

# Why Semantics?

Toby Segaran

# Practical Semantics?



*because a flying pig would be too much*

# Ceci n'est pas

an endorsement of the:

one,

true,

capital-S,

Semantic Web

# Ceci n'est pas

*deep technical coverage of  
“standards”-based file-formats*

# Ceci n'est pas

*an effort to convince you that  
any particular way is always “best”*

# Ceci n'est pas

hypothetical

(i.e. there's money here)

# The Zeitgeist

## STEVEY'S BLOG RANTS

RANDOM WHINING AND STUFF.

MONDAY, OCTOBER 20, 2008

### The Universal Design Pattern

*This idea that there is generality in the specific is of far-reaching importance.*

— Douglas Hofstadter, *Gödel, Escher, Bach*



**e** Ola Bini said...

As someone else mentioned, a glaring hole in this article is that you don't mention RDF and SPARQL. If I needed to have this as external data instead of in the language, I would say RDF and SPARQL is the way to do it.

7:30 AM, OCTOBER 20, 2008

# The Zeitgeist

ReadWriteWeb   ReadWriteTalk   Enterprise   Jobwire



Home   Products   Trends   Best of RWW   Archives

## Is the Relational Database Doomed?

Written by [Tony Bain](#) / February 12, 2009 3:00 PM / [62 Comments](#)   « Prior Post   Next Post »



2. There is also a new crop of databases called "graph databases" gaining traction (with a model based on nodes, relationships and properties), one of them being the open-source neo4j (<http://neo4j.org>)

Using graphs to

Posted by: Mar

14. What is being discussed is scaling to volume. yes rdbms scales badly. however there is other scaling that is much greater problem : scaling to complexity. here rdbms is hopeless, vast number of tables have to be created and this is solidified upfront. also a problem with rdbms is everybody has to put their 2C worth and have a different way of storing the same things, rdbms does not help with standardization. thus millions of people toil developing their dbs all different no science of information emerges.

the future is in semantic databases and entirely different architectures. We at thoughtexpress have spent 10 years researching the problem of complexity and are convinced that rdbms is in fact the root cause of unnecessary complexity found in modern it systems.

Posted by: [pawel lubczonok](#) | February 12, 2009 7:59 PM

Traditional data-modeling

# Tabular Data

---

<b>Restaurant</b>	<b>Address</b>	<b>Cuisine</b>	<b>Price</b>	<b>Open</b>
Deli Llama	Peachtree Rd	Deli	\$	Mon, Tue, Wed, Thu, Fri
Peking Inn	Lake St	Chinese	\$\$\$	Thur, Fri, Sat
Thai Tonic	Branch Dr	Thai	\$\$	Tue, Wed, Thu, Fri, Sat, Sun
Lord of the Fries	Flower Ave	Fast food	\$\$	Tue, Wed, Thu, Fri, Sat, Sun
Marquis de Salade	Main St	French	\$\$\$	Thur, Fri, Sat
Wok this way	Second St	Chinese	\$	Mon, Tue, Wed, Thu, Fri, Sat, Sun
Luna Sea	Autumn Dr	Seafood	\$\$\$	Tue, Thu, Fri, Sat
Pita Pan	Thunder Rd	Middle Eastern	\$\$	Mon, Tue, Wed, Thu, Fri, Sat, Sun
Award Weiners	Dorfold Mews	Fast food	\$	Mon, Tue, Wed, Thu, Fri, Sat
Lettuce Eat	Rustic Parkway	Deli	\$\$	Mon, Tue, Wed, Thu, Fri

*The beloved spreadsheet*

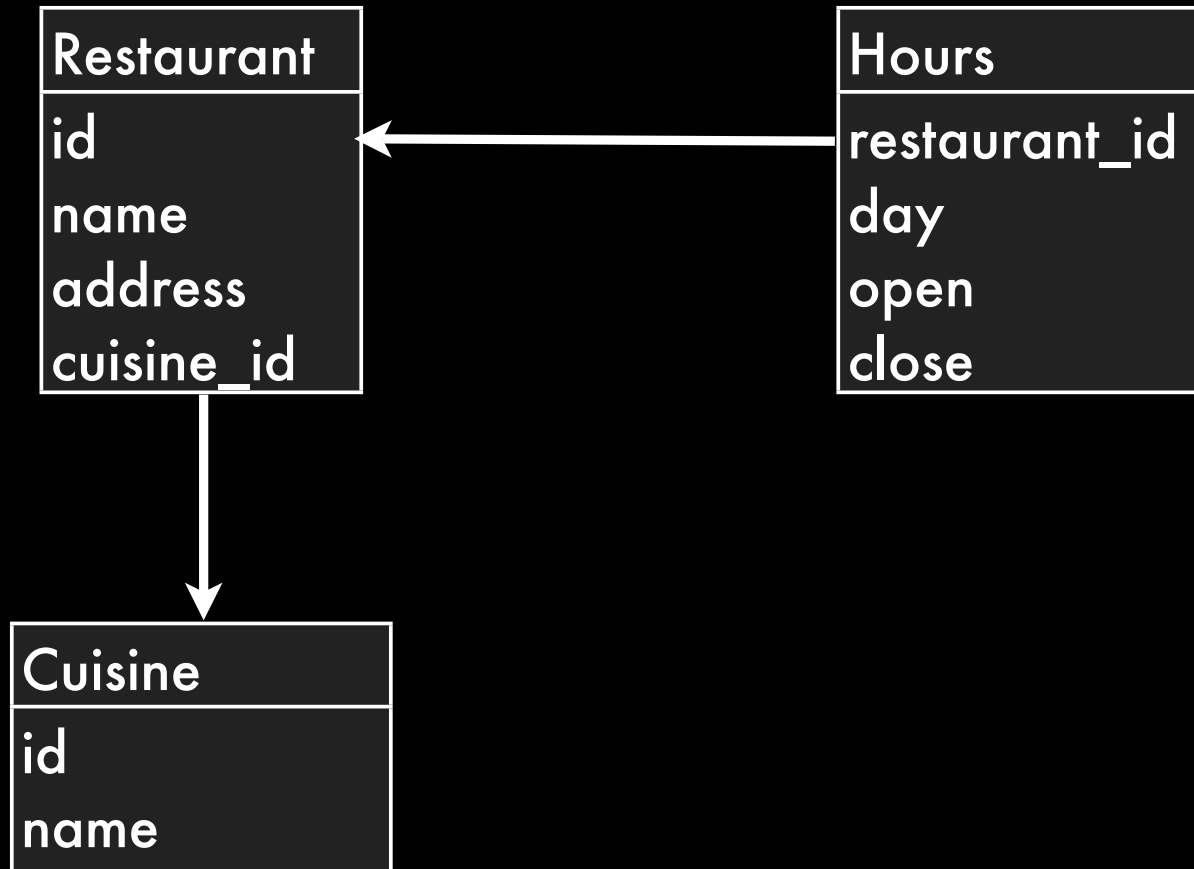
# Tabular Data

---

Restaurant	Address	Cuisine	Price	Open
Deli Llama	Peachtree Rd	Deli	\$	Mon (11a-4p), Tue (11-4), Wed (11-4), Thu (11-7), Fri (11-8)
Peking Inn	Lake St	Chinese	\$\$\$	Thur (5p-10p), Fri (5p-1a), Sat (5p-1a)
etc...				

Too much information, not enough cells

# A simple schema



*Allows for simple queries*

# A simple schema

<b>id</b>	<b>name</b>	<b>address</b>	<b>price</b>
1	Deli Llama	Peachtree	\$
2	Peking Inn	Lake St	\$\$\$
	...		

<b>restaurant_id</b>	<b>day</b>	<b>open</b>	<b>close</b>
1	Mon	11	16
1	Tue	11	16
1	Thu	11	19
2	Fri	5	23
...			

*Filled with data*

# Some new data...

<b>Bar</b>	<b>Address</b>	<b>DJ</b>	<b>Best Drink</b>
The Bitter End	14th Ave	No	Beer
Peking Inn	Lake St	No	Scorpion Bowl
Hammer Time	Wildcat Dr	Yes	Hennessey
Marquis de Salade	Main St	Yes	Martini

*This doesn't fit into our schema...*

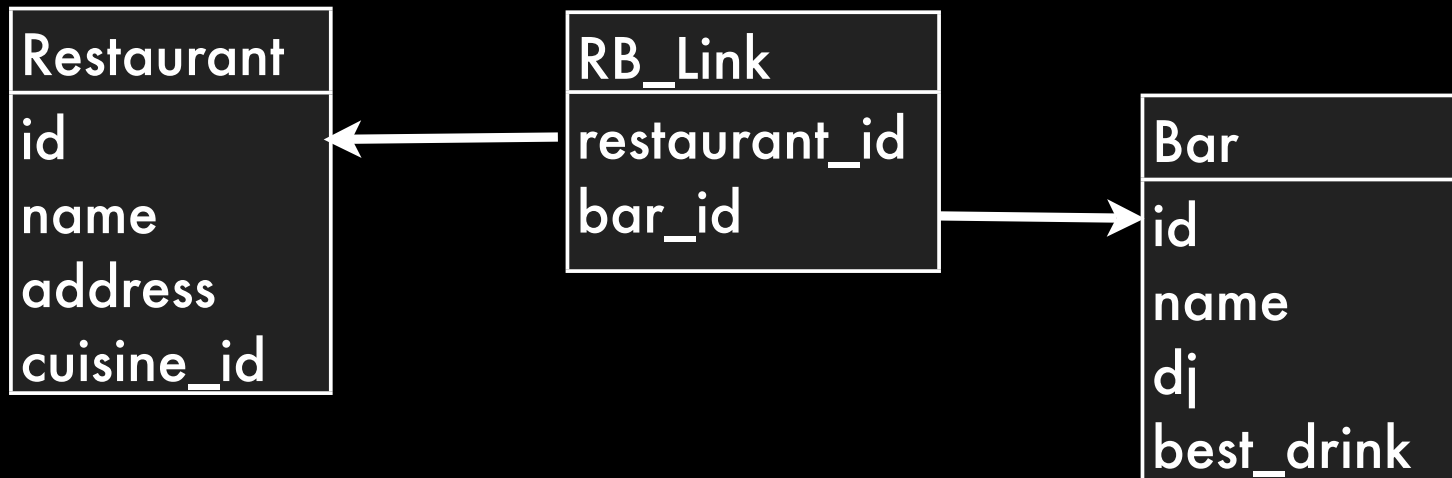
# Half empty columns?

<b>Restaurant</b>	<b>Address</b>	<b>Price</b>	<b>DJ</b>	<b>Best Drink</b>
Deli Llama	Peachtree Rd	\$		
Peking Inn	Lake St	\$\$\$	No	Scorpion Bowl
Thai Tonic	Branch Dr	\$\$		
Lord of the Fries	Flower Ave	\$\$		
Marquis de Salade	Main St	\$\$\$	Yes	Martini
Wok this way	Second St	\$		
Luna Sea	Autumn Dr	\$\$\$		
Pita Pan	Thunder Rd	\$\$		
Award Weiners	Dorfold Mews	\$		
Lettuce Eat	Rustic Parkway	\$\$		
Hammer Time	Wildcat Dr		Yes	Hennessey
The Bitter End	14th St		No	Beer

Maybe ok now, but can't this keep happening?

