A Wiki and SOA Endpoint for Rules in Open Vocabulary, Executable English

Adrian Walker

reengineeringllc.com

Background Slides for a Demo at the
International RuleML Symposium on Rule Interchange and Applications
October 2007, Orlando, Florida

Experience

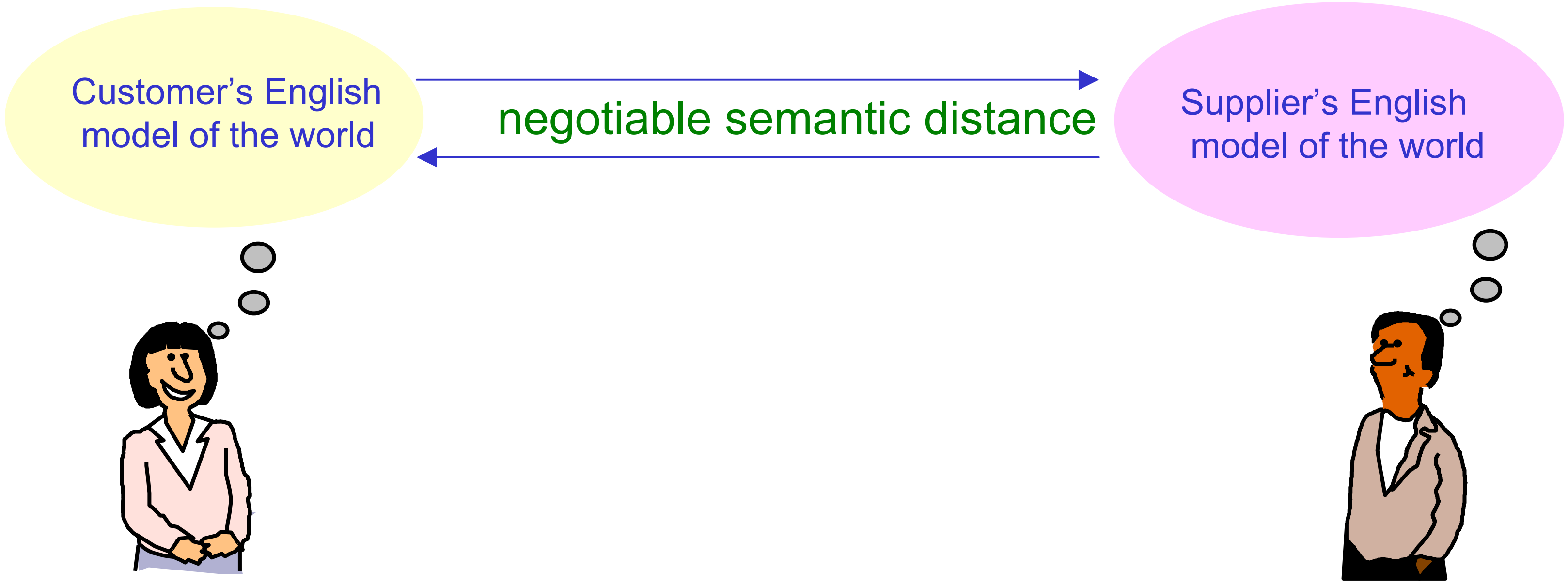
- Author of over 20 papers, and an addison-wesley book, on rules systems and databases
- Assistant professor at Rutgers university
- Manager of principles and applications of logic programming, IBM yorktown research laboratory
- Manager, internet development at eventra

  (A manufacturing supply chain company)

Agenda

- Enabling closer dialog between SOA stakeholders
  - Users, business analysts, enterprise architects, and system engineers
- A wiki for content in open vocabulary, executable English
  - Write applications a rules in English, run them, and get English explanations
- Google indexes and retrieves content in executable English
  - Acts as a kind of registry
- The wiki engine as an SOA knowledge endpoint on the web
- A supply chain example
- Automatic generation of complex, distributed SQL
  - With English explanations of results
- Capturing generated SQL for re-use
- Summary

