The Marvelous Structure of Reality

Joseph M Hellerstein
UC Berkeley
“The important thing is to not stop questioning ... One cannot help but be in awe when contemplating the mysteries of eternity, of life, of the marvelous structure of reality.”

- Albert Einstein
A Modest Agenda

- Mythology
- History
- Philosophy
- Art
- History, again
- Marvelous structures in reality
- On beauty, complexity and fruit
A Myth:
The Strictures of Structure
Out of chaos came tables.
And it was good.
(Usually.)
But in time our eyes were opened to the strictures of structure.
Only a Houdini could figure out all those keys.
On the Internet there was no time for this schema nonsense.
Yet everyone fears chaos.
We needed a happy medium.
Along came the web
and brought us...
Semi-structured data.
fin

This myth brought to you by the world-wide web consortium, a host of software companies, and contributions from viewers like you.
But seriously ...

* It’s not that semi-structured is bad

* It’s just that semi-structured is not semi-structured
But seriously ...

It's not that semi-structured is bad

It's just that semi-structured is not semi-structured
A Modest Agenda

- Mythology
- History
- Philosophy
- Art
- History, again
- Marvelous structures in reality
- On beauty, complexity and fruit
Meanwhile, in Computing History...

- **1959**: Hans P. Luhn describes Keyword in Context (KWIC).
- **1969**: Edward F. Codd publishes first papers on the relational model.
- Structured/Unstructured dichotomy
The Pillars of Modern InfoSystems

- “Unstructured” document retrieval
- “Structured” databases

Assertion (following J. Derrida)

- This dichotomy is simultaneously meaningless and useful

Let us revisit each...
Codd’s data independence was a SW engineering lesson:

- whenever: \( \frac{d\text{App}}{dt} \ll \frac{d\text{Env}}{dt} \)
- shield apps from changes via Data Independence
- requires engineered structure
Unstructured Data

In many cases, data wasn't intended for an app!

Then for what?

(Soylent Green is ...)

PEOPLE!

Yet behind all human discourse is "deep structure" (F. de Saussure)
“Shakespeare described seven ages of man, [Shakespeare 1599], starting from infancy and leading to senility. The history of information retrieval parallels such a life. The popularization of the idea of information retrieval started in 1945, with Vannevar Bush's article (still cited 96 times in the 1988–1995 Science Citation Index). [Bush 1945]. And, given the current rate of progress, it looks like it will finish by 2015 or so, the standard life-span for someone born in 1945. By that time, most research tasks will be performed on a screen, not on paper ...”

-- Michael Lesk, “The Seven Ages of Information Retrieval”
... here's an Inverted Index

<table>
<thead>
<tr>
<th>Term</th>
<th>DocID</th>
<th>Position</th>
<th>Score</th>
</tr>
</thead>
<tbody>
<tr>
<td>age</td>
<td>1</td>
<td>4</td>
<td>0.968071</td>
</tr>
<tr>
<td>article</td>
<td>1</td>
<td>40</td>
<td>0.066731</td>
</tr>
<tr>
<td>born</td>
<td>1</td>
<td>75</td>
<td>0.478281</td>
</tr>
<tr>
<td>bush</td>
<td>1</td>
<td>51</td>
<td>0.909534</td>
</tr>
<tr>
<td>bush</td>
<td>1</td>
<td>39</td>
<td>0.351692</td>
</tr>
<tr>
<td>citation</td>
<td>1</td>
<td>49</td>
<td>0.932534</td>
</tr>
<tr>
<td>cited</td>
<td>1</td>
<td>42</td>
<td>0.654436</td>
</tr>
<tr>
<td>current</td>
<td>1</td>
<td>56</td>
<td>0.021070</td>
</tr>
<tr>
<td>described</td>
<td>1</td>
<td>2</td>
<td>0.512205</td>
</tr>
<tr>
<td>finish</td>
<td>1</td>
<td>65</td>
<td>0.202019</td>
</tr>
<tr>
<td>given</td>
<td>1</td>
<td>54</td>
<td>0.939977</td>
</tr>
<tr>
<td>history</td>
<td>1</td>
<td>18</td>
<td>0.204082</td>
</tr>
<tr>
<td>idea</td>
<td>1</td>
<td>30</td>
<td>0.378829</td>
</tr>
<tr>
<td>index</td>
<td>1</td>
<td>50</td>
<td>0.793114</td>
</tr>
<tr>
<td>infancy</td>
<td>1</td>
<td>11</td>
<td>0.288201</td>
</tr>
<tr>
<td>information</td>
<td>1</td>
<td>20</td>
<td>0.267157</td>
</tr>
<tr>
<td>information</td>
<td>1</td>
<td>32</td>
<td>0.356823</td>
</tr>
<tr>
<td>leading</td>
<td>1</td>
<td>13</td>
<td>0.128374</td>
</tr>
<tr>
<td>lifespan</td>
<td>1</td>
<td>72</td>
<td>0.703298</td>
</tr>
<tr>
<td>life</td>
<td>1</td>
<td>25</td>
<td>0.737414</td>
</tr>
</tbody>
</table>
.. and here is Eric Brewer's Search Engine
Where do we go from here?

