

sitBCN: Knowledge Graphs with SAP Data

SULZER ESPAÑA



Barcelona, 22.11.2024 – Antonio Leites

sitBCN: Knowledge Graphs with SAP Data



Agenda



- **Introduction to Knowledge Graphs**
- **Building a Knowledge Graph**
- **SAP and Knowledge Graphs**

sitBCN: Knowledge Graphs with SAP Data



Agenda



Introduction to Knowledge Graphs

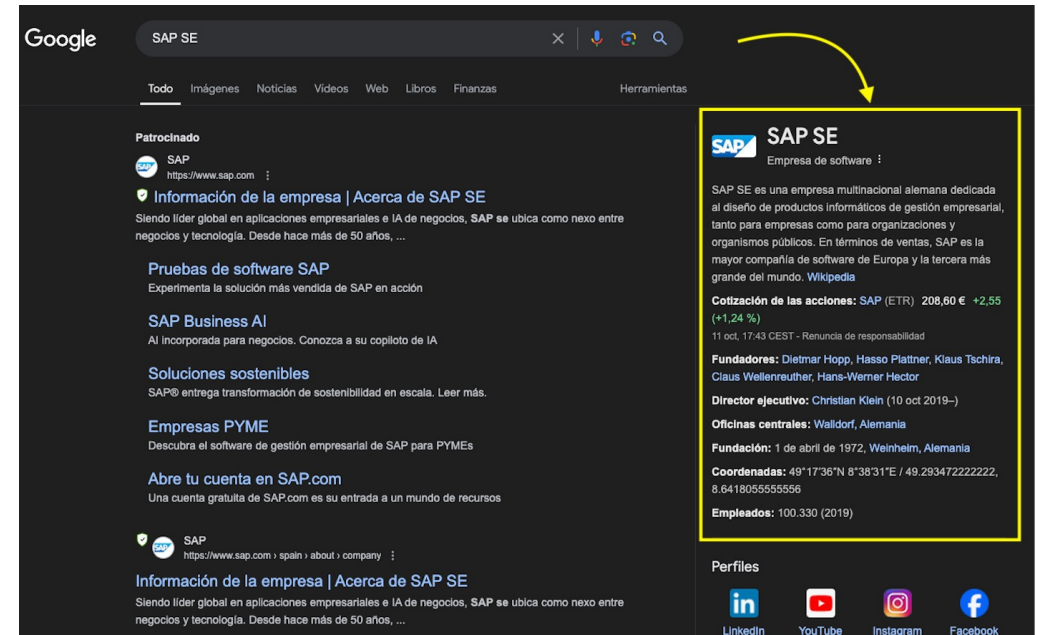
Building a Knowledge Graph

SAP and Knowledge Graphs

Knowlegde Graphs

WEB SEARCH EXPERIENCE

- The term “**knowledge graph**” was introduced in 2012 by Google
- Today, knowledge graphs are a common part of our daily Web search experience.
- Search results not only include a list of Web pages with the string “SAP SA”, but also provide a **structured** and **interconnected network of information** about the *entity* “SAP SA”.
- Key aspect of **knowledge graphs**: they encode “**things, not strings**”, that is: **entities**, their **properties** and **relations** to **other entities**.



Google

SAP SE

Todo Imágenes Noticias Videos Web Libros Finanzas Herramientas

Patrocinado

SAP
https://www.sap.com

Información de la empresa | Acerca de SAP SE
Siendo líder global en aplicaciones empresariales e IA de negocios, SAP se ubica como nexo entre negocios y tecnología. Desde hace más de 50 años, ...

Pruebas de software SAP
Experimenta la solución más vendida de SAP en acción

SAP Business AI
AI incorporada para negocios. Conozca a su copiloto de IA

Soluciones sostenibles
SAP® entrega transformación de sostenibilidad en escala. Leer más.

Empresas PYME
Descubra el software de gestión empresarial de SAP para PYMEs

Abre tu cuenta en SAP.com
Una cuenta gratuita de SAP.com es su entrada a un mundo de recursos

SAP
https://www.sap.com › spain › about › company

Información de la empresa | Acerca de SAP SE
Siendo líder global en aplicaciones empresariales e IA de negocios, SAP se ubica como nexo entre negocios y tecnología. Desde hace más de 50 años, ...

SAP SE
Empresa de software

SAP SE es una empresa multinacional alemana dedicada al diseño de productos informáticos de gestión empresarial, tanto para empresas como para organizaciones y organismos públicos. En términos de ventas, SAP es la mayor compañía de software de Europa y la tercera más grande del mundo. Wikipedia

Cotización de las acciones: SAP (ETR) 208,60 € +2,55 (+1,24 %)
11 oct, 17:43 CEST - Renuncia de responsabilidad

Fundadores: Dietmar Hopp, Hasso Plattner, Klaus Tschira, Claus Wellenreuther, Hans-Werner Hector

Director ejecutivo: Christian Klein (10 oct 2019–)

Oficinas centrales: Walldorf, Alemania

Fundación: 1 de abril de 1972, Weinheim, Alemania

Coordenadas: 49°17'36"N 8°38'31"E / 49.293472222222, 8.6418055555556

Empleados: 100.330 (2019)

Perfiles

LinkedIn YouTube Instagram Facebook

In the knowledge graph, you will see properties of the entity “SAP SE”, for example its description or a link to the SAP website. You will also see relations to other entities in the knowledge graph, for example headquarters of SAP. Clicking e.g. on “Walldorf” will lead you to the knowledge graph entry for Walldorf, including other types of properties and relations, like the population “14,646” and the “region” relation to the entity “Karlsruhe”.

Knowledge Representation with Graphs

FROM DATA TO KNOWLEDGE

Data:

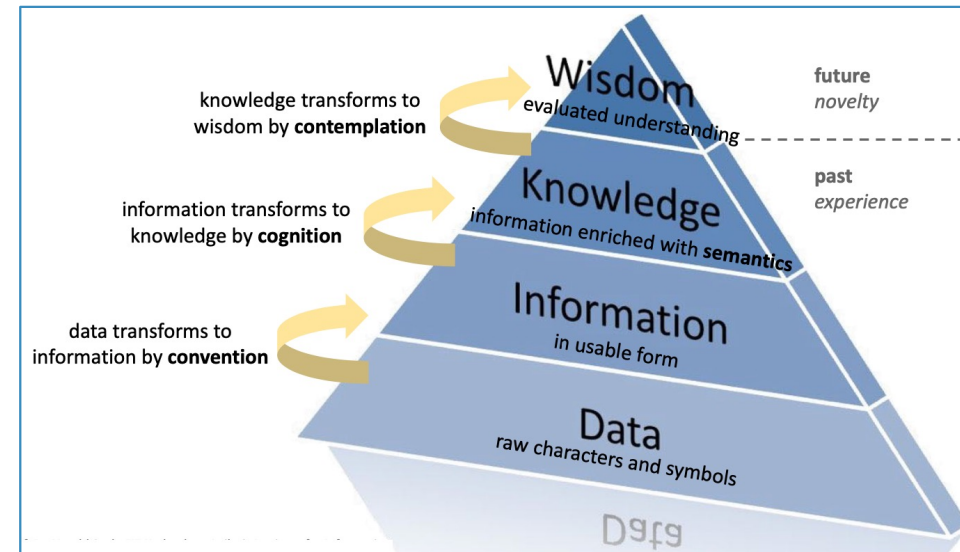
- Data is **raw**.
- It **simply exists** and has **no significance** beyond its existence (in and of itself).
- it can exist in any form, usable or not.

Information:

- Information is **data** that has been **given meaning** by way of **relational connection**.
- This "meaning" can be useful, but does not have to be.
- Information is **contained in descriptions**.
- Information answers to questions that begin with such words as **who, what, when, where, and how many**.

Knowledge:

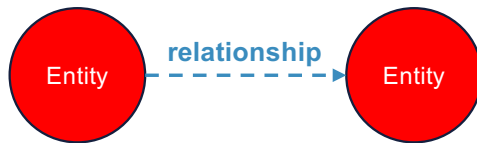
- **Knowledge** is the appropriate **collection of information**, such that its **intent** is to be **useful**.
- **Wisdom** is the ability to **make** sound judgments and **decisions**.
- **Understanding** is a continuum that leads **from data**, through **information** and **knowledge**, and ultimately to **wisdom**.



DIKW Pyramid, Ackoff 1989

Knowledge Graphs

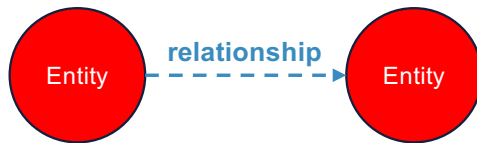
WHAT IS A KNOWLEDGE GRAPH?



- ✓ **Relationships:** Interconnections between entities. (**informations**)

Knowledge Graphs

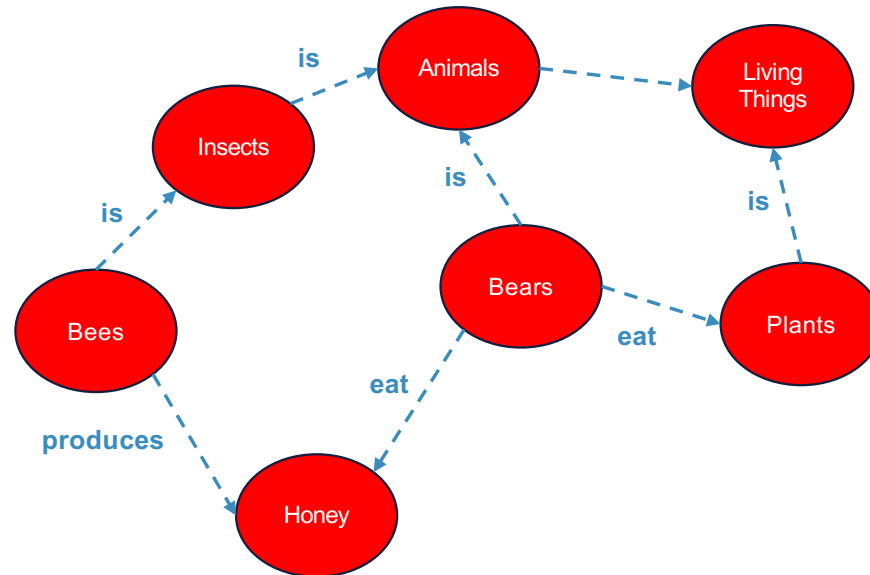
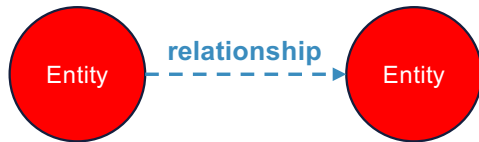
WHAT IS A KNOWLEDGE GRAPH?



