From models to theories

(HP2T)

Index

- **World models – intuition**
- Words, alphabet
- Assertional languages
- Assertions
- Assertional theories
- Key notions

World models - intuition

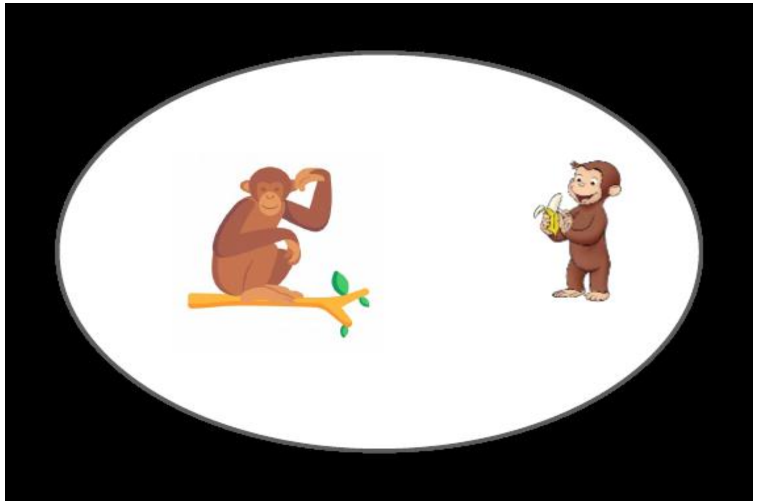
Intuition (World models): **Models** depict a specific instance of the world, e.g., the world under observation. This instance is called the **intended model**. Representations should

1. … take in input the information encoded in the intended model
2. … define an appropriate linguistic representation of the intended model
3. … make as explicit as possible the mapping from the linguistic representation to the intended model

A **World Model** is any representation which encodes the three types of information described above.

Linguistic vs. analogic representations

- There is a tree
- There is a banana
- The monkey is eating a banana
- The monkey is sitting on a tree
- The monkey is scratching its head



Observation 1. Model theory provides us with a formal mechanism for formalizing the content of the analogical representation on the right.

Observation 2. Which monkey? There are two monkeys! The language used in the linguistic representation is ambiguous! We need to provide a formalization of the linguistic representation on the left.

Observation 3. The intended model and its linguistic description are the two key elements, neither of which can be neglected, of world models.

World models – intuition (continued)

Example (world models): Examples of world models are all the reference models used in Computer Science (see dedicated lecture). **In none of them the intended model is formalized**. For instance:

1. Any description of the world in **natural language** is a world model. The syntax if informally defined. The intended model is left implicit relying on the commonsense meaning of words. The interpretation of the speaker's intended model relies exclusively on the personal knowledge of the listener.
2. **ER and EER models** describe the properties and relations between etypes. The syntax is formally defined. With respect to natural languages, an important simplification derives from a much clearer and simper syntax.
3. **Relational databases** describe the properties and relations among entities. The syntax is formally defined. The situation is slightly improved by the unique names assumption, that is: different words have different meanings, multiple occurrences of the same word have the same meaning.

At different levels, all these world models suffer from the **problem of ambiguity.**

Index

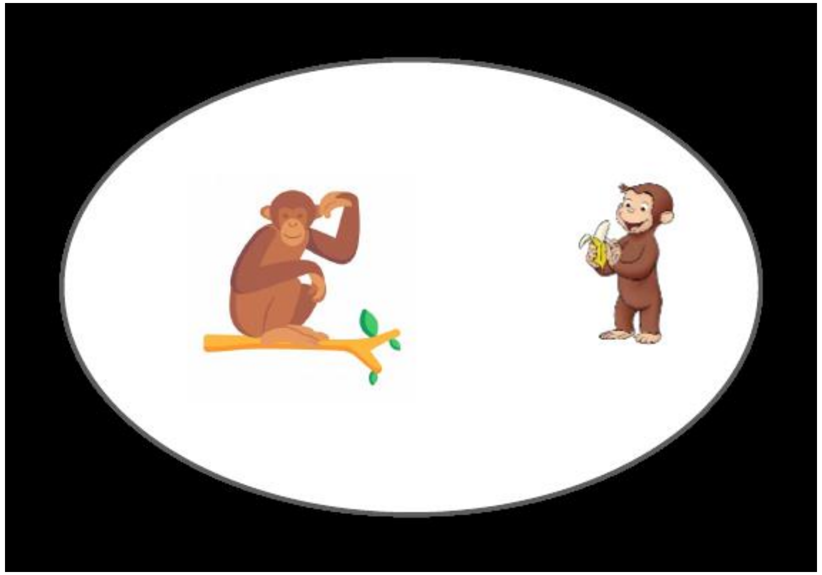
- World models - intuition
- **Words, alphabet**
- Assertional languages
- Assertions
- Assertional theories
- Key notions

Percepts, words, concepts

Intuition (percept, word) A percept, to be a percept, must be linguistically described as such. We say that a percept **p** is **named** or, also, **denoted** by a **word w**, if **p** is the **intended meaning** of **w.** We also say that **p** is the **interpretation** of **w.**

Intuition (concept)(1) A **concept** is the mapping **from a word to the percept it names**. Words are also said to be **lexicalized concepts.**

Example (concept). If the picture on the left is an analogical representation of a domain and intended model, the following are examples of concepts.



- monkey#1: < Chita, “the monkey you see left”>
- monkey#2: < The_monkey_on_the_tree, “the monkey you see on the tree”>
- Monkey: <{monkey#1, monkey#2}>

(1) Miller, G. A., Beckwith, R., Fellbaum, C., Gross, D., & Miller, K. J. (1990). Introduction to WordNet: An on-line lexical database. *Int. journal of lexicography*, *3*(4), 235-244.

Alphabet

Intuition (alphabet). An **alphabet A** is a set of words **w**, that is

$$\mathbf{A} = \{\mathbf{w}\}$$

Observation (word, alphabet) We need to have words which allow us to name all domain **percepts** of relevance, that is, entities, entity properties, entity relations, etypes, etype properties, etype relations.

Observation (percept, concept). Each percept is associated one and only percept.