Enabling Closer Dialog Between SOA Stakeholders

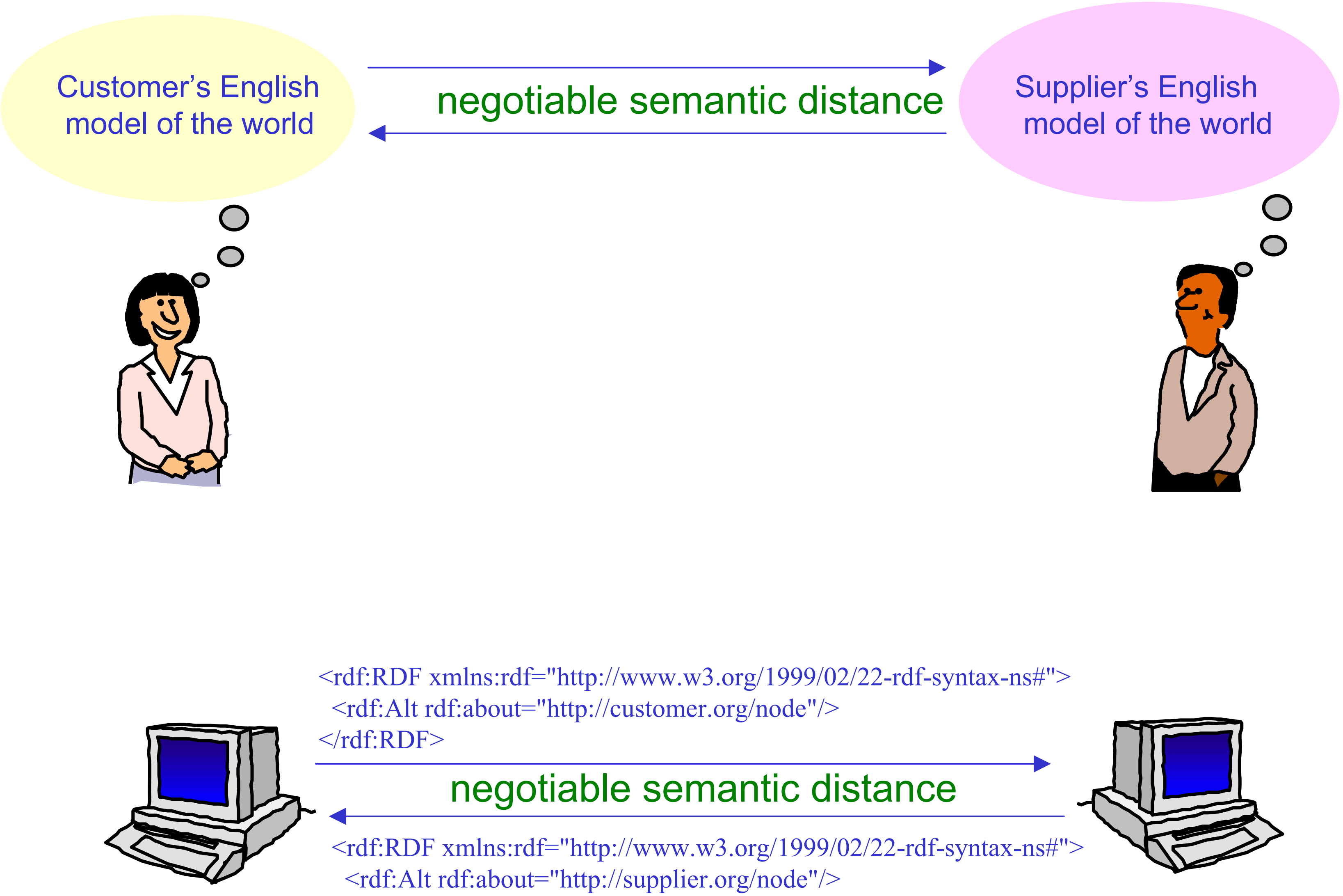


Customer’s English model of the world

negotiable semantic distance

Supplier’s English model of the world

Enabling Closer Dialog Between SOA Stakeholders



Customer’s English model of the world

negotiable semantic distance

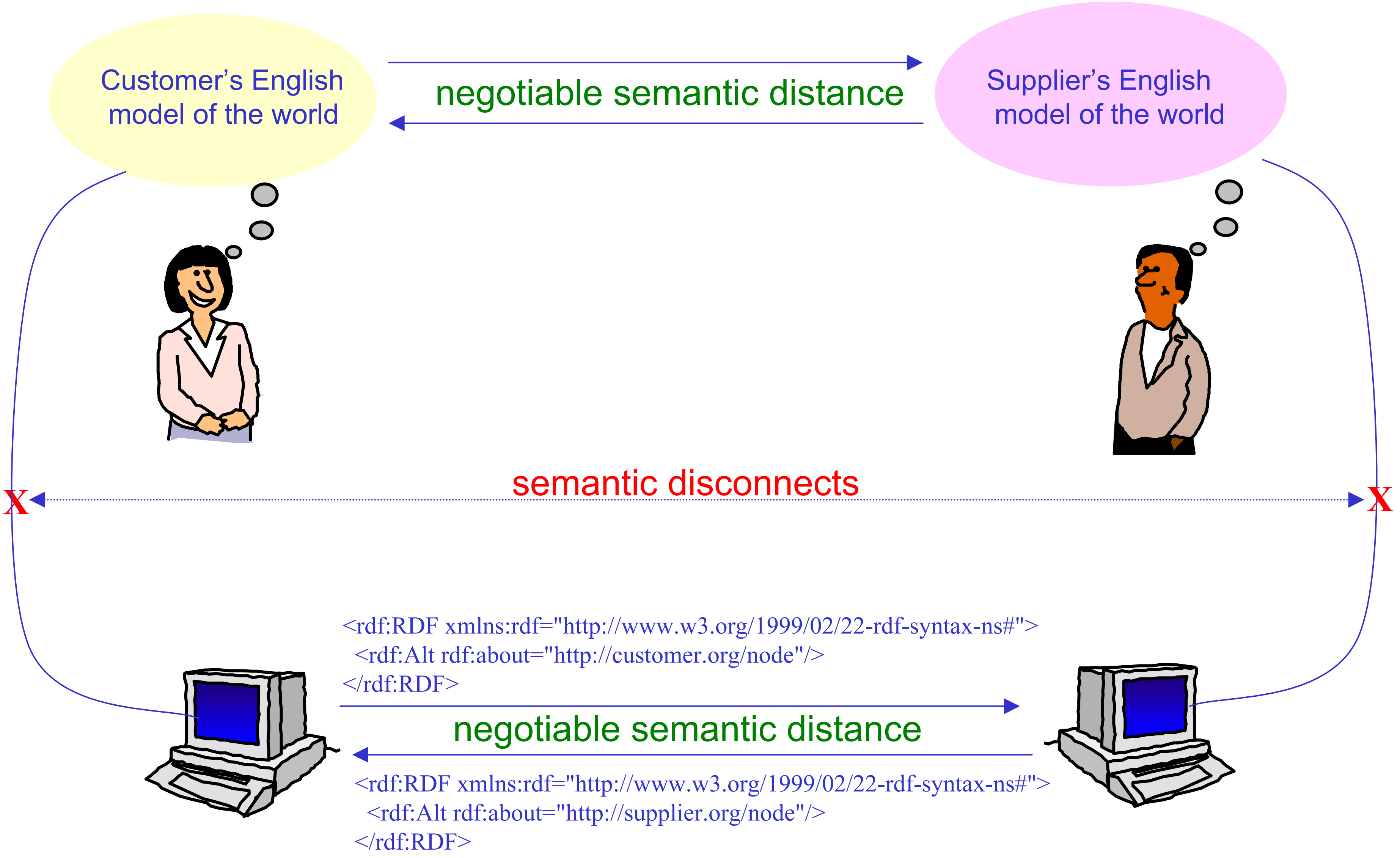
Supplier’s English model of the world

```
<rdf:RDF xmlns:rdf="http://www.w3.org/1999/02/22-rdf-syntax-ns#">
  <rdf:Alt rdf:about="http://customer.org/node"/>
</rdf:RDF>
```

negotiable semantic distance

```
<rdf:RDF xmlns:rdf="http://www.w3.org/1999/02/22-rdf-syntax-ns#">
  <rdf:Alt rdf:about="http://supplier.org/node"/>
</rdf:RDF>
```

Enabling Closer Dialog Between SOA Stakeholders



Customer’s English model of the world

negotiable semantic distance

Supplier’s English model of the world

semantic disconnects

X

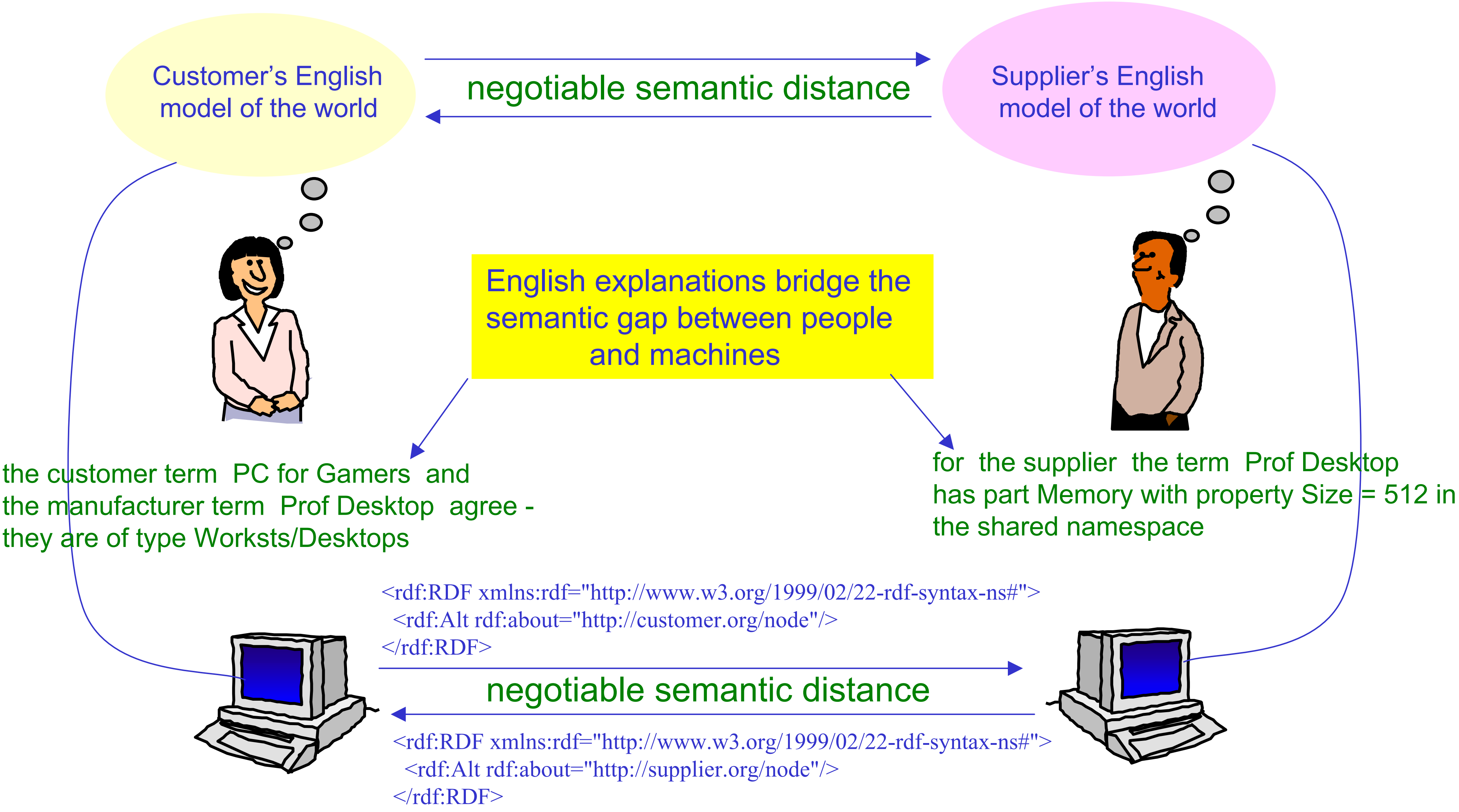
X

```
<rdf:RDF xmlns:rdf="http://www.w3.org/1999/02/22-rdf-syntax-ns#">
  <rdf:Alt rdf:about="http://customer.org/node"/>
</rdf:RDF>
```

negotiable semantic distance

```
<rdf:RDF xmlns:rdf="http://www.w3.org/1999/02/22-rdf-syntax-ns#">
  <rdf:Alt rdf:about="http://supplier.org/node"/>
</rdf:RDF>
```

