API4KB

API4KB Working Group

Meeting Update

D.Sottara, PhD
OMG Technical Meeting – Spring 2013, Reston, VA

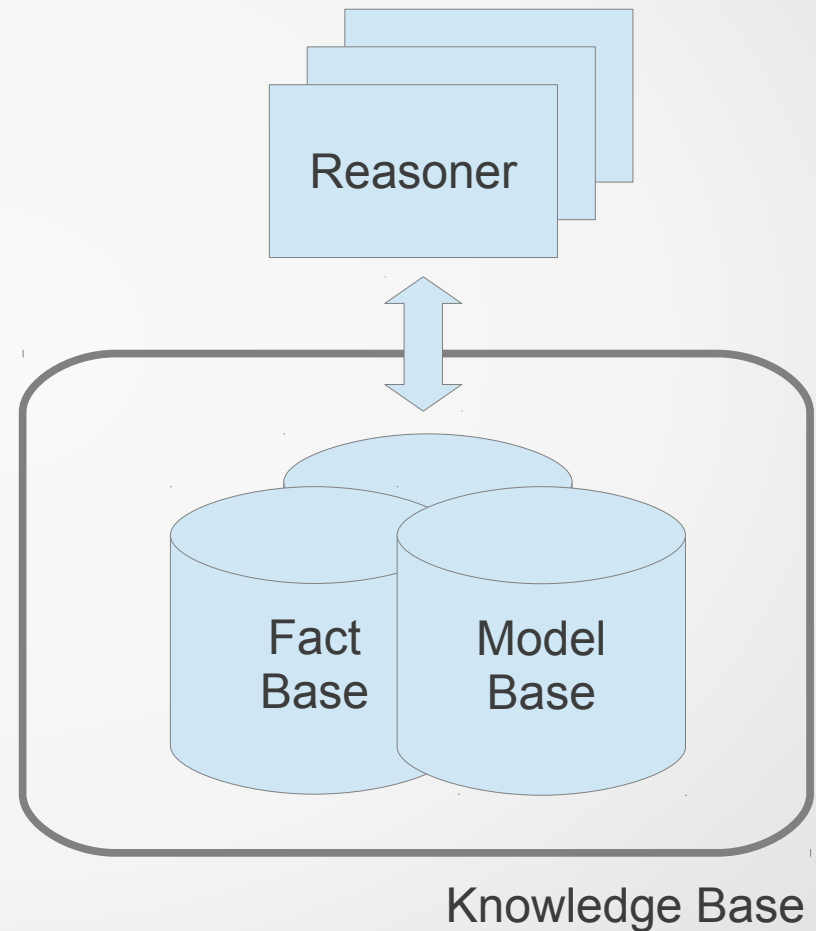# Requirements - Summary

- Problem

    Provide *uniform access* to Knowledge Bases

    - Simple, static KBs

    - Working Memories with Hybrid reasoners

    - Multiple KBs with loosely coupled, heterogeneous reasoners

    - ....

API4KB

- Composite KBs :

- Ontologies
(T-box + A-box)

- Rulebases
(Rules + Facts)

- Predictive Models
(Models + Datasets)