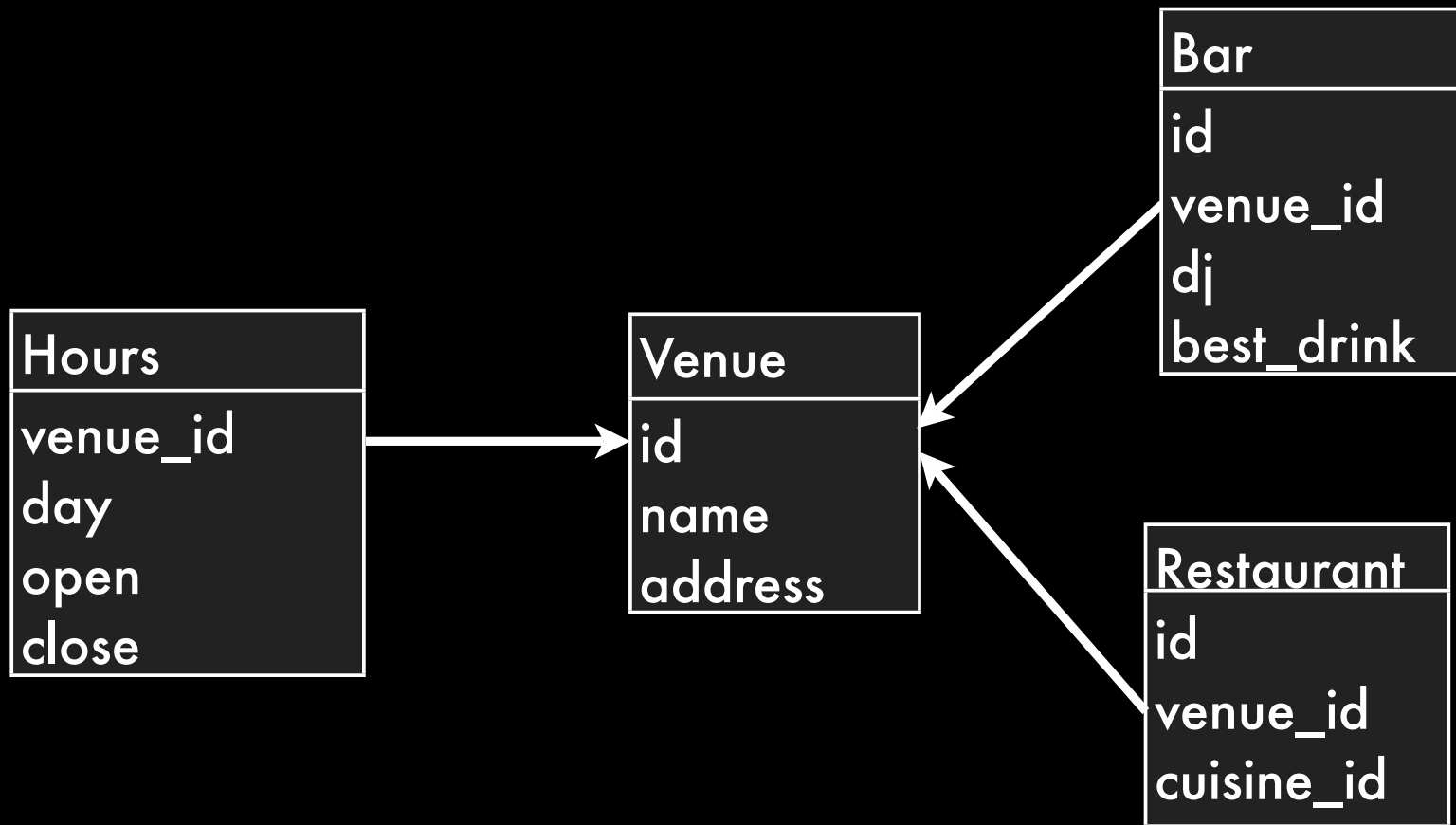
# Link the tables?

---



But now the information is duplicated :(

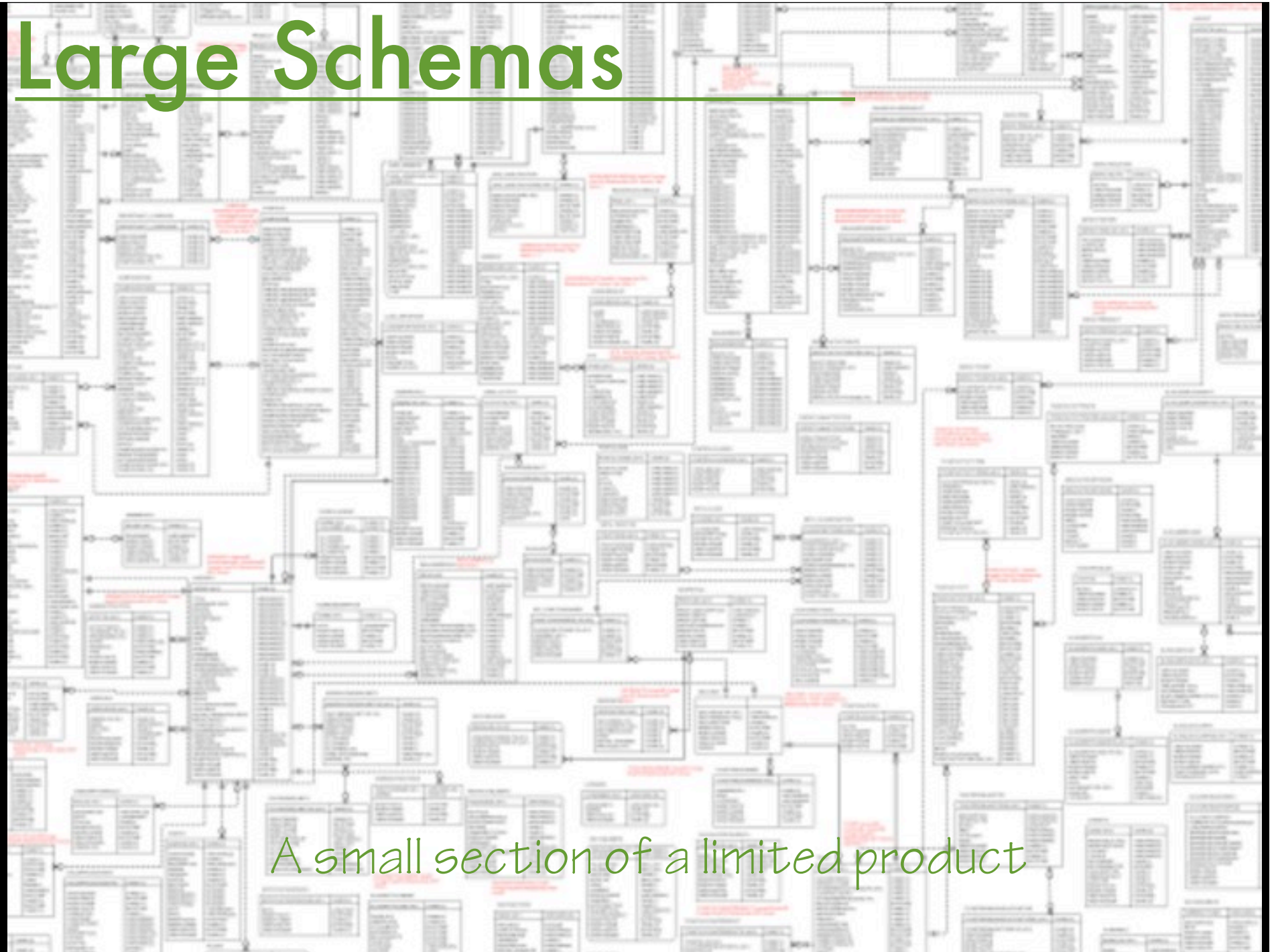
# Split place / purpose



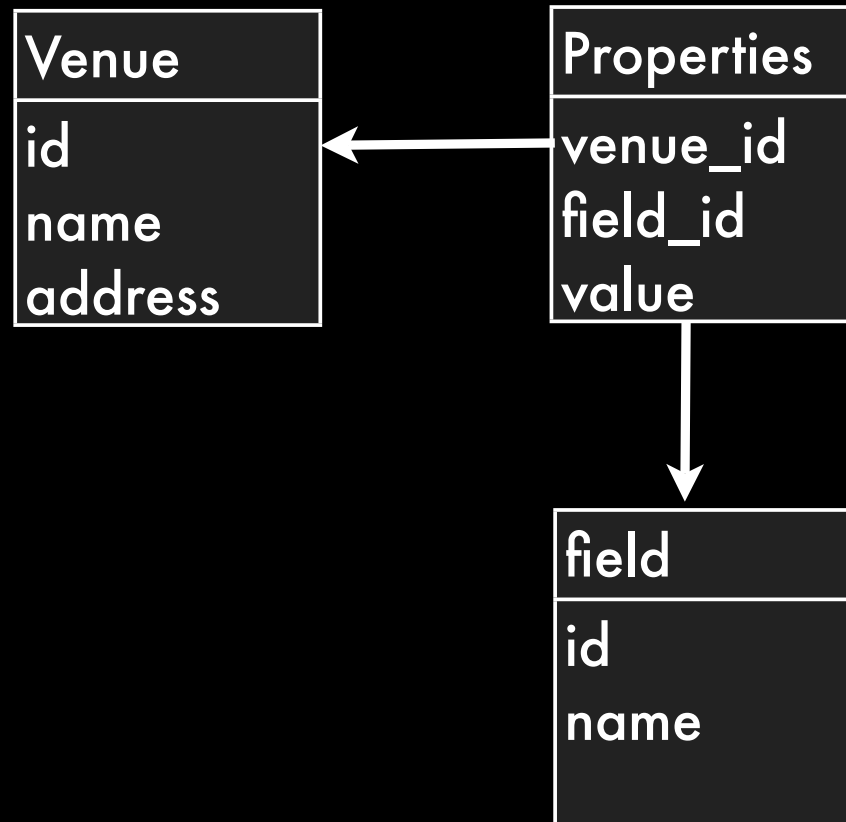
*Better, but now we have to “migrate”*

# Large Schemas

*A small section of a limited product*



# A flexible schema?



Does this look familiar?

# Add some data

id	name	address
1	Deli Llama	Peachtree Rd
2	Peking Inn	Lake St
	...	

venue_id	field_id	value
1	1	Deli
1	2	\$
2	1	Chinese
2	2	\$\$\$
2	3	Scorpion Bowl
2	4	No

id	name
1	Cuisine
2	Price
3	Specialty Cocktail
4	DJ?

simple enough...

# Add live music info

id	name	address
1	Deli Llama	Peachtree Rd
2	Peking Inn	Lake St
3	Thai Tonic	Branch Dr

venue_id	field_id	value
1	1	Deli
1	2	\$
2	1	Chinese
2	2	\$\$\$
2	3	Scorpion Bowl
2	4	No
3	5	Yes
3	6	Jazz

id	name
1	Cuisine
2	Price
3	Specialty Cocktail
4	DJ?
5	Live Music
6	Music Genre

*No schema change required*

Explicit semantics

# The basic data unit

subject → predicate → object

Remember this from grammar class?