- ✓ **Relationships:** Interconnections between entities. (**informations**)

Knowledge Graphs

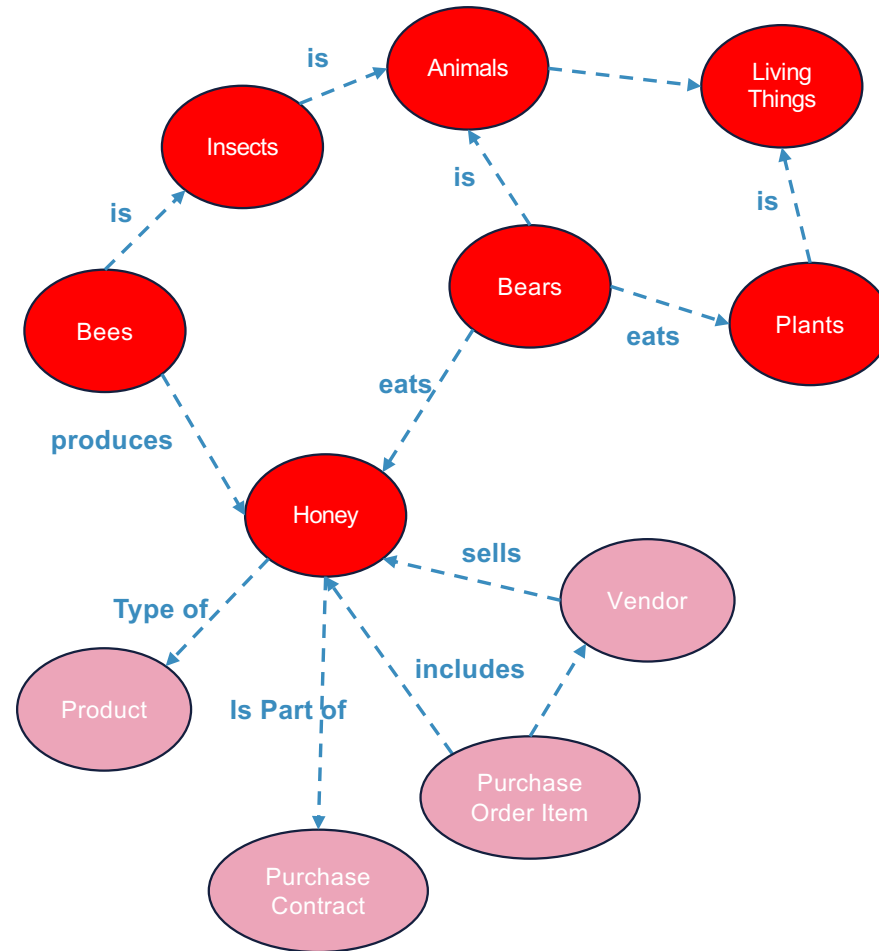
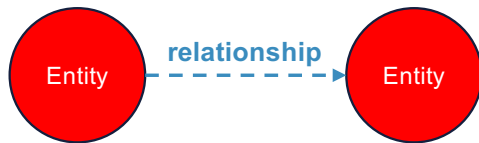
WHAT IS A KNOWLEDGE GRAPH?



- ✓ **Relationships:** Interconnections between entities. (**informations**)
- ✓ **Knowledge:** The whole relationships among entities

Knowledge Graphs

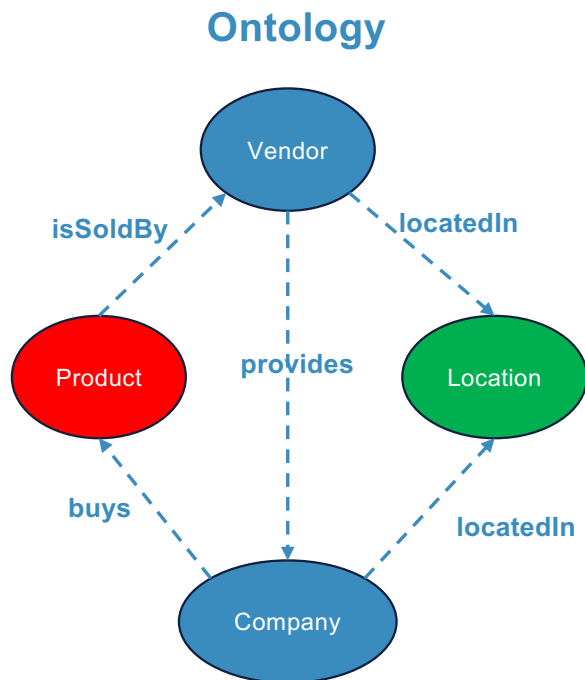
WHAT IS A KNOWLEDGE GRAPH?



- ✓ **Relationships:** Interconnections between entities. (**information**)
- ✓ **Knowledge:** The whole relationships between entities

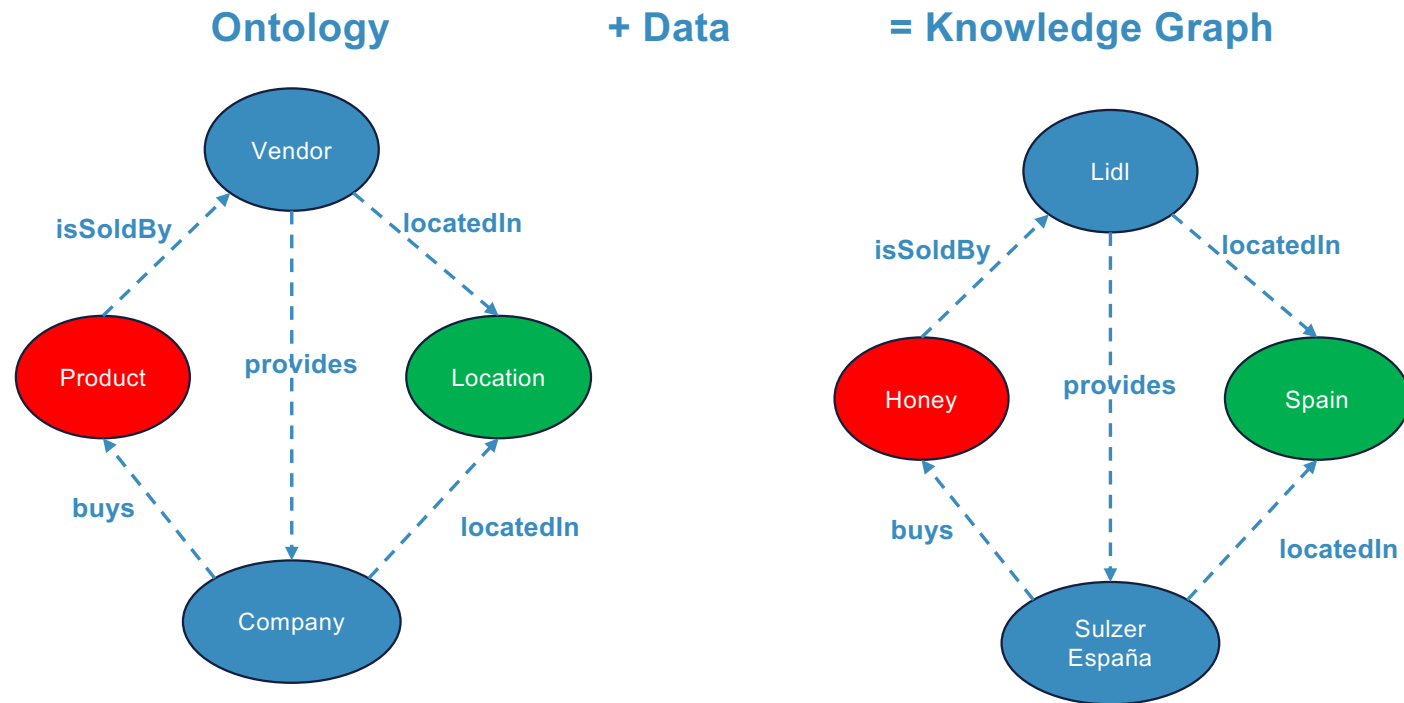
Knowledge Graphs: Ontology

HOW KNOWLEDGE GRAPHS CAN BE MODELLED?



Knowledge Graphs: Ontology

HOW KNOWLEDGE GRAPHS CAN BE MODELLED?



Triples: Semantic Facts
 <Subject, Predicate, Object>

- <Honey , isSoldBy, Lidl>.
- <Sulzer España , buys, Honey>.
- <Lidl , provides, Sulzer España>.
- <Lidl , locatedIn, Spain>.
- <Sulzer España , locatedIn, Spain>.

A knowledge graph is a structured representation of data (adhering to ontology rules)

Knowledge Graphs

A DIFFERENT APPROACH FOR DATA INTEGRATION

- **Classic Approach:**

- ✓ Use of **data warehouses** with **one unified schema**.

- **Knowledge Graphs:**

- ✓ there is **no common data model** across the data sources.
- ✓ For various types of entities like companies, persons or cities, the **knowledge graph models** are created in a **data** and **use case** driven manner.
- ✓ “All models are wrong, but some are useful”, what is said about models in statistics can also be said about modelling in knowledge graphs. There is **no perfect knowledge graph model** of an entity like “person” or “company”. There are **only models** that more or less **fit to given use cases**.

Knowledge Graphs

WHY USE KNOWLEDGE GRAPHS?

- **Complex Relationships:**

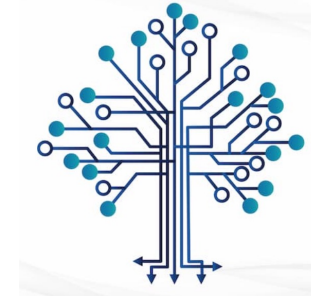
- ✓ Capture **rich semantic** connections that are difficult to express in **traditional databases**.

- **Flexibility and Agility:**

- ✓ Easily adapt to **evolving data** and changing **business requirements**.
- ✓ Add new **entities, relationships** and **properties** **without modifying** the underlying schema.

- **Efficient Querying:**

- ✓ Optimized queries involving multiple relationships and entities.





Agenda



Introduction to Knowledge Graphs



Building a Knowledge Graph



SAP and Knowledge Graphs

Knowlegde Graphs on linked business data

KNOWLEDGE GRAPH GENERATION PROCESS

- **Data Extraction**

- ✓ **Extract metadata** (for ABAP tables, business objects, CDS views etc.) from various SAP internal and external data sources

- **Modelling**

- ✓ Define **Target Model for the Use Case**. The model captures only types of entities and their required properties for the Use Case **using RDF**

- **Knowledge Graph Generation**

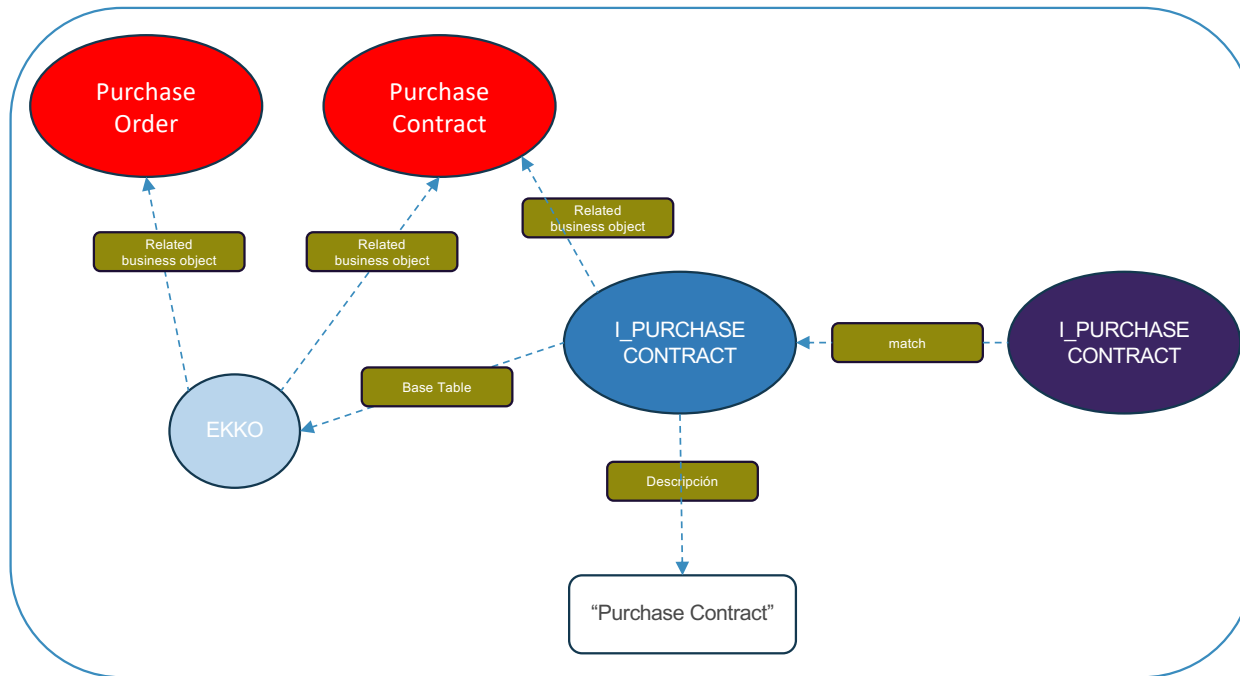
- ✓ Transform extracted data to the target model and upload the data to a Knowledge Graph DB

- **Knowledge Graph Provision**

- ✓ Provide the graph to applications via APIs.

