Structure in Text: Extraction and Exploitation

Prabhakar Raghavan

Verity™

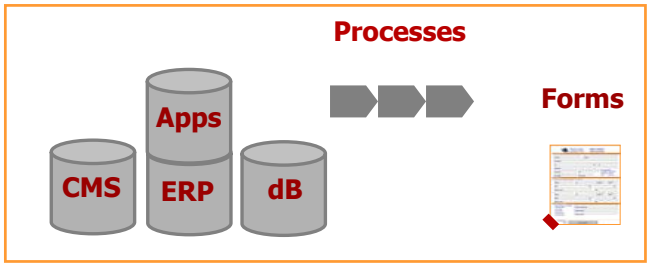
Agenda

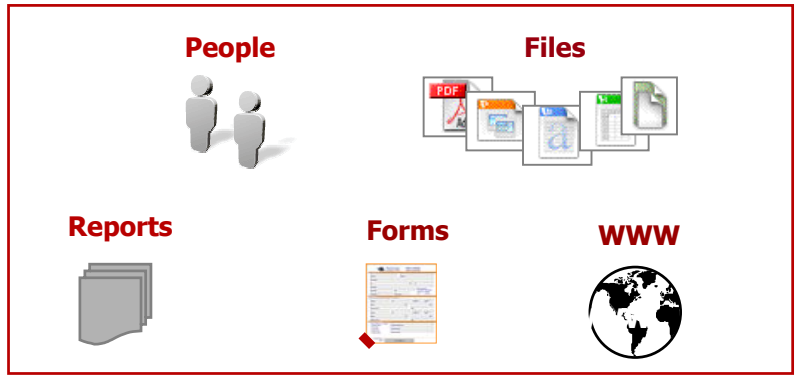
- Market overview
  - Structured and unstructured search – what convergence?
- Application scenarios
  - What my customers (think they) want
  - What it will take
- Technical tofu
  - Distance metrics for XML documents
  - Linear algebra as a foundation
- Summary/research agenda