# Restaurants as triples

<b>subject</b>	<b>predicate</b>	<b>object</b>
S1	cuisine	"Deli"
S1	price	"\$"
S1	name	"Deli Llama"
S2	cuisine	"Chinese"
S2	price	"\$"
S2	name	"Peking Inn"
S2	best drink	"Scorpion Bowl"
S2	address	"Lake St"
S2	DJ?	"No"
S4	name	"Fendalton"
S4	contained-by	S5
S5	name	"Christchurch"
S1	location	S4
S6	name	"Downtown"
S6	contained-by	S7
S7	name	"Wellington, NZ"
S2	location	S6

Machine readable and almost human readable

# ...or as a graph



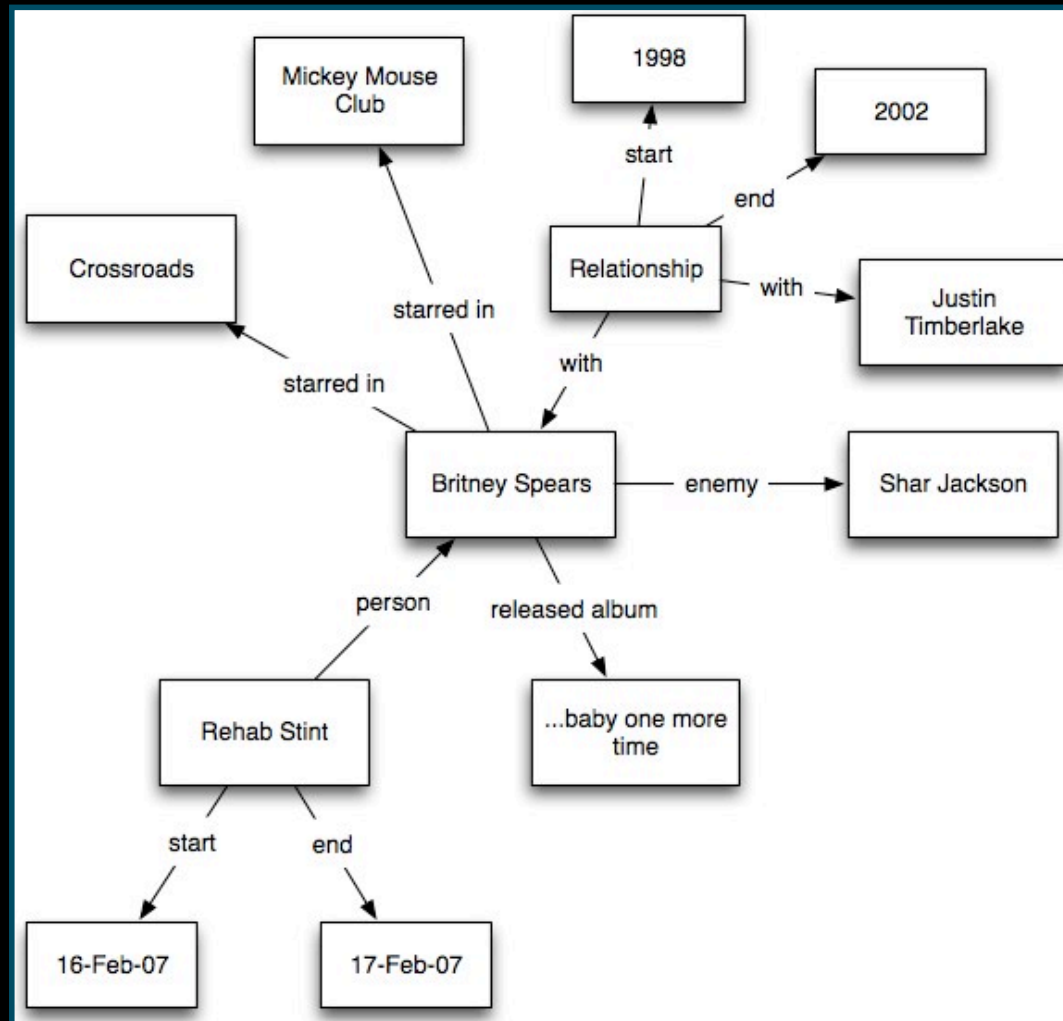
An intuitive way of looking at semantic data

# Returning to the table

<b>Restaurant</b>	<b>Address</b>	<b>Cuisine</b>	<b>Price</b>	<b>Open</b>
Deli Llama	Peachtree Rd	Deli	\$	Mon, Tue, Wed, Thu, Fri
Peking Inn	Lake St	Chinese	\$\$\$	Thur, Fri, Sat
Thai Tanic	Branch Dr	Thai	\$\$	Tue, Wed, Thu, Fri, Sat, Sun
Lord of the Fries	Flower Ave	Fast food	\$\$	Tue, Wed, Thu, Fri, Sat, Sun
Marquis de Salade	Main St	French	\$\$\$	Thur, Fri, Sat
Wok this way	Second St	Chinese	\$	Mon, Tue, Wed, Thu, Fri, Sat, Sun
Luna Sea	Autumn Dr	Seafood	\$\$\$	Tue, Thu, Fri, Sat
Pita Pan	Thunder Rd	Middle Eastern	\$\$	Mon, Tue, Wed, Thu, Fri, Sat, Sun
Award Weiners	Dorfold Mews	Fast food	\$	Mon, Tue, Wed, Thu, Fri, Sat
Lettuce Eat	Rustic Parkway	Deli	\$\$	Mon, Tue, Wed, Thu, Fri