Enabling Closer Dialog Between SOA Stakeholders



Customer’s English model of the world

negotiable semantic distance

Supplier’s English model of the world

English explanations bridge the semantic gap between people and machines

the customer term PC for Gamers and the manufacturer term Prof Desktop agree - they are of type Workststs/Desktops

for the supplier the term Prof Desktop has part Memory with property Size = 512 in the shared namespace

```
<rdf:RDF xmlns:rdf="http://www.w3.org/1999/02/22-rdf-syntax-ns#">
  <rdf:Alt rdf:about="http://customer.org/node"/>
</rdf:RDF>
```

negotiable semantic distance

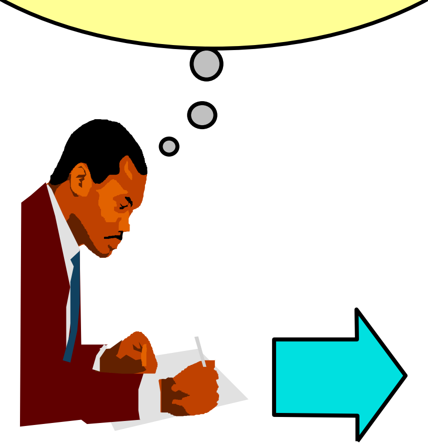
```
<rdf:RDF xmlns:rdf="http://www.w3.org/1999/02/22-rdf-syntax-ns#">
  <rdf:Alt rdf:about="http://supplier.org/node"/>
</rdf:RDF>
```

Agenda

- Enabling closer dialog between SOA stakeholders
  - Users, business analysts, enterprise architects, and system engineers
- A wiki for content in open vocabulary, executable English
  - Write applications in English, run them, and get English explanations
- Google indexes and retrieves content in executable English
  - Acts as a kind of registry
- The wiki engine as an SOA knowledge endpoint on the web
- A supply chain example
- Automatic generation of complex, distributed SQL
  - With English explanations of results
- Capturing generated SQL for re-use
- Summary

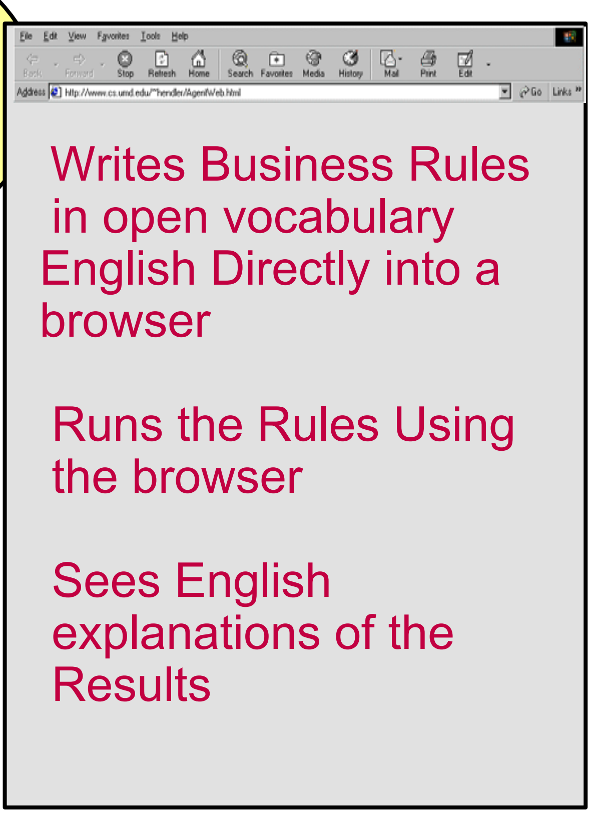
A Wiki for content in open vocabulary, executable English

Products x and z can be
substituted for product y,
but only in the Fall



End User /
Business Analyst

Semantics3

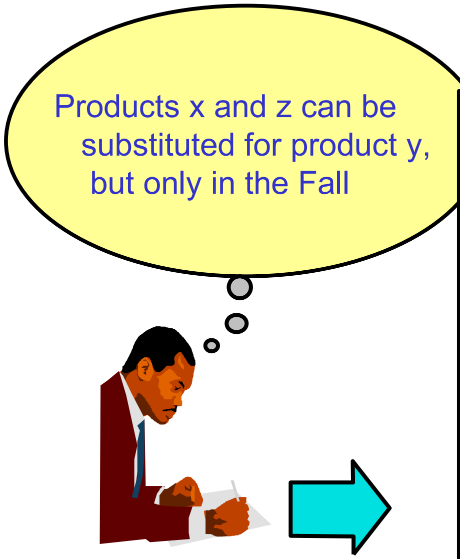

File Edit View Favorites Tools Help
Back Forward Stop Refresh Home Search Favorites Media History Mail Print Edit
Address http://www.cs.umd.edu/~hendler/AgentWeb.html
Go Links

Writes Business Rules
in open vocabulary
English Directly into a
browser

Runs the Rules Using
the browser

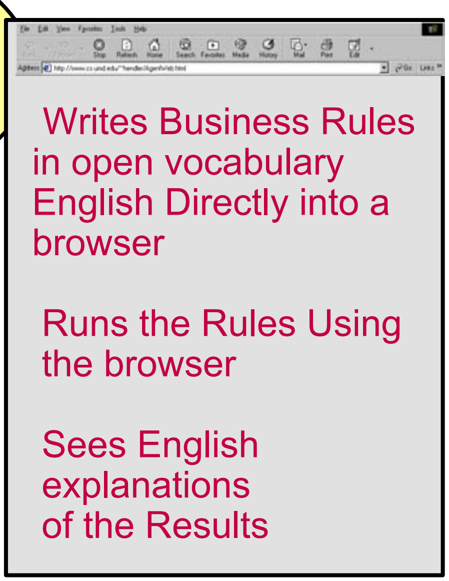
Sees English
explanations of the
Results