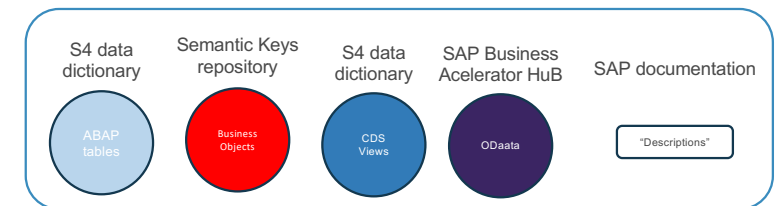
Knowledge Graphs on linked business data

EXAMPLE KNOWLEDGE GRAPH IMPLEMENTING AND ENABLING SEMANTIC ACCESS TO THE TABLE “EKKO”.



A knowledge graph contains entities, properties and relations.

Underlying metadata sources from different systems



The business objects, like the entities and relations have global and unique identifier, which enable the flexible integration of new (meta) data sources.

Key value of a knowledge graph: it stores entities in the graph.

The “related business object” relation, constitutes an extensible semantic access layer across the domain specific metadata items.

Knowlegde Graphs on linked business data

RDF AS THE TECHNICAL BASIS FOR KNOWLEDGE GRAPHS

Technical basis for **generating**, **storing** and **querying** knowledge graphs is the graph data model **RDF** (Resource Description Framework), a set of standards developed by the W3C



RDF Triple

```
1 # Prefixes for classes
2 @prefix abatable: <http://schema.sap.com/ABAPTable/> .
3 @prefix cdsview: <http://schema.sap.com/CDSView/> .
4 @prefix bo: <http://schema.sap.com/BusinessObject/> .
5 @prefix entitySet: <http://schema.sap.com/ODATAEntitySet/> .
6
7 # Prefixes for properties
8 @prefix s4: <http://schema.sap.com/ns/s4#> .
9 @prefix cds: <http://schema.sap.com/ns/cds#> .
10
11 # Prefixes for general RDF vocabularies, here only SKOS
12 @prefix skos: <http://www.w3.org/2004/02/skos/core#> .
13
14 abatable:EKKO s4:relatedBusinessObject bo:PurchaseOrder .
15 abatable:EKKO s4:relatedBusinessObject bo:PurchaseContract .
16
17 cdsview:I_PURCHASECONTRACT s4:relatedBusinessObject bo:PurchaseContract.
18
19 cdsview:I_PURCHASECONTRACT s4:baseTable abatable:EKKO.
20
21 cdsview:I_PURCHASECONTRACT cds:description "Purchase Contract".
22
23 entitySet:I_PURCHASECONTRACT skos:exactMatch cdsview:I_PURCHASECONTRACT.
```

Triples

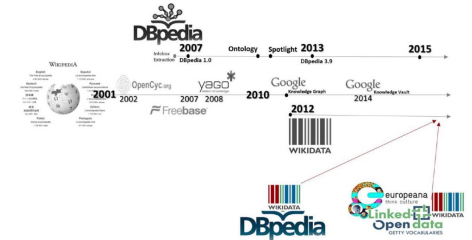
- The actual RDF data in the example starts after the prefix definition in line 14
 - **Subject** (in line 14) **abatable:EKKO**
 - **Predicate** (in line 14) **s4:relatedBusinessObject**
 - **Object** (in line 14) **bo:PurchaseOrder**

RDF Vocabularies

- **Vocabularies** are a **key means** to realize the knowledge graph capability of distinguishing “things from strings”.
- In the example, we have two different vocabularies: **cdsview:** and **entitySet:** and we can distinguish two different entities which have the same name: **I_PURCHASECONTRACT**
- RDF also allows to create formal models for vocabularies via technologies like OWL or SHACL.)
- Based on use case needs, we can create and extend our use our own, SAP specific vocabularies and combine them with public vocabularies like SKOS

Knowlegde Graphs on linked business data

SPARQL RDF QUERY LANGUAGE FOR KNOWLEDGE GRAPHS



SPARQL is based on RDF **Turtle** serialization and **basic graph pattern matching**.

- A **Graph Pattern** (Triple pattern) is a RDF Triple that contains variables at any arbitrary place (Subject, Property, Object)

Graph Pattern = Turtle + Variables

SPARQL Query & Update Editor only Editor and results Results only

Unnamed × SPARQL Select template × dbo:Film × sap:Vendor × sap:Material × sap:Plant × Unnamed × Unnamed ×

PurchaseOrder_450000002 × Purchase orders grouped × Infos from PO Items (material, pl... ×

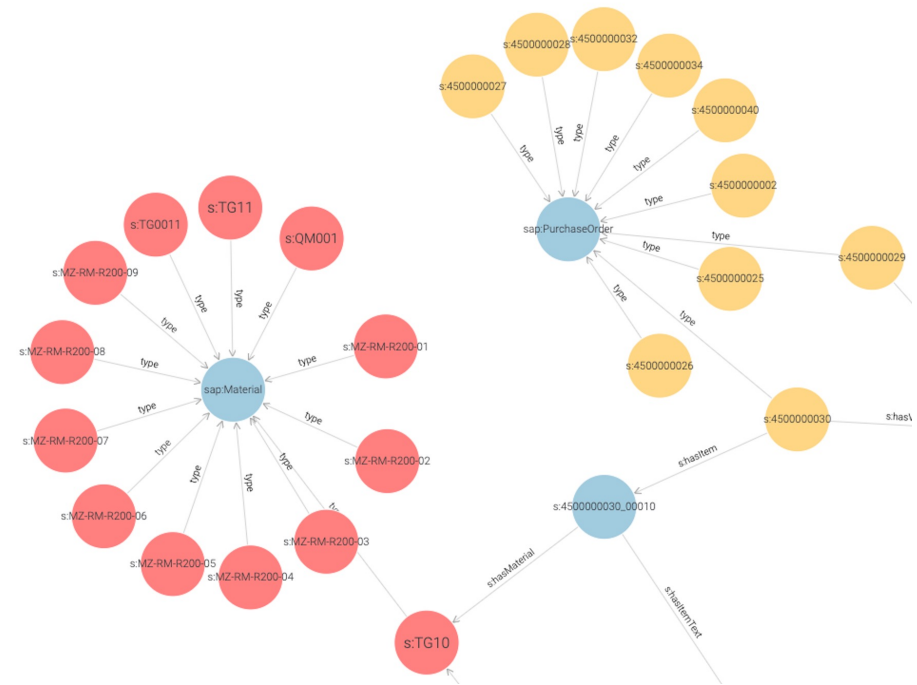
```

1 PREFIX s: <http://sulzer.es/sap-modeU#>
2 PREFIX sap: <http://sulzer.es/sap-ontology#>
3 PREFIX rdf: <http://www.w3.org/1999/02/22-rdf-syntax-ns#>
4
5 SELECT ?r ?y
6 WHERE { <s:450000002> ?r ?y .
7       FILTER (?r != <s:hasItem>)
8     }
9
10
11

```

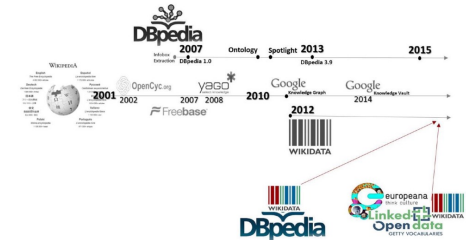
Run

Table	Raw Response	Pivot Table	Google Chart
Download as			
Filter query results			
Showing results from 1 to 5 of 5. Query took 1.9s, moments ago.			
	r		y
1	rdf:type		sap.PurchaseOrder
2	shasVendor		s.USSU-VSF02
3	shasDate		"20171008"
4	s.vendorName		"WaveCrest Labs"
5	shasDocumentCurrency		"USD"



Knowlegde Graphs on linked business data

SPARQL RDF QUERY LANGUAGE FOR KNOWLEDGE GRAPHS



SPARQL is based on RDF **Turtle** serialization and **basic graph pattern matching**.

- A **Graph Pattern** (Triple pattern) is a RDF Triple that contains variables at any arbitrary place (Subject, Property, Object)

Graph Pattern = Turtle + Variables

```

SPARQL Query & Update
Editor only Editor and results Results only

Purchase orders grouped | Infos from PO Items (material, pl... | sap.PurchaseOrder | sap.PurchaseOrder | sap.Plant | +

1 PREFIX s: <http://sulzer.es/sap-model#>
2 PREFIX sap: <http://sulzer.es/sap-ontology#>
3 PREFIX rdf: <http://www.w3.org/1999/02/22-rdf-syntax-ns#>
4
5 SELECT ?deliveryCity ?vendorCity ?vendorName ?purchaseOrder ?material ?materialDescription ?quantity ?unit
6 WHERE {
7   ?purchaseOrder a <sap:PurchaseOrder> .
8   ?purchaseOrder <s:hasItem> ?poItem .
9   ?poItem <s:receivingPlant> ?plant .
10  ?plant <s:hasAddress> ?address .
11  ?address <s:city> ?deliveryCity .
12
13  # Link to the vendor and retrieve the vendor's city
14  ?purchaseOrder <s:hasVendor> ?vendor .
15  ?vendor <s:city> ?vendorCity .
16  ?vendor <s:vendorName> ?vendorName .
17
18  # Link to the material details
19  ?poItem <s:hasMaterial> ?material .
20  ?material <s:hasDescription> ?materialDescription .
21
22  # Retrieve the quantity and unit of measure for the item
23  ?poItem <s:hasQuantity> ?quantity .
24  ?material <s:hasBaseUnit> ?unit .
25
26 }
27
28 GROUP BY ?purchaseOrder ?deliveryCity ?vendorName ?vendorCity ?material ?materialDescription ?quantity ?unit
29 ORDER BY ?deliveryCity ?vendorName
30
Run
    
```

	deliveryCity	vendorCity	vendorName	purchaseOrder	material	materialDescription	quantity	unit
1	'Palo Alto'	'Blacksburg'	'Domestic US Subcontractor A'	s:4500000032	s:TG11	'Trad. Good 11,PD,Reg.Trading'	'3.000'	'ST'
2	'Palo Alto'	'Muncie'	'Domestic US Supplier 1'	s:4500000030	s:TG10	'Trad. Good 10,PD,Third Party'	'3.000'	'ST'
3	'Palo Alto'	'Bismarck'	'Domestic US Supplier 2'	s:4500000028	s:TG0011	'Trading Good 0011,PD,Regular Proc.'	'3.000'	'ST'
4	'Palo Alto'	'Bismarck'	'Domestic US Supplier 2'	s:4500000034	s:TG10	'Trad. Good 10,PD,Third Party'	'3.000'	'ST'
5	'Palo Alto'	'Wichita'	'Domestic US Supplier 6 (Returns)'	s:4500000027	s:QM001	'QM Regular'	'3.000'	'ST'
6	'Palo Alto'	'Wichita'	'Domestic US Supplier 6 (Returns)'	s:4500000040	s:QM001	'QM Regular'	'3.000'	'ST'
7	'Palo Alto'	'Norwalk'	'WaveCrest Labs'	s:4500000002	s:MZ-RM-R200-06	'BKR-200 Brakes'	'3.000'	'ST'
8	'Palo Alto'	'Norwalk'	'WaveCrest Labs'	s:4500000002	s:MZ-RM-R200-08	'BKR-200 Pedal Kit'	'3.000'	'ST'
9	'Palo Alto'	'Norwalk'	'WaveCrest Labs'	s:4500000002	s:MZ-RM-R200-07	'BKR-200 Derailleur Gears'	'3.000'	'ST'
10	'Palo Alto'	'Norwalk'	'WaveCrest Labs'	s:4500000002	s:MZ-RM-R200-02	'BKR-200 Handle Bars'	'3.000'	'ST'
11	'Palo Alto'	'Norwalk'	'WaveCrest Labs'	s:4500000002	s:MZ-RM-R200-03	'BKR-200 Seat'	'3.000'	'ST'
12	'Palo Alto'	'Norwalk'	'WaveCrest Labs'	s:4500000002	s:MZ-RM-R200-09	'BKR-200 Drive Train'	'3.000'	'ST'
13	'Palo Alto'	'Norwalk'	'WaveCrest Labs'	s:4500000002	s:MZ-RM-R200-01	'BKR-200 Frame'	'3.000'	'ST'
14	'Palo Alto'	'Norwalk'	'WaveCrest Labs'	s:4500000002	s:MZ-RM-R200-04	'BKR-200 Wheels'	'3.000'	'ST'
15	'Palo Alto'	'Norwalk'	'WaveCrest Labs'	s:4500000002	s:MZ-RM-R200-05	'BKR-200 Forks'	'3.000'	'ST'

