉の意味:** 相手が「オントロジー」や「セマンティック」という言葉で何を指しているかを確かめる。「オントロジー」には少なくとも三つの用法がある(セクション 2)。
- **すでにある言葉:** 項目の名前に使う言葉の一覧(語彙)や、概念の定義(オントロジー)は、作る前に既存のものを探す。W3C の勧告が勧めている [137]。分野ごとの標準はセクション 7 と セクション 8 にまとめた。
- **組み合わせ:** 候補は、どれか一つを選ぶものではない。同じものだと分かる仕組み(RDF)が、定義から導く仕組み(OWL)の土台になる(セクション 4.4)。

この木と四つの問いは、仕様や公式文書の記述から筆者が組み立てたものである。それぞれの候補を実際に導入した結果は、本調査では評価していない。

以上の整理から、次の三点が言える。

1. 用語の混乱は、同じ語が流れごとに別のものを指すことから生じている。「オントロジー」は知識表現系、Web 標準系、製品でそれぞれ違うものを、「セマンティック」は Web 標準系と分析系で違う課題を指す。
2. 流れごとに、解く問題が違う。組織をまたぐデータスペースには Web 標準系が、組織内の分析には分析系が合う。
3. 生成 AI の普及が、意味の定義を機械に渡す必要性を、どの流れでも高めていると推測される。渡し方は二つに分かれている。Apache Ossie のように構造化した定義として渡す方向と、Open Knowledge Format のように文章のまま渡す方向である(セクション 9.5)。

12 確認できなかった点と留意事項

次の事項は一次情報で裏づけられなかった。本文では事実として書いていない。

- ・「データモデル標準化」という語そのものの定義。本文の説明は、デジタル庁の政府相互運用性フレームワークと IPA の手引きにもとづく。
- ・ ISO/IEC 11179 (メタデータレジストリ)と ISO 15926 の適用範囲と発行年。
- ・ ANSI/X3/SPARC の三層スキーマの年。
- ・ Wikidata の公開日。
- ・ Power BI 改称の発表日。改称の事実は Microsoft の公式文書で確認したが、時期は二次情報による。
- ・ Microsoft と Google の OSI への関与。
- ・ 調査会社によるセマンティックレイヤの定義。
- ・ 過去の共通化の試み(CWM、JOLAP、RDF Data Cube 語彙)が、実際にどれだけ使われたか。版や手続きの記録は確認したが、採用の実績は確認していない。
- ・ KL-ONE を OWL の祖先と明言する一次情報。
- ・ 論文「The Semantic Web」本文の定義文。

データスペースプロトコルの ISO/IEC での審議段階は、ISO 本体ではなく各国標準化機関が掲載する記録で確認した(セクション 10)。

Cyc の開始年、OSI の発表内容、各社の製品説明は当事者自身の記述に依拠している。

13 付録 A :用語集

表 30: 用語集

用語	説明	出典
RDF (Resource Description Framework)	Web 上で情報を表現する枠組み	[47]
OWL (Web Ontology Language)	形式的に定義された意味を持つオントロジー言語	[13]
SKOS (Simple Knowledge Organization System)	知識組織化体系を共有しリンクする共通データモデル	[41]
SHACL (Shapes Constraint Language)	RDF グラフを条件に照らして検証する言語	[48]
JSON-LD	Linked Data を直列化する JSON ベースの形式。現行は 1.1	[45], [49]

用語	説明	出典
DCAT (Data Catalog Vocabulary)	データカタログ間の相互運用性のための RDF 語彙。最新は第 3 版(2024-08-22 勧告)	[55]
MOF (Meta Object Facility)	OMG のメタモデル定義の基礎	[76]
BFO (Basic Formal Ontology)	ISO/IEC 21838-2:2021 に収められた上位オントロジー	[50]
GQL	プロパティグラフのデータ構造と操作を定義する ISO/IEC 規格	[54]
SAMM (Semantic Aspect Meta Model)	RDF と Turtle で規定されたメタモデル	[70]
OSI (Open Semantic Interchange)	2025 年に発表されたセマンティックモデル交換の取り組み。現 Apache Ossie (Incubating)	[16], [17]
OKF (Open Knowledge Format)	知識を、YAML の前書きつき Markdown ファイルのディレクトリで表す形式。Google Cloud が 2026 年に発表した	[18], [114]
ODS-RAM	Open Data Spaces Reference Architecture Model	[128]

参考文献

- [1] Apache Ossie, 「Apache Ossie (Incubating): The New Name for OSI (2026-07-10)」. 2026 年 7 月. 参照: 2026 年 10 月 2 日. [Online]. 入手先: <https://ossie.apache.org/updates/ossie-enters-apache-incubator/>
- [2] C. K. Ogden と I. A. Richards, The Meaning of Meaning. Harcourt, Brace & World, 1923. 参照: 2026 年 10 月 3 日. [Online]. 入手先: https://archive.org/download/bwb_W9-COP-402/bwb_W9-COP-402_djvu.txt
- [3] デジタル庁, 「デジタル庁, 政府相互運用性フレームワーク (GIF) 概要資料 (2025-03-25)」, デジタル庁, 3 月 2025. 参照: 2026 年 10 月 3 日. [Online]. 入手先: https://www.digital.go.jp/assets/contents/node/basic_page/field_ref_resources/c1bdb4a2-850f-48ec-a938-a8369f820ac1/d47def21/20250325_policies_development_management_outline_03.pdf
- [4] T. R. Gruber, 「A Translation Approach to Portable Ontology Specifications」, Knowledge Acquisition, vol. 5, no. 2, pp. 199–220, 6 月 1993, doi: [10.1006/knac.1993.1008](https://doi.org/10.1006/knac.1993.1008).
- [5] W3C, 「W3C, OWL Web Ontology Language Overview (REC 2004-02-10)」, W3C, 2 月 2004. 参照: 2026 年 10 月 2 日. [Online]. 入手先: <https://www.w3.org/TR/2004/REC-owl-features-20040210/>
- [6] IEC, 「IEC 63278-1:2023 Asset Administration Shell for industrial applications, Part 1 (IEC webstore)」, IEC, IEC 63278-1:2023, 12 月 2023. 参照: 2026 年 10 月 2 日. [Online]. 入手先: <https://webstore.iec.ch/en/publication/65628>
- [7] 情報処理推進機構, 「IPA, データの相互運用性向上のためのガイド 第二版(2021-03-25)」, IPA, 3 月 2021. 参照: 2026 年 10 月 3 日. [Online]. 入手先: <https://www.ipa.go.jp/digital/data/ug65p90000001lp7-att/000089404.pdf>