A Wiki for content in open vocabulary, executable English


Products x and z can be substituted for product y, but only in the Fall

End User / Business Analyst

Semantics3

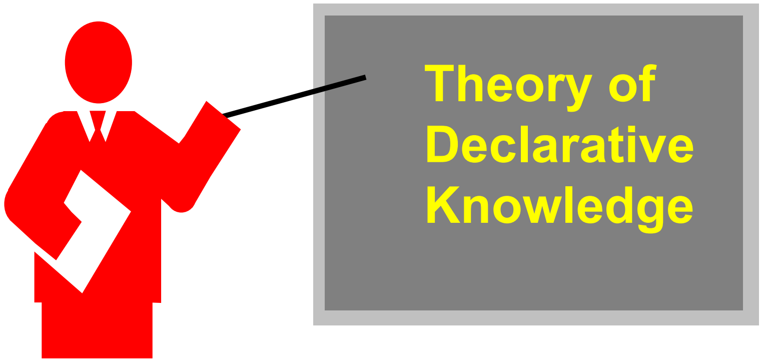


Writes Business Rules in open vocabulary English Directly into a browser

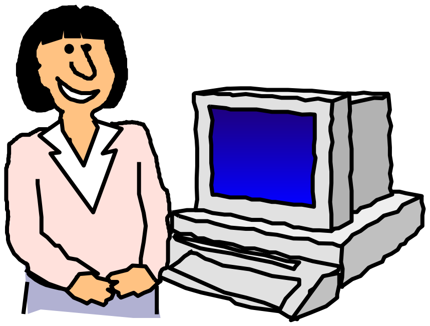
Runs the Rules Using the browser

Sees English explanations of the Results

Semantics2

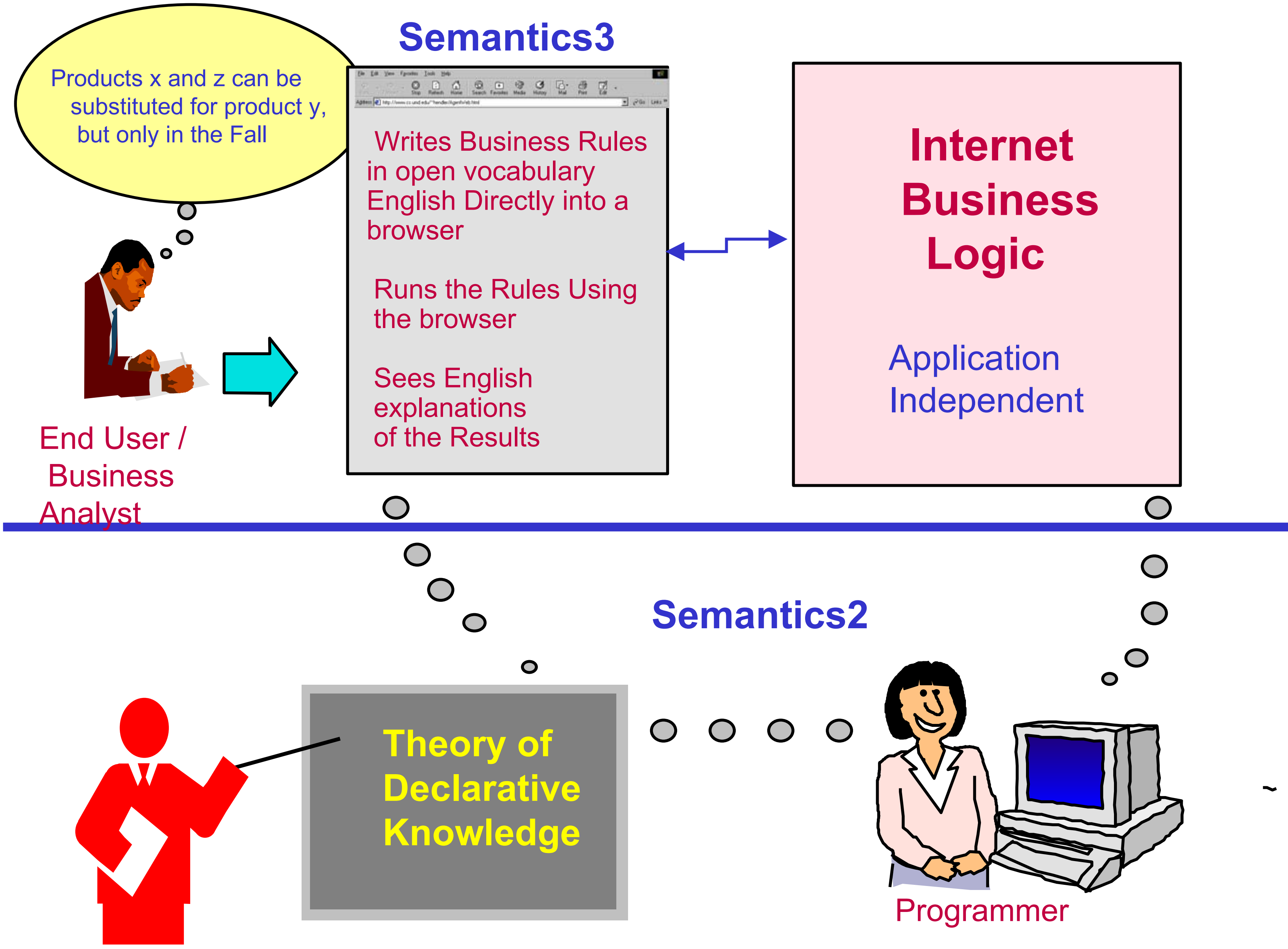


Theory of Declarative Knowledge



Programmer

A Wiki for content in open vocabulary, executable English



Products x and z can be substituted for product y, but only in the Fall

End User / Business Analyst

Semantics3

Writes Business Rules in open vocabulary English Directly into a browser

Runs the Rules Using the browser

Sees English explanations of the Results

Internet Business Logic

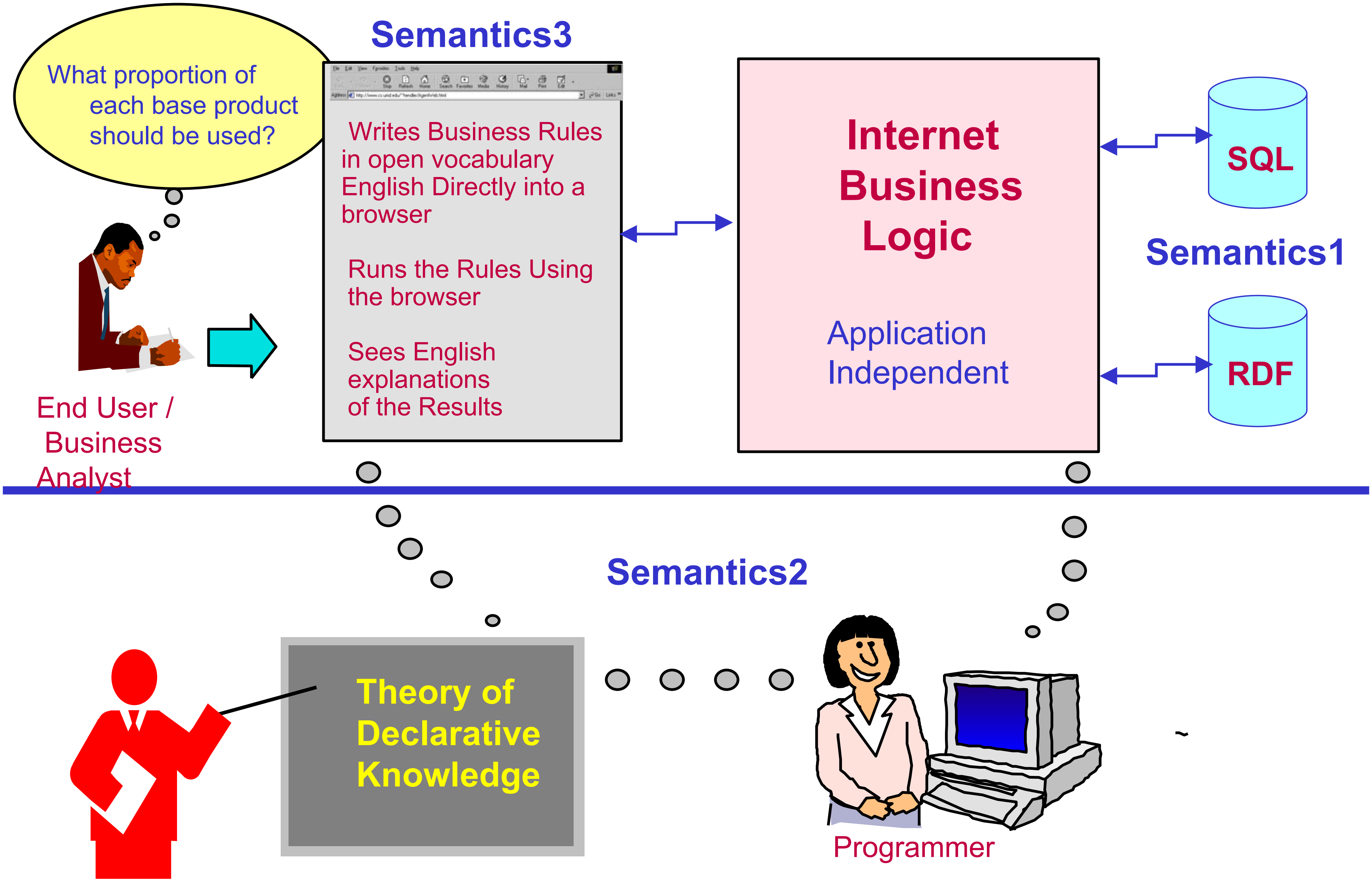
Application Independent

Semantics2

Theory of Declarative Knowledge

Programmer

A Wiki for content in open vocabulary, executable English



What proportion of each base product should be used?