Knowledge Graphs: simplified example

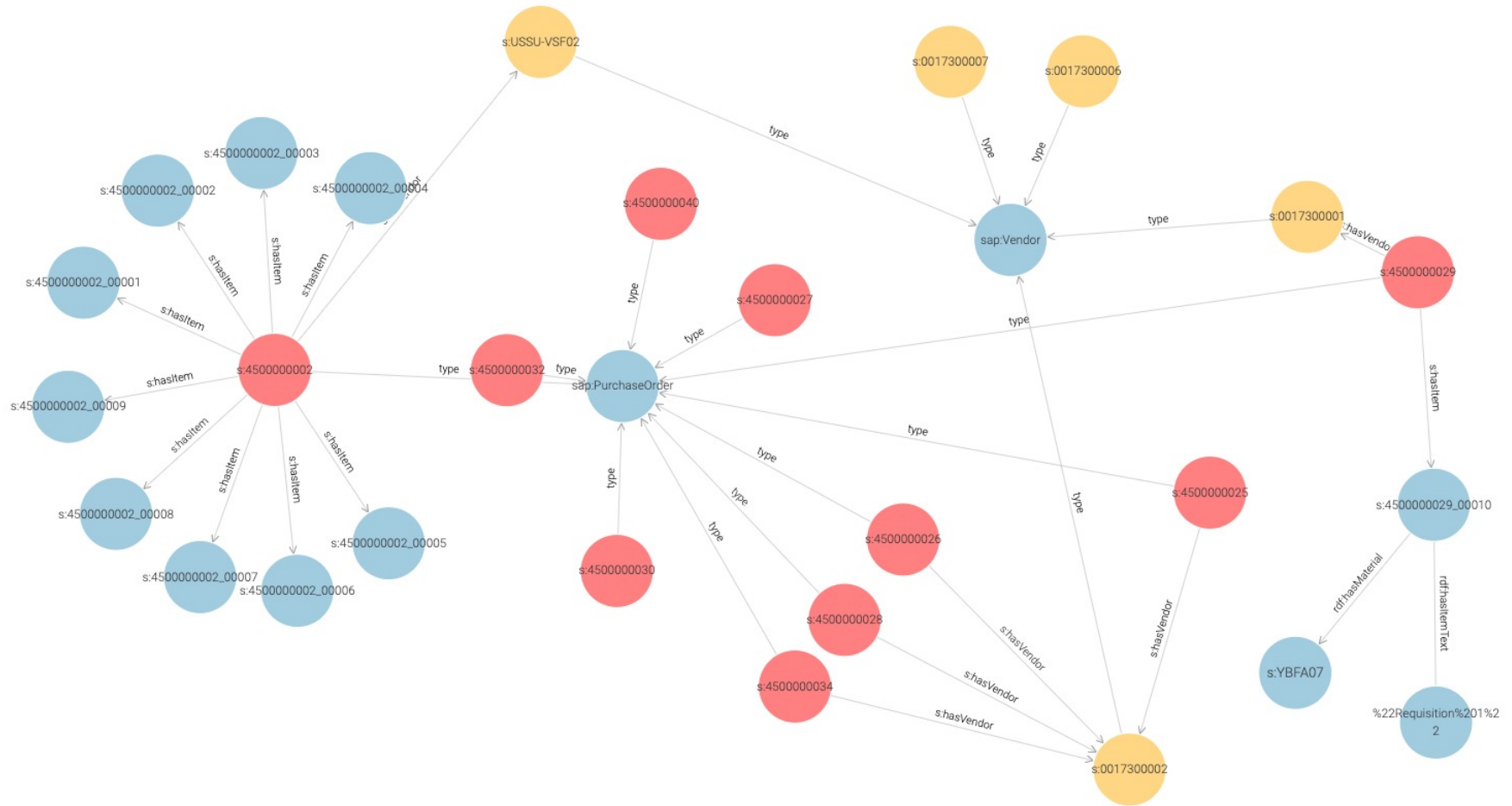
ABAP REPORT EXTRACTING RDF-TRIPLES

```
1 Z_CREATE_RDF_TRIPLE
2 *# Report z_create_rdf_triple
3 *#
4 REPORT z_create_rdf_triple line-size 180.
5
6 DATA: lt_purchase_order TYPE TABLE OF i_purchaseorder,
7       lt_purchase_order_item TYPE TABLE OF i_purchaseorderitem,
8       lt_material TYPE TABLE OF i_material,
9       ls_material_text TYPE i_material_text,
10      lt_supplier TYPE TABLE OF i_supplier,
11      ls_supplier TYPE i_supplier,
12      lv_subject TYPE string,
13      lv_predicate TYPE string,
14      lv_object TYPE string,
15      lv_triple TYPE string,
16      lv_filename TYPE string VALUE '/tmp/rdf_triples.ttl',
17      lv_count TYPE i VALUE 0,
18      lv_quantity TYPE string,
19      lv_netamount TYPE string.
20
21 DATA: lt_summary TYPE TABLE OF string,
22       lv_summary TYPE string.
23
24 * Open the dataset (file) for writing
25 OPEN DATASET lv_filename FOR OUTPUT IN TEXT MODE ENCODING DEFAULT.
26
27 * Write the prefixes at the beginning of the file
28 TRANSFER @prefix s: <http://sulzer-es/sap-mode/ > . ' TO lv_filename. "Instances
29 TRANSFER @prefix sap: <http://sulzer-es/sap-ontology# > . ' TO lv_filename. "Ontology
30 TRANSFER @prefix rdf: <http://www.w3.org/1999/02/22-rdf-syntax-ns# > . ' TO lv_filename.
31 TRANSFER @prefix rdfs: <http://www.w3.org/2000/01/rdf-schema# > . ' TO lv_filename.
32 TRANSFER @prefix xsd: <http://www.w3.org/2001/XMLSchema# > . ' TO lv_filename.
33 TRANSFER @prefix owl: <http://www.w3.org/2002/07/owl# > . ' TO lv_filename.
34 TRANSFER @prefix ex: <http://example.org/sap-mode/ > . ' TO lv_filename.
35
36 TRANSFER '' TO lv_filename. " Blank line for readability
37
38 * Define RDF Classes and subclass relationships
39 TRANSFER 'sap:Document rdf:type owl:Class ; rdfs:subClassOf sap:Document . ' TO lv_filename.
40 TRANSFER 'sap:PurchaseDocument rdf:type owl:Class ; rdfs:subClassOf sap:Document . ' TO lv_filename.
41 TRANSFER 'sap:MasterData rdf:type owl:Class . ' TO lv_filename.
42 TRANSFER 'sap:Vendor rdf:type owl:Class ; rdfs:subClassOf sap:MasterData . ' TO lv_filename.
43 TRANSFER 'sap:Material rdf:type owl:Class ; rdfs:subClassOf sap:MasterData . ' TO lv_filename.
44 TRANSFER 'sap:PurchaseOrder rdf:type owl:Class ; rdfs:subClassOf sap:PurchaseDocument . ' TO lv_filename.
45 TRANSFER 'sap:PurchaseOrderItem rdf:type owl:Class ; rdfs:label "Purchase Order Item" ; rdfs:subClassOf sap:PurchaseOrder . ' TO lv_filename.
46
47 TRANSFER '' TO lv_filename. " Blank line for readability
48
49 * Define RDF Properties
50 TRANSFER 's:hasVendor a rdf:Property ; rdfs:label "hasVendor" . ' TO lv_filename.
51 TRANSFER 's:hasVendor rdfs:domain sap:PurchaseOrder ; rdfs:range sap:Vendor . ' TO lv_filename.
52 TRANSFER 's:hasDate a rdf:Property ; rdfs:label "has Date" ; rdfs:comment "Represents the date associated with a PurchaseOrder" . ' TO lv_filename.
53 TRANSFER 's:hasDate rdfs:domain sap:PurchaseOrder ; rdfs:range xsd:date . ' TO lv_filename.
54 TRANSFER 's:vendorName a rdf:Property ; rdfs:label "vendor name" ; rdfs:comment "Represents the name of the Vendor associated with a PurchaseOrder or Vendor entity" . ' TO lv_filename.
55 TRANSFER 's:vendorName rdfs:domain sap:Vendor ; rdfs:range xsd:string . ' TO lv_filename.
56 TRANSFER 's:country a rdf:Property ; rdfs:label "country" ; rdfs:comment "Represents the country associated with a Vendor entity" . ' TO lv_filename.
57 TRANSFER 's:country rdfs:domain sap:Vendor ; rdfs:range xsd:string . ' TO lv_filename.
58 TRANSFER 's:city a rdf:Property ; rdfs:label "city" ; rdfs:comment "Represents the city associated with a Vendor entity" . ' TO lv_filename.
59 TRANSFER 's:city rdfs:domain sap:Vendor ; rdfs:range xsd:string . ' TO lv_filename.
60 TRANSFER 's:hasItem a rdf:Property ; rdfs:label "hasItem" ; rdfs:comment "Associates a Purchase Order with one of its items" . ' TO lv_filename.
61 TRANSFER 's:hasItem rdfs:domain sap:PurchaseOrder ; rdfs:range rdfs:Resource ; owl:propertyChainAxiom ( sap:PurchaseOrder sap:hasItem ) . ' TO lv_filename.
62 TRANSFER 's:hasMaterial a rdf:Property ; rdfs:label "hasMaterial" . ' TO lv_filename.
63 TRANSFER 's:hasMaterial rdfs:domain sap:PurchaseOrderItem ; rdfs:range sap:Material . ' TO lv_filename.
64 TRANSFER 's:hasItemText a rdf:Property ; rdfs:label "Item Text" ; rdfs:comment "Text associated to the PO Item" . ' TO lv_filename.
65 TRANSFER 's:hasItemText rdfs:domain sap:PurchaseOrderItem ; rdfs:range xsd:string . ' TO lv_filename.
66 TRANSFER 's:hasNetAmount a rdf:Property ; rdfs:label "hasNetAmount" ; rdfs:comment "Represents the net amount for a Purchase Order Item" . ' TO lv_filename.
67 TRANSFER 's:hasNetAmount rdfs:domain sap:PurchaseOrderItem ; rdfs:range xsd:decimal . ' TO lv_filename.
68 TRANSFER 's:hasCurrency a rdf:Property ; rdfs:label "hasCurrency" ; rdfs:comment "Represents the currency of the net amount for a Purchase Order Item" . ' TO lv_filename.
69 TRANSFER 's:hasCurrency rdfs:domain sap:PurchaseOrderItem ; rdfs:range xsd:string . ' TO lv_filename.
70 TRANSFER 's:hasDescription a rdf:Property ; rdfs:label "hasDescription" ; rdfs:comment "Represents a textual description or name for a material" . ' TO lv_filename.
71 TRANSFER 's:hasDescription rdfs:domain sap:Material ; rdfs:range xsd:string . ' TO lv_filename.
72 TRANSFER 's:hasBaseUnit a rdf:Property ; rdfs:label "hasBaseUnit" ; rdfs:comment "Represents the base unit of measure for a material" . ' TO lv_filename.
73 TRANSFER 's:hasBaseUnit rdfs:domain sap:Material ; rdfs:range xsd:string . ' TO lv_filename.
74 TRANSFER 's:hasQuantity a rdf:Property ; rdfs:label "hasQuantity" ; rdfs:comment "Represents the quantity of an item" . ' TO lv_filename.
75 TRANSFER 's:hasQuantity rdfs:domain sap:PurchaseOrderItem ; rdfs:range xsd:decimal . ' TO lv_filename.
76
77 TRANSFER '' TO lv_filename. " Blank line for readability
78
```

```
1 Z_CREATE_RDF_TRIPLE
2 /*
3 * Select first 10 Purchase Orders from I_PURCHASEORDER
4 * SELECT * FROM i_purchaseorder INTO TABLE @lt_purchase_order UP TO 10 ROWS.
5
6 * Fetch vendor details using the I_Supplier CDS view
7 * SELECT * FROM i_supplier INTO TABLE @lt_supplier
8 * FOR ALL ENTRIES IN @lt_purchase_order
9 * WHERE supplier = @lt_purchase_order-supplier.
10
11 * Loop through Purchase Orders to create RDF triples
12 LOOP AT lt_purchase_order INTO DATA(ls_purchase_order).
13
14 * Triple to define the PurchaseOrder as an instance of PurchaseOrder
15 lv_subject = |s:| ls_purchase_order-purchaseorder |}.
16 lv_predicate = 'rdf:type'.
17 lv_object = 'sap:PurchaseOrder'.
18 lv_triple = |<| lv_subject |>|<| lv_predicate |>|<| lv_object |> .|.
19 TRANSFER lv_triple TO lv_filename.
20 ADD 1 TO lv_count.
21
22 * Triple for PurchaseOrder
23 lv_subject = |s:| ls_purchase_order-purchaseorder |}.
24 lv_predicate = 's:hasDate'.
25 lv_object = |"|" ls_purchase_order-PurchaseOrderDate |"|.
26 lv_triple = |<| lv_subject |>|<| lv_predicate |>|<| lv_object |> .|.
27 TRANSFER lv_triple TO lv_filename.
28 ADD 1 TO lv_count.
29
30 * Triple for Vendor
31 lv_predicate = 's:hasVendor'.
32 lv_object = |"|" ls_purchase_order-supplier |"|.
33 lv_triple = |<| lv_subject |>|<| lv_predicate |>|<| lv_object |> .|.
34 TRANSFER lv_triple TO lv_filename.
35 ADD 1 TO lv_count.
36
37 READ TABLE lt_summary INTO ls_supplier WITH KEY supplier = ls_purchase_order-supplier.
38 IF sy-subrc = 0.
39
40 * Vendor Name triple
41 lv_predicate = 's:vendorName'.
42 lv_object = |"|" ls_supplier-SupplierName |"|.
43 lv_triple = |<| lv_subject |>|<| lv_predicate |>|<| lv_object |> .|.
44 TRANSFER lv_triple TO lv_filename.
45 ADD 1 TO lv_count.
46
47 * Vendor Name triple
48 lv_subject = |s:| ls_purchase_order-supplier |}.
49 lv_predicate = 's:vendorName'.
50 lv_object = |"|" ls_supplier-SupplierName |"|.
51 lv_triple = |<| lv_subject |>|<| lv_predicate |>|<| lv_object |> .|.
52 TRANSFER lv_triple TO lv_filename.
53 ADD 1 TO lv_count.
54
55 * Vendor City triple
56 lv_predicate = 's:city'.
57 lv_object = |"|" ls_supplier-CityName |"|.
58 lv_triple = |<| lv_subject |>|<| lv_predicate |>|<| lv_object |> .|.
59 TRANSFER lv_triple TO lv_filename.
60 ADD 1 TO lv_count.
61
62 * Vendor Country triple
63 lv_predicate = 's:country'.
64 lv_object = |"|" ls_supplier-Country |"|.
65 lv_triple = |<| lv_subject |>|<| lv_predicate |>|<| lv_object |> .|.
66 TRANSFER lv_triple TO lv_filename.
67 ADD 1 TO lv_count.
68
69 * Triple to define the Vendor as an instance of Vendors
70 lv_predicate = 'rdf:type'.
71 lv_object = 'sap:Vendor'.
72 lv_triple = |<| lv_subject |>|<| lv_predicate |>|<| lv_object |> .|.
73 TRANSFER lv_triple TO lv_filename.
74 ADD 1 TO lv_count.
75
76 ENDIF.
77
78 * Summary information for the Purchase Order
79 CONCATENATE 'Purchase Order:' ls_purchase_order-purchaseorder 'Date:' ls_purchase_order-PurchaseOrderDate 'Vendor:' ls_purchase_order-supplier 's:
80 APPEND lv_summary TO lt_summary.
81
82 */
```