- [8] Eclipse Dataspace Working Group, 「Dataspace Protocol 2025-1, Catalog Protocol (source md)」, Eclipse Foundation, 2025. 参照: 2026 年 10 月 2 日. [Online]. 入手先: <https://raw.githubusercontent.com/eclipse-dataspace-protocol-base/DataspaceProtocol/2025-1/specifications/catalog/catalog.protocol.md>
- [9] European Commission, 「European Commission, New European Interoperability Framework (2017, brochure of COM(2017) 134 Annex 2)」, European Commission, 2017. 参照: 2026 年 10 月 2 日. [Online]. 入手先: https://ec.europa.eu/isa2/sites/default/files/eif_brochure_final.pdf
- [10] 情報処理推進機構, 「ODS-RAM 3. レイヤ (ODS 公式 GitBook)」, IPA, 2026. 参照: 2026 年 10 月 2 日. [Online]. 入手先: <https://open-dataspaces.gitbook.io/ods-docs/jp/ods-ram/03-layers>
- [11] Databricks, 「Databricks blog, Redefining the Semantics Data Layer (2026-04-02)」. 2026 年 4 月. 参照: 2026 年 10 月 2 日. [Online]. 入手先: <https://www.databricks.com/blog/redefining-semantics-data-layer-future-bi-and-ai>
- [12] R. Studer, V. R. Benjamins, と D. Fensel, 「Knowledge Engineering: Principles and Methods」, Data & Knowledge Engineering, vol. 25, no. 1-2, pp. 161-197, 3 月 1998, doi: [10.1016/S0169-023X\(97\)00056-6](https://doi.org/10.1016/S0169-023X(97)00056-6).
- [13] W3C, 「W3C, OWL 2 Document Overview (Second Edition) (REC 2012-12-11)」, W3C, 12 月 2012. 参照: 2026 年 10 月 2 日. [Online]. 入手先: <https://www.w3.org/TR/2012/REC-owl2-overview-20121211/>
- [14] Open Data Spaces, 「open-dataspaces/SDK-for-semantics, ods.ttl (ODS Ontology)」. 2026 年 3 月. 参照: 2026 年 10 月 3 日. [Online]. 入手先: <https://raw.githubusercontent.com/open-dataspaces/SDK-for-semantics/main/Documents/WebServiceAspectModel/ods.ttl>
- [15] Palantir Technologies, 「Palantir Foundry docs, Ontology overview」. 2026 年. 参照: 2026 年 10 月 2 日. [Online]. 入手先: <https://www.palantir.com/docs/foundry/ontology/overview/>
- [16] Snowflake, 「Snowflake press release 2025-09-23 (Yahoo Finance copy)」. 2025 年 9 月. 参照: 2026 年 10 月 2 日. [Online]. 入手先: <https://finance.yahoo.com/news/snowflake-salesforce-dbt-labs-more-130100051.html>
- [17] Apache Incubator, 「Apache Incubator, Ossie Project Incubation Status」. 2026 年 6 月. 参照: 2026 年 10 月 2 日. [Online]. 入手先: <https://incubator.apache.org/projects/ossie.html>
- [18] S. McVeety と A. Hormati, 「Google Cloud Blog, Introducing the Open Knowledge Format (2026-06-12)」. 2026 年 6 月. 参照: 2026 年 10 月 4 日. [Online]. 入手先: <https://cloud.google.com/blog/products/data-analytics/how-the-open-knowledge-format-can-improve-data-sharing/>
- [19] Eclipse Dataspace Working Group, 「Dataspace Protocol 2025-1, Common (source md)」, Eclipse Foundation, 2025. 参照: 2026 年 10 月 2 日. [Online]. 入手先: <https://raw.githubusercontent.com/eclipse-dataspace-protocol-base/DataspaceProtocol/2025-1/specifications/common/common.protocol.md>
- [20] M. Uschold と M. Gruninger, 「Ontologies and Semantics for Seamless Connectivity」, SIGMOD Record, vol. 33, no. 4, pp. 58-64, 12 月 2004, doi: [10.1145/1041410.1041420](https://doi.org/10.1145/1041410.1041420).
- [21] N. Guarino, 「Formal Ontology and Information Systems」, Formal Ontology in Information Systems: Proceedings of FOIS'98, IOS Press, 6 月 1998, pp. 3-15. 参照: 2026 年 10 月 2 日. [Online]. 入手先: <https://www.loa.istc.cnr.it/old/Papers/FOIS98.pdf>
- [22] W3C, 「W3C Semantic Web Activity」. 2013 年. 参照: 2026 年 10 月 2 日. [Online]. 入手先: <https://www.w3.org/2001/sw/>
- [23] Data Spaces Support Centre, 「DSSC Blueprint v2.0, Data Models」, DSSC, 2025. 参照: 2026 年 10 月 2 日. [Online]. 入手先: <https://archive.dssc.eu/space/BVE2/1071255252.md>
- [24] AtScale, 「AtScale glossary, Semantic Layer」. 2026 年. 参照: 2026 年 10 月 2 日. [Online]. 入手先: <https://www.atscale.com/glossary/semantic-layer/>