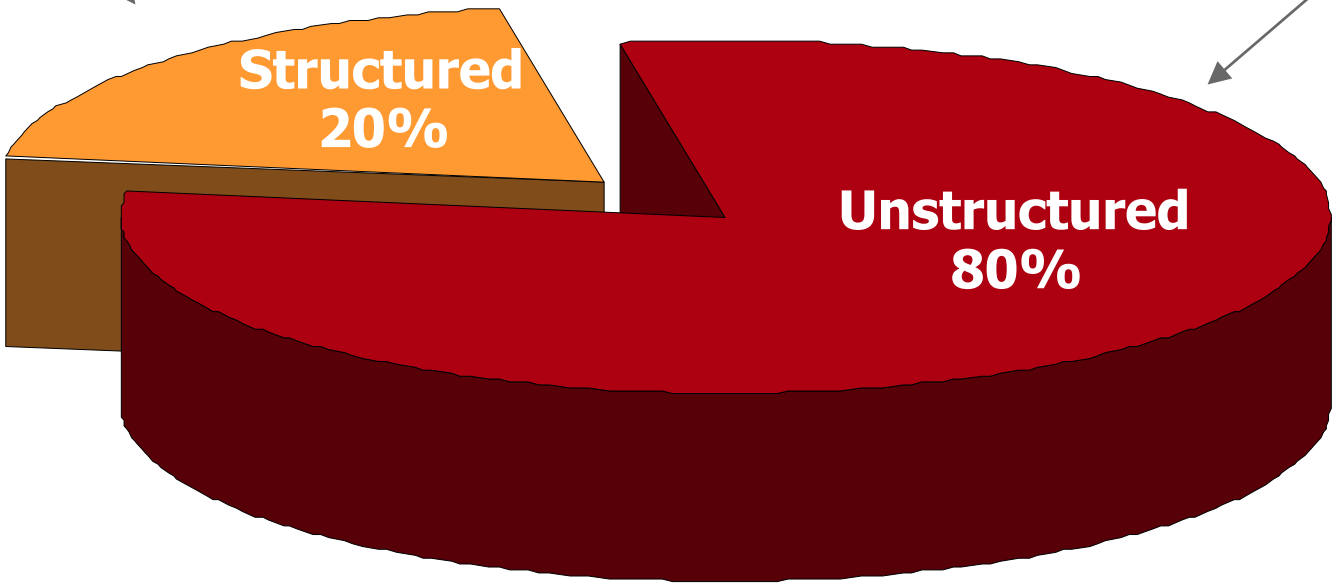
Verity™

Market overview

Unstructured vs. Structured Data in the Enterprise

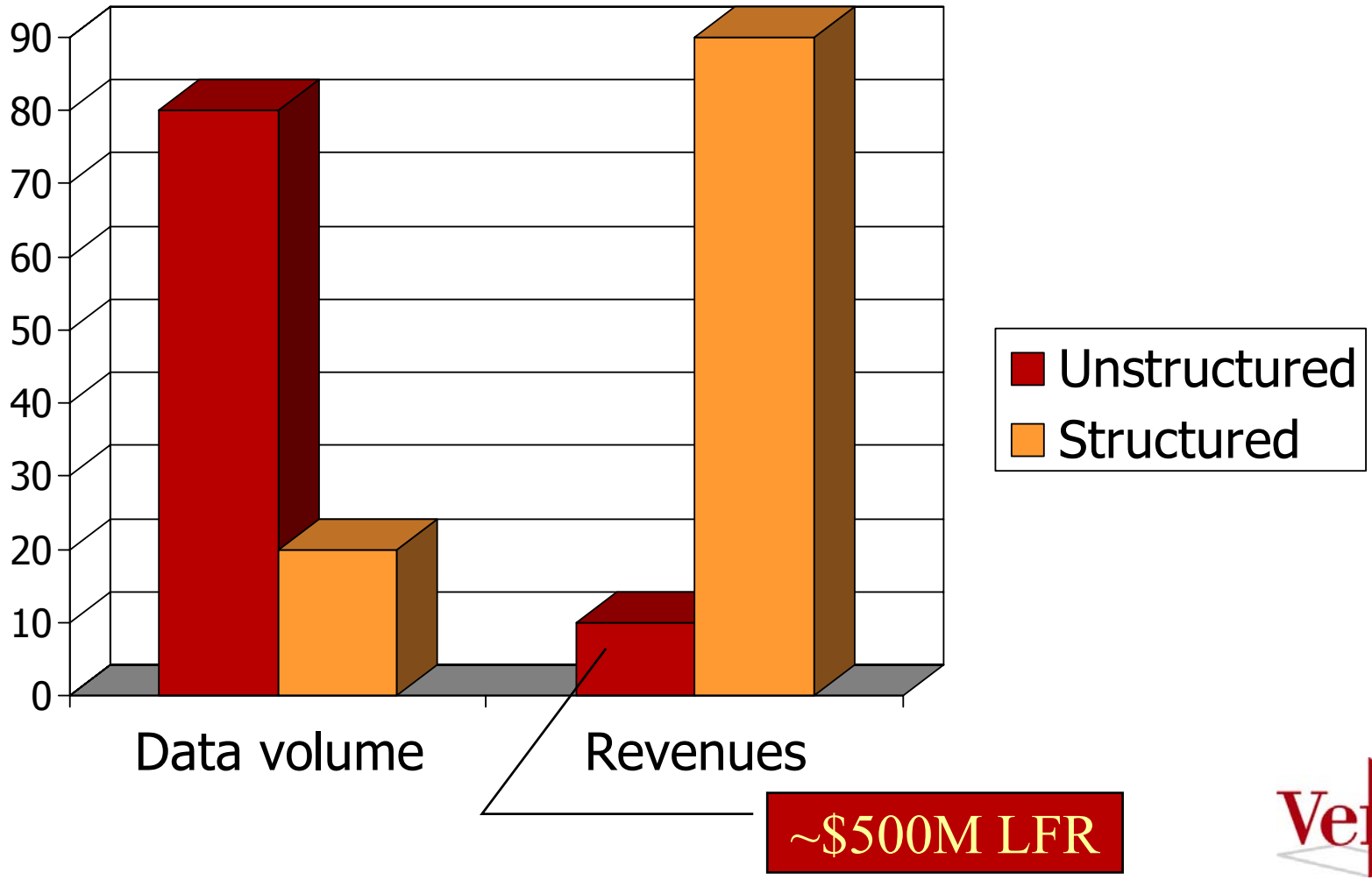






Verity™

Data: Volume vs. Revenues

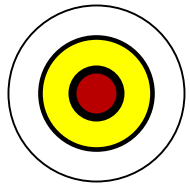


Consequences

- **Value creation demands:**
- **Extract structure**
  - Document capture/conversion
  - Classification, linguistic tagging
  - Entity/relation extraction
  - The results will be noisy
- **Exploit structure**
  - Ad-words
  - Structured navigation
  - XML querying

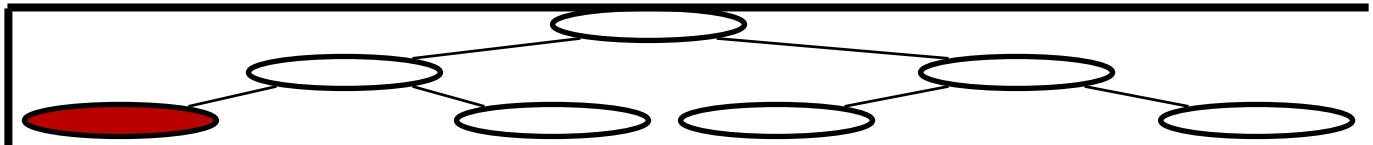
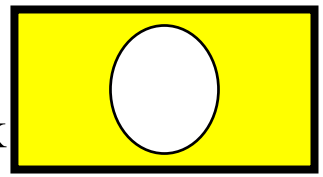
Verity™