Concepts implicit in words (example)

Example (concepts). We have the following:

- a *name* naming an entity (e.g., *Stefania*)
- An *adjective* naming a property of an entity or a set of entities (e.g., *high, beautiful*)
- a *noun* naming a relation between entities or sets of entities (e.g., *friend, owner*)
- a *verb* naming a relation (e.g., to *talk to, to walk with*)
- a *noun* naming a class (e.g., *person, female*)
- a verb naming a class of events (e.g., talking, walking)

Concepts implicit in words (example, continued)

Example (Using words in sentences). We have the following:

- Stefania is beautiful
- Stefania home beautiful (not a sentence!)
- Friends help *(the word friend denotes a class)*
- Stefania is a friend of Mario *(the word friend denotes a relation)*
- Mario has Stefania as a friend *(the word friend denotes a relation, inverse with respect to the previous example)*

Observation 1 (Ambiguity: one word, two different types of percept). For instance, above, the word friend denotes an entity (thing) a class, and a relation

Observation 2 (Ambiguity: one word, two different percepts of the same type). For instance: Java (denotes 2 etype and one entity), car (denotes more than one etype)

Index

- World models - intuition
- Words, alphabet
- **Assertional languages**
- Assertions
- Assertional theories
- Key notions

Languages

Intuition (Language). We take Chomsky's notion of language. Intutively, a **language L** is any notation (**Alphabet + Formation Rules)** which can be used to generate sentences, defined by humans, agreed upon by humans, which allows to describe and share information about analogical representations.

Examples (Language). Any natural language, the language of signs, Java, Python, the graphic notation of ER / EER Graphs, or of tables.

Assertional Languages – Intuition

Intuition (Assertional language). An assertional language L_a is a language which is used to describe all the elements of an analogical representation, that is: percepts, domains, facts, models.

Example (Assertional language). All the reference models, use an assertional language. Natural language allows to say us much more, but can be restricted to allow only for assertions.

Observation (Assertional language). Some assertional languages are graphical. All the assertional languages in logical formal models are type 3 languages.

Assertional Languages

Definition (Assertional Language). Assertional languages are pairs

$$\mathbf{L}_a = \langle \mathbf{A}_a, \mathbf{FR} \rangle$$

where

$$\mathbf{A}_a = \{\mathbf{w}\}$$

is an assertional **alphabet $\mathbf{A}_a$** , that is, a set of words **w** that can be used to write assetions, and

$$\mathbf{FR} = \{\mathbf{fr}\}$$

is a set of **formation rules fr**, with the restriction that they must be able to describe the facts of domains of interpretation.

From Domains to Assertional Languages

Observation (Domain) As from before, a Domain D is defined as follows:

$$D= \{p\}$$

where percepts p are of three types, that is

$$D = < U, \{C\}, \{R\} >$$

where: U is a set of **entities**, {C} is a set of **classes** of entities, {R} is a set of n-ary **relations** among entities.

Observation (Assertional Languages). Similarly to domains, an assertional language L_A is defined as follows:

$$L_a = <A_a, FR>$$

where different types of words describe different types of percepts and the formation rules provuse the means to compose words as a function of how percepts compose to build facts **(compositionality of semantics).**

From Domains to Assertional Languages (continued)

Observation 1 (Choosing percepts) When describing the intended model the key element is the choice of percepts:

- For any percept p ∈ D= {p}, which is **of relevance** to the current modeling task, define a distinct word w ∈ A_a = {w}.

The choice of the percepts defines the expressiveness of world models. Irrelevant classes of percepts can be eliminated, for instance: entities as in ER/ EER models, classes as in relational data bases. In other cases specific classes or relations are dropped.

All percepts can be dropped using words which are assertions naming facts, for instance: from «Near(Paolo,Rocky)» to «NearPaoloRocky» or «A». In this case the set of formation rules FR is empty.

From Domains to Assertional Languages (continued)

Observation (From Percepts to Words) The mapping from percepts to the specific words is in control of the modeler. The usual choice is to choose words whose intuitive meaning are the intended percepts.

Observation 2 (Choosing the formation rules). There are world models which consider the same percepts and are different only in the formation rules they consider. The choice of the formation rules is a modeling decision, driven by the goal of suitably modeling the intended analogical representation. See examples below.

Example – Natural language alphabet and formation rules

Example (Natural language). The origins of Trento on the river-route to Bolzano and the low Alpine passes of Brenner and the Reschen Pass over the Alps **are disputed**. Some scholars maintain it was a Rhaetian settlement: the Adige area was however influenced by neighbouring populations, **including** the (Adriatic) Veneti, the Etruscans and the Gauls (a Celtic population). According to other theories, the **latter** instead founded the city during the 4th century BC.

Observation 1 (Syntax and semantics). Natural language assertions have informal syntax and semantics. Highly ambiguous. Everybody understands them, modulo ambiguity. Ambiguity is pervasive. Used in the requirements phase in most SE projects. Problem unsolved in building natural language processing AI systems

Observation (types of percepts). All types of percepts are allowed.

Example – Pseudo-natural language alphabet and formation rules

Example 1 (Pseudo-natural language). A sequence of statements:

- There is a tree.
- There is a banana.
- The monkey is eating a banana.
- Monkeys eat bananas
- Monkeys are animals

Observation 1 (syntax and semantics). Pseudo natural language assertions have informal syntax and semantics. Everybody understands them, modulo ambiguity. Ambiguity mostly limited to words. Widely used in high-value applications where the user needs to understand the details of the applications, witout having a backgournd in formal logic.

