

KARMA

Pedro Szekely and Craig A. Knoblock

pszekely@isi.edu, knoblock@isi.edu

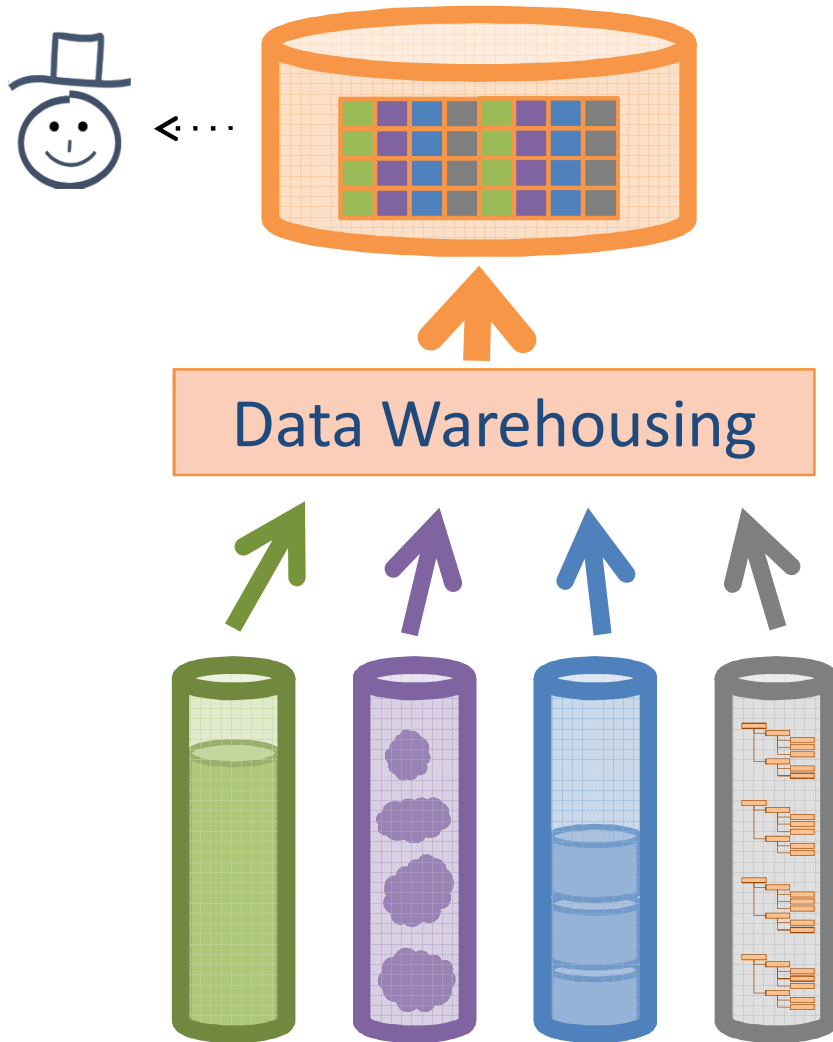
University of Southern California, Information Sciences Institute

Work in collaboration with Mohsen Taheriyani, Jason Slepicka, Bo Wu, Dipsy Kapoor, Jose Luis Ambite, Yao-Yi Chiang, Aman Goel, Shubham Gupta, Maria Muslea, Kristina Lerman and many students.

