Graph Querying Meets HCI: State of the Art and Future Directions

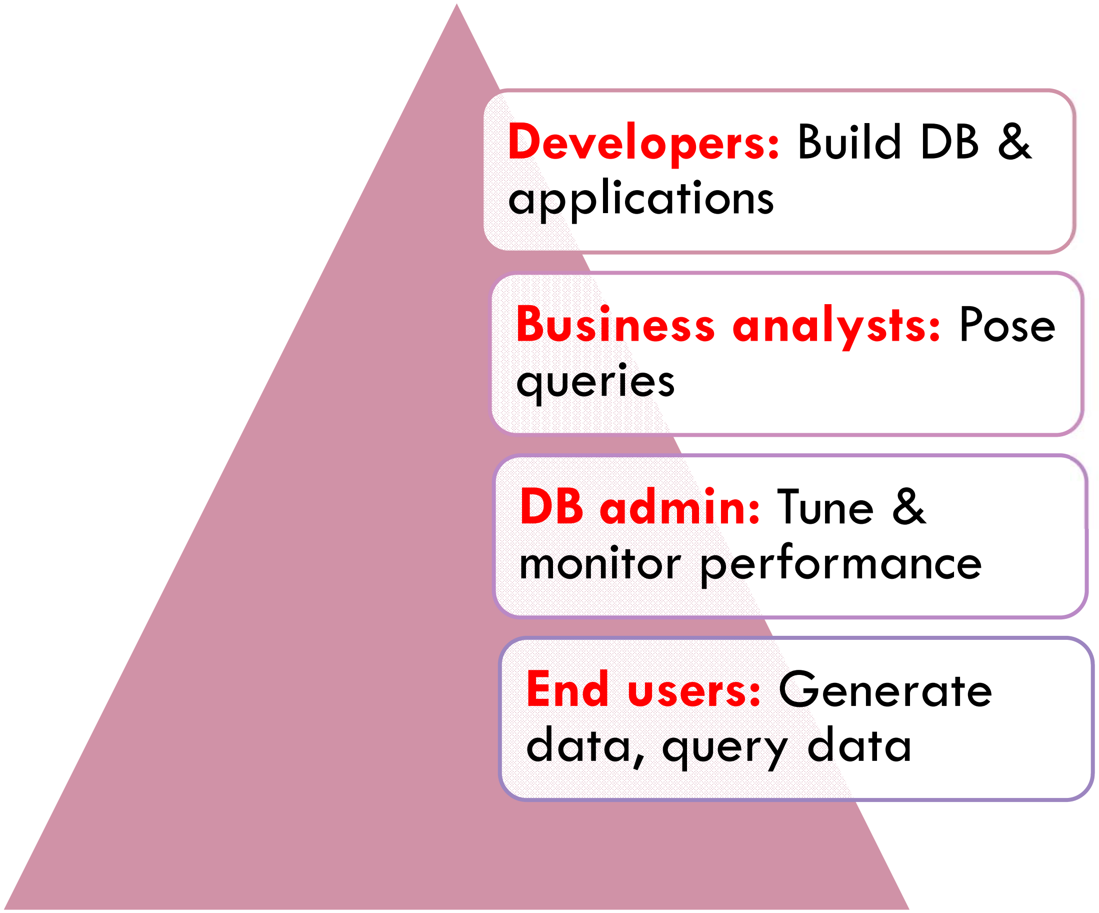
Sourav S Bhowmick
Nanyang Technological Univ
Singapore

Byron Choi
Hong Kong Baptist Univ
Hong Kong

Chengkai Li
The Univ of Texas at Arlington
USA

First Generation Data Management

Driven primarily by enterprises to store and query data



Developers: Build DB & applications

Business analysts: Pose queries

DB admin: Tune & monitor performance

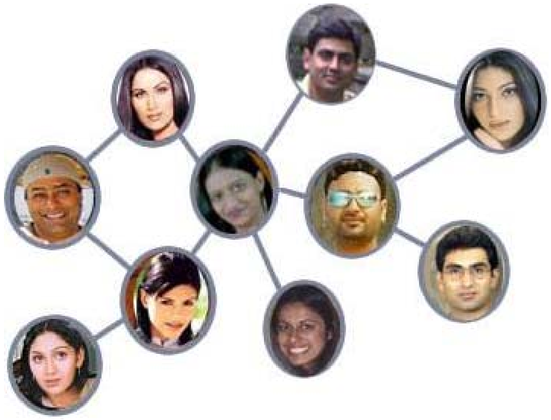
End users: Generate data, query data

```
ELECT customers, orde
FROM customers, orders
WHERE customers.id = orders.id = products.id
AND orders.product_id = products
ORDER BY customers.name,
REATE TABLE customer_profile(
rofile_id INTEGER,
name VARCHAR(50),
prof_factor INTEGER,
date_start DATE,
date_end DATE,
create_by VARCHAR(10),
ified_by VARCHAR (10));
celist SET price = 0.00
ULL;
```

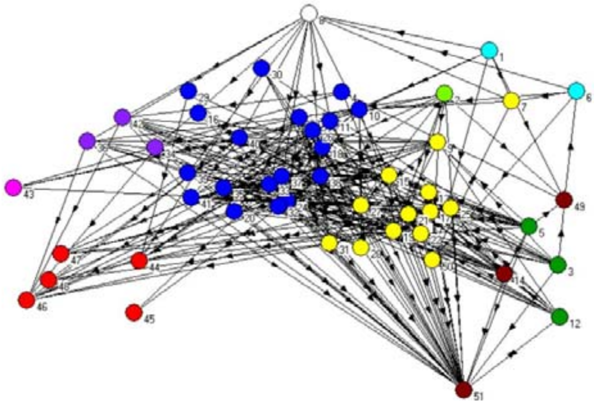
- **Performance**
- **Functionality**

Abadi et al. **The Beckman report on database research.** Commun. ACM 59(2): 92-99 (2016)

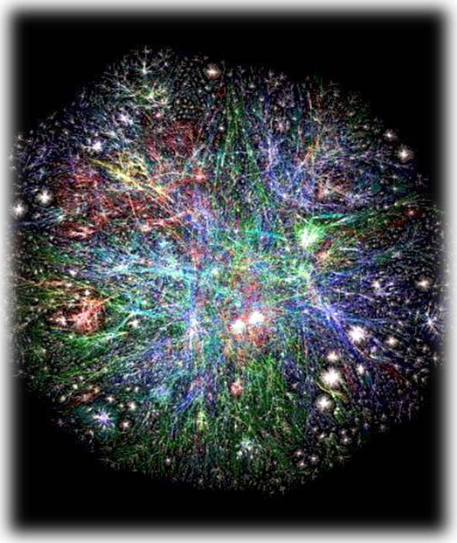
Emergence of Network Data



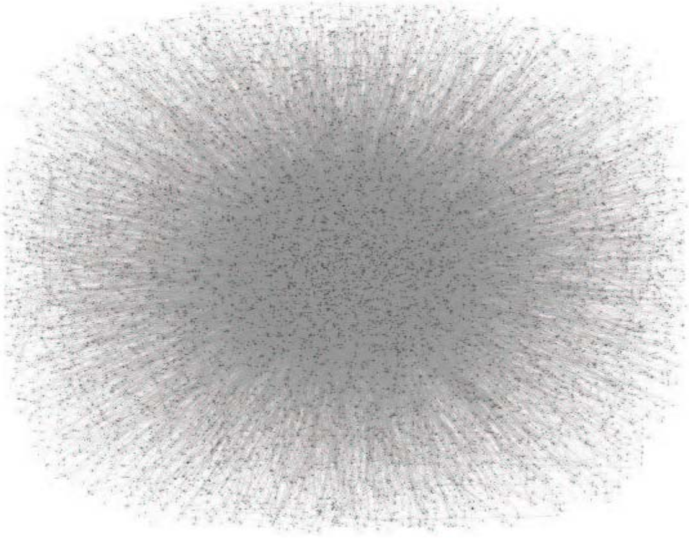
Social network



Ecological network



WWW

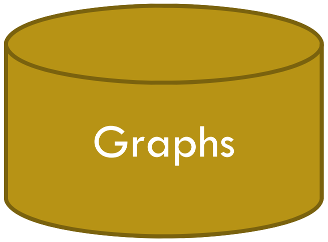


Human PPI network

The emergence of network maps:

Movie Actor Network, 1998;
World Wide Web, 1999.
C elegans neural wiring diagram 1990
Citation Network, 1998
Metabolic Network, 2000;
PPI network, 2001

Querying Graphs



Graphs

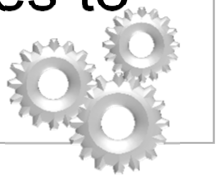
A large set of small/medium-sized graphs
A large graph/network
Massive graph

Query Formulation

- •Formal query language
- SPARQL, Cypher

Query Processing

- •Efficient algorithms and optimization techniques to process queries “quickly”



Fifth Generation Data Management

Data management has democratized

biologist

chemist

End Users: Generator, processor, and consumer

DB illiterate

Increasingly complex data and computation

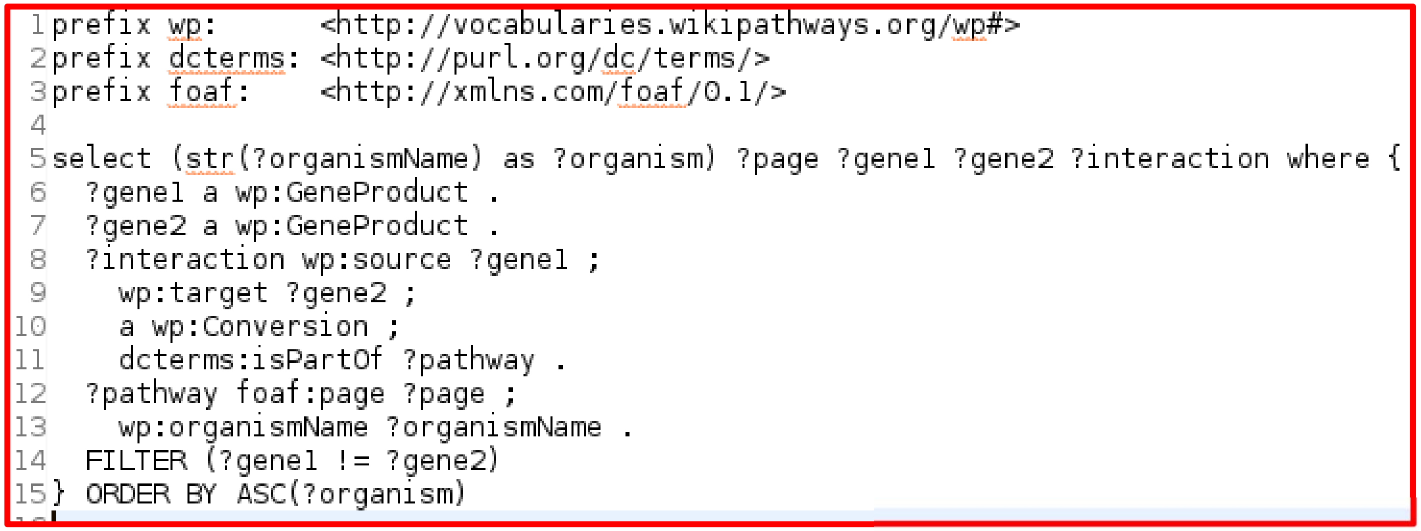
Resides everywhere

social scientist

journalist

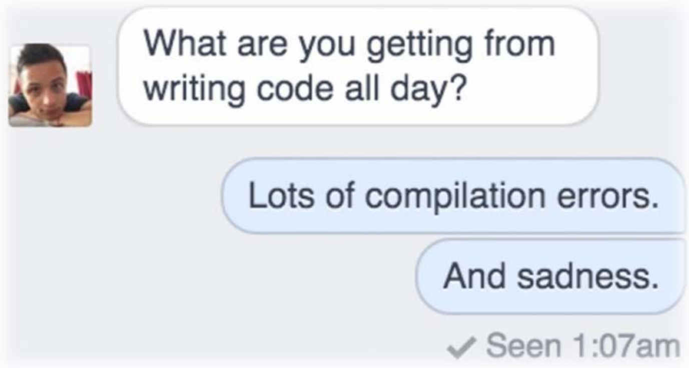
Abadi et al. **The Beckman report on database research.** Commun. ACM 59(2): 92-99 (2016)

Querying Graphs: The First Generation Approach



```
prefix wp:      <http://vocabularies.wikipathways.org/wp#>
prefix dcterms: <http://purl.org/dc/terms/>
prefix foaf:    <http://xmlns.com/foaf/0.1/>

select (str(?organismName) as ?organism) ?page ?gene1 ?gene2 ?interaction where {
  ?gene1 a wp:GeneProduct .
  ?gene2 a wp:GeneProduct .
  ?interaction wp:source ?gene1 ;
    wp:target ?gene2 ;
    a wp:Conversion ;
    dcterms:isPartOf ?pathway .
  ?pathway foaf:page ?page ;
    wp:organismName ?organismName .
  FILTER (?gene1 != ?gene2)
} ORDER BY ASC(?organism)
```

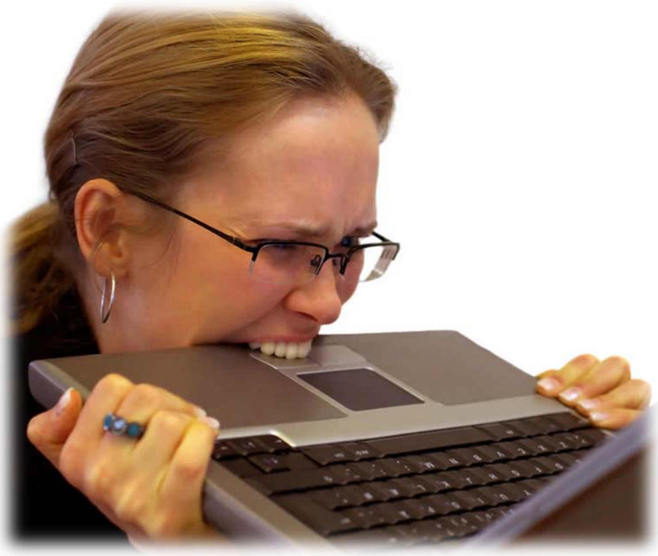


What are you getting from writing code all day?

Lots of compilation errors.

And sadness.

✓ Seen 1:07am



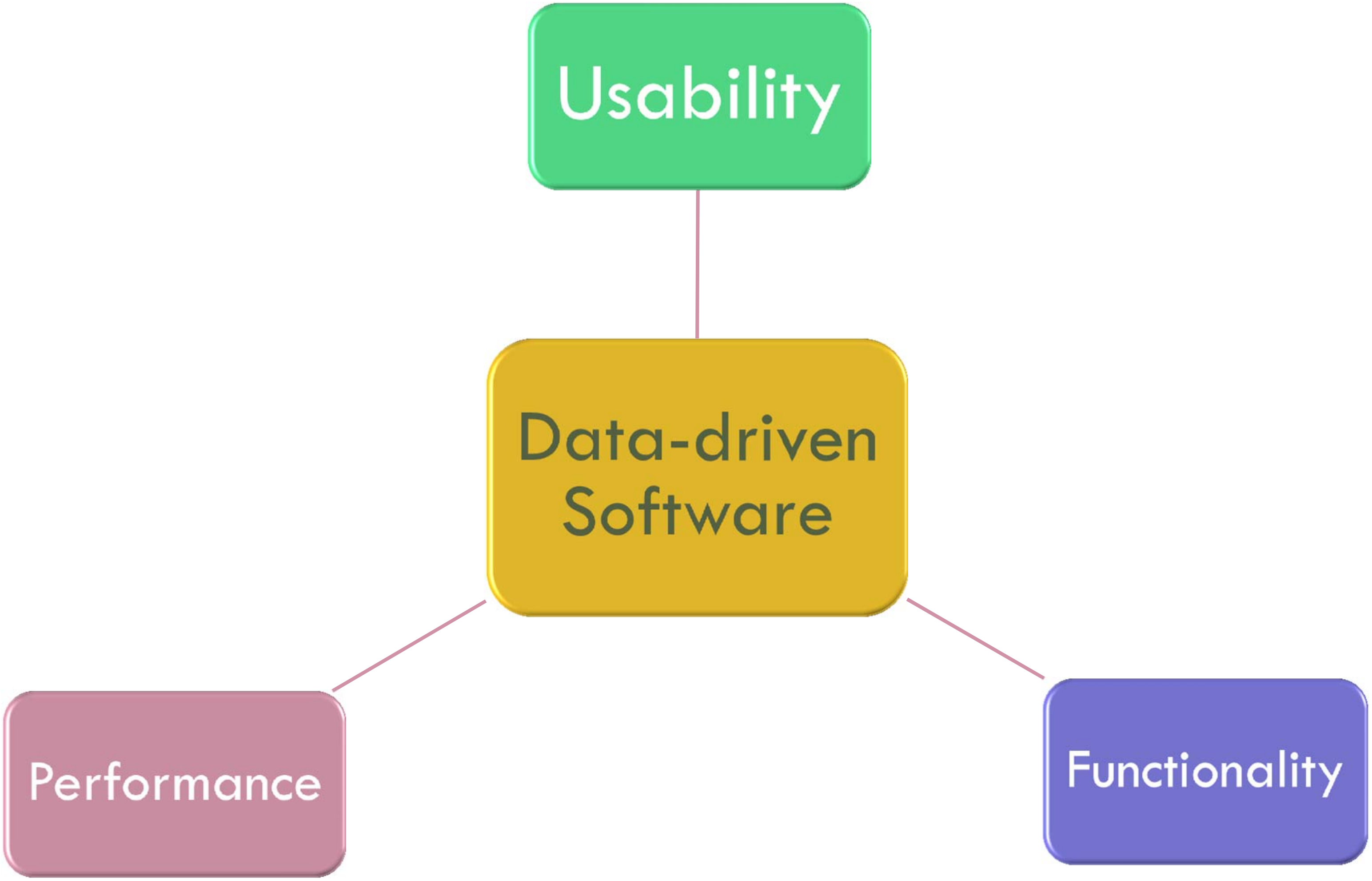
Reality Check!

Reality

> “ Thirty years of research on query languages can be summarized by: we have moved from SQL to XQuery. At best we have moved from one declarative language to a second declarative language with roughly the same level of expressiveness. It has been well documented that end users will not learn SQL; rather SQL is notation for professional programmers.

The Lowell Database Research Self-Assessment,
Communication of the ACM (May 2005)

Usability Matters!



Usability

Data-driven Software

Performance

Functionality

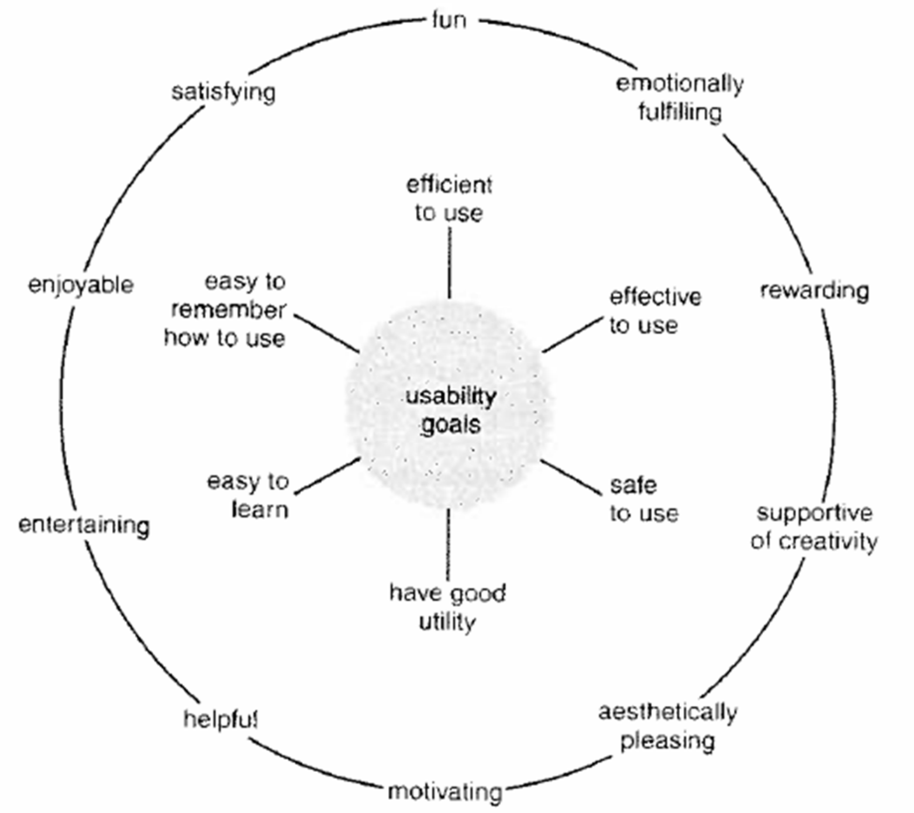
Usability [Preece et al.]

What is it?