- Subverted the structured/unstructured dichotomy!?
- Without opposition, terms lose all meaning?
- And yet, the methodology may still be useful (Derrida, again)
- What are the methodological lessons?
A Key Methodological Distinction

- Engineered Structure (DBs) vs. “Found” Structure (IR)

- We will be returning to this throughout
A Modest Agenda

- Mythology
- History
- Philosophy
- Art
- History, again
- Marvelous structures in reality
- On beauty, complexity and fruit
A (?) Brief (?) Detour (?)

- A peek at some 20th Century Philosophy/Criticism
- And some related Art History
Others have worried about structure

- Databases
  - Structured/Unstructured
- Philosophy & Criticism
  - Structuralism/Deconstruction
- Art
  - Structurism/Bricolage
Derrida Addressed our Dichotomy

(Following C. Lévi-Strauss)
Contrast the Bricoleur with the Engineer

The Bricoleur potters about with odds-and-ends, puts things together out of bits and pieces. “Tinkerer”.

The Engineer forms stable structures out of “whole cloth”

Bricoleur/Engineer

Bricolage:
- Juxtaposition without requiring rationality
- Enables what Derrida calls “play”
- Addressing & affirming provisional truths

Engineering
- Stable structures with little or no “play”
- Engineer must be at center of his discourse
- A God-like figure. A myth.
- (Really, engages in bricolage after all.)
If the Engineer is really a Bricoleur...

- This subverts the dichotomy between engineering/bricolage
- Just as we saw with structured/unstructured
- But the Derrida response is to affirm the play in this false dichotomy
- rather than mourn the loss of simplicity
A Modest Agenda

- Mythology
- History
- Philosophy
- Art
- History, again
- Marvelous structures in reality
- On beauty, complexity and fruit
Now in Art

Structurism: “[to] achieve the highest degree of ‘reality’ possible for the new art . . . it was necessary that it be as similar in structure as possible to the structure of nature’s reality process” — Charles Biederman

“Capturing” structure
Art History, Cont.

- M. Duchamp’s “Found” art
- Bricolage (e.g. Tom Sachs)

Again a dichotomy. Intentional “play”!
M. Duchamp’s “Found” art

Bricolage (e.g. Tom Sachs)

Again a dichotomy. Intentional “play”!
M. Duchamp’s “Found” art

Bricolage (e.g. Tom Sachs)

Again a dichotomy. Intentional “play”!
A Modest Agenda

- Mythology
- History
- Philosophy
- Art
- History, again
- Marvelous structures in reality
- On beauty, complexity and fruit
Returning to Safer Ground...

Let us reflect on IR and DB history & culture
The Strange History of Information Retrieval

- Far, far ahead of its time
- Initial relevance with digital typesetting (1970’s)
- Growing like weeds in the Web era
- though the pioneers have passed
  - HP Luhn, 1896-1964
  - Gerard Salton, 1927-1995
Contrast with Relational History

- 1970: Identified and heralded for existing business applications
- 1974: Two major implementations underway
- 1980: Commercialization
- 1990: Big business

Pioneers still social-engineering
  - Witness recent Lowell Report
Upshot on Comparative History Exercise

- IR community being “bricolated”
- DB folks still busy self-engineering
- Which field is healthier?
- Hmm...
So Much for History, Philosophy and Art...

- What can we learn from them?
- Recurring themes
  - Engineered vs. Found Structure
  - Exploiting the “play” between the two
DB Lessons

We know the relational lessons:

- Simple structure provides resilience to change
- A priori modeling ensures consistent data
- Strict semantics, understandable systems

Lessons in Software Engineering!

- Culturally, a goal-oriented field
- Derrida's "engineer"
Lessons from IR?