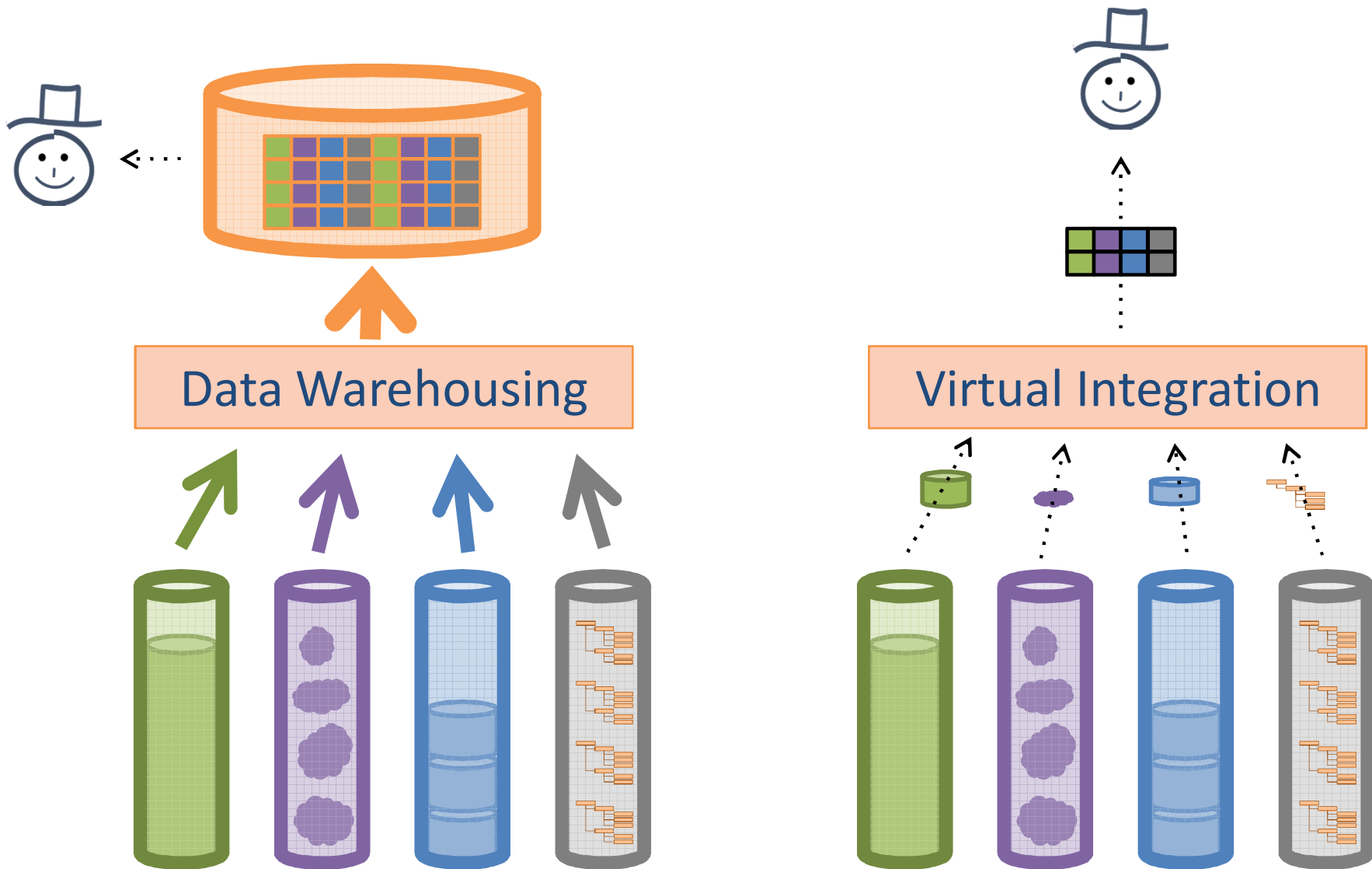
Outline

- Integrating data silos
- Our Karma tool
- Use cases

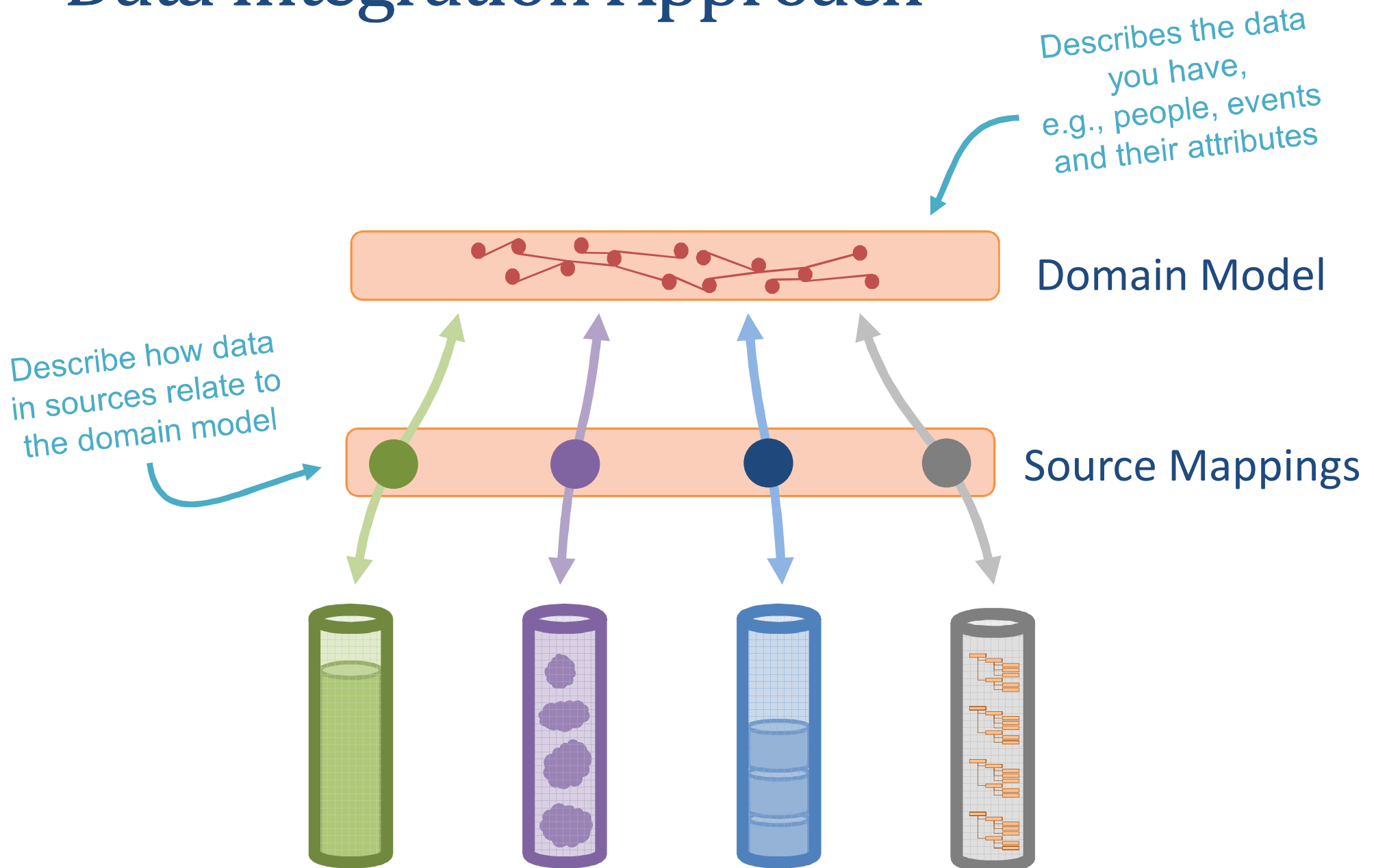
Data Integration Approaches



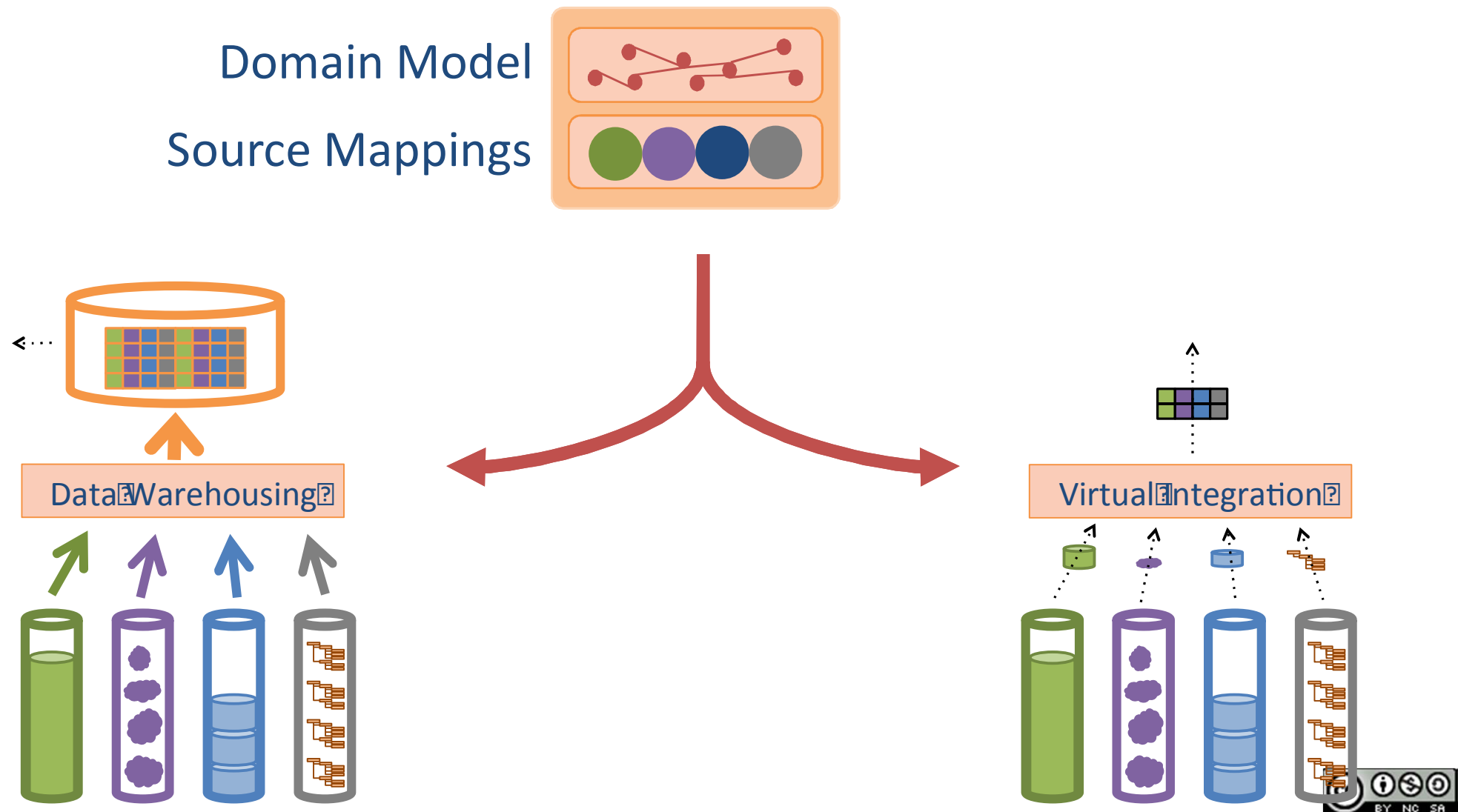
Data Integration Approaches



Data Integration Approach



Information Integration Using Source Mappings

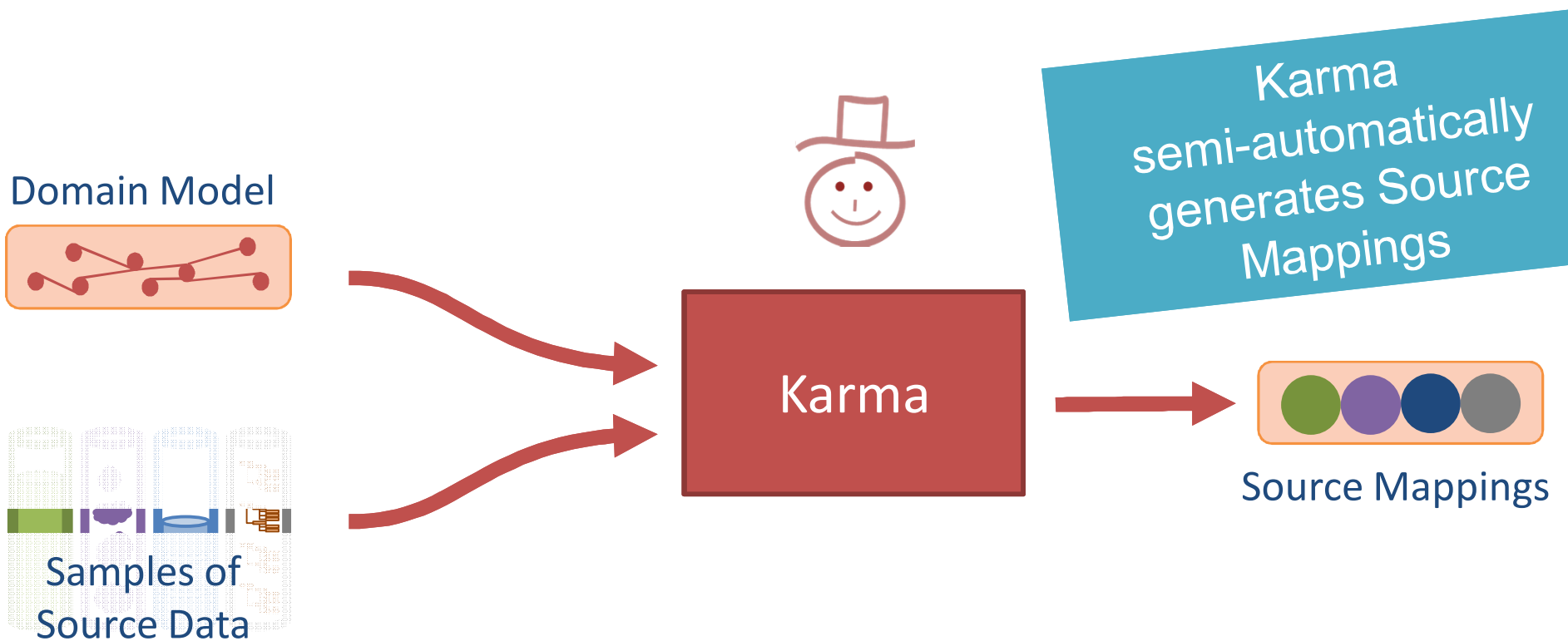


Karma:

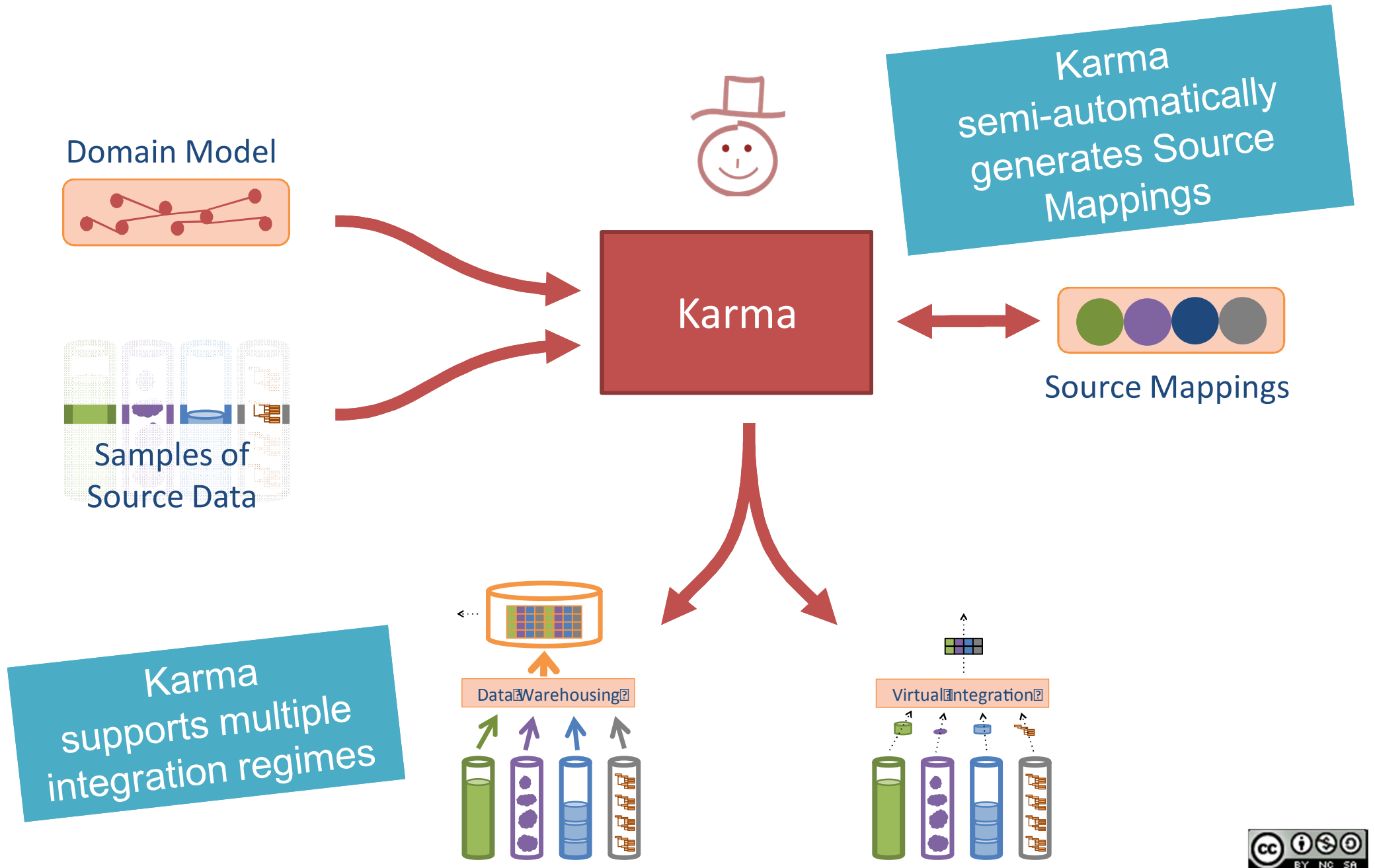
Our Information Integration Toolkit



Information Integration Using Karma



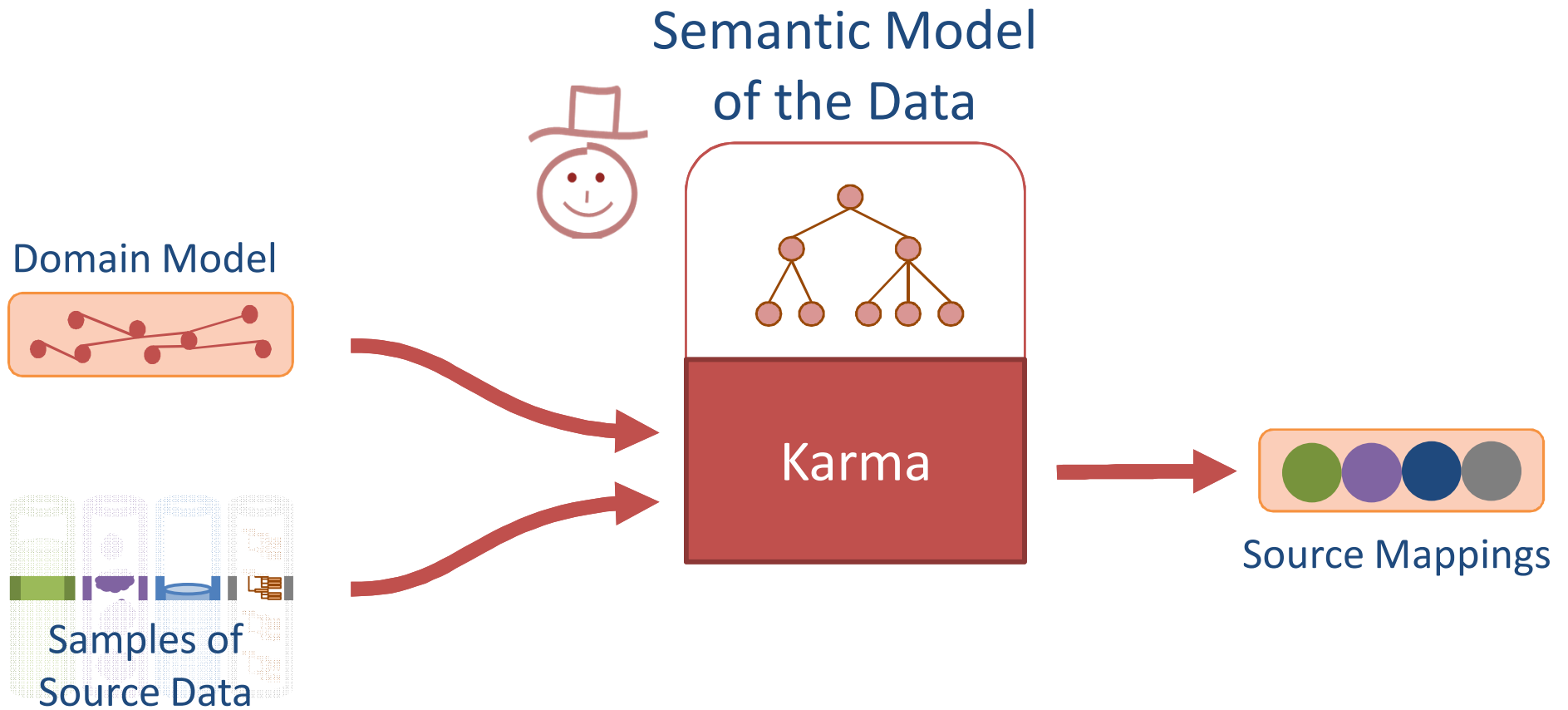
Information Integration Using Karma



Karma's Secret Sauce



Karma Understands Your Data



Karma semi-automatically builds a semantic model of your data

Semantic Types: Meaning of Data in Columns

Semantic vs Syntactic Types

String	String	String	Date	String	String
Perpetrator	Nationality	Location	Time	Type	Description
Values					
Zian Akhtar Mehmood Afzal	Riot	Seen in demonstration o
Abdul Nomaz Farooq Nomaz	Somalia	Nairobi	12/24/2011 17:00Z	IED	...
...	...	Nairobi
Khair Shahed Riaz Afredi Zoha Afredi	12/24/2011 13:00Z	IED	...