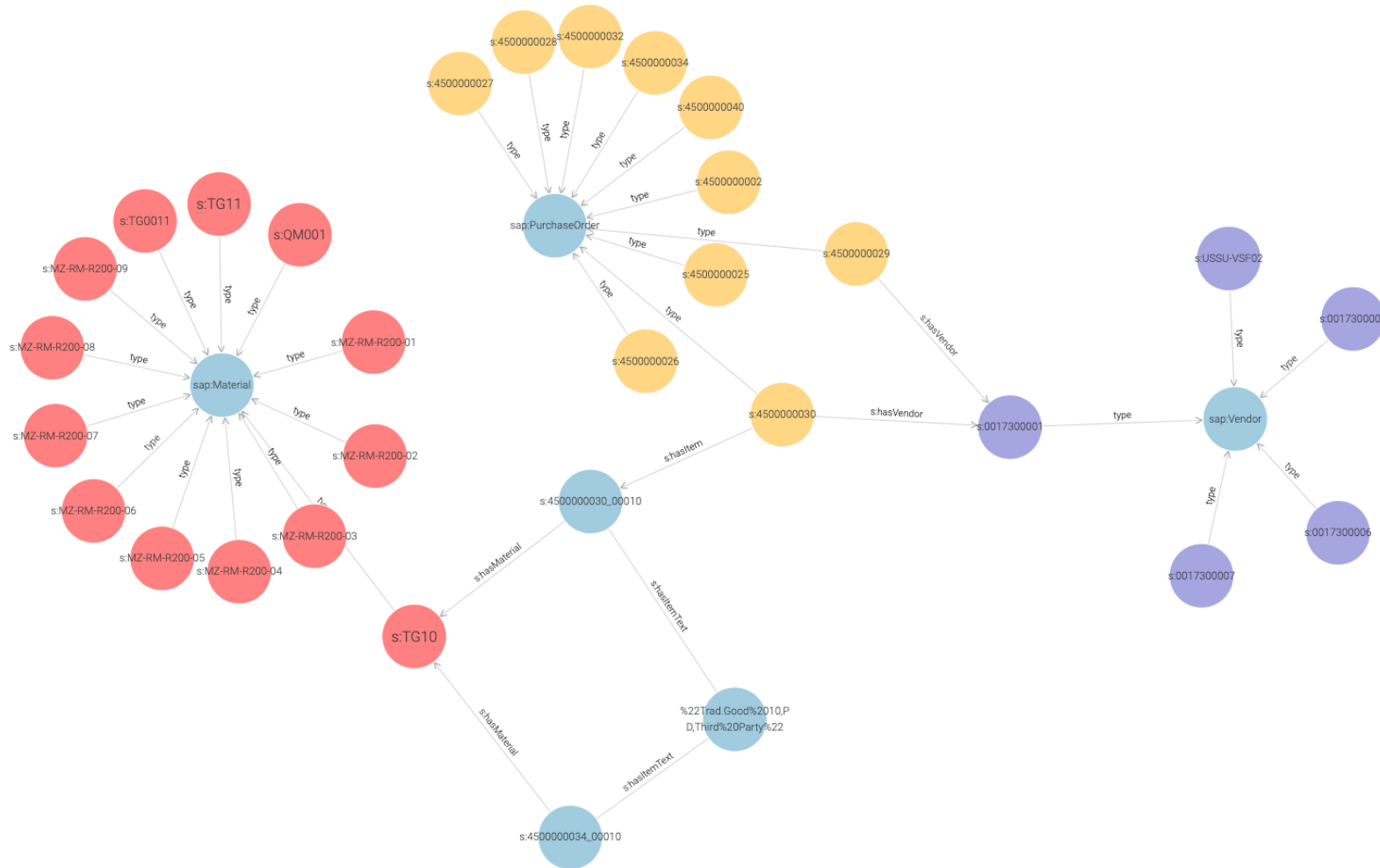
Knowlegde Graphs: simplified example

REPRESENTATION OF THE KNOWLEDGE GRAPH (GRAPHDB)



Knowlegde Graphs: simplified example

REPRESENTATION OF THE KNOWLEDGE GRAPH (GRAPHDB)





Agenda



Introduction to Knowledge Graphs



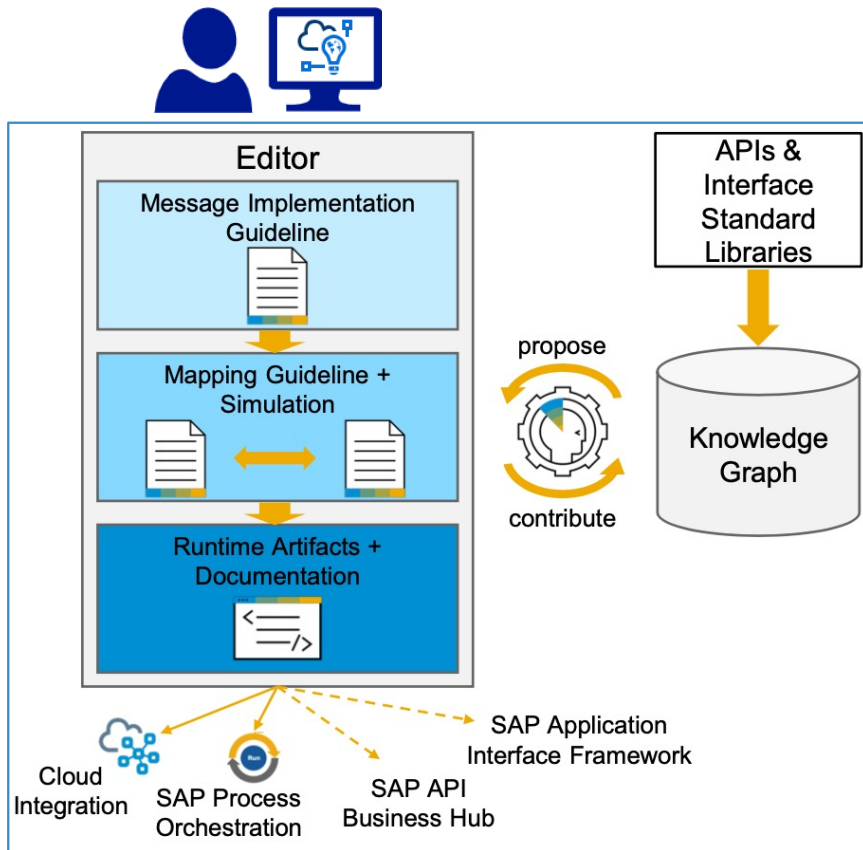
Building a Knowledge Graph



SAP and Knowledge Graphs

Knowlegde Graphs in SAP

SAP BTP: INTEGRATION ADVISOR



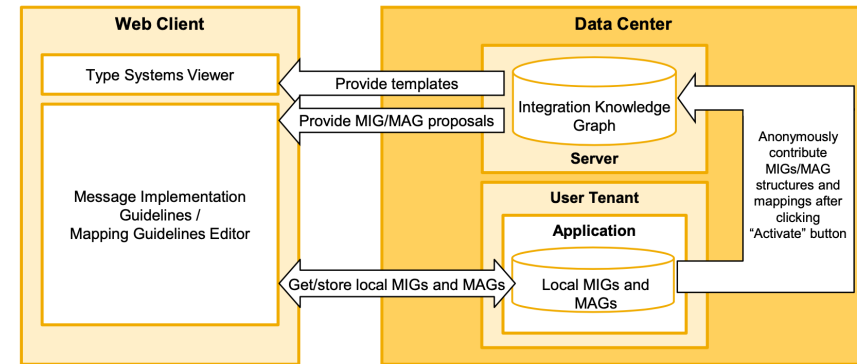
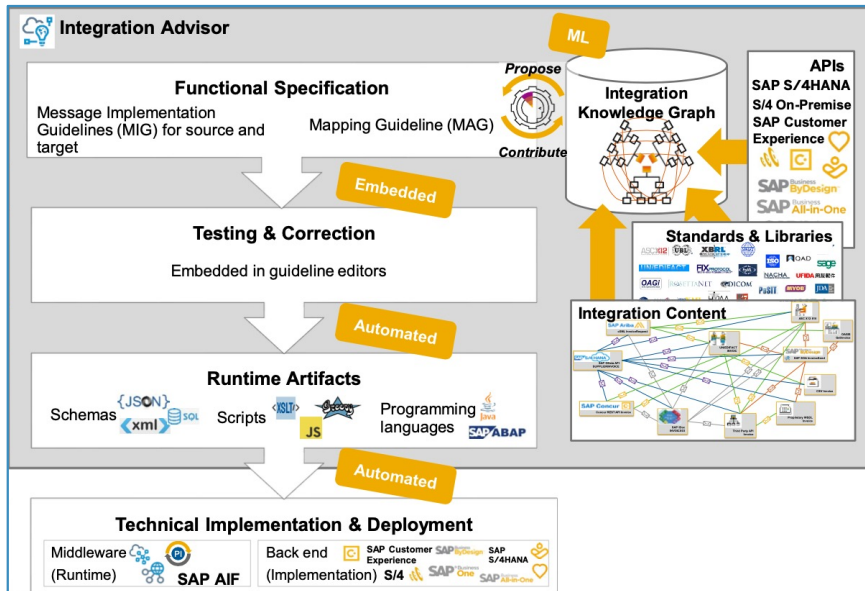
SAP Cloud Platform Integration Advisor drives a new and effective approach in realizing Integrations.