Round pegs and square holes

	Unstructured	Structured
Unit of Retrieval		
Selection for Retrieval	Distance Metric 	Hard Selection 
Results	Consumed by humans: Limited Number. "Sort of" right; OK to get it wrong – Human "ratification".	Consumed by machines: Composition: Need correct answers. Nuclear powerplant syndrome.

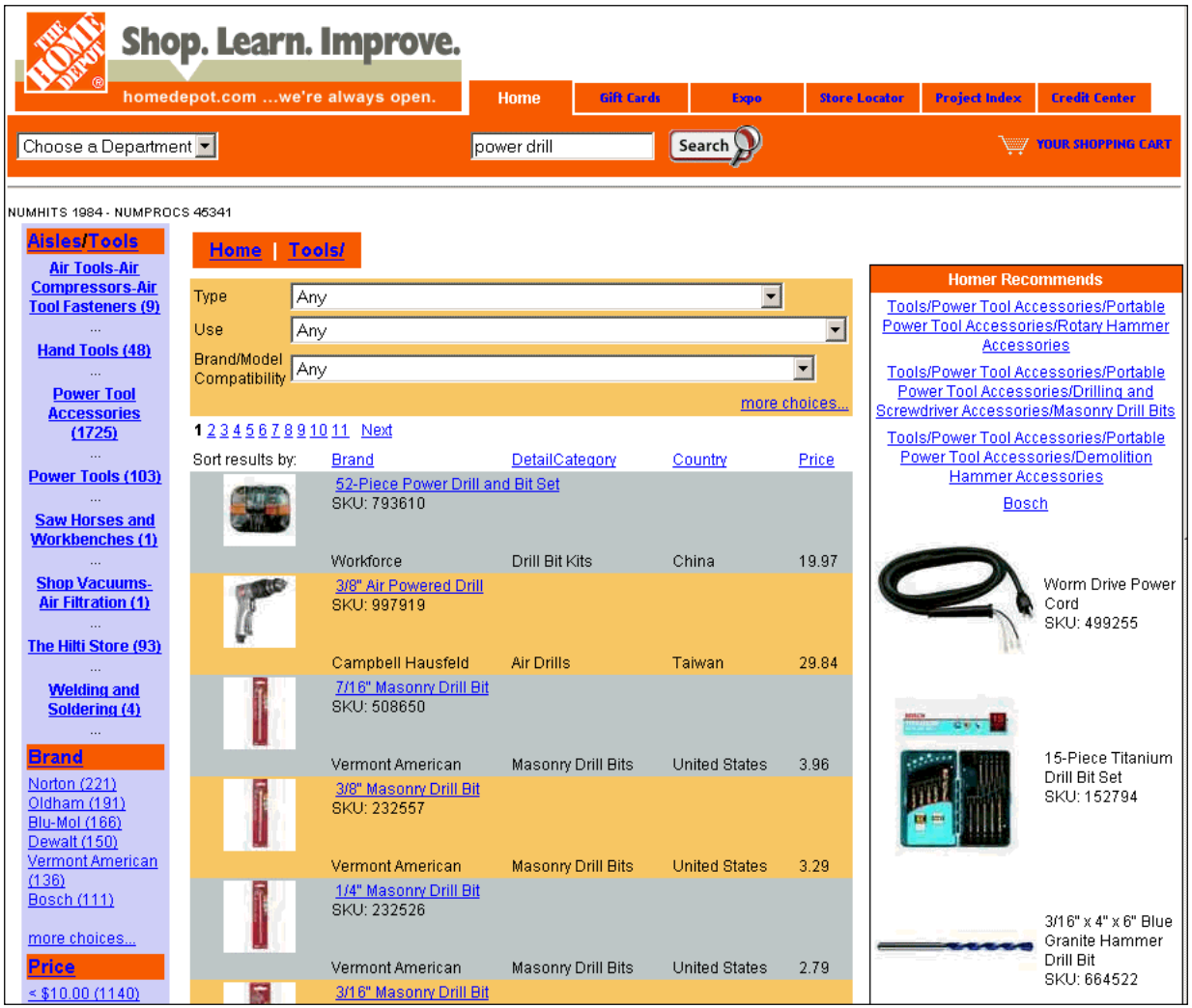
Semi-structured text: Round pegs meet square holes

Semi-structured

Unit of Retrieval	
Selection for Retrieval	Select then rank  ??  Semi-soft measures
Results	Will still get it wrong sometimes. Want to dramatically limit human ratification.

