System Building: How does it help or hinder research?

Anthony K. H. Tung
National University of Singapore
atung@comp.nus.edu.sg
www.comp.nus.edu.sg/~atung/publication/system.ppt

Outline

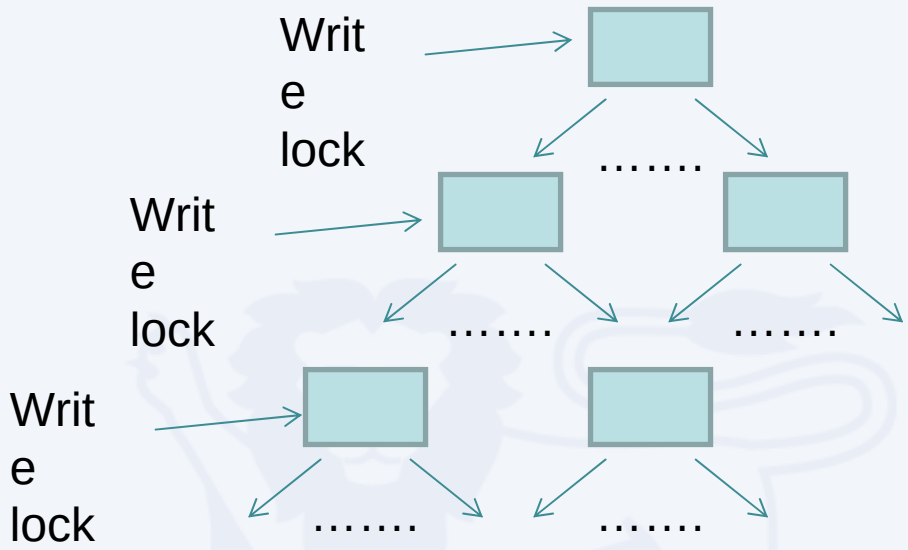
- Some fallacies of research we are facing and how system implementation can help
- What type of systems should we build?
- Should young faculties try to build system?
- Conclusion and Acknowledgement

Fallacy 1: Miss important factors that must be considered in real application

Example: Inventing a index for moving objects that have very fast query performance

Then concurrency control come in!

Updates lock up the pages and throughput in term of number of queries/s and updates suffers…



Expect to see more of such things with the popular use of R-tree etc. for handling probabilistic moving objects, etc ☺.

Fallacy 2: Inconsistent Stand (朝令夕改)

Example 1:

Year 1: Published a paper that claim to speed up frequent pattern mining by not generating 2^100 candidates. The experiments however did not involve a pattern with 100 items.

Year 2: Published a paper that could potentially generate 2^100 candidates for frequent pattern mining

Example 2:

Year 1-3: Published papers that claim horizontal representation (row format) is better than vertical representation (column format) for mining frequent patterns

Year 4: Published a paper that use inverted list(column format) for mining frequent patterns in gene expression data

Fallacy 3: Empty promises

Example:

Write a paper A on query processing of probabilistic data assuming data instances are **independent** and claiming that data instances that are **correlated/anti-correlated** can be easily handled.

Write many papers which are extension of paper A (including a journal version) but none on handling data dependency at all!

Fallacy 4: Taking things out of context

Example:

Subspace clustering was invented for handling high dimensional data (10-100 dimensions) because (i) there might not be clusters in higher dimension (ii) users need to understand the relevant dimensions because there are so many dimensions (iii) number of attribute combinations is very high and a search is needed to find the right combination

We now have lots of work on subspace outliers detection, subspace neighbors and subspace skylines that work only for less than 8 dimensions and with specified subspace

Fallacy 5: Making things unduly complicated

Use lots of complicated algorithms and formulas for problems when simple solutions and explanation exist.

Impact in real life become limited.

How can system implementation help?

- **In general, these fallacies can be avoided by simply observing good research practice. System implementation however help a lot by:**
  - Putting idea into practice bringing in all factors that will affect system performance
  - Need to make careful and consistent choice since idea implemented take a lot of effort to roll back
  - Can’t make empty promise since problems must be solved in order for system to work
  - Can’t take things out of context in a real situation
  - Have to make things simple but effective in order not to build a very “fat” system

What systems to build?