Semantics3

Writes Business Rules in open vocabulary English Directly into a browser

Runs the Rules Using the browser

Sees English explanations of the Results

Internet Business Logic

Application Independent

SQL

Semantics1

RDF

End User / Business Analyst

Semantics2

Theory of Declarative Knowledge

Programmer

A Wiki for content in open vocabulary, executable English

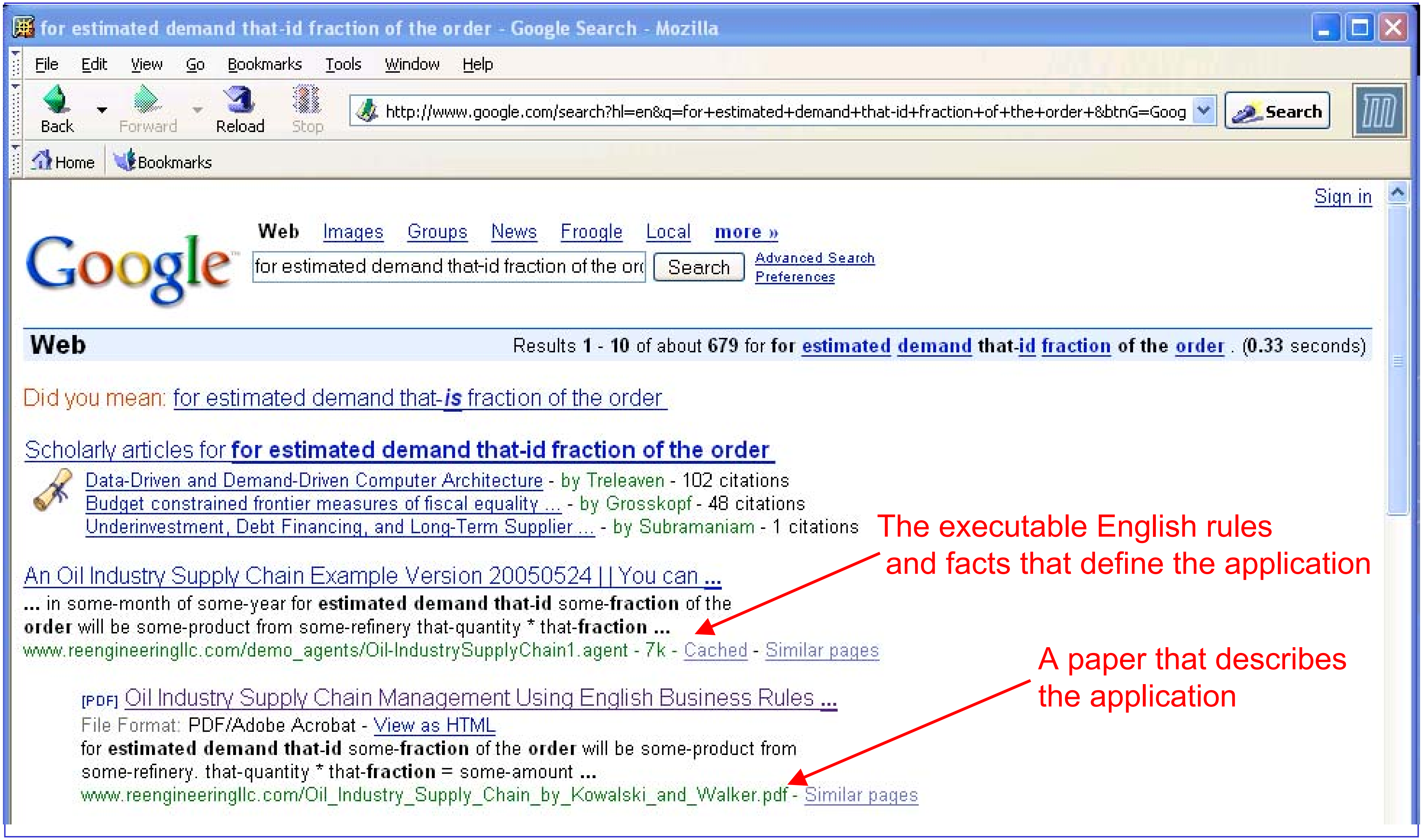
- The vocabulary is open, and so -- to a large extent -- is the syntax
  - ***not*** yet-another-controlled-English-system
- No dictionary or grammar maintenance is needed
- Strict English semantics is achieved via a trade off
  - if you want two English sentences to mean the same thing, you must say so
  - you must use place holders, such as “some-name” and “a-number”
- But, you are free to write executable English knowledge containing...
  - technical terms or jargon -- Wildcat, Upstream, Mud (oil industry)
  - government acronyms and usage -- SRB, Single Regeneration Budget
  - logical expressions -- (A c,t) [that-C c t => (E c1) [that-C1 c1 t and c partof c1 at t]]
- Although the system is open vocabulary, it can be used to query and manage:
  - controlled vocabularies, taxonomies and ontologies

Agenda

- Enabling closer dialog between SOA stakeholders
  - Users, business analysts, enterprise architects, and system engineers
- A wiki for content in open vocabulary, executable English
  - Write applications in English, run them, and get English explanations
- Google indexes and retrieves content in executable English
  - Acts as a kind of registry
- The wiki engine as an SOA knowledge endpoint on the web
- A supply chain example
- Automatic generation of complex, distributed SQL
  - With English explanations of results
- Capturing generated SQL for re-use
- Summary

Google indexes and retrieves content in executable English

Search: for estimated demand that-id fraction of the order


for estimated demand that-id fraction of the order - Google Search - Mozilla
File
Edit
View
Go
Bookmarks
Tools
Window
Help
Back
Forward
Reload
Stop
http://www.google.com/search?hl=en&q=for+estimated+demand+that-id+fraction+of+the+order+&btnG=Goog
Search
Home
Bookmarks
Sign in
Google
Web
Images
Groups
News
Froogle
Local
more »
for estimated demand that-id fraction of the or
Search
Advanced Search
Preferences

Web Results **1** - **10** of about **679** for **for estimated demand that-id fraction of the order** . (**0.33** seconds)

Did you mean: for estimated demand that-*is* fraction of the order

Scholarly articles for **for estimated demand that-id fraction of the order**

Data-Driven and Demand-Driven Computer Architecture - by Treleaven - 102 citations
Budget constrained frontier measures of fiscal equality ... - by Grosskopf - 48 citations
Underinvestment, Debt Financing, and Long-Term Supplier ... - by Subramaniam - 1 citations

An Oil Industry Supply Chain Example Version 20050524 | | You can ...

... in some-month of some-year for **estimated demand that-id** some-**fraction** of the **order** will be some-product from some-refinery that-quantity * that-**fraction** ...
www.reengineeringllc.com/demo_agents/Oil-IndustrySupplyChain1.agent - 7k - Cached - Similar pages

[PDF] Oil Industry Supply Chain Management Using English Business Rules ...
File Format: PDF/Adobe Acrobat - View as HTML
for **estimated demand that-id** some-**fraction** of the **order** will be some-product from some-refinery. that-quantity * that-**fraction** = some-amount ...
www.reengineeringllc.com/Oil_Industry_Supply_Chain_by_Kowalski_and_Walker.pdf - Similar pages

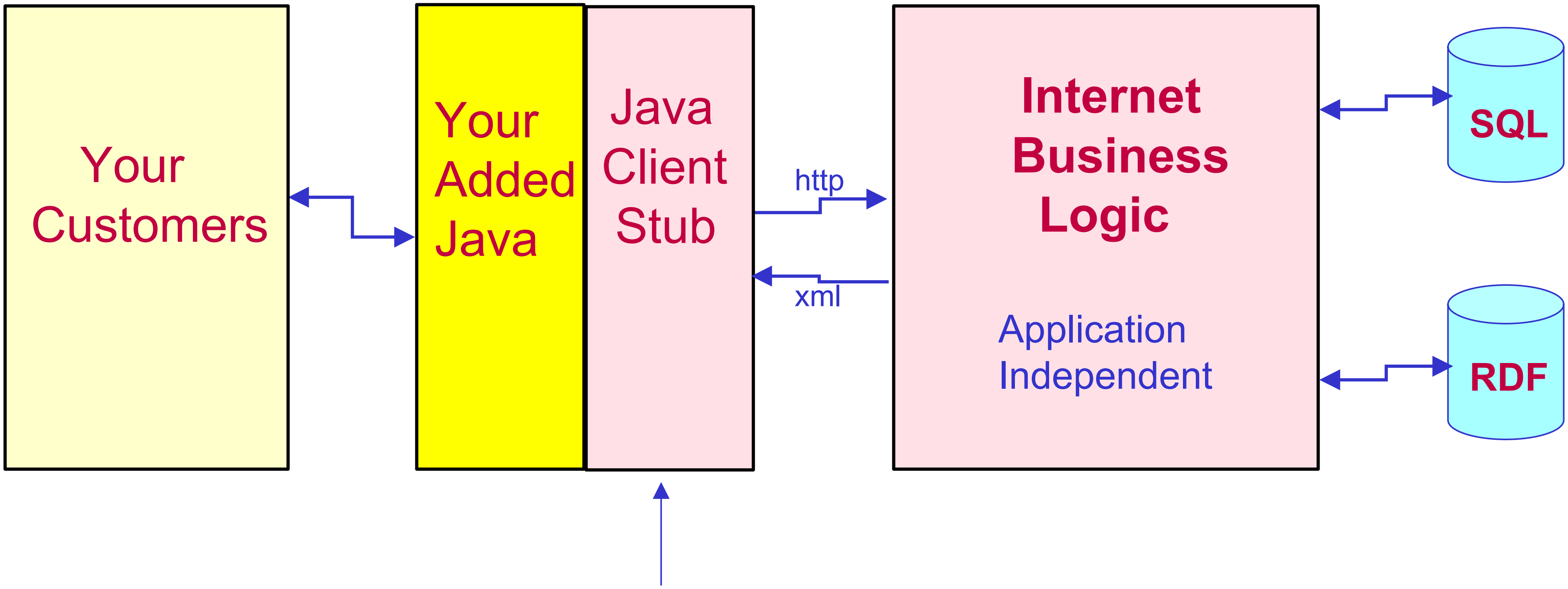
The executable English rules and facts that define the application

