Kyu-Young Whang Speaks Out

on Academia and Startups in Korea, Probabilistic Counting, Main-memory Query Optimization, How to Avoid Being a Hostage of Pressure Publishing, and More

by Marianne Winslett

http://dblab.kaist.ac.kr/Prof/

Welcome to this installment of ACM SIGMOD Record's series of interviews with distinguished members of the database community. I'm Marianne Winslett, and today we are at the ICDE 2007 Conference in Istanbul. I have here with me Kyu-Young Whang, who is a professor of computer science at Korea Advanced Institute of Science and Technology, and the director of the Advanced Information Technology Research Center. Before joining KAIST, Kyu-Young was a member of technical staff at IBM TJ Watson Research Center. His research interests are very broad, encompassing many different kinds of database systems as well as data mining and data streaming. He is an editor-in-chief of the VLDB Journal, a former member of the VLDB Endowment, and an IEEE Fellow. Kyu-Young's PhD is from Stanford. So, Kyu-Young, welcome!

What led you to move from a research lab to academia?

I truly enjoyed working at IBM and working with the brilliant people there. But I had a sense of duty to make a contribution to the Asia-Pacific database community and to bring up the level of database research in Korea. Working with the brilliant students at KAIST was a very important motivation.

Recently we have started to see an exodus of researchers from academia to industrial labs, motivated by the urge to get access to real-world data. If you were a young researcher at IBM today, would you still leave for academia?

I think I certainly would. Actually, my motivation to move to academia was to be able to teach brilliant students, to have them and to see them make a contribution, as well as doing the research myself and making contributions myself. In fact, I wanted my students to be as competitive and as capable as anyone educated in the top universities in the world. I think some of them have excellent potential to achieve that objective.

From your time at Stanford, you are familiar with academic life in the US. How is life as an academic database researcher in Korea different from in the US?

I would like to mention two aspects regarding this question. One is the availability of information, and its timeliness. The second is the infrastructure supporting research activity. In earlier days, say 20 years ago, the availability of information was a problem in Korea, and probably in other nations in the Asia-Pacific region, lagging by months and even years. If a conference were held in the US, it would take 3 months to get proceedings, and then it would take another 6 months for it to be disseminated. But now, the technology makes information dissemination almost immediate, and this problem does not exist anymore.

Nevertheless, the infrastructure is still a problem. Basically, people in Korea are very busy. The main reasons are that we have less adequate infrastructure, including administrative support, technical support, and many other things. Ever-evolving evaluation systems and the rapid change in the socio-economic systems, etc., give much less time for the researchers to concentrate on research. We need a more stable research environment for those researchers, and the students as well, to make bigger and longer term research contributions.

What do you view as the major contributions and impacts of your research?

I would like to mention three equally important pieces. The first one is to pioneer the new notion of probabilistic counting. Probabilistic counting statistically counts the number of unique values that are in an attribute, or in a multi-set, in linear time, with an arbitrary specified error bound. Being able to control the error bound is very important. This was a surprising result, because it was a common belief that counting unique values required sorting, which is a much more costly operation. Probabilistic counting is now being used in IBM's DB2, and is actively used in other areas, such as approximate query answering, data mining, sampling, and data streaming.

I initiated the work with Morton Astrahan and Mario Schkolnick in 1981, at IBM Almaden Research Center, which was then called San Jose Research Lab. We developed three algorithms and many other people contributed to this project, including Nigel Martin, Mark Wegman, and Philip Flajolet, who was a visiting scientist at that time, and who made a very complicated analysis of one of those algorithms. This work was reported in *Information Systems* in 1987 and later in *ACM TODS* in 1990, and is highly cited these days.

The second contribution is pioneering work on the query optimization model of main memory resident relational DBMS. Office-By-Example, called OBE, is the first full blown implementation of main memory resident relational DBMS. It was developed at IBM Watson, around 1983-85, in a project led by Moshe Zloof. I implemented the query optimizer, and the key problem was that we needed a new cost model for the main memory resident data base. Traditionally, we counted the number of disk I/Os and something like that, in relation with disk based systems, but in main memory systems you don't have disk I/Os, theoretically. Counting CPU cycles for the new cost model would be next to impossible, or at least impractical. So, we needed a model for how to count those cycles and to evaluate the cost model.

