**Sets, Individuals, Mappings: Model**

Contexts (sets):

(Relation, Relationship, Role, Dimension);

Occurrences (individuals):

(Context, Kind, Resource, Dimension);

Mappings (set / individual relationships):

(Context : Dimension, Occurrence : Measure, Attribute : Unit, Value : Value);

**Layered Representation: Augmentation / Inference**

Contexts:

Dimension : Mapping Context

Role, Dimension : Mapping Occurrence

Relationship, Role, Dimension : Mapping Attribute

Relation, Relationship, Role, Dimension : Mapping Value

Occurrences:

Dimension : Mapping Context

Resource, Dimension : Mapping Occurrence

Kind, Resource, Dimension : Mapping Attribute

Context, Kind, Resource, Dimension : Mapping Value

Mappings:

Value

Attribute, Value

Occurrence, Attribute, Value

Context, Occurrence, Attribute, Value

**Augmentation / Inference: Layers**

Aggregation: Clustering (Types).

Activation: Classification (Attributes in Type Context).

Alignment: Regression (Values in Attributes Context).

**Model Semantics:**

Data: Occurrences. (Tomato, price, 10); Aggregation (Clustering / map reduce?).

Information: Contexts (items price), Schema, Rows (various items price: tomato, price, 10; banana, price, 15, product relation: Activation).

Knowledge: Mappings, Behavior, Transforms (price over time variations). Alignments (Inference / Matching Attributes Values) (i.e.: relations, contexts: entails).

Population: Input Data statements with contexts. Contexts, Occurrences, Mappings inferences.

APIs: Augmentations, Query, Traversal, Matching, Transforms.

**Ontology Matching: Relations / Relationships**

Entity Relationship instance asserted as a reified concept with its type and attributes or as a series of triple statements which describes the given Entity Relationship instance via individual assertions. Bidirectional translation.

**Ontology Matching: Dimensional Alignments (Mappings):**

Mappings (set / individual relationships):

(Context : Dimension, Occurrence : Measure, Attribute : Unit, Value : Value);

Order. Comparison. Relations.

**Functional API: Monads / Transforms**

Resource / Layer?

Context / Occurrence / Mapping?

Mapping: Selector Monad. Matching Resource / Role set?

Context / Occurrence Monads Instances (Layers Hierarchies Monads): Aligned / Matching Entities resolution (Augmentations Agggregations / Activation / Alignments matchings). Versioned graph: stateless / functional. Mappings assertions matching.