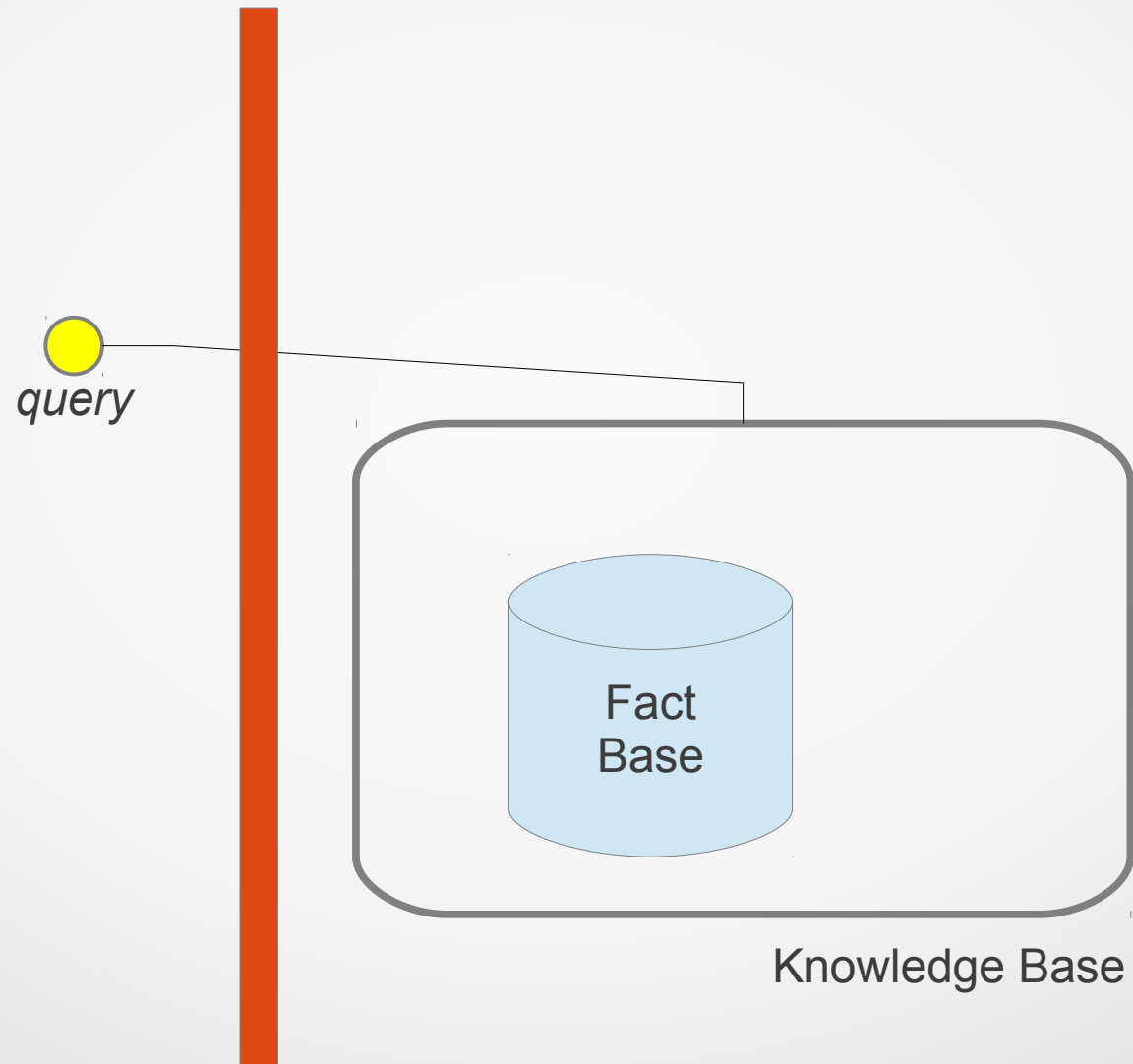- Business Processes
(Processes + Instances)

- ...

