AUTOCOMPLETION FOR STRUCTURED QUERYING

Database Research Group
University of Michigan, Ann Arbor

Arnab Nandi & H.V. Jagadish

Explosion in Structured Data

Size of data is
increasing…
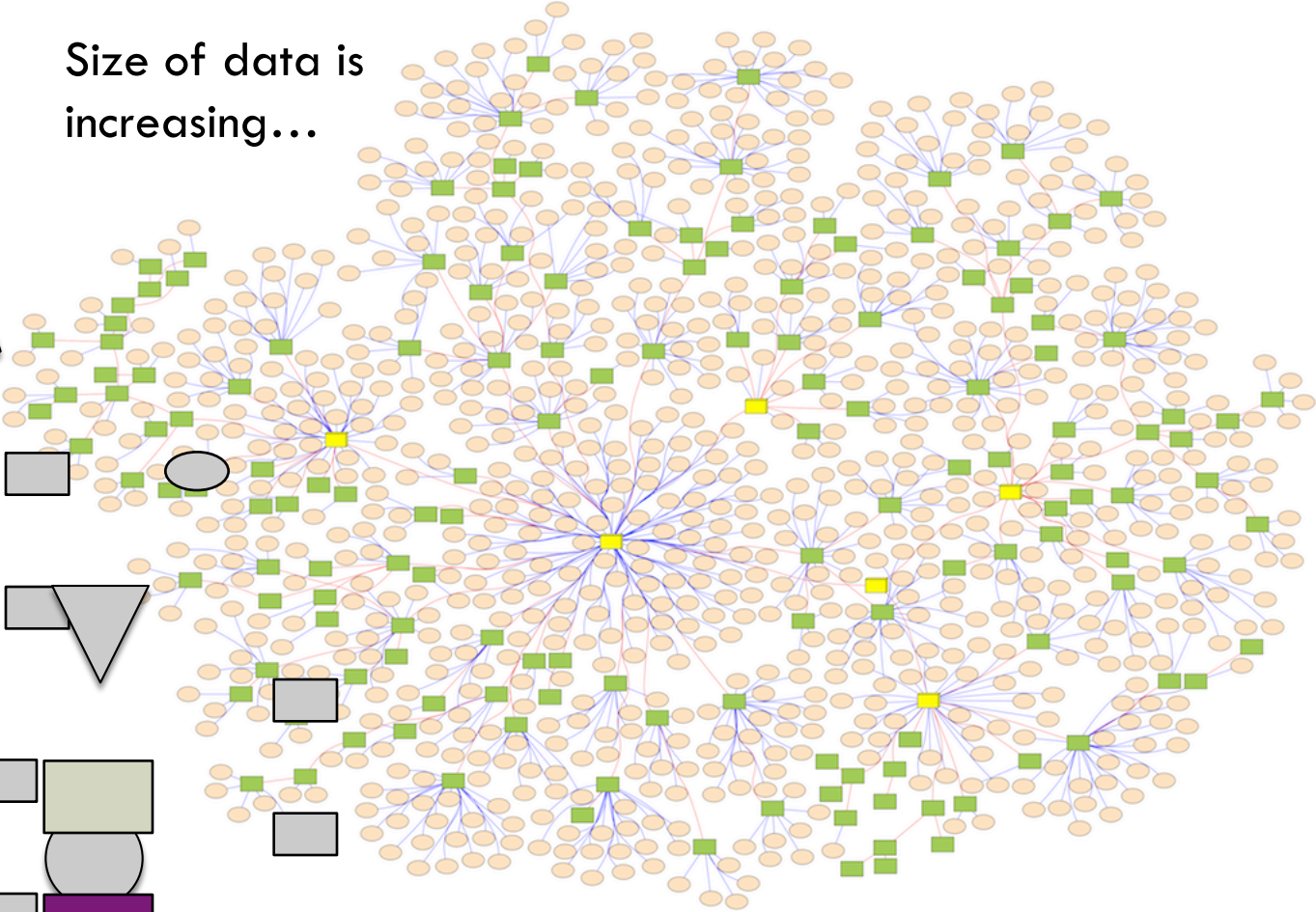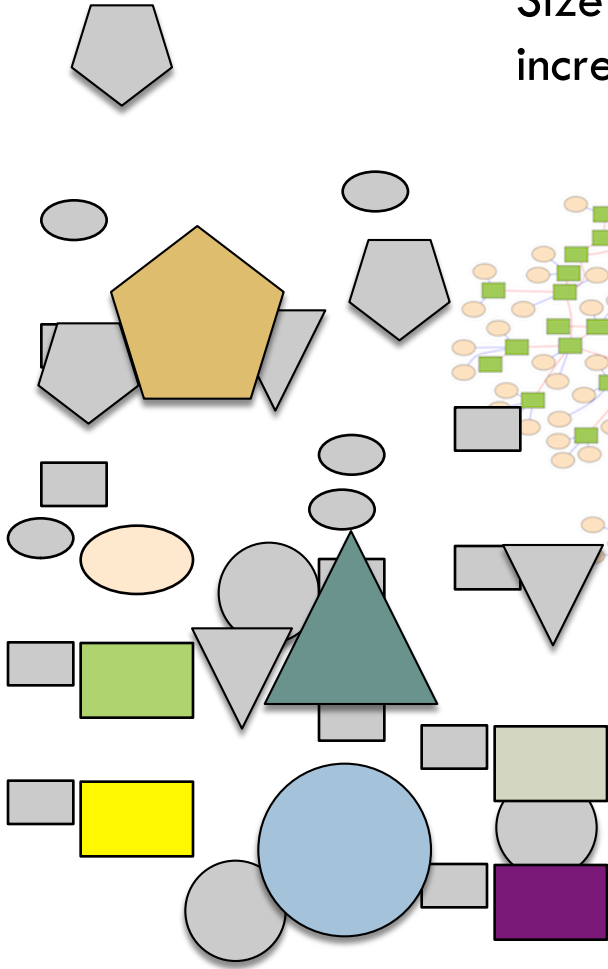
…and so is its complexity