A paper that describes the application

Agenda

- Enabling closer dialog between SOA stakeholders
  - Users, business analysts, enterprise architects, and system engineers
- A wiki for content in open vocabulary, executable English
  - Write applications in English, run them, and get English explanations
- Google indexes and retrieves content in executable English
  - Acts as a kind of registry
- The wiki engine as an SOA knowledge endpoint on the web
- A supply chain example
- Automatic generation of complex, distributed SQL
  - With English explanations of results
- Capturing generated SQL for re-use
- Summary

The Wiki engine as an SOA endpoint on the Web

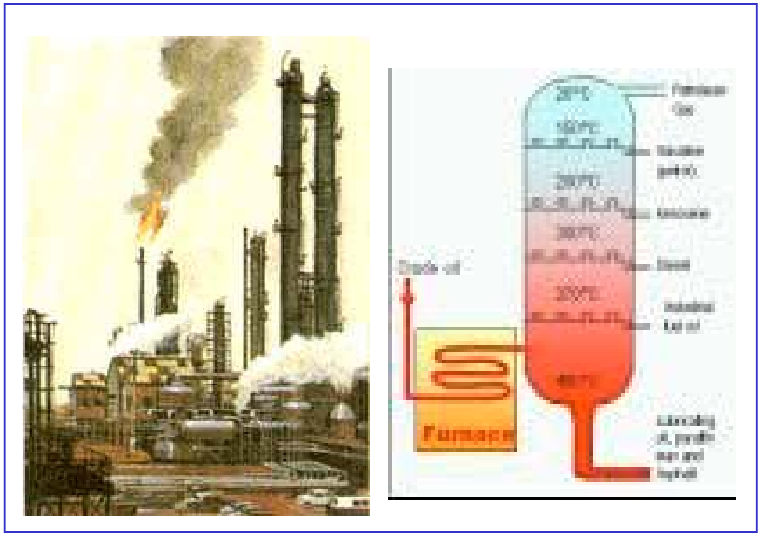


Downloadable from http://www.reengineeringllc.com/iblClient1.java

Agenda

- Enabling closer dialog between SOA stakeholders
  - Users, business analysts, enterprise architects, and system engineers
- A wiki for content in open vocabulary, executable English
  - Write applications in English, run them, and get English explanations
- Google indexes and retrieves content in executable English
  - Acts as a kind of registry
- The wiki engine as an SOA knowledge endpoint on the web
- A supply chain example
- Automatic generation of complex, distributed SQL
  - With English explanations of results
- Capturing generated SQL for re-use
- Summary

A supply chain example



Refinery

Terminal Storage

Terminal Pickup

Service Station

A supply chain example

- A customer needs 1000 gallons of product y in October
- Products x and z can be substituted for product y, but only in the Fall
- Combine products x, y and z to fill the order
- Combination depends on:
  - How much of each product is available from each refinery
  - Available transportation from each refinery to the customer area

-- Example based on

“Oil Industry Supply Chain Management Using English Business Rules Over SQL”

by Ted Kowalski and Adrian Walker,

www.reengineeringllc.com/Oil_Industry_Supply_Chain_by_Kowalski_and_Walker.pdf

A supply chain example

Facts:

estimated demand this-id in this-region is for this-quantity gallons of this-finished-product in this-month of this-year

523	NJ	1000	product-y	October	2005

in this-season an order for this-product1 can be filled with the alternative this-product2

Fall	product-y	product-x
Fall	product-y	product-z

in this-month the refinery this-name has committed to schedule this-amount gallons of this-product

October	Shell Canada One	500	product-y
October	Shell Canada One	300	product-x
October	Shell Canada One	800	product-z
October	Shell Canada One	10000	product-w

we have this-method transportation from refinery this-name to region this-region

truck	Shell Canada One	NJ
rail	Shell Canada One	NJ

A supply chain example

Rules:

```
estimated demand some-id in some-region is for some-quantity gallons of some-finished-product
   in some-month of some-year
for estimated demand that-id some-fraction of the order will be some-product from some-refinery
that-quantity * that-fraction = some-amount
-----------------------------------------------------------------------------------------------------------------------------
for demand that-id that-region for that-quantity that-finished-product we use that-amount that-product from that-refinery
```

```
estimated demand some-id in some-region is for some-quantity gallons of some-finished-product
   in some-month of some-year
for demand that-id for that-finished-product refinery some-refinery can supply some-amount gallons of some-product
for demand that-id the refineries have altogether some-total gallons of acceptable base products
that-amount / that-total = some-long-fraction
that-long-fraction rounded to 2 places after the decimal point is some-fraction
-------------------------------------------------------------------------------------------------
for estimated demand that-id that-fraction of the order will be that-product from that-refinery
```

```
estimated demand some-id in some-region is for some-amount gallons of some-product in some-month of some-year
sum a-num :
   for demand that-id for that-product refinery some-name can supply some-num gallons of some-product1 = a-total
---------------------------------------------------------------------------------------------------------
for demand that-id the refineries have altogether that-total gallons of acceptable base products
```

A supply chain example

An answer table:

```
for demand this-id this-region for this-quantity this-finished-product we use this-amount this-product from this-refinery
=========================================================================================================================
      523    NJ           1000       product-y              190.0      product-x          Shell Canada One
      523    NJ           1000       product-y              310.0      product-y          Shell Canada One
      523    NJ           1000       product-y              500.0      product-z          Shell Canada One
```

To run or change this example, please point IE6, Netscape7 or Mozilla to the demo Oil-IndustrySupplyChain1 at www.reengineeringllc.com

A supply chain example

An explanation:

```
     estimated demand 523 in NJ is for 1000 gallons of product-y in October of 2005
     for estimated demand 523 0.19 of the order will be product-x from  Shell Canada One
     1000 * 0.19 = 190
     ---------------------------------------------------------------------------------------------------
     for demand 523 NJ for 1000 product-y we use 190 product-x from  Shell Canada One


     estimated demand 523 in NJ is for 1000 gallons of product-y in October of 2005
     for demand 523 for product-y refinery  Shell Canada One  can supply 300 gallons of product-x
     for demand 523 the refineries have altogether 1600 gallons of acceptable base products
     300 / 1600 = 0.1875
     0.1875 rounded to 2 places after the decimal point is 0.19
     ---------------------------------------------------------------------------------------------------------------
     for estimated demand 523 0.19 of the order will be product-x from  Shell Canada One


  estimated demand 523 in NJ is for 1000 gallons of product-y in October of 2005
  sum   eg-amount   :
    for demand 523 for product-y refinery eg-refinery can supply eg-amount gallons of eg-product1   =   1600
  ------------------------------------------------------------------------------------------------------------------
  for demand 523 the refineries have altogether 1600 gallons of acceptable base products
```

To run or change this example, please point IE6, Netscape7 or Mozilla to the demo Oil-IndustrySupplyChain1 at www.reengineeringllc.com

Agenda

- Enabling closer dialog between SOA stakeholders
  - Users, business analysts, enterprise architects, and system engineers
- A wiki for content in open vocabulary, executable English
  - Write applications in English, run them, and get English explanations
- Google indexes and retrieves content in executable English
  - Acts as a kind of registry
- The wiki engine as an SOA knowledge endpoint on the web
- A supply chain example
- Automatic generation of complex, distributed SQL
  - With English explanations of results
- Capturing generated SQL for re-use
- Summary