- [25] A. Hogan **ほか**, 「Knowledge Graphs」, ACM Computing Surveys, vol. 54, no. 4, 7月2021, doi: [10.1145/3447772](https://doi.org/10.1145/3447772).
- [26] 情報処理推進機構, 「ODP メタデータエクステンジ (L4) Protocol (ODS 公式 GitBook)」, IPA, 2026. 参照: 2026年10月2日. [Online]. 入手先: <https://open-dataspaces.gitbook.io/ods-docs/jp/odp/fundamentarupurotokoru/metadtaekusuchenjl4/protocol.md>
- [27] D. L. McGuinness, 「Ontologies Come of Age」, Spinning the Semantic Web: Bringing the World Wide Web to Its Full Potential, MIT Press, 2003. 参照: 2026年10月3日. [Online]. 入手先: <https://web.archive.org/web/20201129074402/http://www.ksl.stanford.edu/people/dlm/papers/ontologies-come-of-age-mit-press-%28with-citation%29.htm>
- [28] A. Pras と J. Schoenwaelder, 「On the Difference between Information Models and Data Models」, IETF, RFC 3444, 1月2003. 参照: 2026年10月3日. [Online]. 入手先: <https://www.rfc-editor.org/rfc/rfc3444.txt>
- [29] W3C, 「W3C, Ontologies (Standards > Semantic Web. Wayback 2013 収録)」. 2013年. 参照: 2026年10月3日. [Online]. 入手先: <https://web.archive.org/web/2013/http://www.w3.org/standards/semanticweb/ontology>
- [30] E. F. Codd, 「A Relational Model of Data for Large Shared Data Banks」, Communications of the ACM, vol. 13, no. 6, pp. 377–387, 6月1970, doi: [10.1145/362384.362685](https://doi.org/10.1145/362384.362685).
- [31] P. P.-S. Chen, 「The Entity-Relationship Model—Toward a Unified View of Data」, ACM Transactions on Database Systems, vol. 1, no. 1, pp. 9–36, 3月1976, doi: [10.1145/320434.320440](https://doi.org/10.1145/320434.320440).
- [32] D. B. Lenat, 「CYC: A Large-Scale Investment in Knowledge Infrastructure」, Communications of the ACM, vol. 38, no. 11, 11月1995, 参照: 2026年10月2日. [Online]. 入手先: <https://papers.baulab.info/papers/knowledge/Lenat-1995.pdf>
- [33] R. J. Brachman と J. G. Schmolze, 「An Overview of the KL-ONE Knowledge Representation System」, Cognitive Science, vol. 9, no. 2, pp. 171–216, 4月1985, doi: [10.1207/s15516709cog0902_1](https://doi.org/10.1207/s15516709cog0902_1).
- [34] J.-M. Cambot と B. Liautaud, 「Relational database access system using semantically dynamic objects (US Patent 5,555,403)」. 1996年9月. 参照: 2026年10月3日. [Online]. 入手先: <https://patents.google.com/patent/US5555403A/en>
- [35] W3C, 「W3C, RDF Model and Syntax Specification (REC 1999-02-22)」, W3C, 2月1999. 参照: 2026年10月2日. [Online]. 入手先: <https://www.w3.org/TR/1999/REC-rdf-syntax-19990222/>
- [36] M. Ashburner, C. A. Ball, J. A. Blake, D. Botstein, **ほか**, 「Gene Ontology: tool for the unification of biology」, Nature Genetics, vol. 25, no. 1, pp. 25–29, 5月2000, doi: [10.1038/75556](https://doi.org/10.1038/75556).
- [37] T. Berners-Lee, J. Hendler, と O. Lassila, 「The Semantic Web」, Scientific American, vol. 284, no. 5, 5月2001, 参照: 2026年10月2日. [Online]. 入手先: <https://www.scientificamerican.com/article/the-semantic-web/>
- [38] W3C, 「W3C, XML Schema Part 1: Structures (REC 2001-05-02)」, W3C, 5月2001. 参照: 2026年10月2日. [Online]. 入手先: <https://www.w3.org/TR/2001/REC-xmlschema-1-20010502/>
- [39] Object Management Group, 「OMG, About the Common Warehouse Metamodel Specification Version 1.1」, OMG, 3月2003. 参照: 2026年10月4日. [Online]. 入手先: <https://www.omg.org/spec/CWM/1.1/About-CWM>
- [40] W3C, 「W3C, SPARQL Query Language for RDF (REC 2008-01-15)」, W3C, 1月2008. 参照: 2026年10月2日. [Online]. 入手先: <https://www.w3.org/TR/2008/REC-rdf-sparql-query-20080115/>
- [41] W3C, 「W3C, SKOS Reference (REC 2009-08-18)」, W3C, 8月2009. 参照: 2026年10月2日. [Online]. 入手先: <https://www.w3.org/TR/2009/REC-skos-reference-20090818/>

