

## ▶ C I M I L L I O N P R O B L E M

**In the 21st century the problem has been upped to the C I Million problem. (Think IoT)**

For the application level, Dylan Schiemann recommends using Comet, “a catchall term describing the collection of techniques, protocols, and implementations that address making low-latency data transit to the browser both viable and scalable.” On p109 of the O’Reilly published book “Even Faster Websites”, Mr. Schiemann states that “Comet can easily provide more simultaneous connections than a traditional web server, and faster data transit between the client and the server.”

A similar approach was used to solve the C500K problem by Urban Airship.

## ▶ R E F E R E N C E S

Even Faster Websites, Steve Souders, 2009, O’Reilly Media, [http://www.amazon.ca/Even-Faster-Web-Sites-Performance/dp/0596522304/ref=sr\\_1\\_1?ie=UTF8&qid=1425652054&sr=8-1&keywords=Even+Faster+Web+sites](http://www.amazon.ca/Even-Faster-Web-Sites-Performance/dp/0596522304/ref=sr_1_1?ie=UTF8&qid=1425652054&sr=8-1&keywords=Even+Faster+Web+sites)

<http://www.metabrew.com/article/a-million-user-comet-application-with-mochiweb-part-3>

<http://urbanairship.com/blog/2010/08/24/c500k-in-action-at-urban-airship/>

<http://blog.arungupta.me/rest-vs-websocket-comparison-benchmarks/>



D E R R I C K L A U

## ▶ The C I 0 K P r o b l e m

### ▶ H I S T O R Y

<http://www.kegel.com/c10k.html>

[http://en.wikipedia.org/wiki/C10k\\_problem](http://en.wikipedia.org/wiki/C10k_problem)

In 1999 Kegel argued we should be able to support 10000 simultaneous users on a single web server of reasonable hardware, on the assumption that the software was architected properly. He then listed out guidelines for architecting and coding such software.

### ▶ C I 0 K S O L V E D

**The majority of Kegel’s work was focused on the architecture of the web server, and all modern web servers now implement his strategies.**

Twitter: @Derrick\_Lau  
E-mail: derrick.lau@live.ca