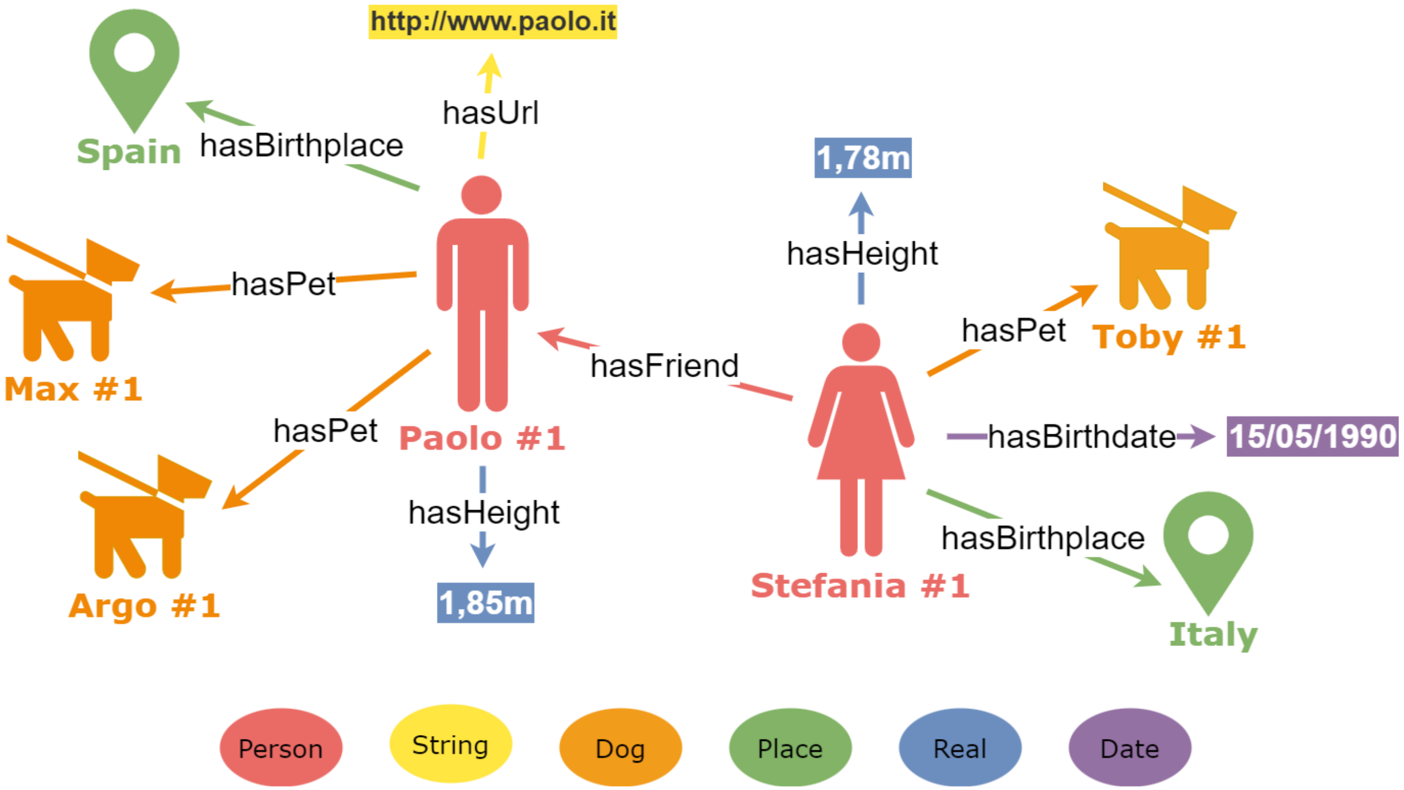
Observation (types of percepts). All types of percepts are allowed.

Example – database alphabet and formation rules

Employee			
Name	**Role**	**Nationality**	**Supervises**
Fausto	Professor	Italian	Rui
Rui	Student	Chinese	Bisu
Bisu	Student	Indian	-
entity	etype	property	Relation

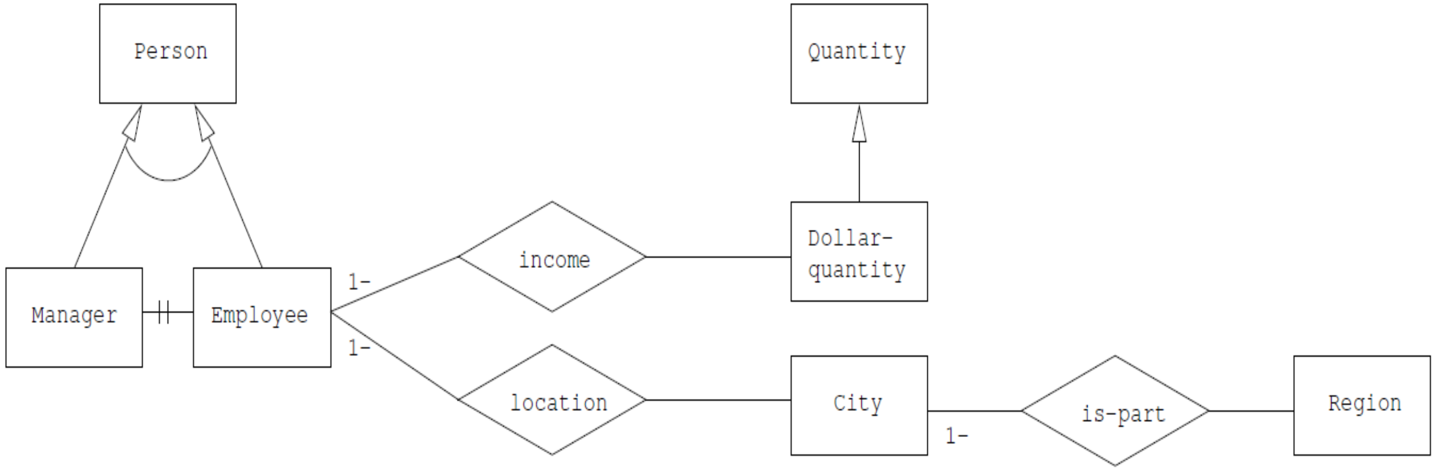
Observation. DBs use a table notation where, roughly speaking, rows describe **entities** plus their **properties** and **relations**, while columns name **etypes**, **entity properties** and **entity relations**. The formation rules are those which allow to build tables.

