network inference

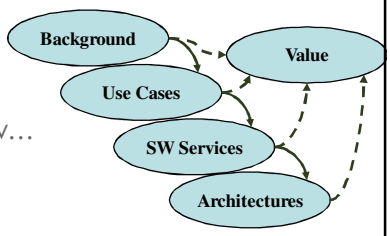
TTI Vanguard
February 2005 – All that Data

Enterprise Semantic Web:
The Changing Face of Corporate Data

Jeffrey Pollock
March 2005

Prelude

- Presentation Goals
  - Straight Talk; **No Hype**!
  - **Education** – what, why, how…
  - Explain **Value-drivers** & ROI



- Presentation Takeaways
  - If you're serious about data, you need to be serious about '**semantic technology**'
  - The core enabling technologies and services are available to you now – **RDF, OWL & Core Services**
  - Semantic Web technology is not perfect, it may never be, but it is showing real value for enabling an **Enterprise Semantic Web**

network inference

Core Tenets

- Information is power
  - As always, for nations and businesses
- A change is required
  - Collection rate outstrips understanding
- The change must be fundamental
  - OO, Java, RDB, XML, EAI (etc) aren't enough
- Perfect is the enemy of the good
  - W3C standards aren't perfect, but ARE powerful

network inference

3

Information Overload

Q: In General, why are semantics important?

- Maintenance and Support of "Data Meaning" is the largest, most expensive, and time-consuming portion of IT budgets
  - Mars Climate Observer – Pounds v. Metric force per second
  - Master Data – persistent and consistent interpretation of data
  - Product Hierarchies – Is my PDA a phone or a camera or both?
- Over $200B is spent each year on integration issues involving the meaning of data
  - Classic "n-squared" problem is more severe at the data layer
  - Data-to-data interfaces outnumber "pipes"
  - Tightly-coupled = Brittle = expensive to change
- Information growth is accelerating – FAST!
  - 2002-2005 – more new data than all of history
  - 5 exabytes of new digital data created in 2002 – enough for .5 million new Libraries of Congress

network inference

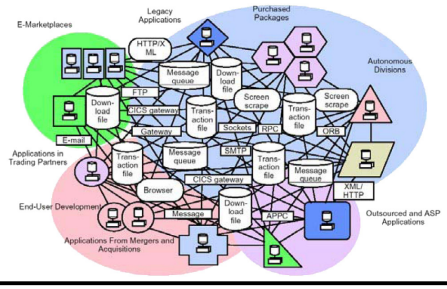
4

Infrastructure Complexity

Q: For business, why are semantics important?

EAI? <XML>? Warehouse? BAM? EII? ETL?







network inference

5

Graphs and Ontology

Q: How can the Semantic Web help?

- "Small Pieces Loosely Joined"
  - Small Pieces: RDF Triples – data graphs
  - Loosely Joined: OWL Models – concept collections **defined with precision**
- Fundamentally Different Than…
  - XML – taxonomic tree structures
  - Java – object structures (frames)
  - RDB – two-dimensional/tabular structures
- From "Trees to Piles of Leafs"
  - RDF Triples, collections of atomic data
  - Ontologies are "views" into the whole
  - "Strict" trees (XML) are not required
    - Ontologies are multi-dimensional, with formal, logically-sound, data semantics

Some Definitions

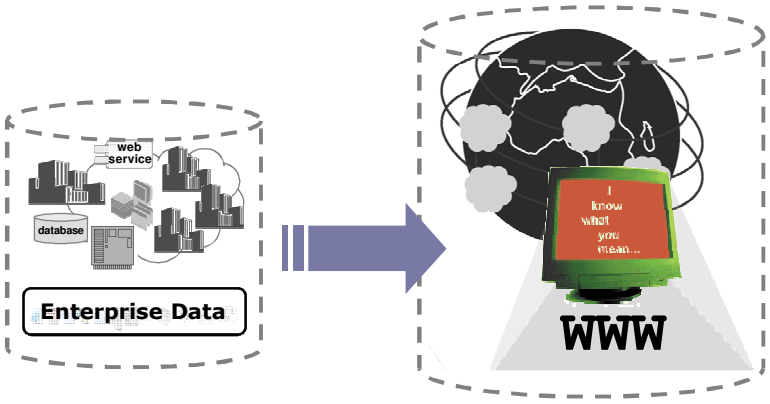
Semantics
Taxonomy
Schema
Ontology
Inference

network inference

6

It's Business, not Social

Q: What is the Enterprise Semantic Web?



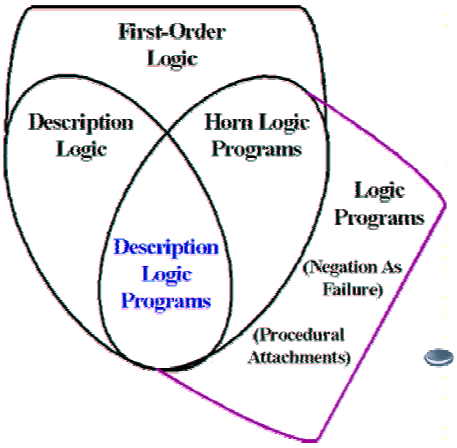
TODAY ***SOMEDAY***

network inference

7

Math Matters

Q: Why are rules and logics crucial?



- What if your relational database gave unreliable query responses?
  - Can pigs fly?
  - How many units did you sell last quarter?
  - Is this annual audit report reliable?
- If the data or rule is there, the query ought to know of it…

network inference

8

Two Sides of a Coin

Q: Why do scruffy/neat people matter?

- Artificial Intelligence (AI) factions and "analysis levels"
  - "**Neats**" – theoreticians, symbologists, proving truth
  - "**Scruffies**" – engineers, connectionists, getting a job done
- Theory and practice meeting in the middle…
  - Modern RDBMS
  - Nuclear Power
  - Stealth Technology
  - …more great things





Web Ontology Language

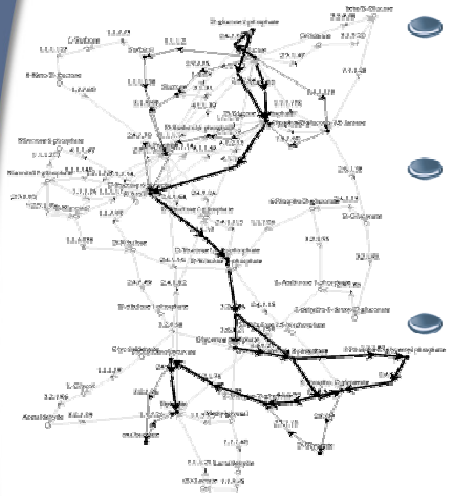
Resource Description Framework

network inference

9

"Web-Effect" + Standards

Q: Isn't this just AI all over again?



- Ssshhhhhh!
  - OWL/RDF are a type of AI "Knowledge Representation"
- Traditional AI
  - "Silos" of knowledge
  - Incompatible algorithms
  - Inconsistent "expertise"
- Semantic Web
  - Everything has a URI
    - Models, classes, data
  - Predictable semantics
    - Std. Triples & Axioms

network inference

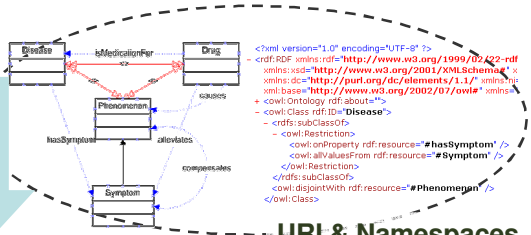
10

Top-Down/Bottom-Up

Q: How do you operationalize this stuff?