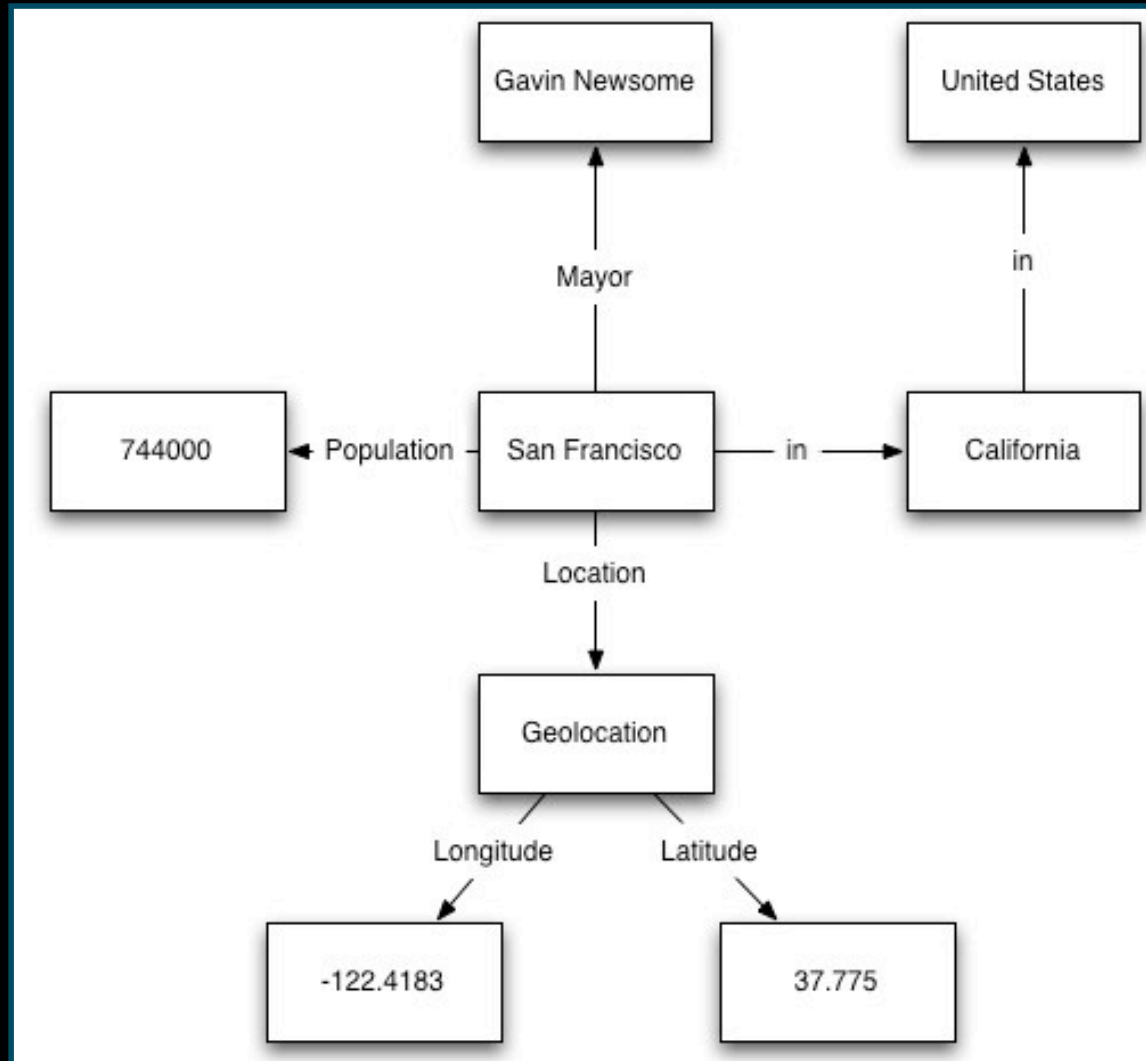
*Do you see the semantics?*

# Celebrities



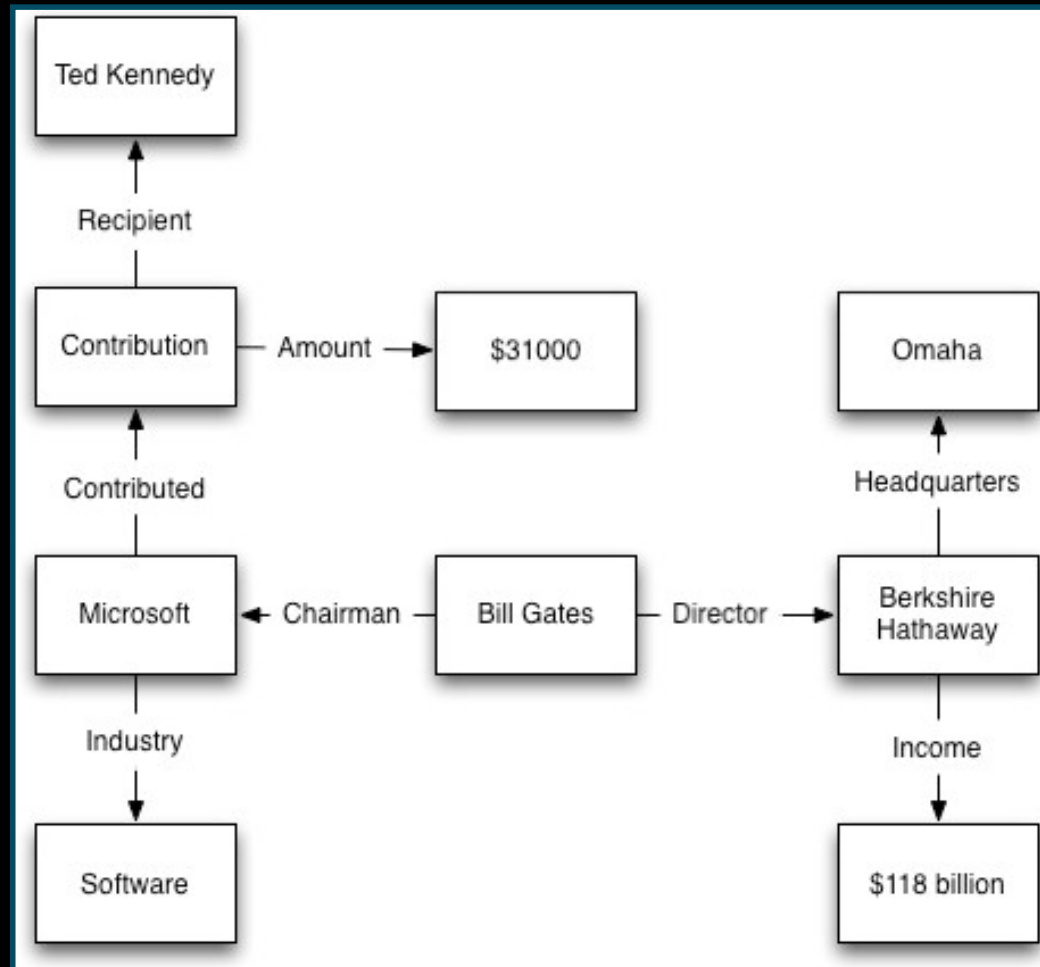
*So much to say about Britney*

# Places



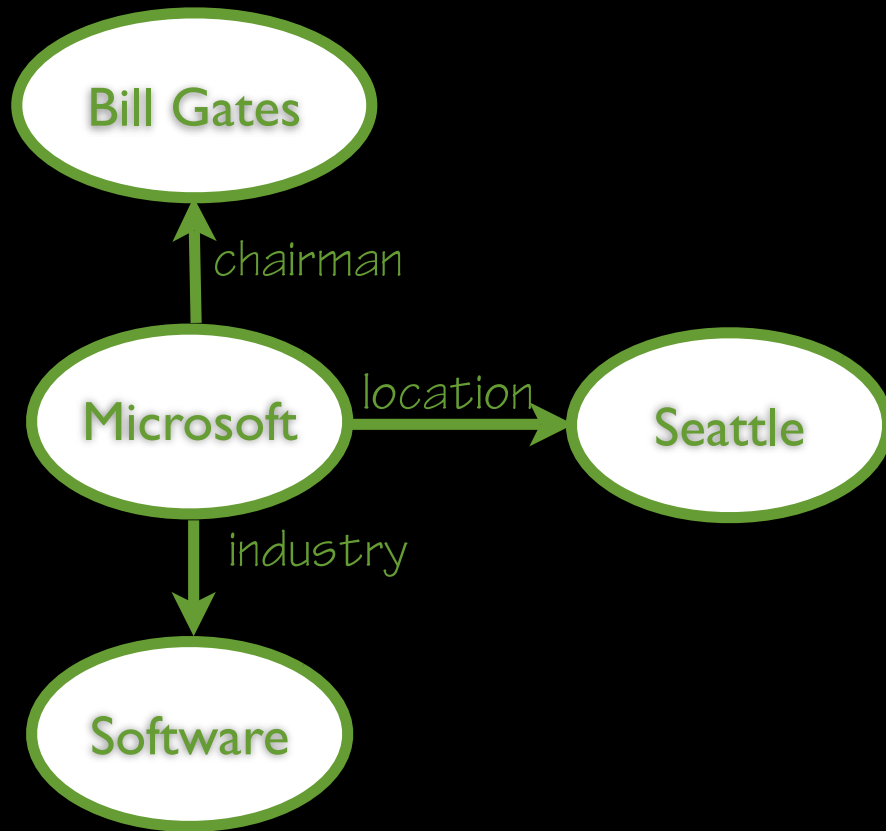
*Location and demographics*

# Companies



Political contributions, and business data

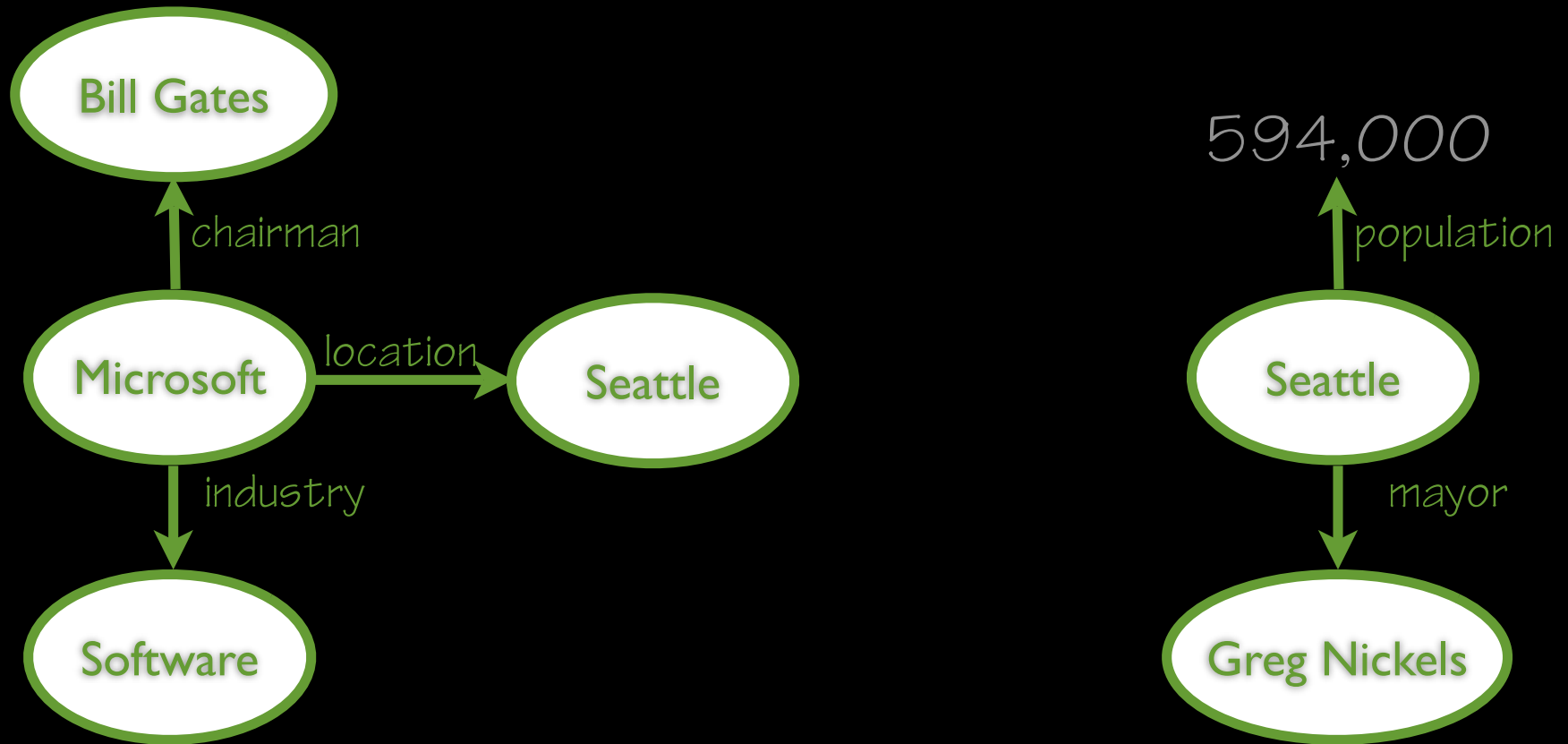
# Easy merging



*Say you've got a business data set*

# Easy merging

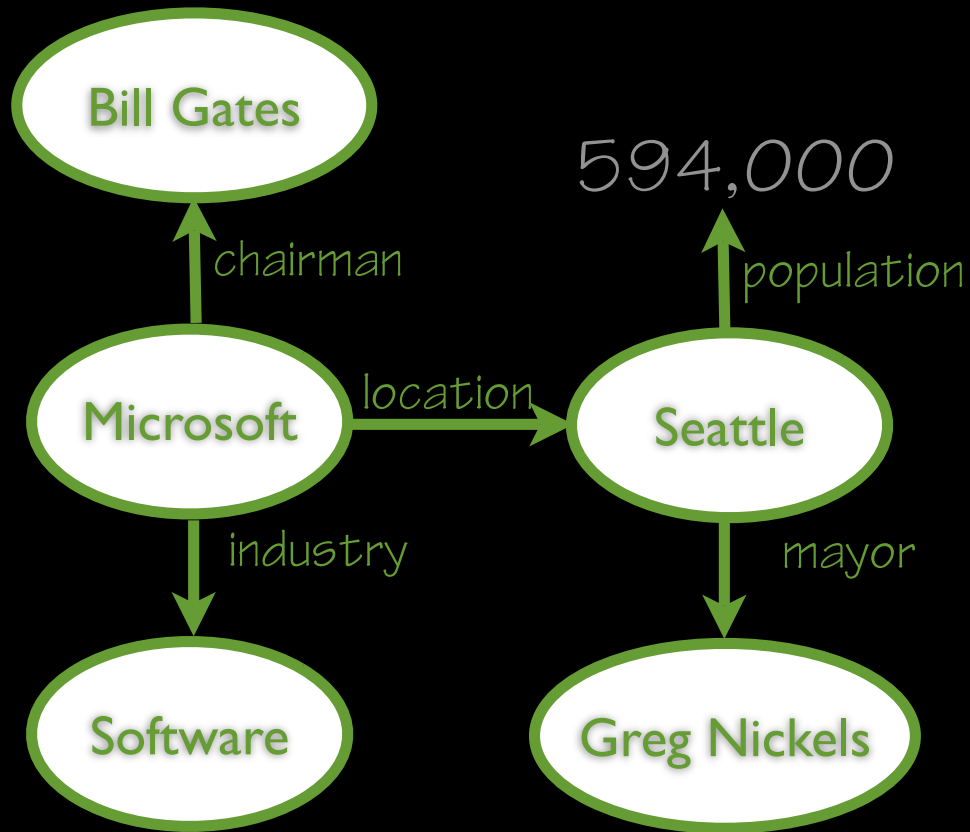
---



*And then you manage to get a places data set*

# Easy merging

---



*Merge the nodes, no schema changes necessary*

# Designing for change

**django**



 **Pylons**

*We believe this for code, why not data?*

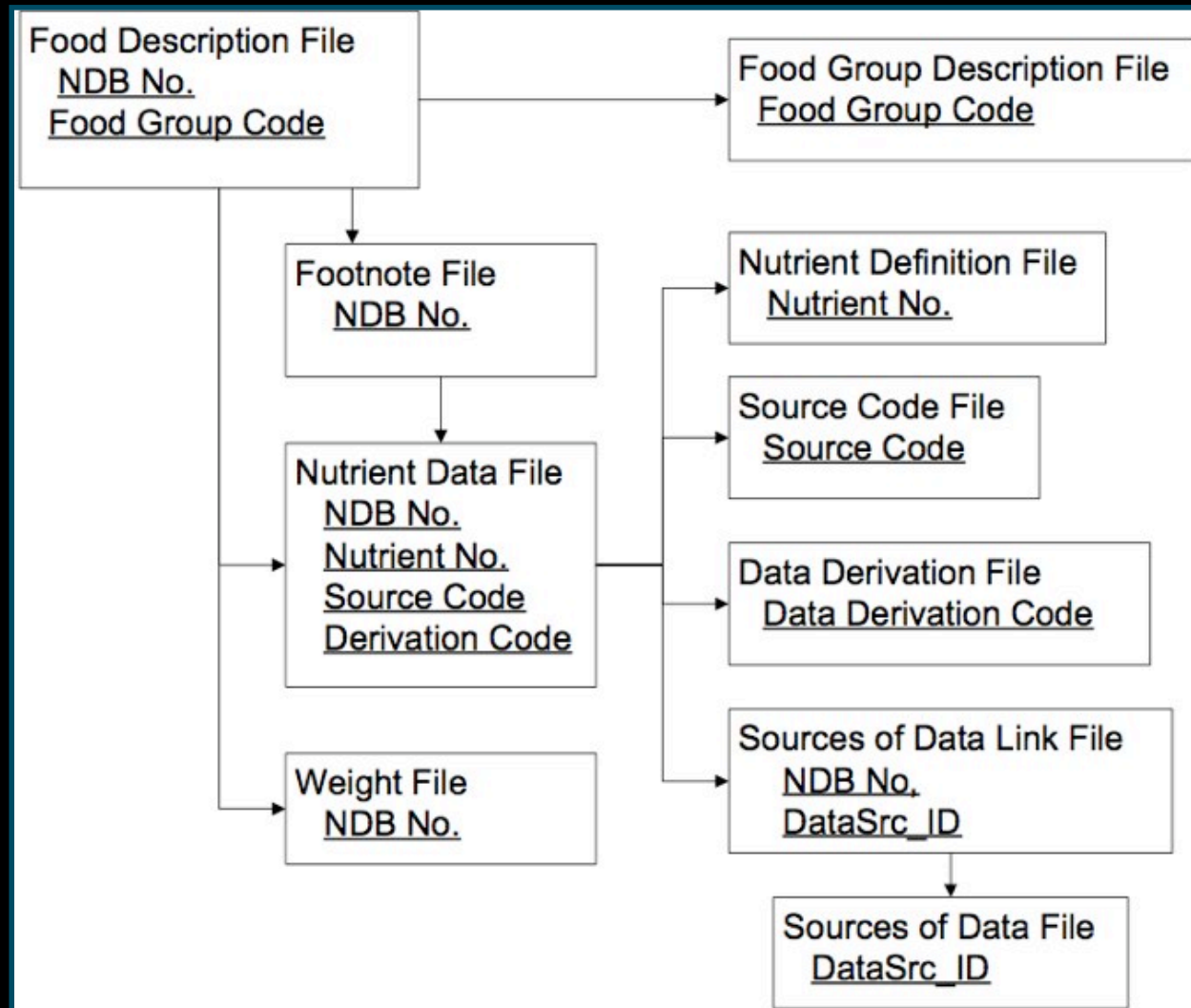
Data is for sharing

# Data is for sharing

12secondstv 23 30Boxes 411Sync 43Things A9 AdobeOnAir  
AdobeShare AevumObscurum aideRSS AIM AIMPhoneline Alexa  
AlexaThumbnail AlexaTopSites AlexaWebInfo Amazon  
AmazonEC2 AmazonFulfillmentWebService AmazonPayments  
AmazonQueue AmazonS3 AmazonSimpleDB AMEE AOLOpenAuth  
AOLOpenMail AOLPictures AOLVideo AOLWebAIM  
AonawareDictionary ArcWeb Arkayne ArtBeat Authorize.Net Backpack  
BackType Basecamp BBC Bebo BeenVerified BigTribe bLaugh  
Blinksale Blip.tv Blipfm Blogger Bloglines BlueDot BookMooch Box.net  
Brightkite BroadsoftXtended BTWeb21C buySAFE Buzznet CafePress  
Calais CapitolWords Carma ClearForest Clickatell Clicky CNET  
COLOURlovers CommissionJunction Compete CreativeCommons  
CrunchBase DabbleDB Dapper DataUnisonEbayResearch Daylife  
DBCreditCheck DBpedia deCarta del.icio.us DemocracyInAction DHL  
Digg DigitalPodcast Dipity Discogs Disqus Dooblet dopplr Dropio EarthTools  
EasyUtil eBay EchoNest EditGrid Entrez eSideWalk ESRIArcGISJavaScript  
ESV Eventful EveOnline Evernote Evoca ExactTarget Facebook  
Faces Faviconz FedEx FedSpending FeedBurner FeedMap ffwf FilesTube  
Findory FireEagle Flickr FollowTheMoney FormSpring Fotolia  
Freebase Freewheel FreeYourID FreshBooks FriendFeed  
FUTEFWikipedia GamesRadar Garmin geocoder  
geocoderCanada GeolQ GeoNames GetMapping Gigablast  
GlobeXplorer Google GoogleAdSense GoogleAdWords  
GoogleAjaxFeeds GoogleAJAXLanguage  
GoogleAJAXLibraries GoogleAjaxSearch  
GoogleAppEngine GoogleAppsEmailMigration GoogleBase  
GoogleBookSearchBookViewability GoogleCalendar GoogleChart  
GoogleCheckout GoogleClientAuth GoogleCodeSearch GoogleContacts  
GoogleCustomSearch GoogleDesktop GoogleDocsList  
GoogleEarth GoogleFinancePortfolio GoogleGears GoogleGearsGeolocation  
GoogleHealth GoogleHomepage GoogleMaps  
GoogleMapsFlash GoogleMashupEditor GoogleNotebook  
GooglePicasa GoogleSafeBrowsing GoogleSocialGraph  
GoogleSpreadsheets GoogleSSO GoogleStaticMaps  
GoogleTalk GoogleTranslate GoogleVisualization GoogleWebAuth  
GoogleWebmasterTools GrouperVideo Gruvr Harvest Hi5 Highrise Hiogi  
HolidayService hon.jp hostip HotelsCombined HotOrNot Identica  
iGoogleThemes IIGAlerts ILike ImageLoop Imeem indeed  
InnerGearsCityState Interfax InternetArchive InternetVideoArchive Jaduka  
Jaiku JamBase Jots JotSpot Kayak Kuler LastFM Lat49 LibraryThing  
Lingr LinkedIn LinkShare LiveContactsJS LiveJournal  