Not useful for information integration



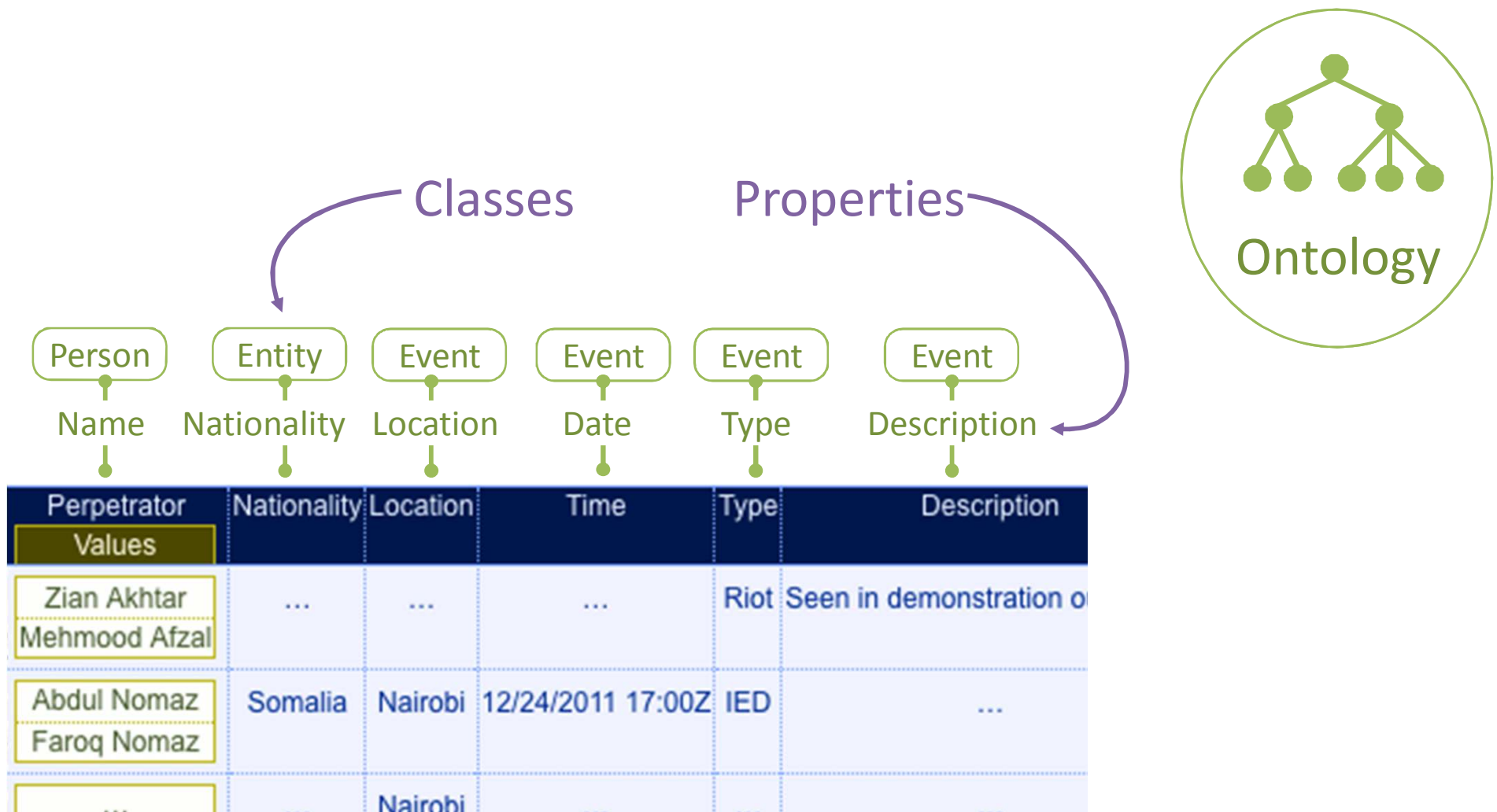
Semantic Types Capture the Meaning of Data in Columns

Semantic vs Syntactic Types

Name of Person	Nationality of Entity	Location of Event	Date of Event	Type of Event	Description of Event
Perpetrator	Nationality	Location	Time	Type	Description
Zian Akhtar Mehmood Afzal	Riot	Seen in demonstration of
Abdul Nomaz Faroq Nomaz	Somalia	Nairobi	12/24/2011 17:00Z	IED	...
...	...	Nairobi
Khair Shahed Riaz Afredi Zoha Afredi	12/24/2011 13:00Z	IED	...



Semantic Types Defined Using an Ontology



Karma Learns the Semantic Types

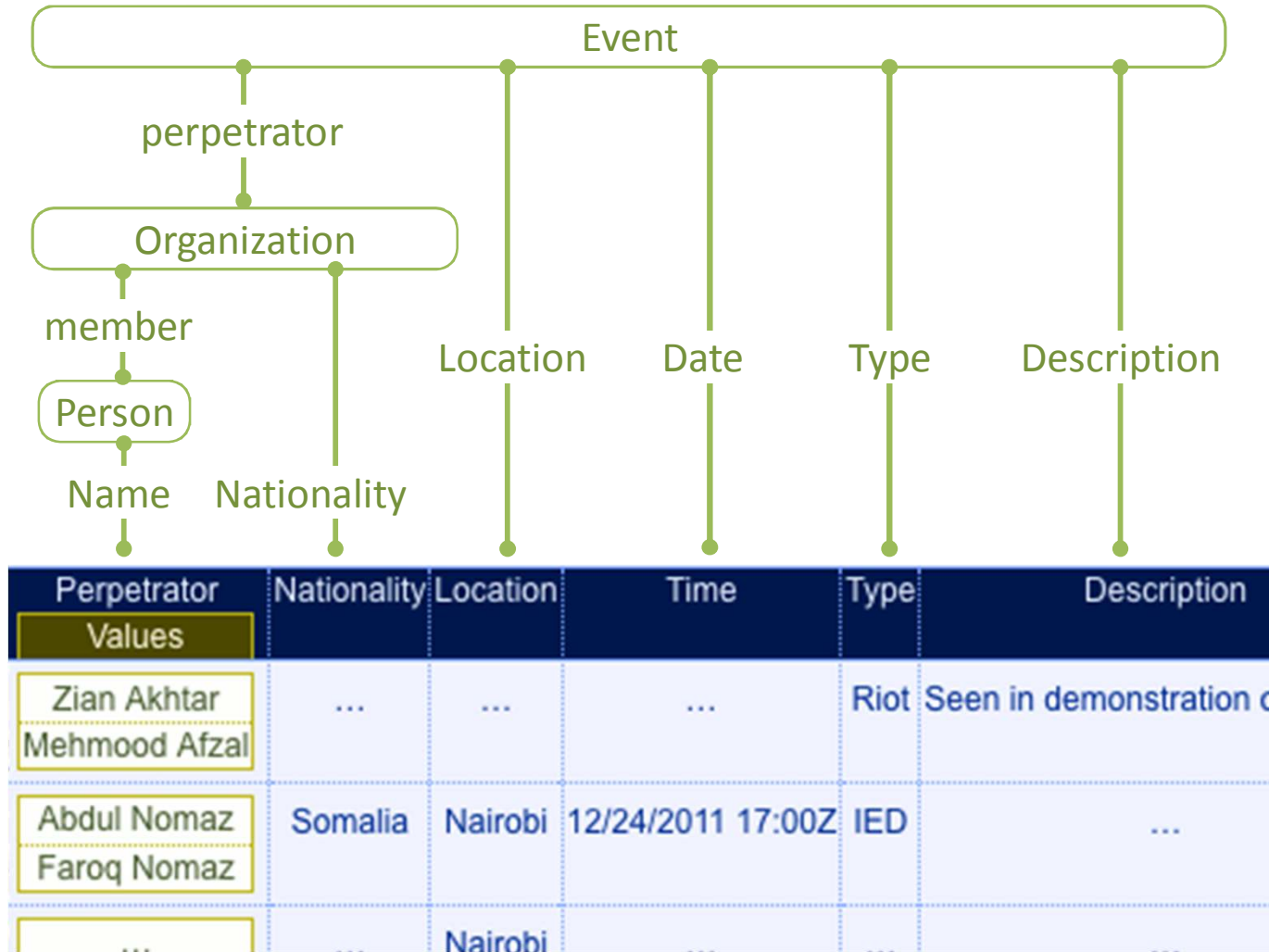
1. User specifies them once
2. Karma learns features to recognize them
3. Next time Karma sees similar data it automatically proposes semantic types

Person	Entity	Event	Event	Event	Event
Name	Nationality	Location	Date	Type	Description
Perpetrator	Nationality	Location	Time	Type	Description
Values					
Zian Akhtar Mehmood Afzal	Riot	Seen in demonstration o
Abdul Nomaz Faroq Nomaz	Somalia	Nairobi	12/24/2011 17:00Z	IED	...
...	...	Nairobi