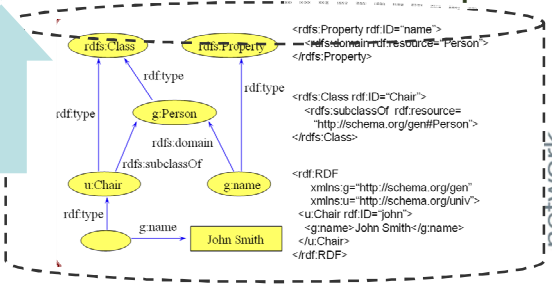
Model Ontologies



URI & Namespaces



Create RDF Triples



network inference

Use Cases?

- Well…*actually*; "Capability Cases"
- Management, Query, Enforcement
  - Manage information more efficiently
  - Query information more effectively
  - Enforce policies/rules/security with agility
- Key Concepts:
  - Querying via ontology beats SQL or Google
    - …for certain uses
  - Managing metadata w/ontology is agile
    - …inference does the hard work for you
  - Enforcing policies w/ontology is powerful
    - …model-driven and highly dynamic

network inference

12

Use Cases

- Concept-based Query
- Smart, Inference Query
- Information Mediation
- Taxonomy/Rule Alignment
- Multi-level Security/Data Views
- Secure Audit/Compliance
- Process Orchestration

network inference

13

"Semantic" Service Utilities?

- Enable enterprise use cases/capabilities
- A rose by any other name…
  - Utility Computing, Service Grid; Grid Computing
  - Network-Centric Enterprise Services
  - Enterprise Nervous System
  - …?
- Key Concepts:
  - Service-based (SOA/Web Services)
  - Transient utilities (addressable remote functions)
  - Autonomic (capable of automated interaction)
  - Loose-Coupling (protocols AND data)
  - Adaptive data interfaces (graph-based ontology)

network inference

14

SW Service Utilities

- Metadata Repository
  - Taxonomy Alignment
  - Policy and Rule Management
- Mediation Engine
  - Query Brokering
  - Federated View Management (Security)
- Registry Service
- Process Engine
- Unstructured Text Service
  - Extraction (Entities, Verb Phrases, Expressions, etc)
  - Annotation (Link extracted clusters to ontology)

15

network inference

Architecture Patterns?

- Build your own utilities
  - (or buy COTS tools that use these patterns!)
- How many ways to "skin a cat?"
  - Proprietary (algorithms and data formats)
  - RDF (data and some inference)
  - OWL (powerful, correct inference)
  - Hybrid (right tools for the job)
- Key Concepts:
  - New "kinds" of architecture layers and tradeoffs
  - Mapping and wrapping (your legacy systems)
  - Separate data from ontology (different uses)
  - Understand your challenges and needs!

16

network inference

Semantic Architectures

PHYSICAL

- RDF Database
- RDF Façade
- OWL Knowledge Base
- OWL Façade
- OWL Instance Store
- Hybrid Fidelity Reasoning

LOGICAL

- Layered Logical Architectures

network inference

17

Industries in Need

- Pain Killers?
  Oftentimes, there is a short-term need that predicates a fast, "stop the bleeding," type approach. Usually custom, one-off solutions are chosen because they are known quantities – at the cost of significant long-term issues like TCO and inflexibility.
- Penicillin?
  Certain industries are experiencing decades long systemic pain that software "pain-killers" can no longer help with. These industries are in need of solutions that solve the root of the problem – not the surface "bleeding."
  - Defense Industry
    - Network-Centric Enterprise Services (NCES)
    - Department of Homeland Security (DHS)
  - Manufacturing
    - Automotive Supply Chain
    - Electronics & CPG Supply Chain
  - Life Sciences
    - Process Automation / Risk Analysis

network inference

18

ROI Value Drivers

- New Information Capabilities
  - **What**: Doing something that you haven't done before to provide the business new value.
    Such As:
    - Linking annual risk strategies to live data inspection
    - Assembling SOA service integrations faster
    - Testing realtime "what-if" financial scenarios
    - Securing XML SOA payload with no code
- Total Cost of Ownership (reduction)
  - **What**: Doing what you are already doing, but doing it more efficiently at less operational cost.
  - **Such As**:
    - More productive integration maintenance staff
    - More productive naming/metadata staff

network inference

19

Some Figures

- Caveat Emptor…
  - *"There's lies, damn lies, and ROI predictions"*
- Proven ROI Potential*
  - 20-90% reduction in labor hours, cycle time, inventory levels, operating cost, development time, and development cost
  - 2-50x improvements in quality, service levels and productivity
  - 20-80% reduction in total cost of ownership
  - 3-12 month positive return on investment
  - 30-300x positive ROI over 3 years
  - Mitigated risk, reduced vulnerability to fraud, liability or litigation

network inference

20 **TopQuadrant market research leveraging Gartner, IDC, Forrester and Giga*

Uptake Momentum

- Commercial
  - Audi, General Motors, Cisco Systems, Fed-Ex, France Telecom, Nokia, Fujitsu, Hitachi, NEC, Unisys, Pfizer, General Electric…
- Federal
  - Federal CIO, SICoP, Navy, Army, DLA, EPA, NCI, DHS, CBP, NGA, MoD, NATO + other foreign federal
- Vendors/Systems Integrators
  - Oracle, IBM, Microsoft, Adobe, Hewlett Packard, WebMethods, CGEY, BearingPoint, SAIC, Lockheed Martin, SRI, MITRE, Booz Allen Hamilton
- Standards Bodies
  - W3C, OMG (MDA), OASIS (DCML, ebXML), ISO

network inference

No Free Lunch

THE BAD NEWS:

TRAIN YOUR PEOPLE!

THE GOOD NEWS:

DBA's may already know most of this stuff…
Object-Oriented analysts have a BIG head-start…

Tools are available:

Modeling – GUI driven, Visio-based, Rational-based, etc.
Inference – Java based, C++, App Server based, etc.
Rules – GUI driven, J2EE integration, .NET components, etc.
Database – RDF databases, open source, triples stores

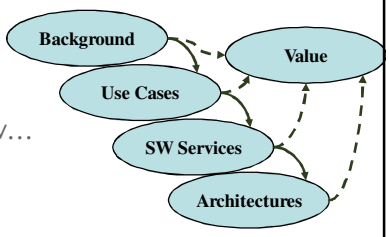
Trainers are available:

Software Vendors (Network Inference, @Semantics, etc.)
TopQudrant – Dedicated RDF and OWL training

network inference

Postscript

- Presentation Goals
  - Straight Talk; **No Hype**!
  - **Education** – what, why, how…
  - Explain **Value-drivers** & ROI



- Presentation Takeaways
  - If you're serious about data, you need to be serious about '**semantic technology**'
  - The core enabling technologies and services are available to you now – **RDF, OWL & Core Services**
  - Semantic Web technology is not perfect, it may never be, but it is showing real value within the enterprise – an **Enterprise Semantic Web**