How well users can use the system's functionality

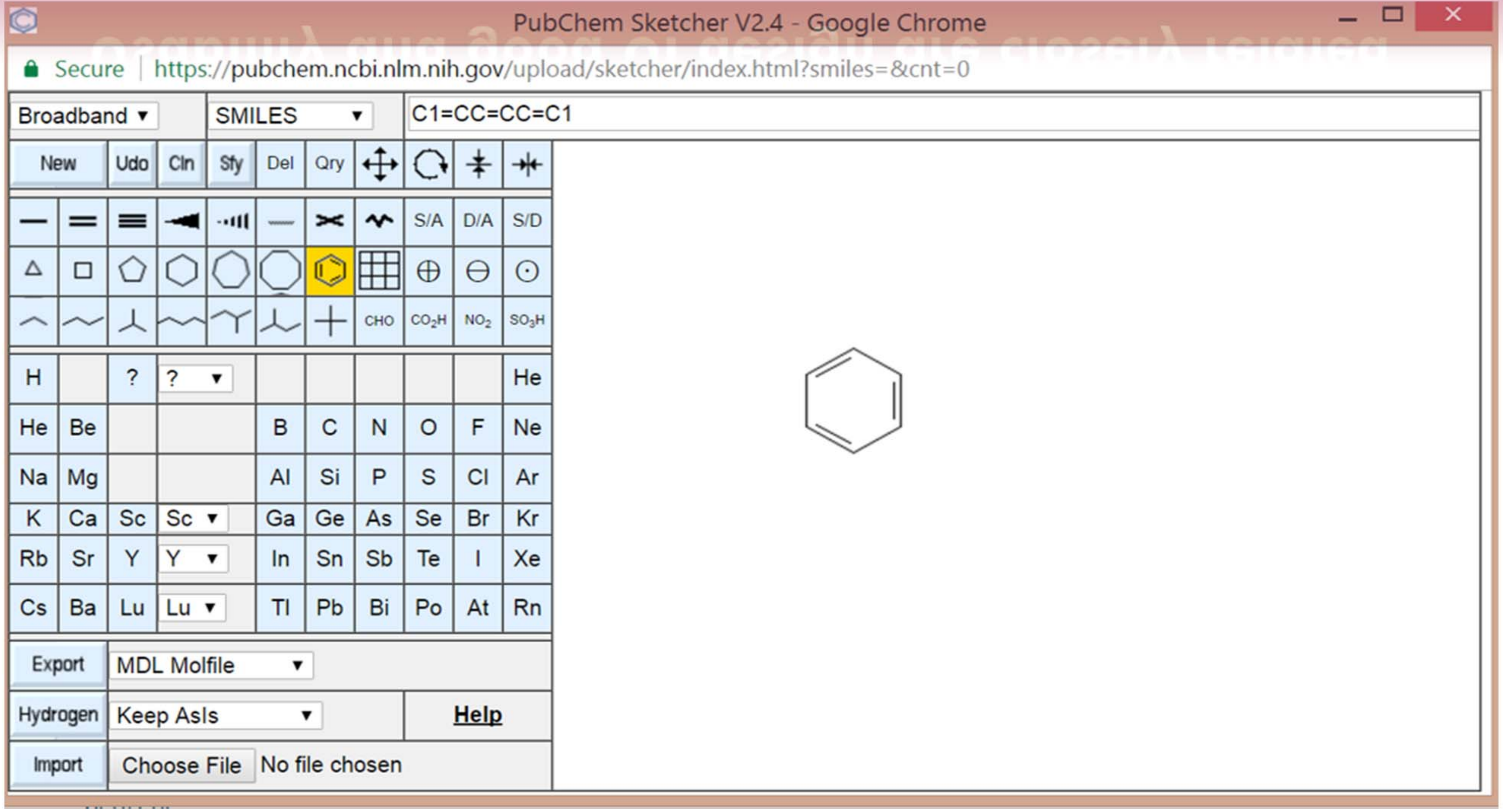
Dimensionality

- Learnability: is it easy to learn?
- Efficiency: once learned, is it fast to use?
- Memorability: is it easy to remember what you learned?
- Errors: are errors few and recoverable?
- Satisfaction: is it enjoyable to use?



Visual Graph Querying

Usability and good UI design are closely related



PubChem Sketcher V2.4 - Google Chrome

Secure | https://pubchem.ncbi.nlm.nih.gov/upload/sketcher/index.html?smiles=&cnt=0

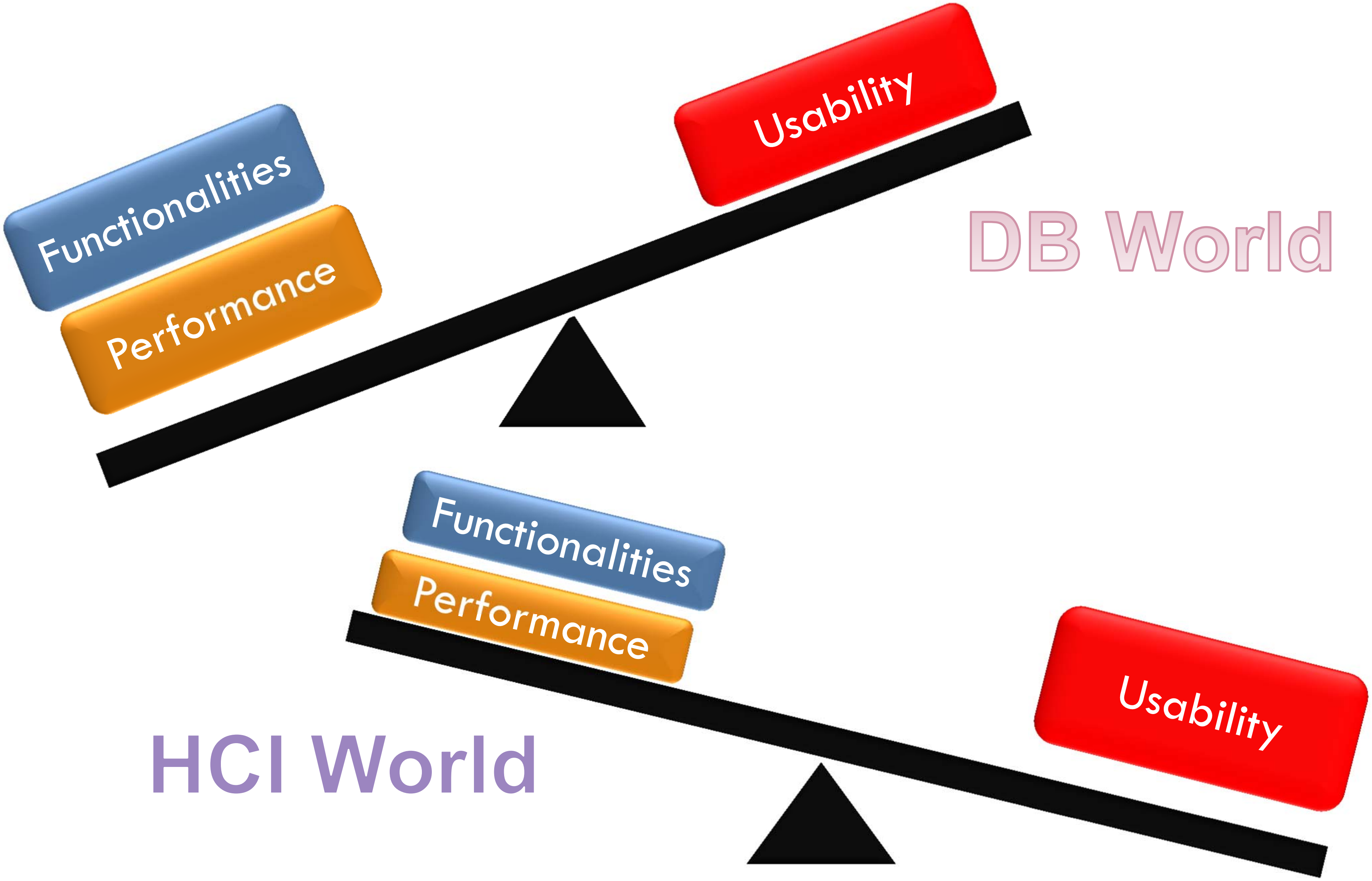
Broadband ▾ SMILES ▾ C1=CC=CC=C1

Export: MDL Molfile ▾

Hydrogen: Keep AsIs ▾

Import: Choose File No file chosen

Different Worlds



Functionalities

Performance

Usability

DB World

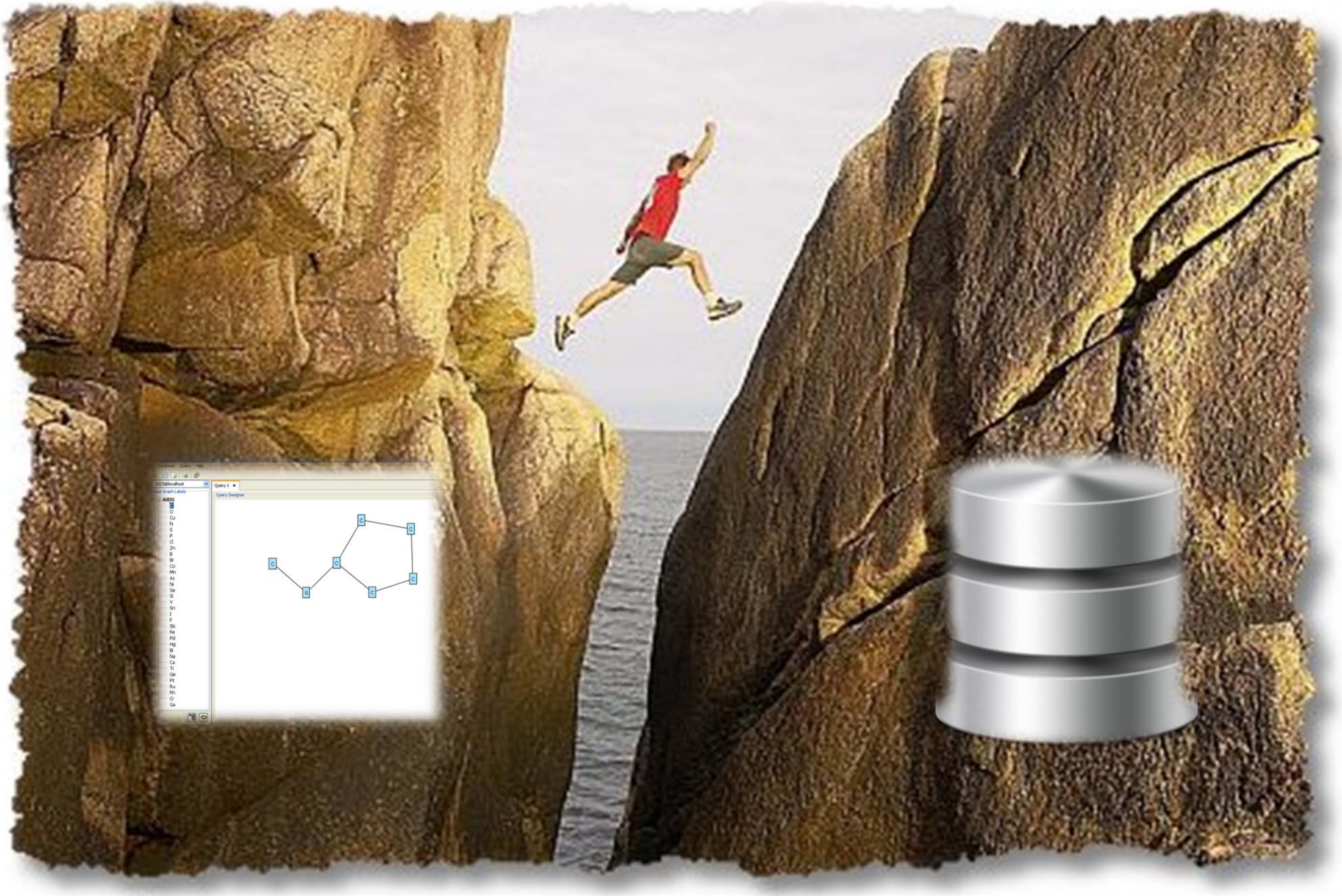
Functionalities

Performance

Usability

HCI World

The Chasm for 40+ Years



Graph Querying Meets HCI

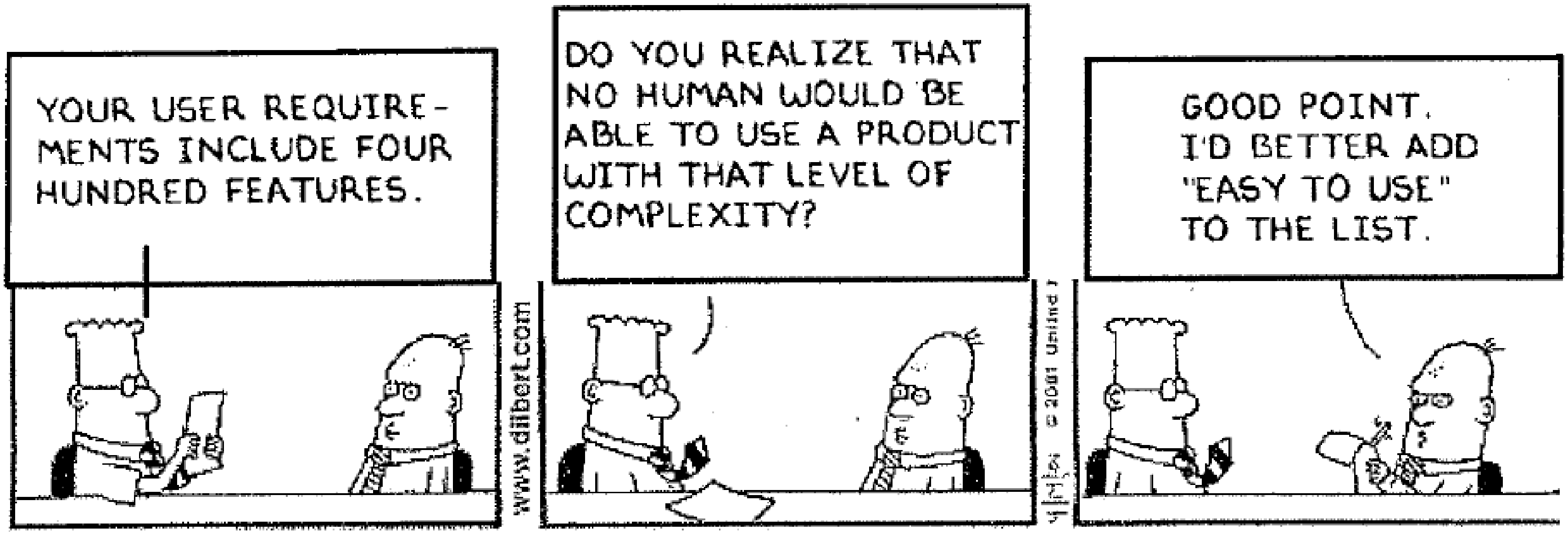
HCI

DB

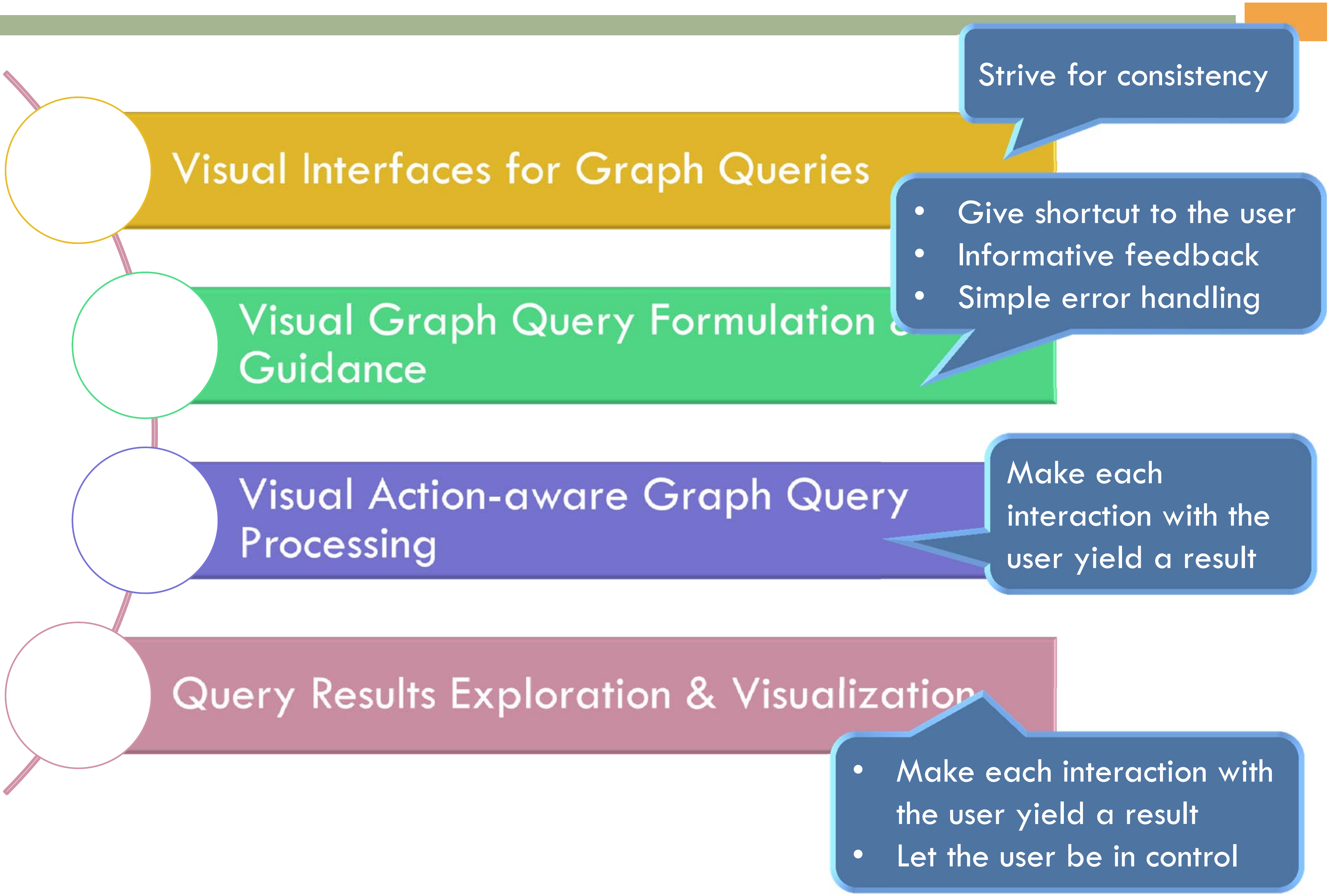
`

Lessons from HCI: Schneiderman’s 8 Golden Rules

- Strive for consistency.
- Give shortcuts to the user.
- Offer informative feedback.
- Make each interaction with the user yield a result.
- Offer simple error handling.
- Permit easy undo of actions.
- Let the user be in control.
- Reduce short-term memory load on the user.