- **System with a central thesis**
  - Example: TIMBER(Native XML database)
- **System with a particular architecture**
  - Example: Bestpeer
- **System on emerging applications**
  - Example: Trio, MystiQ(probabilistic database)

Pure Research
闭门造车

System development for the research community should be somewhere between these two extremes

Well studied Industrial System
索然无味

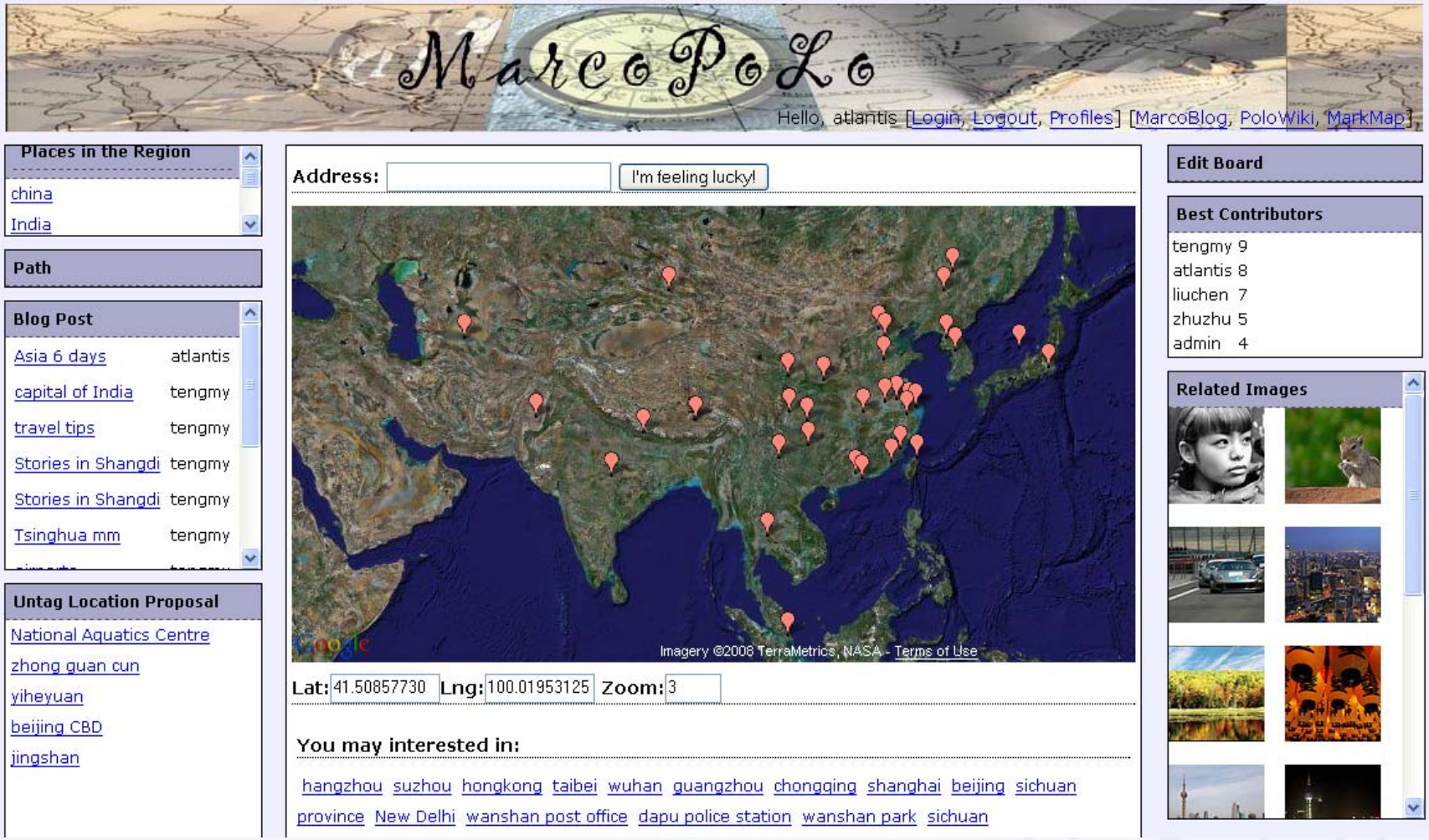
What about young faculties?

