The GOSSPLE social network

Davide Frey
INRIA, Rennes

Principal Investigator: Anne-Marie Kermarrec (INRIA)

The team: X. Bai, M. Bertier, A. Boutet, D. Frey, K. Huguenin, V. Leroy, A. Moin, G. Tan, C. Thraves (INRIA) & R. Guerraoui (EPFL)

The Web revolution

Web content is generated by you, me, your friends and millions of others

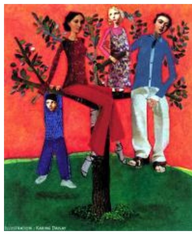
(Two faces of) social networking has taken off at an unexpected scale and speed

There is a gold mine of information out there

Are we all happy with Google?

A real-world example

Alice's family



English-speaking Rennes

Accomodation

Baby-sitting

« English-Speaking baby-sitter Rennes »

Same request in Lille

1- AMERICAN GIRL, NATIVE ENGLISH SPEAKING BABYSITTER IN LILLE.
2- Assistants in France • View topic - English-speaking Baby-sitting.
3- [PDF] GOSSPLE: personalized and decentralizedqueries

My own request

Gossple Paper ☺

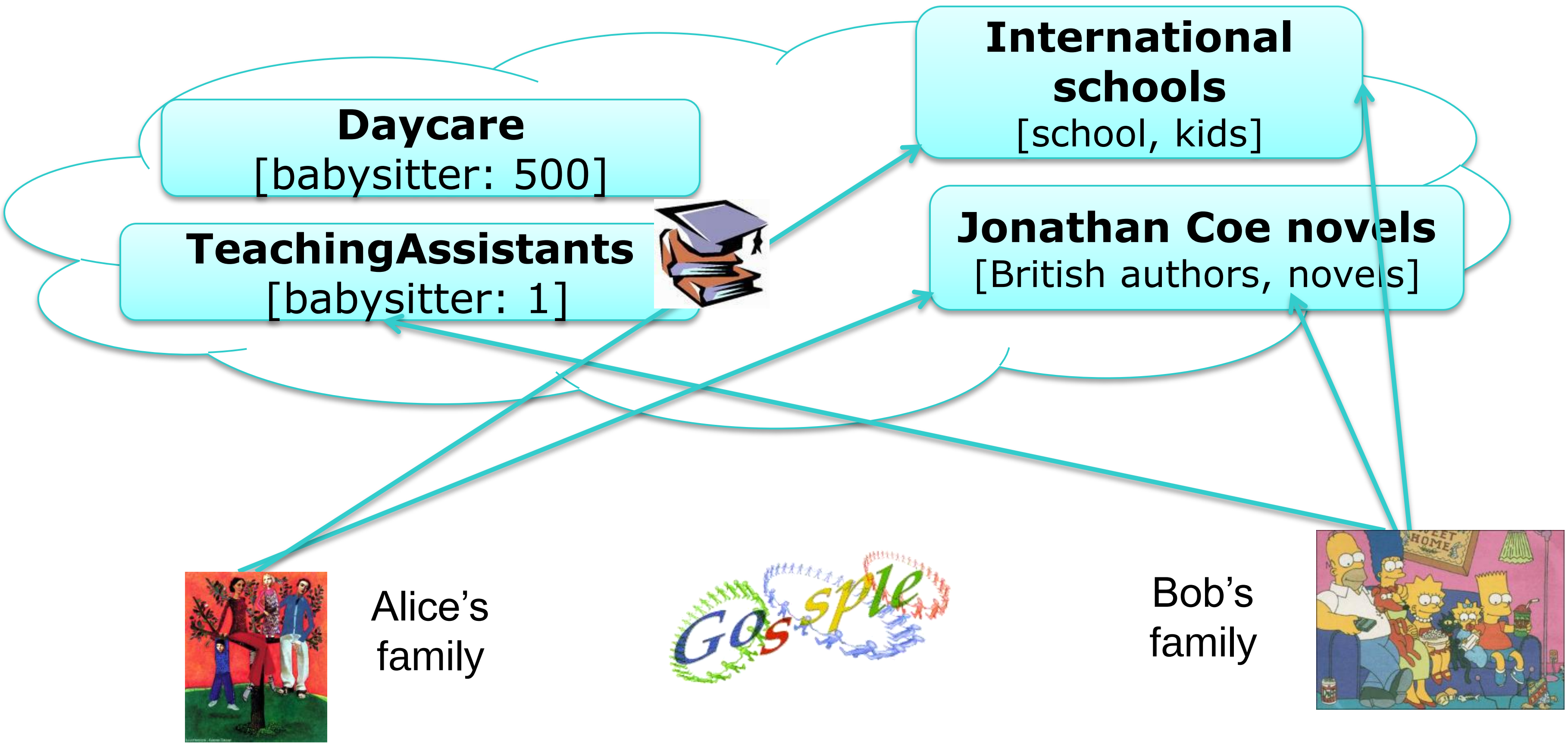
What if Bob knew?

Personalization: explicit social connections do not help

- 10/26/2009: Google Social Search (I finally found my friend's New York blog!)
- PeerSpective [MGD06]
- Network-Aware search [ABLS08]

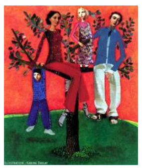
Implicit social connections can help

Personalized query



Leveraging implicit connections

Query expansion



English speaking baby sitter

↓

Query expansion

↓

English speaking baby sitter
Teaching assistant

Top-k

[English speaking, baby sitter]