YOUR USER REQUIREMENTS INCLUDE FOUR HUNDRED FEATURES.
DO YOU REALIZE THAT NO HUMAN WOULD BE ABLE TO USE A PRODUCT WITH THAT LEVEL OF COMPLEXITY?
GOOD POINT. I'D BETTER ADD "EASY TO USE" TO THE LIST.
www.dilbert.com
© 2001 United

Tutorial Overview



Visual Interfaces for Graph Queries

Strive for consistency

Visual Graph Query Formulation & Guidance

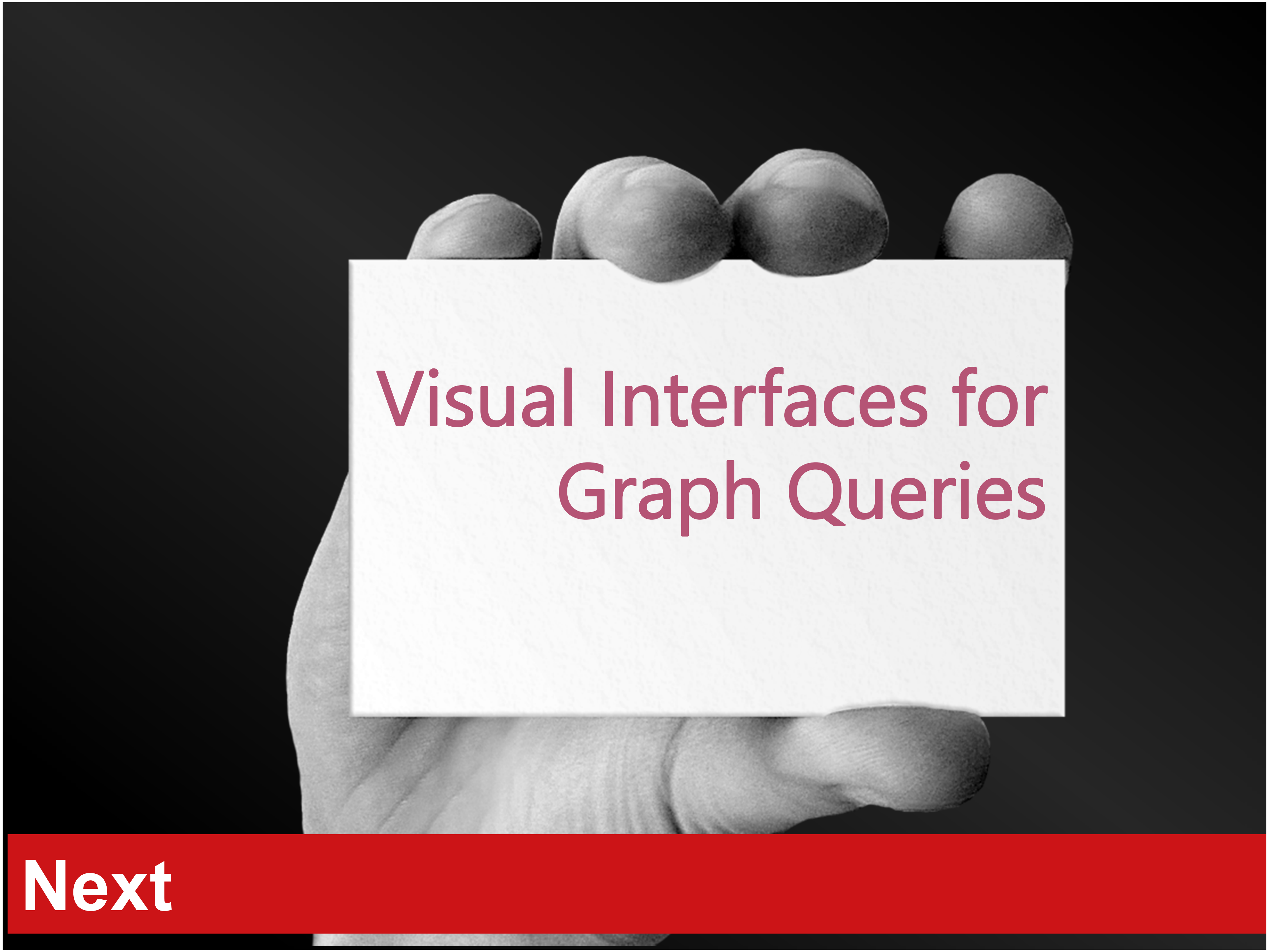
- Give shortcut to the user
- Informative feedback
- Simple error handling

Visual Action-aware Graph Query Processing

Make each interaction with the user yield a result

Query Results Exploration & Visualization

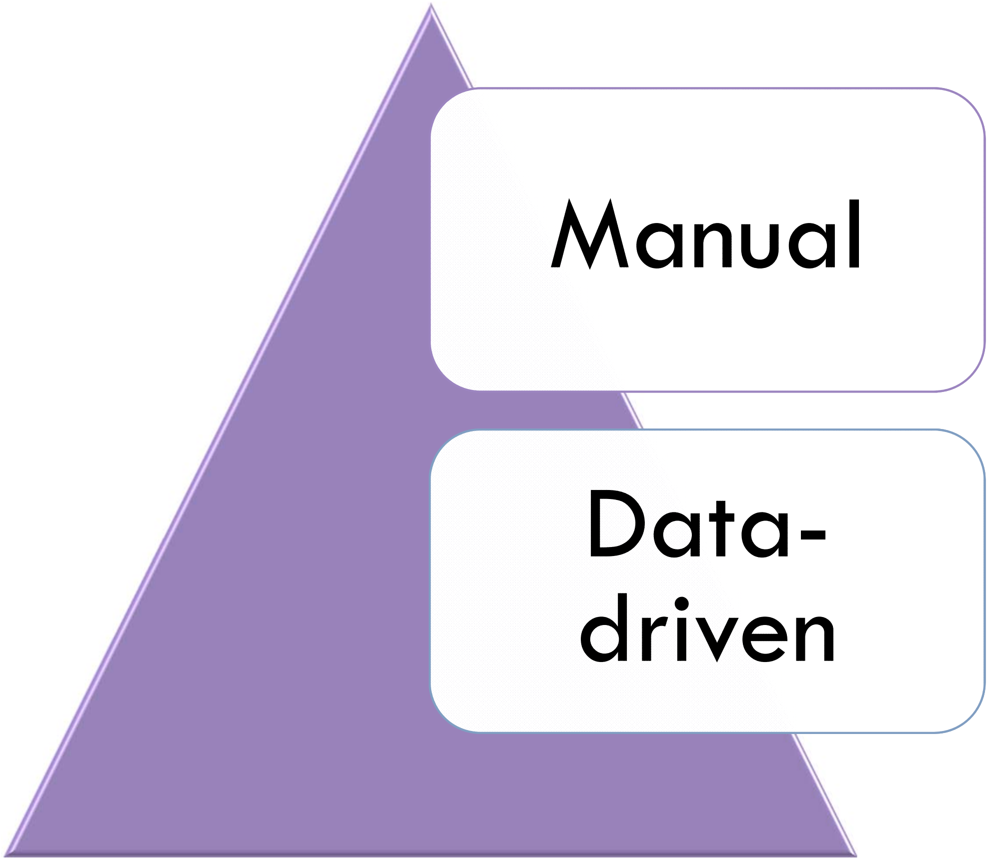
- Make each interaction with the user yield a result
- Let the user be in control



Visual Interfaces for Graph Queries

Next

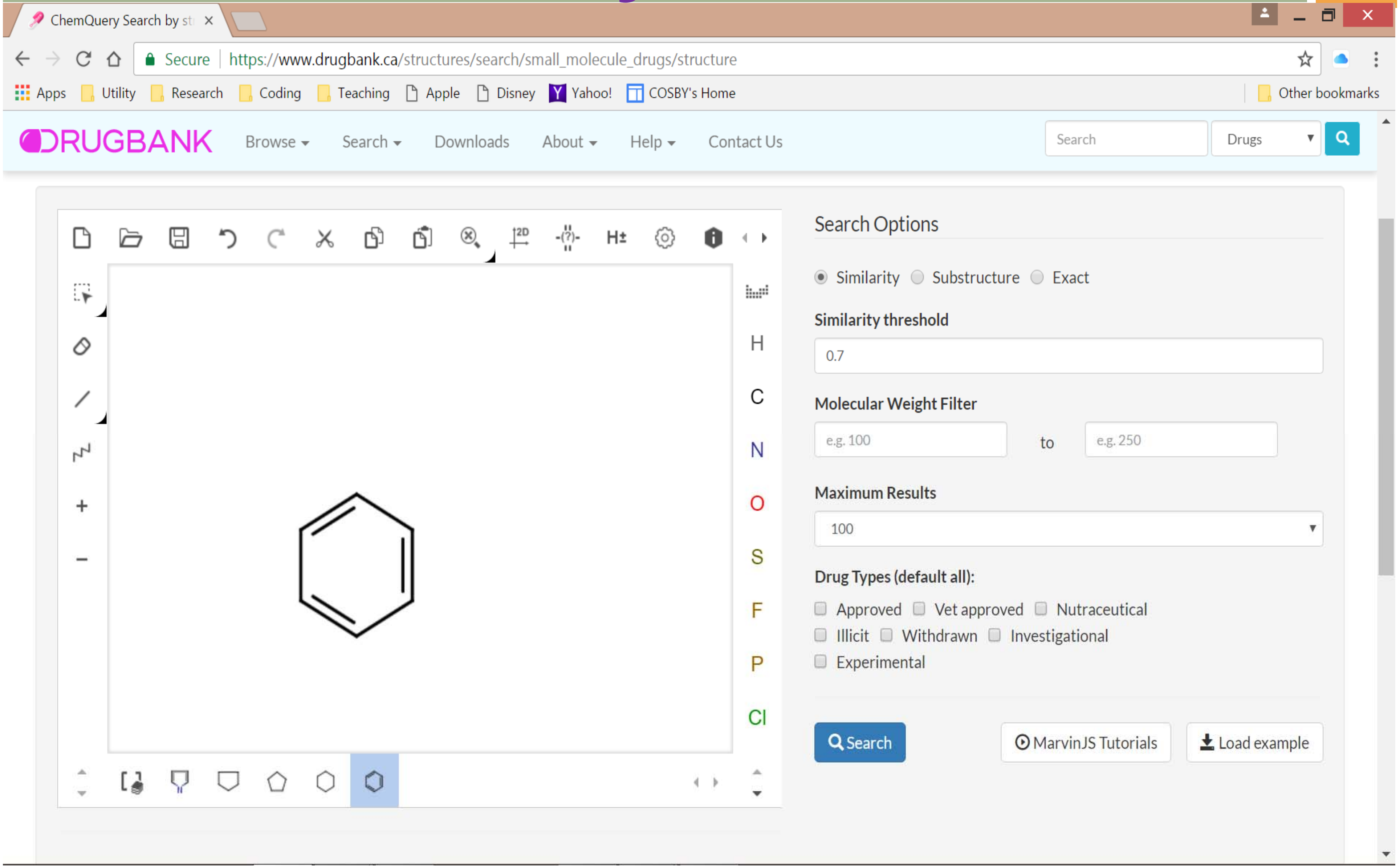
Visual Graph Query Interfaces


Manual

Data-driven

Functionalities
vs
Aesthetics

Manual Visual Graph Query Interfaces



Search Options

● Similarity ○ Substructure ○ Exact

Similarity threshold

0.7

Molecular Weight Filter

e.g. 100 to e.g. 250

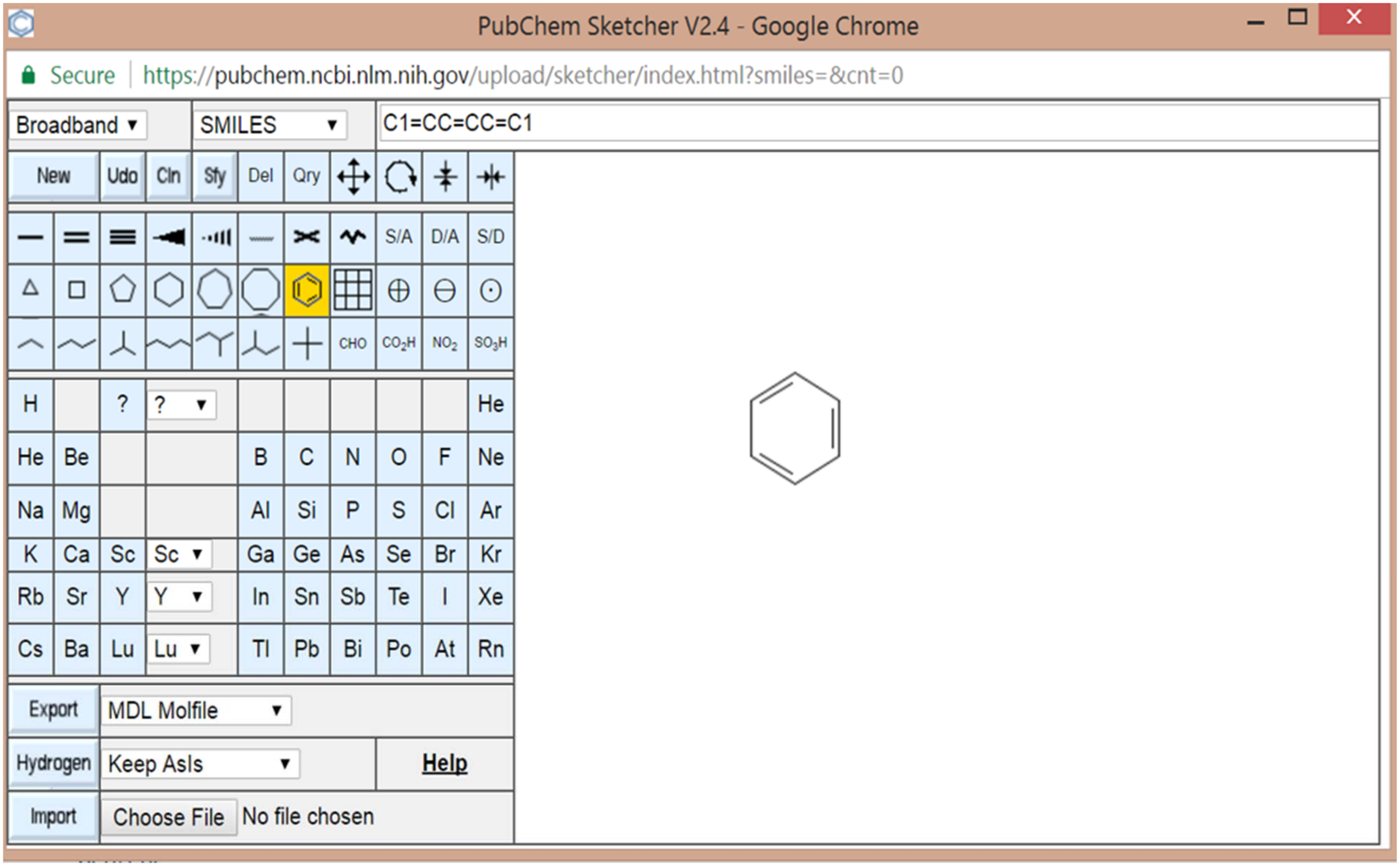
Maximum Results

100

Drug Types (default all):

□ Approved □ Vet approved □ Nutraceutical
□ Illicit □ Withdrawn □ Investigational
□ Experimental

Manual Visual Graph Query Interfaces



PubChem Sketcher V2.4 - Google Chrome

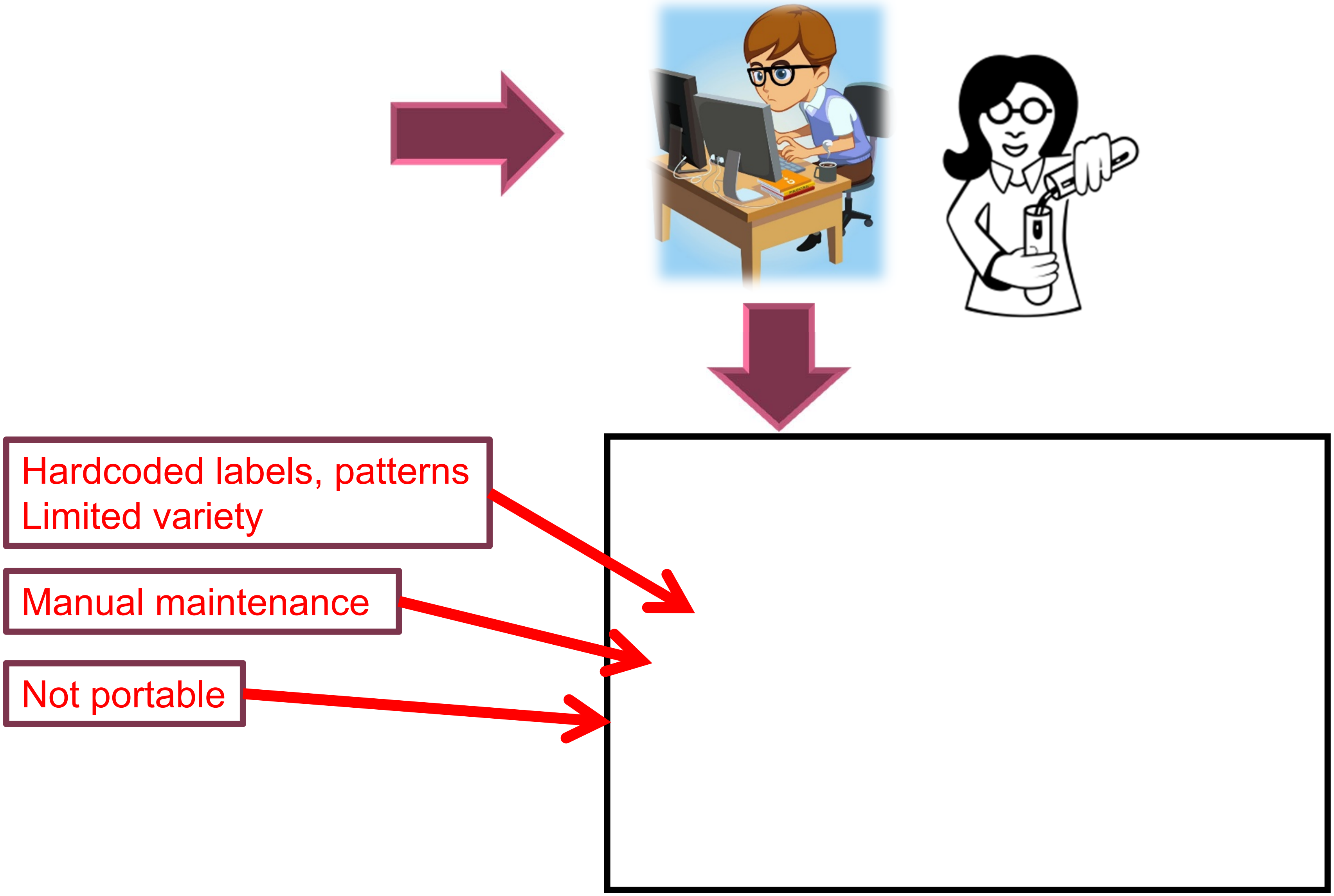
Secure | https://pubchem.ncbi.nlm.nih.gov/upload/sketcher/index.html?smiles=&cnt=0