- Human discourse awash in structure
- Extract structure into simple models
- Glory not in subtlety!
  - 80% information in 20% of the structure
- Culturally, an organic, evolving field
- Bricolage!
Summing Up

- Structured/Unstructured echoes
  - Engineering/Bricolage
  - In content and culture

Useful?
- Methodological distinctions useful
- And we should “play” with the subverted structured/unstructured dichotomy
Moving Forward

- Opportunities for bricolage?
- Opportunities for engineering reality?
- The play’s the thing!
A Modest Agenda

- Mythology
- History
- Philosophy
- Art
- History, again
- Marvelous structures in reality
- On beauty, complexity and fruit
Some Marvelous Structures in Reality
Beautiful Structures Being Found

- The physical world (sensors)
  - Naturally tabular, numeric data
  - Amenable to (continuous) relational queries

- The cyber world
  - Your software is talking, are you listening?
  - Your network is talking, are you listening?
Tiny Sensor Nodes

- Think PC-AT with \( k \) sensors and a radio
- Emits \( k \)-tuples of readings
- Power-constrained
Wireless Sensor Networks

To deploy lots and lots of these:

- Must be cheap
- Must be zero-admin: pref. disposable
- Must form ad-hoc, multi-hop networks
- Network will have much higher BW “inside” than to the outside world
Begging to be Queried!

- Not like a traditional network
  - point-to-point comm (e.g. email)
  - client-server comm (e.g. web)
- Much more like a database
  - External user requests properties of the sensed environment
- TinyDB is our query engine
  - (SIGMOD ‘03, IPSN ‘03, OSDI ‘02)
A “big picture” of the data: wavelet histogram

From “support” graph to comm graph

Beautiful Structure Here
A “big picture” of the data: wavelet histogram

From “support” graph to comm graph
Beautiful Structure Here

- A “big picture” of the data: wavelet histogram
- From “support” graph to comm graph
A “big picture” of the data: wavelet histogram

From “support” graph to comm graph

Beautiful Structure Here
Beautiful Structure Here

- A “big picture” of the data: wavelet histogram
- From “support” graph to comm graph
Beautiful Structure Here

- A “big picture” of the data: wavelet histogram
- From “support” graph to comm graph
A “big picture” of the data: wavelet histogram

From “support” graph to comm graph

Beautiful Structure Here

A Binomial Tree!
Found Structure!

- Full Binary Support Tree yields Binominal Comm Tree!
- Other interesting mappings
  - E.g. computing transitive closures of network routing tables
- A new query optimization problem
  - Consider all legal support graphs and all mappings to (satisfying) comm graphs
Your Software is Talking...
Your Software is Talking...
Your Network is Talking...

[localhost:/tmp] jmh# tcpdump -i en1 | more
tcpdump: listening on en1
21:32:17.943316 192.168.1.1.1901 > 239.255.255.250.1900: udp 269
21:32:17.945706 192.168.1.1.1901 > 239.255.255.250.1900: udp 325
21:32:17.947629 192.168.1.1.1901 > 239.255.255.250.1900: udp 253
21:32:17.949585 192.168.1.1.1901 > 239.255.255.250.1900: udp 245
21:32:17.952023 192.168.1.1.1901 > 239.255.255.250.1900: udp 289
21:32:17.954257 192.168.1.1.1901 > 239.255.255.250.1900: udp 265
21:32:17.956747 192.168.1.1.1901 > 239.255.255.250.1900: udp 319
21:32:17.959460 192.168.1.1.1901 > 239.255.255.250.1900: udp 317
21:32:17.961872 192.168.1.1.1901 > 239.255.255.250.1900: udp 321
21:32:17.964121 192.168.1.1.1901 > 239.255.255.250.1900: udp 313
21:32:18.429168 dns1.snfcca.sbcglobal.net.domain > 192.168.1.102.49691: 40780 NXDomain* 0/1/0 (118)
21:32:18.451257 dns1.snfcca.sbcglobal.net.domain > 192.168.1.102.49691: 39111 NXDomain 0/1/0 (119)
21:32:19.471971 dns1.snfcca.sbcglobal.net.domain > 192.168.1.102.49691: 21449 NXDomain* 0/1/0 (120)
21:32:19.494274 192.168.1.1.1901 > 239.255.255.250.1900: udp 269
21:32:19.497075 192.168.1.1.1901 > 239.255.255.250.1900: udp 325
21:32:20.950827 192.168.1.1.1901 > 239.255.255.250.1900: udp 245
21:32:20.953487 192.168.1.1.1901 > 239.255.255.250.1900: udp 289
21:32:20.958325 192.168.1.1.1901 > 239.255.255.250.1900: udp 319
21:32:20.965544 192.168.1.1.1901 > 239.255.255.250.1900: udp 313
21:32:49.678317 192.168.1.102.50167 > epoch.cs.berkeley.edu.http: S 1427962431:1427962431(0) win 32768 <mss 1460,nop,wscale 0,nop,
Found Structure on the Internet