Example – Entity graph alphabet and formation rules



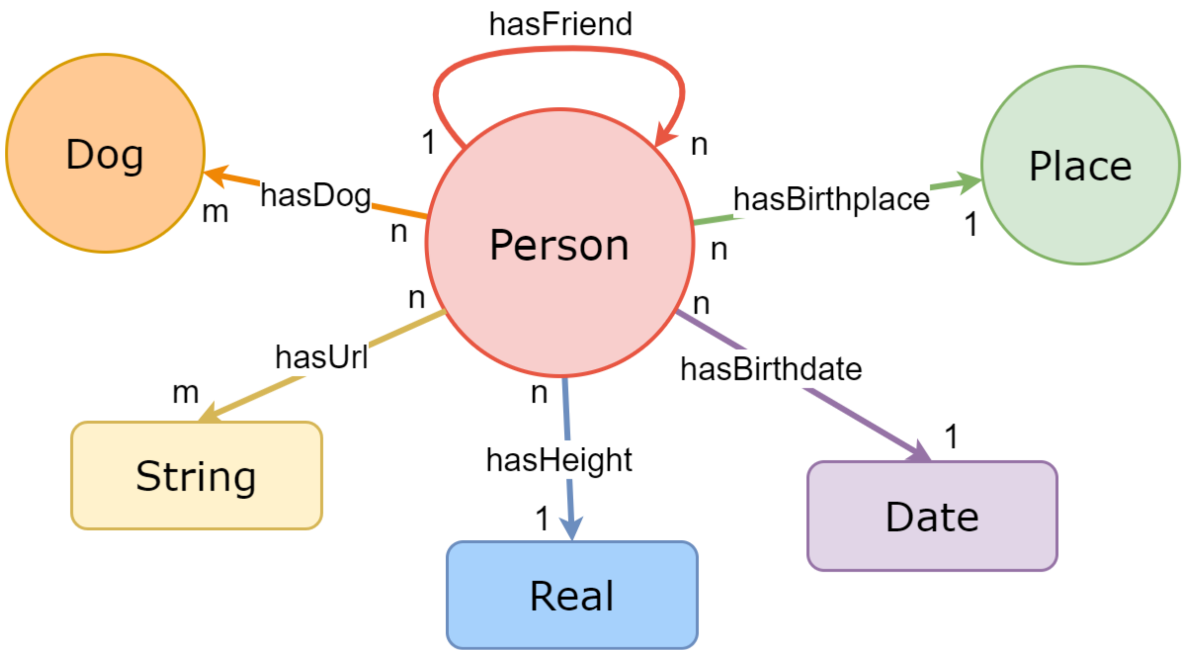
Observation. With respect to the DB language: (1) same percepts, (2) the formation rules of a graph, (3) closer to the analogical representation, (4) more complex to implement. (5) syntax and semantics are formalized.

Example –ER model alphabet and formation rules



Observation. ER models use a graph notation with boxes of various shapes where the shape carries information about **etypes** and links carry information about **entity properties** and **relations**. **No entities.** Formation rules are formalized.

Example – Etype graph alphabet and formation rules



Observation (Etype Graph). With respect to ER models: (1) same information, (2) more flexible, (3) can be implemented (4) arity notation non-standard and not suggested (5) syntax and semantics are formalized (6) etype graphs implemented as knowledge level data structures (historically called «**ontologies**»).

Index

- World models - intuition
- Words, alphabet
- Assertional languages
- **Assertions**
- Assertional theories
- Key notions

From facts to assertions

Definition (Assertion). An **Assertion** a is an **atomic sentence**, that is, a sentence which cannot be decomposed into simpler sentences, which unambiguously describes a single **fact**. For any assertion a we have

$$a \in L_a = \{a\}$$

Observation (From facts to assertions) The mapping from facts to assertions is in full control of the modeler. We have the following:

- For formation rules there is no specific recipe. The general idea is that they should generate a syntax which makes it as easy as possible for people to understand the underlying facts (see above);
- A special case, see above, is when there is no need of formation rules.

Types of assertions

Definition (Assertion). An **assertion** a has one of the following five forms

- Assertion starting fact: $u_i \in C_j$,
- Assertion starting fact: *Tuple of Units memberOf relation:* $< u_1, \ldots, u_n > \in R^n$,
- Assertion starting fact: *Class subsetOf Class:* $C_i \subseteq C_j$,
- Assertion starting fact: *Relation subsetOf relation:* $R_i^n \subseteq R_j^n$
- Assertion starting fact: *Relation subsetOf tuple of classes and viceversa:*
  - $R^n \subseteq C_1 \times \ldots \times C_n$
  - $C_1 \times \ldots \times C_n \subseteq R^n$

Different assertional languages differ in the choice of facts to be asserted. See examples below.

Example – Natural language assertions

Example (Natural language, example assertions in yellow). "The origins of Trento on the river-route to Bolzano and the low Alpine passes of Brenner and the Reschen Pass over the Alps **are disputed**." Some scholars maintain "it was a Rhaetian settlement:" the "Adige area was however influenced by neighbouring populations", **including** the (Adriatic) Veneti, the Etruscans and the Gauls (a Celtic population). According to other theories, "the **latter** instead founded the city during the 4th century BC".

Observation (types of assertions). All types of assertions are allowed.

Example – Pseudo - natural language alphabet and formation rules

Example 1 (Pseudo-natural language). A sequence of statements:

- There is a tree.
- There is a banana.
- The monkey is eating a banana.
- Monkeys eat bananas.
- Monkeys are animals.

Observation (types of assertions). All types of assertions are allowed.