- **At least prepare for it. Meanwhile, learn and work with the senior faculties.**
- **Very strong data system research in NUS(Lucky me)**
- **Bestpeer(www.bestpeer.com)**
  - 8 years, 4 graduated phds, a few post-docs, 2 more phd and other students to build –
  - Presently in version 2
  - it has generated 6 SIGMOD, 1 VLDB, 4 ICDE papers, and 1000+ citations
  - it has been spun-off
  - Involved Fudan, Tsinghua and Renmin U. in research that revolve around the system as well
- **Working now on the MarcoPolo project lead by Prof. Beng Chin Ooi**

MarcoPolo: A MashUp Travellog

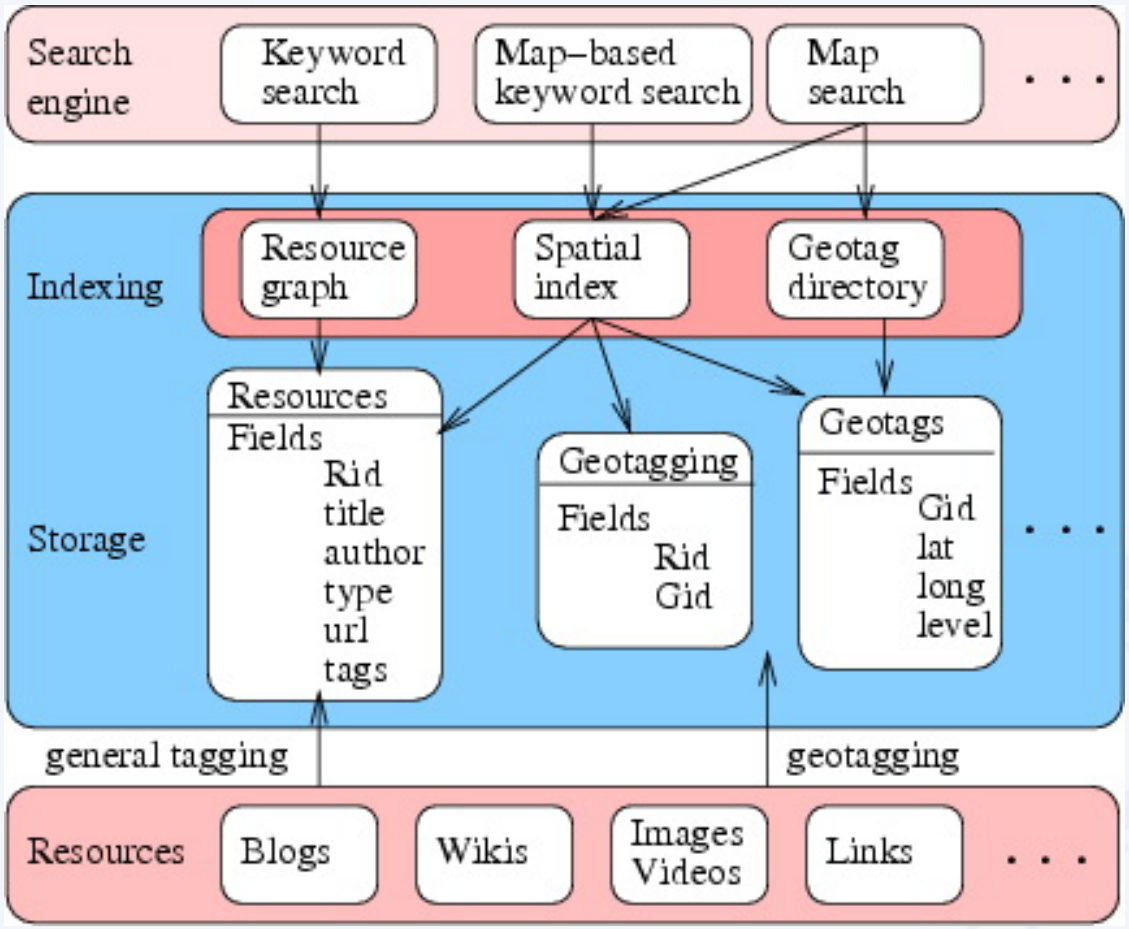
- **The plane (virtual overlay) is the map of geo-tags – personal dataspace**
- **Users tag, browse, search travel-related information through the map.**
- **Text format of common geo-tags (given by users) are mapped to geo-tags (with Lat. & Long.) of MarcoPolo:**
  - Users contribute the hierarchical geo-tags in maps.
  - Automatically mark information of objects (wikis, blogs, and multimedia objects) to the map through geo-tags.
- **URL: www.langG.com.cn**