- [42] Google, 「Google Search Central Blog, Introducing schema.org (2011-06-02)」. 2011 年 6 月. 参照: 2026 年 10 月 2 日. [Online]. 入手先: <https://developers.google.com/search/blog/2011/06/introducing-schemaorg-search-engines>
- [43] Google, 「Google blog, Introducing the Knowledge Graph: things, not strings (2012-05-16)」. 2012 年 5 月. 参照: 2026 年 10 月 2 日. [Online]. 入手先: <https://blog.google/products/search/introducing-knowledge-graph-things-not/>
- [44] W3C, 「W3C, DCAT (REC 2014-01-16)」, W3C, 1 月 2014. 参照: 2026 年 10 月 2 日. [Online]. 入手先: <https://www.w3.org/TR/2014/REC-vocab-dcat-20140116/>
- [45] W3C, 「W3C, JSON-LD 1.0 (REC 2014-01-16)」, W3C, 1 月 2014. 参照: 2026 年 10 月 2 日. [Online]. 入手先: <https://www.w3.org/TR/2014/REC-json-ld-20140116/>
- [46] W3C, 「W3C, The RDF Data Cube Vocabulary (W3C Recommendation, 2014-01-16)」, W3C, 1 月 2014. 参照: 2026 年 10 月 4 日. [Online]. 入手先: <https://www.w3.org/TR/vocab-data-cube/>
- [47] W3C, 「W3C, RDF 1.1 Concepts and Abstract Syntax (REC 2014-02-25)」, W3C, 2 月 2014. 参照: 2026 年 10 月 2 日. [Online]. 入手先: <https://www.w3.org/TR/2014/REC-rdf11-concepts-20140225/>
- [48] W3C, 「W3C, Shapes Constraint Language (SHACL) (REC 2017-07-20)」, W3C, 7 月 2017. 参照: 2026 年 10 月 2 日. [Online]. 入手先: <https://www.w3.org/TR/2017/REC-shacl-20170720/>
- [49] W3C, 「W3C, JSON-LD 1.1 (REC 2020-07-16)」, W3C, 7 月 2020. 参照: 2026 年 10 月 2 日. [Online]. 入手先: <https://www.w3.org/TR/2020/REC-json-ld11-20200716/>
- [50] ISO/IEC, 「ISO/IEC 21838-2:2021 BFO (IEC webstore)」, ISO/IEC, ISO/IEC 21838-2:2021, 11 月 2021. 参照: 2026 年 10 月 2 日. [Online]. 入手先: <https://webstore.iec.ch/publication/72703>
- [51] Microsoft, 「Microsoft Learn, Datasets renamed to semantic models」. 2023 年 11 月. 参照: 2026 年 10 月 2 日. [Online]. 入手先: <https://learn.microsoft.com/en-us/power-bi/connect-data/service-datasets-rename>
- [52] C. Webb, 「Chris Webb, Power BI Datasets Renamed To Semantic Models (2023-11-18)」. 2023 年 11 月. 参照: 2026 年 10 月 2 日. [Online]. 入手先: <https://blog.crossjoin.co.uk/2023/11/18/thoughts-on-power-bi-datasets-being-renamed-to-semantic-models/>
- [53] Microsoft Research, 「Microsoft Research blog, GraphRAG (2024-02-13)」. 2024 年 2 月. 参照: 2026 年 10 月 2 日. [Online]. 入手先: <https://www.microsoft.com/en-us/research/blog/graphrag-unlocking-llm-discovery-on-narrative-private-data/>
- [54] ISO/IEC, 「ISO/IEC 39075:2024 GQL (IEC webstore)」, ISO/IEC, ISO/IEC 39075:2024, 4 月 2024. 参照: 2026 年 10 月 2 日. [Online]. 入手先: <https://webstore.iec.ch/publication/94107>
- [55] W3C, 「W3C, DCAT Version 3」, W3C, 8 月 2024. 参照: 2026 年 10 月 2 日. [Online]. 入手先: <https://www.w3.org/TR/vocab-dcat-3/>
- [56] dbt Labs, 「dbt Labs blog, Open Sourcing MetricFlow (2025-10-14)」. 2025 年 10 月. 参照: 2026 年 10 月 2 日. [Online]. 入手先: <https://www.getdbt.com/blog/dbt-labs-affirms-commitment-to-open-semantic-interchange-by-open-sourcing-metricflow>
- [57] Eclipse Foundation, 「Eclipse Foundation newsroom, EDWG advances two open protocols toward ISO/IEC standardisation (2025-12-02)」. 2025 年 12 月. 参照: 2026 年 10 月 2 日. [Online]. 入手先: <https://newsroom.eclipse.org/node/43194>
- [58] Snowflake, 「Snowflake blog, OSI Updates: Specification Live (2026-01-27)」. 2026 年 1 月. 参照: 2026 年 10 月 2 日. [Online]. 入手先: <https://www.snowflake.com/en/blog/open-semantic-interchanges-specs-finalized/>
- [59] W3C, 「W3C, RDF 1.2 Concepts and Abstract Data Model」, W3C, 4 月 2026. 参照: 2026 年 10 月 2 日. [Online]. 入手先: <https://www.w3.org/TR/rdf12-concepts/>