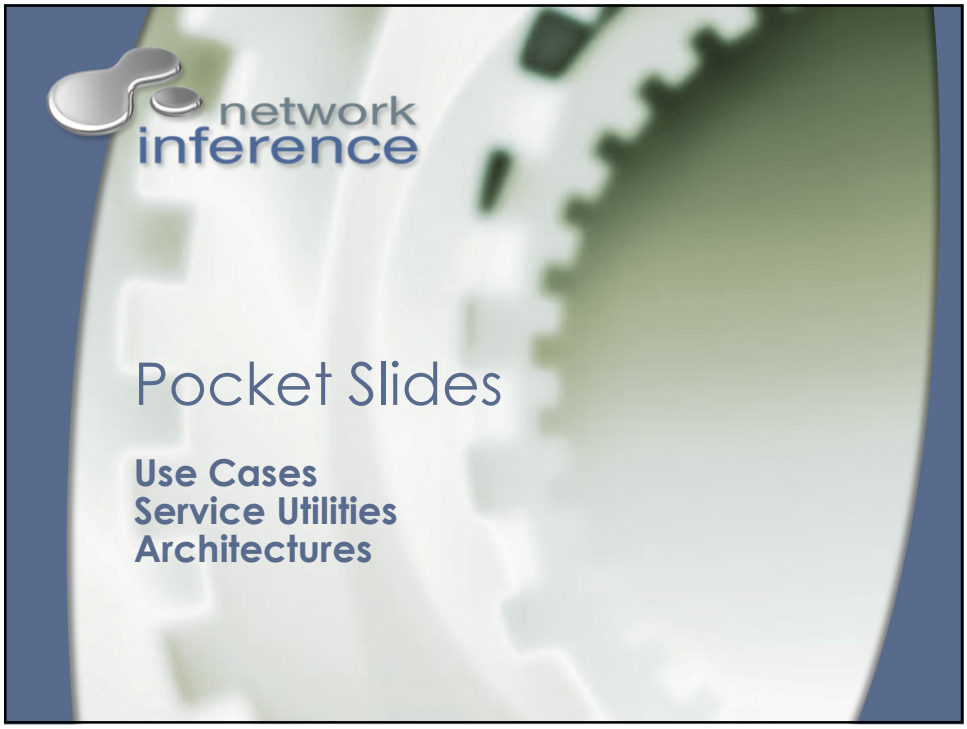
network inference

23

network inference

Questions?

Jeff Pollock
Vice President, Technology
jeff.pollock@networkinference.com


network
inference
Pocket Slides
Use Cases
Service Utilities
Architectures


network
inference
Enterprise Use Cases
Concept-based Query
Smart Query
Information Mediation
Taxonomy/Rule Alignment
Information View Management
Smart Audit and Compliance
Process Orchestration

Use Cases?

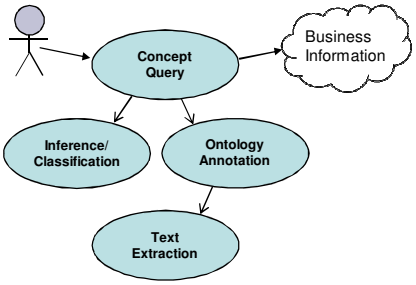
- Well…*actually*; "Capability Cases"
- Management, Query, Enforcement
  - Manage information more efficiently
  - Query information more effectively
  - Enforce policies/rules/security with agility
- Key Concepts:
  - Querying via ontology beats SQL or Google
    - …for certain uses
  - Managing metadata w/ontology is agile
    - …inference does the hard work for you
  - Enforcing policies w/ontology is powerful
    - …model-driven and highly dynamic

network inference

27

Concept-based Query

Use Case	Concept-based Query
Actors	Human Agent, Software Agent, Unstructured Information, Semi-Structured Information, Structured Information
Description	The ability to issue queries about concepts, rather than string matching.
Business Value	Finding the right information faster.
Motivation	Traditional, "bottom-up" approaches for search rely upon symbol matching algorithms, and in advanced cases, latent semantic indexing techniques. These approaches cannot, by themselves, solve problems that require ontology or thesaurus to link words to synonyms and/or the business concepts they belong to.
Why OWL/RDF?	RDF – schemaless data RDF – federated/webbed data OWL – correct, predictable reasoning OWL – "real" ontology (v. XML/RDB) OWL – vendor independence
Key Uses	Search and Retrieval, Guided Navigation, Portals, Federated Query, Data Integration, Business Intelligence



Q: Vehicles made in UK?

A:

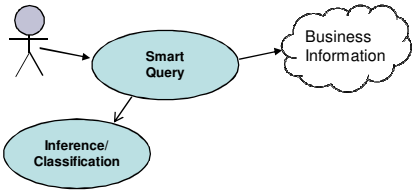
Triumph	Motorcycle	Great Britain
Austin Healy	Coupe	Warwick
Centurion	Tank	United Kingdom
Greeves	Cycle	England
Bedford	Truck	London

network inference

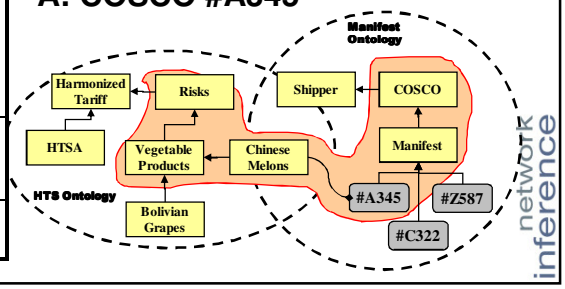
28

Smart Query

Use Case	Smart Query
Actors	Human Agent, Software Agent, Unstructured Information, Semi-Structured Information, Structured Information
Description	The ability to derive implicit knowledge from stated models in a consistent and predictable way.
Business Value	Finding "hidden" information and rules. Doing so across federated schemas w/minimal code.
Motivation	Traditional search approaches can only join disparate information using lexical equivalencies. The most sophisticated approaches (Google) use proprietary lexical pattern matchers to determine a subset of possible associations. RDB technology relies on manually implemented joins in the schema and in functions/procedures.
Why OWL/RDF?	RDF – atomic, portable instances OWL – correct, predictable reasoning OWL – solid theoretic underpinnings OWL – vendor "plugability"
Key Uses	Search and Retrieval, Federated Query, Data Integration, Business Intelligence, Risk Management



Q: What shipments are at risk?
A: COSCO #A345

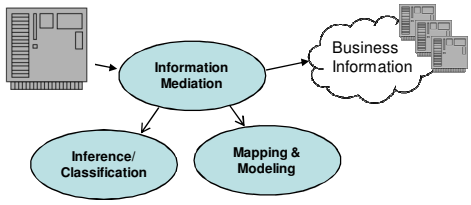


network inference

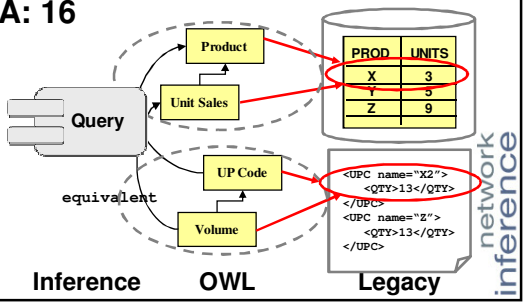
29

Information Mediation

Use Case	Information Mediation
Actors	Software Agent, Unstructured Information, Semi-Structured Information, Structured Information
Description	The ability to mediate information from different sources, formats, and contexts into relevant views.
Business Value	Getting what information you need in the right format for consumption by legacy systems.
Motivation	Traditional EAI, EII, B2B approaches rely on outdated data formats (XML, RDB, OO), which result in brittle, high-effort master reference data. Older formats are merely strucure, not ontology. Additionally, older metadata formats are unsound, incomplete and require ad hoc customizations that don't age well over a sustainment lifecycle.
Why OWL/RDF?	RDF – schemaless instance cache RDF/OWL – easily federated OWL – powerful domain modeling OWL – vendor "plugability"
Key Uses	Integration, Metadata Management, Portals, Federated Query, Data Warehousing, Business Intelligence