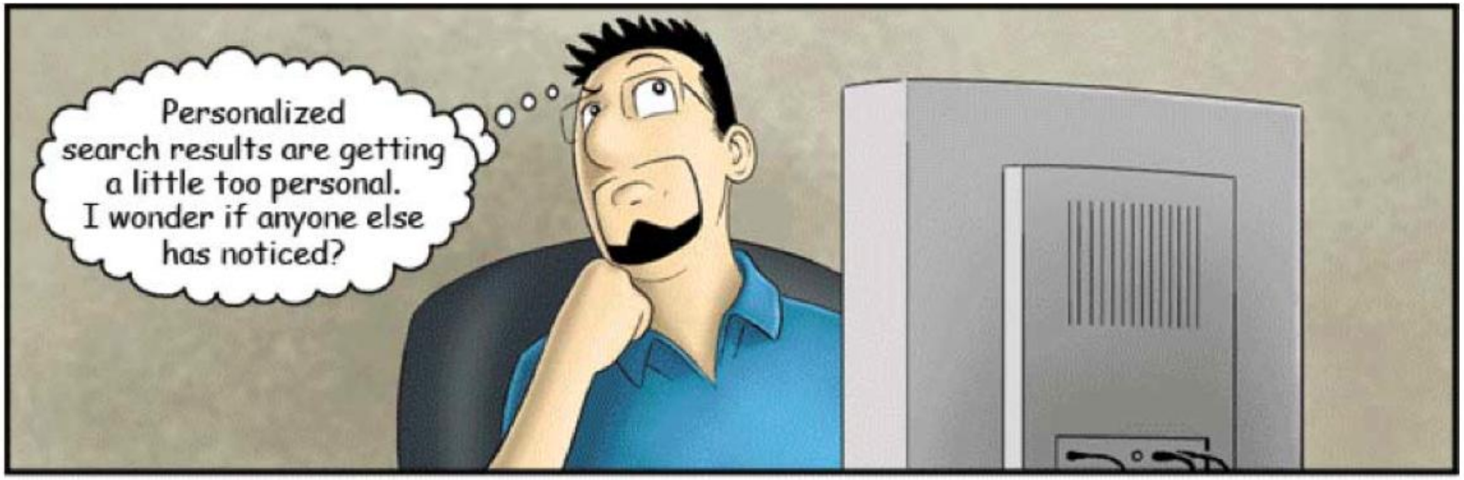
↓

Top-K

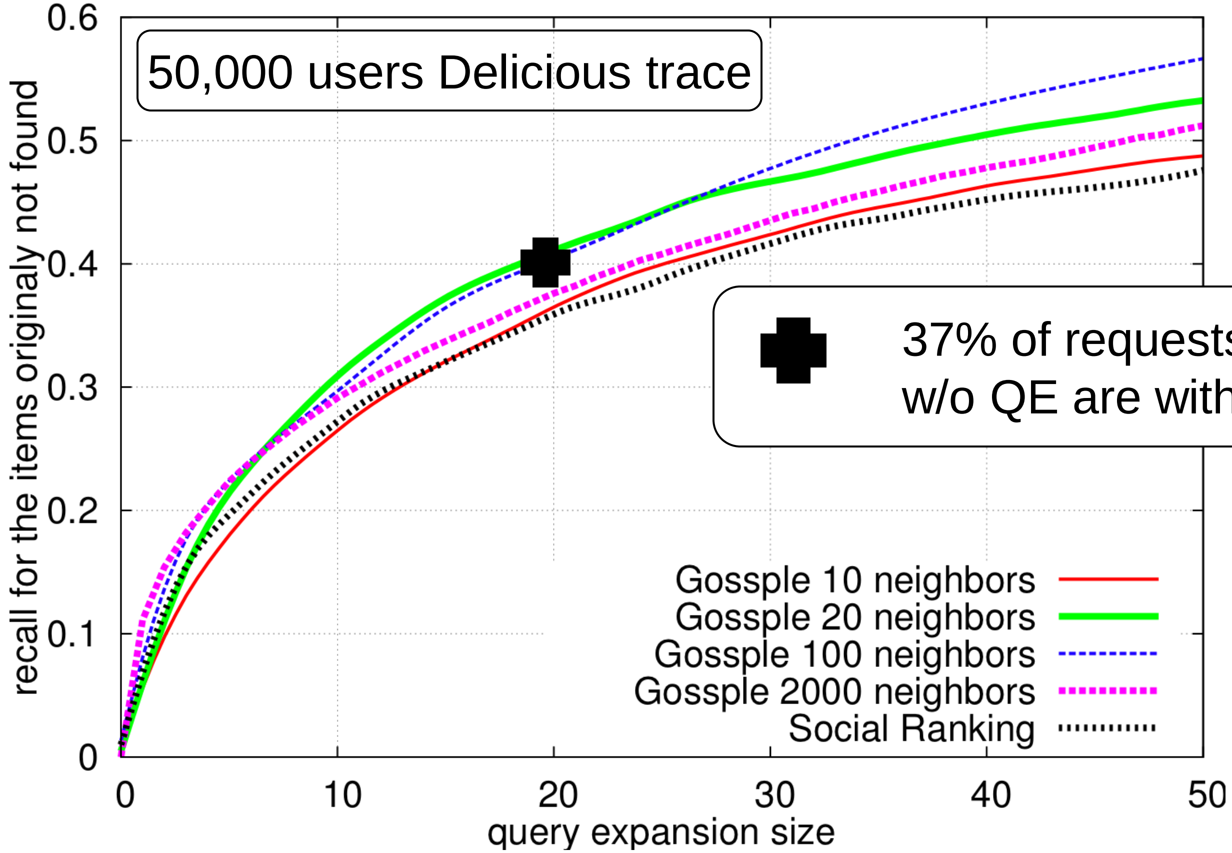
↓

http://www.assistant.fr

A case for personalization through **implicit** social connections



Personalized query expansion



37% of requests not satisfied w/o QE are with 10 neighbors

Achieving personalization in large systems

Through decentralization

Personalisation calls for decentralization

Scalability/Reactivity

- Enable to manage metadata at a user's granularity
- Cope with dynamics

What else?

If you only knew the power of the Dark Side.
– Darth Vader

Personalisation calls for decentralization (2)

Fighting the Big Brother is watching you's attitude

- e.g. New terms of uses of Facebook (2009), Beacon feature of Facebook (2007)
- Twitter

__You retainyourrights__to any Content yousubmit, post or display on or through the Services. By submitting, posting or displaying Content on or through the Services, __yougrant us a worldwide, non-exclusive, royalty-freelicense__(with the right to sublicense) to use, copy, reproduce, process, adapt, modify, publish, transmit, display and distributesuch Content in any and all media or distribution methods (nowknown or laterdeveloped).

Complex without global knowledge

GOSSPLE in a Nutshell

Personalizedapproach to favor individuals as opposed to large masses

Decentralized approach to provide scalability, reactivity and privacy

Applications: query expansion, top-k, search, recommendation, …

The Gossple social network

The Gossple social network

Provide a node with the $c << N$ "best friends"

- How to decide which nodes should befriends?
- How to discover such friends?

Which nodes should be "friends"?

- Tagging similarity
- Cosine similarity
- Multi-interest similarity

Interest-based Web 2.0 applications

- Users characterized by a profile
- Collaborative tagging systems

- Model
  - U(sers) × I(tems) ×T(ags)
  - $Tagged_u(i, t)$: User u annotates item i with tag$t$
  - $Profile(u)=\{Tagged_u(i, t)\}$

1: Tagging similarity

- *Efficient network-aware search in collaborative tagging sites* [ABLS, VLDB'08]
- User score: common tagging actions

2: Item cosine similarity

Normalized overlap

- bigger overlap increases the score
- no shared interests decreases it
- directly takes into account the weight of items

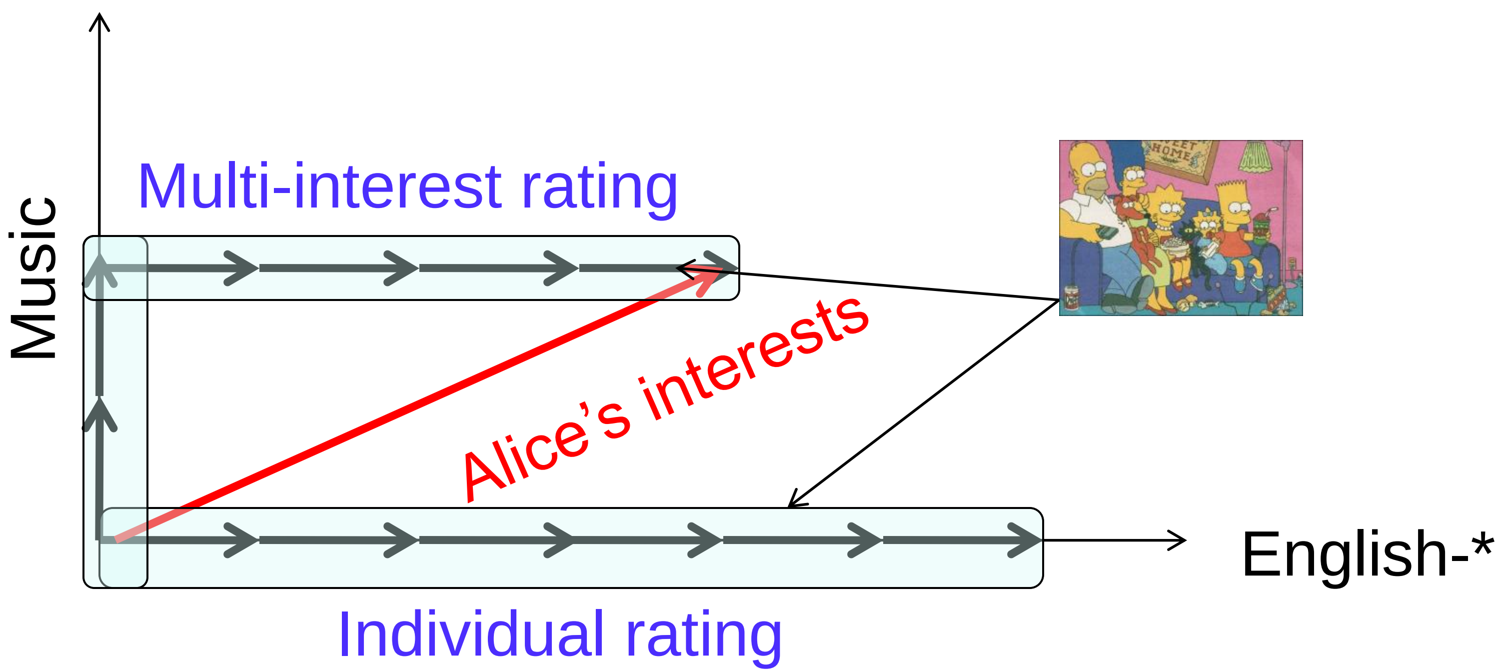
$$\cos(\vec{v}_1, \vec{v}_2) = \frac{\vec{v}_1 \vec{v}_2}{\|\vec{v}_1\| \|\vec{v}_2\|}$$

$$ItemCos(\vec{u}_1, \vec{u}_2) = \frac{|Items(\{\vec{u}_1\})| \bigcap |Items(\{\vec{u}_2\})|}{\sqrt{|Items(\{\vec{u}_1\})| . |Items(\{\vec{u}_2\})|}}$$

Individual rating might be too restrictive

3: Coping with multi-interests

Item cosine similarity: favours specific and dominant interests



3: Multi-Interest cosine similarity

- Rate the set of friends **as a whole** instead of each potential neighbor
- Choose a set of neighbors that covers the user's interests

$$SetItemVect(set) = \sum_{p \in set} \frac{(ItemVect(p) \otimes ItemVect(n))}{\left\| ItemVect(p) \right\|}$$

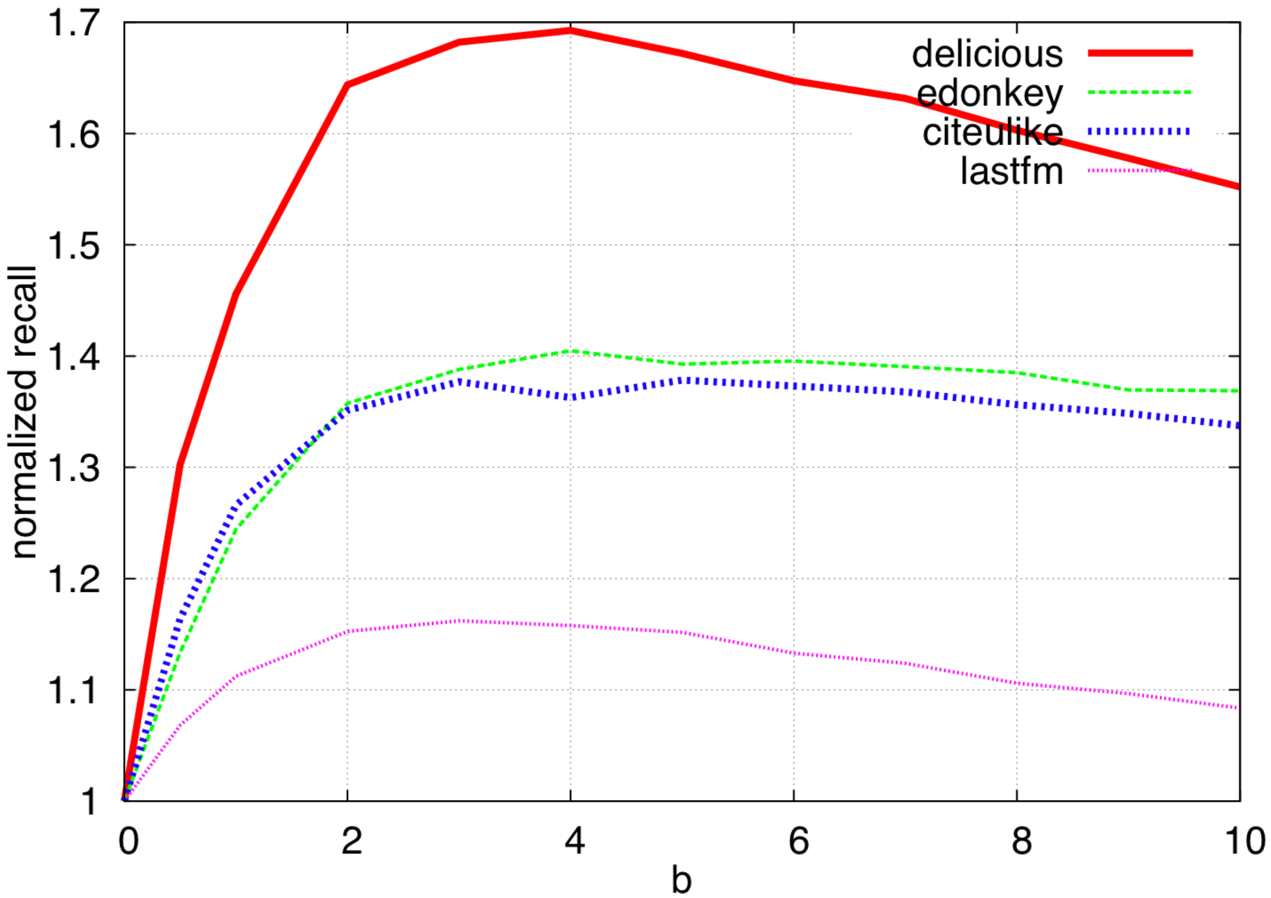
Items of interest for nodes in Neigbhor(n)