Reasoner

Fact
Base

Model
Base

Knowledge Base

Requirements - Summary

- Multiple Bindings
  - (Java | C++ | …) Library, REST, SOAP, …

- Distributed, open environment
  - Prefer stateless, idempotent, concurrent calls
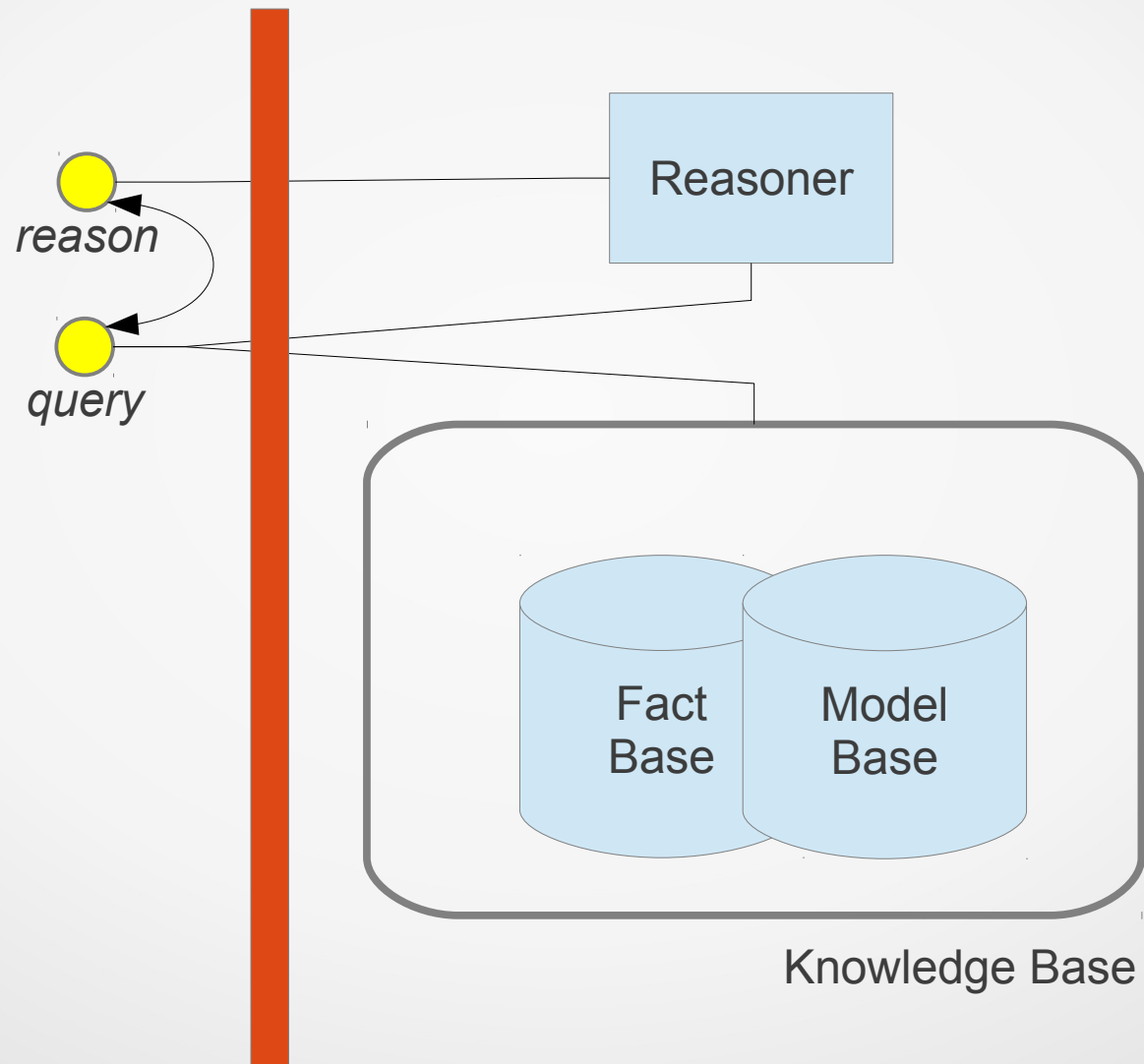
- Post-coordination
  - Simple APIs, highly configurable