Q: How many X product sold?
A: 16

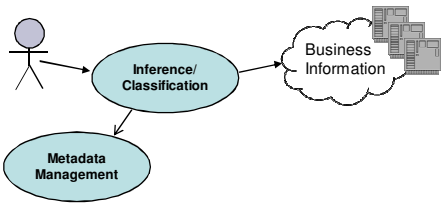


network inference

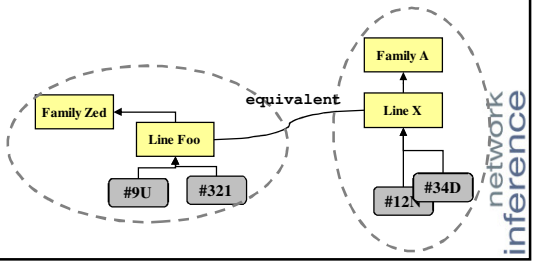
30

Taxonomy/Rule Alignment

Use Case	Taxonomy/Name/Rule Alignment
Actors	Data architect, Metadata analyst, Software Agent, Legacy data schemas
Description	Knowing what you know. Building master reference data for IT alignment across programs, organizations and software product families.
Business Value	Reduced taxonomy and business rules maintenance for complex IT environments.
Motivation	Approaches that do not use ontology – XML repositories, Thesaurus approaches, and RDB data dictionaries – are extraordinarily difficult to monitor, evolve, and perform "what-if" scenarios with. Even small changes require schema rebuilding, query reformulation, and application re-coding.
Why OWL/RDF?	RDF – schemaless repository RDF – RDB style transactions OWL – powerful domain modeling OWL – "evolvable" taxonomy OWL – classification business rules
Key Uses	ERP term alignment, Financial Reporting, Regulatory Compliance, Identity Management, Portals, Business Intelligence & Analytic Reporting



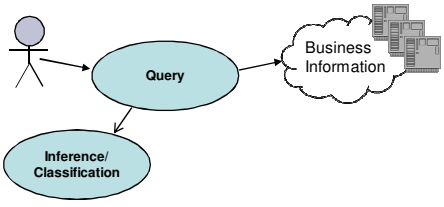
Q: What products are type X?
A: #9U, #321, #12N, #34D



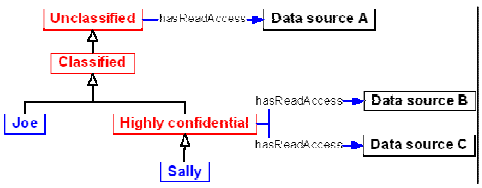
network inference

Multi-level Security/Views

Use Case	Information View Management
Actors	Data architect, Metadata analyst, Security controller, Software Agent, Legacy data schemas
Description	Use a distinct model-based policy layer for identifying profiles and access privileges to federated data.
Business Value	Reduced reliance on hard-coded security permissions layers; enabling more agile, yet fully secure, information dissemination – especially for federated data.
Motivation	Authorization techniques – used to create secure data views – typically require a large degree of coding and integration because of the need to secure data at the smallest atomic level. In homogenous environments this is usually not a problem, but the challenge becomes exponentially difficult in heterogeneous systems.
Why OWL/RDF?	OWL – multidimensional permissions OWL – "evolvable" security OWL – closed loop guarantee
Key Uses	Portals, Business Intelligence & Analytic Reporting, Service-Oriented Architecture fine-grained data security



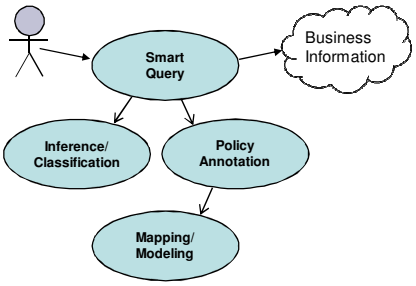
Q: Who can see comp data?
A: Sally



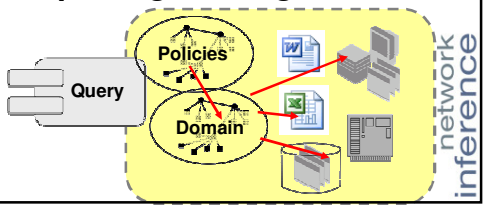
network inference

Secure Audit/Compliance

Use Case	Audit and Compliance
Actors	Regulatory controller, Auditor, Data architect, Metadata analyst, Software Agent, Legacy data schemas
Description	Implement mathematically correct and provable compliance policies (HIPPA, SOX, etc.) on legacy data.
Business Value	More reliable and consistent audit procedures.
Motivation	Although most modern compliance regulations are primarily procedural in nature, most have a distinct IT information audit requirement. Older integration technologies cannot enforce business policy in a global manner because integrations are inherently fragmented.
Why OWL/RDF?	OWL – strict semantics for policies OWL – federated, yet global OWL – accurate, repeatable inference OWL – policies linked to strategies
Key Uses	Financial Reporting, Regulatory Compliance, Identity Management, Portals, Business Intelligence & Analytic Reporting, Sarbanes Oxley, HIPPA



Q: Where are my SOX policies?
A: Reporting ontologies

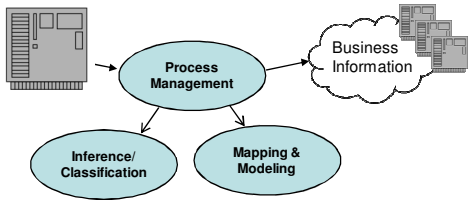


network inference

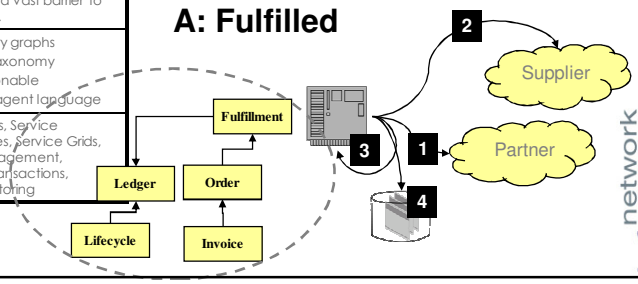
33

Process Orchestration

Use Case	Process Orchestration
Actors	Integration architect, Business Process Analyst, Software Agent, Web Services
Description	Allow software and services to coordinate long-lived and complex business transactions by using the semantics of an interface description.
Business Value	Less integration overhead for complex system-to-system interactions.
Motivation	Integration with classic EAI, EII and Web Services still require hard-coded scripts to manage multi-step transactions. For environments, particularly B2B, where services may be transient, this creates a vast barrier to efficient connectivity.
Why OWL/RDF?	RDF – federated query graphs OWL – "evolvable" taxonomy OWL – machine actionable OWL – purpose built agent language
Key Uses	Semantic We Services, Service Oriented Architectures, Service Grids, Business Process Management, Business to Business transactions, Business Activity Monitoring

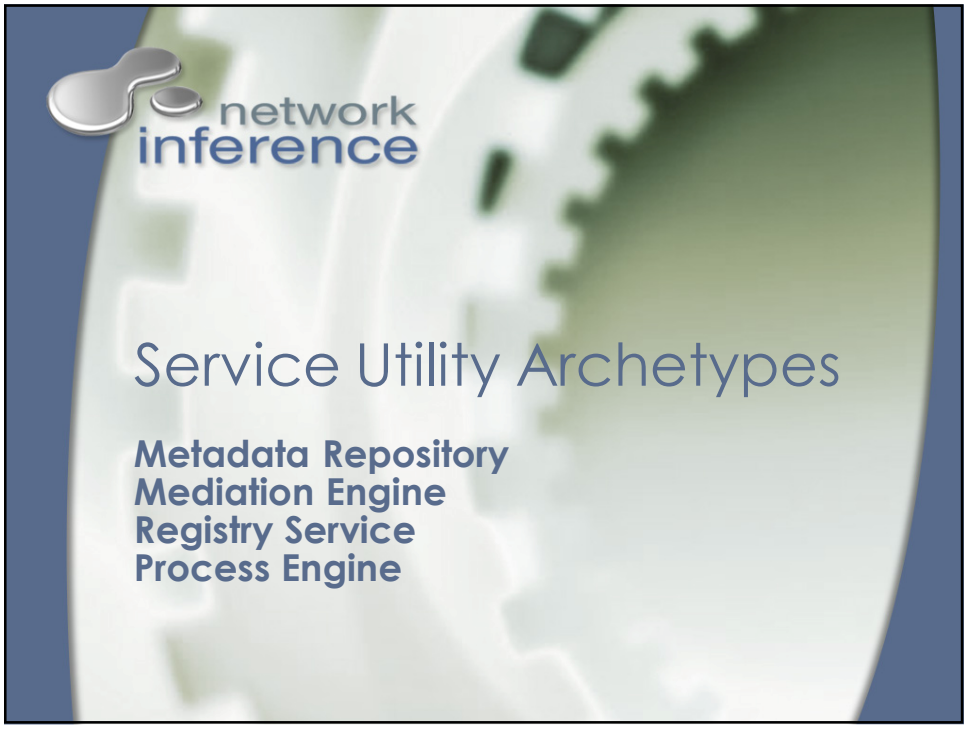


Q: Resolve invoice status?
A: Fulfilled



network inference

34



"Semantic" Service Utilities?