Automatic generation of complex, distributed SQL

Rules for finding SQL data on the Internet:

A data table

```
we have this-method transportation from refinery this-name to region this-region
================================================================================
            truck                                       Shell Canada One    NJ
            rail                                        Shell Canada One    NJ
```

A rule that says how to find the table on the internet

```
url:www.example.com   dbms:9i   dbname:ibldb   tablename:T1   port:1521   id:anonymous   password:oracle
--------------------------------------------------------------------------------------------------------
we have this-method transportation from refinery this-name to region this-region
```

To run or change this example, please point IE6, Netscape7 or Mozilla to the demo Oil-IndustrySupplyChain1 at www.reengineeringllc.com

Automatic generation of complex, distributed SQL

A SQL query generated automatically from the supply chain knowledge:

```
select distinct x6,T2.PRODUCT,T1.NAME,T2.AMOUNT,x5 from
T6 tt1,T6 tt2,T5,T4,T3,T2,T1,T6,
(select x3 x6,T6.FINISHED_PRODUCT x7,T6.ID x8,tt1.ID x9,tt2.ID x10,sum(x4) x5 from
T6,T6 tt1,T6 tt2,
((select T6.ID x3,T3.PRODUCT1,T1.NAME,T2.AMOUNT x4,T2.PRODUCT from
T1,T2,T3,T4,T5,T6,T6 tt1,T6 tt2 where
T1.NAME=T2.NAME and T1.REGION=T6.REGION and T2.MONTH1=T4.MONTH1 and
T2.MONTH1=T6.MONTH1 and T2.PRODUCT=T3.PRODUCT2 and T4.MONTH1=T6.MONTH1 and
T3.PRODUCT1=T6.FINISHED_PRODUCT and T3.SEASON=T4.SEASON and T3.SEASON=T5.SEASON and
T4.SEASON=T5.SEASON and T6.ID=tt1.ID and T6.ID=tt2.ID and tt1.ID=tt2.ID)
union
(select T6.ID x3,T2.PRODUCT,T1.NAME,T2.AMOUNT x4,T2.PRODUCT from
T1,T2,T3,T4,T5,T6,T6 tt1,T6 tt2 where
T1.NAME=T2.NAME and T1.REGION=T6.REGION and T2.MONTH1=T6.MONTH1 and
T2.PRODUCT=T6.FINISHED_PRODUCT and T6.ID=tt1.ID and T6.ID=tt2.ID and tt1.ID=tt2.ID)
) group by T6.FINISHED_PRODUCT,T6.ID,tt1.ID,tt2.ID,x3) where
T6.ID=tt2.ID and tt1.ID=T6.ID and T6.FINISHED_PRODUCT=x7 and T6.ID=x8 and tt1.ID=x8 and
tt2.ID=x8 and T1.NAME=T2.NAME and T1.REGION=tt2.REGION and T2.MONTH1=T4.MONTH1 and
T2.MONTH1=tt2.MONTH1 and T2.PRODUCT=T3.PRODUCT2 and
T3.PRODUCT1=tt1.FINISHED_PRODUCT and T3.PRODUCT1=tt2.FINISHED_PRODUCT and
T3.SEASON=T4.SEASON and T3.SEASON=T5.SEASON and T4.MONTH1=tt2.MONTH1 and
T4.SEASON=T5.SEASON and T6.ID=x6 and tt1.FINISHED_PRODUCT=tt2.FINISHED_PRODUCT and
tt1.ID=tt2.ID and tt1.ID=x6 and tt2.ID=x6
order by x6,T2.PRODUCT,T1.NAME,T2.AMOUNT,x5;
```

Automatic generation of complex, distributed SQL

- It would be difficult to
  - write the SQL query reliably by hand
  - manually reconcile it with the business knowledge specified in the rules.
- Yet this is a ***simple*** example!

- How do we know that the automatically generated SQL yields results that are correct with respect to the business rules ?
- We can get step-by-step business level English explanations
  - same as in the non-SQL case

Automatic generation of complex, distributed SQL

- Could a programmer write more readable SQL by hand ?

- Yes, but we would need to add comments in English to help people to reconcile the hand-written query with the business knowledge

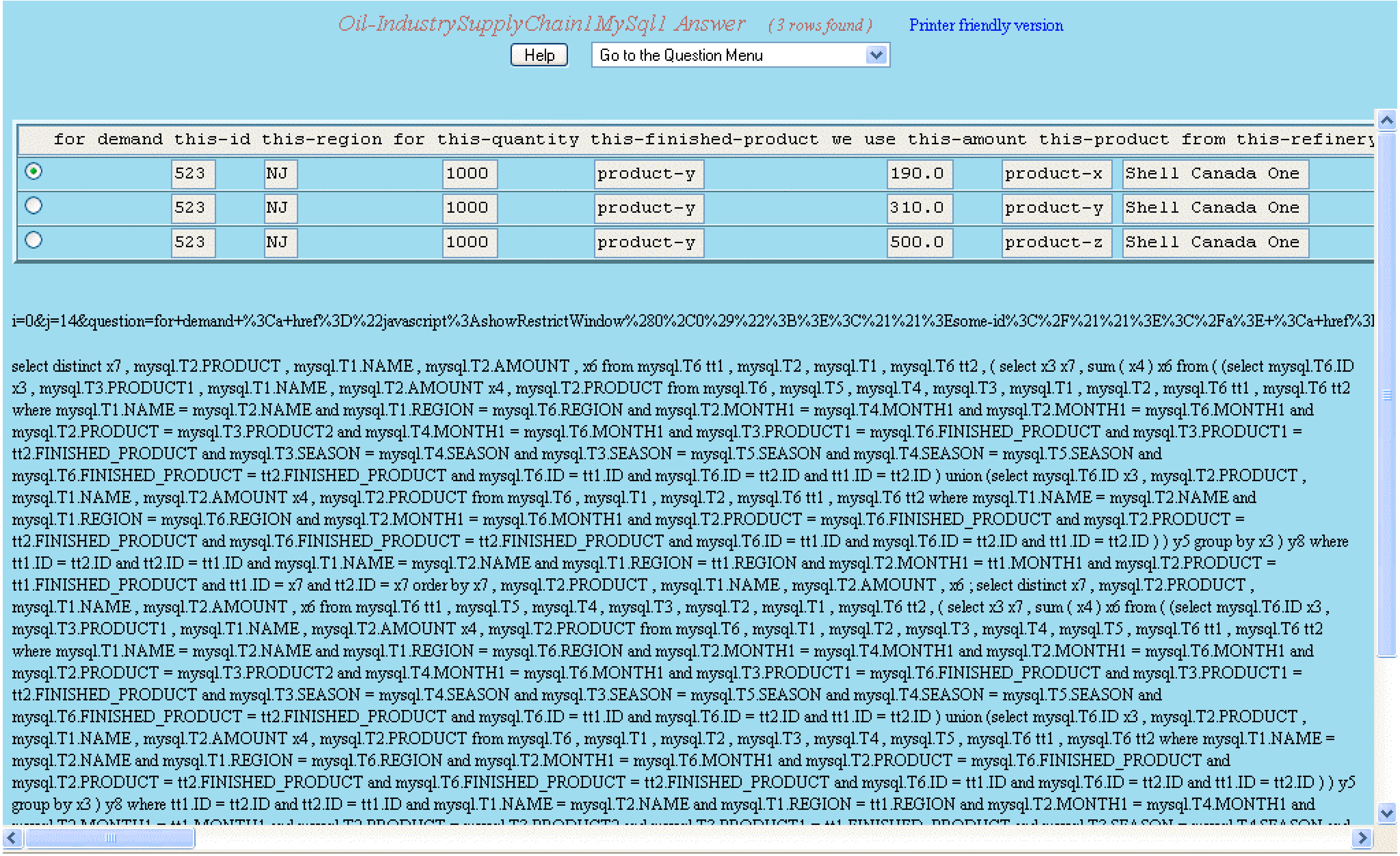
- By their nature, the comments would not be used during machine processing
  - Correctness of the hand written-SQL would rely on lengthy, and error prone manual verification

- Comments are sometimes not kept up to date with the code!