Broadband ▾ | SMILES ▾ | C1=CC=CC=C1

Export: MDL Molfile ▾

Hydrogen: Keep AsIs ▾

Import: Choose File No file chosen

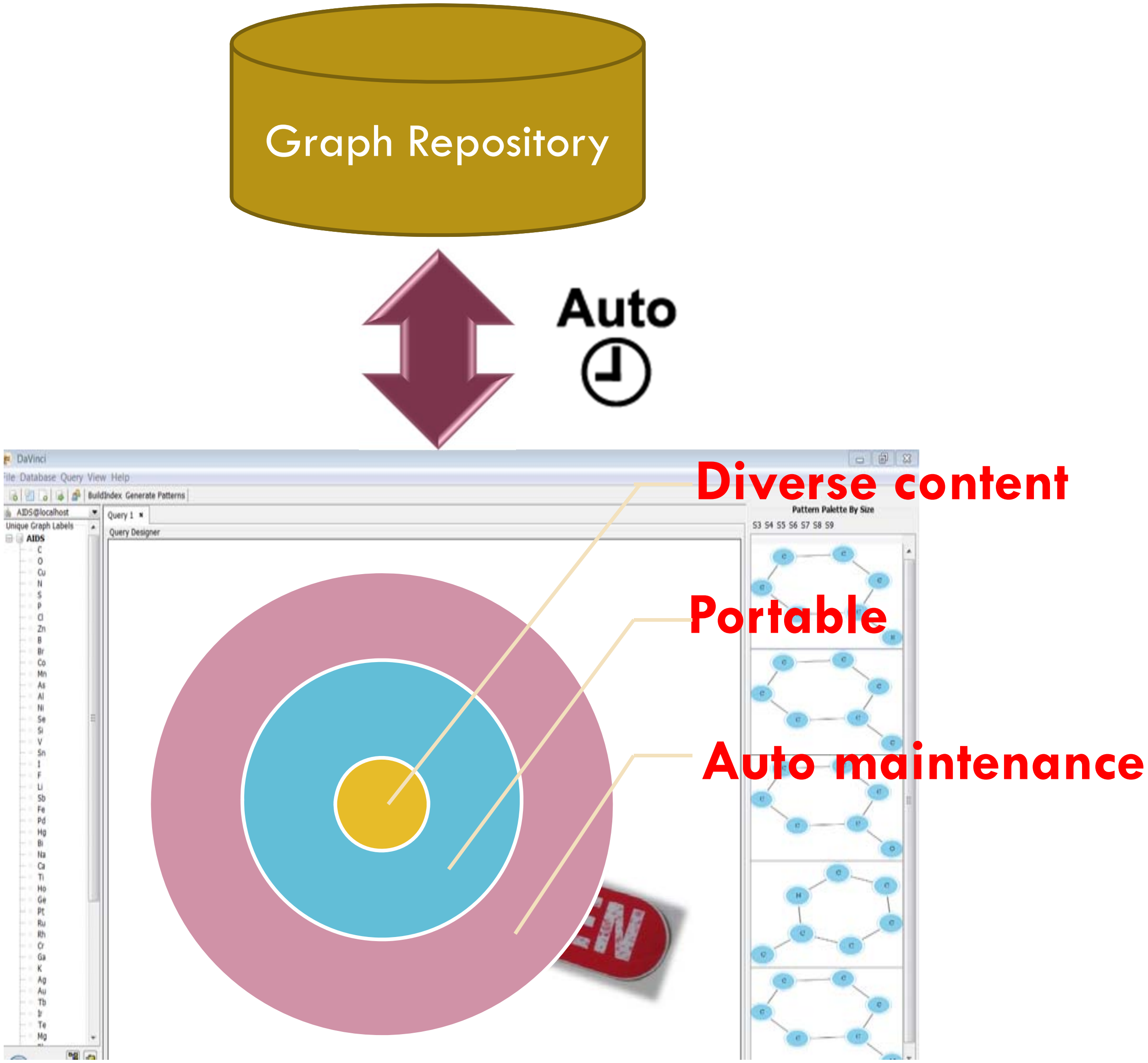


Hardcoded labels, patterns
Limited variety

Manual maintenance

Not portable

Data-driven Visual Interface Construction & Maintenance



Graph Repository

Auto

Diverse content

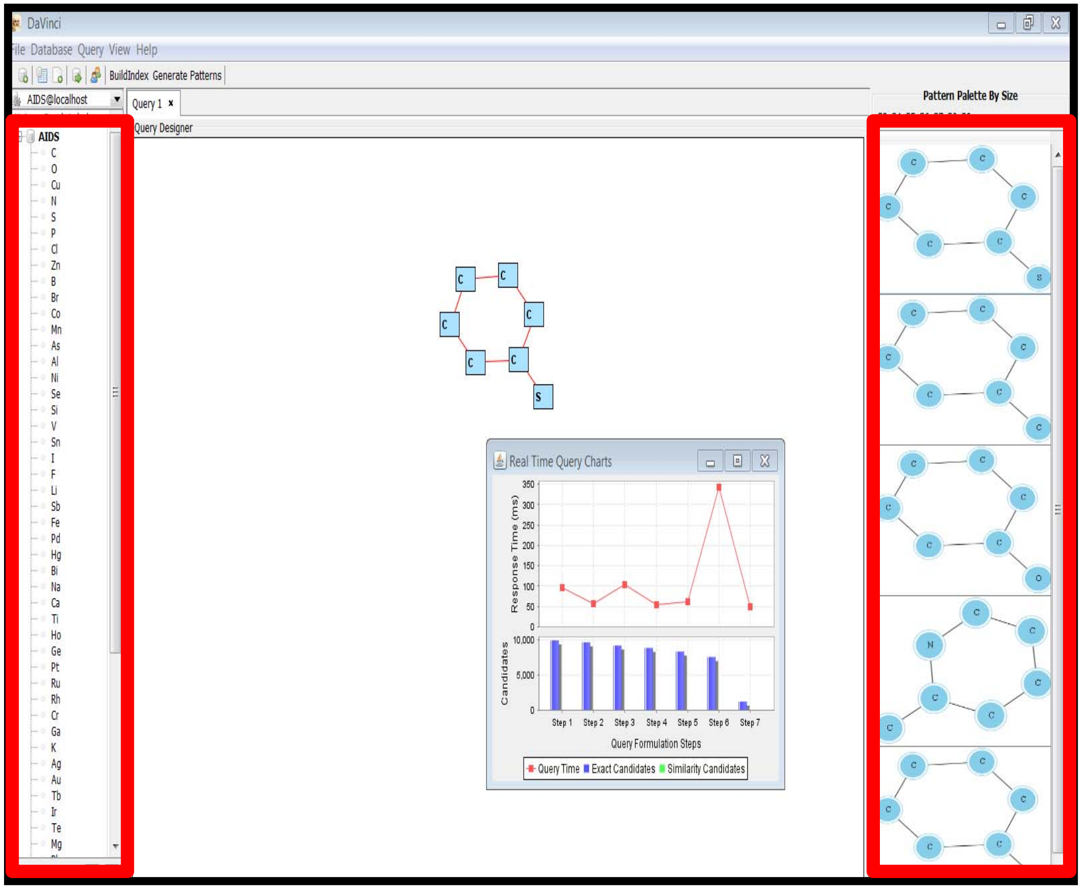
Portable

Auto maintenance

Data-driven Construction

Content selection

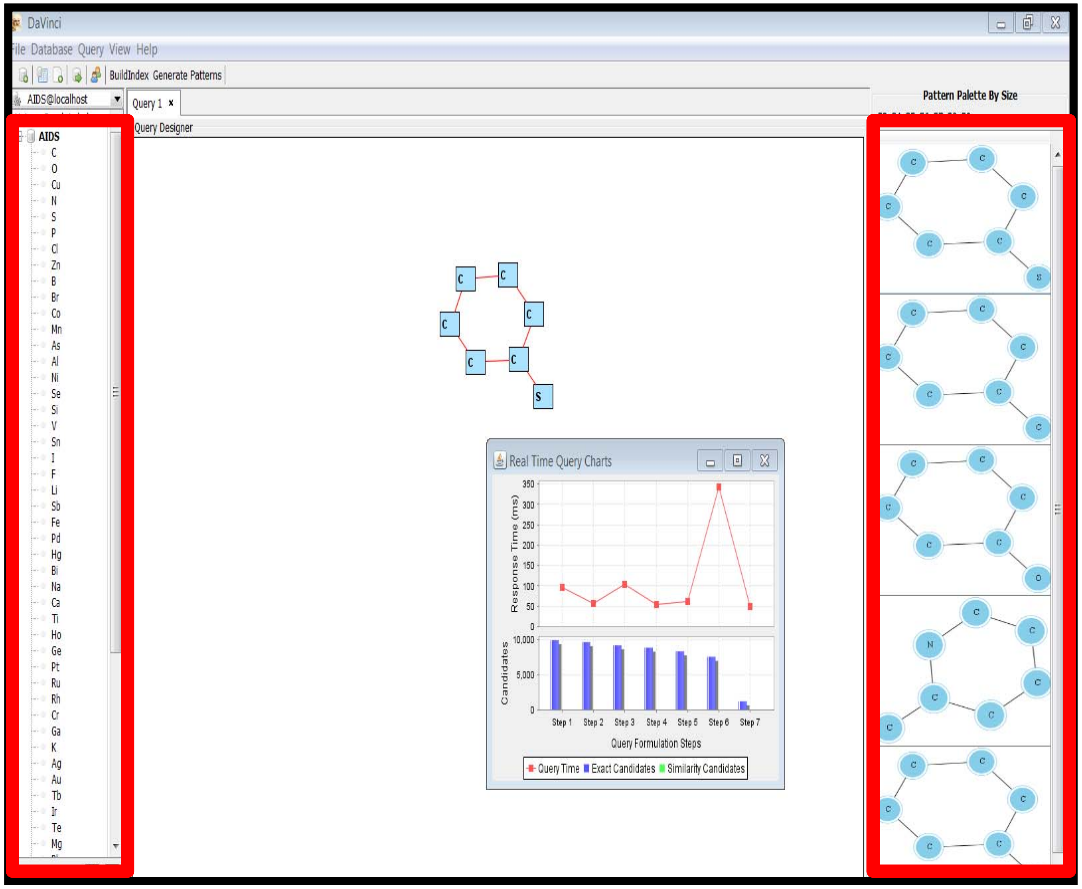
- Which patterns should be in the palette?
  - Formulate query easily and faster
  - Give shortcuts
- Issues
  - Size of the palette
  - Maximally covers the DB
  - Minimal redundancy among patterns
  - Aesthetics-aware



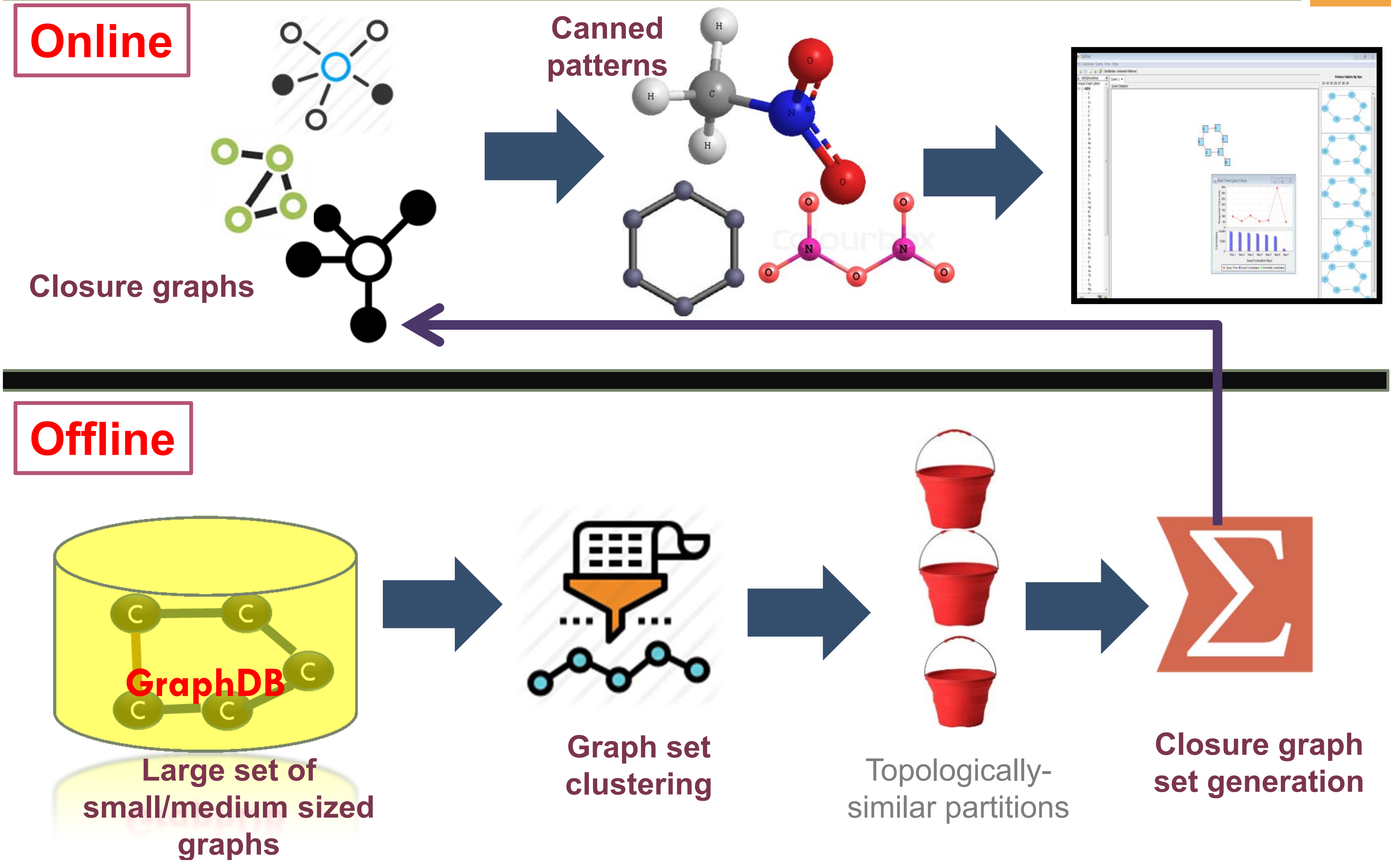
Data-driven Maintenance

Content Maintenance

- How do we maintain the labels and patterns as underlying data changes?
- Issues
  - Real-time maintenance
  - Batch vs Incremental
  - Enhance usability (gain in coverage and reduction in redundancy)
  - Leverage usage patterns and query workload (if available)



DAVINCI: Initial Effort [ICDE 15, VLDB 16]



Online

Canned patterns

Closure graphs

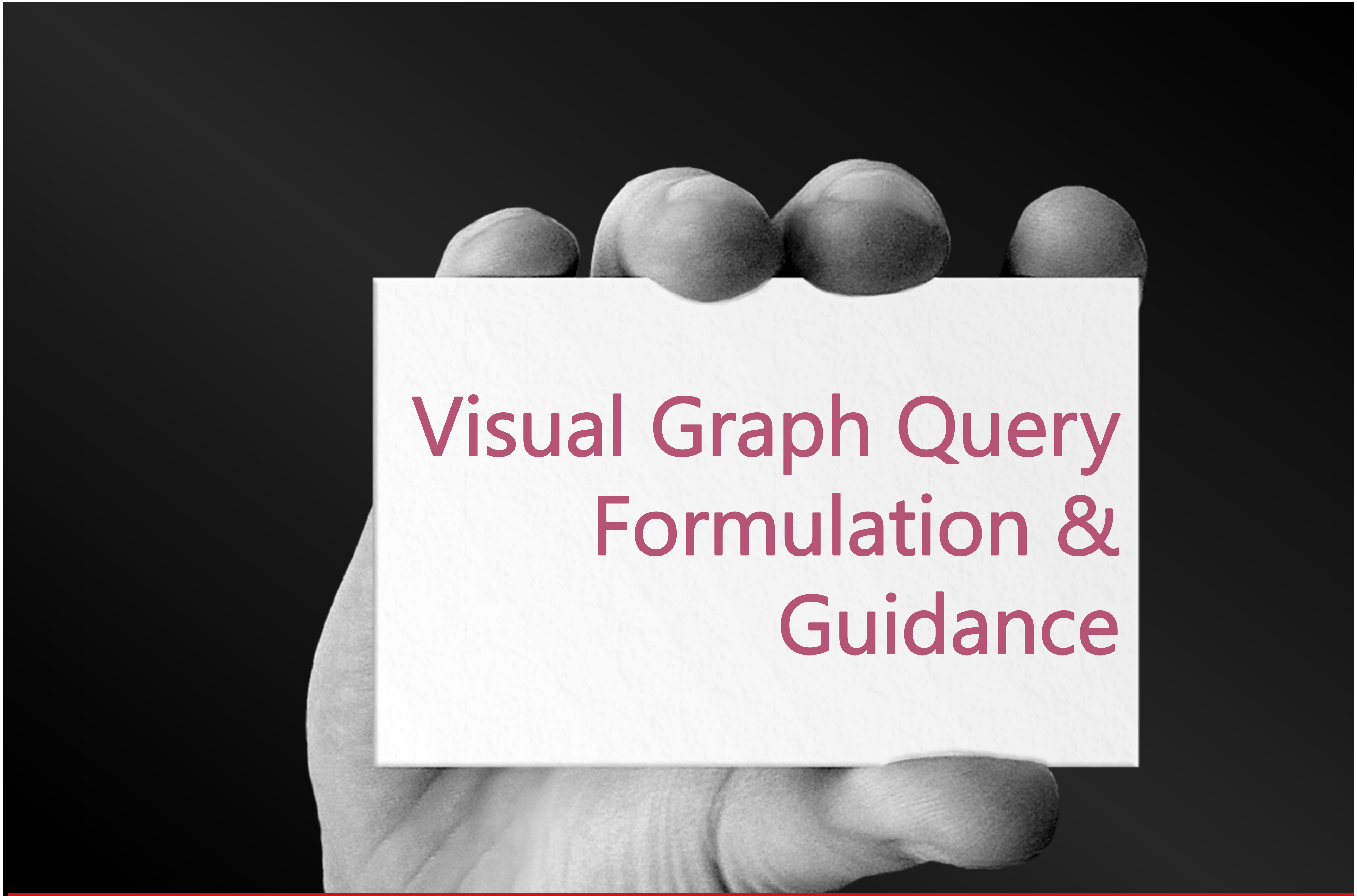
Offline

Large set of small/medium sized graphs

Graph set clustering

Topologically-similar partitions

Closure graph set generation



Visual Graph Query Formulation & Guidance

Next

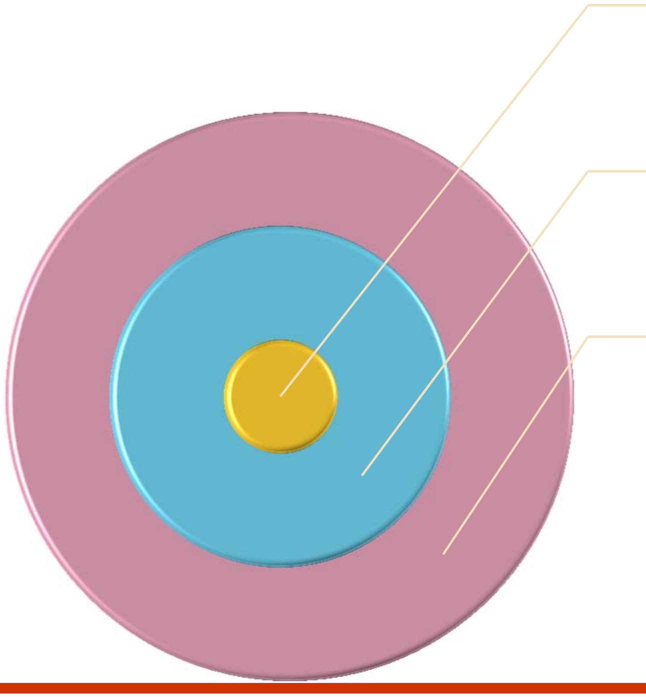
Opportune Query Feedback

Modeling feedback

❑ An alert or notification for a secondary task when a user is working on a primary task

Needs

- Detect efficiently
- Notify opportunely
  - Ineffective to notify at the end of query formulation



Efficient detection algorithm

Immediate notification

Cognitive impact on users doesn't matter!

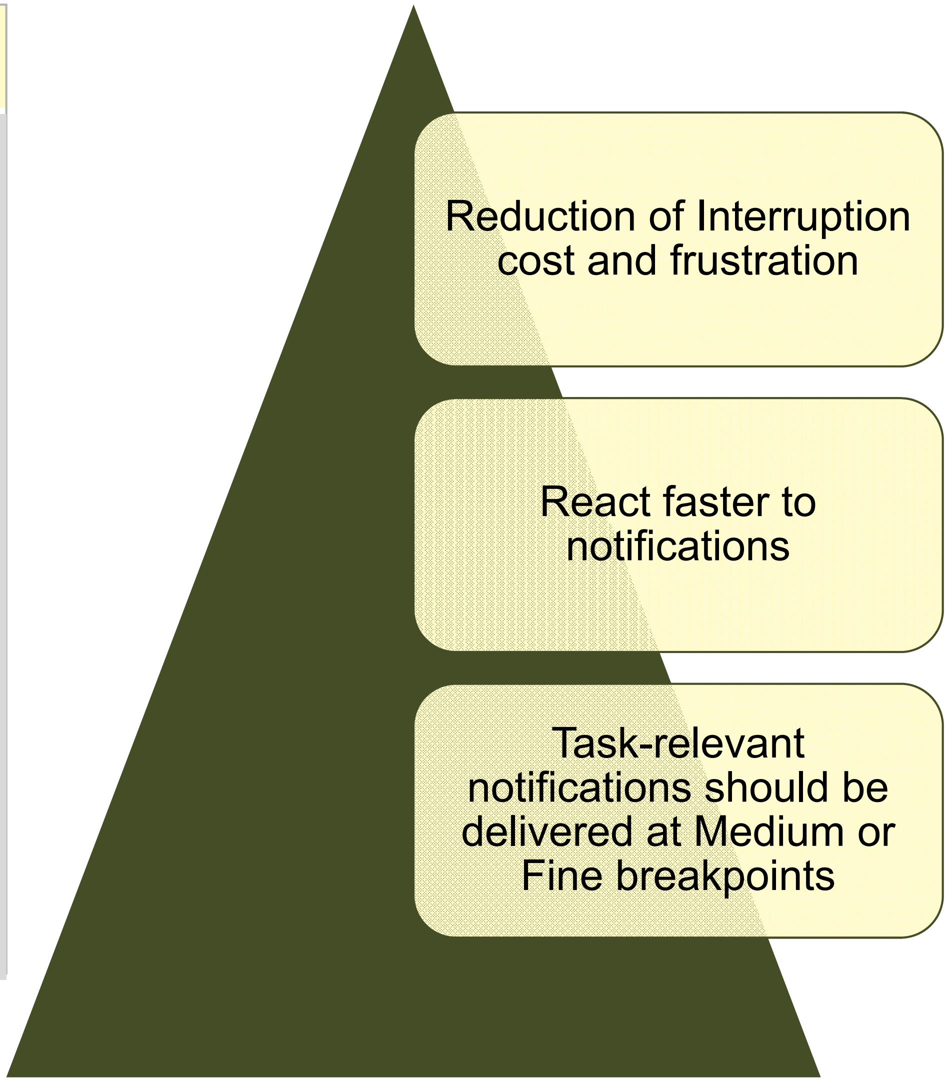
Delivering notifications inopportunely can negatively impact task completion time, lead to more errors, and increase user frustration.

When to notify?