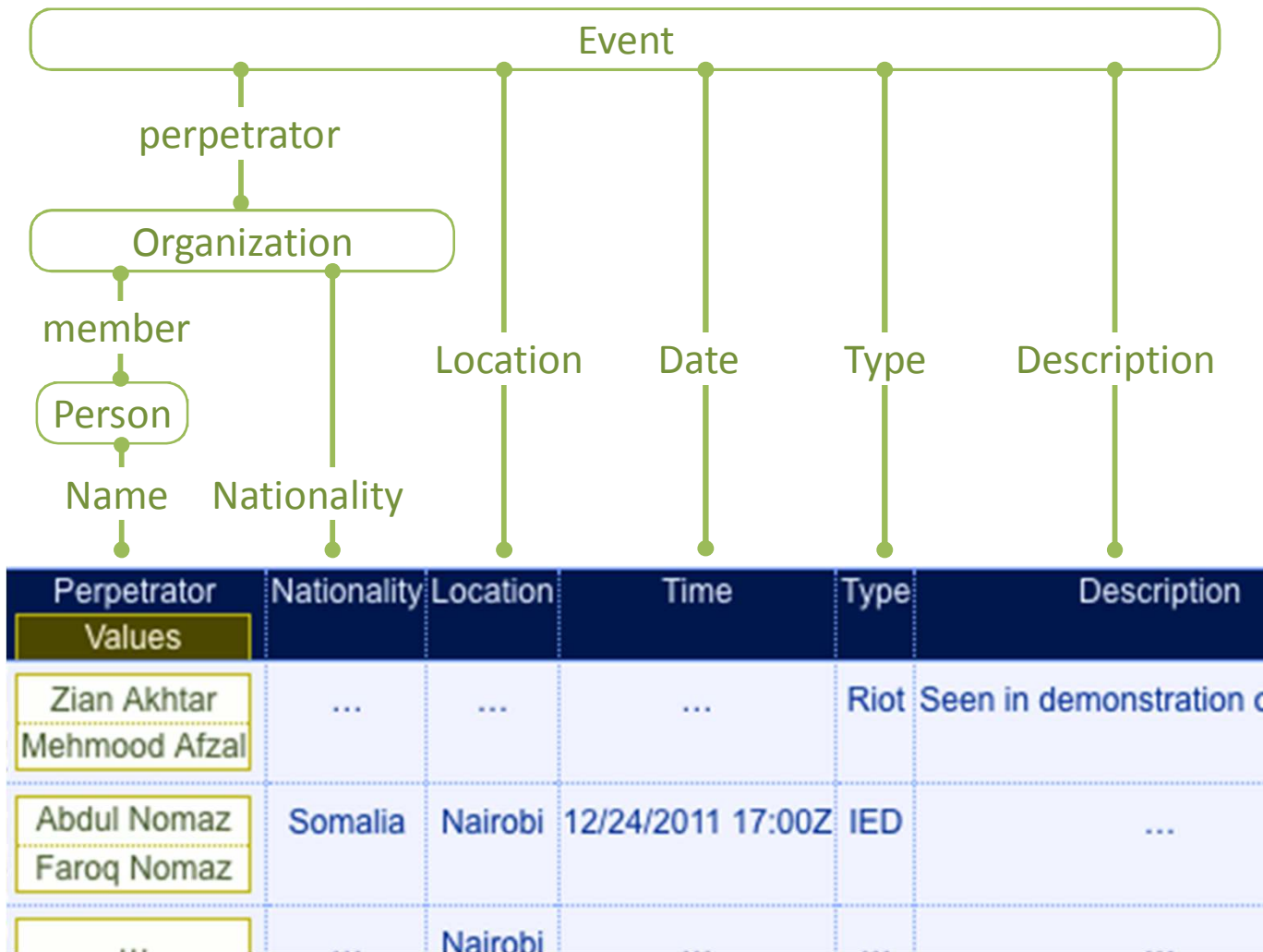
Relationships Among Columns



Relationships Among Specified in Terms of Classes and Properties

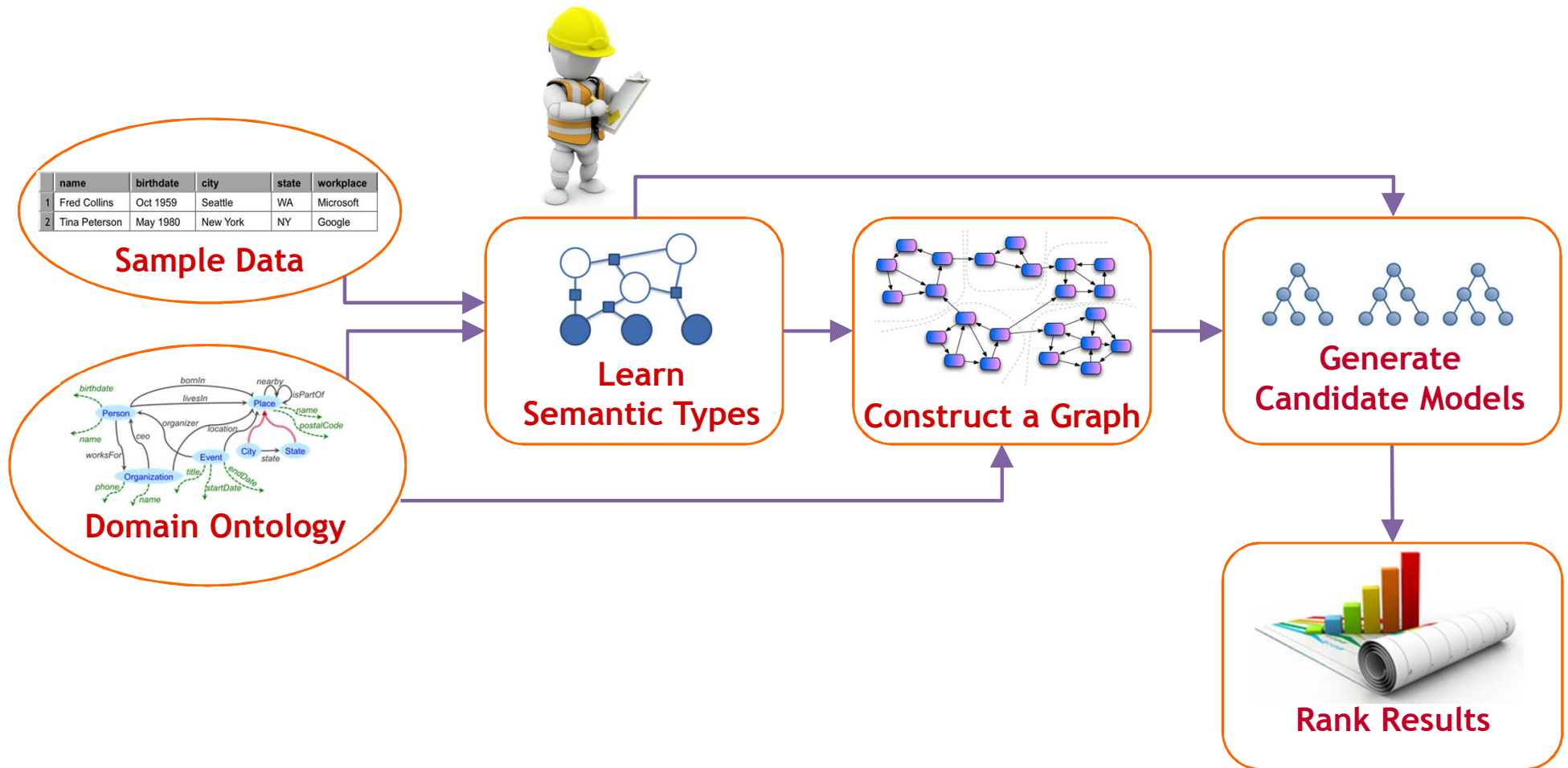


Karma Automatically Infers Relationships



1. Karma automatically finds relationships using the ontology
2. When proposed relationships are incorrect, the user adjusts them

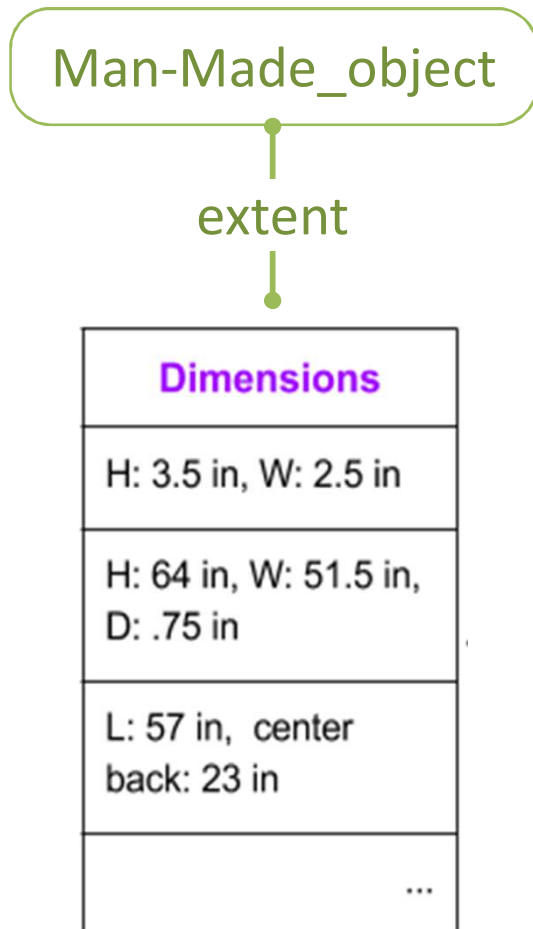
Approach



Learning Semantic Types



Learning Semantic Types



1. User specifies
2. System learns

Learning Semantic Types

Man-Made_object

extent

Dimensions
H: 3.5 in, W: 2.5 in
H: 64 in, W: 51.5 in, D: .75 in
L: 57 in, center back: 23 in
...

Extent
52.1 x 71.4 cm (20 1/2 x 28 1/8 in.)
9 3/4 x 7 9/16 in.
H: 19 x W: 15 1/4 x D: 8 1/4 in.
...

System Suggests Semantic Types

Man-Made_object

extent

Dimensions
H: 3.5 in, W: 2.5 in
H: 64 in, W: 51.5 in, D: .75 in
L: 57 in, center back: 23 in
...

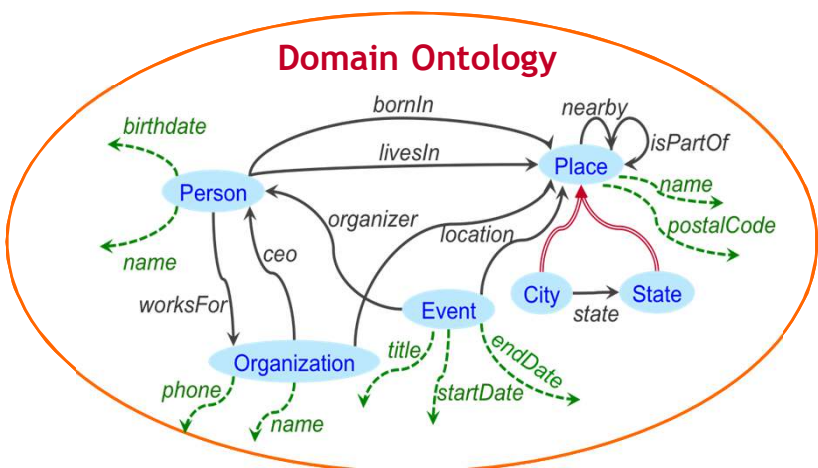
Man-Made_object

extent

Extent
52.1 x 71.4 cm (20 1/2 x 28 1/8 in.)
9 3/4 x 7 9/16 in.
H: 19 x W: 15 1/4 x D: 8 1/4 in.
...

Learning Semantic Types

- Requirements:
 - Learn from a small number of examples
 - Distinguish both string and numeric values
 - Can be learned quickly and is highly scalable to large numbers of semantic types



	Person	Person	City	State	Organization
	name	birthdate	name	name	name
	name	date	city	state	workplace
1	Fred Collins	Oct 1959	Seattle	WA	Microsoft
2	Tina Peterson	May 1980	New York	NY	Google

Approach for Textual Data

Dimensions
H: 3.5 in, W: 2.5 in
H: 64 in, W: 51.5 in, D: .75 in
L: 57 in, center back: 23 in
...

Each semantic label has a **characteristic** set of **tokens**

Each column of data is a **document**

Use **information retrieval** techniques to **compare** documents

Labeled data is indexed using **Apache Lucene**

Compare documents using **TF-IDF cosine** similarity

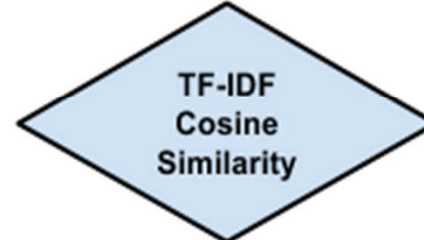
Semantic Types for Text Data

Dimensions
H: 3.5 in, W: 2.5 in
H: 64 in, W: 51.5 in, D: .75 in
L: 57 in, center back: 23 in
...

Term: TF-IDF score
h: 0.375
w: 0.336
in: 0.491
centre: 0.241
back: 0.301
.....

Extent
52.1 x 71.4 cm (20 1/2 x 28 1/8 in.)
9 3/4 x 7 9/16 in.
H: 19 x W: 15 1/4 x D: 8 1/4 in.
...

Term: TF-IDF score
h: 0.414
w: 0.364
d: 0.245
cm: 0.354
in: 0.395
.....



$$tf(t, d) = frequency^{1/2}$$

$$idf(t) = 1 + \log\left(\frac{numDocs}{docFreq + 1}\right)$$

$$sim(q, d) = \frac{V(q) \cdot V(d)}{|V(q)| \cdot |V(d)|}$$

Approach for Numeric Data

Total Population	Number of people
107875	11070
47823	41542
60704	33039
81034	780058
.....