More data, More querying

- *What's new?*
  - Explosion of **structured** data
  - **Humans** as **primary** consumer of data

- Way we interact with data needs rethinking
  - Database Usability

Querying Structured Data

```
SELECT * FROM
(SELECT Top 1 t1.farmno, avg(t1.Mort0Pct) * 100 AS SixFlockAvgMort0,
min((AvgAdjustedPrimeCost2 - AdjustedPrimeCost) * 100) AS APCRating FROM
(SELECT top 6 zst.FarmNo, (cast(zst.MortalityWeek0 AS float) /
cast(zst.HeadPlaced AS float)) AS Mort0Pct, zst.settledate
FROM zrstSettle zst
WHERE zst.MinGuaranteeFlockExcludeFlag = 0
AND zst.ApprovedFlag = 1
AND zst.farmno = '2073'
ORDER BY zst.settledate DESC) t1
INNER JOIN zrstSettle ON t1.FarmNo = zrstSettle.FarmNo
WHERE zrstSettle.farmno = '2073' AND
zrstSettle.MinGuaranteeFlockExcludeFlag = 0
AND zrstSettle.ApprovedFlag = 1
GROUP BY t1.farmno, zrstSettle.settledate
ORDER BY SixFlockAvgMort0 DESC) t2
WHERE farmno = '2073'
ORDER BY t2.SixFlockAvgMort0 DESC
```

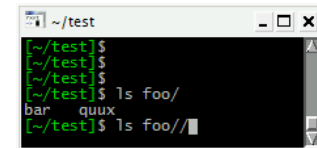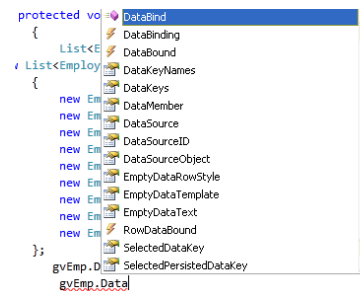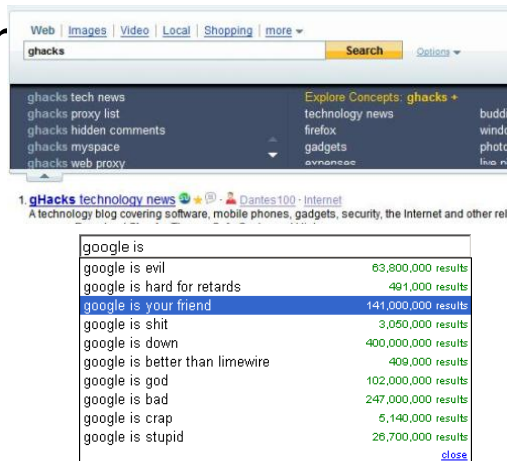HARD