Breakpoint

- The moment of transition between two observable, meaningful units of task execution, and reflects a change in perception or action **[Newton, 1973]**
- Coarse, Medium, and Fine
- Best moment to interrupt a user is on breakpoints between tasks **[Iqbal & Bailey, CHI 2008]**
  - Defer the notification to appear in the next breakpoint

Adopt defer-to-breakpoint-based strategy for interrupting query formulation tasks

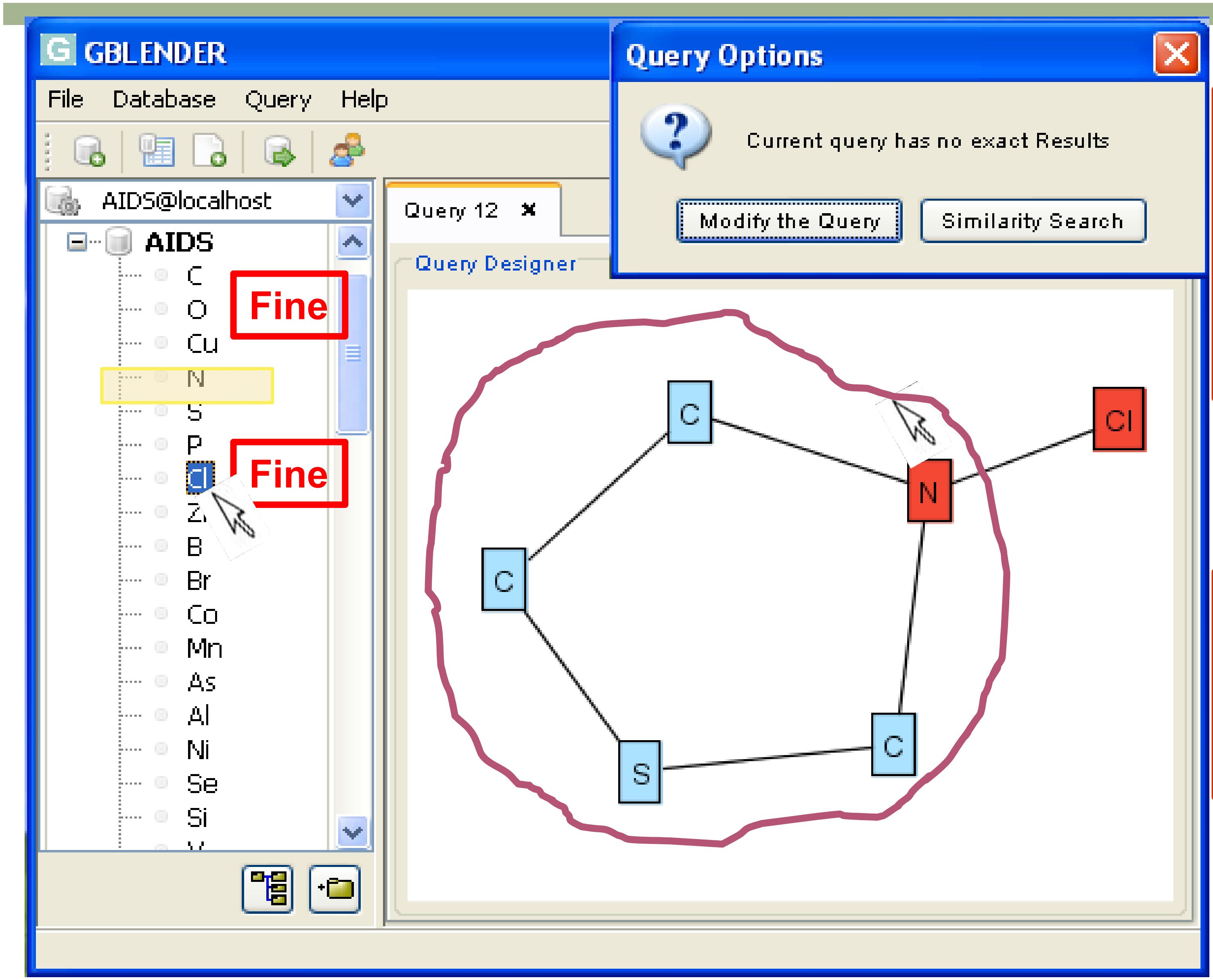


Reduction of Interruption cost and frustration

React faster to notifications

Task-relevant notifications should be delivered at Medium or Fine breakpoints

Modeling Optimal Notification Time

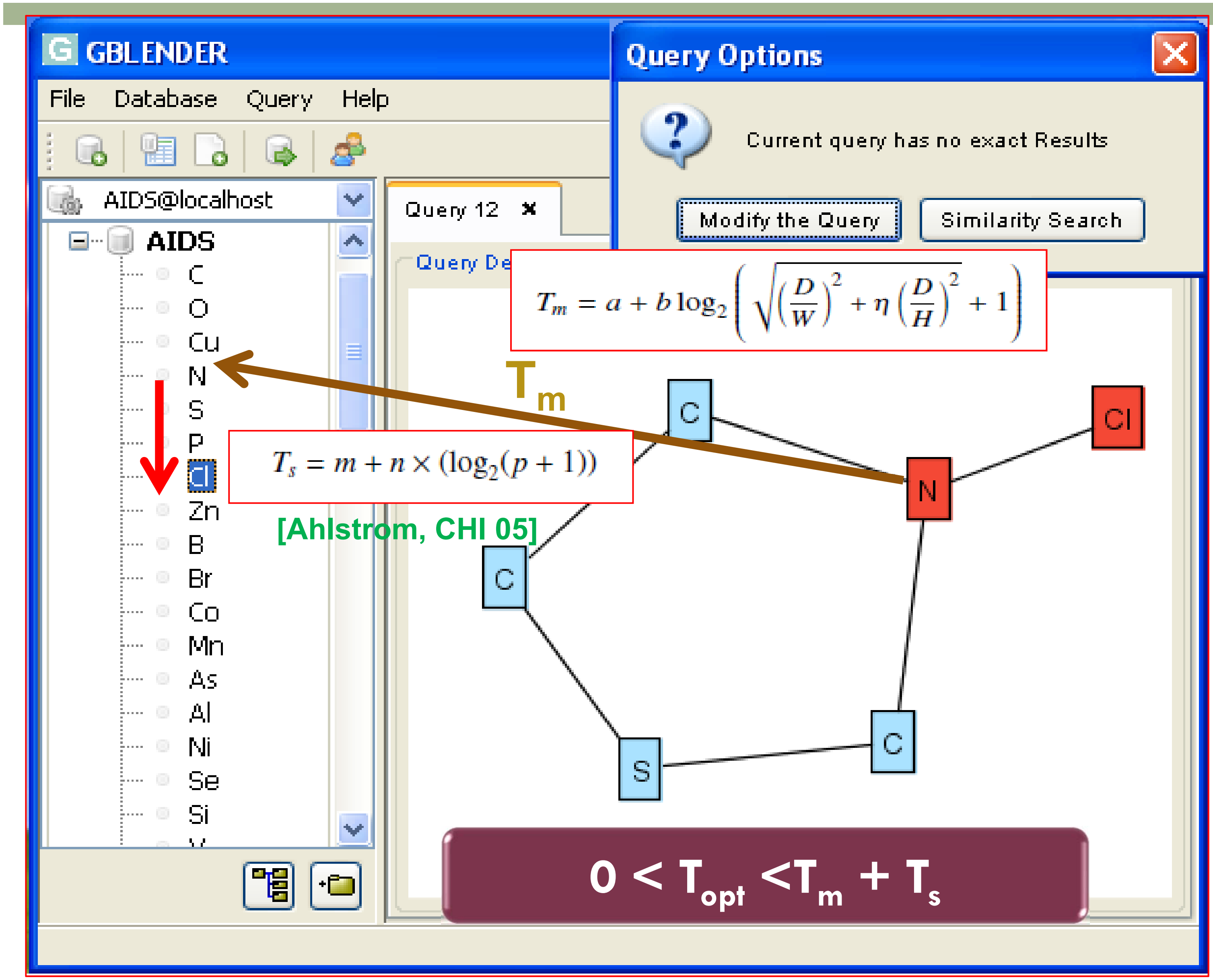


Current query has no exact Results

Deliver notification before the construction of the succeeding query condition is finished (Optimal breakpoints)

How do we estimate the time available for deliver of notification at optimal breakpoint?

HCI-Inspired Quantitative Model



Current query has no exact Results

$$T_m = a + b\log_2\left(\sqrt{\left(\frac{D}{W}\right)^2 + \eta\left(\frac{D}{H}\right)^2} + 1\right)$$

[Accott & Zhai, CHI 03]

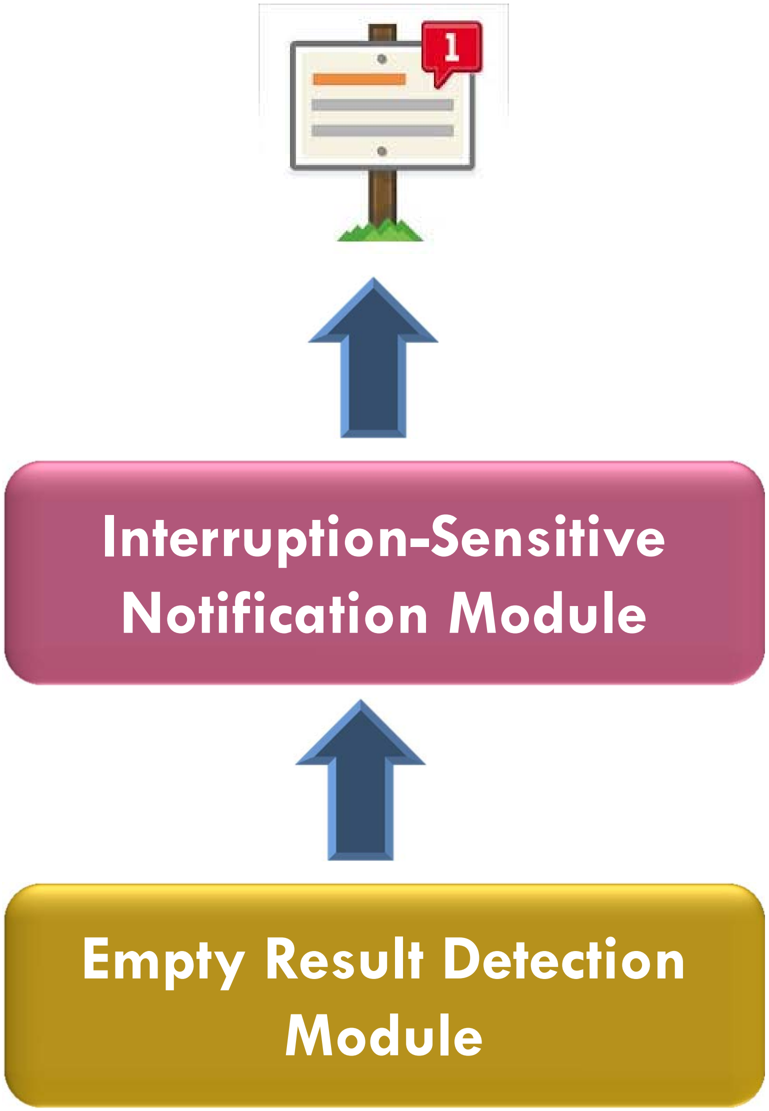
$$T_s = m + n \times (\log_2(p+1))$$

[Ahlstrom, CHI 05]

$$0 < T_{opt} < T_m + T_s$$

The iSERF Framework

[CIKM 15]


Interruption-Sensitive Notification Module
Empty Result Detection Module

Cursor moving towards Schema Panel

- ❑Compute movement time $\mathbf{T}_m$
- ❑Suspend notification by $\mathbf{T}_m$ time

Cursor in Schema Panel

- ❑Compute selection time $\mathbf{T}_s$
- ❑Suspend notification by $\mathbf{T}_s$ time for item to be selected

Notification delivery

- ❑Deliver appropriate notification identifying condition(s) for empty result

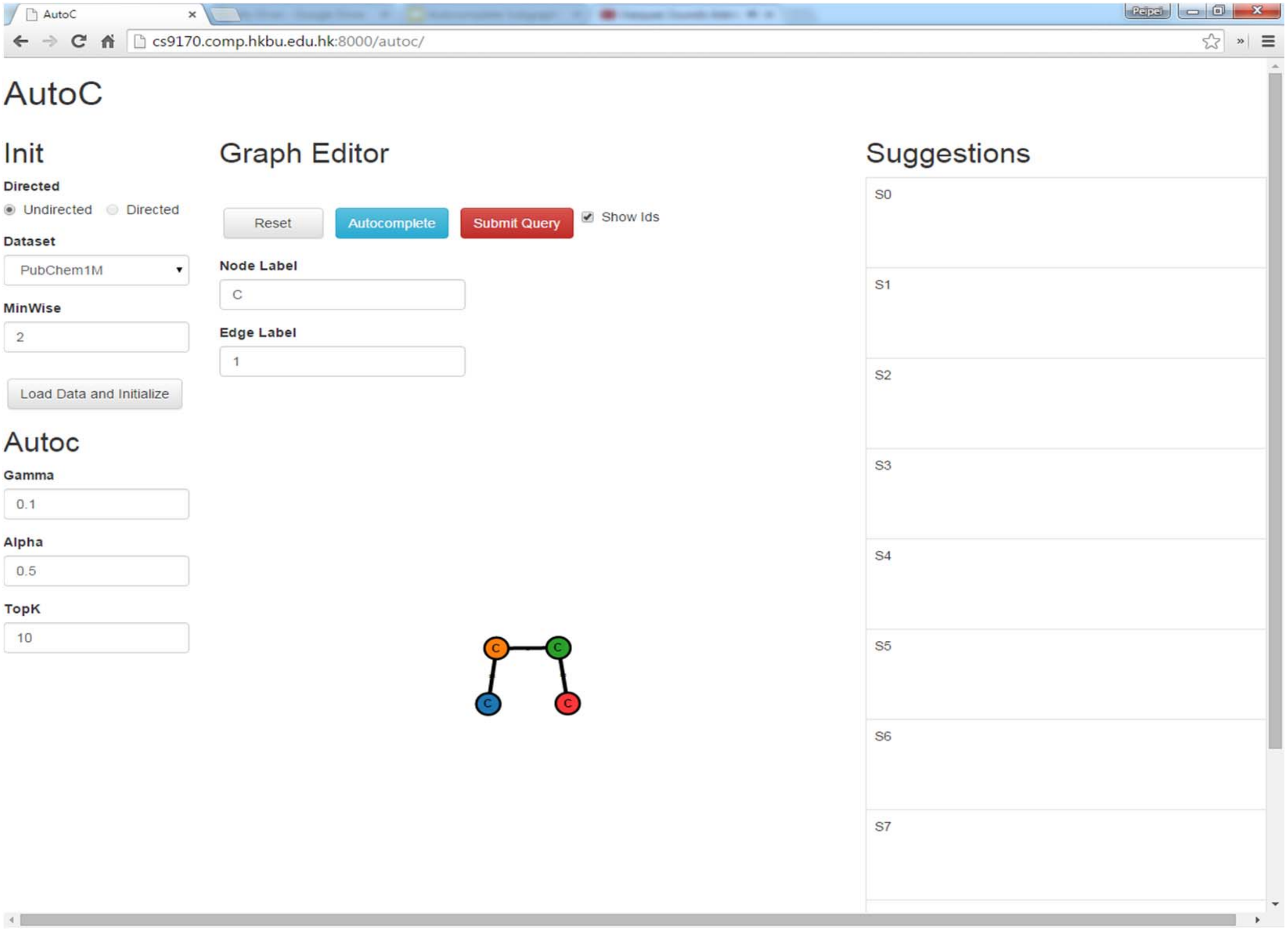
More on the Feedback Module

Query Autocompletion

Action Guidance

Query Autocompletion Demo

- ❑ http://www.comp.hkbu.edu.hk/~csppyi/autog/



AutoC

Init

Directed

● Undirected ○ Directed

Dataset

PubChem1M ▾

MinWise

2

Autoc

Gamma

0.1

Alpha

0.5

TopK

10

Graph Editor

☑ Show Ids

Node Label

C

Edge Label

1

Suggestions

S0

S1

S2

S3

S4

S5

S6

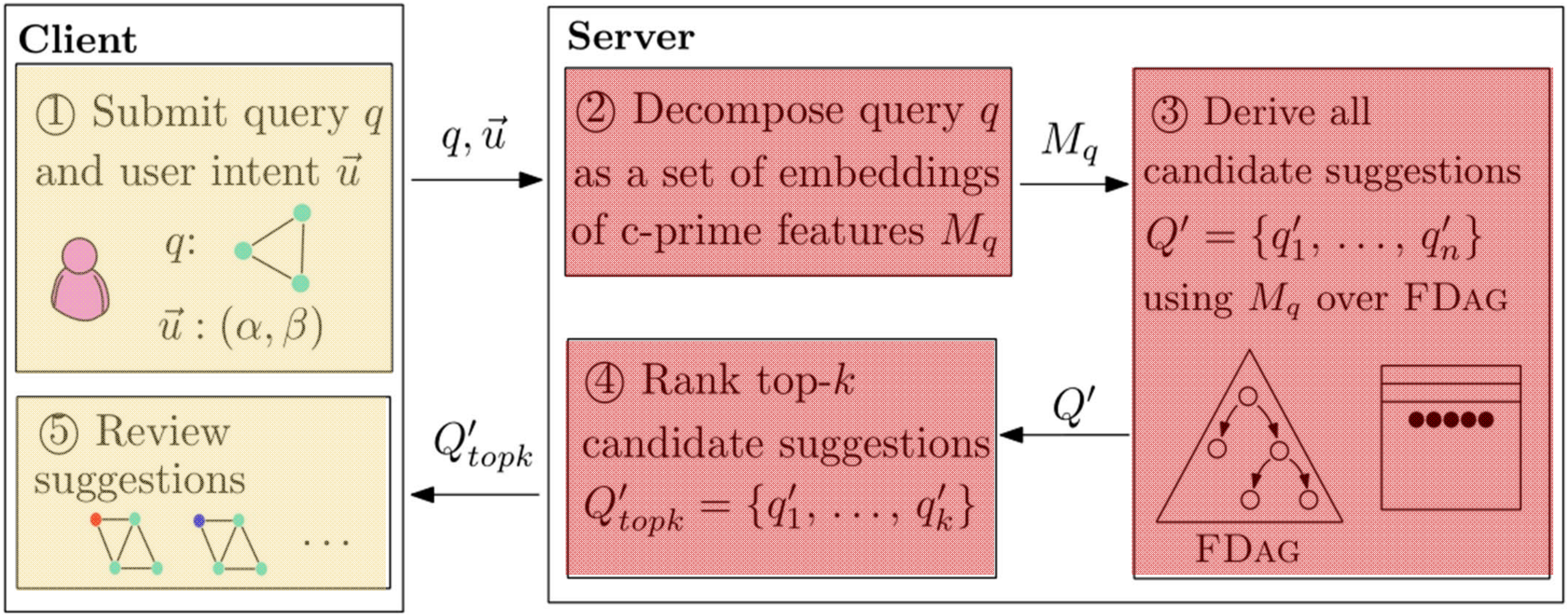
S7

Autocompletion Comparisons

	Keyword Search	Visual Graph Query
User Action	keystroke	Drag, click
Atomic Unit	char: 'a', 'b', 'c', ...	edge: C-C, C=C, ...
Logical Unit	keyword: "world", "clock", ...	subgraphs
Query	concatenated keywords	graphs
	"world clock", "world cup", ...	C=C-C=C-C=C, ...

The AutoG Framework

[VLDB 16, VLDBJ 17]



Client

① Submit query q and user intent $\vec{u}$

q:

$\vec{u} : (\alpha, \beta)$

⑤ Review suggestions

$\ldots$

$q, \vec{u}$

Q'_{topk}

Server

② Decompose query q as a set of embeddings of c-prime features M_q

M_q

③ Derive all candidate suggestions $Q' = \{q'_1, \ldots, q'_n\}$ using M_q over FDag

FDag

Q'

④ Rank top-k candidate suggestions $Q'_{topk} = \{q'_1, \ldots, q'_k\}$

User Preference / Intent

User intent value of a query (suggestion) set

$$\mathsf{util}(Q') = \alpha \times \frac{1}{k} \sum_{q' \in Q'} \mathsf{sel}(q') + \beta \times \frac{1}{k(k-1)} \sum_{q'_i, q'_j \in Q', i \neq j} \mathsf{dist}(q'_i, q'_j)$$

(MCCS) Distance between two graph suggestions

$$\mathsf{dist}(g_1, g_2) = 1 - \frac{|\mathsf{cs}(g_1, g_2)|}{\mathsf{max}\{|g_1|, |g_2|\}}$$

Property: `util` is submodular → greedy

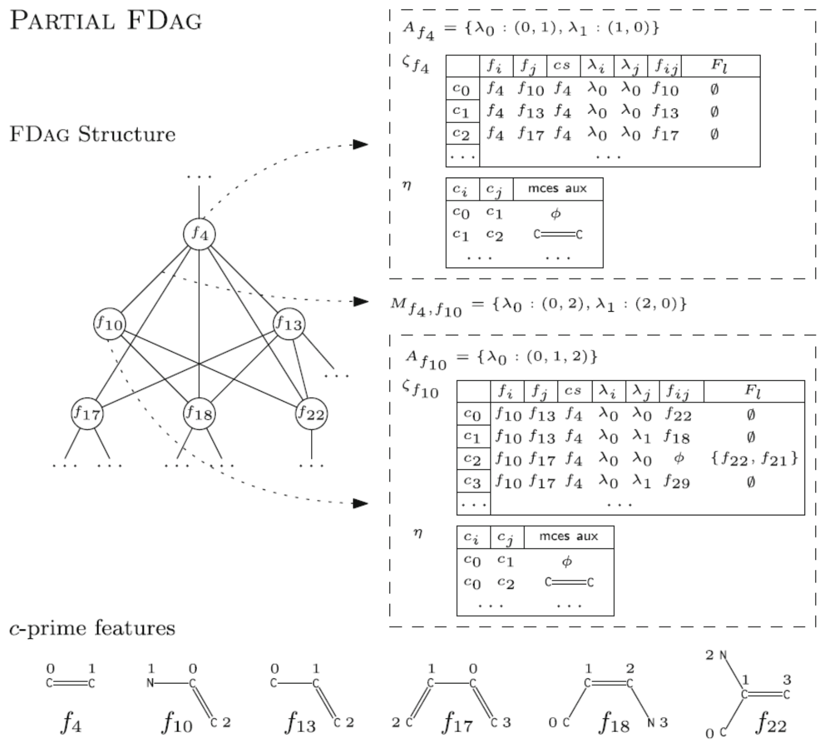
Optimizations

The FDAG index

- ❑Index c-Prime features and their pairwise compositions
- ❑Prune automorphic suggestions (redundant suggestions) early

Online ranking

- ❑Approximate selectivities of query suggestions
- ❑Prune empty suggestions early
- ❑Optimize diversity computation
  - ❑trimming the common parts between suggestions



PARTIAL FDAG

FDAG Structure

$A_{f_4} = \{\lambda_0 : (0,1), \lambda_1 : (1,0)\}$

ζ_{f_4}

	f_i	f_j	cs	λ_i	λ_j	f_{ij}	F_l
c_0	f_4	f_{10}	f_4	λ_0	λ_0	f_{10}	$\emptyset$
c_1	f_4	f_{13}	f_4	λ_0	λ_0	f_{13}	$\emptyset$
c_2	f_4	f_{17}	f_4	λ_0	λ_0	f_{17}	$\emptyset$
. . .				. . .			

