HashCache is hot

Princeton University invention makes Internet access faster

PATRICK JONAS

EVERY year, the Massachusetts Institute of Technology's (MIT) Technology Review magazine comes up with its annual list of 10 top emerging technologies.

This year, there are Indian names in the list.

One of the inventions that made the list is the result of work done by a team of researchers led by Prof Vivek Pai at Princeton University. It is called HashCache and has drawn recognition as a revolutionary way to expand Internet access around the world.

"HashCache is a system that stores Web content to disk, and reuses it when possible, so that your Internet access appears to be much faster," Prof Pai told tabla!.

It has the potential to expand Internet use in developing regions around the world, taking advantage of cheaper components like large disks.

To understand how HashCache works, see report below.

There is another Indian face in Prof Pai's team. Mr Anirudh Badam is a graduate student at Princeton who leads the project, working closely with Prof Pai. The other members of the team are Prof Kyoung Soo Park, now at the University of Pittsburgh, Prof Larry Peterson, the department chair at Princeton, and research scientist Marc Fiuczynski who helps arrange and coordinate the test deployments in various parts of the world.

The new system is currently being tested at the Kokrobitey Institute in Ghana and Obafemi Awolowo University in Nigeria.

Prof Pai was born in Mumbai and moved to the United States with his parents in 1970, when he was just two years old. He married an Indian and often visits India with her.

"Part of the motivation for looking at

Cache them if you can... (from left) Mr Badam, Prof Peterson, Prof Pai and Mr Fiuczynski. PHOTO: PRINCETON UNIVERSITY

technologies for the developing world was my experience when I was last in India, trying to get my in-laws' computer working again," said Prof Pai.

Added Mr Badam: "I was very excited and enthusiastic right from the day Vivek first proposed that we find a solution to this challenging problem. The result of our efforts in this direction is HashCache and we are all very excited about it."

The Hyderabad native is a third-year PhD student at Princeton. He moved to the US in 2006 after doing his degree in computer science from the Indian Institute of Technology in Chennai.

Mr Badam has been interested in high performance servers since his days at the IIT. "Being from a developing country, I have had first hand experience of using slow and intermittent Internet connections. Never until I came to the US did I realise what good Internet actually is. This was also a motivating reason behind me being very enthused to solve this problem,"

❗ *Once I got the opportunity to work with Prof Pai, there was no looking back.*

— Mr Anirudh Badam

Mr Badam told tabla!.

He added that Prof Pai was previously involved in developing the world's fastest Web server (Flash Web Server) and later in the development of the world's fastest Web proxy cache (iMimic Web Proxy Cache). "So, once I got the opportunity to work with him, there was no looking back," he said.

The Technology Review presented the list in New Delhi on March 2 at the inaugural EmTech India conference which Prof Pai unfortunately could not attend. "We recently had a new baby, and I'm in the middle of the school semester, so my travel is a bit constrained at the moment," he said.

What about Mr Badam?

He is busy working on new technologies at Princeton. When tabla! wrote to him at 5pm (Singapore time), he replied immediately. He was in the lab, at 4am in Princeton!

➥ patrickj@sph.com.sg

How you can cache in...

WE ASKED Prof Vivek Pai to explain it simple terms how HashCache works. This is his explanation:

"Let's say that you want to visit a Web page, like http://www.cs.princeton.edu/~vivek

To your browser, that page actually looks like many different pieces, which have names like http://www.cs.princeton.edu/~vivek/index.html http://www.cs.princeton.edu/~vivek/nsg.css

Each of those pieces is called an object. So, to build that page, your browser is sending multiple requests, one for each object. If another colleague at your paper also visited the

page, his browser would also send that same sequence of requests. All that traffic would flow from the US to Singapore twice.

What a cache does is store each object as it comes in, and then checks to see whether a request can be satisfied from the object it has stored, instead of being re-fetched all the way from the US.

The way that caches typically operate is that they have to store each object somewhere on disk. Since it's hard to work with long names like http://www.cs.princeton.edu/~vivek-/index.html the cache reduces it to a number, say 125371242. This process is called hashing. However, it then has to keep in RAM some information that

says that this particular file is located at a given location on the disk. That portion is called the index. As the size of your disk grows, the size of this index also grows.

What HashCache does is to get rid of this index. Instead, once it calculates the hash value for a file, it uses that same number to determine what location on the disk it should use to store that file.

There are some tricks involved and some complications, but that's the basic idea. By eliminating this mapping, HashCache does not need to keep a lot of information in RAM. Since it's easier to buy lots of disks rather than lots of RAM, HashCache allows you to build caches using much cheaper systems.