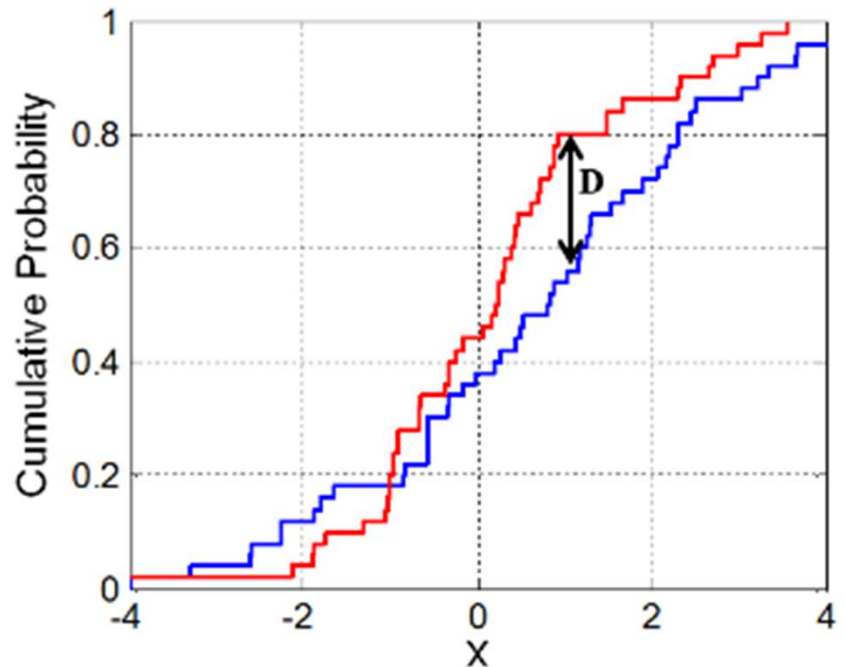
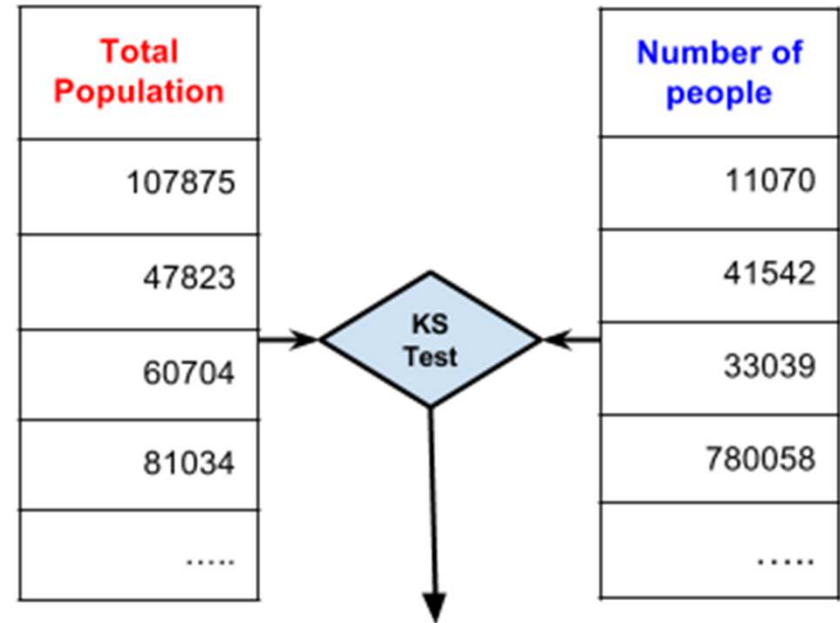
Distribution of values in different semantic type is different

E.g., distribution of **population** is different from distribution of **temperatures**

Use **Statistical Hypothesis testing** to see which distribution fits best

Approaches: Welch's T-test, Mann-Whitney U-test and **Kolmogorov-Smirnov Test**

Approach for Numeric Data



$$D_{N_1, N_2} = \sup_x |F_{1, N_1}(x) - F_{2, N_2}(x)|$$

Combined Approach

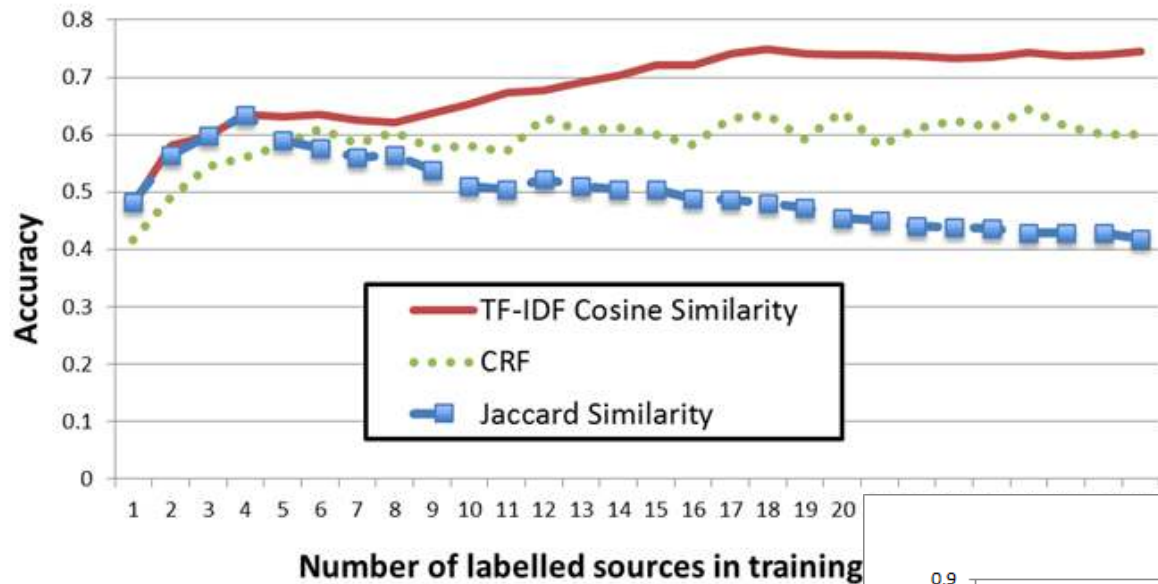
- Combined Approach:
 - Training
 - Add new example data as training for either textual or numeric types
 - If ambiguous, train as both textual and numeric
 - Testing
 - If textual, apply tf/idf
 - If numeric apply KS-test
 - If ambiguous and at least 70% numeric apply KS-test
 - otherwise tf/idf
- Top-k suggestions returned based on the confidence scores

Evaluation of Semantic Typing

- ❖ **Museum Dataset** – 29 data sources from different art museums in the US.
Ontologies: EDM, AAC, FOAF, SKOS, Dublin Core Metadata Terms, ORE, ElementsGr2
- ❖ **City Dataset** – 10 data sources about various cities in the world - manually extracted from DBpedia.
Ontology: DBpedia Ontology

Evaluation

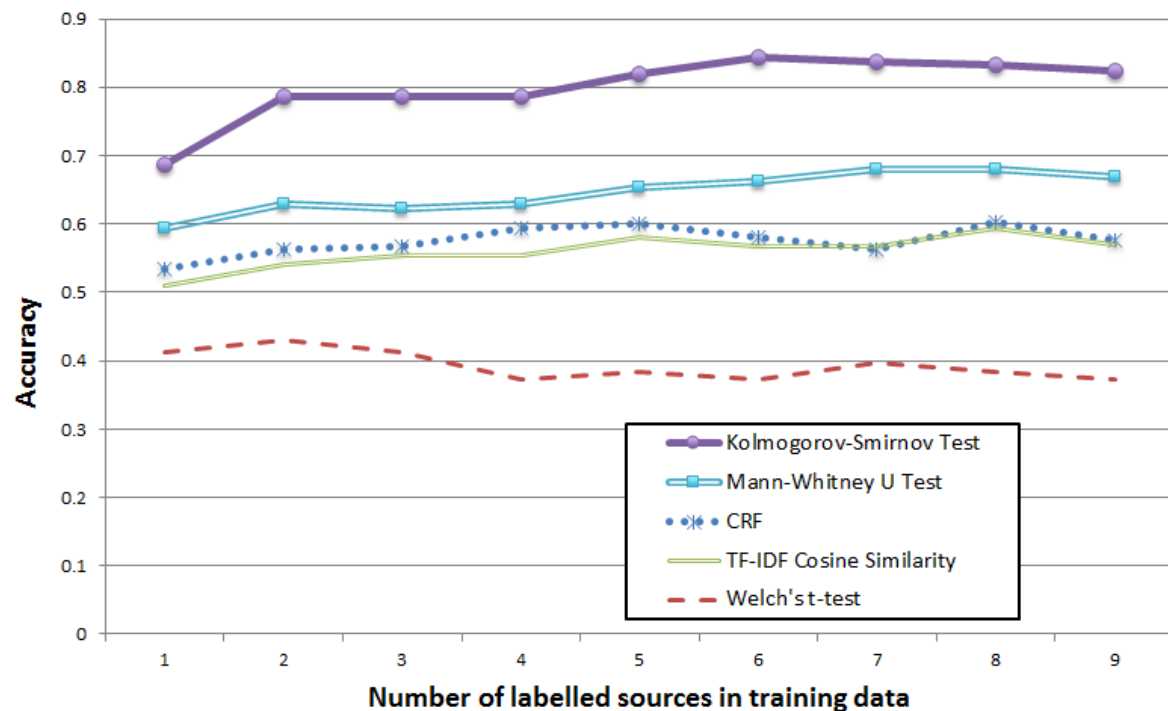
Average Top-1 Accuracy (Textual Data)



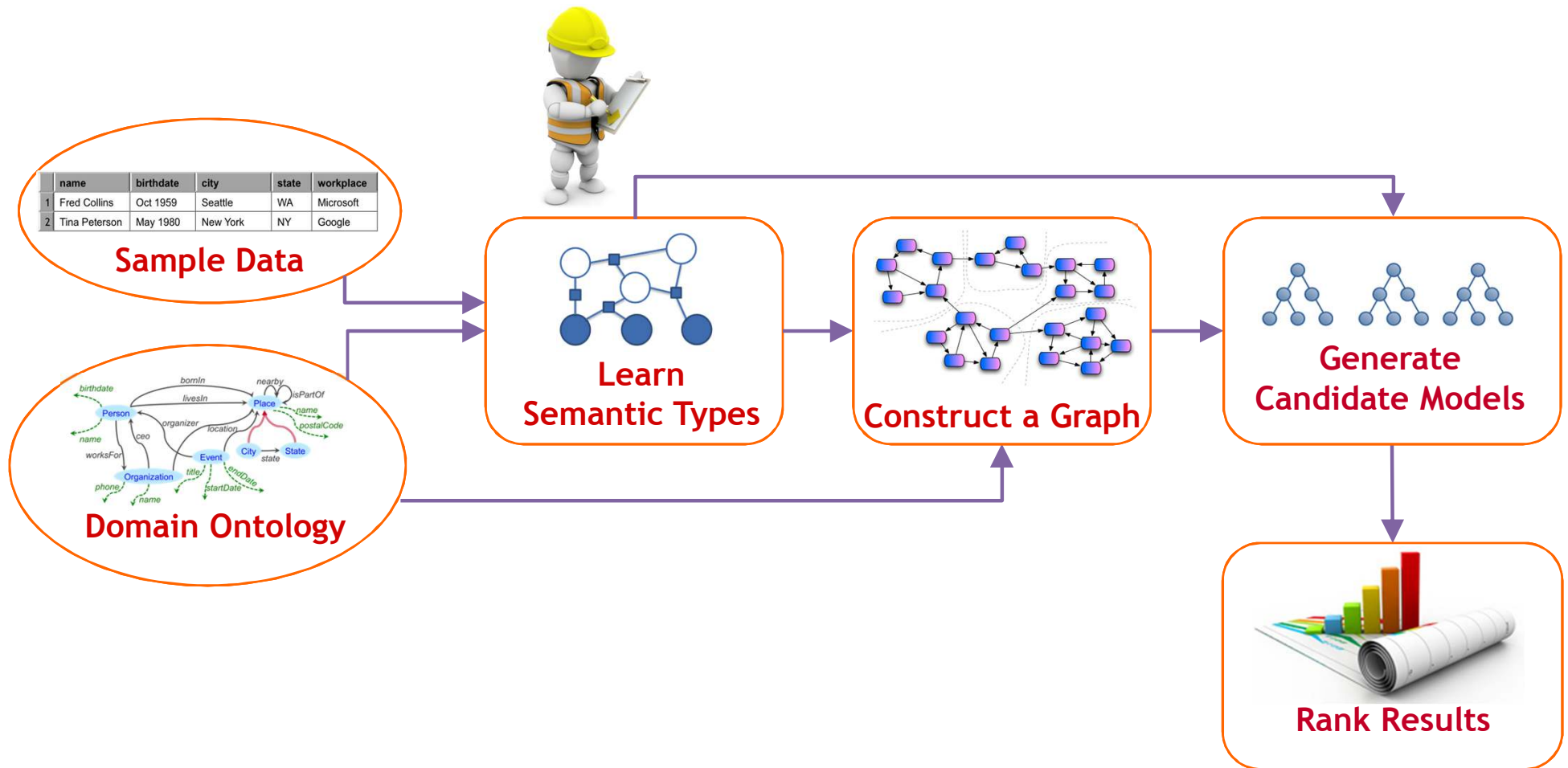
Combined approach achieves 97% accuracy on the top-4 accuracy