- Enable enterprise use cases/capabilities
- A rose by any other name…
  - Utility Computing, Service Grid; Grid Computing
  - Network-Centric Enterprise Services
  - Enterprise Nervous System
  - …?
- Key Concepts:
  - Service-based (SOA/Web Services)
  - Transient utilities (addressable remote functions)
  - Autonomic (capable of automated interaction)
  - Loose-Coupling (protocols AND data)
  - Adaptive data interfaces (graph-based ontology)

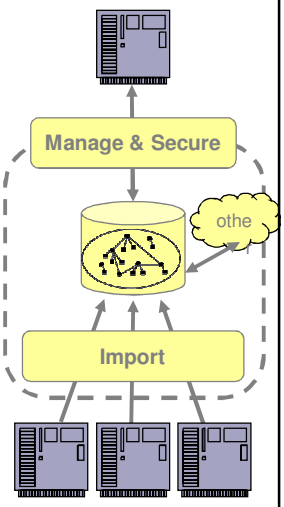
network inference

36

Metadata Repository

Metadata repositories are transaction safe, workflow-capable locations for storing enterprise vocabularies, taxonomy, names, content blobs in a highly interconnected manner.

- Metadata repositories are services
  - They may be used in applications or middleware
- Metadata repositories are "master references"
  - Master reference data is metadata, rules, and operational data that can be referenced from all enterprise systems reliably – knowing that they contain up-to-date and accurate data from other sources.
- Metadata repositories are semantics aware
  - RDF triples and OWL axioms improve meaning
- Metadata repositories unify organizations
  - Data from different sources, times, and contexts can be centrally managed, linked to legacy systems, and provide a common local for rules and vocabularies.

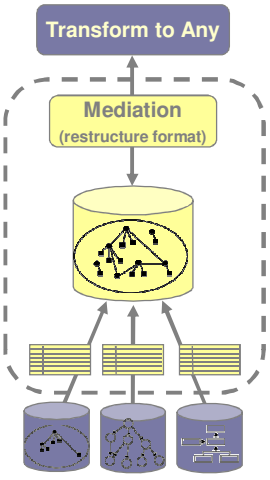


37

Mediation Engine

Mediation engines are tools that can dynamically output data in different formats and syntax that comply to disparate schema types. They enable automated data transformation.

- Mediation engines are services
  - They may be used in applications or middleware
- Mediation engines work with structured data
  - Unstructured and semi-structured data must first be bound to a schema prior to creating the mediation maps: text extraction and annotation
- Mediation engines let business agree to disagree about schemas and data
  - Use your own XML, relational or OO schemas
- Mediation engines enable interoperability of data from many format types
  - Automated transformation, any-to-any, is possible

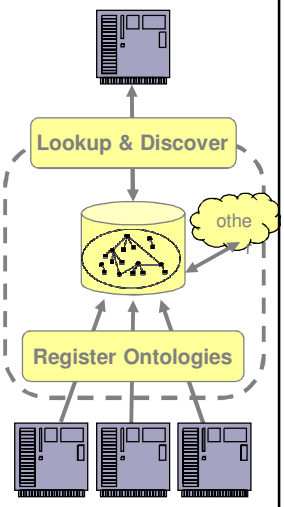


38

Registry Service

Semantic registries provide a federated registry for the semantics of schemas, ontologies, and applications – they will become the DNS for application semantics.

- Registries are services
  - They may be used in applications or middleware
- Semantic registries are smart directories
  - Registries are a neutral place to register information about services, applications, schemas etc.
- Semantic content registries use OWL and/or RDF for internal metadata and inference
  - Better, standard, data and knowledge representation
- Semantic content registries enable discovery
  - With inference, discovery of new content and services will be as easy as issuing an ad-hoc query

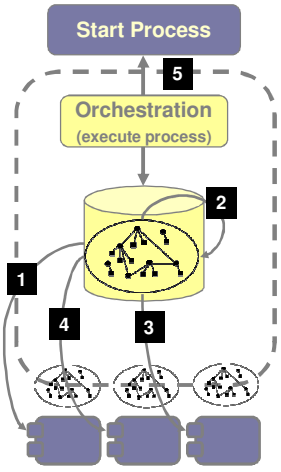


39

Process Engine

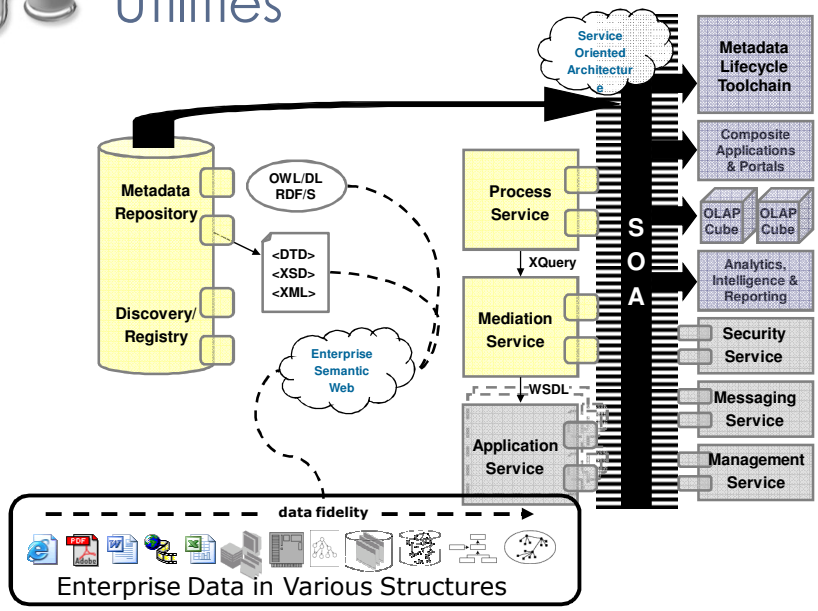
Process engines orchestrate multi-step and long-lived transactions by intelligently determining which services to use and how to collaborate with them in order to fulfill a business request

- Process engines are services
  - They may be used in applications or middleware
- Process engines act on workflow/procedures
  - A process engine can interpret a process model and interoperate with annotated web services
- Process engines use OWL ontologies
  - Better, standard, data and knowledge representation for process models as well as interface descriptions
- Process uses Mediation, Repository, Registry
  - Process engines use registry for finding services, mediation for transforming data, and repository for storing process models