Normalized not to take into account non shared interests

$$SetScore(n, set) = SetItemVect(set).ItemVect(n) *$$
$$\cos(SetItemVect(set)., ItemVect(n))^b$$

Distribution

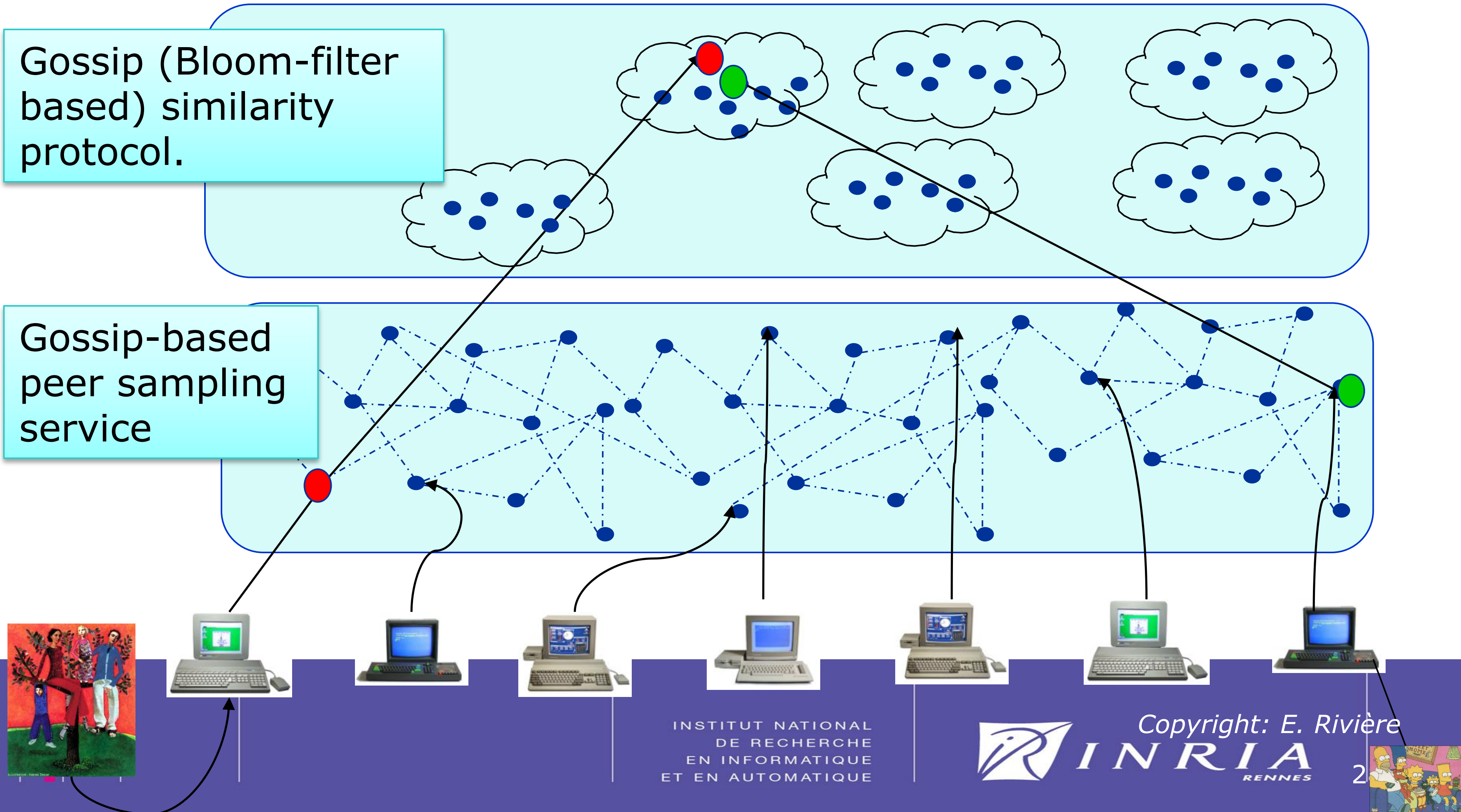
How good are Gossple friends?



How to discover the *c* "best friends"?

Through gossip

Piling up gossipprotocols



Gossip-based computing

Parameter Space: Peer selection, Data exchanged, Data processing)

Active thread

Wait (T time units)
P <- **selectPeer**()
myDescriptor<- (my@,0)
buffer <- merge
    (**dataExchanged**(view),{myDescriptor})
send buffer to p

receive buffer from p
    buffer <- merge(buffer, view)
view<- **dataProcessing**(buffer)

increaseage(view)

Passive Thread

(p,view_p) <- waitMessage()

myDescriptor<-(my@,0)
    buffer <-merge (**dataExchanged**(view),{myDescriptor})
send buffer to p

-increaseage(view)
buffer <- merge(view_p, view)
view<-**dataProcessing**(buffer)

increaseage(view)

Overlay maintenance

	LpbCast [EGKK 01,03]	Cyclon [VGS 05]	T-man [JMB 09]
Peer selection	Random	Oldest	Random
Data exchange	List of neighbours	½ List of neighbours	List of neighbours
Data processing	Random merging	Age-based merging	Proximity Based merging

Decentralized computations

	Aggregation [JMB 05]	System size Estimation	Slicing [JK 06]
Peer selection	Random	Random	Random
Data exchange	value	value	Attribute value Random value
Data processing	Aggregation Average	Aggregation	Attribute/random matching

Gossple social network

Friends

@IP:port	132.154.8.5:2020	
Bloom Filter	010111011001	
Profile	www.inria.fr:inria, computer www.assistants.fr: baby-sitter, english ...	
Update time	5	