LiveSpacesPhotoControl LiveVideo LivingStones LOCSRW LuluPublishing  
Lyricsfly LyricWiki Ma.gnolia MacromediaNews MailChimp Map24  
Maplight MapPoint MapQuest Mapstraction MaxMindGeoIP  
MechanicalTurk Mediod MetaCarta MicrosoftMSDN MindTouchDekiWiki  
Mobivity MOO Moreover MSNMessenger MTV Multimap  
MusicBrainz MusicMobs MyBlogLog Myplaylist MySpace  
MyVox NASA Naver NCAAMensCollegeBasketball Nestoria Netflix  
Netvibes NewsCloud NewsGator NewsIsFree NewYorkTimesBestSellers  
NewYorkTimesCampaignFinance NewYorkTimesCommunity  
NewYorkTimesCongress NewYorkTimesTimesPeople Ning NOAA Notifixious  
NPR Omidrive Ontok OntokWikipedia Oodle openDada OpenDOAR  
OpenLayers OpenPatentServices OpenSecrets OpenSocial OpenStrands  
OpenStreetMap Orb Orkut OSOpenSpace OutsideIn Pageflakes Panoramio  
PayPal Peekacity Phanfare PhishTank Photobucket Pingdom Pixagogo  
Plaxo Plazes Poly9FreeEarth Pownce PriceGrabber PriceRunner  
Prodigem Prosper PubSub Pushpin PushpinLocationData QuarkRank Rakuten  
RapLeaf RawSugar Rentometer ReutersSpotlight Revver  
Rhapsody Ribbit Ringful Riya Rove Salesforce Scribd ScribdPaper  
SecondLife SeeqPod Shadows SharedBook

The API flood

# Example: USDA



*Many separate text files, implied semantics*

# Example: Geonames

<b>Subject</b>	<b>Predicate</b>	<b>Object</b>
/3017382	Name	“France”
/3017382	wikipediaArticle	<a href="http://fr.wikipedia.org/wiki/France">http://fr.wikipedia.org/wiki/France</a>
/3017382	population	64094000
/3030967	Name	“Bourgogne”
/3030967	parentFeature	/3017382
/3038033	Name	“Alsace”
/3038033	parentFeature	/3017382

*Explicit semantics, no need to interpret structure*

# Serialization

---

```
@prefix geo: <http://www.geonames.org/ontology#>.
```

```
<http://sws.geonames.org/3017382/>
```

```
  geo:name "France";  
  geo:population 64094000.
```

```
<http://sws.geonames.org/3038033/>
```

```
  geo:name "Alsace";  
  geo:parentFeature <http://sws.geonames.org/3017382/>.
```

*The N3 format, intended to be easy to read*

# Namespaces

```
@prefix geo: <http://www.geonames.org/ontology#> .
```

```
@prefix rdf: <http://www.w3.org/1999/02/22-rdf-syntax-ns#> .
```

```
@prefix dc: <http://purl.org/dc/elements/1.1/> .
```

*How we know if we're using the same verbs*

# Namespaces

---

Vocabulary prefix	Vocabulary name	Domain	Documentation
dc	Dublin Core	Document metadata	See <a href="#">documentation on the web.</a>
foaf	Friend-Of-A-Friend	Personal profiles and social networks	See <a href="#">included documentation.</a>
vcard	VCard	Personal and business addresses	See <a href="#">included documentation.</a>
vcal	VCalendar	Events and other calendar items	See <a href="#">included documentation.</a>
review	hReview	Reviews	See <a href="#">included documentation.</a>
sioc	SIOC	Blogs, discussion forums, Q&A sites	See <a href="#">included documentation.</a>
gr	GoodRelations	Product price specification, delivery and payment etc.	See <a href="#">included documentation.</a>
dbpedia	DBPedia	Generic vocabulary	See <a href="#">documentation on the web.</a>
fb	Freebase	Generic vocabulary	See <a href="#">documentation on the web.</a>