40

Utilities



41

Utility Specifications

Separation of "semantic" concerns

	Metadata Service	Mediation Service	Registry Service	Process Service
Key APIs	ebXML, JAX-R, OWL-API, proprietary	Query (XQuery, proprietary), XML results	UDDI, ebXML	OWL-S, WSML, BPEL
Transactions	Request/Reply ACID Object-based	Request/Reply Publish/Subscribe Query-based	Request/Reply Query-based	Publish/Subscribe Request/Reply ACID
Input	Management APIs	Query	Query	Management API/invocation
Output	Objects and Documents (XML, OWL, RDF)	XML Documents	Pointers/URIs	Multi-step actions and confirmations
Uses (other services)	Mediation, Registry	Metadata, Registry	Metadata	Registry, Mediation, Metadata
Maturity	Medium (COTS available)	Medium (COTS available)	Low (COTS w/out OWL)	Low (COTS w/out OWL/WSML)

network inference

42

network inference

Component Architecture Patterns

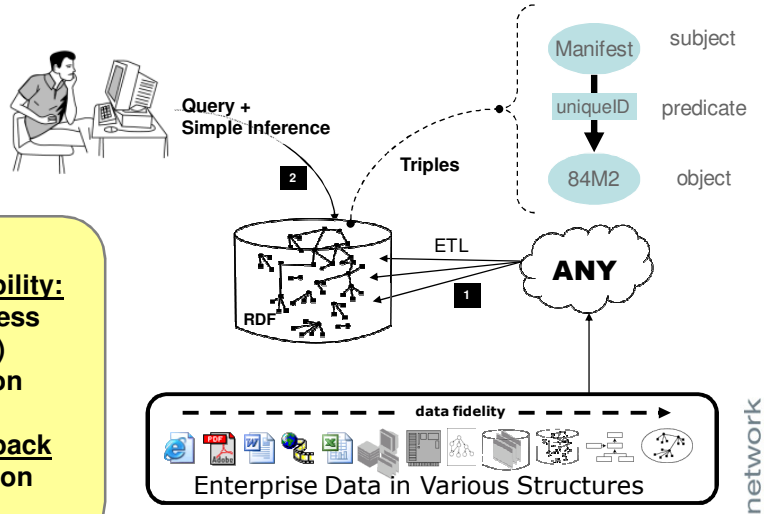
RDF Database
RDF Facade
OWL Knowledge Base
OWL Facade
OWL Instance Store
Hybrid Fidelity Reasoning

Architecture Patterns?

- Build your own utilities
  - (or buy COTS tools that use these patterns!)
- How many ways to "skin a cat?"
  - Proprietary (algorithms and data formats)
  - RDF (data and some inference)
  - OWL (powerful, correct inference)
  - Hybrid (right tools for the job)
- Key Concepts:
  - New "kinds" of architecture layers and tradeoffs
  - Mapping and wrapping (your legacy systems)
  - Separate data from ontology (different uses)
  - Understand your challenges and needs!

network inference

RDF Database (1)



Key Capability:
- **Schemaless**
- **(Organic)**
- **Federation**

Key Drawback
- **Conversion**

network inference

45

RDF Database (2)

Name: **RDF Database**

What: A database system with a native RDF triples graph format at its core. May include traditional database management system utilities such as a query interface, import/export, schema support, and session management. Advanced capabilities may include basic inferencing, federation, optimizations, and clustering.

Solution: Use directed graphs as a better data format for frequently changing, or complexly-related data. RDF databases are characterized as "schemaless" because they may grow organically. Directed graphs federate easily because schemas need not be 'aligned' for heterogeneous queries.

Variations: "Triplesbase," "Graph management,"

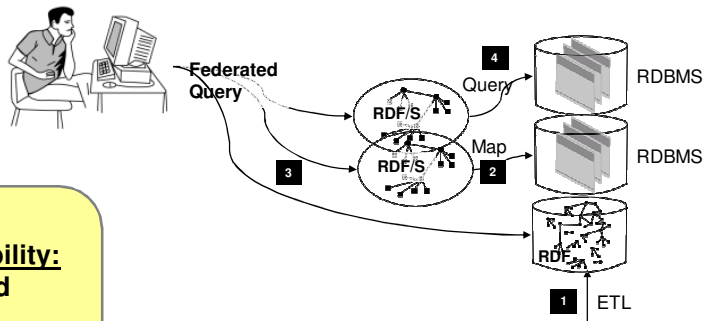
Implementations: Sesame, Kowari, RDFdb, RDF Gateway, @Semantics

Maturity: [Commodity, Mature, **Evolving**, Prototype, Research]

network inference

46

RDF Façade (1)



Key Capability:
- **Federated**
- **(Query)**
- **Organic/Graph**

Key Drawback
- **Simplistic**

network inference

RDF Façade (2)

Name: **RDF Facade**

What: A brokering system that enables (a) a mapping between relational DB schema and RDF/schema and (b) a SQL query to be generated to the underlying legacy database. Thus, relational sources may be federated and annotated as if they were RDF graph-based systems.

Solution: Federating graph based systems may prove to be much simpler due to different algebra for distributed query. Linking in a federated system is performed using shared predicates or subjects across different physical implementations. This approach takes advantage of the URI "web-effect" of RDF data.

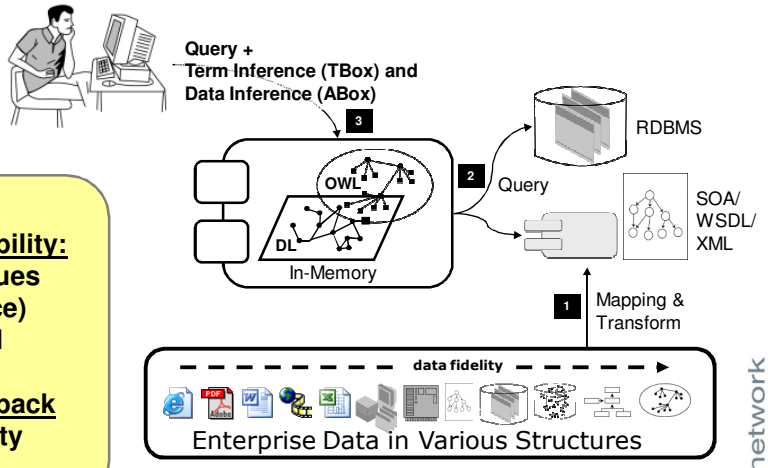
Variations: "Triplesbase," "Graph management,"

Implementations: Kowari, D2RQ/D2R-Map (Jena extensions), @Semantics?

Maturity: [Commodity, Mature, Evolving, **Prototype**, Research]

network inference

OWL Knowledge Base (1)



Key Capability:
- **Data Values**
- **(Inference)**
- **Powerful**

Key Drawback
- **Scalability**

network inference

49

OWL Knowledge Base (2)

Name: **OWL-DL Knowledge Base**

What: A traditional "knowledge base" in the KR sense – built-in inference and rich data structures allow for discovery of implicit knowledge in the system. A complete OWL KB uses both "tbox" and "abox" containment for concepts and instances respectively. An OWL KB provides a rich formalism for changing business data and rules, greater design precision allows for more flexible query and data structures.

Solution: An in-memory, query-able, graph system for (a) schema concepts and (b) data values stores data/metadata as description logics – algorithms implemented as "chaining-rules" or "tableaux" can deduce new facts from stated models/data.

Variations: DL-Inference Engine, DLDB

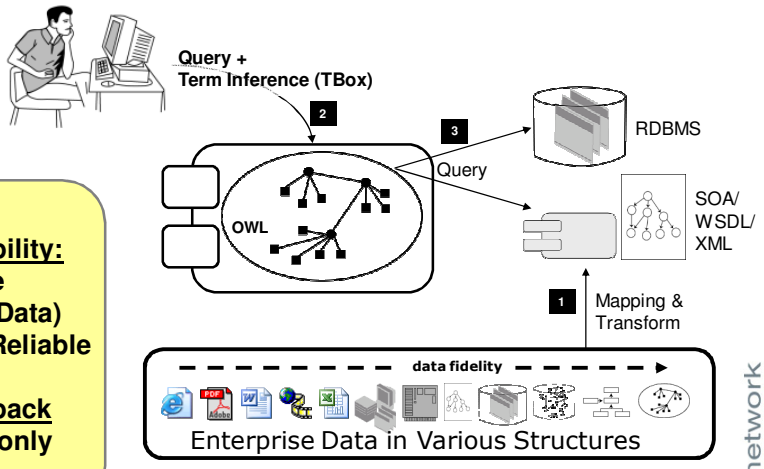
Implementations: Network Inference Cerebra, Racer Systems, Pellet, FaCT

Maturity: [Commodity, Mature, **Evolving**, Prototype, Research]

network inference

50

OWL Façade (1)



Key Capability:
- **Inference**
- **(Implicit Data)**
- **Correct/Reliable**

Key Drawback
- **Schema-only**

network inference

OWL Façade (2)

Name: **OWL Façade**

What: A method to layer OWL ontology "over" existing software systems to expose one or more views into the underlying data. The OWL façade may reliably re-present the system's data, possibly with additional metadata, for exposure and query via a third-party inference system.

