Kinds: example aggregate CSPO Kinds from an statement.

Models: APIs. (Message I/O)

flows messages
levels: Reify the lower - ctxs. *

levels: Reify Context IDs.

Metamodel (IDs → Context): APIs.
L mappings

Facets (Context → Context) data schema behaviour. cls/inst. views. APIs.
L mappings

Functional (Mappings)
• Augmentations

Semantic Aggregation

Alignment

Dimensional Activation
(Meta Resources Augment.)

(reactive secondary Augment. routes)

Context<ID>: Models/Facets Quads CSPOs.
(Producer/Observer: MOQs) template Person/Emp. transform Address/Position

Mapping<Context<ID>, Context<ID>>:
L Context<ID>: *: Metamodel reified levels populated / resolved.
(augment.)

Domain models
*
Annotation messages

dataflow
Augmentation: Routes templates / transforms
Flows: Apps, types, layouts.

Message: Input / transformed Quads. (instances)
(Augment. / mappings) L Align (raw) - Augment. routes.
(ontology matching) resolve Contexts] Msg<Ctx>

* [Query: SPO → C
Assert: val/ctx/S/P.
L mapping L template

Context: Models, Metamodel, Facets, Levels.

Encoding.

Event Bus

Signatures

Routes / Dataflow.

* Flows: Apps type layouts

Augmentation: Messages (domain driven)

Backend (upper / levels ontology matching)

• APIs: Services / Protocol Layers (services, etc)
(Node, Peer, Client, Connector, etc.) dialogs, etc.
REST HATEOAS / Reactive / Event driven
Addressing: URI Abstraction.

0. Outline

1: Mission / Vision

1.1: Problem

1.2: Current landscape

1.3: Solution Description

1.3.1.: Use Cases (domain translation)

2. Approach

2.1: Augmentation dataflow K.B. (distrib.)

2.2.: Ontology Matching / Routes: Map <foo> mappings?

2.3.: Reactive / Event Driven

($\underbrace{\text{Dialog Protocol}}_{\text{Proto. APIs.}}$. $\underbrace{\text{REST HATEOAS APIs}}_{\text{Svc. APIs.}}$)

3. RDF Introduction

4. RDF Quads / Object Mapping / DOM.

5. Models

5.1: Context Quads Layers structure.

5.2: Meta Resources (MM Contexts hier.)

5.3: Meta Model (IDs → Context Mappings)

5.3.1: Facets (Context → Context Mappings)

5.3.1.1: Functional

5.3.1.2: Semantic

5.3.1.3: Dimensional

Facets Resources:
Facets Contexts hier.
Data/Schema/Behavior
class/instance views:
APIs
*: Levels Flows Msgs.
Layouts.

GLORIA

5.3.2: Levels: Reify Ctx. Hiers * Flows Msgs.
Layouts
populate/resolve

6: Context Reactive Abstraction

7: Encoding

8: Signatures (Context Kind domain/range)
Kinds: streams producer/consumer.

9: Routes/Dataflow: Signatures pub/sub bindings
I/O (because possible trigger (route) Mappings. by Query Messages (Msg Msgs)
11.2.3: Mappings I/O events. matches I/O signatures. P + C

10: Event Bus
(I/O routes: dispatch Mappings + Response)
* Apps (auto) domain driven Msgs. Flow Layout

11. Augmentation / Messages (query/assert Mappings Msg → Msg.)
↳ Mappings

11.1: Metamodel/Facets ~~Augmentations~~ Mappings ents.: (Route rates Mappings domain Msgs)
(Aggregation, Alignment, Activation)
Query/Assert Model/Facets Mappings.

11.2: Reactive ~~Message~~ Events ~~Augmentations~~ Mappings.
Query / Assert Domain Mappings.
triggers routes dataflow (Message → Message)
└ Mappings.
* Alternatively stated by models ref.

→ Mapping I/O (11.2.3) + events.
ctx
└→ Query → Message$_P$ (Query) ⟨Msg⟨ctx⟩$_S$, Msg⟨ctx⟩$_O$⟩
└→ Assert → Augmentation$_P$ (Assert) ⟨Msg⟨ctx⟩$_S$, Ctx⟨id⟩$_O$⟩

Possible cells.