c entries

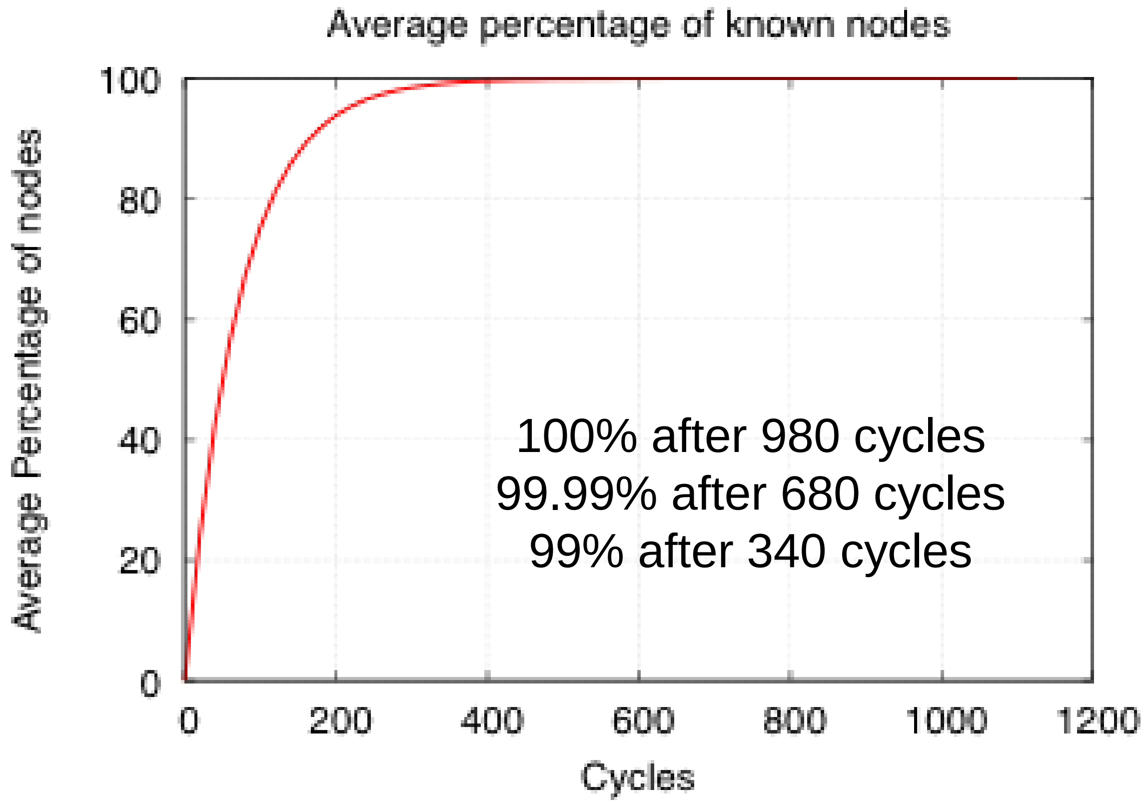
Uniform sample

@IP: port	102.14.18.1:2110	
Bloom Filter	100100000110	
Update time	30	

k entries

Uniform sampling

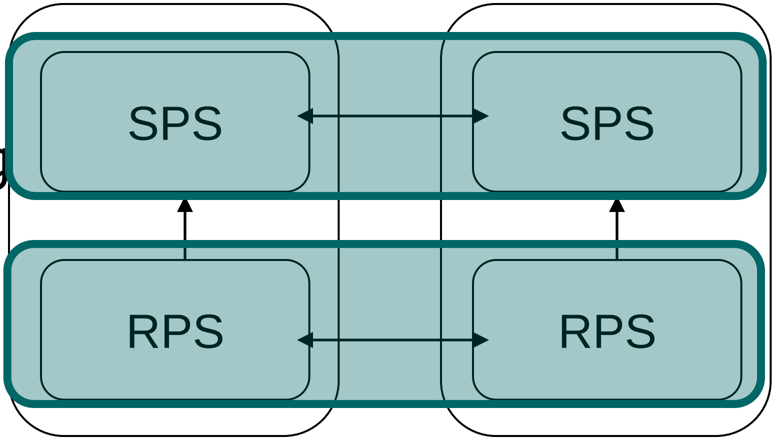
- O(n/k log n) iterations.



INSTITUT NATIONAL
DE RECHERCHE
EN INFORMATIQUE
ET EN AUTOMATIQUE

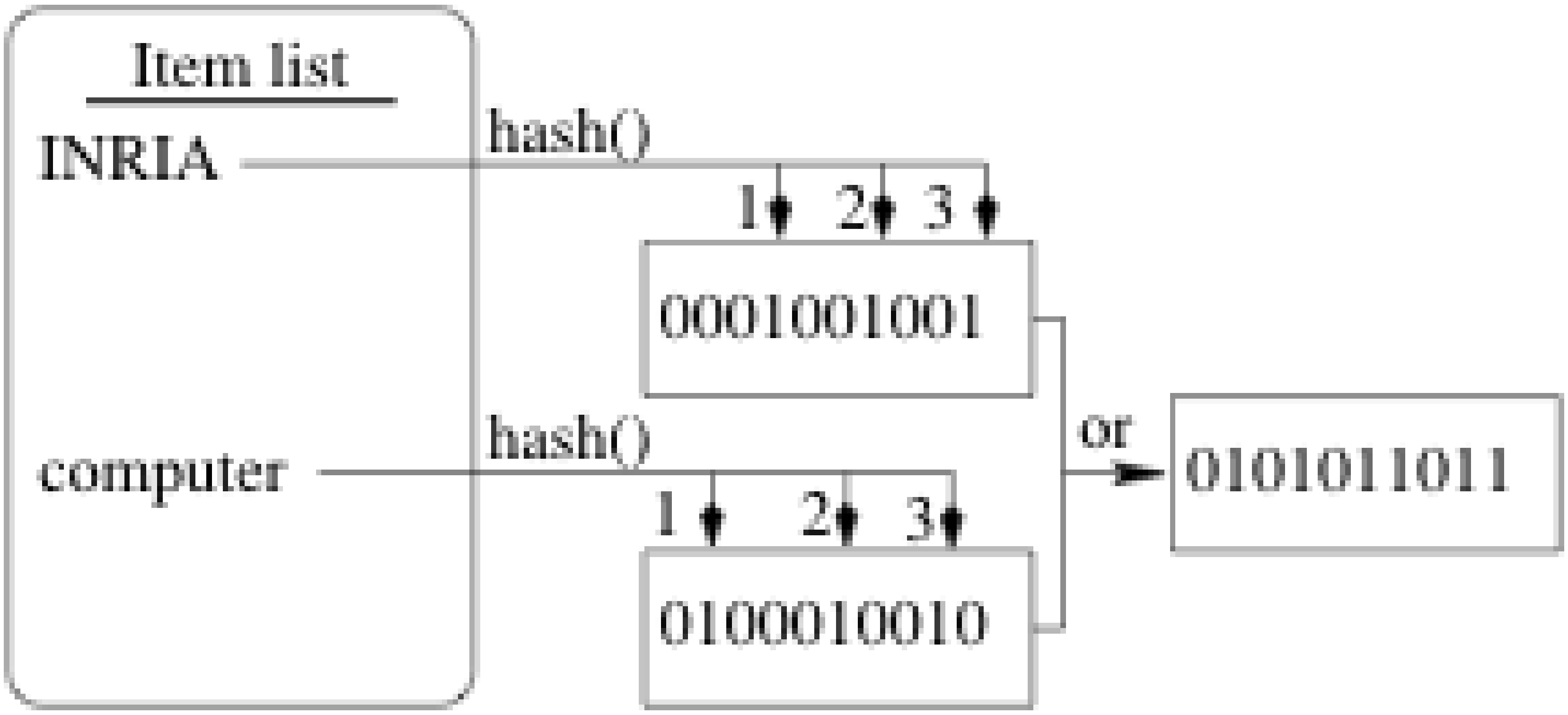
Building the social network

- Two gossip protocols
  - Similarity-based Peer Sampling
  - Random Peer Sampling

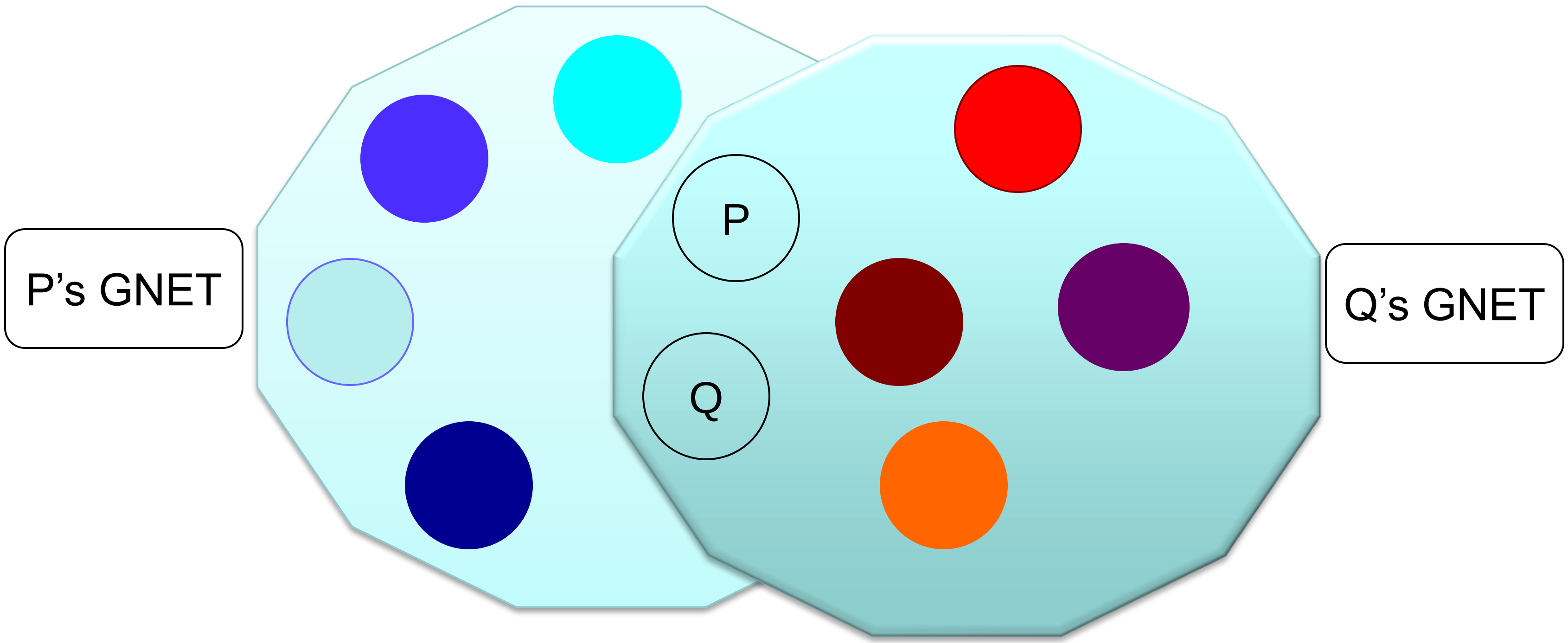


- When *p*encounters *q*
  - Evaluate distance between*p*
    - and q, based on individual **similarity** metric
    - and potential new view, based on **set similarity** metric
  - Use of Bloom filters to limit the communication overhead