Example – sets of database assertions

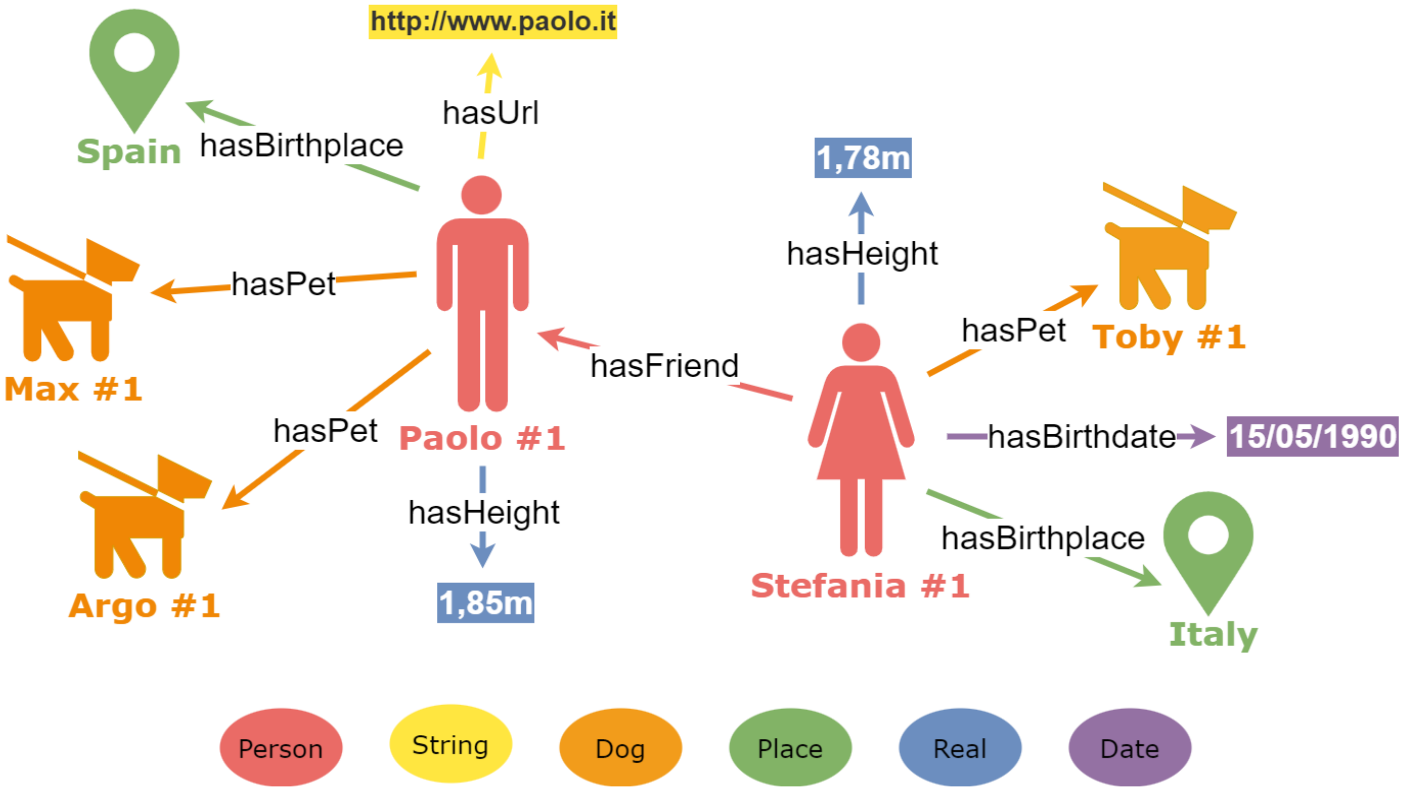
Employee			
Name	**Role**	**Nationality**	**Supervises**
Fausto	Professor	Italian	Rui
Rui	Student	Chinese	Bisu
Bisu	Student	Indian	-
entity	etype	property	Relation

Observation. DBs use a table notation where, roughly speaking, rows describe entities plus their properties and relations, while columns name entities, etypes, entity properties and entity relations.

DB assertions (in Logic) = {Professor(Fausto), Student(Rui), Student(Bisu), Nationality(Fausto, Italian), Nationality (Rui, Chinese), Nationality (Bisu, Indian), Supervises(Fausto, Rui), Supervises(Rui, Bisu)}

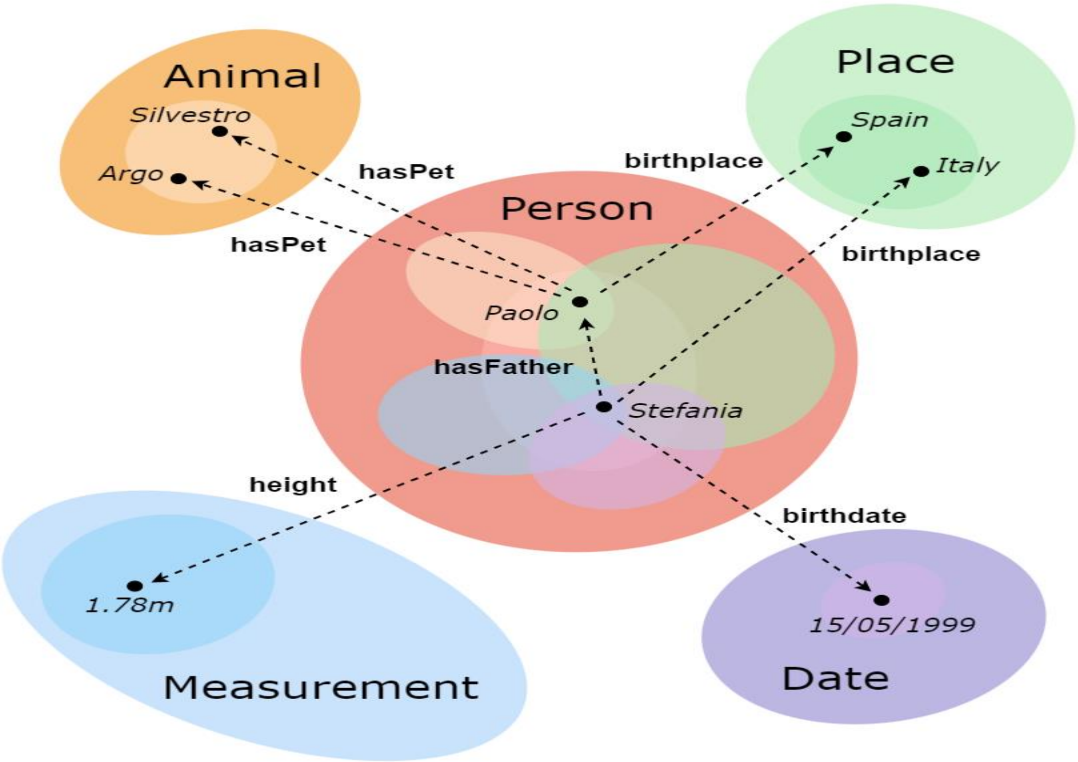
Observation (Types of assertions). Only assertions about entities**.** Etype assertions are **not** allowed in DBs.