- [60] Apache Ossie, 「Apache Ossie, Core Metadata Specification (version 0.2.0.dev0)」, Apache Software Foundation, 2026. 参照: 2026 年 10 月 2 日. [Online]. 入手先: <https://raw.githubusercontent.com/apache/ossie/main/core-spec/spec.md>
- [61] Neo4j, 「Neo4j blog, RDF vs. Property Graphs (2024-06-04)」. 2024 年 6 月. 参照: 2026 年 10 月 2 日. [Online]. 入手先: <https://neo4j.com/blog/knowledge-graph/rdf-vs-property-graphs-knowledge-graphs/>
- [62] Common Core Ontologies, 「Common Core Ontologies README (GitHub)」. 2026 年. 参照: 2026 年 10 月 2 日. [Online]. 入手先: <https://raw.githubusercontent.com/CommonCoreOntology/CommonCoreOntologies/develop/README.md>
- [63] W3C, 「W3C, RDF Schema 1.1 (REC 2014-02-25)」, W3C, 2 月 2014. 参照: 2026 年 10 月 2 日. [Online]. 入手先: <https://www.w3.org/TR/2014/REC-rdf-schema-20140225/>
- [64] W3C, 「W3C, SPARQL 1.1 Overview (REC 2013-03-21)」, W3C, 3 月 2013. 参照: 2026 年 10 月 2 日. [Online]. 入手先: <https://www.w3.org/TR/2013/REC-sparql11-overview-20130321/>
- [65] W3C, 「W3C, DCAT Version 2」, W3C, 2 月 2020. 参照: 2026 年 10 月 2 日. [Online]. 入手先: <https://www.w3.org/TR/vocab-dcat-2/>
- [66] Gaia-X, 「Gaia-X Architecture Document 24.04, Conceptual model」, Gaia-X AISBL, 2024. 参照: 2026 年 10 月 2 日. [Online]. 入手先: https://docs.gaia-x.eu/technical-committee/architecture-document/24.04/gx_conceptual_model/
- [67] International Data Spaces Association, 「IDS Information Model README (GitHub)」. 2022 年. 参照: 2026 年 10 月 2 日. [Online]. 入手先: <https://raw.githubusercontent.com/International-Data-Spaces-Association/InformationModel/develop/README.md>
- [68] 情報処理推進機構, 「ODS-RAM 用語集 (ODS 公式 GitBook)」, IPA, 2026. 参照: 2026 年 10 月 2 日. [Online]. 入手先: <https://open-dataspaces.gitbook.io/ods-docs/jp/ods-ram/glossary.md>
- [69] EDM Council, 「EDM Council, FIBO」. 2026 年. 参照: 2026 年 10 月 2 日. [Online]. 入手先: <https://spec.edmcouncil.org/fibo/>
- [70] Eclipse Semantic Modeling Framework, 「Eclipse ESMF, SAMM Specification」, Eclipse Foundation, 2026. 参照: 2026 年 10 月 2 日. [Online]. 入手先: <https://eclipse-esmf.github.io/samm-specification/snapshot/index.html>
- [71] SEMIC, 「SEMIC, DCAT-AP 3.0.1」, European Commission, 10 月 2025. 参照: 2026 年 10 月 2 日. [Online]. 入手先: <https://semiceu.github.io/DCAT-AP/releases/3.0.1/>
- [72] JSON Schema, 「json-schema.org, What is JSON Schema?」 2026 年. 参照: 2026 年 10 月 2 日. [Online]. 入手先: <https://json-schema.org/overview/what-is-jsonschema>
- [73] Google Cloud, 「Google Cloud blog, Opening up the Looker semantic layer (2024-08-15)」. 2024 年 8 月. 参照: 2026 年 10 月 2 日. [Online]. 入手先: <https://cloud.google.com/blog/products/business-intelligence/opening-up-the-looker-semantic-layer>
- [74] Apache Ossie, 「Apache Ossie README (GitHub)」. 2026 年. 参照: 2026 年 10 月 2 日. [Online]. 入手先: <https://raw.githubusercontent.com/apache/ossie/main/README.md>
- [75] Apache Ossie, 「Apache Ossie, Ontology Specification (version 0.2.0.dev0)」, Apache Software Foundation, 2026. 参照: 2026 年 10 月 2 日. [Online]. 入手先: <https://raw.githubusercontent.com/apache/ossie/main/ontology/ontology.md>
- [76] Object Management Group, 「OMG, Meta Object Facility (MOF) 2.5.1」, OMG, 2016. 参照: 2026 年 10 月 2 日. [Online]. 入手先: <https://www.omg.org/spec/MOF/2.5.1/>
- [77] Dublin Core Metadata Initiative, 「DCMI History」. 2026 年. 参照: 2026 年 10 月 2 日. [Online]. 入手先: <https://www.dublincore.org/about/history/>
- [78] Dublin Core Metadata Initiative, 「DCMI Metadata Terms」. 2020 年. 参照: 2026 年 10 月 2 日. [Online]. 入手先: <https://www.dublincore.org/specifications/dublin-core/dcmi-terms/>
- [79] Schema.org, 「Schema.org, About」. 2026 年. 参照: 2026 年 10 月 2 日. [Online]. 入手先: <https://schema.org/docs/about.html>