Solution: Annotate OWL models with additional metadata that links model concepts to physical legacy source data. For example, annotate an ontology concept to a column in a database or a tag-set within an XML schema in a Web Service. Then use an inference engine to deduce "answer sets" from the model and retrieve external instance data values.

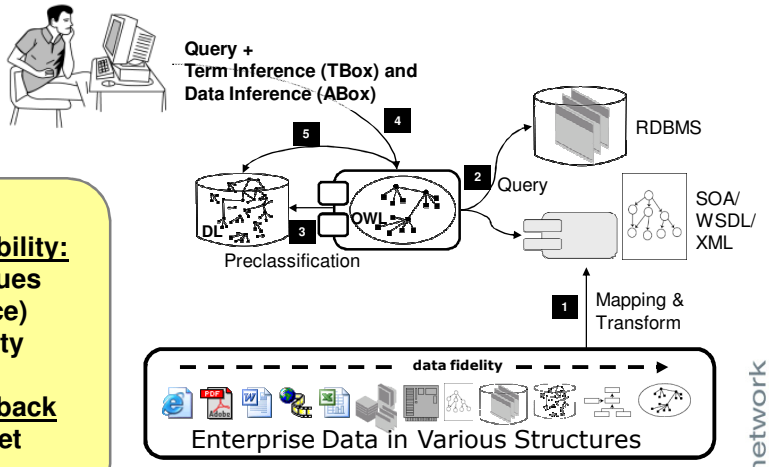
Variations: "OWL External Instance Retrieval"

Implementations: Network Inference Cerebra, CYC-KB?

Maturity: [Commodity, Mature, **Evolving**, Prototype, Research]

network inference

OWL Instance Store (1)



Key Capability:
- **Data Values**
- **(Inference)**
- **Scalability**

Key Drawback
- **DL-subset**

network inference

53

OWL Instance Store (2)

Name: **OWL Instance Store**

What: In recognition of the scalability limitations of "abox" instance reasoning for in-memory inference engines, the instance store uses formal equations to preclassify instance data into a persisted format (relational or RDF). Then, more optimized queries may be issued to vast quantities of classified OWL instance data.

Solution: Reasoning with databases. Instance data is preclassified via an inference, where assertions are stored within a DB, along with taxonomic information from the ontology, thus reducing the reasoning complexity (best case) to pure "tbox" – assuming relations among instances are not present (aka: role-free).

Variations: "Synthetic ABox Reasoning"

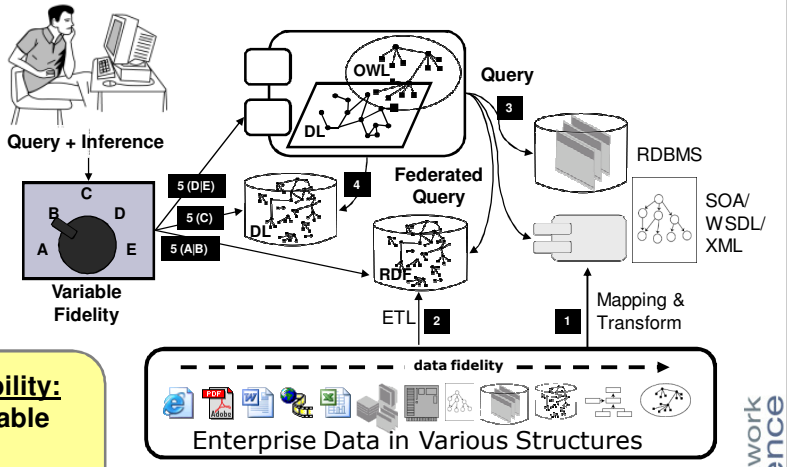
Implementations: Network Inference Cerebra, UMan. iStore,

Maturity: [Commodity, Mature, Evolving, **Prototype**, Research]

network inference

54

Hybrid Fidelity Reasoning (1)



Key Capability:
- **Configurable**

Key Drawback
- **It's Fiction!**

network inference

Hybrid Fidelity Reasoning (2)

Name: **Hybrid Fidelity Reasoning**

What: A hybrid approach to reasoning on semantic web data; in recognition that different queries will necessitate different levels of entailment, or reasoning. Generally speaking, the capability should be in place to sacrifice performance for power, or deductive power for performance within a federated knowledge system.

Solution: The collection of previous patterns, in a shared architecture, with a common query façade layered in front. The query façade should allow developers the capability to "select" an inference power/performance or execute an "anytime algorithm" that gives answers progressively, as more inference completes.

Variations: Anytime Reasoning, (related: Approximate Entailment)

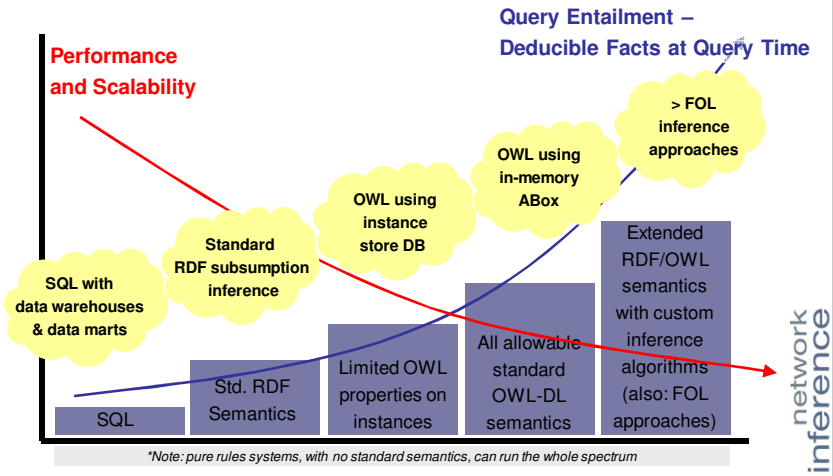
Implementations: Various research, university testbeds

Maturity: [Commodity, Mature, Evolving, Prototype, **Research**]

network inference

Hybrid Fidelity Reasoning (3)

All things being equal, a query with greater entailment will always be less performant than the same query with less entailment.



*Note: pure rules systems, with no standard semantics, can run the whole spectrum

network inference

57

Pattern Summary

	RDF Database	RDF Façade	OWL Façade	OWL Instance Store	OWL Knowledge Base	Hybrid Fidelity
Core Data Format	RDF Graphs	RDF Graphs	OWL Concepts	OWL Concepts & DB Data	OWL Concepts & Instances	RDF & OWL (all)
Inference Capability	Simple Subsumption/ custom RDF	Simple Subsumption/ custom RDF	OWL "TBox"	OWL "TBox" & some "ABox"	OWL "TBox & "ABox"	RDF & OWL
Performance	High – (billions & fast)	High – DB quality + RDF inference	High – DB quality + OWL inference	Medium – OWL "TBox" for concepts and data (100's millions)	Low – OWL in-memory "TBox" and "Abox" (millions)	Variable
Schema Federation	RDF/S	RDF/S	OWL-DL	OWL-DL	OWL-DL	RDF/S & OWL-DL
Data Federation	RDF Triples	RDF Triples via RDF/S mappings	RDF Triples via OWL-DL mappings	RDF Triples via OWL-DL mappings	RDF Triples via OWL-DL mappings	RDF Triples via OWL-DL or RDF/S
Maturity	High (evolving)	Medium (prototype)	High (evolving)	Medium (prototype)	High (evolving)	Low (research)