Reduced the training time from 110s to 0.45s

Average Top-1 Accuracy (Numeric Data)

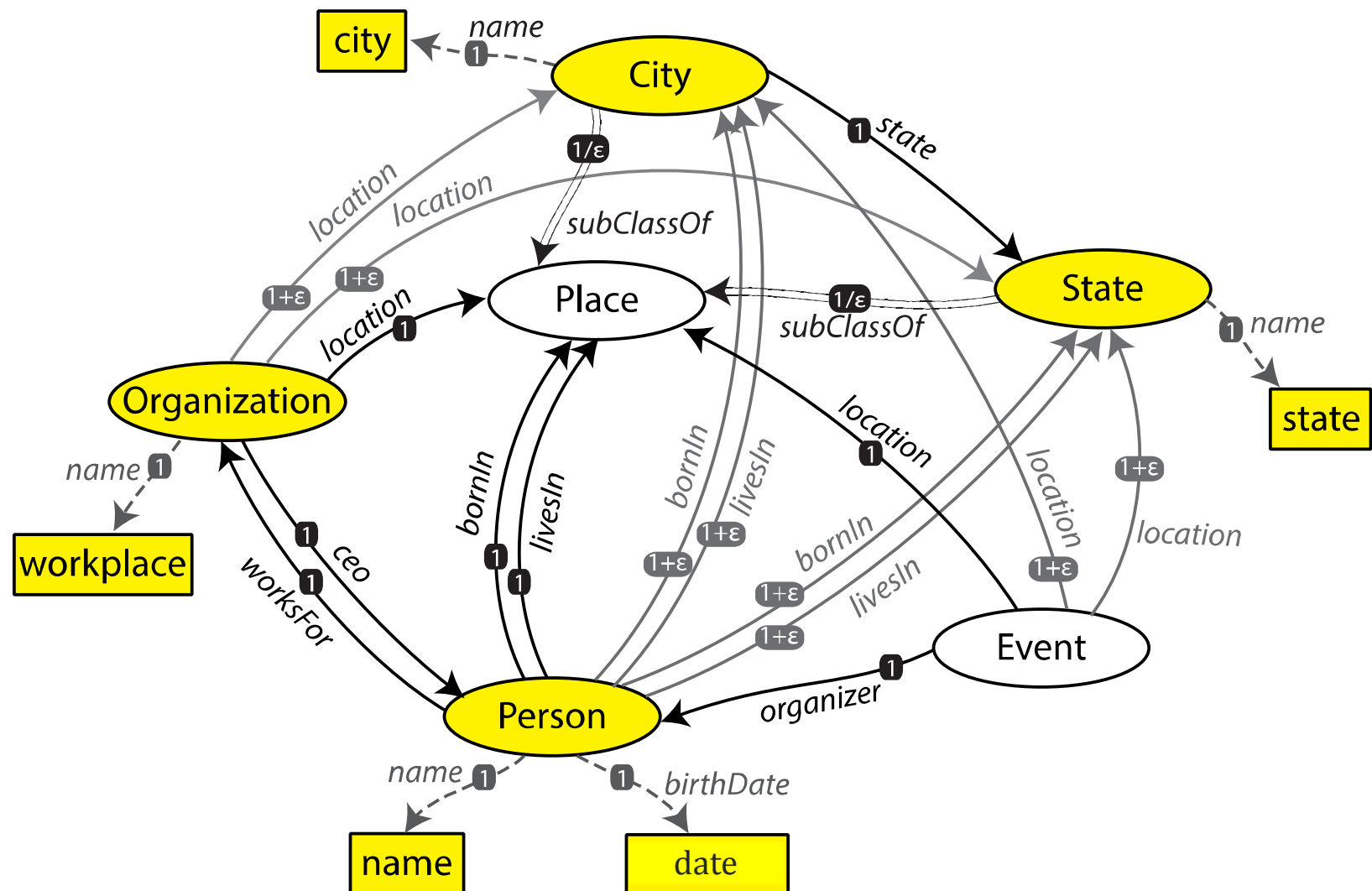


Approach



Construct a Graph

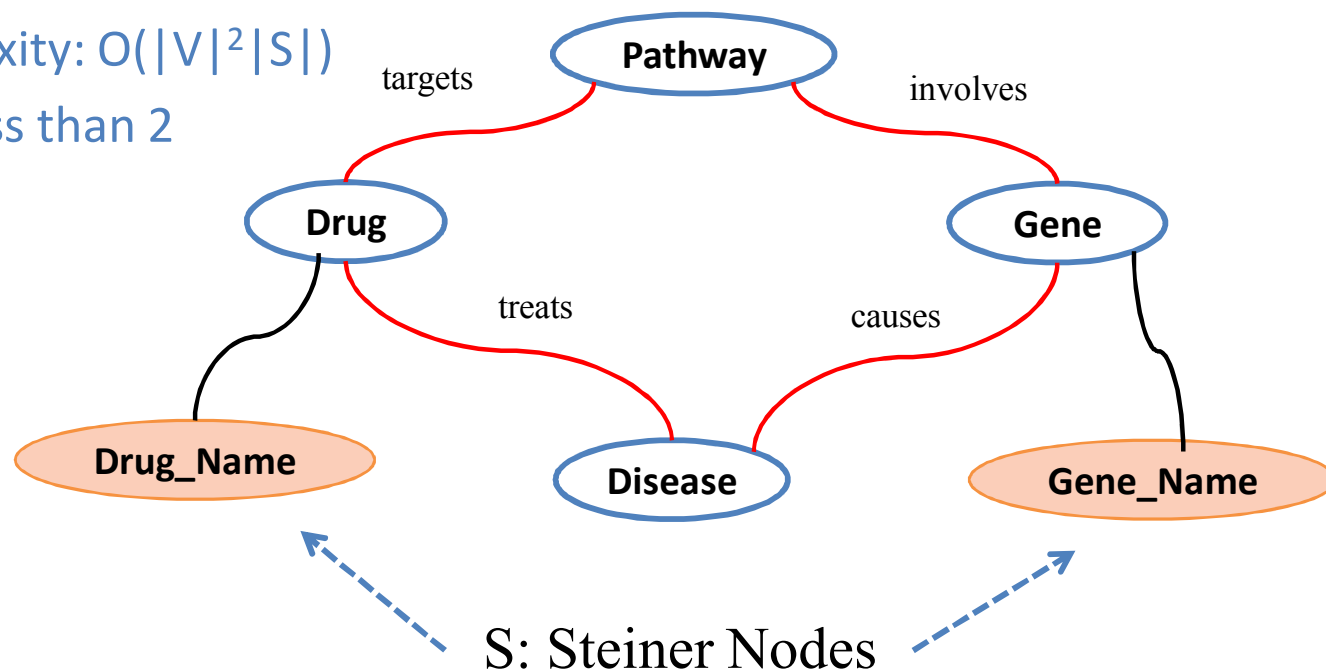
Construct a graph from semantic types and ontology





Inferring the Relationships

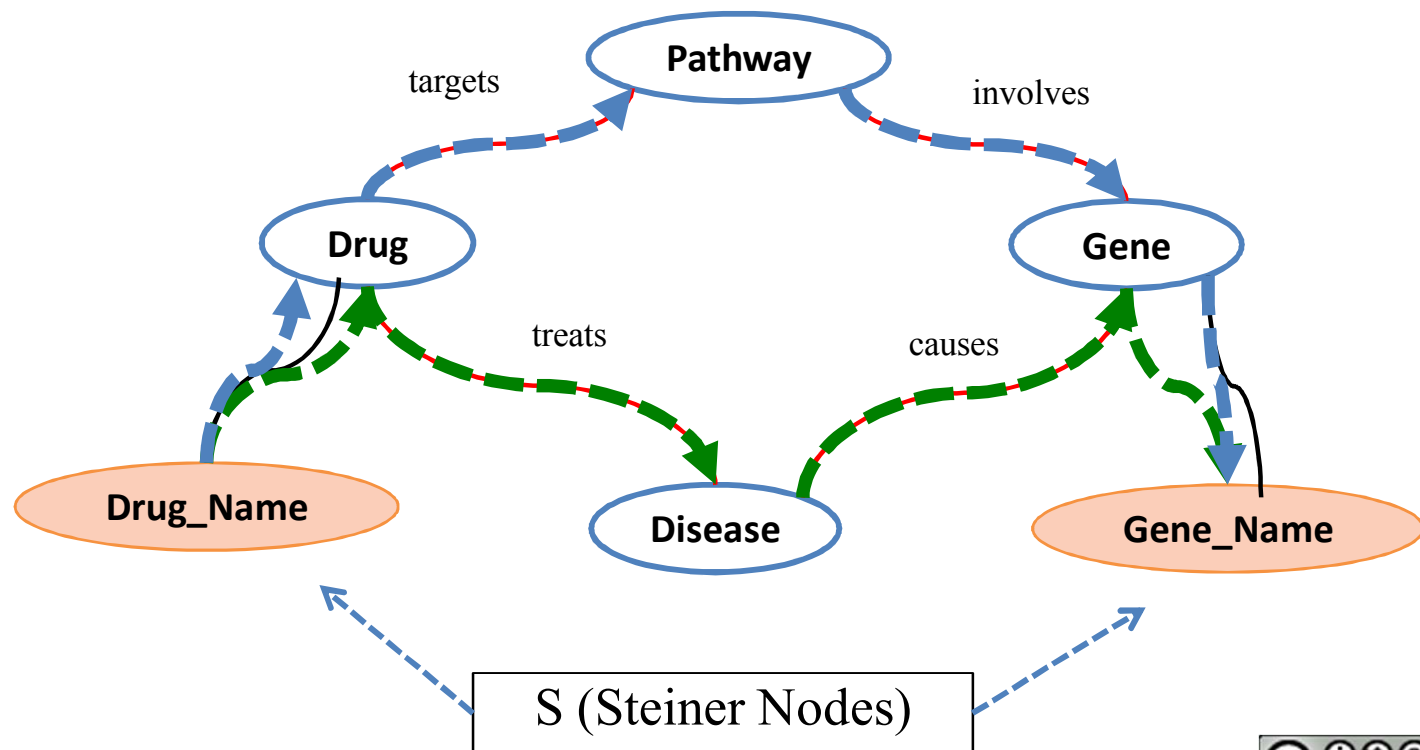
- Search for minimal explanation (source description)
- Steiner tree connecting semantic types over ontology graph
 - Given graph $G=(V,E)$, nodes $S \subset V$, cost $c: E \rightarrow \mathbb{R}$
 - Find a tree of G that spans S with minimal total cost
 - Unfortunately, NP-complete
- Approximation Algorithm [KMB, 1981]
 - Worst-case time complexity: $O(|V|^2|S|)$
 - Approximation Ratio: less than 2

Drug_Name	Gene_Name
Antineoplastic	ABCB1
Antineoplastic	ABCC4
Atorvastatin	ABCB1



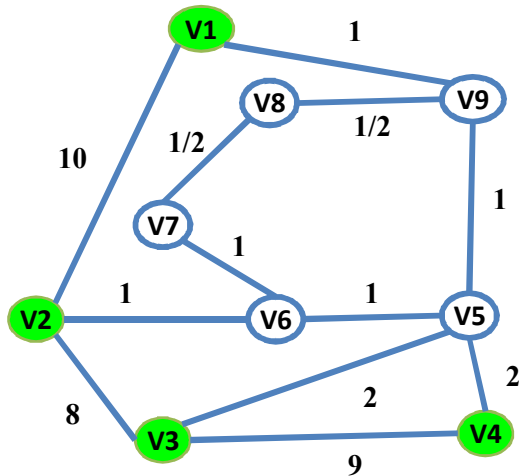
Inferring the Relationships

- Search for minimal explanation (source description)
- Multiple explanations:
 - Drug that targets pathway that involves gene ()
 - Drug that treats disease caused by gene ()