- Simplifies and reduces the effort to a greater extent.
- No need to have add-on tools to document the specifications, no worry about mismatch of specifications document versus artifacts,
- No separate tools for mapping and built-in **ever growing AI based knowledge graph**.

B2B implementation on SAP Cloud Platform Integration and Process Integration is empowered with Integration Advisor. The whole process is simplified and **SAP Cloud Platform Integration Advisor** is definitely an *Integration game changer!*

Knowledge Graphs in SAP

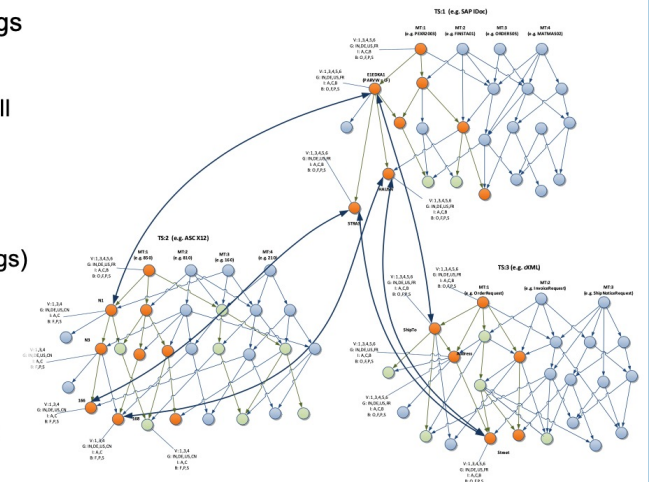
SAP BTP: INTEGRATION ADVISOR



Machine learning using crowd-sourced learnings

Key Capabilities

- **Complete libraries** in form of type systems and all their details with their precise semantics
- Based on **actual, productive** integration artifacts
- Collected by **contributions**
 - Cloud-based MIG/MAG editor
 - Importing existing integration content (mappings)
 - Importing payloads (instance data)
- Centrally provides the **real use of interfaces** of different type systems and how they have been mapped in different contexts
 - Focuses on semantics
 - Using transitivity closure and where/how often used across all libraries and contributions



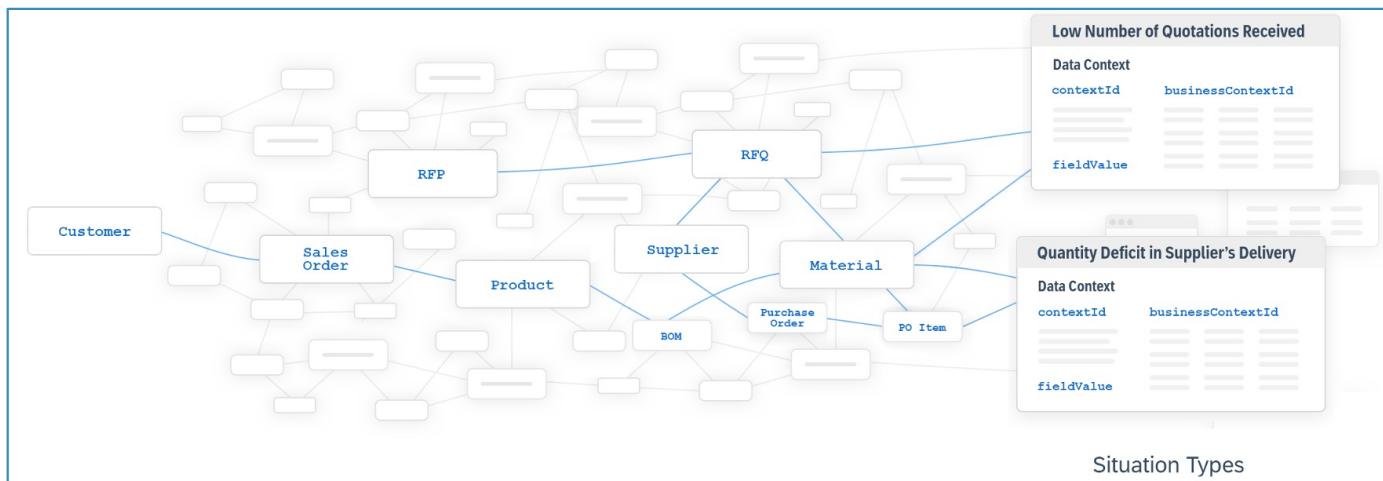
Knowledge Graphs in SAP

S/4HANA: SITUATION HANDLING / INTELLIGENT SITUATION AUTOMATION

Situation Knowledge Graph templates.

- Pending Supplier Confirmation.
- Low number of Quotations Received
- Stock Transport Order Overdue

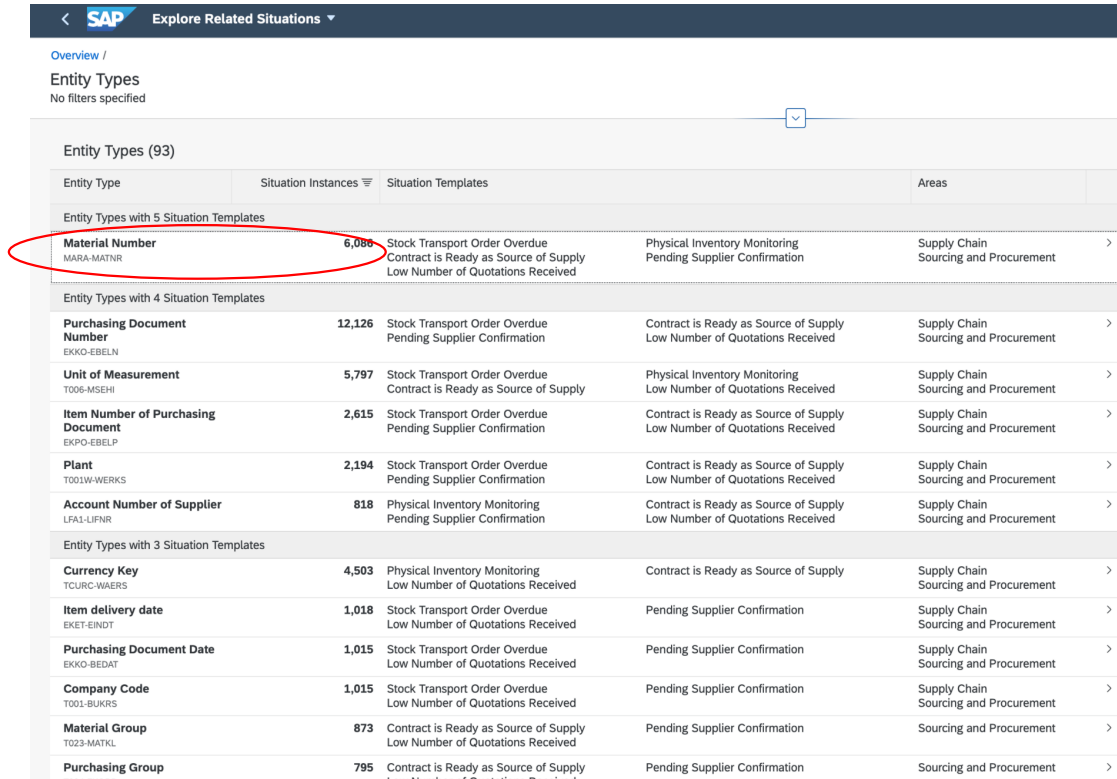
The knowledge graph is built by extracting, analyzing, and transforming a **wide range of metadata**, which is available in the various systems at SAP:



- **S/4HANA** for the virtual data model,
- **Business Situation Type API** for situation master data,
- **FIORI Apps Reference Library** about applications,
- **API Business Hub** about publicly available APIs, CDS views, and Events,
- **Extensibility Cockpit** about business contexts,

Knowledge Graphs in SAP

S/4HANA: EXPLORE RELATED SITUATIONS (NEW APP SINCE 2022)



Entity Types (93)

Entity Type	Situation Instances	Situation Templates	Areas
Entity Types with 5 Situation Templates			
Material Number MARA-MATNR	6,086	Stock Transport Order Overdue Contract is Ready as Source of Supply Low Number of Quotations Received	Physical Inventory Monitoring Pending Supplier Confirmation Supply Chain Sourcing and Procurement
Entity Types with 4 Situation Templates			
Purchasing Document Number EKKO-EBELN	12,126	Stock Transport Order Overdue Pending Supplier Confirmation	Contract is Ready as Source of Supply Low Number of Quotations Received Supply Chain Sourcing and Procurement
Unit of Measurement T006-MSEHI	5,797	Stock Transport Order Overdue Contract is Ready as Source of Supply	Physical Inventory Monitoring Low Number of Quotations Received Supply Chain Sourcing and Procurement
Item Number of Purchasing Document EKPO-EBELP	2,615	Stock Transport Order Overdue Pending Supplier Confirmation	Contract is Ready as Source of Supply Low Number of Quotations Received Supply Chain Sourcing and Procurement
Plant T001W-WERKS	2,194	Stock Transport Order Overdue Pending Supplier Confirmation	Contract is Ready as Source of Supply Low Number of Quotations Received Supply Chain Sourcing and Procurement
Account Number of Supplier LFA1-LIFNR	818	Physical Inventory Monitoring Pending Supplier Confirmation	Contract is Ready as Source of Supply Low Number of Quotations Received Supply Chain Sourcing and Procurement
Entity Types with 3 Situation Templates			
Currency Key TCURC-WAERS	4,503	Physical Inventory Monitoring Low Number of Quotations Received	Contract is Ready as Source of Supply Supply Chain Sourcing and Procurement
Item delivery date EKET-EINDT	1,018	Stock Transport Order Overdue Low Number of Quotations Received	Pending Supplier Confirmation Supply Chain Sourcing and Procurement
Purchasing Document Date EKKO-BEDAT	1,015	Stock Transport Order Overdue Low Number of Quotations Received	Pending Supplier Confirmation Supply Chain Sourcing and Procurement
Company Code T001-BUKRS	1,015	Stock Transport Order Overdue Low Number of Quotations Received	Pending Supplier Confirmation Supply Chain Sourcing and Procurement
Material Group T023-MATKL	873	Contract is Ready as Source of Supply Low Number of Quotations Received	Pending Supplier Confirmation Sourcing and Procurement
Purchasing Group T034-EVZDD	795	Contract is Ready as Source of Supply Low Number of Quotations Received	Pending Supplier Confirmation Sourcing and Procurement

Explore Related Situations app using the Situation Knowledge Graph

New Analytical application, to provide insights that go beyond the scope of a single identified problem.

Explore the relationships between situations, business entities, and other solution-related information.

In this application, we follow an interaction pattern that is driven by the business entities themselves.

The entry page shows you which business entity types have the most issues related to the different situation templates and business areas.