- [80] 情報処理推進機構,「共通語彙基盤 (IMI)」. 2021 年. 参照: 2026 年 10 月 2 日. [Online]. 入手先: <https://imi.go.jp/>
- [81] Gene Ontology Consortium,「Gene Ontology documentation」. 2026 年. 参照: 2026 年 10 月 2 日. [Online]. 入手先: <https://geneontology.org/docs/ontology-documentation/>
- [82] OBO Foundry,「OBO Foundry」. 2026 年. 参照: 2026 年 10 月 2 日. [Online]. 入手先: <https://obofoundry.org/>
- [83] SNOMED International,「SNOMED International, What is SNOMED CT」. 2026 年. 参照: 2026 年 10 月 2 日. [Online]. 入手先: <https://www.snomed.org/what-is-snomed-ct>
- [84] HL7,「HL7 FHIR Overview」. 2023 年 3 月. 参照: 2026 年 10 月 2 日. [Online]. 入手先: <https://hl7.org/fhir/>
- [85] IOGP JIP36,「CFIHOS (JIP36)」. 2026 年. 参照: 2026 年 10 月 2 日. [Online]. 入手先: <https://www.jip36-cfihos.org/>
- [86] Industrial Digital Twin Association,「IDTA home」. 2026 年. 参照: 2026 年 10 月 2 日. [Online]. 入手先: <https://industrialdigitaltwin.org/en/>
- [87] ECLASS,「ECLASS introduction」. 2026 年. 参照: 2026 年 10 月 2 日. [Online]. 入手先: <https://eclass.eu/en/eclass-standard/introduction>
- [88] OAGi,「OAGi, Industrial Ontologies Foundry」. 2026 年. 参照: 2026 年 10 月 2 日. [Online]. 入手先: <https://oagi.org/pages/industrial-ontologies>
- [89] NIEMOpen,「NIEMOpen, NIEM, a new Open Project 18 years in the making (2023-02-17)」. 2023 年 2 月. 参照: 2026 年 10 月 2 日. [Online]. 入手先: <https://niemopen.org/news/niem-a-new-open-project-18-years-in-the-making/>
- [90] Wikidata,「Wikidata:Introduction」. 2026 年. 参照: 2026 年 10 月 2 日. [Online]. 入手先: <https://www.wikidata.org/wiki/Wikidata:Introduction>
- [91] C. Kritschmar,「Datamodel in Wikidata (Wikimedia Commons, CC0)」. 2016 年 6 月. 参照: 2026 年 10 月 3 日. [Online]. 入手先: https://commons.wikimedia.org/wiki/File:Damodel_in_Wikidata.svg
- [92] Palantir Technologies,「Palantir Foundry docs, Core concepts」. 2026 年. 参照: 2026 年 10 月 3 日. [Online]. 入手先: <https://www.palantir.com/docs/foundry/ontology/core-concepts/>
- [93] Palantir Technologies,「Palantir Foundry docs, Required properties」. 2026 年. 参照: 2026 年 10 月 3 日. [Online]. 入手先: <https://www.palantir.com/docs/foundry/object-link-types/required-properties/>
- [94] Palantir Technologies,「Palantir Foundry docs, Value types」. 2026 年. 参照: 2026 年 10 月 3 日. [Online]. 入手先: <https://www.palantir.com/docs/foundry/object-link-types/value-types-overview/>
- [95] Palantir Technologies,「Palantir Foundry docs, Property metadata」. 2026 年. 参照: 2026 年 10 月 4 日. [Online]. 入手先: <https://www.palantir.com/docs/foundry/object-link-types/property-metadata/>
- [96] Palantir Technologies,「Palantir Foundry docs, Multi-datasource object types (MDOs)」. 2026 年. 参照: 2026 年 10 月 3 日. [Online]. 入手先: <https://www.palantir.com/docs/foundry/object-permissioning/multi-datasource-objects/>
- [97] Palantir Technologies,「Palantir Foundry docs, Interfaces」. 2026 年. 参照: 2026 年 10 月 3 日. [Online]. 入手先: <https://www.palantir.com/docs/foundry/interfaces/interface-overview/>
- [98] Palantir Technologies,「Palantir Foundry docs, Derived properties」. 2026 年. 参照: 2026 年 10 月 3 日. [Online]. 入手先: <https://www.palantir.com/docs/foundry/ontology/derived-properties/>
- [99] Palantir Technologies,「Palantir Foundry docs, Functions」. 2026 年. 参照: 2026 年 10 月 3 日. [Online]. 入手先: <https://www.palantir.com/docs/foundry/functions/overview/>