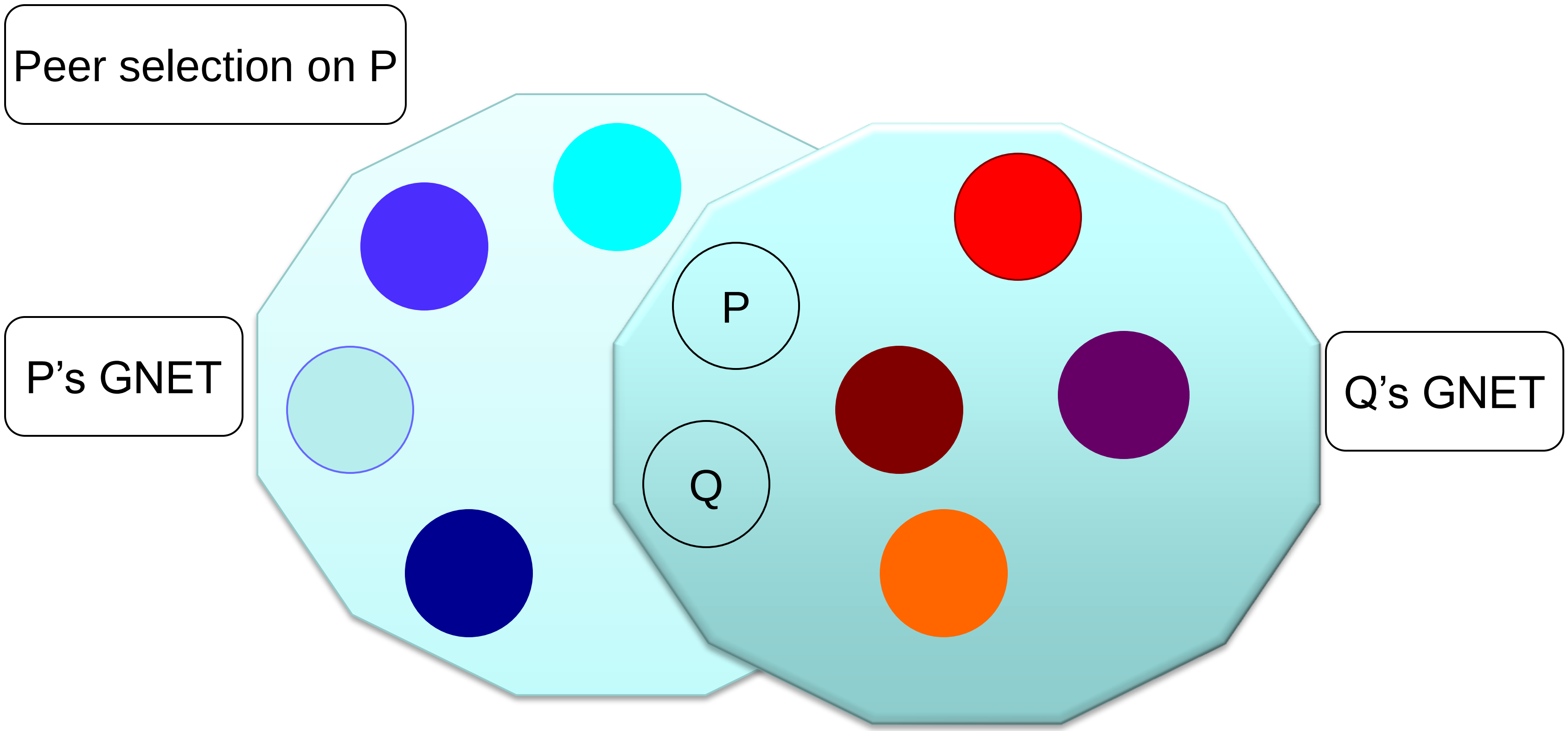
Bloom filter



Similarity Peer Sampling

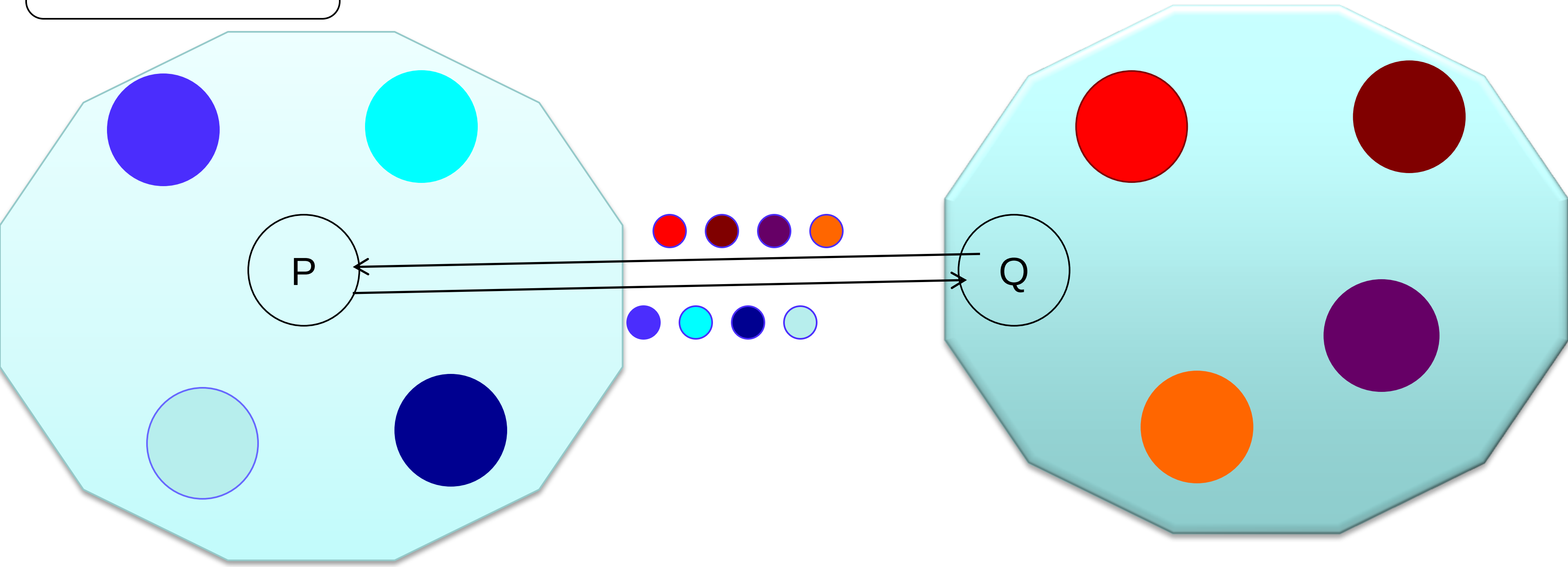


Similarity Peer Sampling

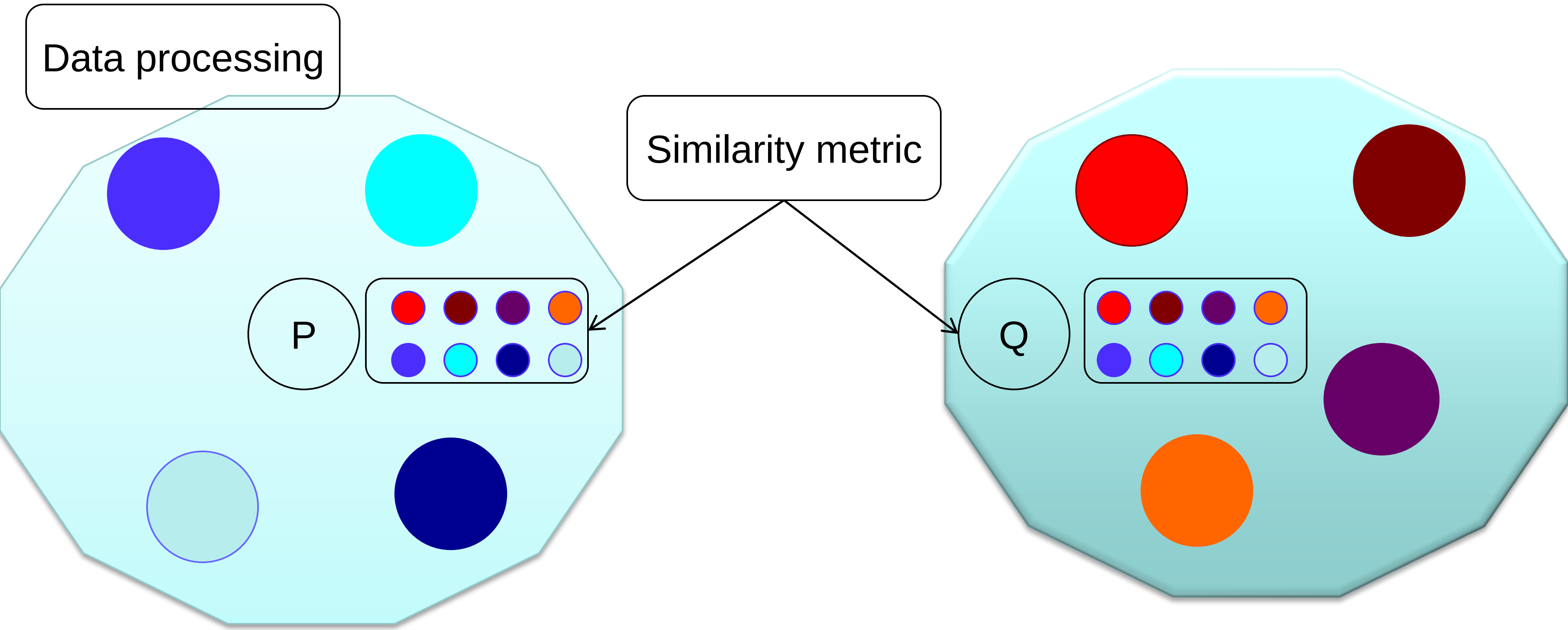


Similarity Peer Sampling

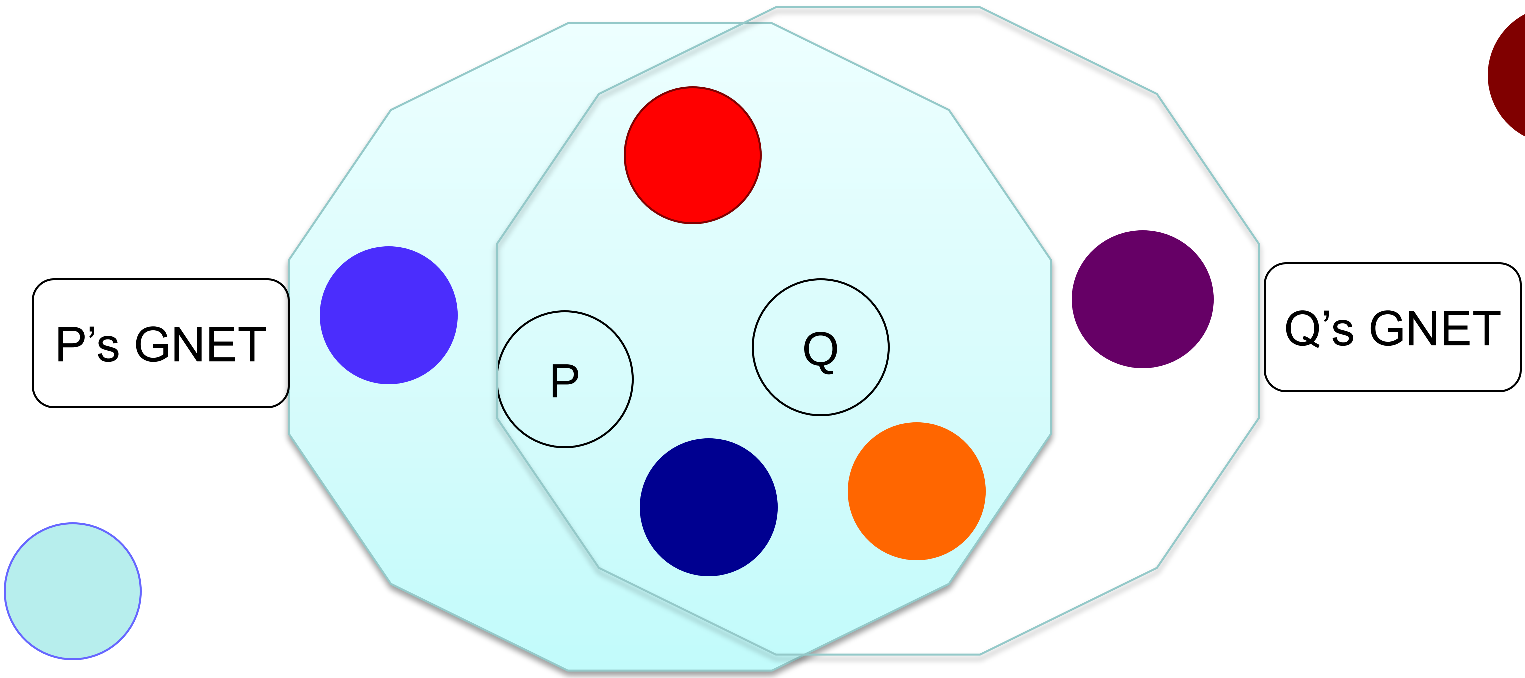
Data exchange



Similarity Peer Sampling



Similarity Peer Sampling

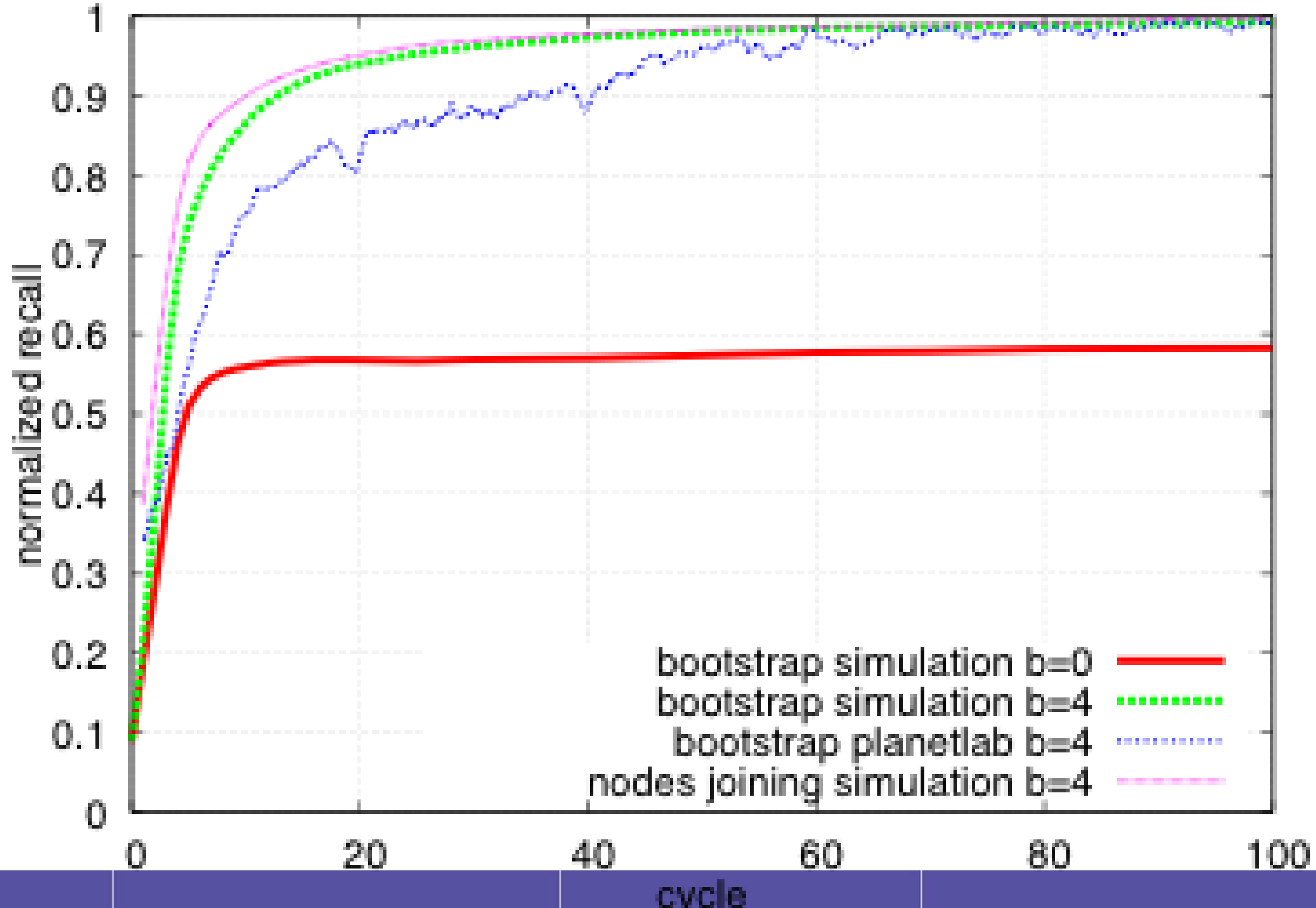


Multi-interest protocol

- Score of any combination: NP hard
- Heuristic: Starting from en empty view, builds the best view of size one, then two etc.

```
DataProcessing ()
    Bestview ={}
    For setSize  from 1 to viewSize do
        Foreach candidate in candidateSet do
            candidateView=bestview U {candidate}
            viewScore=SetScore(candidateView}
        bestCandidate = candidate that got the highest viewScore
        bestView= best View U {bestCandiate}
```

Set item cosine similarity



Illustration

Collaborative top-k query

- Top-k Processing
  - Query $q = \{t_1, \ldots, t_n\}$
  - $Score(i) = f\,(Score_{t1}(i), \ldots, Score_{tn}(i))$
  - *k*items with highest scores as results