Example – sets of Entity graph assertions



Observation (Entity Graph). With respect the DB language: (1) same assertions, (2) more flexible, (3) closer to the analogical representation, (4) more complex to implement. (5) syntax and semantics are formalized.

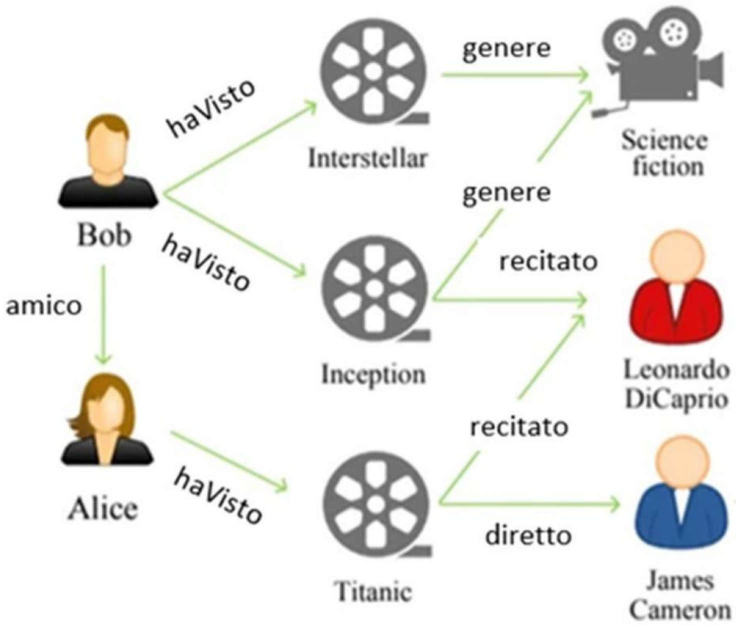
Example – sets of Entity graph facts / assertions



Observation (Venn diagram of an entity Graph). A Venn diagram (analogic) representation of an entity Graph (non standard).

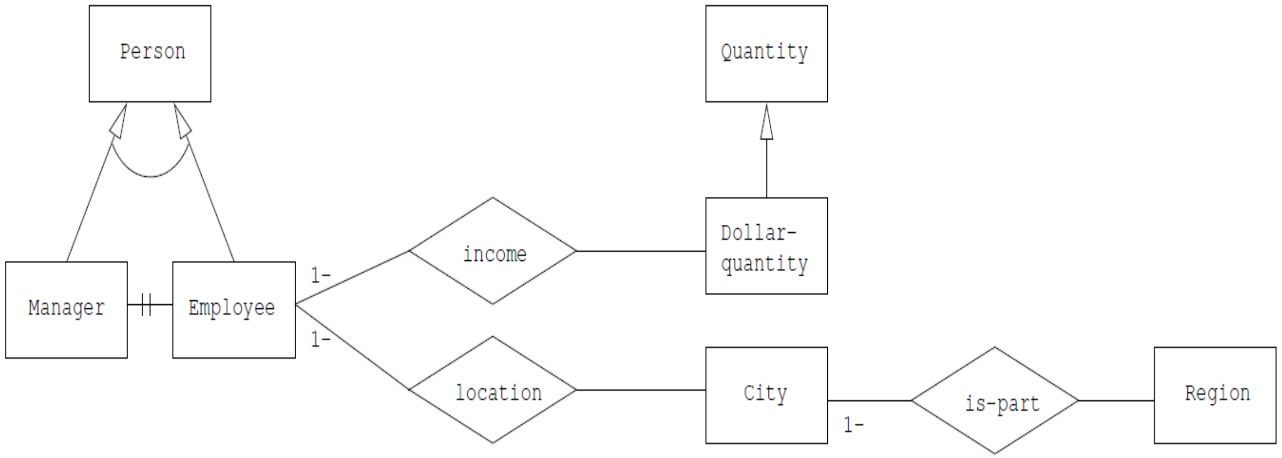
- Entities as nodes
- Etypes as sets of entities
- Relations as sets of tuples

Example – sets of Knowledge graph assertions



Observation (Knowledge Graph KG). A KG is a graph whose language and semantics are not formalized and can be anything (no precise formation rules or definition of which percepts are allowed). In the graph above, all nodes are entities.

Example – sets of ER assertions



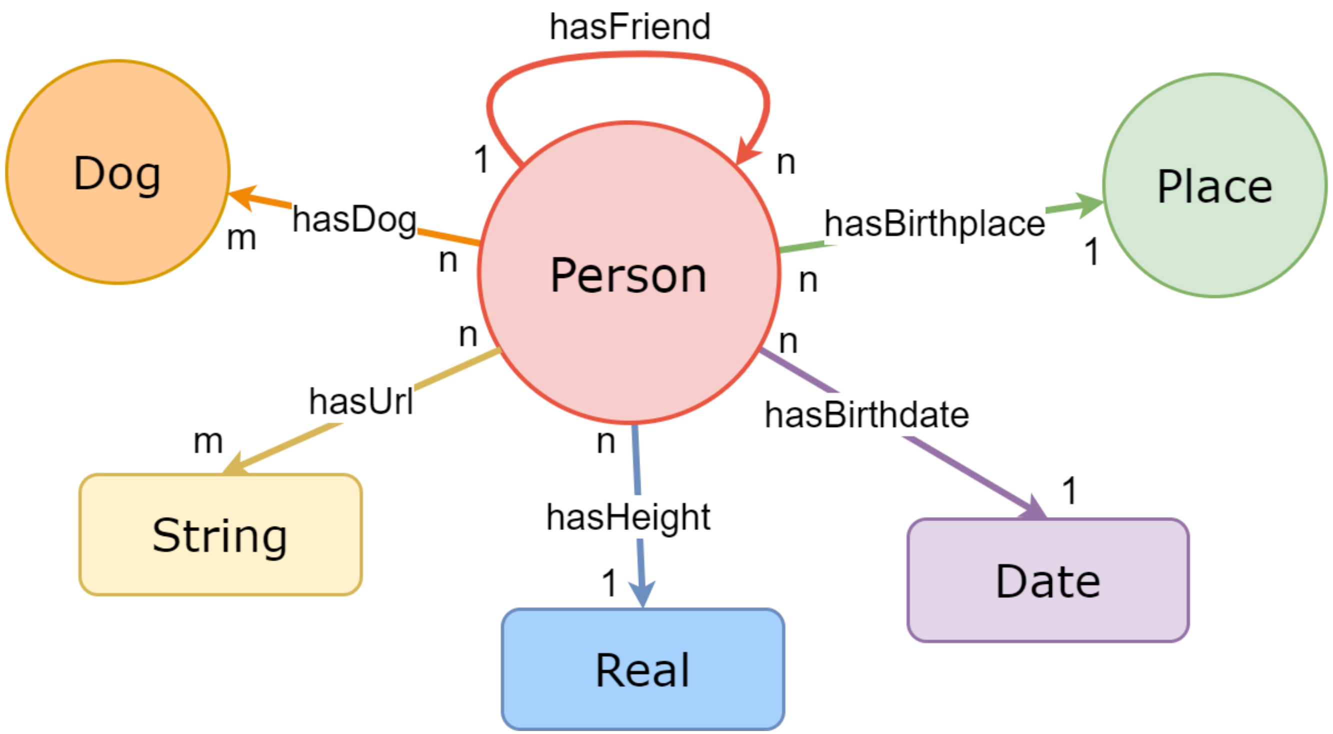
ER assertions relating to etypes (in Logic):