Querying Structured Data
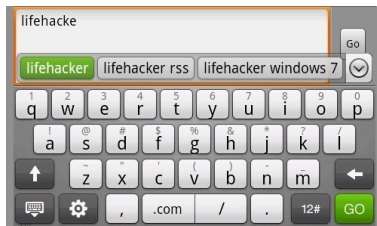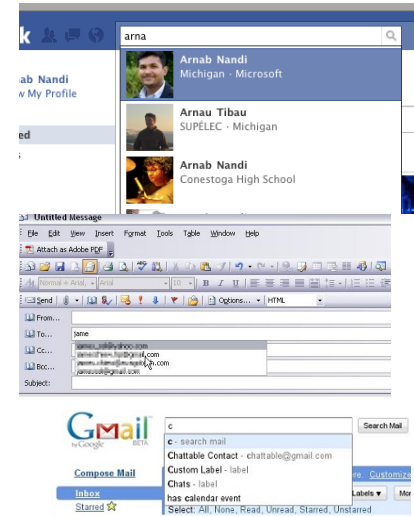
something simple **Search**

EASY

Auto-completion

etion

iance

□ Mechanism that

□ Autocompletion is everywhere

   □ Command prompts

   □ Code Editors

   □ Email Clients / Addressbooks

   □ Mobile phones
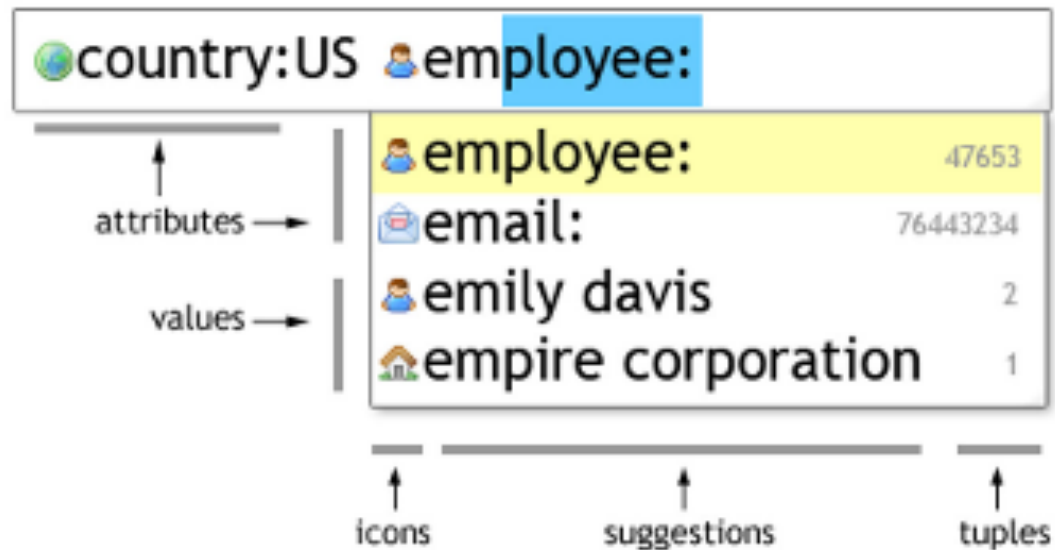
   □ Search

Structured Data Enriches Querying

- The **type** of data matters
  - *movie:titanic vs "titanic"*
- **Multiple** independent criteria to identify entity of interest
  - *movie:titanic actor:leonardo*
- Criteria may be related only through non-obvious entities
  - *actor:leonardo actor:kate*

Completing Structured Queries

☐ Structured Autocompletion Interface

Demo

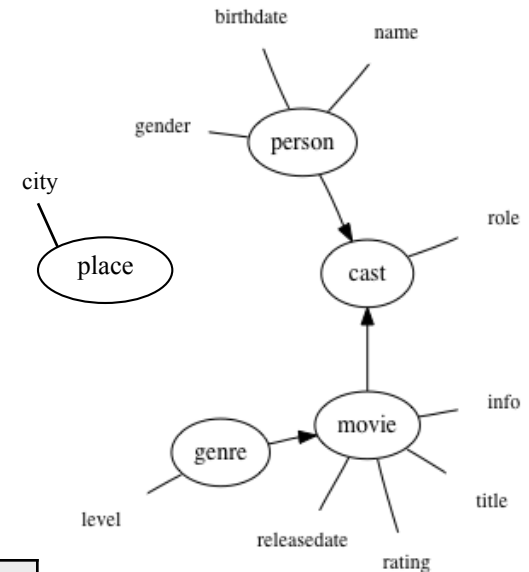- Autocompletion Demo (alt link)

Challenges

- ☐ Schema validation

- ☐ Context-sensitive Suggestions

- ☐ Latency

Schema Validation

- **Some queries are not possible**
  - There is no join connecting
    city and *movie title*