*A short list, accepted by Yahoo BOSS*

# Strong keys



*Seattle, Seattle and Seattle*

# URIs and sameAs

---

<<http://sws.geonames.org/3017382/>> = France

<<http://sws.geonames.org/3038033/>> = Alsace

<<http://dbpedia.org/page/France>> = France

<<http://rdf.freebase.com/rdf/en.france>> = France

*unambiguous and shareable*

# URIs and sameAs

`<http://sws.geonames.org/3017382/> = France`

`<http://sws.geonames.org/3038033/> = Alsace`

`<http://dbpedia.org/page/France> = France`

`<http://rdf.freebase.com/rdf/en.france> = France`

`@prefix geo: <http://www.geonames.org/ontology#>.`

`@prefix owl: <http://www.w3.org/2002/07/owl#>.`

`<http://sws.geonames.org/3017382/>`

`geo:name "France";`

`geo:population 64094000;`

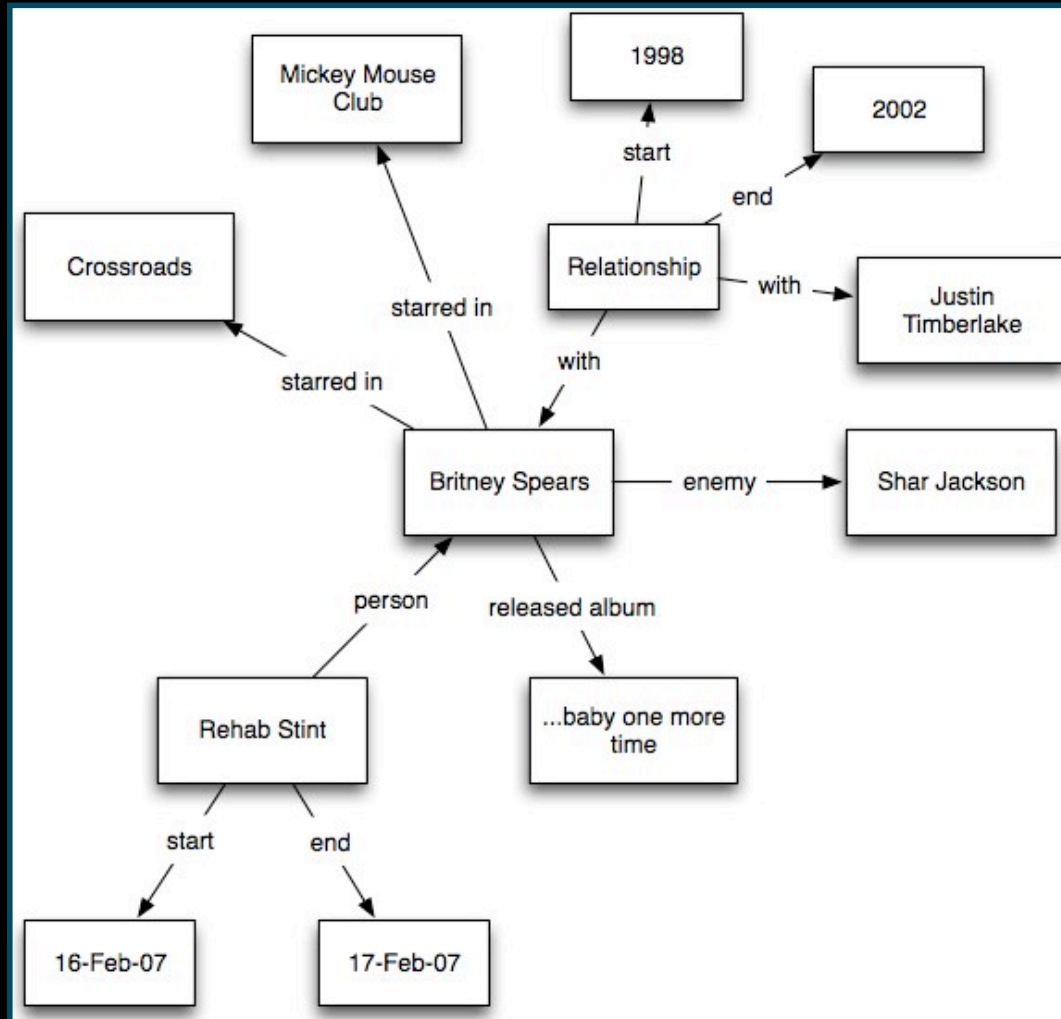
`owl:sameAs <http://dbpedia.org/page/France>;`

`owl:sameAs <http://rdf.freebase.com/rdf/en.france>.`

*the linking data way*

Querying and searching

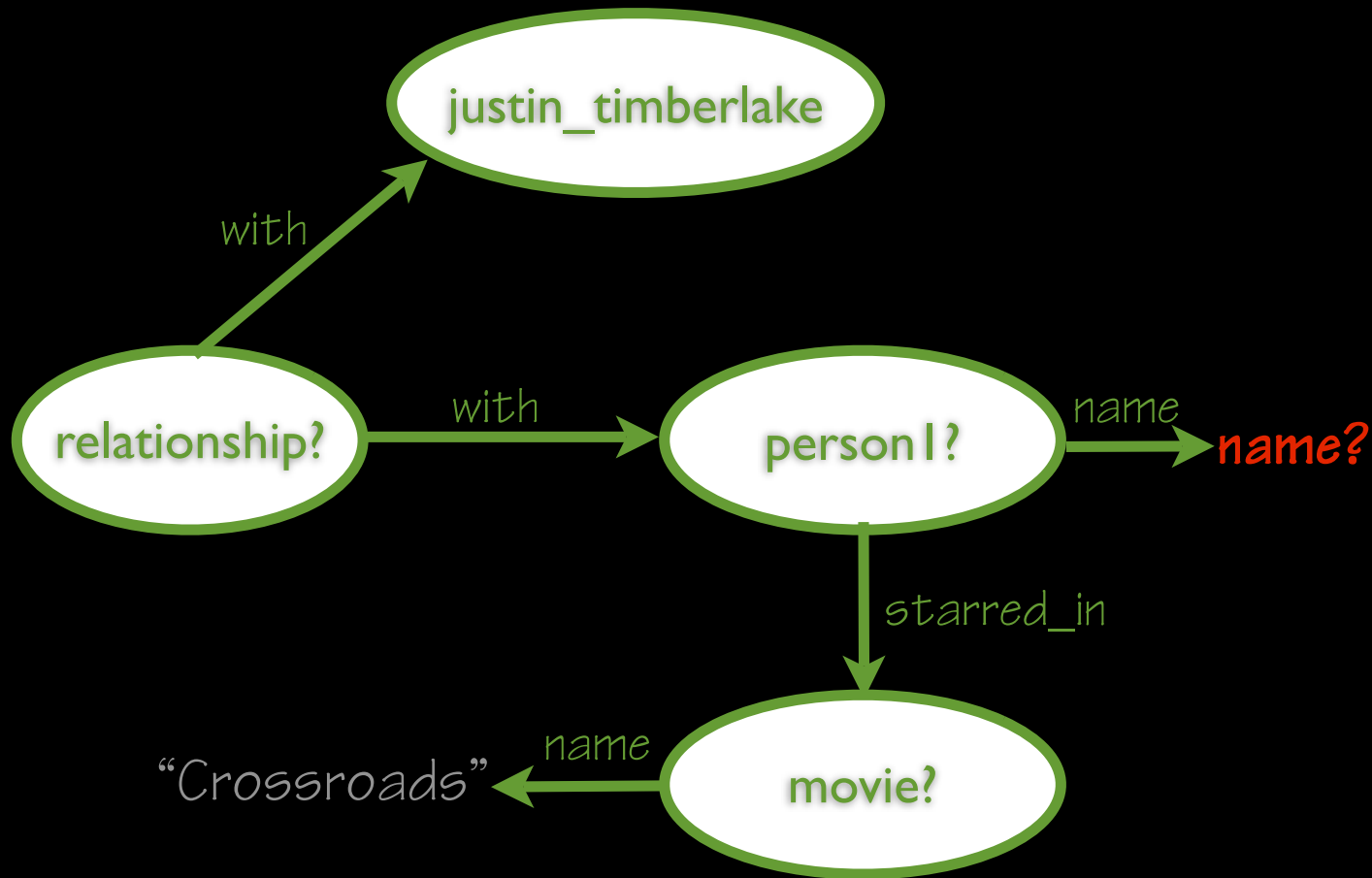
# Querying



*back to our celebrity graph...*

# Querying

---



*a pattern query graph*

# SPARQL

---

```
select ?name
where {?person1 name ?name .
      ?relationship with ?person1 .
      ?relationship with justin_timberlake .
      ?person1 starred_in ?movie .
      ?movie name "Crossroads"}
```

Look, no tables!

# SPARQL

---

```
select ?name
where {?person1 name ?name .
      ?relationship with ?person1 .
      ?relationship with justin_timberlake .
      ?person1 starred_in ?movie .
      ?movie name "Crossroads" .
      ?relationship start ?start .
      FILTER (?start > 1999) }
```

*adding additional filters*

# SPARQL

---

```
select ?name ?role
where {?person name ?name .
      ?person1 ?role ?movie .
      ?movie name "Crossroads"}
```

*predicates can be variables too*

# SPARQL Results

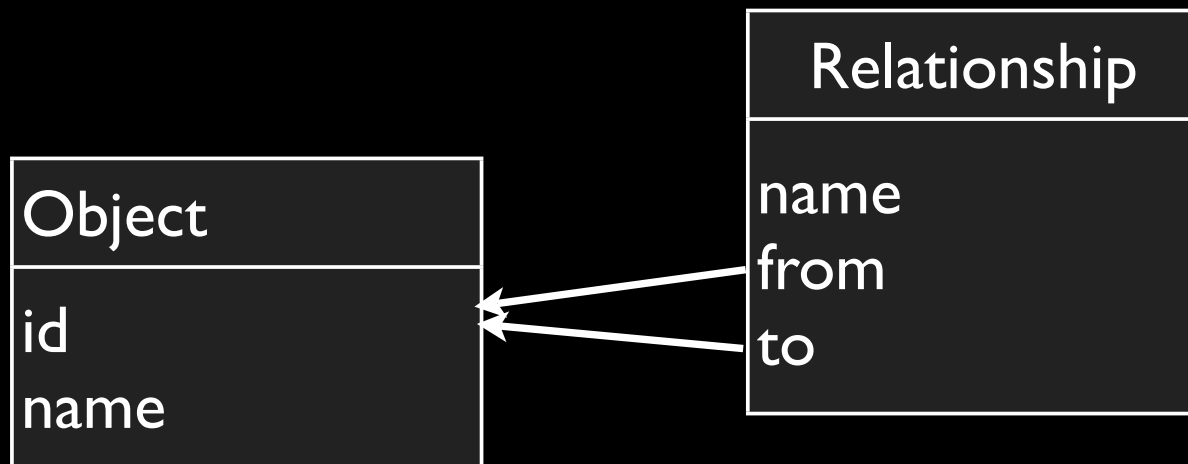
---

```
{"name": "Britney Spears", "role": "starred_in"},  
{"name": "Tamra Davis", "role": "directed"},  
etc...
```

*sets of bindings*

Implementation?

# The silly compromise



*a BAD use of an RDBMS*

# Sesame

---

**New Repository**

Type:

ID:

Title:

*in-memory, native, or stored in an DB*

# Sesame

## Add RDF

Base URI:

Context:

Data format:

Location of the RDF data you wish to upload

RDF Data URL:

Select the file containing the RDF data you wish to upload

RDF Data File:

Enter the RDF data you wish to upload

RDF Content:

RDF/XML or Turtle, right out of the box

# Sesame

## Query Repository

Query

Language:

SPARQL

```
PREFIX fb:<http://rdf.freebase.com/ns/>
PREFIX dc:<http://purl.org/dc/elements/1.1/>
PREFIX rdf:<http://www.w3.org/1999/02/22-rdf-syntax-ns#>

SELECT ?fn WHERE {?film fb:film.film.performances ?perf .
                   ?perf fb:film.performance.actor ?act .
                   ?act dc:title "John Malkovich".
                   ?film dc:title ?fn . }
```

Query:

Limit

results:

100

Include inferred statements

Execute

SPARQL queries

# Sesame

---

## Query Result (2)

Limit results:

100



Fn

"GARDENS OF THE NIGHT"

"CHANGELING"

*Sets of bindings*

# Exhibit

## 63 MIT-related Nobel Prize Winners


The information within this page has been retrieved from [this MIT official source](#) while the thumbnails are included from [Nobelprize.org](#). Here is the [Exhibit JSON data file](#).

