Love Thy Data
(or: Apps Considered Harmful)

Dr. Ora Lassila

Principal Technologist
Cloud Analytics Team
Nokia Location & Commerce

Elected Member
Advisory Board
World Wide Web Consortium (W3C)
Some speaker details

• current and past positions:
  − principal architect with Nokia’s “big data analytics” unit
  − elected member of W3C’s Advisory Board since 1998
  − research positions at Nokia Research, MIT, CMU, HUT
  − venture capitalist, entrepreneur, software engineer

• education:
  − Ph.D (D.Sc) in Computer Science, HUT

• some (perhaps dubious) achievements:
  − co-invented the Semantic Web; co-author of the highest cited article on the topic; co-editor of the original RDF specification
  − software for NASA’s Deep Space 1 (Asteroid Belt in 1998)
  − Grand Prize @ USENIX Intl. Obfuscated C Code Context, 1989
Some speaker details

• current and past positions:
  − principal architect with Nokia’s “big data analytics” unit
  − elected member of W3C’s Advisory Board since 1998
  − research positions at Nokia Research, MIT, CMU, HUT
  − venture capitalist, entrepreneur

• education:
  − Ph.D (D.Sc) in Computer Science, HUT

• some dubious achievements:
  − co-invented the Semantic Web; co-author of the highest cited article on the topic; co-editor of the original RDF specification
  − software for NASA’s Deep Space 1 (Asteroid Belt in 1998)
  − Grand Prize @ USENIX Intl. Obfuscated C Code Context, 1989
This is what I would like to talk about today

• first, let’s have to look at what is going wrong (with information systems development)

• Semantic Web as a possible solution to address some of the above problems

• a bigger picture of how we could acquire, store, process and use data
Part 1: The Problem
First, let’s define what an “app” is

• data + logic + presentation

• a way to package/deliver/deploy the three
  – in some way, this is an antiquated notion
    that mostly comes from the needs of
    developers/publishers (users don’t care)

• we see different kinds of apps, including
  1. perform a specific function
     (e.g., a “camera” app)
  2. present users with some specific data
     (e.g., the “NY Times” app)

• specifically with #2, one is left wondering,
  why not just use the Web…
Issues with data

• typically, data lives in a “silos” and has opaque semantics
  − proprietary data models (semantics)
  − proprietary data formats (syntax)

• this makes the data hard to
  − access (from outside the app)
  − reuse (by other systems)
  − integrate (with data from other sources)

• an app typically “owns” its data, locking users to this particular app

• access/reuse/integration, at best, are engineering endeavors
Issues with logic

- typically, logic is “embedded” in the app and has (at best) opaque semantics
- this makes it hard to
  - access the logic – associate data with this logic except through (and in the context of) the app
  - reuse the logic in some other system
Issues with presentation

• typically, presentation is “fixed”
  -(i.e., decided by developers of the app)

• this makes it hard to
  -flexibly change the presentation per desires
  and preferences of the user
  -reuse the presentation in some other context

• “packaging” content in a (native) app
  excludes the good the Web would give us
  -no linking, no bookmarking
  -no accessibility features (unless the platform
  provides those; cf. reuse of data/content)

• HTML5 to the rescue?
Random examples of bad (and good) apps

• **bad:** NY Times – no linking, bookmarking, text refers to links that are not there

• **bad:** Netflix – similar to the Web site, but offers fewer options in cross-linking, etc.

• **better:** Financial Times – app built using Web standards wins over native

• **better:** Amazon Kindle “cloud reader” – built using Web standards, avoids App Store royalties for in-app purchases

• **better:** Flipboard – allows users to select content via open data
Whether we are talking about data, logic or presentation, locking these in an un-reusable “silo” only further fragments our information space.
Perhaps this is in our future?

Whether we are talking about data, logic or presentation, locking these in an un-reusable “silo” only further fragments our information space.
Always focus on data

• apps and systems come and go, but data has longevity
• always assume that data
  – comes from multiple sources
  – has multiple “owners”
  – spans multiple application domains
• specifically, focus on things that make sharing possible:
  – open formats and models
  – “accessible” semantics
  – also: don’t forget data provenance
Data formats?

• data format (= syntax) is an important issue, but
  – all issues wrt. formats have already been solved
    → no need to reinvent or redefine things
  – once you decide on syntax, you should forget about it

• people seem to think that “format = model”, but this leads to all kinds of issues … also, there is a persistent belief that as long as you understand the syntax, you have “solved the problem” (unfortunately not so)

• people tend to be overly focused on syntax (big mistake)
  – (evidence: current public discussions on how to improve JSON focus on changing the syntax – seriously!)
Data models?

• modern ontological technologies allow the semantics of a domain to be captured in a model (for reuse)

• in many cases, an open (even standard) conceptual model exists for the domain you are interested in
  − but: you typically have to extend it for your own use cases

• checklist if you are defining models:
  − make them extensible, assume people will want to extend
  − assume these models are not used in isolation, but instead they need to interconnect with other models
What establishes (data) semantics?