- [100] Palantir Technologies, 「Palantir Foundry docs, Types reference」 . 2026 年. 参照: 2026 年 10 月 3 日. [Online]. 入手先: <https://www.palantir.com/docs/foundry/object-link-types/type-reference/>
- [101] Palantir Technologies, 「Palantir Foundry docs, Export, edit, and import an Ontology」 . 2026 年. 参照: 2026 年 10 月 3 日. [Online]. 入手先: <https://www.palantir.com/docs/foundry/ontology-manager/export-import/>
- [102] Palantir Technologies, 「Palantir Foundry docs, Shared properties」 . 2026 年. 参照: 2026 年 10 月 3 日. [Online]. 入手先: <https://www.palantir.com/docs/foundry/object-link-types/shared-property-overview/>
- [103] Palantir Technologies, 「Palantir Foundry docs, The Ontology system」 . 2026 年. 参照: 2026 年 10 月 3 日. [Online]. 入手先: <https://www.palantir.com/docs/foundry/architecture-center/ontology-system/>
- [104] B. Stancil, 「The missing piece of the modern data stack」 . 2021 年 4 月. 参照: 2026 年 10 月 3 日. [Online]. 入手先: <https://benn.substack.com/p/metrics-layer>
- [105] dbt Labs, 「dbt Labs docs, dbt Semantic Layer」 . 2026 年. 参照: 2026 年 10 月 3 日. [Online]. 入手先: <https://docs.getdbt.com/docs/use-dbt-semantic-layer/dbt-sl>
- [106] A. Hogan, 「The Semantic Web: Two Decades On」 , Semantic Web, vol. 11, no. 1, 2020, 参照: 2026 年 10 月 3 日. [Online]. 入手先: <https://semantic-web-journal.net/system/files/swj2303.pdf>
- [107] G. Anadiotis, 「Universal semantic layer: Going meta on data, functionality, governance, and semantics」 . 2024 年. 参照: 2026 年 10 月 3 日. [Online]. 入手先: <https://linkeddataorchestration.substack.com/p/universal-semantic-layer-going-meta>
- [108] Apache Ossie, 「Apache Ossie, Roadmap」 , Apache Software Foundation, 2026. 参照: 2026 年 10 月 3 日. [Online]. 入手先: <https://raw.githubusercontent.com/apache/ossie/main/ROADMAP.md>
- [109] Apache Ossie, 「Apache Ossie, OSI Community Update (2026-04-28)」 . 2026 年 4 月. 参照: 2026 年 10 月 2 日. [Online]. 入手先: <https://ossie.apache.org/updates/osi-april-2026-community-update/>
- [110] Java Community Process, 「Java Community Process, JSR 69: Java OLAP Interface (JOLAP)」 . 2012 年 4 月. 参照: 2026 年 10 月 4 日. [Online]. 入手先: <https://jcp.org/en/jsr/detail?id=69>
- [111] AtScale, 「AtScale, AtScale Announces the Open-Source Release of SML: A Standard Semantic Modeling Language (2024-09-10)」 . 2024 年 9 月. 参照: 2026 年 10 月 4 日. [Online]. 入手先: <https://www.atscale.com/press/atscale-announces-the-open-source-release-of-semantic-modeling-language/>
- [112] AtScale, 「semanticdatalayer/SML, Semantic Modeling Language (GitHub)」 . 2024 年 8 月. 参照: 2026 年 10 月 4 日. [Online]. 入手先: <https://github.com/semanticdatalayer/SML>
- [113] S. McVeety と A. Hormati, 「Google Cloud Blog, Open Knowledge format v0.2 tackles agentic trust (2026-07-24)」 . 2026 年 7 月. 参照: 2026 年 10 月 4 日. [Online]. 入手先: <https://cloud.google.com/blog/products/data-analytics/okf-v0-2-adds-trust-signals>
- [114] Google Cloud, 「Open Knowledge Format (OKF) specification, Version 0.2 (GoogleCloudPlatform/open-knowledge-format, SPEC.md)」 , Google Cloud, 8 月 2026. 参照: 2026 年 10 月 4 日. [Online]. 入手先: <https://raw.githubusercontent.com/GoogleCloudPlatform/open-knowledge-format/main/SPEC.md>
- [115] Google Cloud, 「Open Knowledge Format (OKF), README (GoogleCloudPlatform/open-knowledge-format)」 . 2026 年 8 月. 参照: 2026 年 10 月 4 日. [Online]. 入手先: <https://raw.githubusercontent.com/GoogleCloudPlatform/open-knowledge-format/main/README.md>
- [116] Eclipse Dataspace Working Group, 「Dataspace Protocol 2025-1, dspace.jsonld (JSON-LD コンテキスト)」 . 2025 年 8 月. 参照: 2026 年 10 月 3 日. [Online]. 入手先: <https://raw.githubusercontent.com/eclipse-dataspace-protocol-base/DataspaceProtocol/2025-1/artifacts/src/main/resources/context/dspace.jsonld>