Map Region Aggregates



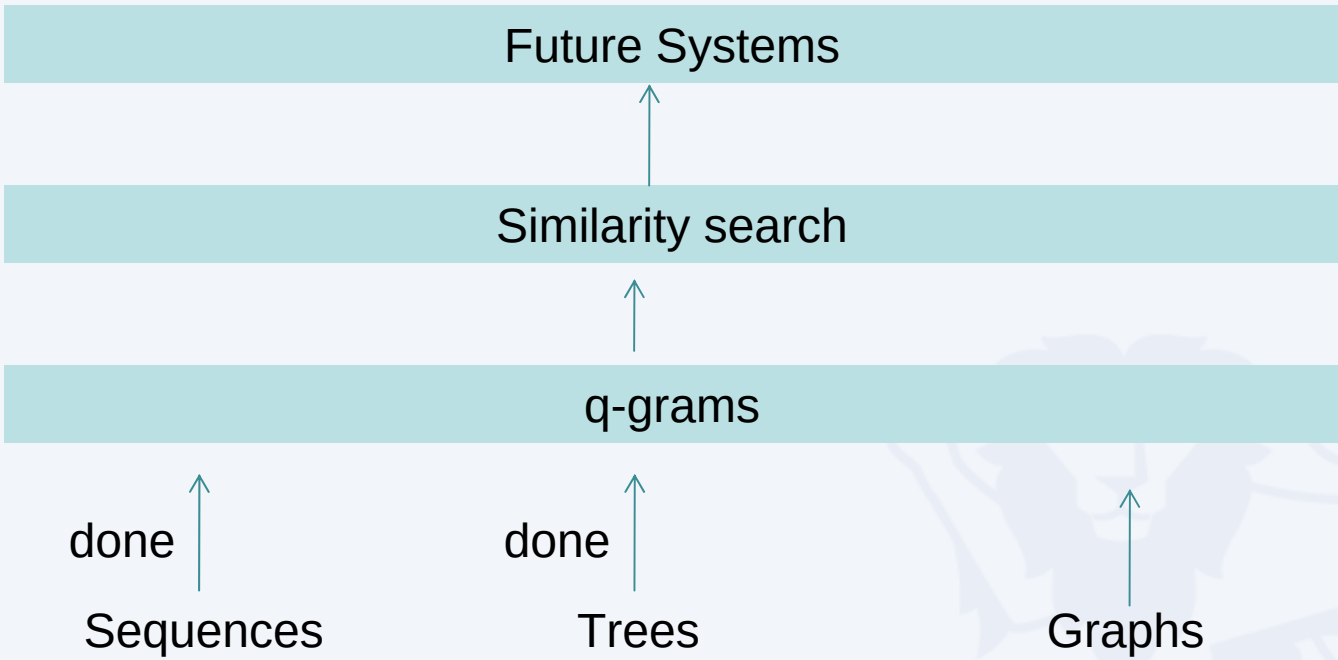
Focus on Specific Geo-tag

MarcoPolo Architecture



Prepare the fundamentals

Example:



Conclusion and Acknowledgement

- **System development in database/internet research is very important in bridging the gap between research and industry. It helps to avoid a lot of fallacies in research.**
- **www.comp.nus.edu.sg/~atung/publication/system.ppt**

This panel proposal is in many ways inspired by the constant effort of our colleague Beng Chin Ooi in persuading us build real, deployable system. The example on the problem of concurrency control in moving object indexes is derived from his paper on B^x-tree.

C. Jensen, D. Lin, B.C.Ooi: Query and Update Efficient B+-Tree Based Indexing of Moving Objects. Int'l Conference on Very Large Data Bases (VLDB), 768-779, Toronto, 2004.