{Manager ⊑ Person, Employee ⊑ Person}

Observation (Intuitive semantics, examples). Professor is an etype, ⊑ semantically means subset, Fausto is a entity, Professor(Fausto) means that the etypes of Fausto is professor.

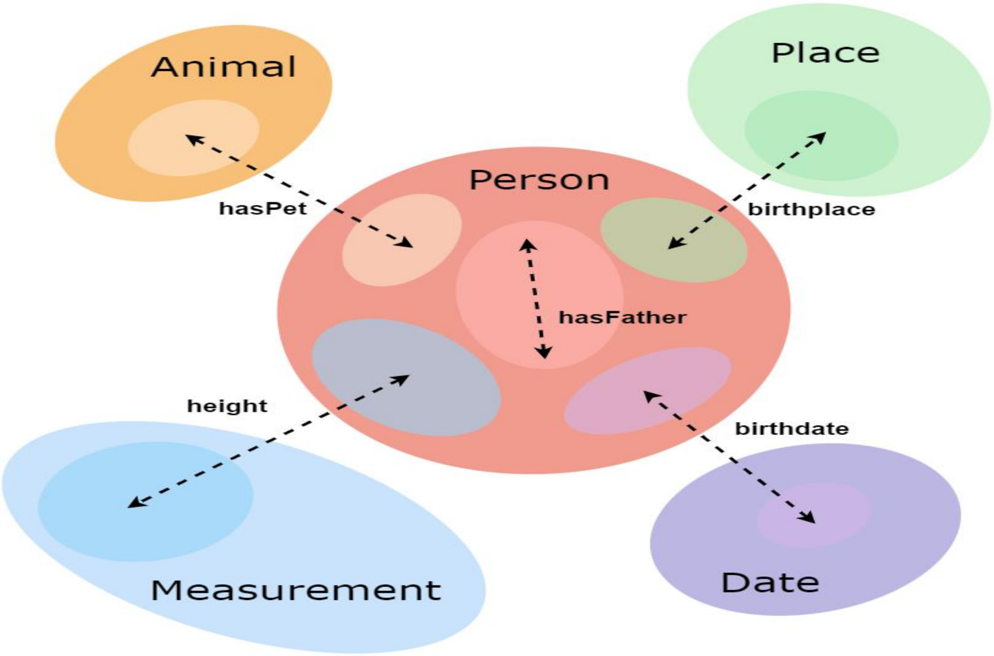
Observation (Types of assertions). Only assertions about etypes**.** Entity assertions are **not** allowed in ER models.

Example – sets of Etype assertions



Observation (Etype Graph). With respect the ER language: (1) same information, (2) more flexible, (3) can be implemented (4) arity notation non-standard and not suggested (5) syntax and semantics are formalized.

Example – sets of etype facts /assertions



Observation (Venn diagram of an entity Graph). A Venn diagram (analogic) representation of an entity Graph.

- No Entities
- Etypes as «zoomed» entities
- Relations as tuples of etypes

Observation (Etype Graph). A Venn diagram representation of an etype Graph. Not very immediate. Compare with the Venn diagram representation of the entity graph

Example – sets of concept assertions

Observation (Concept Graph CG). A snapshot of the **WordNet** output. Wordnet a **digitalized** lexicon which encodes how the meaning of words (concepts) is connected, so-called **Lexical-Semantics. Built to be used by machines»**

Example (linguistic assertions). Some examples (IsA means «subset», sameAs means «the same set»):

- Koto IsA stringed instrument
- Stringed instrument IsA musical instrument
- Musical instrument sameAs instrument
- Instrument IsA device
- ...
- Unit IsA Object
- Object sameAs physical object
- Physical object IsA Physical entity
- Physical entity IsA Entity
- ***Entity sameAs percept /* In this course */***

- S: (n) **koto** (Japanese stringed instrument that resembles a zither; has a rectangular wooden sounding board and usually 13 silk strings that are plucked with the fingers)
  - *direct hypernym* / ***inherited hypernym*** / *sister term*
    - S: (n) stringed instrument (a musical instrument in which taut strings provide the source of sound)
      - S: (n) musical instrument, instrument (any of various devices or contrivances that can be used to produce musical tones or sounds)
        - S: (n) device (an instrumentality invented for a particular purpose) *"the device is small enough to wear on your wrist"; "a device intended to conserve water"*
          - S: (n) instrumentality, instrumentation (an artifact (or system of artifacts) that is instrumental in accomplishing some end)
            - S: (n) artifact, artefact (a man-made object taken as a whole)
              - S: (n) whole, unit (an assemblage of parts that is regarded as a single entity) *"how big is that part compared to the whole?"; "the team is a unit"*
                - S: (n) object, physical object (a tangible and visible entity; an entity that can cast a shadow) *"it was full of rackets, balls and other objects"*
                  - S: (n) physical entity (an entity that has physical existence)
                    - S: (n) entity (that which is perceived or known or inferred to have its own distinct existence (living or nonliving))

Observation (Types of assertions). Only assertions about the meaning / **extension** of concepts are allowed.