η

c_i	c_j	mces aux
c_0	c_1	ϕ
c_1	c_2	C=C
. . .		. . .

$M_{f_4, f_{10}} = \{\lambda_0 : (0,2), \lambda_1 : (2,0)\}$

$A_{f_{10}} = \{\lambda_0 : (0,1,2)\}$

$\zeta_{f_{10}}$

	f_i	f_j	cs	λ_i	λ_j	f_{ij}	F_l
c_0	f_{10}	f_{13}	f_4	λ_0	λ_0	f_{22}	$\emptyset$
c_1	f_{10}	f_{13}	f_4	λ_0	λ_1	f_{18}	$\emptyset$
c_2	f_{10}	f_{17}	f_4	λ_0	λ_0	ϕ	$\{f_{22}, f_{21}\}$
c_3	f_{10}	f_{17}	f_4	λ_0	λ_1	f_{29}	$\emptyset$
. . .				. . .			

η

c_i	c_j	mces aux
c_0	c_1	ϕ
c_0	c_2	C=C
. . .		. . .

c-prime features

f_4 f_{10} f_{13} f_{17} f_{18} f_{22}

More on the Feedback Module

Query Autocompletion

Action Guidiance

Orion

Overview

❑Interactive visual query builder with suggestions

❑Iteratively suggest edges based on their relevance to the user’s query intent, according to the partial query graph so far

  ❑Edge ranking: query-specific random decision paths

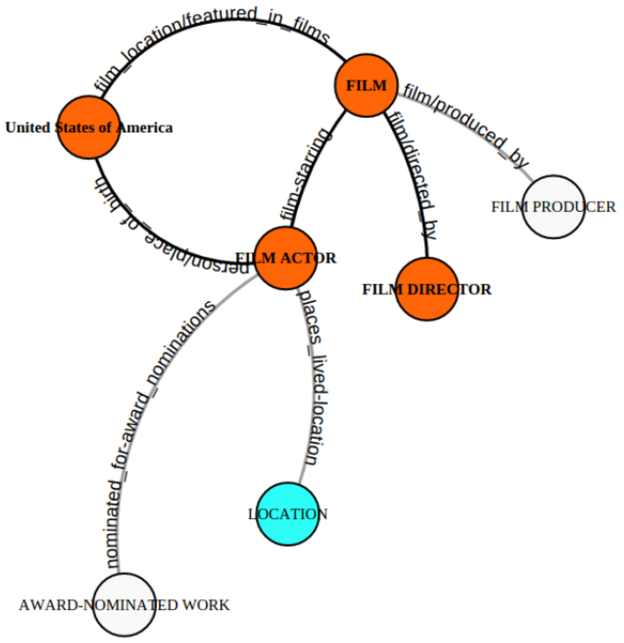
❑The use of statistics based on data graph, query logs, and so on.

Suggestions: Grey nodes/edges

❑Accepted by users: Positive edges (become blue)

❑Reject or ignored by users: Negative edges

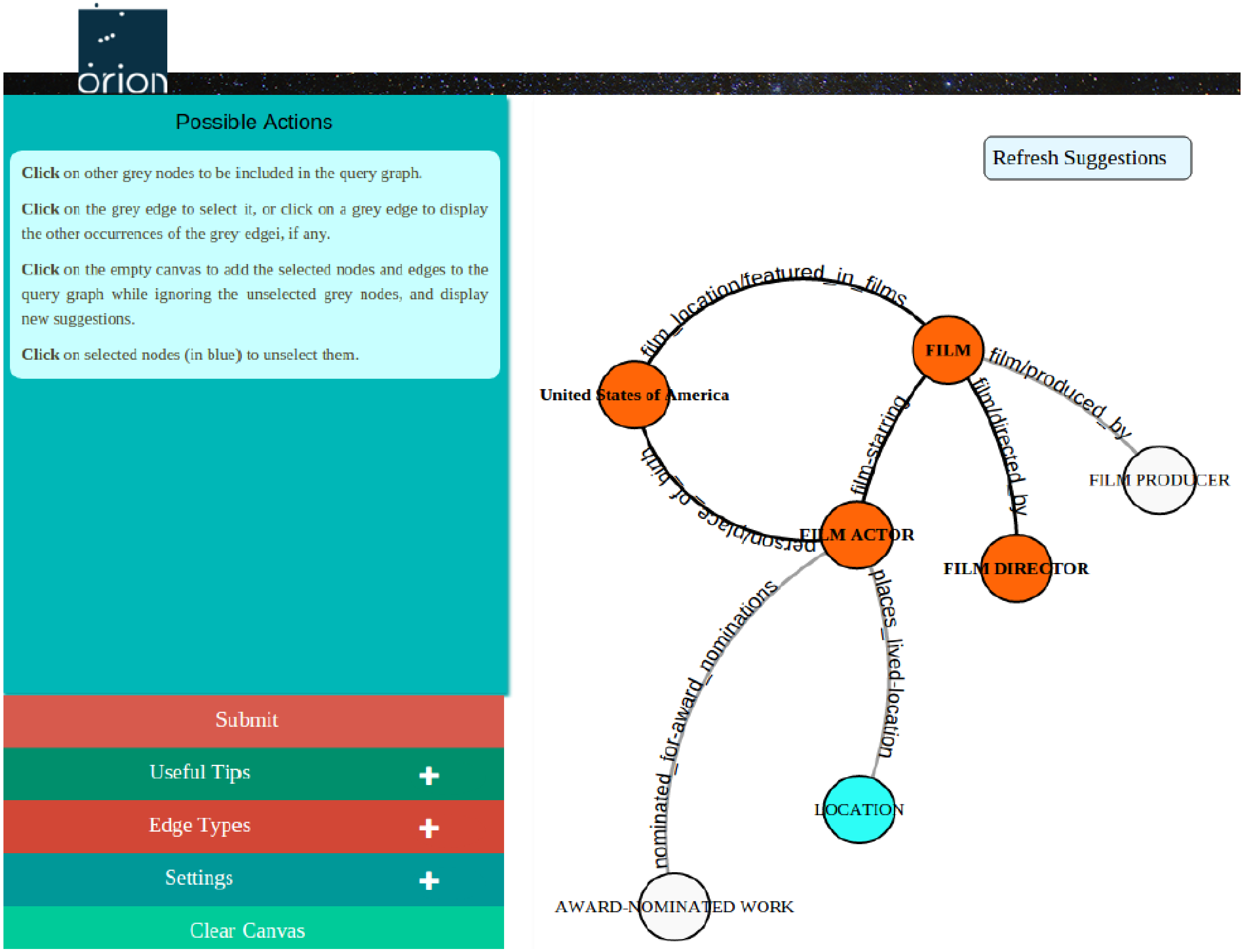
User’s intent can be derived from these edges



Orion GUI

Dynamic list of all possible user actions at any given moment

Control panel for various settings and tips



Possible Actions

Click on other grey nodes to be included in the query graph.

Click on the grey edge to select it, or click on a grey edge to display the other occurrences of the grey edgei, if any.

Click on the empty canvas to add the selected nodes and edges to the query graph while ignoring the unselected grey nodes, and display new suggestions.

Click on selected nodes (in blue) to unselect them.

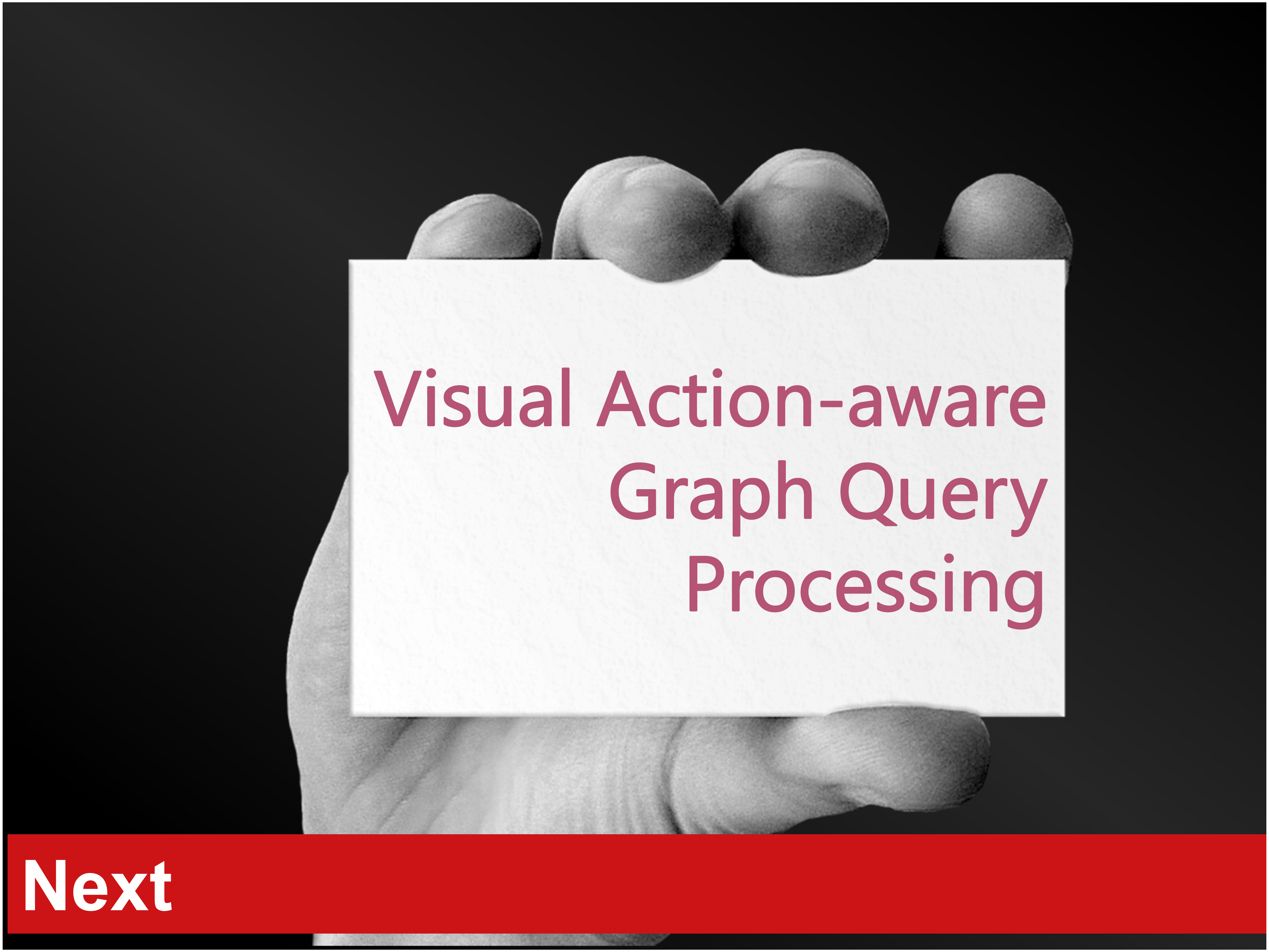
Orion Implementation

□ **Prototype**

http://idir.uta.edu/orion

□ **Video Introduction**

http://bit.ly/2pShvrm



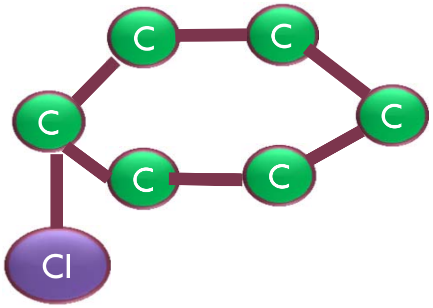
Visual Action-aware Graph Query Processing

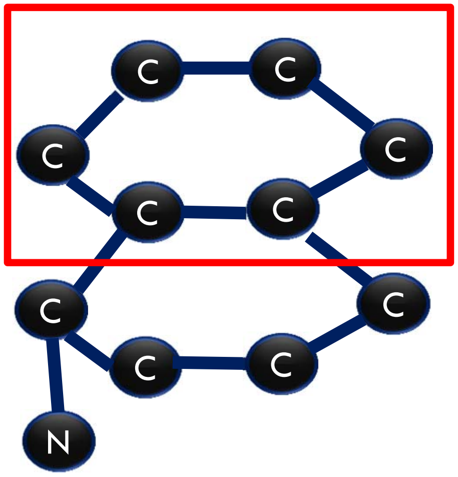
Next

Subgraph Queries

Subgraph Containment

- Given a graph DB D and a query graph Q, find all data graphs in D in which Q is a subgraph
- Subgraph isomorphism from Q to G ∈D



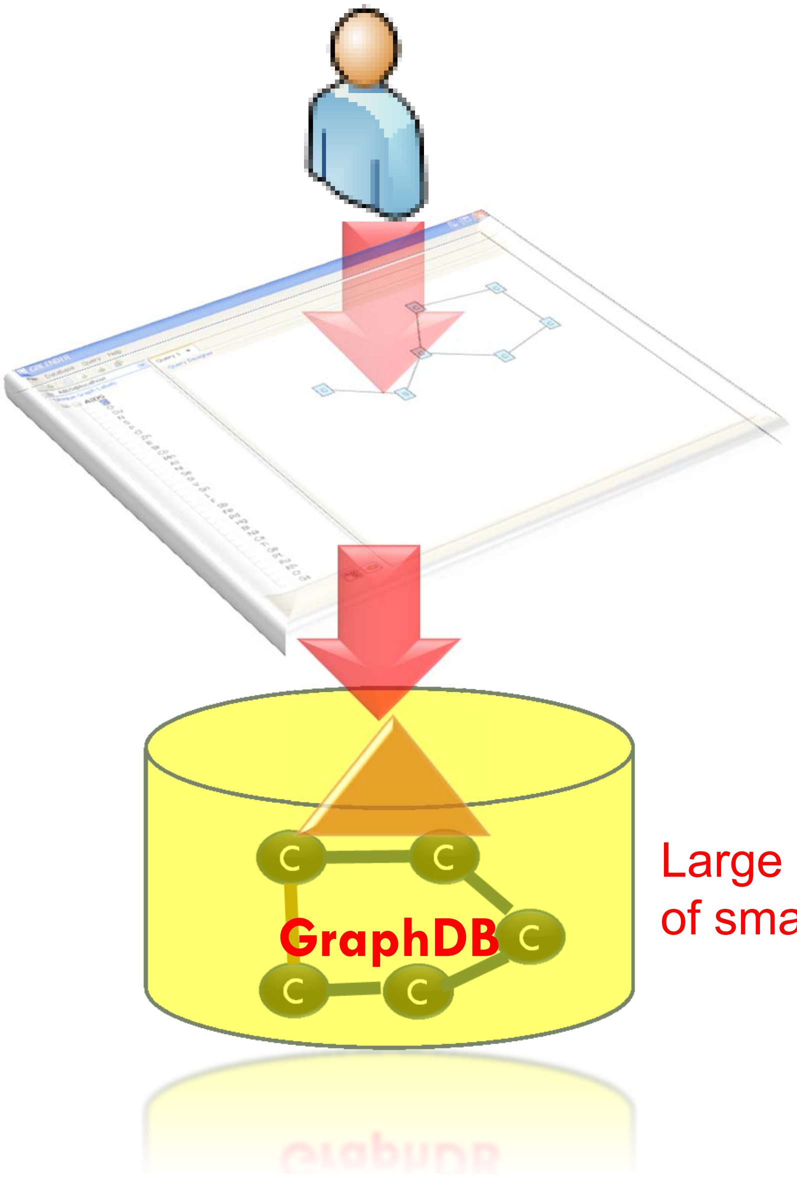


Subgraph Similarity

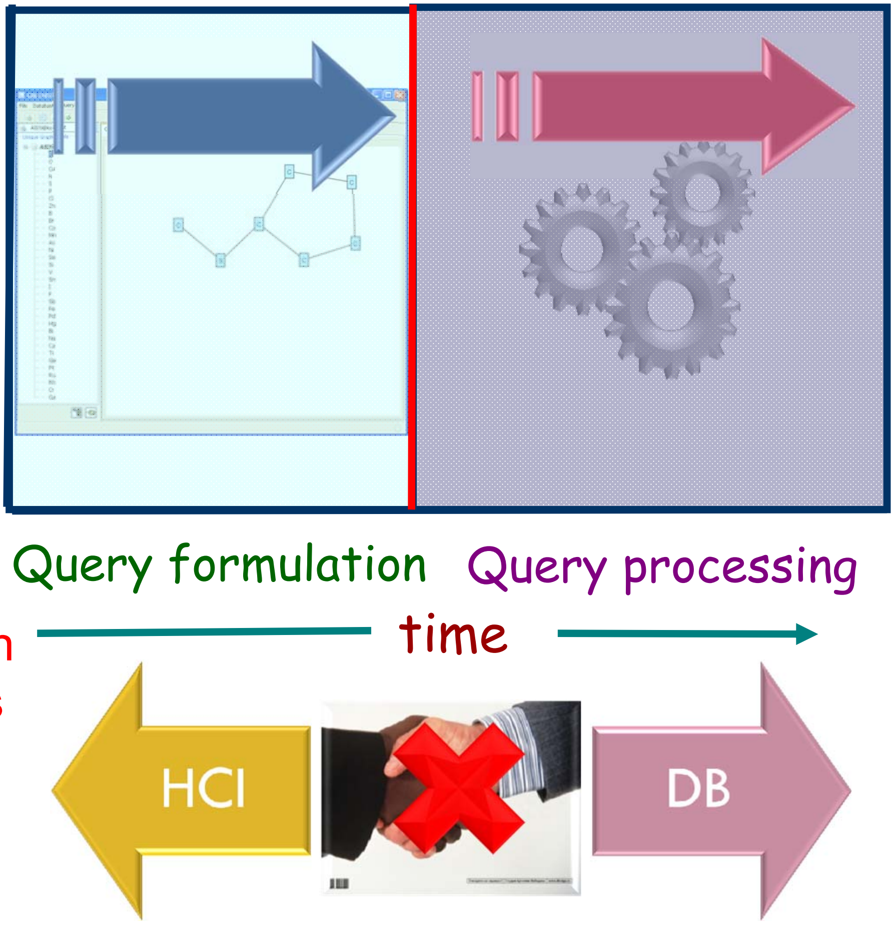
- Data graphs that "approximately" contain the query graph
- Use subgraph distance based on maximum connected common subgraph (MCCS)

Classical Visual Querying Paradigm

40+ years old query paradigm!