Verity

Example – relational taxonomies



Information retrieval platforms today

- Text and parametric search
- Classification
- Clustering and taxonomy induction
  - A modicum of "text mining"
- Index only, no repository
  - No content retrieval
    - (Exception: web search caches)
  - Affects "Return" portion of XML queries

Verity™

A step back: what's all this in aid of?

- End users don't want to
  - Search
  - Browse
  - Solve relevance problems
  - Deal with performance problems
  - Hear of your magic panacea for the above

- ***Users want to solve their business problems***

Verity™

Application scenarios

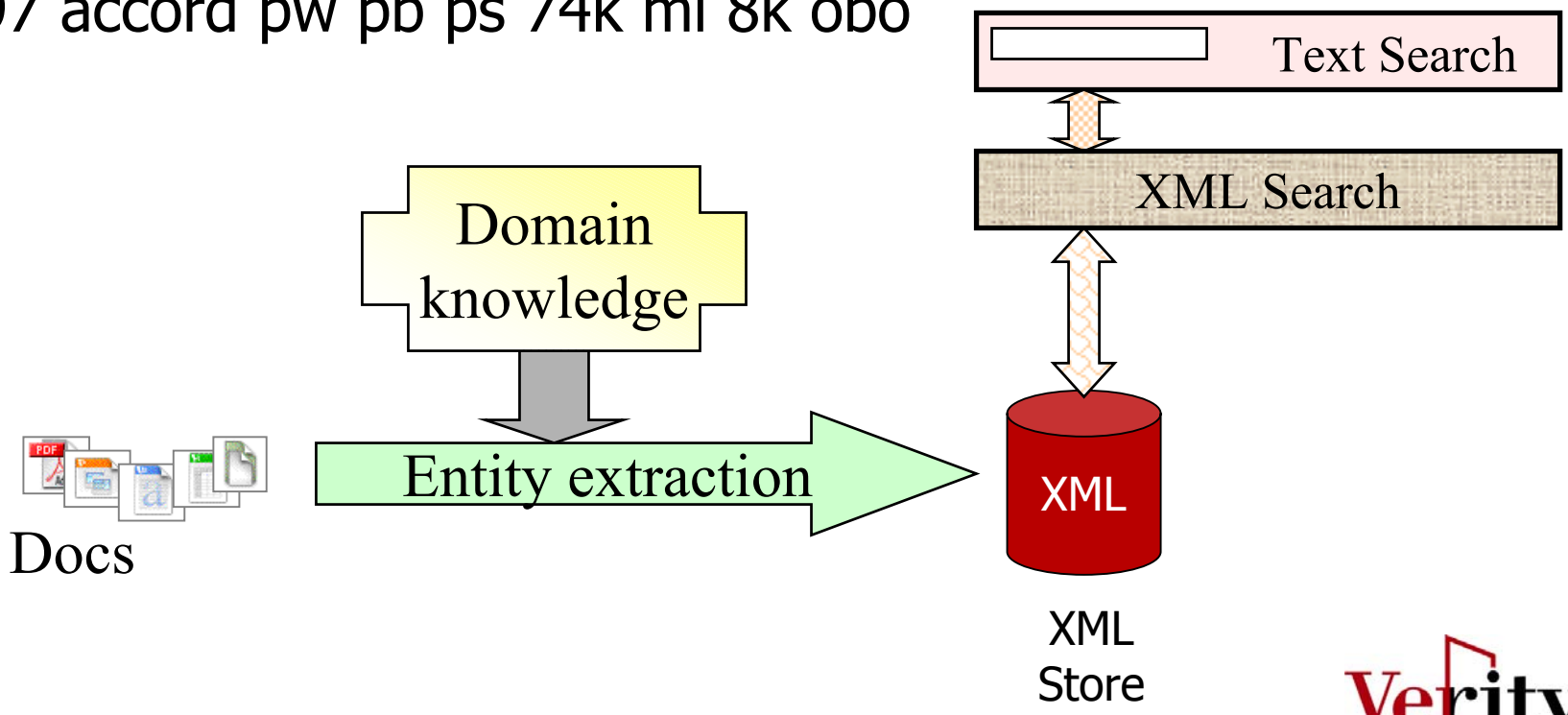
Parental advisory

Warning: Some of the following slides have been produced by Corporate Marketing. The graphic(s) nature of these slides may overwhelm innocent researchers.