- [117] Institute for Standardization of Serbia, 「Institute for Standardization of Serbia, ISO/IEC DIS 26450 project record (ISO status mirror)」. 2026 年 9 月. 参照: 2026 年 10 月 2 日. [Online]. 入手先: <https://iss.rs/en/project/show/iso:proj:93502>
- [118] Eclipse Dataspace Working Group, 「Dataspace Protocol README (tag 2025-1)」. 2025 年 8 月. 参照: 2026 年 10 月 3 日. [Online]. 入手先: <https://raw.githubusercontent.com/eclipse-dataspace-protocol-base/DataspaceProtocol/2025-1/README.md>
- [119] International Data Spaces Association, 「IDSA ids-specification README (tag v0.8, Working Draft 2023-02-01)」, IDSA, 2 月 2023. 参照: 2026 年 10 月 3 日. [Online]. 入手先: <https://raw.githubusercontent.com/International-Data-Spaces-Association/ids-specification/v0.8/README.md>
- [120] International Data Spaces Association, 「IDS-RAM 4.0, Information Layer」, IDSA, 2023. 参照: 2026 年 10 月 3 日. [Online]. 入手先: https://raw.githubusercontent.com/International-Data-Spaces-Association/IDS-RAM_4_0/main/documentation/3_Layers_of_the_Reference_Architecture_Model/3_3_Information_Layer/3_3_InformationLayer.md
- [121] International Data Spaces Association, 「IDSA ids-specification, Dataspace Protocol 2024-1, catalog.protocol.md」, IDSA, 6 月 2024. 参照: 2026 年 10 月 4 日. [Online]. 入手先: <https://raw.githubusercontent.com/International-Data-Spaces-Association/ids-specification/2024-1/catalog/catalog.protocol.md>
- [122] International Data Spaces Association, 「IDSA ids-specification, Dataspace Protocol 2024-1, common.protocol.md」, IDSA, 6 月 2024. 参照: 2026 年 10 月 3 日. [Online]. 入手先: <https://raw.githubusercontent.com/International-Data-Spaces-Association/ids-specification/2024-1/common/common.protocol.md>
- [123] J. Marino, 「Proposal: Adopt JSON Schema for normative serialization (Dataspace-Protocol issue 25)」. 2024 年 9 月. 参照: 2026 年 10 月 3 日. [Online]. 入手先: <https://api.github.com/repos/eclipse-dataspace-protocol-base/DataspaceProtocol/issues/25>
- [124] International Data Spaces Association, 「GitHub API: IDS InformationModel repo metadata」. 2023 年. 参照: 2026 年 10 月 2 日. [Online]. 入手先: <https://api.github.com/repos/International-Data-Spaces-Association/InformationModel>
- [125] International Data Spaces Association, 「IDS Information Model, Contract.ttl (GitHub)」. 2022 年. 参照: 2026 年 10 月 2 日. [Online]. 入手先: <https://raw.githubusercontent.com/International-Data-Spaces-Association/InformationModel/develop/model/contract/Contract.ttl>
- [126] International Data Spaces Association, 「IDS Information Model, Resource.ttl (GitHub)」. 2022 年. 参照: 2026 年 10 月 2 日. [Online]. 入手先: <https://raw.githubusercontent.com/International-Data-Spaces-Association/InformationModel/develop/model/content/Resource.ttl>
- [127] Catena-X Automotive Network, 「Catena-X CX-0003 SAMM Aspect Meta Model」, Catena-X, 2026. 参照: 2026 年 10 月 2 日. [Online]. 入手先: <https://catenax-ev.github.io/docs/standards/CX-0003-SAMMSemanticAspectMetaModel>
- [128] 情報処理推進機構, 「IPA プレス発表 2026-04-01 Open Data Spaces の成果物を公開」. 2026 年 4 月. 参照: 2026 年 10 月 2 日. [Online]. 入手先: <https://www.ipa.go.jp/pressrelease/2026/press20260401.html>
- [129] 情報処理推進機構デジタルアーキテクチャ・デザインセンター, 「Why Open Dataspaces: 設計思想とアーキテクチャパラダイム」, IPA, 4 月 2026. 参照: 2026 年 10 月 3 日. [Online]. 入手先: https://www.ipa.go.jp/digital/architecture/documents/rculhd000000chgw-att/WhyOpenDataspaces_jp.pdf
- [130] Open Data Spaces, 「open-dataspaces, organization profile README」. 2026 年 7 月. 参照: 2026 年 10 月 3 日. [Online]. 入手先: <https://raw.githubusercontent.com/open-dataspaces/.github/main/profile/README.md>

- [131] Open Data Spaces, 「open-dataspaces/SDK-for-semantics, README」 . 2026 年 3 月. 参照: 2026 年 10 月 3 日. [Online]. 入手先: <https://raw.githubusercontent.com/open-dataspaces/SDK-for-semantics/main/README.md>
- [132] Open Data Spaces, 「open-dataspaces/L4-sparql-server-sample, ontology/uasl-core-2026-01-06.ttl」 . 2026 年 1 月. 参照: 2026 年 10 月 3 日. [Online]. 入手先: <https://raw.githubusercontent.com/open-dataspaces/L4-sparql-server-sample/main/ontology/uasl-core-2026-01-06.ttl>
- [133] Open Data Spaces, 「open-dataspaces/L4-sparql-server-sample, README」 . 2026 年 3 月. 参照: 2026 年 10 月 3 日. [Online]. 入手先: <https://raw.githubusercontent.com/open-dataspaces/L4-sparql-server-sample/main/README.md>
- [134] Open Data Spaces, 「open-dataspaces/L4-crawler-service, README」 . 2026 年 3 月. 参照: 2026 年 10 月 3 日. [Online]. 入手先: <https://raw.githubusercontent.com/open-dataspaces/L4-crawler-service/main/README.md>
- [135] Open Data Spaces, 「open-dataspaces/L4-uasl-chat-server, README」 . 2026 年 3 月. 参照: 2026 年 10 月 3 日. [Online]. 入手先: <https://raw.githubusercontent.com/open-dataspaces/L4-uasl-chat-server/main/README.md>
- [136] Model Context Protocol, 「Model Context Protocol, What is the Model Context Protocol (MCP)?」 2026 年. 参照: 2026 年 10 月 4 日. [Online]. 入手先: <https://modelcontextprotocol.io/docs/getting-started/intro>
- [137] W3C, 「W3C, Data on the Web Best Practices (W3C Recommendation, 2017-01-31)」 , W3C, 1 月 2017. 参照: 2026 年 10 月 4 日. [Online]. 入手先: <https://www.w3.org/TR/dwbp/>

14 付録 B : 改変履歴

公開した版ごとの主な変更を、新しい順に示す。版は日付で表す。

表 31: 改変履歴

版	主な変更
2026-10-04.5	PDF のファイル名に版を入れた。ページの冒頭に、PDF をダウンロードするリンクを置いた。内容の変更はない
2026-10-04.4	PDF 版を公開物に加えた。調査報告と前提文書のページから、PDF をダウンロードできる。言い回しを見直した(宙に浮いた指示語、比喩、同じ主張の繰り返しなど)。内容の変更はない
2026-10-04.3	分析系のコラムに、過去の共通化の試み(CWM、JOLAP、RDF Data Cube 語彙、SML)と、今回との違いを加えた。年表に二つの出来事を足した。説明ペーパーの該当のスライドを直した
2026-10-04.2	ライセンスを明記した。本文、図、スライドは CC BY 4.0、スクリプトは Apache License 2.0 とした。改変履歴を付録 B として加えた
2026-10-04	一般公開した最初の版。調査報告、前提文書、初心者向けの説明ペーパー(48 枚)を公開した