

Threads vs Events

Background/Discussion

John Osterhout

- Professor of Computer Science at Stanford
 - Teaches Operating Systems and Software Design
 - Web page: <https://web.stanford.edu/~ouster/cgi-bin/home.php>
- Inventor of Tcl/Tk and Magic VLSI CAD
- Worked in Silicon Valley 1994-2008
 - Sun Microsystems : 1994 - 1998



Why Threads are a Bad Idea

- Invited talk at the 1996 USENIX conference (Jan. 25)
- Background:
 - Threads were a new and popular idea in 1996
 - Threads were being recommended for many applications, especially GUI applications
 - The existing technology for GUI was event driven processing
- No paper written

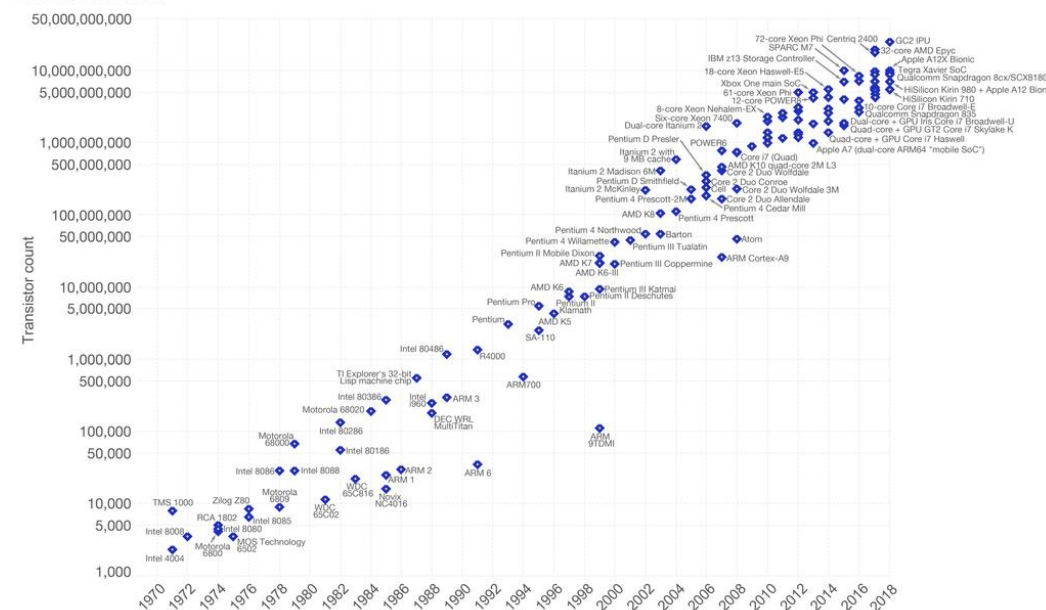
Conceptual View of the Event Loop

```
while(1) {  
    if (eventQue.hasData()) {  
        newEvent=eventQue.getFirst();  
        newEvent.run();  
        if (eventRequestedExit) break;  
    }  
    if (newEvent=getEvent()) eventQue.addLast(newEvent);  
}
```

Follow-Up Discussion

- 1996 was over a generation ago! What has changed since then?
- What is the impact of Moore's Law?
 - Does Moore's Law still hold?
- Why is concurrency so hard to debug?
- Are Wizards still required to code with threads?
- See also:
[Why Events are a Bad Idea](#)

Moore's Law – The number of transistors on integrated circuit chips (1971-2018)
Moore's law describes the empirical regularity that the number of transistors on integrated circuits doubles approximately every two years. This advancement is important as other aspects of technological progress – such as processing speed or the price of electronic products – are linked to Moore's law.



Data source: Wikipedia (https://en.wikipedia.org/wiki/Transistor_count)
The data visualization is available at [OurWorldinData.org](https://ourworldindata.org). There you find more visualizations and research on this topic.

Licensed under [CC-BY-SA](#) by the author Max Rose