Verity™

Domain knowledge in search

- Giants tix pair Saturday game $60
- Sunny 3br 2ba top floor vu 1600+util
- 97 accord pw pb ps 74k mi 8k obo

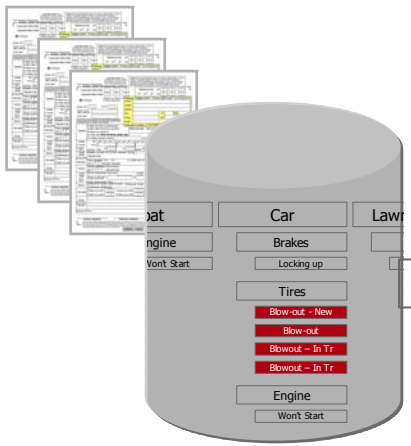


Mining semi-structured text

Product recalls can cost company's millions in profits and loss of brand value.

1. Product "problem tickets" completed at point of origin

2. "Tickets" converted real-time to dynamic, categorized information



3. Smart monitoring identifies critical problems and send alert notice

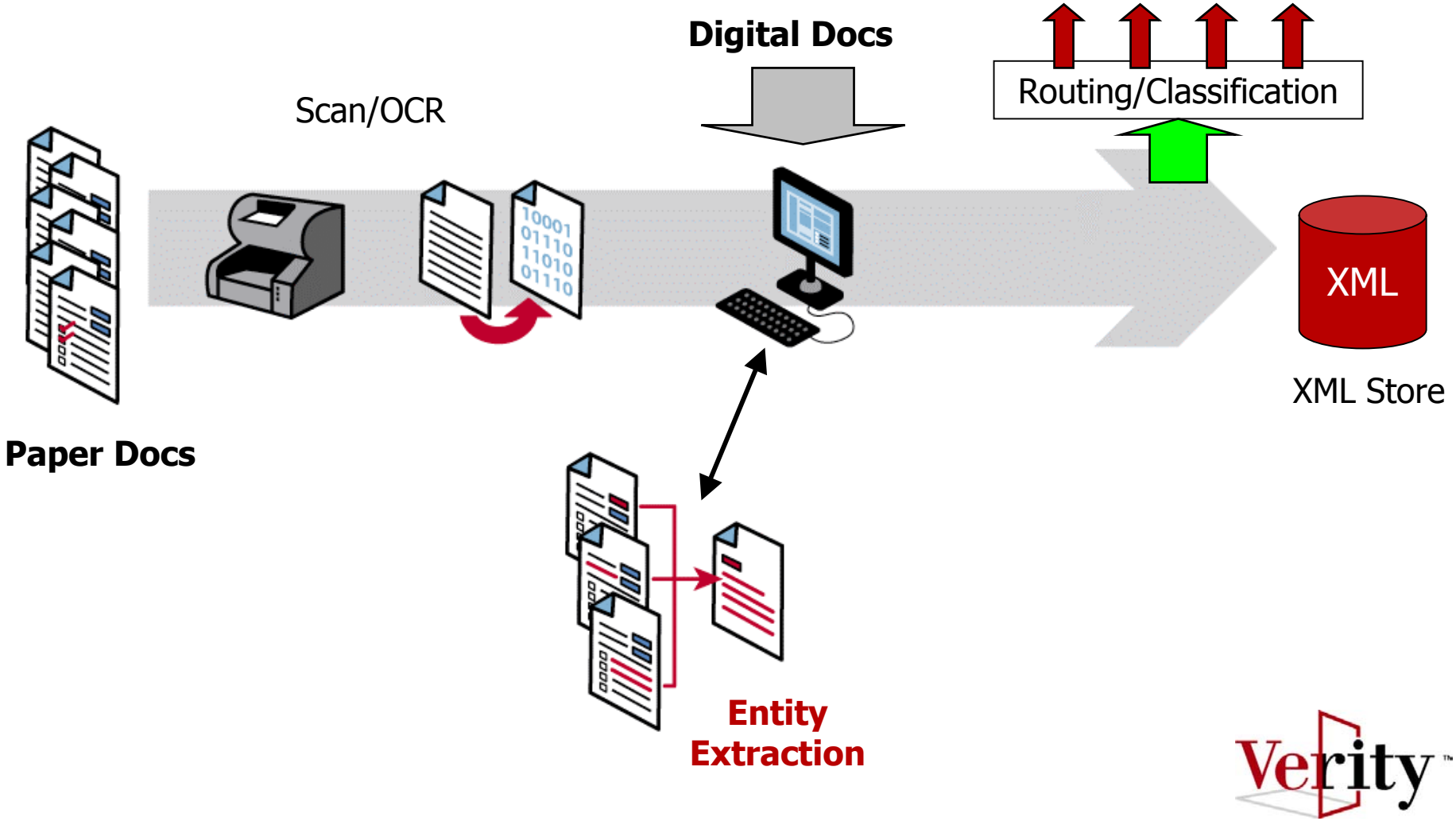
Tires

- Blow-out - New
- Blow-out
- Blowout – In
- Blowout – In Tr

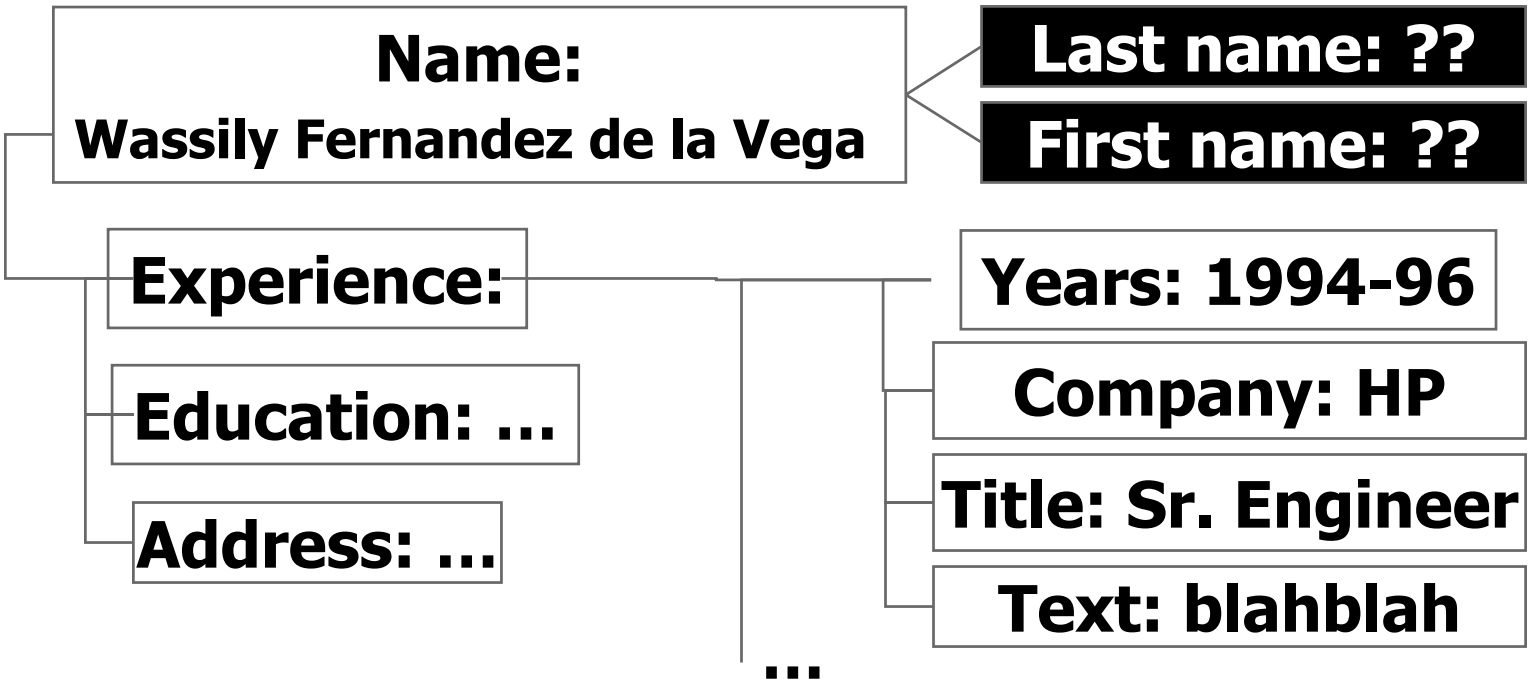
4. Management can research, isolate and rectify problems quickly

Verity™

Resume Extraction/routing



Extracted Resume



More semi-structured user needs

- Find loss-making software companies with revenues of 50-200M last year, and repeated mutual deals with other software companies within 90-day intervals.
  - *Input: SEC Filings, Press Releases*
- Want a family vacation to a warm place with outdoor activities, a few museums and churches, budget 3000 and I prefer to fly Singapore Airlines.
  - *Input: Car, Air, Hotel, Travel info/opinion websites*

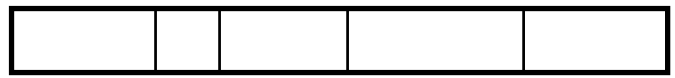
Verity™

What will it take?

- User need parsing
  - Query expression and semantics
- Entity Extraction
  - (Semi-)structure from unstructured text
- Retrieval
  - XML querying on unreliable structure
- Integration
  - Heterogeneous data sources
- It’s ok to get it wrong! (occasionally)

Verity™

Entity extraction and noisy XML

- Static extraction: $60 is a price
  - (for a pair!)
- Relationship extraction
  - Microsoft acquires SAP
- Dynamic extraction
  - Saturday's game is probably 
- Net: you won't get a clean table;
  - Noisy, incomplete XML

Vendors can do the easy portions well.