I proposed at the time to use the system's bottlenecks as the basis of the cost model. In other words, we do system profiling of the executions and identify the bottlenecks, i.e., the pieces of codes that consume most of the time. We count the number of bottlenecks---there are several types of bottlenecks---and assign them different weights; and that replaces counting the number

of disk I/Os in disk resident database systems. We also saw that these bottlenecks in fact correspond to important operations such as predicate evaluation, tuple retrieval, and etc.

This was a fairly new idea at the time, and then, it also influenced some cost models of commercial main memory DBMS, such as TimesTen, the company founded by Marie-Anne Neimat from HP. This work was reported in ACM TODS in 1990.

The third and probably the most important contribution is the development of Odysseus DBMS, which proposed tight coupling of the database management system with information retrieval. Odysseus had a full blown implementation of tight coupling of DB and IR in 1997 and has also made a practical and industrial impact. Patents have been granted in the US and in Korea. Odysseus consists of 450,000 lines of C and C++ high precision code of commercial quality. We demonstrated this system at IEEE ICDE in 2005 in Tokyo, winning the best demonstration award.

Odysseus has made a significant practical and industrial impact by being used as a main search engine, called Naver---this is a fabricated word, meaning "navigator"---of NHN Co. at its start-up phase (1997-2000) and by helping it to grow to be a six-billion-dollar company in a very short period of time. The NHN Company is the number one internet portal site in Korea, and is more popular there than Google. We know that DB/IR integration is becoming a new area of active research.

As you mentioned, the most popular search engine in Korea is Naver (www.naver.com), which has more than twice as many hits and users as Google. The original version of Naver used Odysseus as its internal engine. Since there is a Korean version of Google now, why is Naver still so much more popular?

I think it is their business model. Naver's business model is better suited to the Korean language environment and they collect more Korean language web pages. They also make active use of user-created content, from blogs and communities, and other things. Naver has a lot of loyal users in Korea who produce lots of content through these facilities. And the third reason is that Naver has the affiliation of many important content providers, including newspaper agencies, and they collect all those newspapers and provide those services from the Naver site. People go into Naver and read the newpapers through Naver, rather than going to the newspapers' own sites. This business model and contents are geared to the Korean community and culture, and that makes Naver strong.

Why don't you have a startup that sells Odysseus?

As I mentioned, I think Odysseus has already made a significant impact on industry through Naver. Whether to make that as a company or not is a different question. I am trying very hard to transfer this technology to other companies, including big companies like LG Electronics, which is very well known world wide and is a Korean company. So, I think the role of research is probably to develop the technology and transfer it; starting a company is a secondary concern.

It is not as easy in Korea to start a company, compared to the US. There are many considerations that have to be dealt with. There isn't sufficient infrastructure to make a new startup easy to create and run, so a lot of your time has to be spent in the new startup. Since you are already busy as a professor, how could you do a startup company in addition?

I have heard that you hold doctoral seminars on Saturday evenings when you are in Korea, starting at 8 PM or so and lasting until 2 AM or later. I have also heard that you ask great

questions during these seminars. For our readers who would like to become better question-askers, do you have any tips on how to think of good questions during a seminar?

First I would like to mention that Saturdays have been half working days in Korea until a few years ago, when the government instituted a policy to make it a holiday. We frequently worked late at night so that we can have many hours of quiet time, and Saturdays were good candidates to make this objective, so we frequently worked on Saturdays. Students often came up with potentially very good ideas. What is important is how to make this potentially important idea into a *really* important idea. I usually advise the students to keep two things in mind. One is to substantiate the idea by testing it against known expertise, mostly those accumulated in our lab. This process is like using touch stones to check if a piece of precious metal is genuine gold or not.

Do you mean testing against software that you have in the lab, or are you talking about the other people in the lab?

I'm not exactly talking about software, but when you have some idea, you have to have a model to check the idea against. The model comes from your own expertise, or collectively from our lab. So, with a new idea, you evaluate it piece by piece against what you already know. It is not enough to come up with a new idea out of the blue; it has to be tested out, by using previous experience.

The second test of a new idea is to check its completeness, whether it covers every case, or there is something missing. Completeness is a very effective tool to find the holes in your idea. Oftentimes, people think mainly about soundness, whether the idea makes sense or not, and overlook the completeness aspect. If you check the completeness, you can easily find the holes in a new idea, and those holes make excellent topics for questions.

You are the current president of the Korea Information Science Society, which is the Korean version of ACM. What projects have you planned for the society?