network inference

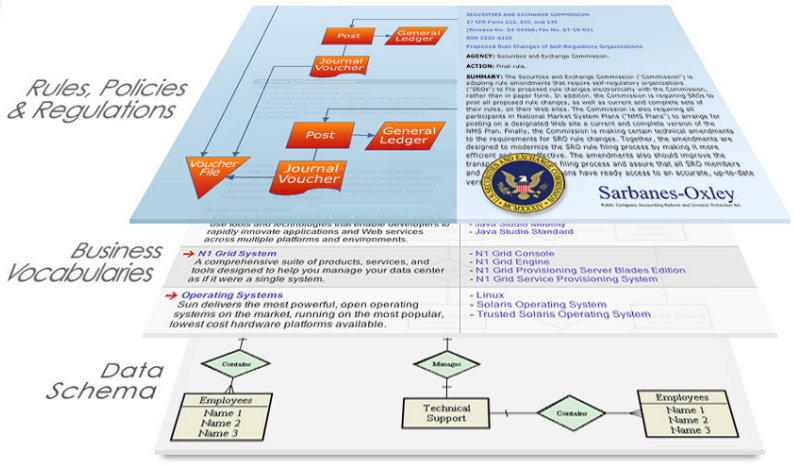
58

network inference

Logical Architecture

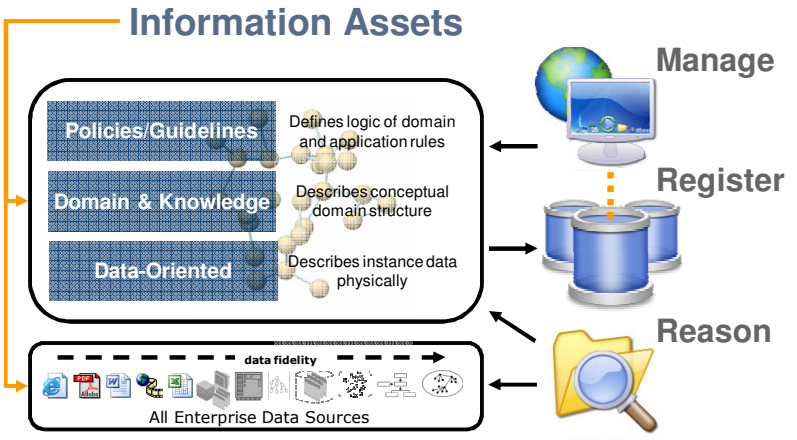
Logical Architecture (1)

Decouple critical business IP from the software that executes the logic



network inference

Logical Architecture (2)



61

network inference

"Pocket" Definitions

Inference?
Logic With Less Code

Deduce **implicit facts** on explicit data at runtime. Get more application knowledge with **less coding.** Keep your business knowledge in **models, not algorithms.** Maybe learn a few new things too…

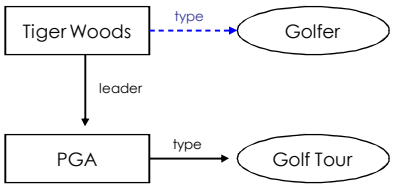
```
<owl:Property rdf:ID="leader">
  <rdfs:subPropertyOf
        rdf:resource="member" />
</owl:Property>

<owl:Class rdf:ID="Golfer">
  <owl:equivalentClass>
    <owl:Restriction>
      <owl:onProperty
            rdf:resource="member" />
      <owl:someValuesFrom
            rdf:resource="Golf Tour" />
    </owl:Restriction>
  </owl:equivalentClass>
</owl:Class>
```

The **leader** of a **tour** is also a **member** of it

A **member** of a **golf tour** is a **golfer**

Therefore, the **leader** of a **golf tour** is a **golfer**

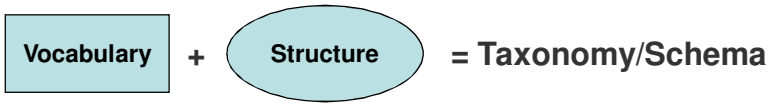


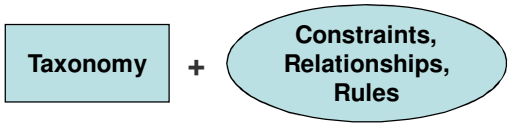
network inference

63

Ontology?

- Many definitions:
  - Vague – "a specification of a conceptualization"
  - Narrow – "the explicit meaning of terms in a vocabulary and relations between those terms"



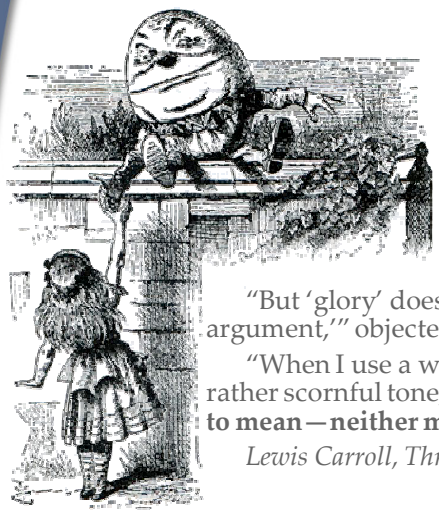


network inference

64

A Little Ironic…

Q: What are semantics?



"I don't know what you mean by 'glory,'" Alice said.

Humpty Dumpty smiled contemptuously. "Of course you don't—till I tell you. I meant 'there's a nice knock-down argument for you!'"

"But 'glory' doesn't mean 'a nice knock-down argument,'" objected Alice.

"When I use a word," Humpty Dumpty said, in a rather scornful tone, "**it means just what I chose it to mean—neither more nor less.**"

Lewis Carroll, Through The Looking-Glass

network inference

65

Semantics? (OWL Formal)

Constructor	DL Syntax	Example
intersectionOf	$C_1 \sqcap \ldots \sqcap C_n$	Human ⊓ Male
unionOf	$C_1 \sqcup \ldots \sqcup C_n$	Doctor ⊔ Lawyer
complementOf	$\neg C$	¬Male
oneOf	$\{x_1 \ldots x_n\}$	{john, mary}
allValuesFrom	$\forall P.C$	∀hasChild.Doctor
someValuesFrom	$\exists P.C$	∃hasChild.Lawyer
maxCardinality	$\leqslant nP$	⩽1hasChild
minCardinali		

CLASS CONSTRUCTORS

AXIOMS

Axiom	DL Syntax	Example
subClassOf	$C_1 \sqsubseteq C_2$	Human ⊑ Animal ⊓ Biped
equivalentClass	$C_1 \equiv C_2$	Man ≡ Human ⊓ Male
disjointWith	$C_1 \sqsubseteq \neg C_2$	Male ⊑ ¬Female
sameIndividualAs	$\{x_1\} \equiv \{x_2\}$	{President_Bush} ≡ {G_W_Bush}
differentFrom	$\{x_1\} \sqsubseteq \neg\{x_2\}$	{john} ⊑ ¬{peter}
subPropertyOf	$P_1 \sqsubseteq P_2$	hasDaughter ⊑ hasChild
equivalentProperty	$P_1 \equiv P_2$	cost ≡ price
inverseOf	$P_1 \equiv P_2^-$	hasChild ≡ hasParent$^-$
transitiveProperty	$P^+ \sqsubseteq P$	ancestor$^+$ ⊑ ancestor
functionalProperty	$\top \sqsubseteq \leqslant 1P$	⊤ ⊑ ⩽1hasMother
inverseFunctionalProperty	$\top \sqsubseteq \leqslant 1P^-$	⊤ ⊑ ⩽1hasSSN$^-$

network inference

66


network inference
Questions?
Jeff Pollock
Vice President, Technology
jeff.pollock@networkinference.com