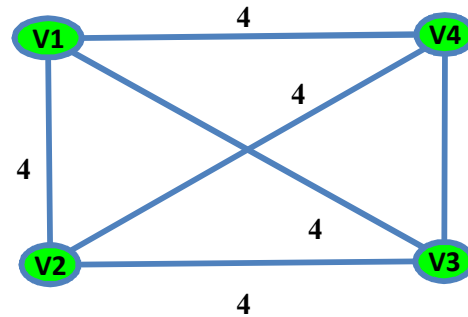


Drug_Name	Gene_Name
Antineoplastic	ABCB1
Antineoplastic	ABCC4
Atorvastatin	ABCB1

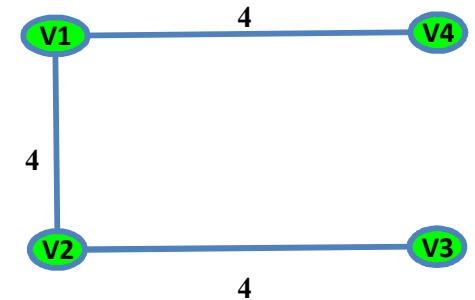
Steiner Tree Algorithm



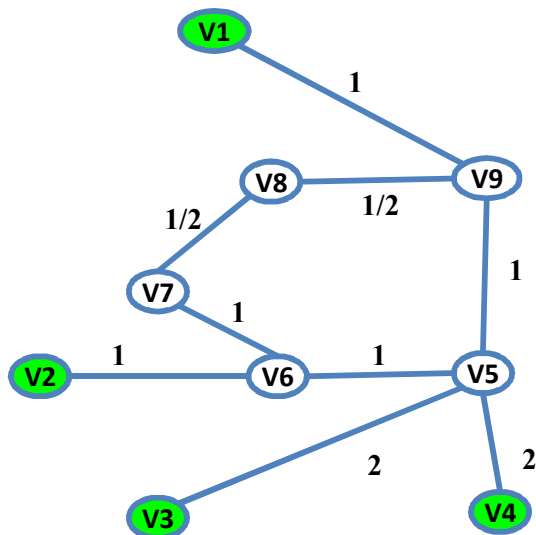
Steiner nodes: {V1, V2, V3, V4}



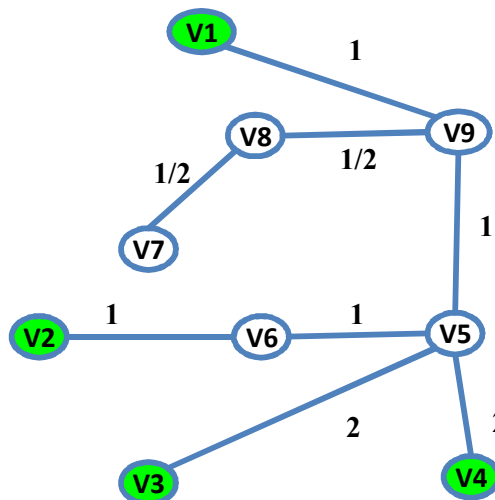
1. construct the complete graph (Nodes: Steiner Nodes, Links Weights: shortest path from each pair in original G)



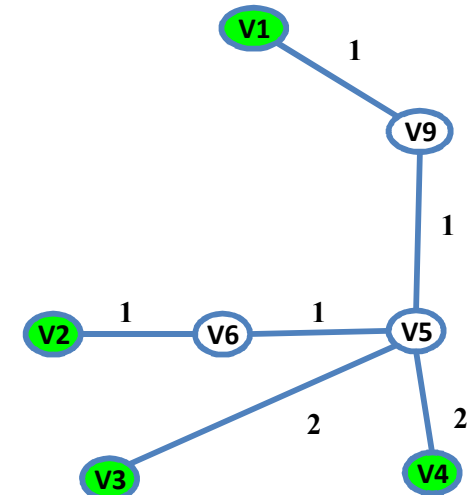
2. Compute MST



3. replace each link with the corresponding shortest path in original G



4. Compute MST



5. remove extra links until all leaves are Steiner nodes

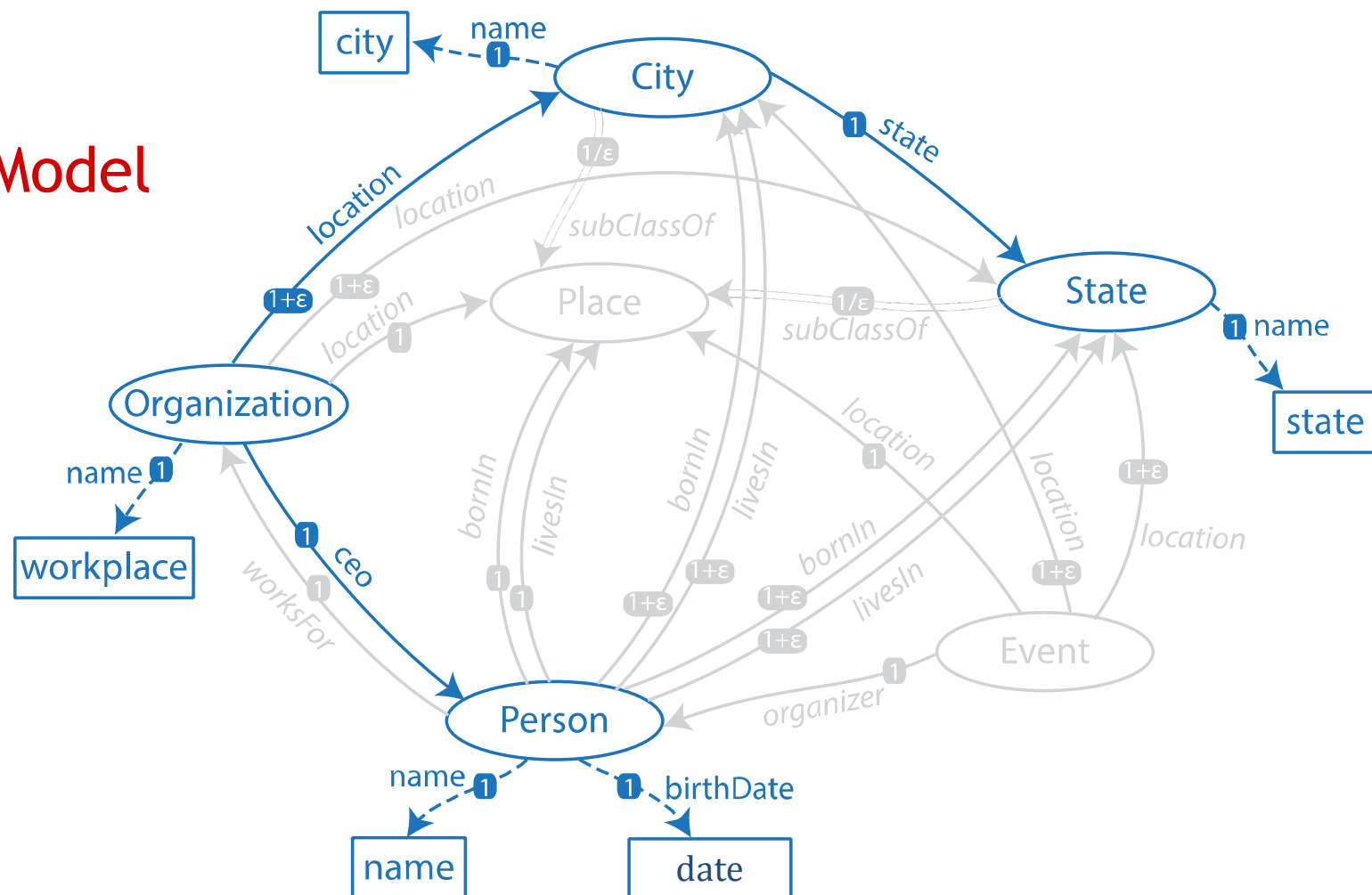


Determine Relationships

Select minimal tree that connects all semantic types

- A customized **Steiner tree algorithm** [Kou & Markowsky, 1981]

Initial Model



Result in Karma

personallInfo

UTF-8

Organization1

ceo

Person1

City1

state location

State1

name

birthDate

name

name

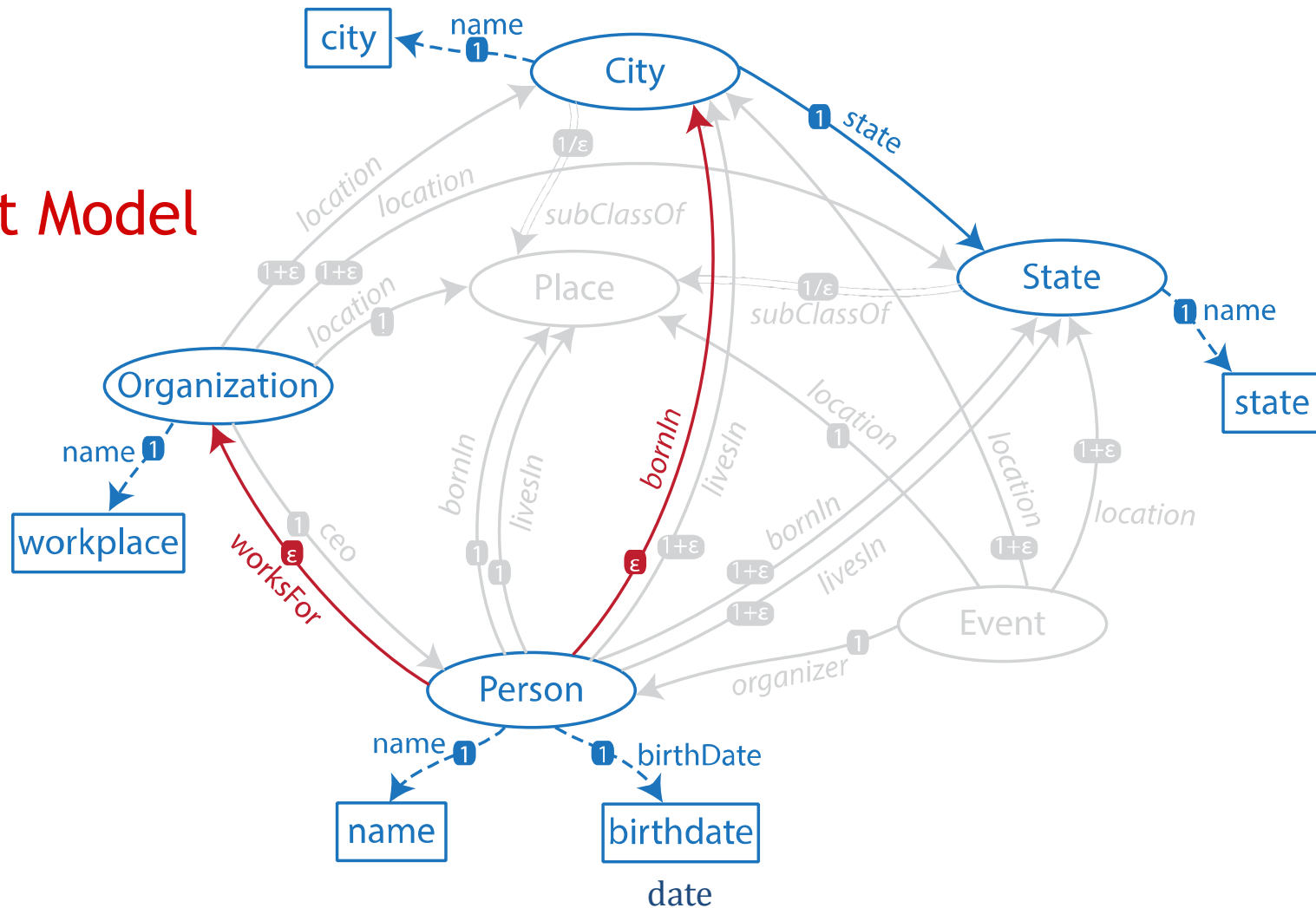
name

name	date	city	state	workplace
Fred Collins	Oct 1959	Seattle	WA	Microsoft
Tina Peterson	May 1980	New York	NY	Google
Richard Smith	Feb 1975	Los Angeles	CA	Apple