Agenda

- Enabling closer dialog between SOA stakeholders
  - Users, business analysts, enterprise architects, and system engineers
- A wiki for content in open vocabulary, executable English
  - Write applications in English, run them, and get English explanations
- Google indexes and retrieves content in executable English
  - Acts as a kind of registry
- The wiki engine as an SOA knowledge endpoint on the web
- A supply chain example
- Automatic generation of complex, distributed SQL
  - With English explanations of results
- Capturing generated SQL for re-use
- Summary

Capturing generated SQL for re-use



for demand	this-id	this-region	for this-quantity	this-finished-product	we use this-amount	this-product	from this-refinery
●	523	NJ	1000	product-y	190.0	product-x	Shell Canada One
○	523	NJ	1000	product-y	310.0	product-y	Shell Canada One
○	523	NJ	1000	product-y	500.0	product-z	Shell Canada One

```
i=0&j=14&question=for+demand+%3Ca+href%3D%22javascript%3AshowRestrictWindow%280%2C0%29%22%3B%3E%3C%21%21%3Esome-id%3C%2F%21%21%3E%3C%2Fa%3E+%3Ca+href%3
```

select distinct x7 , mysql.T2.PRODUCT , mysql.T1.NAME , mysql.T2.AMOUNT , x6 from mysql.T6 tt1 , mysql.T2 , mysql.T1 , mysql.T6 tt2 , (select x3 x7 , sum (x4) x6 from ((select mysql.T6.ID x3 , mysql.T3.PRODUCT1 , mysql.T1.NAME , mysql.T2.AMOUNT x4 , mysql.T2.PRODUCT from mysql.T6 , mysql.T5 , mysql.T4 , mysql.T3 , mysql.T1 , mysql.T2 , mysql.T6 tt1 , mysql.T6 tt2 where mysql.T1.NAME = mysql.T2.NAME and mysql.T1.REGION = mysql.T6.REGION and mysql.T2.MONTH1 = mysql.T4.MONTH1 and mysql.T2.MONTH1 = mysql.T6.MONTH1 and mysql.T2.PRODUCT = mysql.T3.PRODUCT2 and mysql.T4.MONTH1 = mysql.T6.MONTH1 and mysql.T3.PRODUCT1 = mysql.T6.FINISHED_PRODUCT and mysql.T3.PRODUCT1 = tt2.FINISHED_PRODUCT and mysql.T3.SEASON = mysql.T4.SEASON and mysql.T3.SEASON = mysql.T5.SEASON and mysql.T4.SEASON = mysql.T5.SEASON and mysql.T6.FINISHED_PRODUCT = tt2.FINISHED_PRODUCT and mysql.T6.ID = tt1.ID and mysql.T6.ID = tt2.ID and tt1.ID = tt2.ID) union (select mysql.T6.ID x3 , mysql.T2.PRODUCT , mysql.T1.NAME , mysql.T2.AMOUNT x4 , mysql.T2.PRODUCT from mysql.T6 , mysql.T1 , mysql.T2 , mysql.T6 tt1 , mysql.T6 tt2 where mysql.T1.NAME = mysql.T2.NAME and mysql.T1.REGION = mysql.T6.REGION and mysql.T2.MONTH1 = mysql.T6.MONTH1 and mysql.T2.PRODUCT = mysql.T6.FINISHED_PRODUCT and mysql.T2.PRODUCT = tt2.FINISHED_PRODUCT and mysql.T6.FINISHED_PRODUCT = tt2.FINISHED_PRODUCT and mysql.T6.ID = tt1.ID and mysql.T6.ID = tt2.ID and tt1.ID = tt2.ID)) y5 group by x3) y8 where tt1.ID = tt2.ID and tt2.ID = tt1.ID and mysql.T1.NAME = mysql.T2.NAME and mysql.T1.REGION = tt1.REGION and mysql.T2.MONTH1 = tt1.MONTH1 and mysql.T2.PRODUCT = tt1.FINISHED_PRODUCT and tt1.ID = x7 and tt2.ID = x7 order by x7 , mysql.T2.PRODUCT , mysql.T1.NAME , mysql.T2.AMOUNT , x6 ; select distinct x7 , mysql.T2.PRODUCT , mysql.T1.NAME , mysql.T2.AMOUNT , x6 from mysql.T6 tt1 , mysql.T5 , mysql.T4 , mysql.T3 , mysql.T2 , mysql.T1 , mysql.T6 tt2 , (select x3 x7 , sum (x4) x6 from ((select mysql.T6.ID x3 , mysql.T3.PRODUCT1 , mysql.T1.NAME , mysql.T2.AMOUNT x4 , mysql.T2.PRODUCT from mysql.T6 , mysql.T1 , mysql.T2 , mysql.T3 , mysql.T4 , mysql.T5 , mysql.T6 tt1 , mysql.T6 tt2 where mysql.T1.NAME = mysql.T2.NAME and mysql.T1.REGION = mysql.T6.REGION and mysql.T2.MONTH1 = mysql.T4.MONTH1 and mysql.T2.MONTH1 = mysql.T6.MONTH1 and mysql.T2.PRODUCT = mysql.T3.PRODUCT2 and mysql.T4.MONTH1 = mysql.T6.MONTH1 and mysql.T3.PRODUCT1 = mysql.T6.FINISHED_PRODUCT and mysql.T3.PRODUCT1 = tt2.FINISHED_PRODUCT and mysql.T3.SEASON = mysql.T4.SEASON and mysql.T3.SEASON = mysql.T5.SEASON and mysql.T4.SEASON = mysql.T5.SEASON and mysql.T6.FINISHED_PRODUCT = tt2.FINISHED_PRODUCT and mysql.T6.ID = tt1.ID and mysql.T6.ID = tt2.ID and tt1.ID = tt2.ID) union (select mysql.T6.ID x3 , mysql.T2.PRODUCT , mysql.T1.NAME , mysql.T2.AMOUNT x4 , mysql.T2.PRODUCT from mysql.T6 , mysql.T1 , mysql.T2 , mysql.T3 , mysql.T4 , mysql.T5 , mysql.T6 tt1 , mysql.T6 tt2 where mysql.T1.NAME = mysql.T2.NAME and mysql.T1.REGION = mysql.T6.REGION and mysql.T2.MONTH1 = mysql.T6.MONTH1 and mysql.T2.PRODUCT = mysql.T6.FINISHED_PRODUCT and mysql.T2.PRODUCT = tt2.FINISHED_PRODUCT and mysql.T6.FINISHED_PRODUCT = tt2.FINISHED_PRODUCT and mysql.T6.ID = tt1.ID and mysql.T6.ID = tt2.ID and tt1.ID = tt2.ID)) y5 group by x3) y8 where tt1.ID = tt2.ID and tt2.ID = tt1.ID and mysql.T1.NAME = mysql.T2.NAME and mysql.T1.REGION = tt1.REGION and mysql.T2.MONTH1 = mysql.T4.MONTH1 and

Please see FAQ #4 at www.reengineeringllc.com/internet_business_logic_FAQs.html

Summary

- Enabling closer dialog between SOA stakeholders
  - Users, business analysts, enterprise architects, and system engineers
- A wiki for content in open vocabulary, executable English
  - Write applications in English, run them, and get English explanations
- Google indexes and retrieves content in executable English
  - Acts as a kind of registry
- The wiki engine as an SOA knowledge endpoint on the web
- A supply chain example
- Automatic generation of complex, distributed SQL
  - With English explanations of results
- Capturing generated SQL for re-use

Links

1. Semantics2 -- what a reasoner **should** do:

   Backchain iteration: towards a practical inference method that is simple enough to be proved Terminating, sound and complete. Journal of automated reasoning, 11:1-22

2. The English inferencing examples

   Oil-industrysupplychain1

   Oil-industrysupplychain1mysql1

   (And many other examples provided) can be run, changed, and re-run as follows:

   1. Point internet explorer 6, netscape 7, firefox or mozilla to www.reengineeringllc.com
   2. Click on Internet Business Logic
   3. Click on the go button
   4. Click on the help button to see how to navigate through the pages
   5. Select *oil-industrysupplychain1 or oil-industrysupplychain1mysql1*

3. You are cordially invited to write and run your own examples. Shared use of the system is free

4. You can download the java client stub from http://www.Reengineeringllc.Com/iblclient1.Java