Discipline	Relationship	Shared?	Deceased?
12 Chemistry	1 (missing this field)	15 no	47 (missing this field)
13 Economics	25 alumni	48 yes	1 no
9 Medicine/Physiology	25 professor		15 yes
2 Peace	15 research		
27 Physics	1 staff		


THUMBNAILS • DETAILS • TIMELINE

63 Nobelists


sorted by: [last-name](#); [then by...](#) •  grouped as sorted



George A. Akerlof  
Economics, 2001



Sidney Altman  
Chemistry, 1989



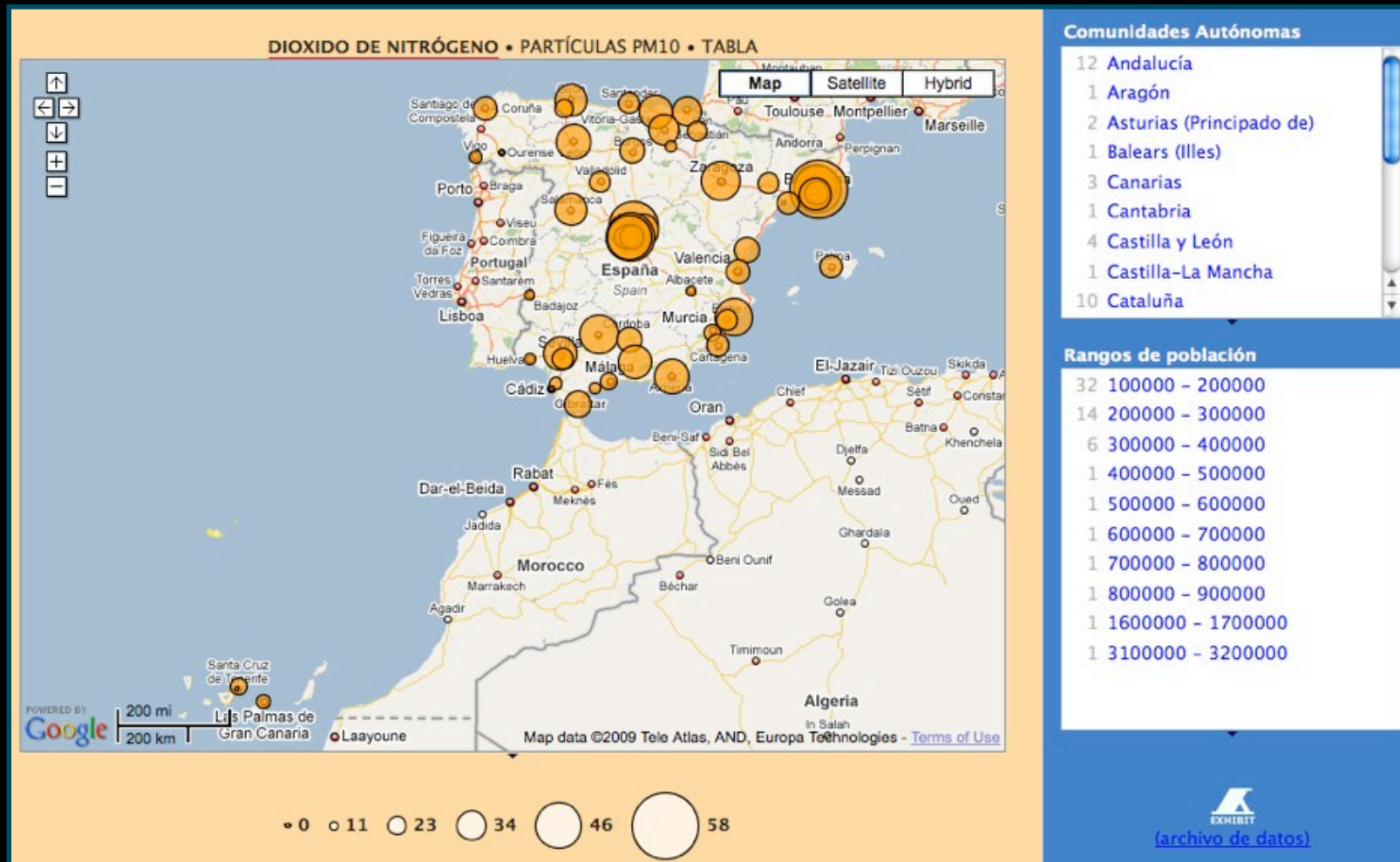
Luis W. Alvarez  
Physics, 1968



Kofi Annan  
Peace, 2001

Pure client-side faceted browsing

# Exhibit



Pure client-side faceted browsing

# Data sources



# FOAF

---

```
<rdf:RDF xmlns:hi5="http://api.hi5.com/"
  xmlns:rdf="http://www.w3.org/1999/02/22-rdf-syntax-ns#"
  xmlns:foaf="http://xmlns.com/foaf/0.1/"
  xmlns:rdfs="http://www.w3.org/2000/01/rdf-schema#"
  xmlns:lang="http://purl.org/net/inkel/rdf/schemas/lang/1.1#">
  <foaf:Person rdf:nodeId="me">
    <foaf:nick>Toby</foaf:nick>
    <foaf:givenName>Toby</foaf:givenName>
    <foaf:surName>Segaran</foaf:surName>
    <foaf:birthday>1-20</foaf:birthday>
    <foaf:img rdf:resource="http://photos3.hi5.com/0057/846/782/qE64Yc846782-01.jpg"/>
    <foaf:weblog rdf:resource="http://blog.kiwitobes.com"/>
    <foaf:gender>male</foaf:gender>
    <lang:masters>en</lang:masters>
    <foaf:homePage
      rdf:resource=
        "http://www.hi5.com/friend/profile/displayProfile.do?userid=358280494"/>
    <foaf:knows>
      <foaf:Person>
        <foaf:nick>Jamie</foaf:nick>
        <rdfs:seeAlso rdf:resource="http://api.hi5.com/rest/profile/foaf/241087912"/>
      </foaf:Person>
    </foaf:knows>
  </foaf:Person>
</rdf:RDF>
```

Hi5 publishes FOAF data

# Geonames

---

## Linked Data

The Features in the *GeoNames Semantic Web* are interlinked with each other. Depending on applicability the following documents are available for a Feature :

The **children** (countries for a continent, administrative subdivisions for a country, ...). As an example the children of France : <http://sws.geonames.org/3017382/contains.rdf>

The **neighbours** (neighbouring countries). As an example the neighbours of France : <http://sws.geonames.org/3017382/neighbours.rdf>

**Nearby** features. Nearby to the Eiffel Tower are Champ de Mars, Trocadéro - Palais de Chaillot, ...: <http://sws.geonames.org/6254976/nearby.rdf>

For the time being the links between geonames and the French INSEE are available in separate documents only : [régions et départements](#) and [arrondissements](#). These links will later be included in the web service documents. More information about the INSEE rdf documents on the [INSEE website](#) (in French).

*Stats, containers and neighbors*

# MusicBrainz

## SPARQL Explorer for <http://dbtune.org/musicbrainz/sparql>

### SPARQL:

```
PREFIX map: <file:/home/moustaki/work/motools/musicbrainz/d2r-server-0.4/mbz_mapping_raw.n3#>
PREFIX rdfs: <http://www.w3.org/2000/01/rdf-schema#>
PREFIX owl: <http://www.w3.org/2002/07/owl#>
PREFIX event: <http://purl.org/NET/c4dm/event.owl#>
PREFIX rel: <http://purl.org/vocab/relationship/>
PREFIX lingvoj: <http://www.lingvoj.org/ontology#>
PREFIX foaf: <http://xmlns.com/foaf/0.1/>
PREFIX rdf: <http://www.w3.org/1999/02/22-rdf-syntax-ns#>
PREFIX tags: <http://www.holygoat.co.uk/owl/redwood/0.1/tags/>
PREFIX db: <http://dbtune.org/musicbrainz/resource/>
PREFIX dc: <http://purl.org/dc/elements/1.1/>
PREFIX geo: <http://www.geonames.org/ontology#>
PREFIX bio: <http://purl.org/vocab/bio/0.1/>
PREFIX mo: <http://purl.org/ontology/mo/>
PREFIX vocab: <http://dbtune.org/musicbrainz/resource/vocab/>
PREFIX xsd: <http://www.w3.org/2001/XMLSchema#>
PREFIX mbz: <http://purl.org/ontology/mbz#>
```

```
SELECT ?title WHERE {
  ?artist foaf:name "Madonna" .
  ?work foaf:maker ?artist .
  ?work dc:title ?title .
}
```

Results:

### SPARQL results:

title
"Burning Up"
"Jump (Axwell remix)"
"Love Profusion (The Passengerz Club Profusion)"
"Causing a Commotion"
"Burning Up"
"Like a Prayer (7" remix edit)"
"Erotica (Master at Work dub)"

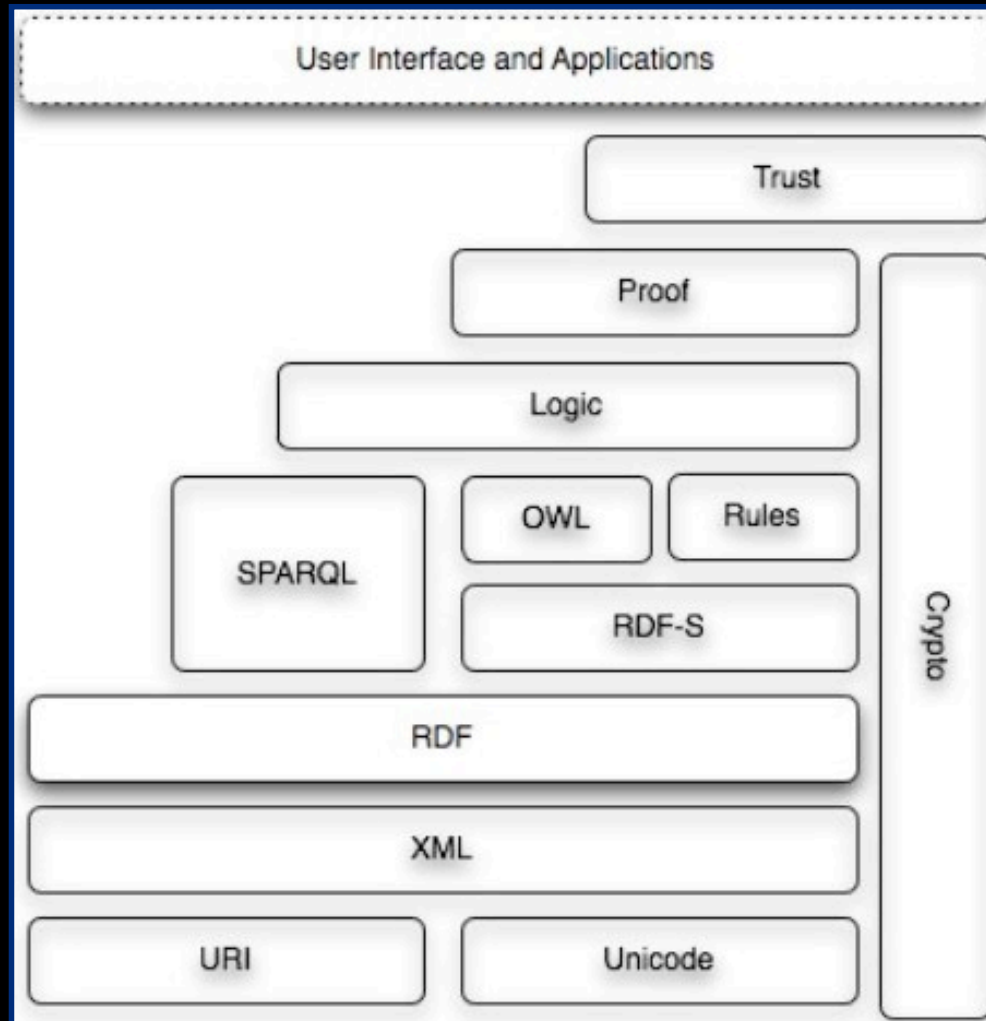
# Finance/XBRL

```
<ifrs-gp:OtherOperatingIncomeTotalFinancialInstitutions contextRef="J2004" decimals="0"
unitRef="EUR">38679000000</ifrs-gp:OtherOperatingIncomeTotalFinancialInstitutions>
<ifrs-gp:OtherAdministrativeExpenses contextRef="J2004" decimals="0"
unitRef="EUR">35996000000</ifrs-gp:OtherAdministrativeExpenses>
<ifrs-gp:OtherOperatingExpenses contextRef="J2004" decimals="0" unitRef="EUR">8700000000</
ifrs-gp:OtherOperatingExpenses>
. . .
<ifrs-gp:OtherOperatingIncomeTotalByNature contextRef="J2004" decimals="0"
unitRef="EUR">10430000000</ifrs-gp:OtherOperatingIncomeTotalByNature>
  <xbrli:context id="BJ2004">
    <xbrli:entity>
      <xbrli:identifier scheme="www.iqinfo.com/xbrl">ACME</xbrli:identifier>
    </xbrli:entity>
    <xbrli:period>
      <xbrli:instant>2004-01-01</xbrli:instant>
    </xbrli:period>
  </xbrli:context>
  <xbrli:context id="EJ2004">
    <xbrli:entity>
      <xbrli:identifier scheme="www.iqinfo.com/xbrl">ACME</xbrli:identifier>
    </xbrli:entity>
    <xbrli:period>
      <xbrli:instant>2004-12-31</xbrli:instant>
    </xbrli:period>
  </xbrli:context>
```