Verity

Retrieval engine

- Integrate results from heterogeneous sources
  - No presumption of a common DTD
- Need a semi-soft notion of query-doc proximity
  - *Camembert distance measure*
  - Ideally with a time-tested mathematical foundation
    - Linear algebra, probability, …

Verity™

Information integration

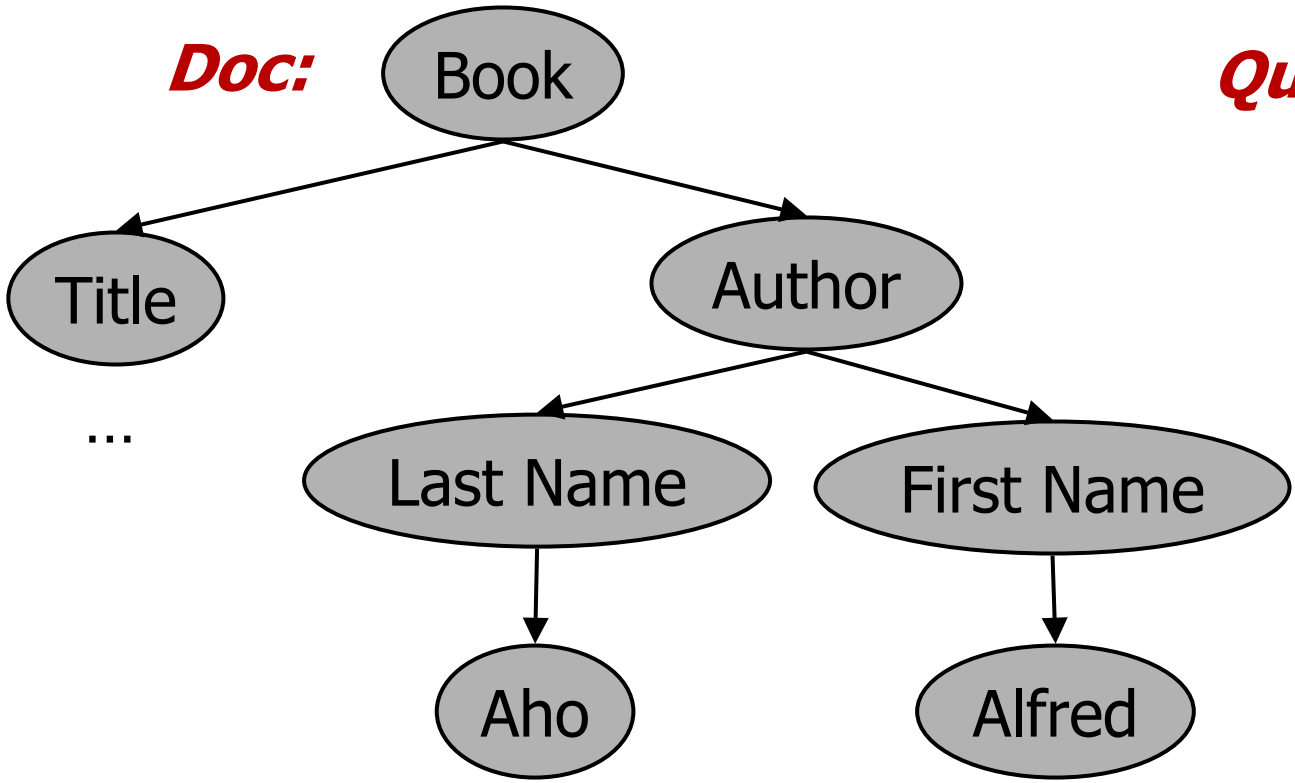
- On-the-fly integration
  - Score/rank aggregation a first step
  - Classic schema normalization
- Query planning and optimization
- Ontology mapping and merging

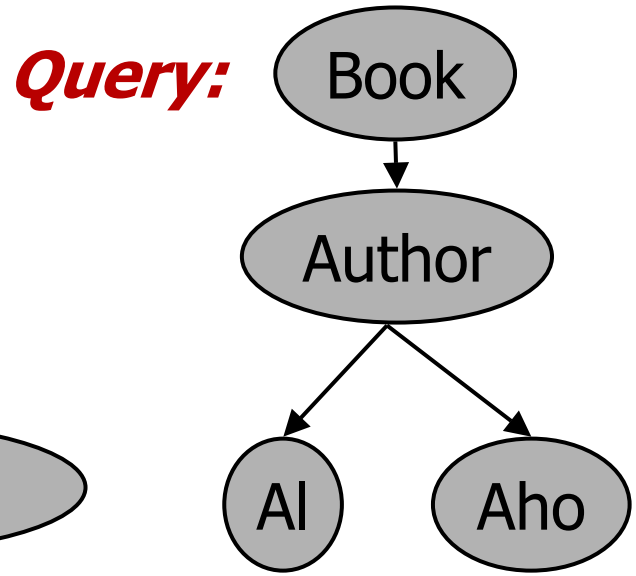
No enterprise vendor really does this well;
Most stop at federation of results.
Web companies beginning to.