Large collection of small graphs

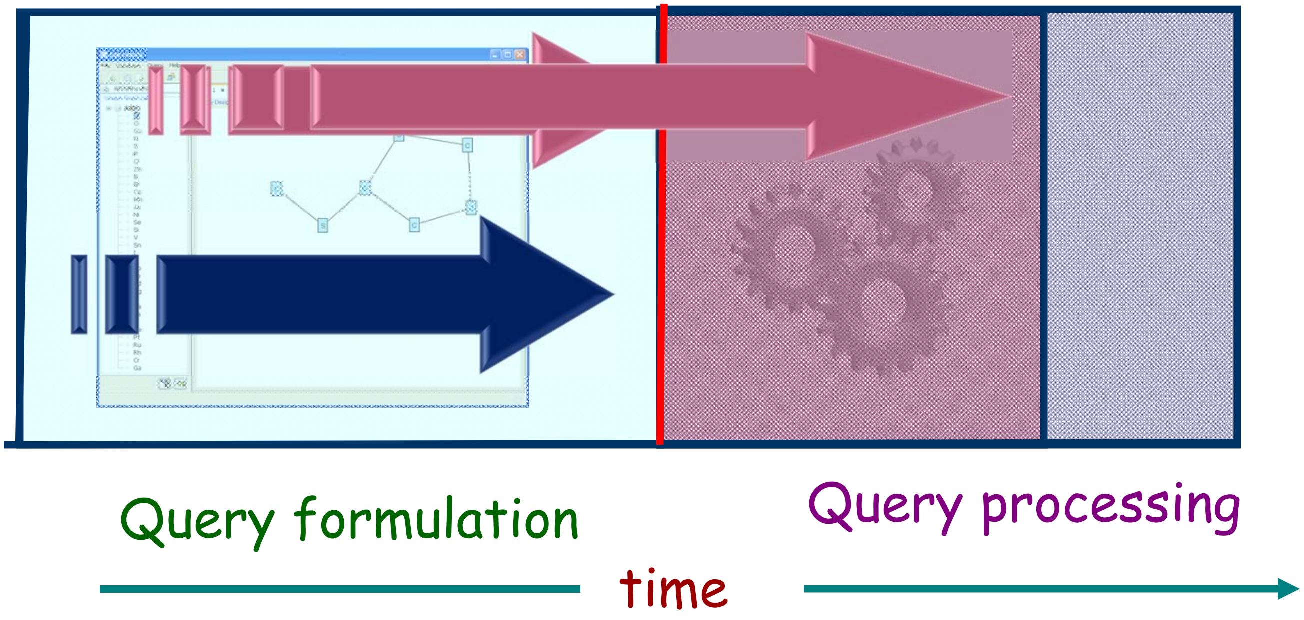


Visual Graph Query Formulation Meets Query Processing

Rethink the classical query paradigm

•Why wait for the complete visual query to be constructed before initiating query evaluation? How can we blend these two steps?

•By initiating query processing “early”, can we significantly reduce the system response time?

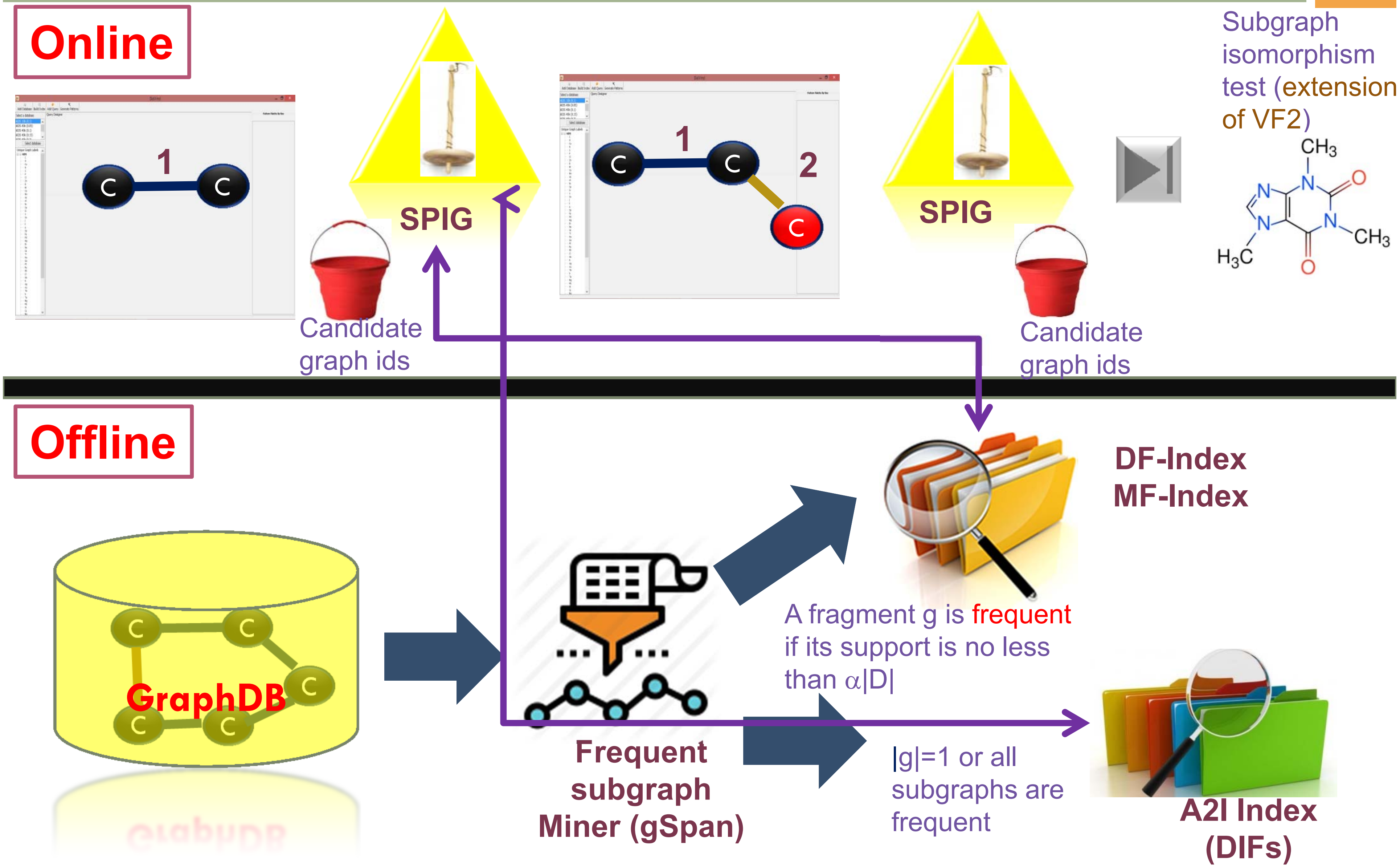


Non-traditional Challenges

- Partial query-aware indexing schemes
- Materialization of intermediate results
- Selectivity-free query processing
- Focus on waiting time of users

“Computing time (power) is getting cheaper but users’ time isn’t..”

Overview of VOGUE [SIGMOD 10, ICDE 12, CIDR 13]



Online

Offline

Candidate graph ids

Candidate graph ids

Subgraph isomorphism test (extension of VF2)

DF-Index
MF-Index

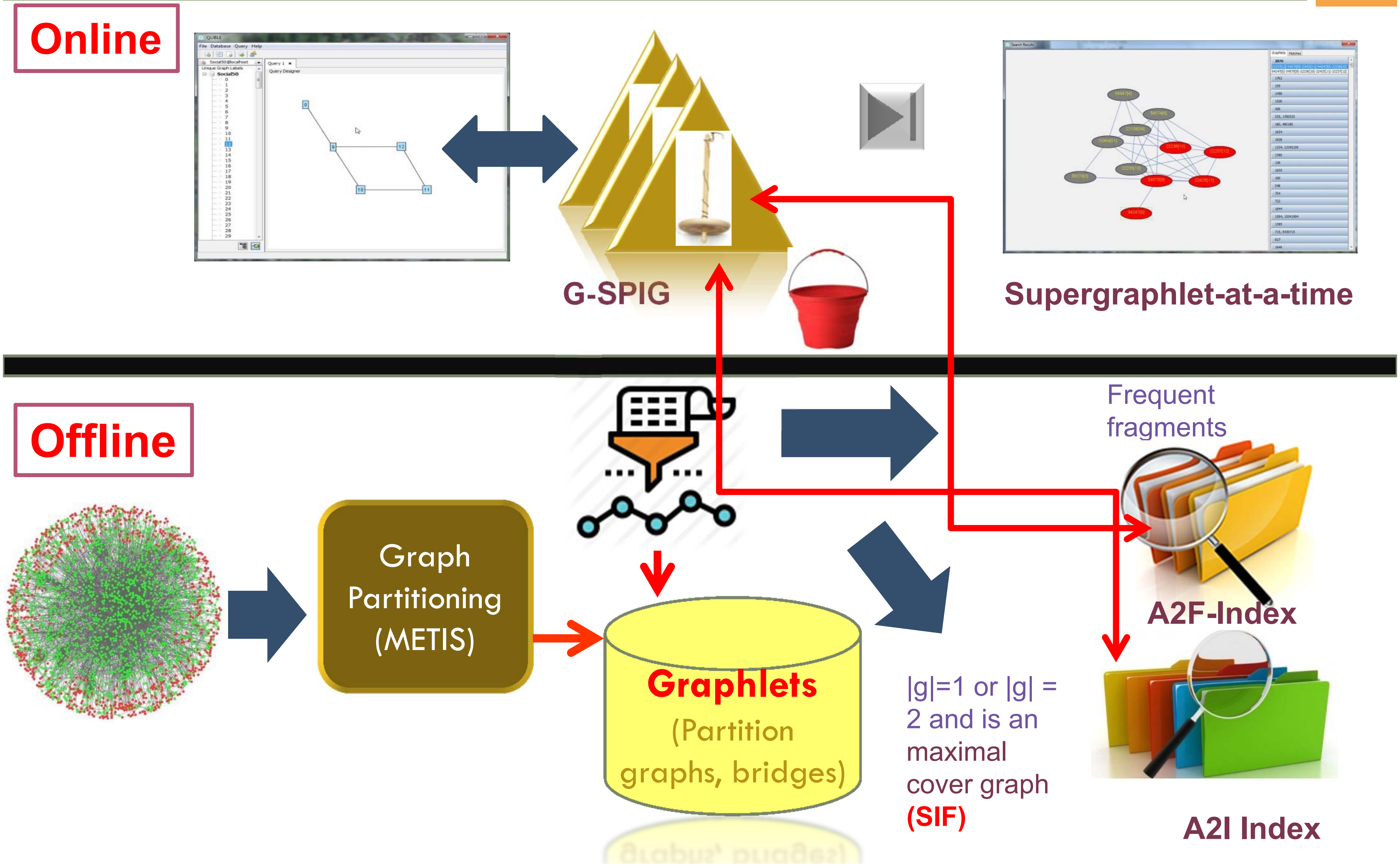
A fragment g is frequent if its support is no less than α|D|

|g|=1 or all subgraphs are frequent

Frequent subgraph Miner (gSpan)

A2I Index (DIFs)

QUBLE: Extension to Large Graphs [VLDB J 14]



Online

G-SPIG

Supergraphlet-at-a-time

Offline

Graph Partitioning (METIS)

Graphlets (Partition graphs, bridges)

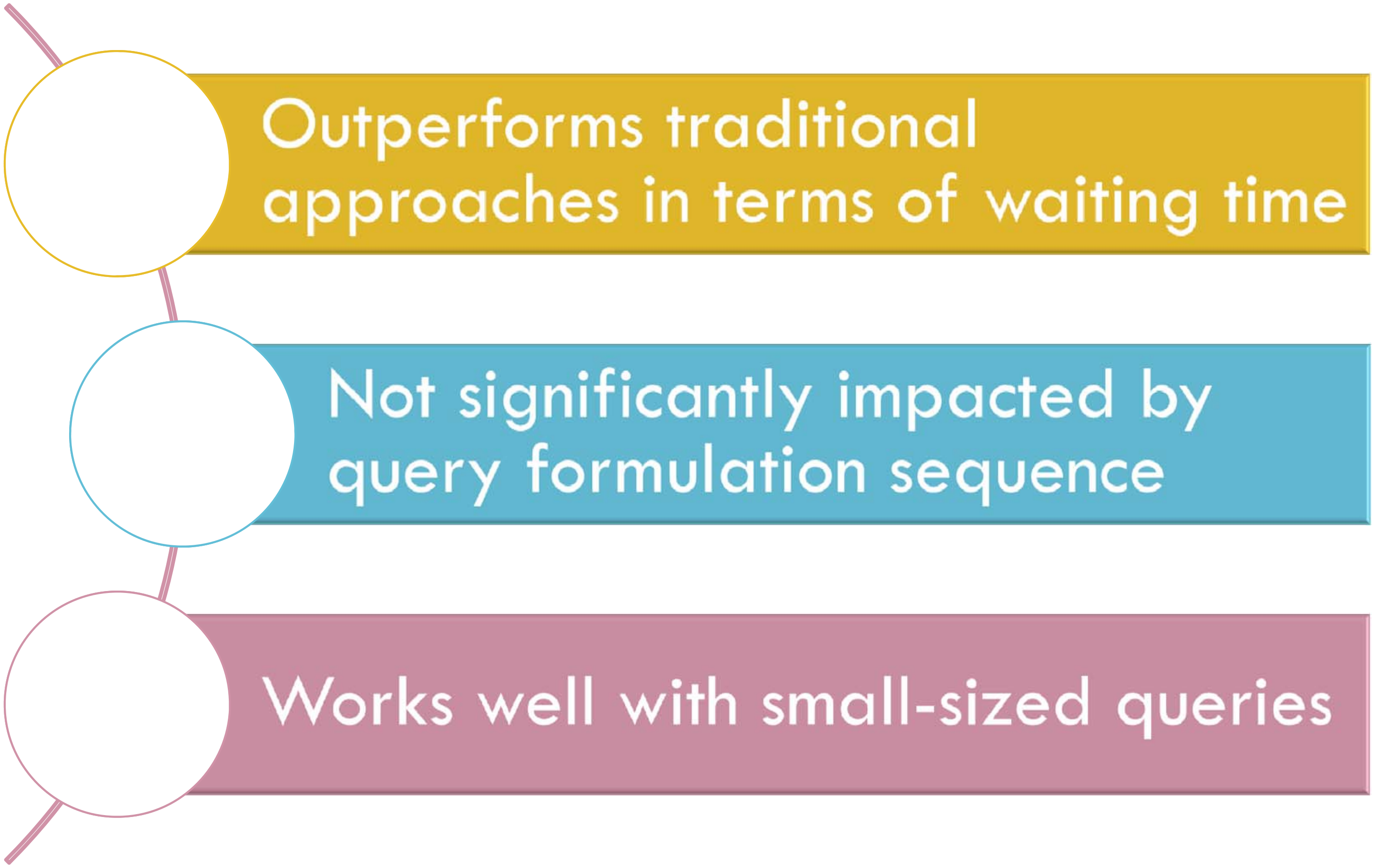
|g|=1 or |g| = 2 and is an maximal cover graph (SIF)

Frequent fragments

A2F-Index

A2I Index

Performance Summary



Outperforms traditional approaches in terms of waiting time

Not significantly impacted by query formulation sequence

Works well with small-sized queries

New!

LASER: Newer version can handle large query graphs and scales to more than million data graphs (10X more than state-of-the-art)!

Challenges for Performance Study

Large-scale performance study

- Traditional approach
  - Randomly extract subgraphs of different size and execute them
- Doesn't work in this paradigm!

Why?

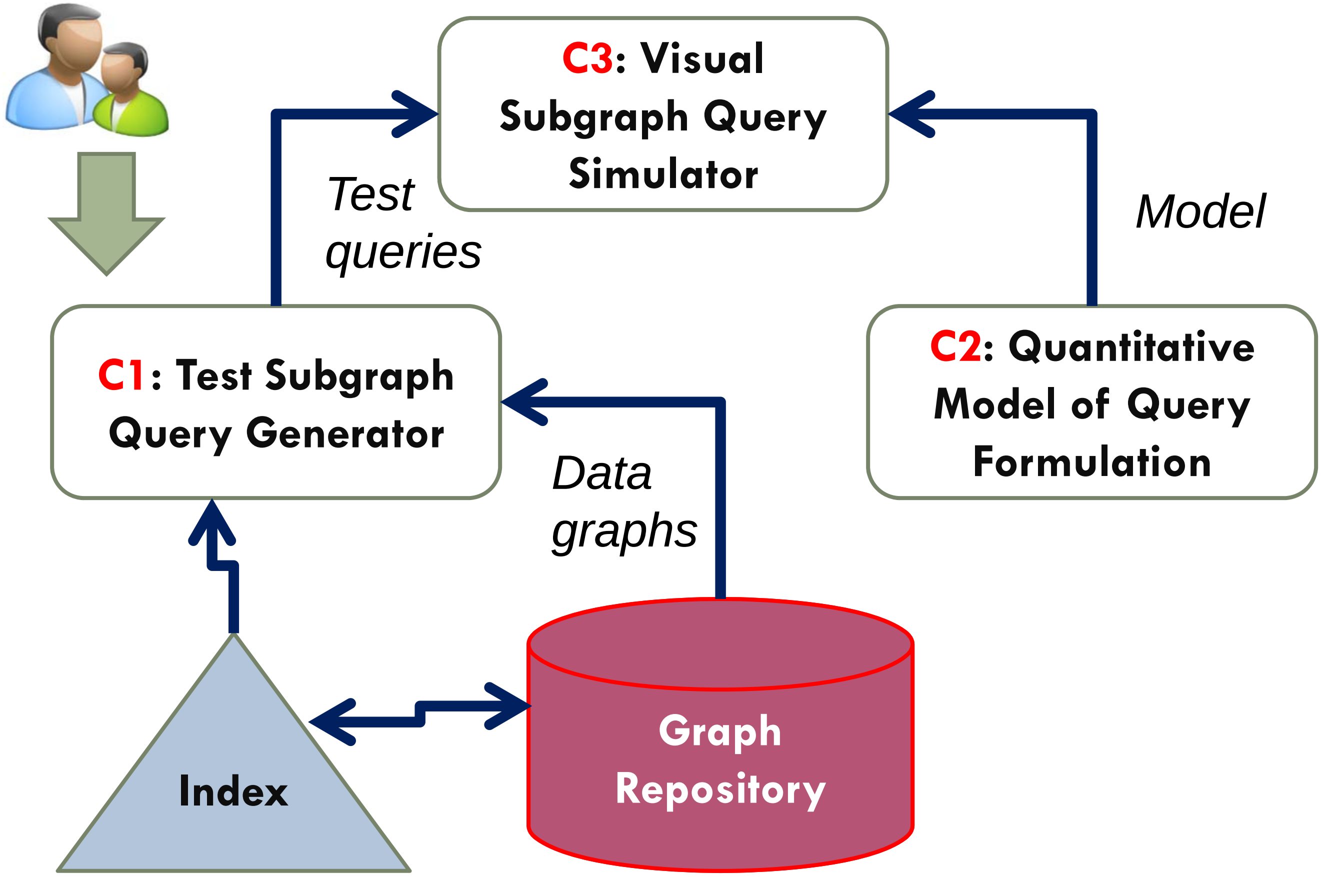
- Queries need to be visually constructed by users
- GUI latency is critical for performance study

Challenge

□ Users are expensive!

□ How do we simulate visual query formulation?