- Logs are typically structured
- Many people run the same software
  - E.g. apache, sendmail, tcpdump, etc.
- Distributed, homogeneous data
- Begging to be federated!
- Querying the Internet
  - Vs. querying over the Internet

But how to scale to millions of nodes?
Internet Query
Processing over DHTs

- Content-based addressing research
- Distributed Hash Tables (DHTs)
- Can be thought of as
  - Indexes, Exchange, pt-to-pt comm channels
- Data Independence + Internet scale
- PIER is our DHT-based Internet query engine (VLDB 03)
DHT Design Goals

- An “overlay” network with:
  - Flexible map of logical IDs to physical nodes
  - Small diameter
  - Small degree
  - Local routing decisions
  - Routing flexibility and robustness to failure
An Example DHT: Chord
An Example DHT: Chord
An Example DHT: Chord
An Example DHT: Chord

Overlayed $2^n$-gons
An Example DHT: Chord

Overlayed $2^n$-gons
An Example DHT: Chord

Overlayed $2^n$-gons
An Example DHT: Chord

Overlayed $2^n$-gons
Routing in Chord

- At most one of each Gon
- E.g. 1-to-0
Routing in Chord

- At most one of each Gon
- E.g. 1-to-0
Routing in Chord

- At most one of each Gon
- E.g. 1-to-0
Routing in Chord

- At most one of each Gon
- E.g. 1-to-0
Routing in Chord

- At most one of each Gon
- E.g. 1-to-0
Routing in Chord

- At most one of each Gon
- E.g. 1-to-0

log \( n \) hops on \( \log n \) Gons!
Consider Aggregation in Chord

- Everybody sends their message to node 0
- Assume greedy jumps (increasing Gon-order)
- Intercept messages and aggregate along the way
Consider Aggregation in Chord

- Everybody sends their message to the root
- Assume greedy jumps (increasing Gin-order)
- Intercept messages and aggregate along the way, hierarchically
Consider Aggregation in Chord

- Everybody sends their message to the root
- Assume greedy jumps (increasing Gon-order)
- Intercept messages and aggregate along the way
Structure Upon Structure!

- Binomial agg in Tapestry/Pastry too!!
  - Found-within-engineered structure!
- Performing Bricolage on others’ engineering
- And engineering on upwards
  - Expect results on this soon from our group
Some Themes Here

- Found structure in common data
- New N.W. structures are engineered
  - Surprisingly beautiful patterns to be “found” in these structures
- A sweet spot for new DB/NW research
- The “play” in querying networked data
- In both the Derrida and Hellerstein senses
Brief Return to Mythology (semi...)

- Closer in spirit to engineering
- Most XML based on business messages, etc.
- Requires data independence with unnormalized data
- Hard for users & (especially!) apps to query
- Hard for systems to index and optimize
- Complexity for its own sake?
This is Not a Pipe

This
is
not
Verb
Adverb
Sentence
Subject
Verb
This
Adverb
not
Adjective
Semistructured
This is Not a Pipe

- There is nice work on finding structure in semi-structured DataGuides, XTRACT.
- But the end result is often deeply structured.
- Not less structured than tables; moreso!
  - I.e. “found complexity”
A Modest Agenda

- Mythology
- History
- Philosophy
- Art
- History, again
- Marvelous structures in reality
- On beauty, complexity and fruit
On Complexity, Beauty and Fruit

- In the Web-DB world...
- Shall we revel in complexity?
- Or feast on the low-hanging fruit?
- Which is more beautiful?
- Can't we do both?
Where's the Fruit?

- Unstructured data, redux
  - Clearly, we were largely absent mid-90's
- Sensors, net monitoring are new “found fruit”
  - We have much to bring to the table
  - The EE’s and the networking folks are trying to do our job...
Some Structure From Hellerstein's Bricolage Garage
Einstein the Religious

Seek out the Marvelous Structure of Reality

E.g. bags of words, sensor readings, etc.
Construct marvelous structures to harness reality

The lessons of data independence

E.g. relational schemas, DHTs, etc.
Find “The Play”: (Two Einsteins > One)

- One trick is to layer engineering on the found
  - E.g. search engines, DHTs, sensor queries

- Another is to find artful odds and ends in the engineering
  - E.g. agg in DHTs, routing for wavelets
A Play for WebDB

- Web/DB’s name & agenda is “play”
- Embrace the methodological dichotomy
  - found & engineered data
- Expand from “web” to “net”
- I promise you fruit.