Verity™

Technical Tofu

Document and query trees





Verity

Vector spaces for XML
(w/V. Kakade, Stanford)

- Goal: Distance metric between XML queries and/or documents
- Approach: embed queries, docs in a vector space
- Benefits
  - Camembert distance metric
    - Query-by-example
  - Full power of linear algebra available
    - Dimension reduction/LSI
    - Advanced machine learning
      - Support Vector Machines for Classification
      - Mining – can cluster docs, elements etc.

Verity™

Background

- Schlieder and Meuss:
  - One axis for each possible sub-tree of any tree
  - Each query/doc becomes a vector in this space
  - Inner product (cosine) similarity measure
  - Exponentially many sub-trees
- JuruXML (IBM Haifa)
  - Use only root-leaf paths as axes
  - Enhanced to cope with sub-path matches
    - Book/Author/Aho vs. Book/Author/Last_Name/Aho
  - Extra computation outside vector space
    - Good for query scoring
    - Loses other benefits commonplace in text retrieval

Verity™

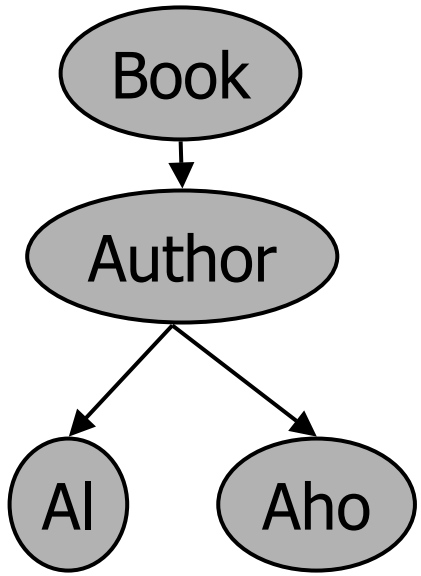
Goals of new encoding

- Avoid exponential blowup
  - Control # axes = index complexity
  - Trade index size for retrieval quality (what's this?)
- Camembert distance: a pure vector inner product
  - Gives us LSI, SVM's and other TLA's
- Net – all text retrieval functionality for XML docs

Verity™

Main ideas

- A filter *F* selects a class of sub-trees
  - E.g., *F* selects all sub-paths of length *k*
  - Sub-trees thus selected are axes
- For *k=2*, this tree generates
  - Book/Author, Author/Al, Author/Aho
- By varying *F* we control
  - Size of index (# axes)
  - Quality of matches
- We almost have everything we wanted
  - Camembert distance measure
  - Full power of linear algebra



Verity

What are we missing?

- Doc=Author/Last_Name/Aho vs. Query=Author/Aho
- How do we capture this in a vector space?
- At index time, compute random substrings of root-leaf paths in docs
  - E.g., Author/Aho is a substring from the Doc
- Do the same for each query
- $\Rightarrow$ Randomized Camembert distance
  - not random docs or queries
  - Analysis possible – what does it say?

Verity

Randomized index

- Reduces Camembert distance computation to a vector space inner product.
- Analysis: *Expected* distance grows with similarity between docs/queries.
  - Gives us the means for search, similar docs, clustering;
  - Use of linear algebra means we can invoke SVM's etc. for document classification/routing.

Verity™

Experiments

- INEX 2002 documents
  - 12000+ articles from IEEE transactions
- Benchmark query suite ("Content Only – CO")
  - Each query specifies a user need
  - Engine must return a sequence of doc elements
- Known judgements for each element in the corpus
  - Relevance – how relevant to the user need
  - Coverage – too deep? too shallow?

  ⇒ Composite quality measure for each query/element pair

Verity™

Experiments

- Encouraging; perhaps not statistically significantly so.
  - Index blowup, retrieval quality compared
  - Quality can be improved using our techniques, relative to treating each doc as a "bag of words".
- Classification and clustering "out of the box"
  - On some data, out-perform tailor-made XML clustering/classification algorithms.
- Kakade MS Thesis, Stanford CS Dept, June 2004.
- Much more detailed experimentation is needed.

Verity™

Summary

Verity™

Research challenges

- Step up on entity extraction – dynamic etc.
- Retrieval platforms
  - Unreliable, noisy structure
  - Semi-soft ("camembert") distance measures
- Information integration
- Pilot vertical domain applications
  - Full semantic web too daunting
  - End-users won't annotate – incentive structure?

Verity™