Refining the Model

Impose constraints on Steiner Tree Algorithm

Correct Model



Final Semantic Model

personallInfo

UTF-8

Person1

bornIn

City1

worksFor

Organization1

state

State1

name

birthDate

name

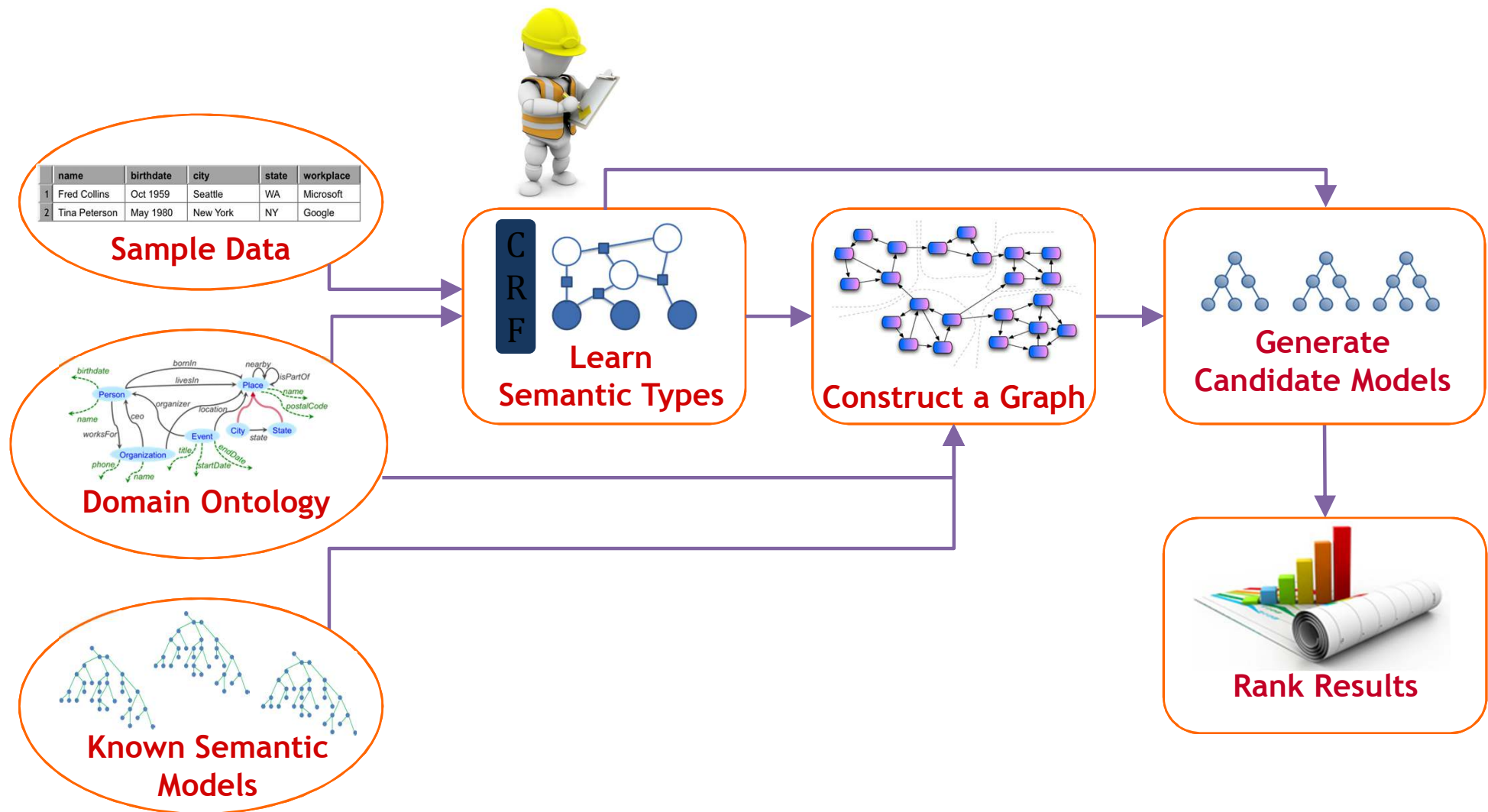
name

name

name	date	city	state	workplace
Fred Collins	Oct 1959	Seattle	WA	Microsoft
Tina Peterson	May 1980	New York	NY	Google
Richard Smith	Feb 1975	Los Angeles	CA	Apple

Improved Approach

Taheriyani et al., ISWC 2013, ICSC 2014



Results on 17 Geospatial Sources

Source Signature	#Attributes	GED	
		Previous work	Current Approach
nearestCity(lat, lng, city, state, country)	5	6	1
findRestaurant(zipcode, restaurantName, phone, address)	4	1	0
zipcodesInCity(city, state, postalCode)	3	3	1
parseAddress(address, city, state, zipcode, country)	5	6	1
citiesOfState(state, city)	2	1	0
ocean(lat, lng, name)	3	2	1
postalCodeLookup(zipCode, city, state, country)	4	6	1
country(lat, lng, code, name)	4	2	0
companyCEO(company, name)	2	1	0
personallInfo(firstname, lastname, birthdate, brithCity, birthCountry)	5	4	1
businessInfo(company, phone, homepage, city, country, name)	6	10	8
restaurantChef(restaurant, firstname, lastname)	3	2	1
findSchool(city, state, name, code, homepage, ranking, dean)	7	8	6
employees(organization, firstname, lastname, birthdate)	4	1	2
education(person, hometown, homecountry, school, city, country)	6	9	4
administrativeDistrict(city, province, country)	3	4	1
capital(country, city)	2	2	1
TOTAL	68	68	29

57% improvement

GED = Graph Edit Distance

Results on 6 Museum Sources

Source Signature	#Attributes	GED	
		Previous work	Current Approach
S1(Attribution, BeginDate, EndDate, Title, Dated, Medium, Dimensions)	7	1	0
S2(ObjectID, ObjectTitle, ObjectWorkType, ArtistName, ArtistBirthDate, ArtistDeathDate, ObjectEarliestDate, ObjectRights, ObjectFacetValue1)	9	2	3
S3(death, birth, name)	3	0	0
S4(accessionNumber, artist, creditLine, dimensions, imageURL, materials, relatedArtworksURL, creationDate, provenance, keywordValues)	10	9	6
S5(AccessionNumber, Classification, CreditLine, Date, Description, DimensionsOrphan, WhatValues, Who, image, relatedArtworksValues)	10	9	5
S6(Artist, ArtistBornDate, ArtistDiedDate, Classification, Copyright, CreditLine, Image, KeywordValues, Ref, SitterValues)	10	8	6
TOTAL	49	29	20

GED = Graph Edit Distance

31% improvement

Karma Use Cases



Source Mapping Phase



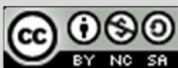
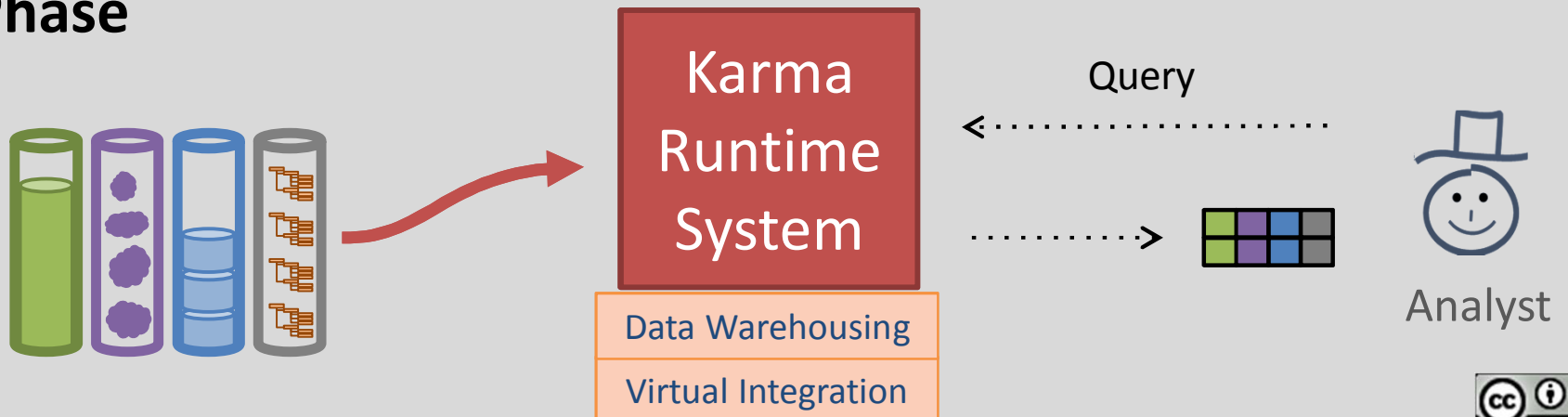
Mapping Phase

Source Mapping and Query Time



Mapping Phase

Query Phase



Related Work

- Mapping Databases into RDF
 - **D2R & R2R** [Bizer & Cyganiak, 2006, Bizer & Shultz, 2010]
 - **Semion** [Nuzzolese, Gangemi, Presutti, & Ciancarini, 2010]
 - Maps a database into RDF using the DB schema
 - Manually defines the mappings of triples to another ontology
- Ontology Matching
 - [Doan et al., 2000]
 - Learn mappings to the ontology using data, but would be analogous to just doing the semantic typing
- Schema Matching
 - [Rahm et al., 2001]
 - Generates alignments between schemas, not a fine-grained model of the data
- Schema mapping
 - Interactively builds detailed mappings, but limited to relational data (Clio [Fagin et al., 2009])
- Semantic Integration of Bioinformatics Data
 - **Bio2RDF** [Belleau et al., 2008]
 - Manual conversion of sources into RDF


Links

<http://www.isi.edu/integration/karma/>

Karma: A Data Integration Tool

Craig Knoblock, Pedro Szekely, Jose Luis Ambite, Shubham Gupta, Maria Muslea, Mohsen Taheriyani, Bo Wu

Information Sciences Institute, University of Southern California



Karma is an information integration tool that enables users to quickly and easily integrate data from a variety of data sources including databases, spreadsheets, delimited text files, XML, JSON, KML and Web APIs. Users integrate information by modeling it according to an ontology of their choice using a graphical user interface that automates much of the process. Karma learns to recognize the mapping of data to ontology classes and then uses the ontology to propose a model that ties together these classes. Users then interact with the system to adjust the automatically generated model. During this process, users can transform the data as needed to normalize data expressed in different formats and to restructure it. Once the model is complete, users can published the integrated data as RDF or store it in a database.

A cool video that illustrates why Karma is significant: Data Sharing and Management Snafu in 3 Short Acts

Karma Innovations

- Ease of use:** Karma uses programming-by-example, learning techniques and a Steiner tree optimization algorithm to automate as much of the process as possible to enable end-users to map their data to a chosen ontology. Users adjust the automatically generated model using a graphical

DOWNLOAD
Karma is available as open source (Apache 2 License): [download](#)

NEWS

Tweets [Follow @KarmaSemWeb](#)

Karma Project at USC @KarmaSemWeb 20 Dec
New release of #Karma available on GitHub with many bug fixes and connectivity to Oracle
[github.com/InformationInt...](#)
[Show Summary](#)

Karma Project at USC @KarmaSemWeb 8 Dec
Mohsen @Taheriyani presented our Karma paper bit.ly/SqpP9 at #ISWC2012, check out the video at bit.ly/TFWFP3
[Expand](#)

Karma Project at USC @KarmaSemWeb 5 Dec
Smithsonian joins #USC @karmasemweb in effort to make digital museum art records more accessible, searchable
[washingtonpost.com/local/smithson...](#)
[Show Summary](#)

Craig Knoblock @cainoblock 4 Dec
USC Researchers Guiding the Smithsonian American Art Museum to the Next Generation of the Internet
[createtrend.com/v/-DEB7487Q32...](#)
[Retweeted by Karma Project at USC](#)
[Tweet to @KarmaSemWeb](#)