*Many countries requiring XBRL reporting*

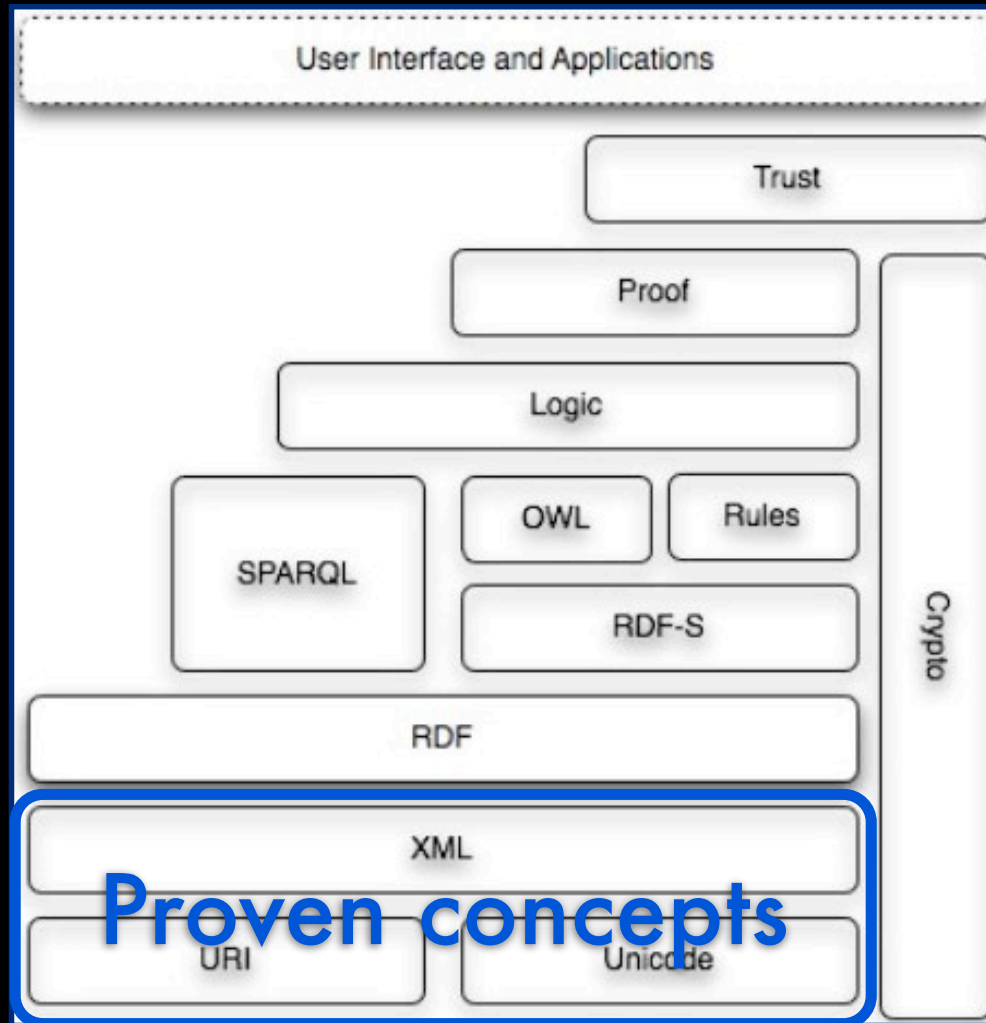
Vision, hype, reality

# Epilogue



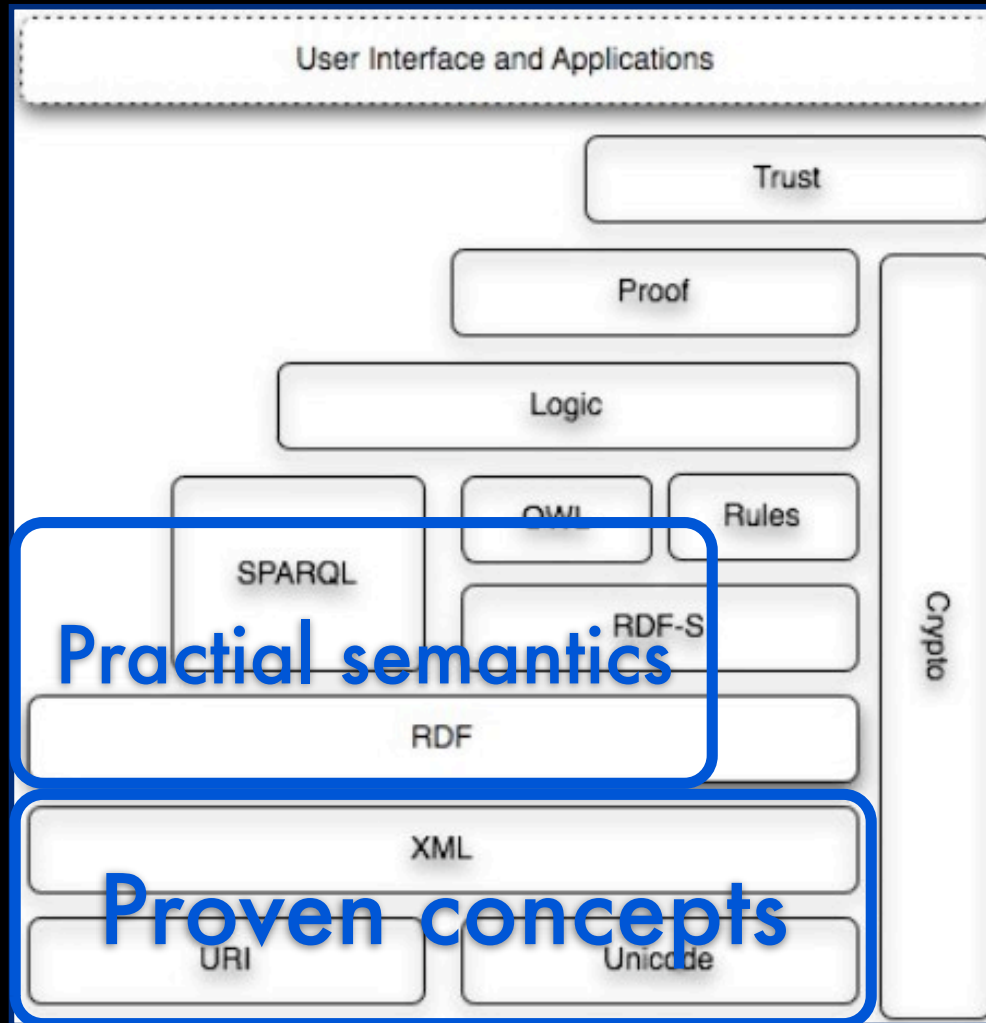
*be very afraid of complicated stacks*

# Epilogue



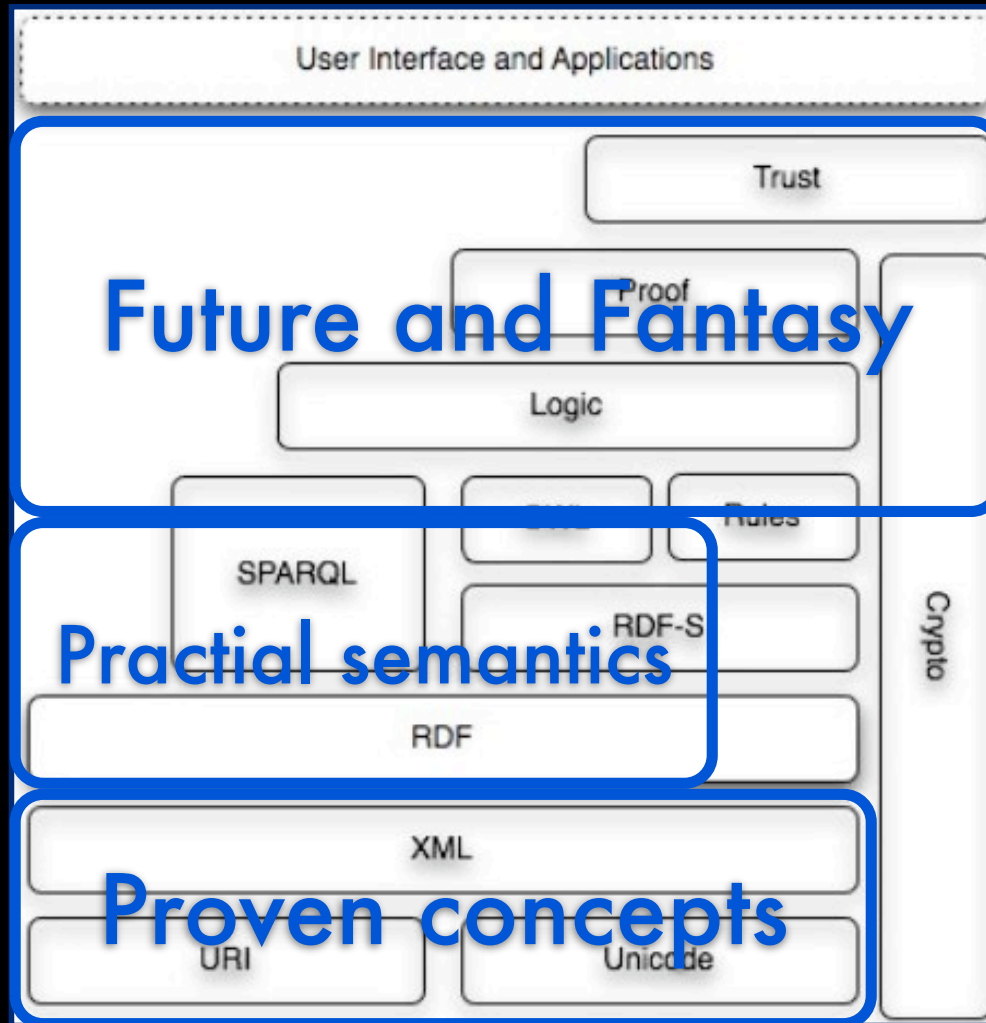
*ok, fair enough*

# Epilogue



*what we talked about today*

# Epilogue



*and here's where we get into trouble*

The end.