API Categories

- Parsing & Translation

- KB Construction

- Query

- Reasoning

- Meta

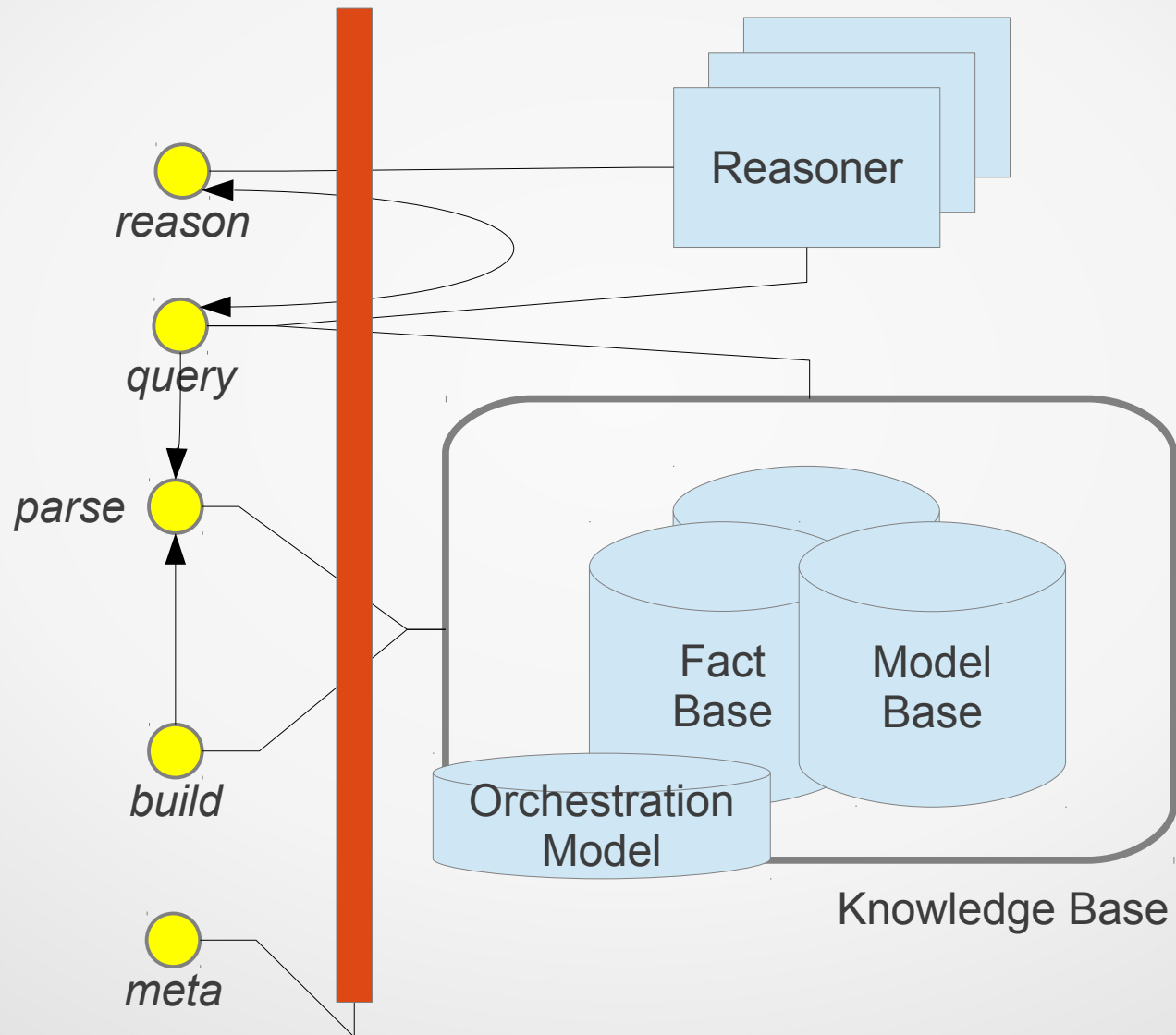# API4KB

query

Fact
Base

Knowledge Base

API4KB

API4KB

Metamodel

- "Knowledge and Reasoning Systems Ontology"

    - Model of Knowledge and Reasoning
      for knowledege-based systems

    – Configuration

    – Operation Specification

    – Explanation

Metamodel Basics

- `KnowledgeBaseAssetDescriptor`

    – Placeholder for KB component model

  `- ID`
    e.g. URI

  `- Location`
    e.g. URL

  `- Content`
    e.g. OWL, Ruleset

  `- Syntax / Serialization`
    e.g. Manchester Syntax, RuleML

Metamodel Basics

- KnowledgeOutcomeDescriptor

  - Placeholder for Reasoning Process outcome descriptor

  - result
    e.g. code

  - Explanation

    - Descriptor

    - Proof (PML?)