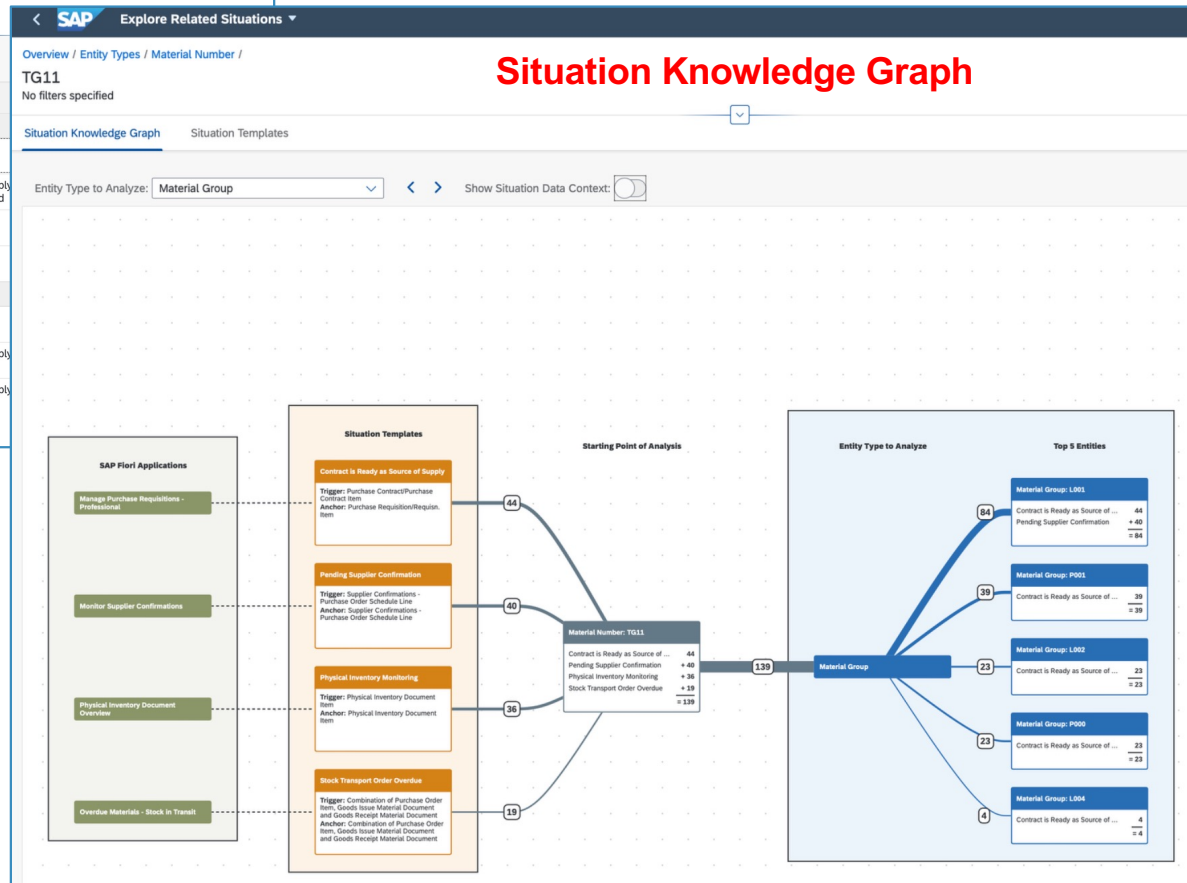
In the example, it is the material, indicated by the "Material Number" entity type, which has a high number of situation instances from five different situation templates

Knowledge Graphs in SAP

S/4HANA: EXPLORE RELATED SITUATIONS APP

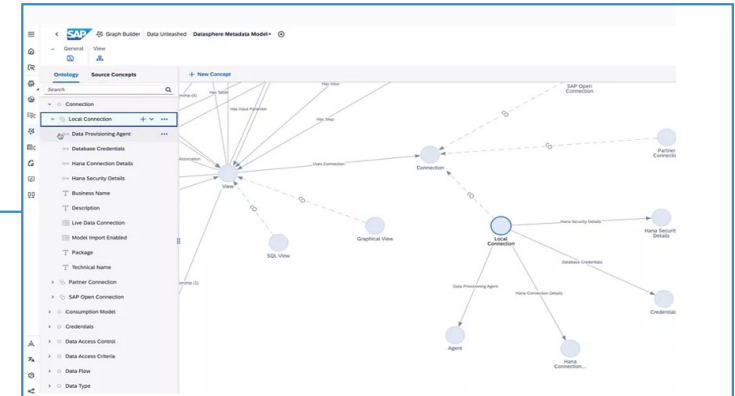
Overview / Entity Types / Material Number
No filters specified

Entity	Situation Instances	Situation Templates
Entities (2,426)		
Entities with 4 Situation Templates		
TG11	139	Stock Transport Order Overdue Contract is Ready as Source of Supply Physical Inventory Monitoring Pending Supplier Confirmation
TG0011	114	Physical Inventory Monitoring Pending Supplier Confirmation Contract is Ready as Source of Supply Low Number of Quotations Received
TG12	89	Stock Transport Order Overdue Contract is Ready as Source of Supply Physical Inventory Monitoring Pending Supplier Confirmation
TG21	65	Stock Transport Order Overdue Contract is Ready as Source of Supply Physical Inventory Monitoring Pending Supplier Confirmation
Entities with 3 Situation Templates		
FLOG-SP14-NO-QM	443	Stock Transport Order Overdue Pending Supplier Confirmation Physical Inventory Monitoring
RM12	105	Physical Inventory Monitoring Pending Supplier Confirmation Contract is Ready as Source of Supply
TG-D020	80	Physical Inventory Monitoring Pending Supplier Confirmation Contract is Ready as Source of Supply



Knowledge Graphs on linked business data

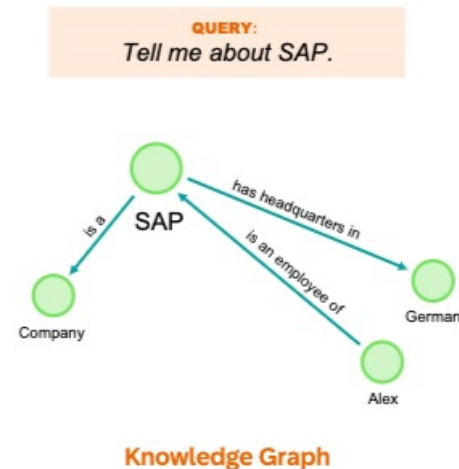
SAP DATASPHERE – H1 2025



Knowledge Graphs

How they can help in leveraging your data and the motif of embedding into a Data Fabric

- **Improved LLM Query Performance**
Knowledge graphs enable LLMs to efficiently recognize and navigate relationships from structured data, reducing query costs
- **Compatible with Ad Hoc Queries**
Well-suited for the types of un-optimized, ad hoc queries often required by messy and arbitrary questions asked in natural language
- **Powerful Data Querying**
Flexible, schema-less structure works well to describe and connect sparse, widely interconnected datasets

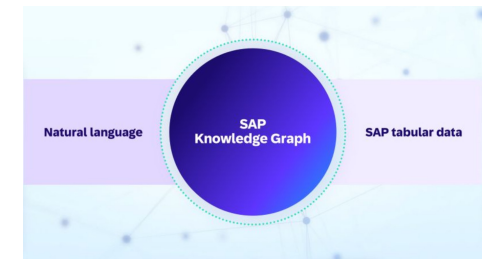


- **SAP Datasphere knowledge graph** automatically creates an **ontology** representing the relationships in the data, including the inherent business context from SAP application sources like SAP S/4HANA
- Extended and augmented via an **ontology editor**
- Finally, the data in SAP Datasphere is automatically applied to this ontology to **create a knowledge graph**

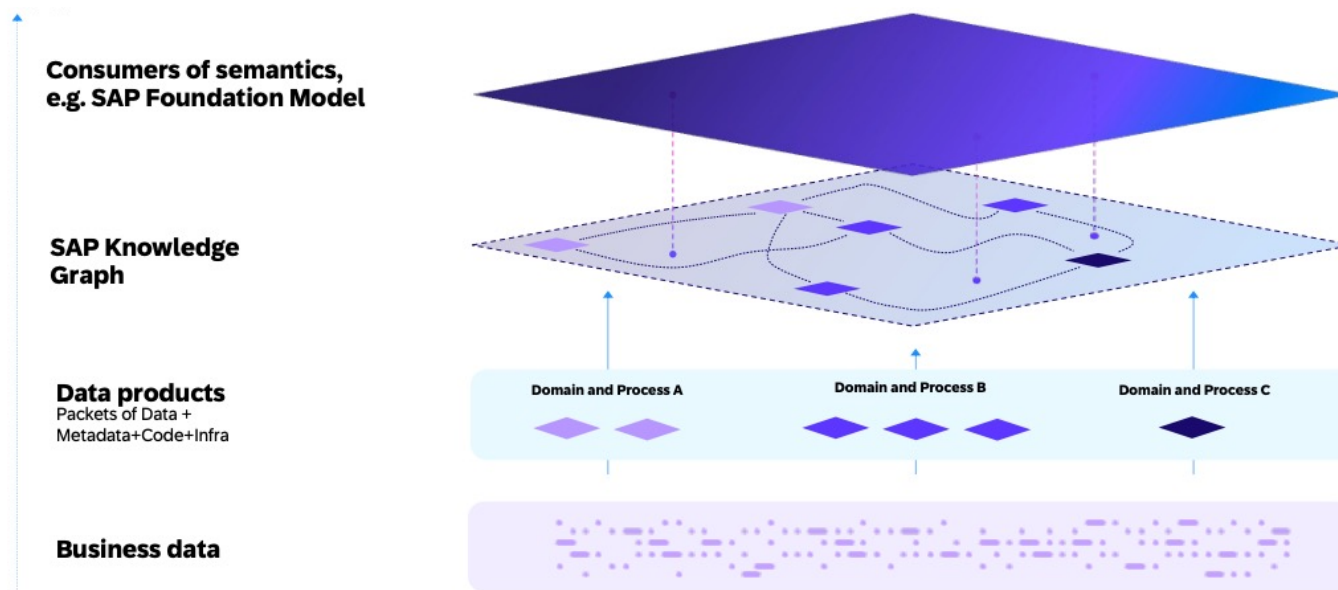
SAP Knowledge Graphs

SEMANTIC LAYER

SAP Knowledge Graph solution captures meaning and context from **Lower-level data layers** to bring precise semantics to **SAP Foundation Model** and LLM applications.