- **Construct adjacency matrix**
  - O(1) lookup

	City	Title	Name
City	✔		
Title		✔	✔
Name			✔

birthdate   name

gender   person

city

place   cast   role

genre → movie   info

level   releasedate   rating   title

Context-sensitive Suggestions

- Rerank suggestions based on existing entered terms.
  - Use both schema and instance information
- Construct conjunctive indexes for all token pairs, triplets, …
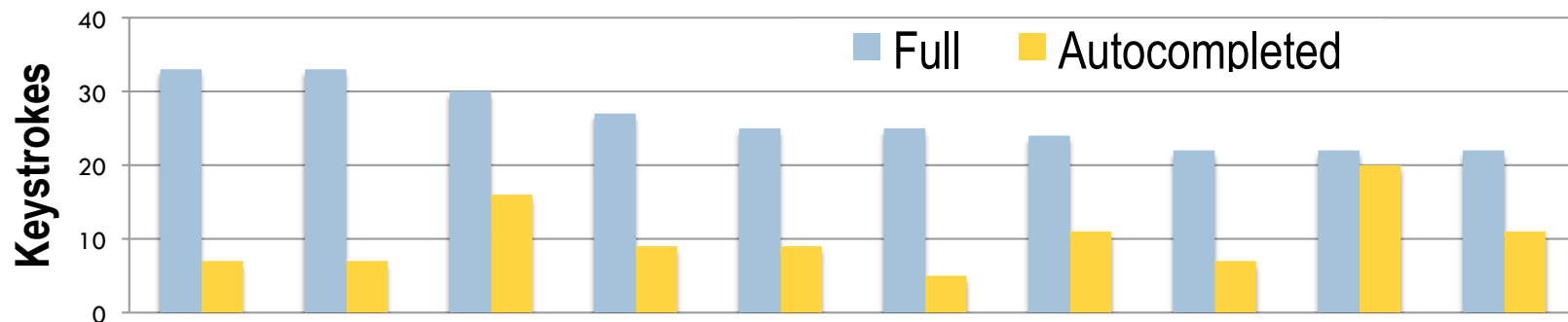 - Markov independence between terms (data dependent)

Latency

- Network + Search cannot keep up with each keystroke

- Ensure each suggestion takes < 100ms to seem instantaneous

- Client-side Index
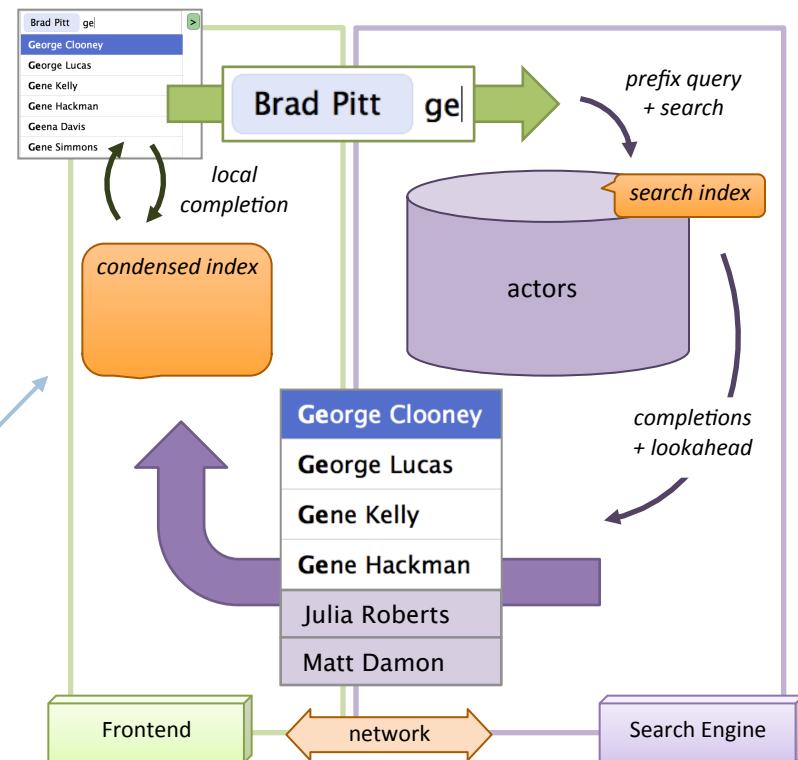
Anecdotal Results

An average of 60% Keystrokes Saved over a workload of 10 multi-token queries from AOL query logs on IMDB dataset, with PageRank and index of 1.9M names, 62K conjunctions

Questions?