VISUAL [ICDE 15, TKDE 17]



C3: Visual Subgraph Query Simulator

Test queries

Model

C1: Test Subgraph Query Generator

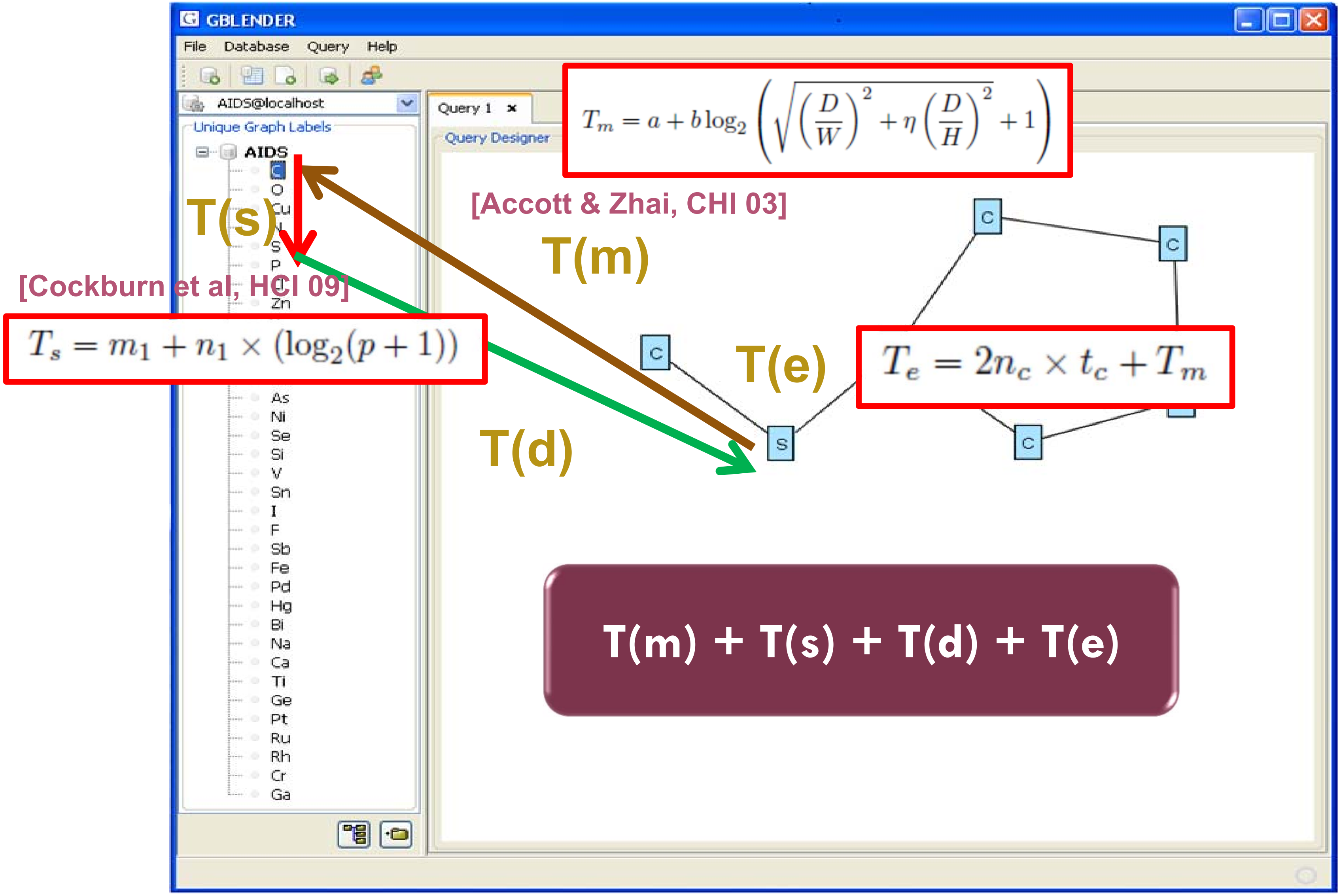
Data graphs

C2: Quantitative Model of Query Formulation

Index

Graph Repository

Quantitative Model for Query Formulation Time



$$T_m = a + b\log_2\left(\sqrt{\left(\frac{D}{W}\right)^2 + \eta\left(\frac{D}{H}\right)^2} + 1\right)$$

[Accott & Zhai, CHI 03]

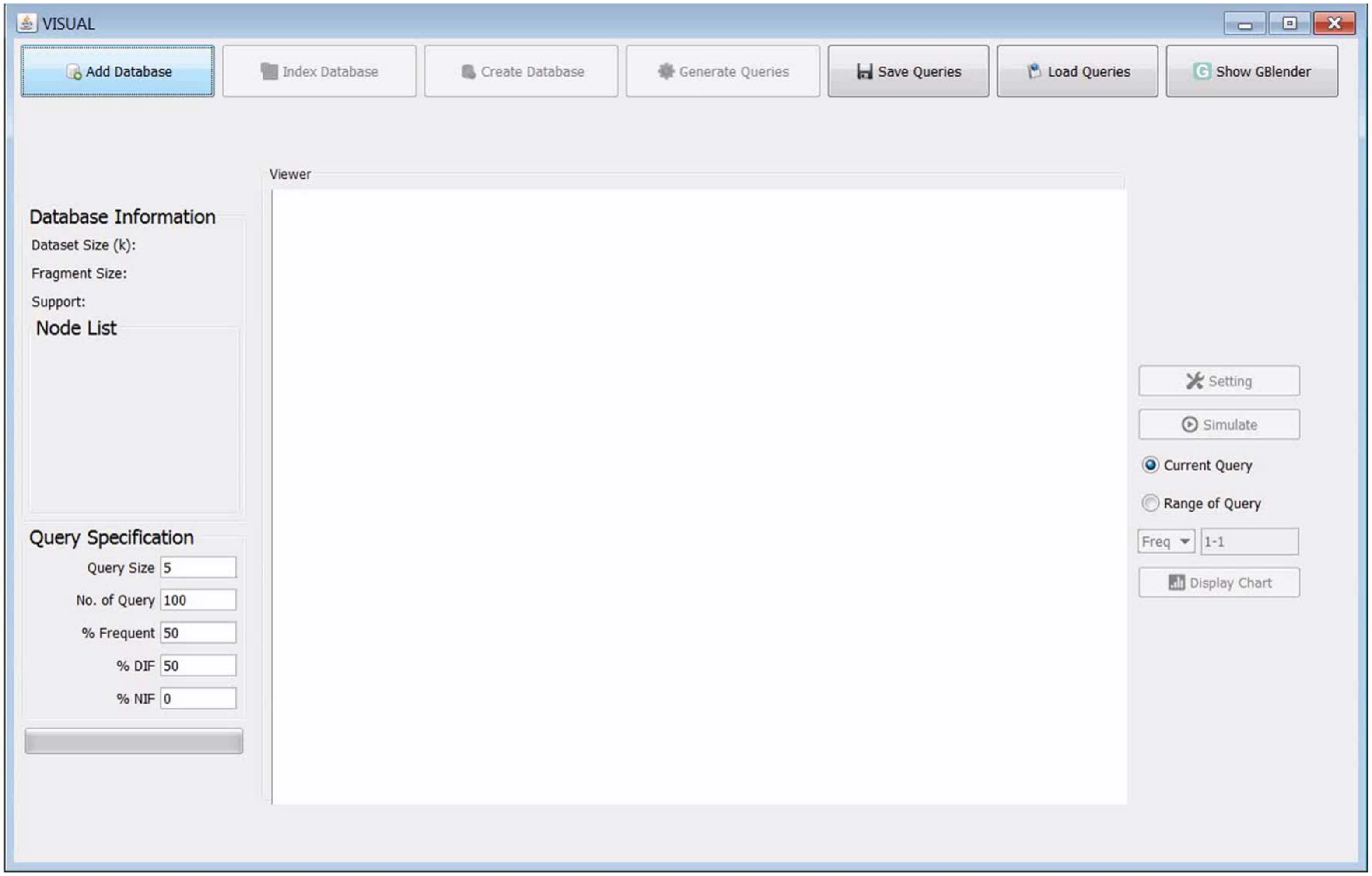
[Cockburn et al, HCI 09]

$$T_s = m_1 + n_1 \times (\log_2(p+1))$$

$$T_e = 2n_c \times t_c + T_m$$

T(m) + T(s) + T(d) + T(e)

VISUAL Demo



Database Information

Dataset Size (k):

Fragment Size:

Support:

Node List

Query Specification

Query Size 5

No. of Query 100

% Frequent 50

% DIF 50

% NIF 0

● Current Query

○ Range of Query

Freq ▾ 1-1

Query Results Exploration & Visualization

Next

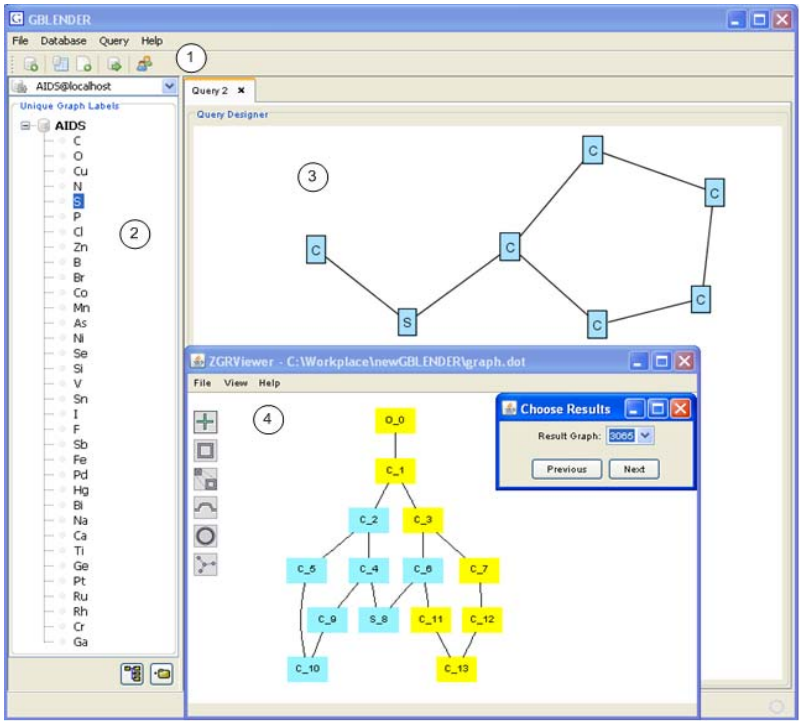
Query Results Exploration

Two Categories

- ❑ Very few efforts!
- ❑ Large set of small graphs vs large networks

Large set of small graphs

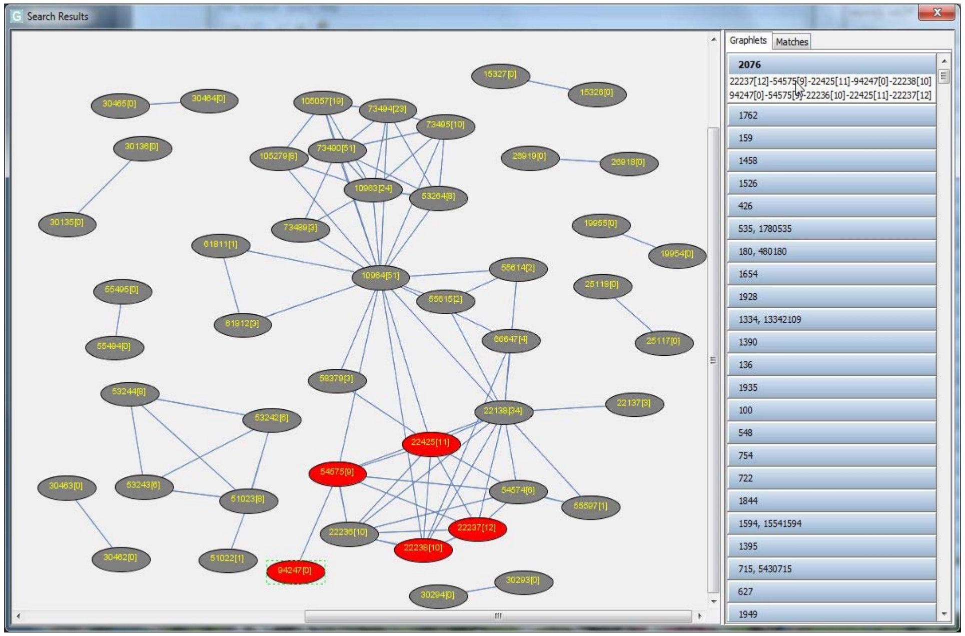
- Typically a decision problem
- Highlight a subgraph that matches the query
- [SIGMOD 10, ICDE 12]



Query Results Exploration

Large Networks

- Summarization-based (SLQ [SIGMOD 14])
- Supergraphlet-at-a-time (QUBLE [VLDBJ 14, SIGMOD 13])
- Feature-based (R2DB [ICDE 12])



2076

22237[12]-54575[9]-22425[11]-94247[0]-22238[10]
94247[0]-54575[9]-22236[10]-22425[11]-22237[12]

1762

159

1458

1526

426

535, 1780535

180, 480180

1654

1928

1334, 13342109

1390

136

1935

100

548

754

722

1844

1594, 15541594

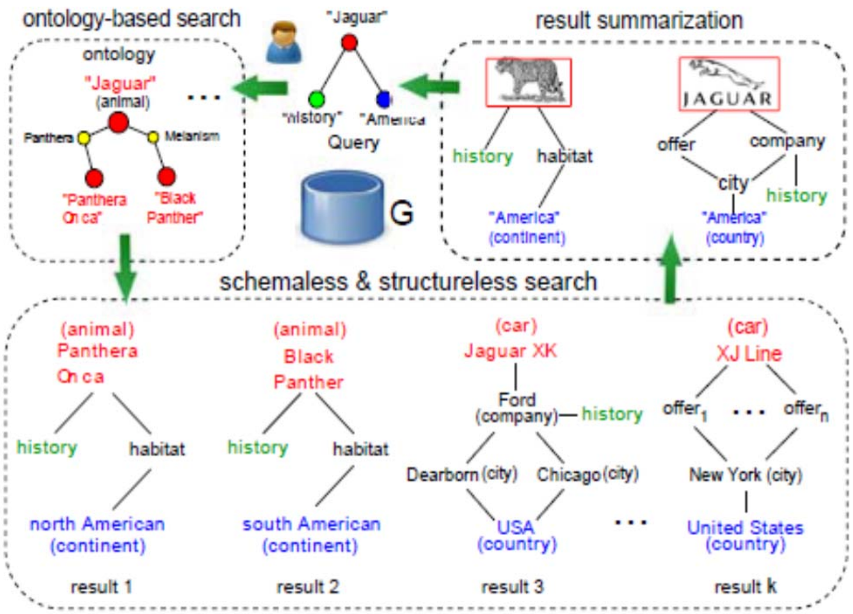
1395

715, 5430715

627

1949

Supergraphlet-at-a-time

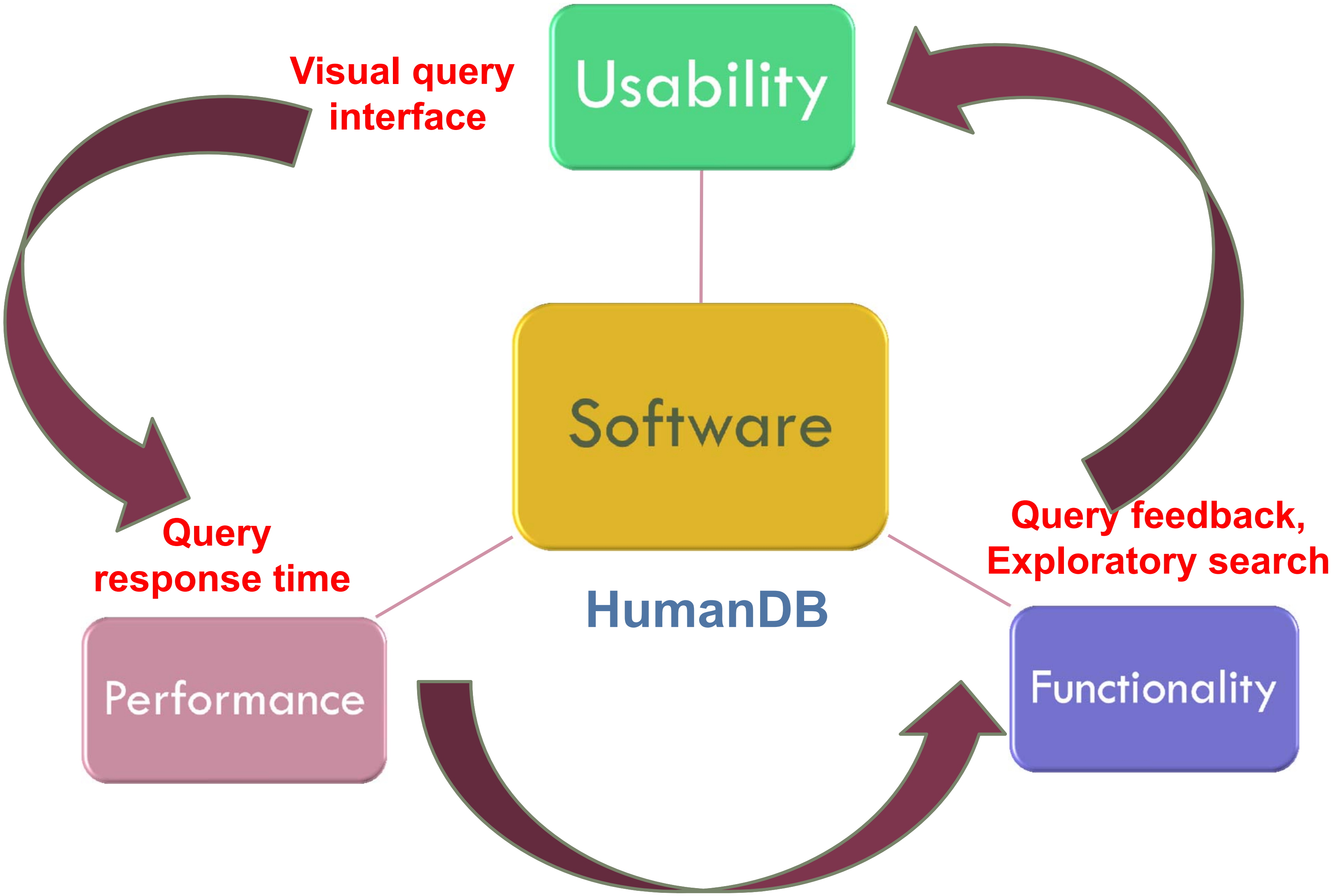


Summarization-based

Next

Conclusions

Bridging Usability, Performance, Functionality

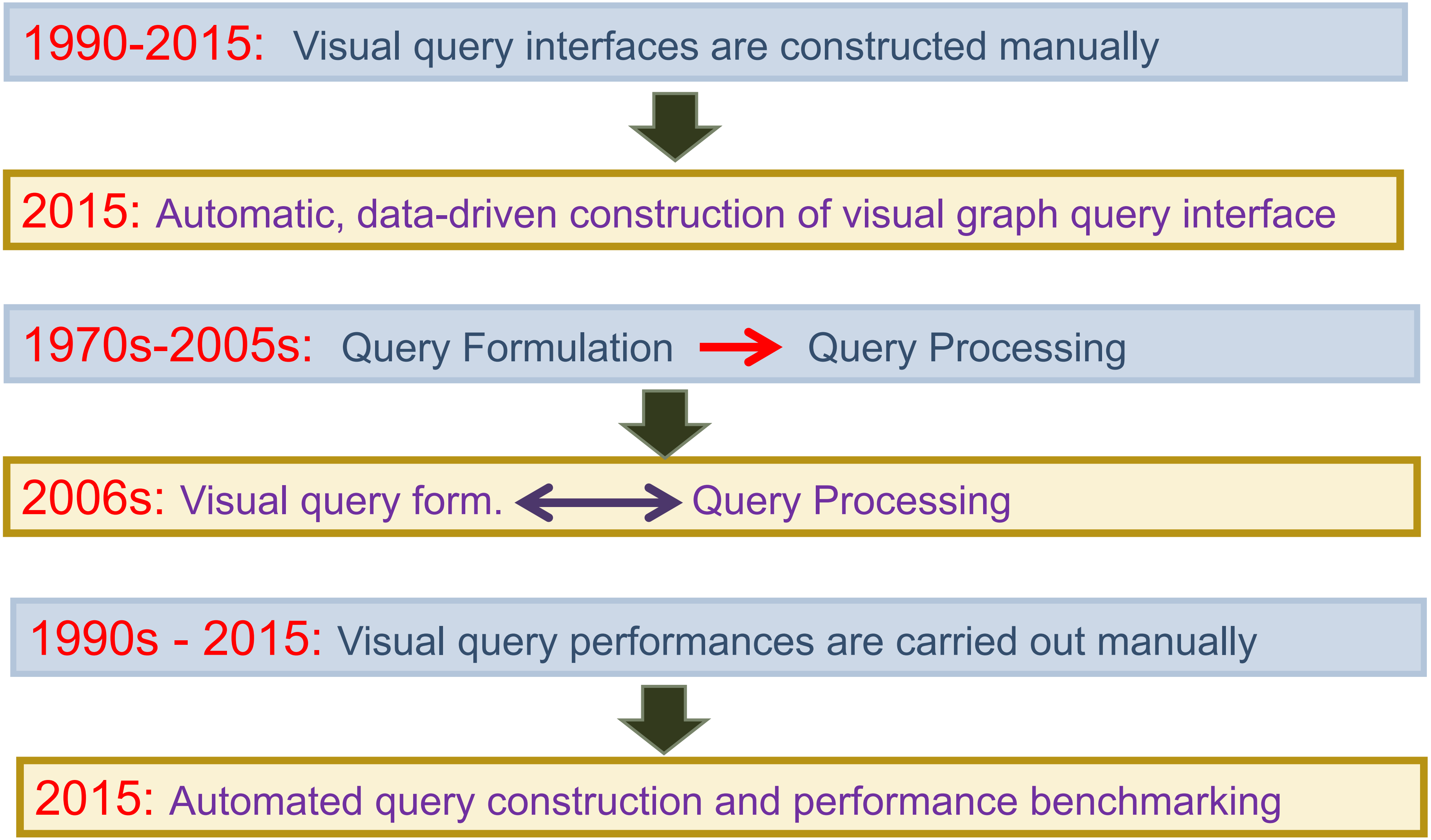


Visual query interface

Query response time

Query feedback, Exploratory search

Shifting Traditions



1990-2015: Visual query interfaces are constructed manually

2015: Automatic, data-driven construction of visual graph query interface

1970s-2005s: Query Formulation → Query Processing

2006s: Visual query form. ↔ Query Processing

1990s - 2015: Visual query performances are carried out manually

2015: Automated query construction and performance benchmarking

Open Research Problems

- More complex graph queries: Homomorphism-based queries, multi-attribute queries, graph simulation
- Visually querying massive graphs
- How can we extend data-driven GUI construction to be aesthetics-aware?
- Multi-faceted exploration and visualization of query results
- HCI-awareness with other types of data?

Final Words

HCI-aware Data Management

- Towards usable data management systems
- Making visual query interface design data-driven
- Making query formulation & processing HCI-driven
- Novel area of research

Multi-disciplinary effort:

Data management
HCI
Cognitive psychology

Broad goal

Stimulating a cultural shift in our thinking by HCI, cognitive psychology and data management to “work” together

Thank You!