First, it is quite an honor for me to be elected the president of the Korea Information Science Society. It's called KISS, and we pronounce it "kiss" with a long I, to distinguish it from "kissing". KISS is the largest and oldest computer and information related professional society in Korea. KISS is very important for domestic researchers and practitioners, because it provides opportunities for them to participate in a variety of academic and industry oriented activities. We note that opportunities for participating in international academic activities are limited by two aspects. One, obviously, research opportunities are not available to all the domestic researchers and practitioners. And two, there are many activities geared to only domestic problems. What concerns us, what is important, and what we wish to achieve are different because Korea, or any other country, is at a different level of development from other countries.

During my tenure as the president, I am trying to reinforce several aspects: strengthening the domestic journals, which are published in the Korean language so that many domestic participants can include their research contributions; initiate a new English-language journal; importing the Fellow system for recognition of contributions of our members; strengthening computer science education from elementary school to high school (which is very important---the colleges and universities are suffering from inadequate education in the computer discipline at those earlier stages of education); and enhancing programming skills in freshman and sophomore classes in college education. There are lots of other important issues that we have to address, too. The KISS has to be the place and the vehicle where our members can address their concerns and achieve their objectives. I am committed to helping our members from this perspective.

What do you think are the ingredients of good teaching or training of systems-oriented graduate students?

Teaching may involve many things, but I would like to focus on working with PhD and Master's students. I think that computer science in general, and especially databases, are quite prototype driven. Although theoretically interesting results are equally important, we very much emphasize their practical impacts. So, I firmly believe in the importance of systems oriented research, and that students must have a strong background in systems and programming. Of course, creativity is of prime importance in PhD training, but a strong background in systems and programming will give students more freedom in choosing their research, and also add practical value to what they invent.

Our Odysseus project provided excellent opportunities to my students in that respect. We have produced many highly skilled system programmers, internationally competitive. In fact, two of my former students worked at IBM Almaden Lab as post-docs recently. One worked with Michael Carey on making the DB2 DBMS object-relational. He made very important contributions there. Another one worked with Guy Lohman and Volker Markl on advanced techniques in query optimization. I heard that their work is almost ready to be made into a product, which is a significant contribution. So, my students and their systems-oriented skills in research capability have been very highly evaluated at IBM Almaden, and I am very proud of them.

Your son is now a member of the Stanford database group. What do you think about his following a similar career to yours?

I am very happy and proud that Steven has been admitted to the Stanford computer science PhD program, which is truly excellent. It is also Steven's big dream to study at Stanford, where very tough and intriguing challenges are waiting for him, and his dream has come true. Actually, he was born at Stanford!

What to study and what career to make is completely up to Steven. In fact, many years ago, I recommended him to be a medical doctor. Apparently he did not like it; instead, he likes computers. So it is completely justified that he become a computer scientist, and even though I am in that profession, I don't mind his becoming a computer scientist, where he can experience many challenges and creativity, and can make practical impacts. Further, I don't hesitate to tell him that the database area has been and will continue to be the area of prime importance because, in the information age, how to deal with and to make the best use of vast amounts of information and data is the key problem to solve. And I don't think it will change for many years to come.

Do you have any words of advice for fledgling or midcareer database researchers or practitioners?

There are many things to advise, but one thing I would like to emphasize is that I think they need a long term vision, an objective or direction. Of course, the vision can change, and sometimes it must change. But still you need to have a direction or vision, as otherwise, you are very susceptible to fast changing fashions of research and practice. You will easily go astray. Many people, especially in academia, tend to become hostages of pressure publishing, and they work on whatever can be made a paper. I don't think this is a good approach. So, you need a vision, a concrete direction, and you should stick to that. And then, eventually, many variations including

those fashions will melt into your own vision, rather than the other way around. That way, your vision will also be reinforced.

If you magically had enough extra time to do one additional thing at work that you are not doing now, what would it be?

I would do exercise, and get myself in shape. This is one of the things I am not doing very well recently, and I certainly will correct it in the coming years. My favorite exercise is to take little walks over the hill close to my home. I probably could do that more often.

If you could change one thing about yourself as a computer science researcher, what would it be?

I think I have many things that I must change. If I name one, it is that people like you and me tend to be addicted to work and achievements. We call them *workaholics*. I don't think this is a problem of only the computer scientists; many people in the present days are workaholics and tend to lose appreciation of their personal lives. Computers have been invented to help people in this respect, but oftentimes they work to the contrary. As we have more technology and computers and other things, you tend to work more, to achieve more. So, I really should change myself in this respect in the coming years. I should spend more time with the family and my parents, who are getting fairly old these days, and also spend time with friends.

Thank you very much for talking with us today.

Thank you.