Parsing API

- `Parse`
  ```
  in   URL asset,
  in   KBADescr descriptor,
  out  Expl result
  ```
  - Verifies whether `asset` is compatible with `descriptor`

- `Translate`
  ```
  in   URL asset,
  in   KBADescr source,
  in   KBADescr target,
  in   URL loc,
  out  Expl result
  ```

  - Parses the `asset` described by `source` and creates a new representation, compatible with `target`, deploying it at `loc`

KB Construction API

- `CreateKB`
  ```
  in  URI kbID,
  in  URL kbLoc,
  out URL effectiveLoc
  ```
  - Tries to ensure that a KB with ID `kbID` exists at `kbLoc`, creating it if necessary

- `SetAsset`
  ```
  in  URI kbID,
  in  URI assetID,
  in  URL assetLocation,
  in  KBADescr descr,
  out Expl result
  ```
  - Tries to ensure that an asset is part of a Knowledge Base

KB Construction API

- `ReadAsset`
  ```
  in  URI kbID,
  in  URI assetID,
  out KBADescr descr
  out KBAsset asset
  ```

  - Returns the asset with `assetID` contained in a KB

- `ExportAsset`
  ```
  in  URI kbID,
  in  URI assetID,
  in  KBADescr descr,
  in  URL targetLocation,
  out Expl result
  ```

  - Exports `assetID` from a KB to `targetLocation`, in a format compatible with `descr`

KB Construction API

- `DeleteAsset`
  ```
  in  URI kbID,
  in  URI assetID,
  out Expl result
  ```

  - Removes the asset with `assetID` contained in a KB

- `UpdateAsset`
  ```
  in  URI kbID,
  in  URI assetID,
  in  KBADescr descr,
  in  URL sourceLocation,
  out Expl result
  ```

  - Updates `assetID` with a new version at `sourceLocation`

Reasoning API

- `CheckConsistency`
  `    in  URI kbID,`
  `    out Expl result`

  - Checks the consistency and the integrity of a knowledge base, returning an explanation with the results

- `Classify`
  `    in  URI kbID,`
  `    in  boolean materialize,`
  `    out Expl result`

  - Computes all possible *type* inferences for individuals in the knowledge base

- `Infer`
  `    in  URI kbID,`
  `    in  boolean materialize`
  `    out Expl result`

  - Computes all possible entailments

Query API

- `Inform`
  ```
  in  URI kbID,
  in  ANY content,
  in  KBADescr descr,
  in  boolean preserveConsistency,
  out Expl result
  ```

  - "Tells" something to the KB, adding the `content` in format `descr`, possibly trying to preserve consistency

Query API

- `QueryIf`
  ```
  in  URI kbID,
  in  ANY content,
  in  KBADescr descr,
  out Expl result
  ```

  – Checks whether `content` is part of the KB or not

- `Query`
  ```
  in  URI kbID,
  in  ANY query,
  in  KBADescr descr,
  in  KBADescr outDescr,
  out ANY answer,
  out Expl result
  ```

  – Queries the KB, submitting a `query` in format `descr`,
    returning `answer` in the format defined by `outDescr`

Query API

- Request
  ```
  in  URI kbID,
  in  ANY query,
  in  KBADescr descr,
  in  KBADescr outDescr,
  in  boolean preserveConsistency,
  in  boolean materialize,
  out ANY answer,
  out Expl result
  ```

  - Combination of `Inform + Infer + Query`.
    The result will be added to the KB only if explicitly stated

Meta API

- `reportMetrics`
  - Returns a descriptive statistics of the content of the knowledge base
- `reportKBCapabilities`
  - Describes the capabilities of the KB, in terms of the metamodel
- `reportReasonerCapabilities`
  - Describes the capabilities of available reasoners, in terms of the metamodel