1. relationship of data to (accessible & declarative) definitions of data types
2. relationship of data to some other data
3. some (procedural) software that “hard-wires” how to process certain kind of data

• all semantics is grounded in the above three
  – note that #1 is recursive
  – the less you have #3, the better
    (and yet, today, most of semantics is captured via #3)
Part 2: The Semantic Web
Characterizing the Semantic Web

• WWW, as conceived, is human-oriented
  – this is both good and bad
  – difficult to automate (particularly unforeseen situations)
  – to employ machines more, we need data

• Semantic Web aims at making it easier to use data in an automated fashion (with implications to interoperability)

• Semantic Web is an “interoperability technology”
  – contrary to many examples about “Web 2.0”, the Semantic Web aims at achieving many things “ad hoc”
  – shared (and accessible) semantics is the key to interoperability
    → Semantic Web aims at using ontologies to model the world
Serendipity defines the Semantic Web

Serendipity in...

**interoperability:** is it possible to interoperate with systems and services we knew nothing about at design time?

**reuse:** when information has accessible semantics, this is easier...

**integration:** can information from various independent sources be combined?
Understanding the Semantic Web vision

• Semantic Web is ultimately about how we want to build information systems, and how we want information technology to serve people

• key challenges:
  1. where does data come from – access to data
  2. how is data processed – the ability to flexibly handle unanticipated situations
  3. how to present data to users – matching the richness of data with the expressiveness of user interaction

• the vision should not be considered in isolation, but as part of a broader vision for information technology
Semantic Web and “culture”

• different domains (of discourse) are their own “cultures” and have languages of their own

• examples from scientific disciplines:
  – biology vs. economics
  – ecology vs. physiology vs. molecular biology
  – proteins: folding vs. expression vs. interactions

• scientific disciplines also use conceptual models (about the world) that are different from others’
  – e.g., different levels of abstraction

• but… “no domain is an island” – domains interconnect
  – museum artifacts → history → geography → travel → …
Semantic Web and “culture”

- Semantic Web was designed to
  - accommodate different points of view
  - be flexible about what it can express – not preferential towards any particular domain or application

- serendipity of combining information in new ways
  - we cannot anticipate all the possible ways in which information is used, combined
  - using Semantic Web formalisms lowers the threshold for “serendipitous reuse”

- a new approach to standardization
  - standardize how things are said, not what is said
Part 3: Future?
“Existential Crisis” of the Semantic Web...?

- Semantic Web was conceived as “integration and interoperability” technology
- it is all grown up: The main technical pieces are in place

BUT...

- what about our dream of being able to ontologically model the world?
“Existential Crisis” of the Semantic Web…?

• prescriptive approaches to the world are known to fail
  – rather, Semantic Web is very much intended to be descriptive

• “global ontology” a bad idea – the broader the scope, the weaker or more complex the resulting ontology

• this is not just a technical challenge…
## Hierarchy of information scales (cf. mapping)

<table>
<thead>
<tr>
<th>1. Mapping <strong>scalar objects</strong>, units of measure, etc.</th>
</tr>
</thead>
<tbody>
<tr>
<td><em>e.g.</em>, UNIX date → ISO 8601 date</td>
</tr>
<tr>
<td>Mostly syntactic, yet often offered as “semantic transformations”</td>
</tr>
<tr>
<td><strong>THIS IS NOT A PROBLEM!</strong></td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>2. Mapping <strong>structured objects</strong></th>
</tr>
</thead>
<tbody>
<tr>
<td><em>e.g.</em>, ovi:Person → facebook:Person</td>
</tr>
<tr>
<td>Doable, particularly if semantics on both sides are <strong>already a good match</strong>, still this may lead to “subsetting”, making round-trips difficult</td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>3. Mapping entire <strong>application data models</strong> (or ontologies) onto other applications’ models</th>
</tr>
</thead>
<tbody>
<tr>
<td><em>e.g.</em>, Nokia Ovi Services → Facebook</td>
</tr>
<tr>
<td>Achieving bijective and transitive mappings much harder, also much of the semantics is embodied in applications’ “business logic”</td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>N</th>
<th>Mapping entire <strong>cultural “contexts”</strong></th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td><em>e.g.</em>, US → France → Finland</td>
</tr>
<tr>
<td></td>
<td><em>note</em>: finland:Café ≠ france:Café</td>
</tr>
<tr>
<td></td>
<td>Is it even possible…? Very difficult, but perhaps not entirely hopeless [Lassila 2006]</td>
</tr>
</tbody>
</table>
“Value chain” for data

• Where does “semantic” data come from?

**symbolic methods**
- reasoning, logic

**non-symbolic methods**
- data mining
- machine learning

**signal processing**
“Value chain” for data – extended view

reusable data

structured sources

unstructured sources
“Value chain” for data – extended view

What’s important?

• multiple models & domains
  ⇒ mapping models & data
  ⇒ provenance
• integration (via reasoning)
  ⇒ identity
Conclusions, last words...

- current way of designing, building and delivering information technology to end users is **broken**
  - information is **isolated**, information space is **fragmented**

- Semantic Web is a set of technologies that can be used to address some of the problems
  - however, covering “a lot of ground” is difficult

- we should **focus on data**, understanding that various means to process is it come and go
  - make it possible to **share** data, and other people will come up with new ways of using your data

- **homework:** what about **business models** for all this?
Thank you!

• questions, comments?

• short rants:  @gotsemantics
• long(er) rants:  http://www.lassila.org/blog
• contact:  ora.lassila@nokia.com

• thanks to:  Ian Oliver,
  Mika Mannermaa,
  Mike Champion