From Domains to Assertional Languages (continued)

Observation (From Domains to Assertional languages) When describing the intended model (described using model theory) the obvious way to define a language is as follows:

- For any percept p ∈ D= {p}, which is **of relevance** to the current modeling task, define a distinct word w ∈ A_a = {w};
- For any fact f ∈ D = {f}, which is **of relevance** to the current domain, construct a distinct assertion a ∈ L_a = $\{a\}$, where the set formation rules FR = {fr} allows to construct assertions following a process similar to that used in constructing facts from percepts.

From Domains to Assertional Languages (continued)

Observation (Domain) As from before, a Domain D can be seen in two ways, as follows:

- D = < U, {C}, {R} >
- D = {f}, that is, the set of all and only the **facts** f that can be composed from percepts.

The first view is called the **intensional** view $\mathbf{D^i}$, the second the **extensional** view $\mathbf{D^e}$.

Observation (Assertional Language). Similarly to domains, an assertional language L_A can be seen in two ways, as follows:

- L_a = <A_a, FR>
- $\mathbf{L_a = \{a\}}$**,** that is, the set of all and only the sentences a which can be composed by applying the formation rules fr ∈ FR to the words in A_a

The first view is called the **intensional** view $\mathbf{L_a{}^i}$, the second the **extensional** view $\mathbf{L_a{}^e}$.

Index

- World models - intuition
- Words
- Assertional languages
- Assertions
- **Assertional theories**
- Key notions

Assertional theories

Observation (Assertional theories). Assertions are linguistic descriptions of facts. **Assertional theories** are linguistic descriptions of models

Definition (Assertional theory) An **assertional theory** T_a is a set of assertions

$$\mathbf{T}_{a} = \{a\}$$

Observation (Assertional theory, assertional language). Given an assertional language L_a, an assertional theory T_a is a subset of L_a.

$$T_a = \{a\} \subseteq L_a$$

Example (Assertional theories). All the three examples above of sets of assertions are assertional theories.

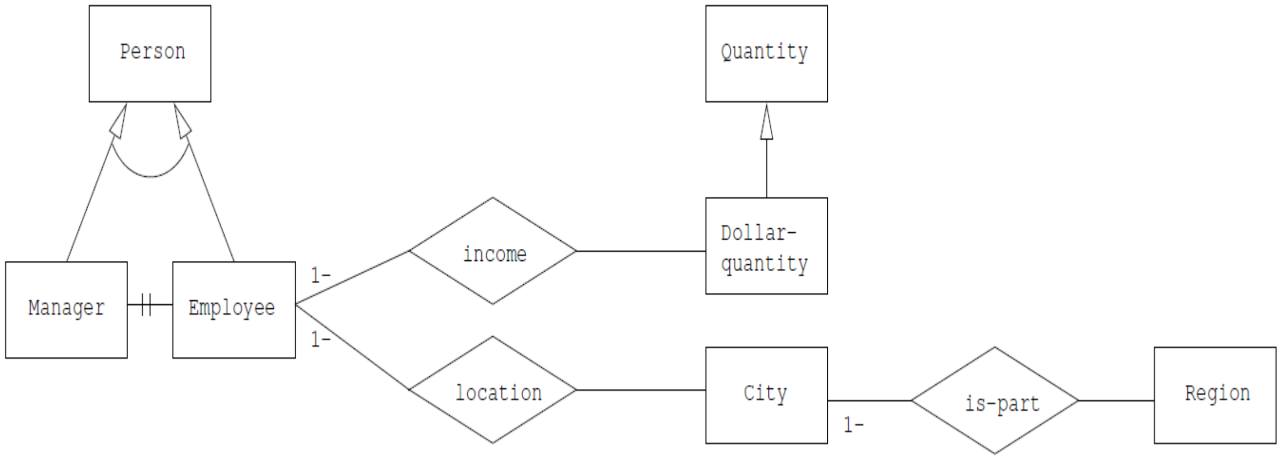
Example – a DB generated Theory

Employee			
Name	**Role**	**Nationality**	**Supervises**
Fausto	Professor	Italian	Rui
Rui	Student	Chinese	Bisu
Bisu	Student	Indian	-
entity	etype	property	Relation

T$_a$ =

{Professor(Fausto), Student(Rui), Student(Bisu),
Nationality(Fausto, Italian), Nationality (Rui, Chinese),
Nationality (Bisu, Indian), Supervises(Fausto, Rui),
Supervises(Rui, Bisu)}

Example – An ER model generate theory



$\mathbf{T}_a$ =

{Manager $\sqsubseteq$ Person, Employee $\sqsubseteq$ Person, …}

Index

- World models - intuition
- Words, alphabet
- Assertional languages
- Assertions
- Assertional theories
- **Key notions**

Set theory – Terminology

Terms naming components of linguistic representations		Set-theoretic terms naming components of analogic representations
— *Entity*		— *Unity, element*
— *Etype*		— *Class / set*
— *(Entity) Property*		— *Tuple*
— *Property (type)*		— *Relation, that is set of tuples*
— *(Entity) Relation*	***Corresponds to***	— *Tuple*
— *Relation (type)*		— *Relation, that is set of tuples*
— *Word / (concept)*		— *Percept*
— *Language*		— *Domain (of interpretation)*
— *Assertion*		— *Fact*
— *Theory / (linguistic representation)*		— *Model / (analogic representation)*

Each terms on the left (e.g., entity) is the name of the element which plays the same role as the element whose name is in the same line, on the right (e.g., element).

Notation (terms used). In informal writing, whenever no confusion arises we will use terms on the right in place of terms on the left, to facilitate the interpretation. Moreover we will drop the terms in parenthesis.

Key notions

- World models
- Words, concepts
- Alphabet
- Assertions
- Languages and assertional languages
- Assertional theories
- Examples of world models: DBs, EER/ ER models, KGs, Entity graphs, etype graphs, Natural Language text, Concept Graphs (CGs), i.e., Digital lexicons

From models to theories

(HP2T)