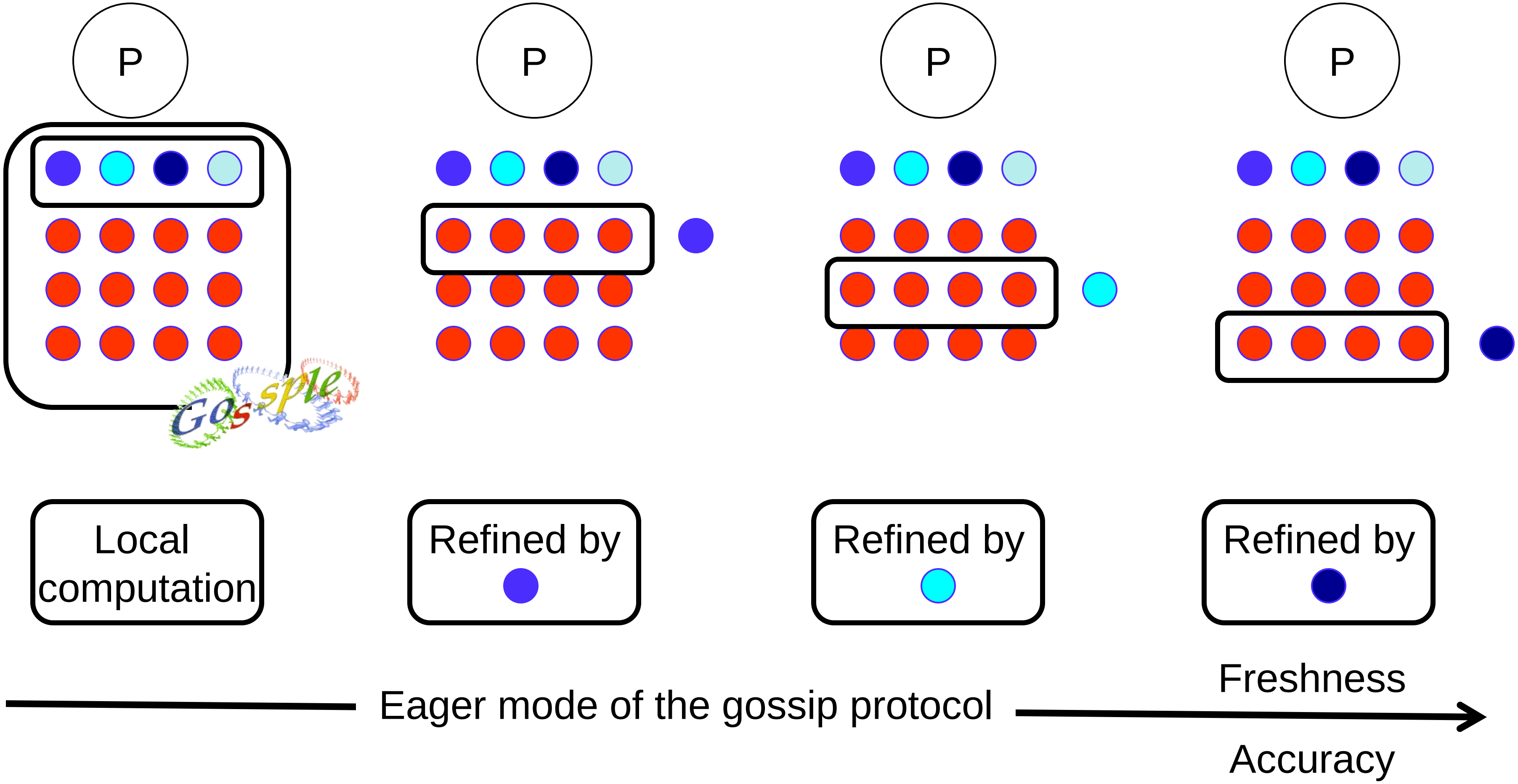
Personalized top-k query

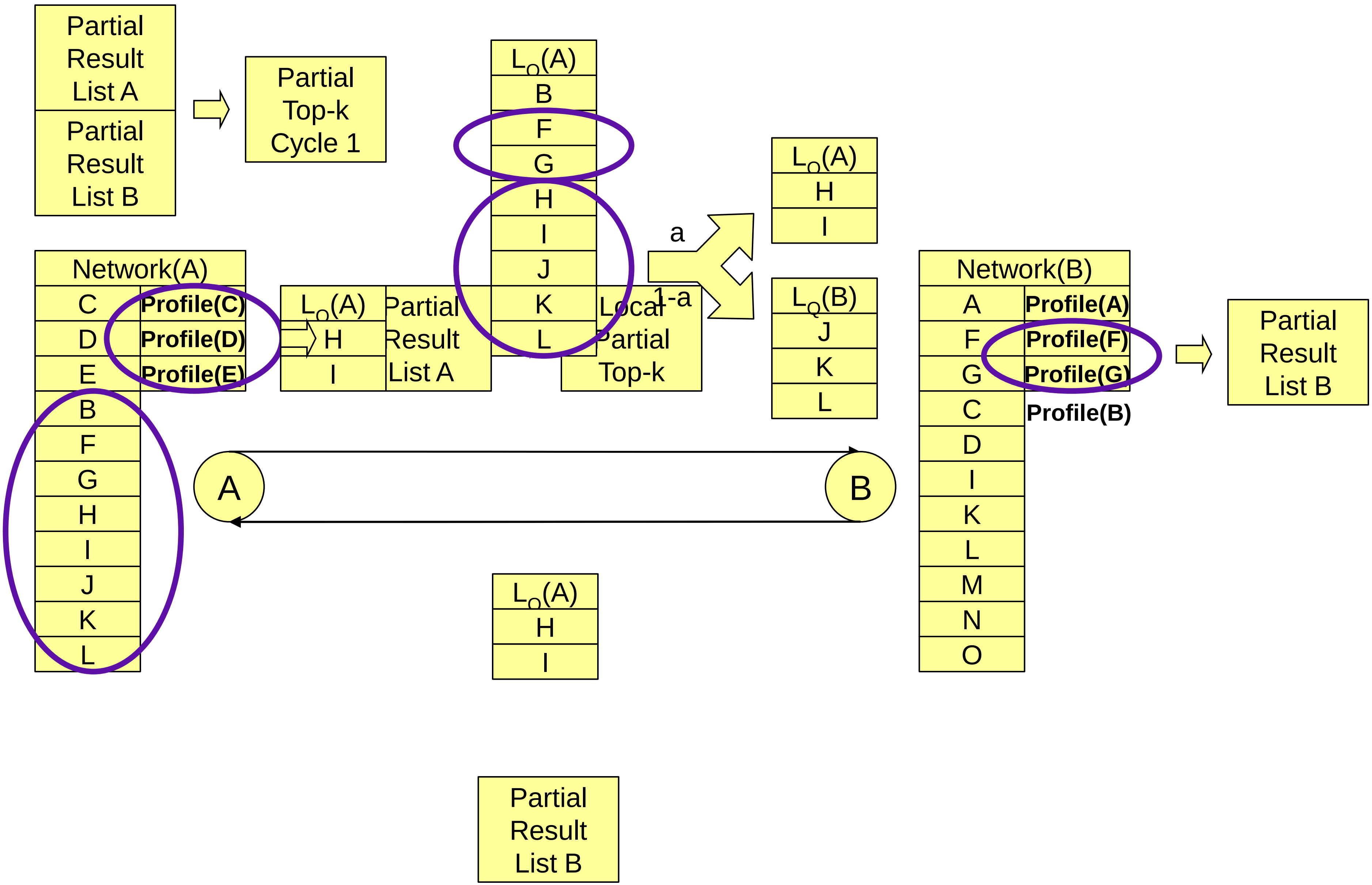
- Considered only similar users (threshold on the tagging similarity metric)

- Centralized approach [ABLS 08] do not scale
- Distributed local processing

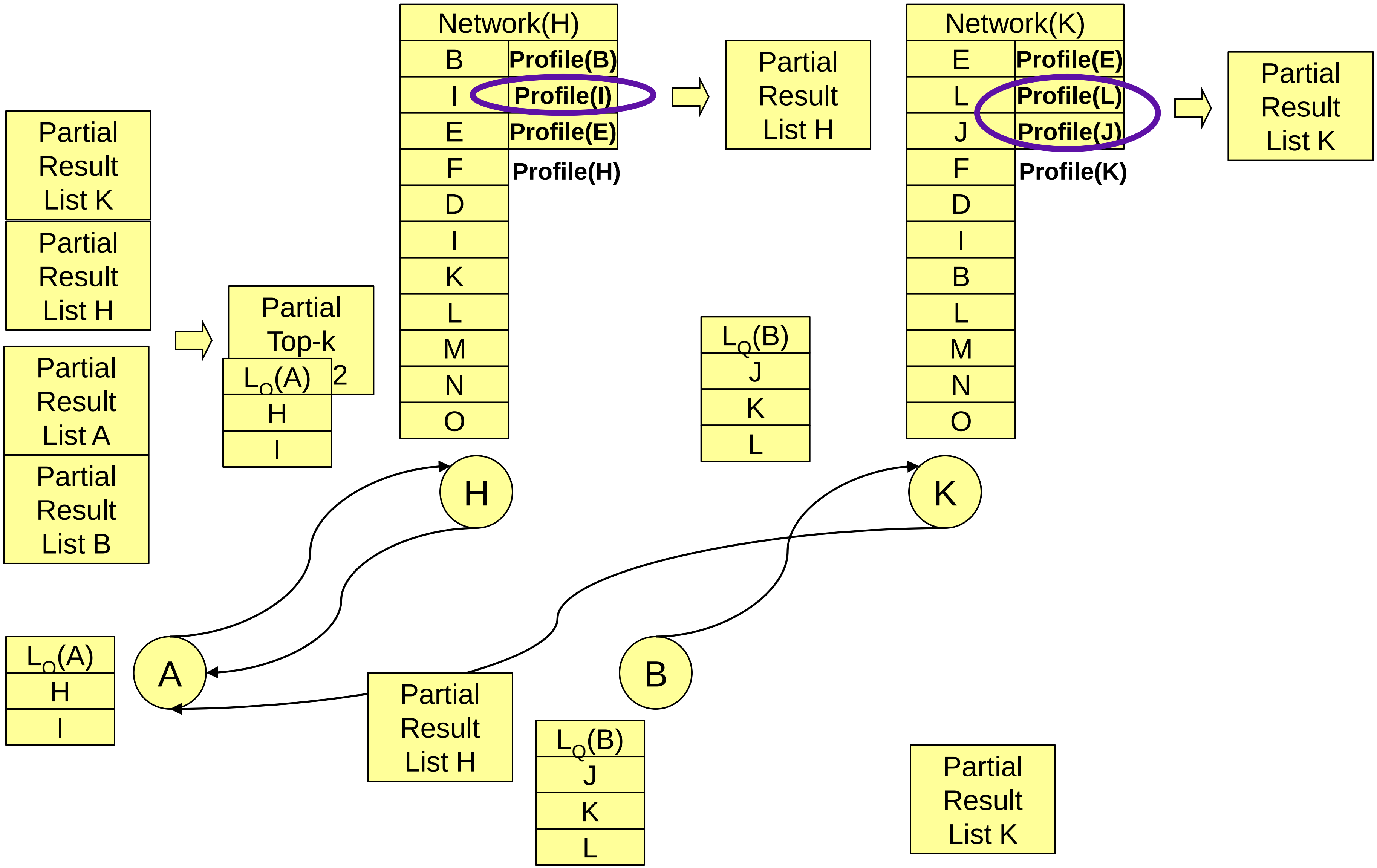
Partitioned processing [BBGKL, EDBT10]