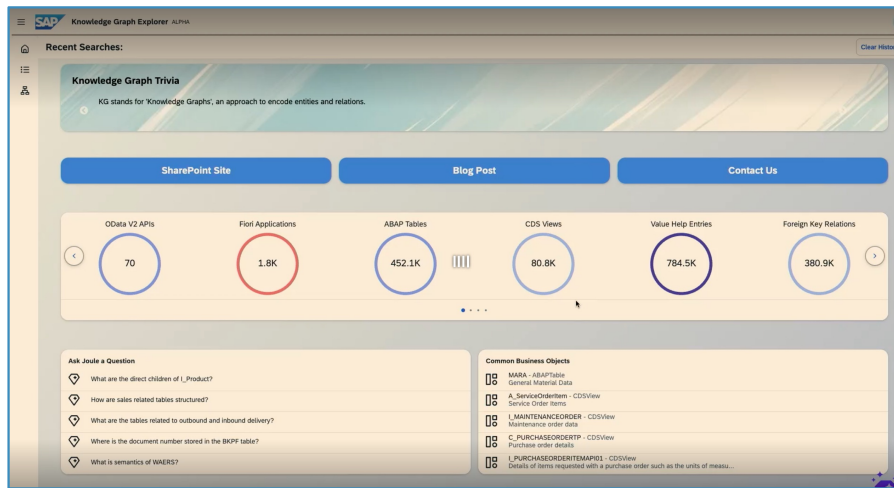


- 452.000 ABAP Tables
- 80.000 CDS Views
- 7,3 Millions of fields



Knowlegde Graphs on linked business data

SAP BTP: KNOWLEDGE GRAPH EXPLORER NAVIGATOR



- 452.000 ABAP Tables
- 80.000 CDS Views
- 7,3 Millions of fields

All Entities (6)

Type	Name	Description	Semantic Keys
CDSView	I_InboundDelivery	Inbound Delivery	InboundDelivery
CDSView	I_InboundDeliveryItem	Inbound Delivery Item	InboundDelivery, InboundDeliveryItem
ABAPTable	LIKP	SD Document: Delivery Header Data	CustomerReturnsDelivery, CustomerReturnsDeliveryItem, InboundDelivery, InboundDeliveryItem, OutboundDelivery, OutboundDeliveryItem
API	LE_SHP_INBOUND_DELL...		GoodsMovementType, INBOUNDDELIVERY, INBOUNDDELIVERYITEM, InboundDelivery, InventorySpecialStockType, InventoryUsabilityCode, LoadingGroup, PURCHASEORDER, PURCHASEORDERITEM, SDDOCUMENT, SDDOCUMENTITEM, SalesDocumentItemCategory, ShipmentItem, StorageLocation, Supplier
EntitySet	C_InboundDeliveryItemO...		INBOUNDDELIVERY, INBOUNDDELIVERYITEM
EntitySet	C_InboundDeliveryObjPg		INBOUNDDELIVERY

Showing Results: (1-6 of 6)

Entry Details

Data Source Details: LE_SHP_INBOUND_DELIVERY_OBJ_PG_SRV

Type: API
API URI: http://schema.sap.com/sule/Service/LE_SHP_INBOUND_DELIVERY_OBJ_PG_SRV

Description: GoodsMovementType, INBOUNDDELIVERY, INBOUNDDELIVERYITEM, InboundDelivery, InventorySpecialStockType, InventoryUsabilityCode, LoadingGroup, PURCHASEORDER, Part, SDDOCUMENT, SDDOCUMENTITEM, SalesDocumentItemCategory, ShipmentItem, StorageLocation, Supplier

Semantic Keys: GoodsMovementType, INBOUNDDELIVERY, INBOUNDDELIVERYITEM, InboundDelivery, InventorySpecialStockType, InventoryUsabilityCode, LoadingGroup, PURCHASEORDER, Part, SDDOCUMENT, SDDOCUMENTITEM, SalesDocumentItemCategory, ShipmentItem, StorageLocation, Supplier

Entity Table

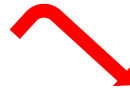
Entity Set Name	CDS View Name	CDS Description
C_BPFNodeDetails		
C_BPFNodeDetailsSet		
C_InboundDeliveryItemObjPg	C_InboundDeliveryItemObjPg	Inbound Delivery Item Object Page
C_InboundDeliveryObjPg	C_InboundDeliveryObjPg	Inbound Delivery Object Page
C_RelPurOrdSikTransOrdDir	C_RelPurOrdSikTransOrdDir	
C_SDDocumentPartnerCard	C_SDDocumentPartnerCard	Business Partner for Fact Sheets
C_SerialNmbEquipNmbDelivery	C_SerialNmbEquipNmbDelivery	Serial and Equipment Numbers
I_AdvShipReceiverRlvnce	I_AdvShipReceiverRlvnce	Advanced Shipping and Receiving Relevence
I_Customer_VH	I_Customer_VH	
I_FieldLogisticIndicator	I_FieldLogisticIndicator	

Knowlegde Graphs on linked business data

SAP BTP- KG EXPLORER NAVIGATOR: LOOKING FOR INBOUND DELIVERY

The screenshot shows the SAP Knowledge Graph Explorer interface. The top navigation bar includes 'Entity Browser', 'API LE_SHP_INBOUND_DELIVERY_OBJPG_SRV', and 'CDSView C_InboundDeliveryItemObjPg'. The main content area is titled 'Data Source Details: C_InboundDeliveryItemObjPg'. It features a 'Type' field with the value 'CDSView' and a 'Document URL' field. Below this, there are several key-value pairs: 'Description: Inbound Delivery Item Object Page', 'Business Document URL', 'Line of Business', 'Detailed Description', and 'Application Area: Semantic Keys'. At the bottom, there is a 'Fields' section with a table listing various fields.

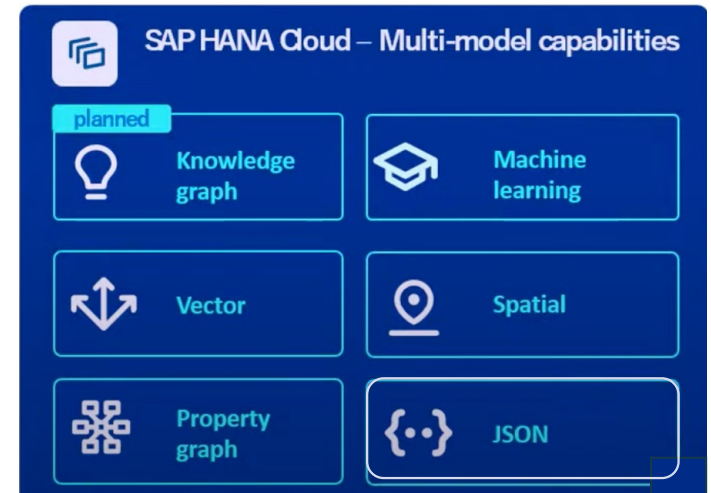
DPP	Field Name
	ActualDeliveredQtyInBaseUnit
	ActualDeliveryQuantity
	BaseUnit
	Batch
	ConversionFactor
	DeliveryDocumentItem
	DeliveryDocumentItemCategory



The screenshot shows the SAP Knowledge Graph Explorer interface. The top navigation bar includes 'Entity Browser', 'API LE_SHP_INBOUND_DELIVERY_OBJPG_SRV', 'CDSView C_InboundDeliveryItemObjPg', and 'Field BaseUnit'. The main content area is titled 'Data Source Details: BaseUnit'. It features a 'Type' field with the value 'CDSField' and a 'Description' field. Below this, there are several key-value pairs: 'Direct Data Source: Entity: LIPS, Field: MEINS', 'Base Data Source: Entity: I_InboundDeliveryItem, Field: BaseUnit', 'Master Data Source: Entity: T006, Field: MSEHI', and 'Direct Parent: C_InboundDeliveryItemObjPg'. At the bottom, there are expandable sections for 'Views Referencing This Column' and 'Foreign Key Relations'.

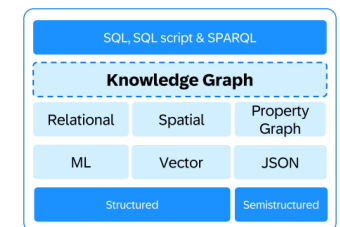
Knowledge Graphs: Connecting the Facts

SAP HANA CLOUD'S KG ENGINE FOR BUSINESS CONTEXT



With the upcoming **knowledge graph engine**, SAP HANA Cloud gains the **ability to manage and query semantically connected relationships** – all within the same database as other relational and multi-model engines.

- **Native RDF Triple Store:** One of the key features of the SAP HANA Cloud knowledge graph engine is its native support for RDF triple store
- **SQL and SPARQL Interoperability:** The knowledge graph engine will support SPARQL, the specialized query language for knowledge graph data, while tightly integrating it with SQL. This is achieved with the introduction of new SQL.



SAP HANA Cloud **knowledge graph engine** vs the SAP HANA Cloud **graph engine** designed to handle **property graphs**



Thank you!

Q & A



Antonio Leites
Sulzer España

antonio.leites@sulzer.es



Backup Slides



Introduction to Knowledge Graphs



Building a Knowledge Graph



SAP and Knowledge Graphs

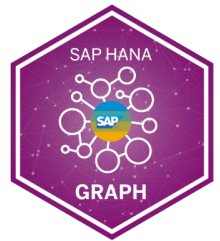
Knowlegde Graphs vs Property Graphs

SAP HANA CLOUD ENGINES

SAP HANA Cloud **knowledge graph engine** vs the SAP HANA Cloud **graph engine** designed to handle **property graphs**

Property Graphs: These graphs are ideal for **analysing relationships** between entities.

- In a property graph, **data** is represented as **nodes** (entities) and **edges** (relationships), where each edge can hold different properties that describe the connection between nodes.

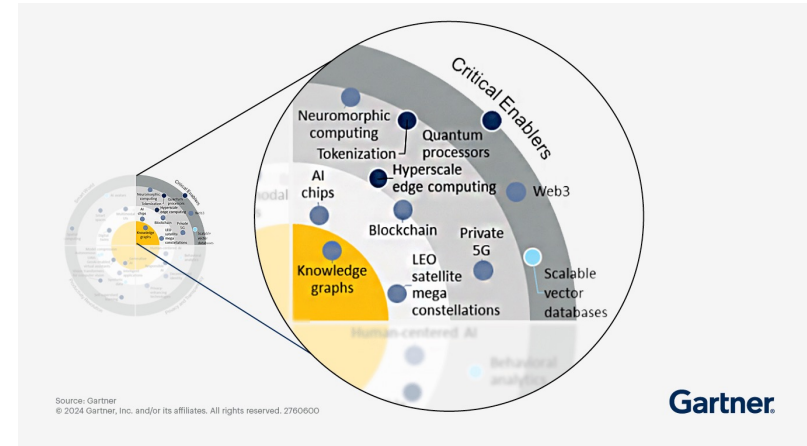
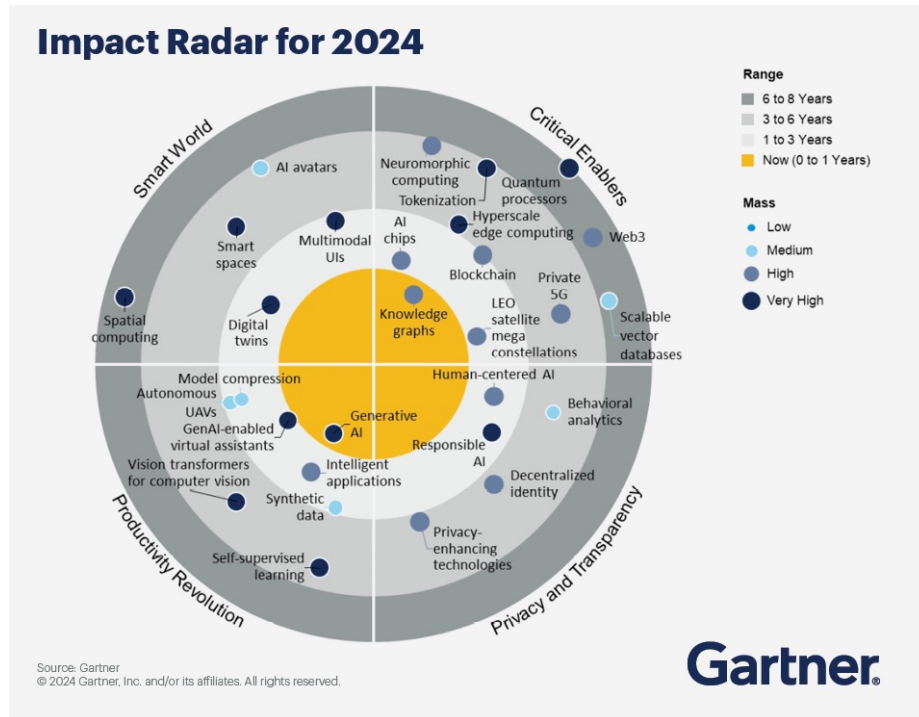


Knowledge Graphs: focus on **establishing facts** and the **logical relationships** between those facts.

- They use a **subject-predicate-object** structure (triples), where each fact is stored as a relationship.

Knowlegde Graphs: Emerging Technologies and Trends

THE GARTNER EMERGING TECH IMPACT RADAR



Critical Enablers This theme focus on expectations for **emerging applications** — some of which will enable new use cases and others that will enhance existing experiences .

Knowledge Graphs (KGs) are **machine-readable data structures** that describe the relationship between heterogeneous data via a network of nodes and links.

GenAI actions for tech product leaders:

...

Prioritize the most prevalent use cases, such as **enterprise search/knowledge mining** and virtual agents, as these evidently already deliver real value to users.

...