**Objectives:**

Rosetta Stone like engine enabling Ontology (Data, Schema / Information, Knowledge / Behavior) discovery, matching and integration.

Reactive Service Bus for pluggable integration of application and translation of gestures between domains allowing workflows alignment and discovery of applications domains behavior.

**Models: Sets, Individuals, Mappings**

Sets, Groups, Categories: TBD.

Models:

Types (types in sets roles):

(Relation : Statement, Relationship : Kind, Role : SPO, Dimension : U);

Individuals (individuals / sets types instances):

(Context : Statement, Class : Kind, Resource : SPO, Dimension : U);

Mappings (type / individual relationships):

(Context : Dimension, Occurrence : Measure / SPO, Attribute : Unit / Kind, Value : Value / Statement);

**Layers: RDF Quads Representation. Augmentation / Inference Matrix (FCA)**

(Context, Occurrence, Attribute, Value) Pattern:

CSPOs: TypeID / InstanceID.

CSPOs: Objects / Attributes (FCA scaling).

Lattice traversal / Concepts (Contexts) Augmentations. Objects / Attributes mappings (CSPOs Context types TypeID / InstanceID).

Types Model:

(Dimension, Relation, Relationship, Role);

(Role, Dimension, Relation, Relationship);

(Relationship, Role, Dimension, Relation);

(Relation : Statement, Relationship : Kind, Role : SPO, Dimension : U);

Individuals Model:

(Dimension, Context, Class, Resource);

(Resource, Dimension, Context, Class);

(Class, Resource, Dimension, Context);

(Context : Statement, Class : Kind, Resource : SPO, Dimension : U);

Mappings Model:

Models metadata, properties and upper alignments / augmentations relationships Model data.

(Value, Context, Occurrence, Attribute);

(Attribute, Value, Context, Occurrence);

(Occurrence, Attribute, Value, Context);

(Context, Occurrence, Attribute, Value);

**Layers: Augmentations / Inference**

Aggregation: Clustering (Contexts types Occurrences).

Activation: Classification (Contexts types Occurrences Attributes).

Alignment: Regression (Contexts types Occurrences Attributes Values).

**Model Semantics:**

Data: Individuals. Mappings. Data Occurrences Aggregation.

Data: Individuals Model.

(Dimension, Context, Class, Resource);

Mappings (type / individual relationships):

(Context : Dimension, Occurrence : Measure / SPO, Attribute : Unit / Kind, Value : Value / Statement);

Information: Types. Mappings. Type Occurrences Attributes. Activation.

Information: Types Model. Schema.

(Dimension, Relation, Relationship, Role);

Mappings (type / individual relationships):

(Context : Dimension, Occurrence : Measure / SPO, Attribute : Unit / Kind, Value : Value / Statement);

Knowledge: Individuals / Types Mappings (Attributes) Values. Alignment.

Knowledge: Behaviors.

Mappings (type / individual relationships):

(Context : Dimension, Occurrence : Measure / SPO, Attribute : Unit / Kind, Value : Value / Statement);

**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. Upper Ontology assertions. Augmentations. TBD.

**Functional API: Monads / Transforms**

Resource / Layer?

Context / Occurrence / Mapping?

Mapping: Selector Monad. Matching Resource / Role set?

Context / Occurrence Monads wrapping Layers Hierarchy Contexts.

Entity Alignment / Matching resolution via Functional Augmentations: Agggregations / Activation / Alignments (upper / dimensional matchings). Versioned graph: stateless / functional. Mappings assertions matching.

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

**TODO Items:**

* Explain Dimensional / Sets layout. Diagram. Models examples.
* Explain Layer Context, Occurrence, Attribute, Value Pattern: for SPO Statement and for Models Statements.
* Explain Layers Aggregation  Activation / Alignment:
* Aggregation: Context type instance aggregates type instance child Occurrences (parent Context type instances) matching grouping criteria (Encoding).
* Activation: Which Attributes has Context Occurrence (according to its Kind in Context / Role) in this Occurrence.
* Alignment: Context Occurrence Attributes Values (according to its Kind in Context / Role).
* Functional Encoding (Sets): Types Dimension, Role, Relationship, Relation Kinds applied over Individuals Dimension, Resource, Class, Context. Augmented IDs. Encode hierarchies / reifications. Example: Subject / Object / Predicate Role / Relationship mask, Subject / Object / Predicate Resource / Class mask. Masks: Predicates of Set memberships. Functional Mappings.
* Mappings Model: Sets / Dimensional Semantics.
* Mappings / Functional Encoding: Upper Dimensional Matchings / Augmentations. Mappings Model masks matchings reflects / leads to Types / Individuals Models Augmentations / Assertions.
* Mappings / Functional Encoding: Relation Statements / Relationship views / matchings examples.
* Mappings Upper Alignments examples (dates, marital status, hiring). Relation Relationship statements order / context properties (Dimensional Alignments).
* Functional APIs Query / Browse / Traversal / Transforms examples.
* ImplementationItems & drafts documents topics.