Collaborative top-k processing





Partial Result List A
Partial Result List B
Partial Top-k Cycle 1
$L_Q(A)$
B
F
G
H
I
J
K
L
a
1-a
$L_Q(A)$
H
I
$L_Q(B)$
J
K
L
Network(A)
C
Profile(C)
D
Profile(D)
E
Profile(E)
B
F
G
H
I
J
K
L
$L_Q(A)$
H
I
Partial Result List A
Local Partial Top-k
Network(B)
A
Profile(A)
F
Profile(F)
G
Profile(G)
C
Profile(B)
D
I
K
L
M
N
O
Partial Result List B
A
B
$L_Q(A)$
H
I
Partial Result List B



Partial Result List K
Partial Result List H
Partial Result List A
Partial Result List B
Partial Top-k 2
$L_Q(A)$
H
I
Network(H)
B
Profile(B)
I
Profile(I)
E
Profile(E)
F
Profile(H)
D
I
K
L
M
N
O
Partial Result List H
$L_Q(B)$
J
K
L
Network(K)
E
Profile(E)
L
Profile(L)
J
Profile(J)
F
Profile(K)
D
I
B
L
M
N
O
Partial Result List K
H
K
A
B
$L_Q(A)$
H
I
Partial Result List H
$L_Q(B)$
J
K
L
Partial Result List K

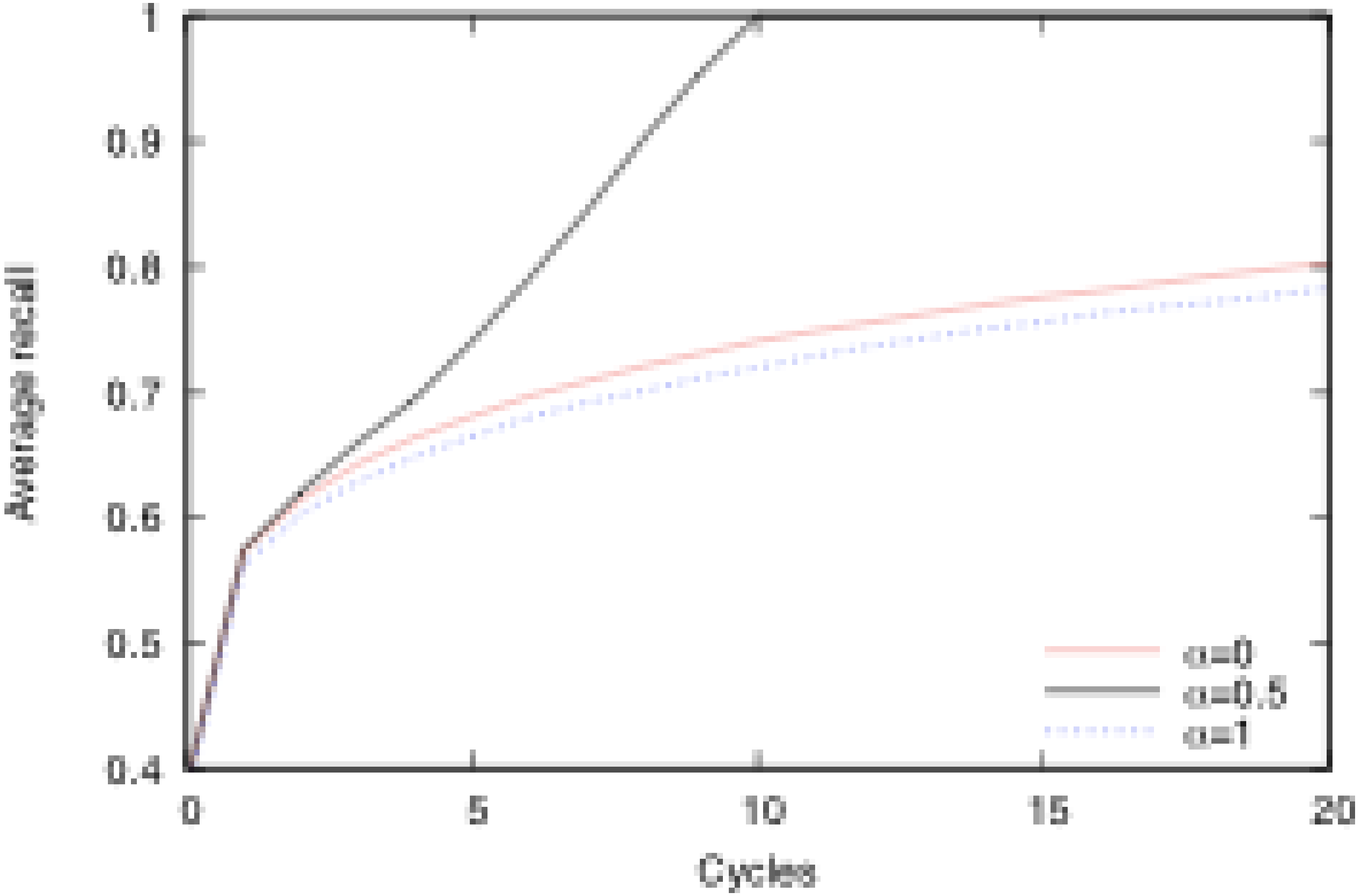
Personalized top-k processing

Collaborative top-kprocessing

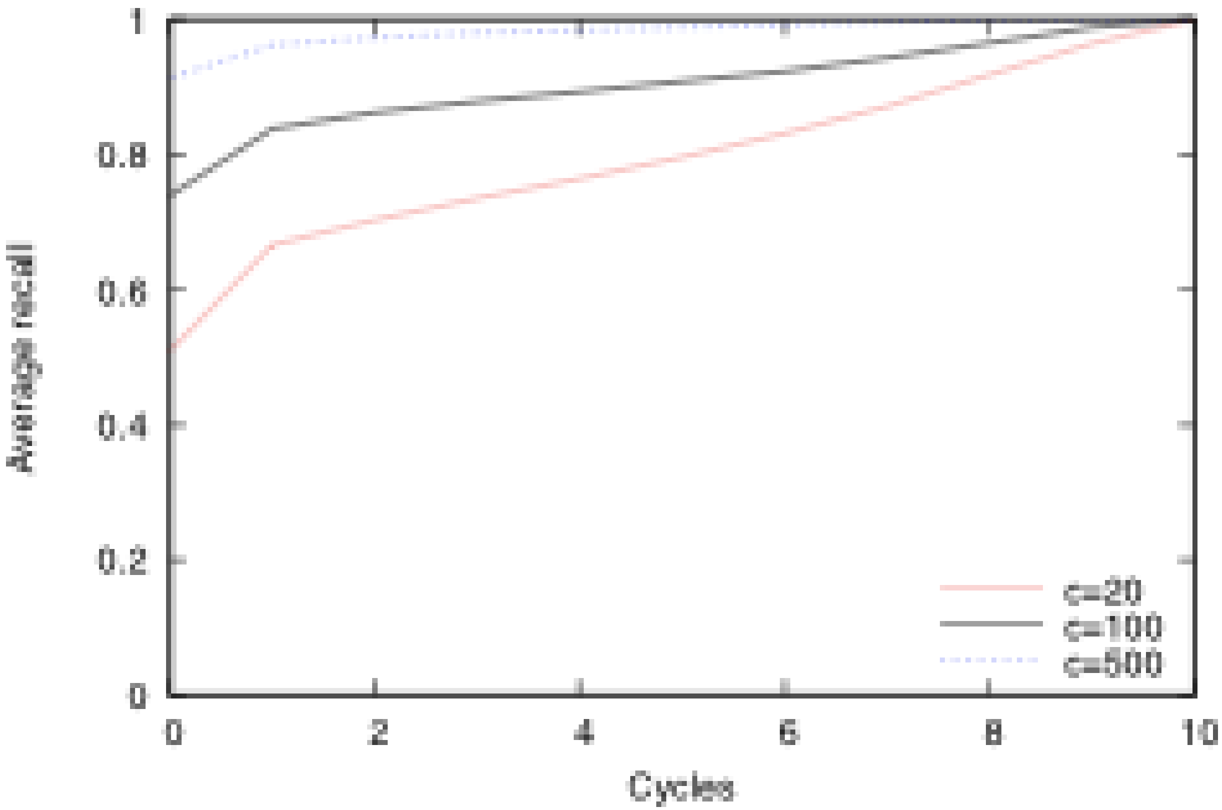
Stop condition

- the Gossple social network has been exhausted OR
- the user is happy

Evaluation (100,000 delicious users)



Impact of the number of stored profiles



To take away

A case for personalization:

- **implicit social connections**
- **efficient gossip protocol**

Applications

- **Query expansion**: harvest the personalized information, compute locally
- **Top-k processing**: discover the right helpers, compute remotely
- Recommendation/search

What I did not talk about

- Privacy
  - Gossip on behalf
- Arbitrary behaviors
  - Bombing
- Large-scale indexing

Thank you

SNDS Workshop. July 29, 2010, Zurich, Switzerland. Co-located with PODC 2010.

Submission Deadline: May 20, 2010