Moore's Law

[Moore's Law](https://www.techopedia.com/definition/2369/moores-law) is probably the best-known "law" in the computer world. It's named for Intel founder Gordon Moore. In a 1965 paper, he noticed that the the number of transistors on an [integrated circuit](https://www.techopedia.com/definition/2366/integrated-circuit-ic) doubled about every two years. This meant that the [chips](https://www.techopedia.com/definition/2169/chip) had more functionality than before for the same price. In other words, as time went on, the chips did more for less.

You've probably seen this in your own life. When you buy a new computer, it's generally faster than the last one you bought - and costs less as well.

Moore's Law is not only observable in [microprocessors](https://www.techopedia.com/definition/2874/microprocessor), but also in [memory](https://www.techopedia.com/definition/2787/memory) and [storage](https://www.techopedia.com/definition/1115/storage) space. It seems there is no limit, but chip makers can squeeze only so many circuits on those silicon wafers. On the other hand, [quantum computers](https://www.techopedia.com/definition/5735/quantum-computer) may offer a solution, though they're still a long way off from mainstream use.

Metcalfe's Law

As [devices](https://www.techopedia.com/definition/2185/device) get cheaper, more people will buy them. And the more people buy them, the more valuable the network of devices becomes.

[Metcalfe's Law](https://www.techopedia.com/definition/29066/metcalfes-law) is attributed to Bob Metcalfe, one of the originators of the [Ethernet](https://www.techopedia.com/definition/5280/ethernet) networking protocol. He proposed that if there are N users of a telecommunications network, the value of the network is N2. Each new person that joins the phone network adds to the number of possible connections, barring things like language differences. The same goes for [social networking sites](https://www.techopedia.com/definition/4956/social-networking-site-sns). If you're a [Facebook](https://www.techopedia.com/definition/4941/facebook) member, you probably joined because all of your friends are on Facebook.

Reed's Law

Following Metcalfe's Law, Reed's Law, developed by computer scientist David P. Reed, says that the utility of large networks can scale exponentially with the size of the network. In other words, the number of possible subgroups of a network is 2N - N - 1, where N is the number of people using a network. This law is the reason that things can "go [viral](https://www.techopedia.com/definition/5577/viral)" in [social media](https://www.techopedia.com/definition/4837/social-media). It also explains the popularity of social networking services. As more people join them, the more they can connect with others who share their interests, increasing the value of the network.

As Reed himself [explains](http://www.reed.com/dpr/locus/gfn/reedslaw.html):

We can see this scale-driven value shift in the history of the Internet. The earliest usage of the Internet was dominated by its role as a terminal network, allowing many terminals to selectively access a small number of costly timesharing hosts. As the Internet grew, much more of the usage and value of the Internet became focused on pairwise exchanges of email messages, files, etc. following Metcalfe's Law. And as the Internet started to take off in the early 90's, traffic started to be dominated by 'newsgroups' (Internet discussion groups), user created mailing lists, special interest websites, etc., following the exponential GFN law. Though the previously dominant functions did not lose value or decline as the scale of the Internet grew, the value and usage of services that scaled by newly dominant scaling laws grew faster. Thus many kinds of transactions and collaboration that had been conducted outside the Internet became absorbed into the growth of the Internet's functions, and these become the new competitive playing field.

Beckstrom's Law

Rod Beckstrom has a more complex model of network utility. Beckstrom's Law defines the value of a network as the benefits minus the costs summed over the total number of members of a network.

Taylor Buley, writing for Forbes, gives a good concrete [example:](https://www.forbes.com/2009/07/31/facebook-bill-gates-technology-security-defcon.html)

Here’s an example: Say you purchase $100 worth of stuff from Amazon each month over the course of a year. You could probably buy that stuff offline for about the same price, but you might pay extra for gas to drive to the store and an opportunity cost for your time. If the brick-and-mortar commerce cost amounted to $50 a month on top of the $100 you spent on books, then the value of Amazon’s network to you would be $600 a year. Subtract from that the cost of connecting to Amazon’s network, perhaps $40 a month for an Internet connection and computer hardware, and you have a value of something like $120.

Brooks' Law

Going from the world of networks to the world of [software engineering](https://www.techopedia.com/definition/13296/software-engineering), we have Fred Brooks, author of the classic book *The Mythical Man-Month*. [Brooks' Law](https://www.techopedia.com/definition/18085/brooks-law), which comes from the book, should serve as a warning to any [product manager](https://www.techopedia.com/definition/28370/product-manager). "Adding manpower to a late project makes it later."

While managers may assume that more people working on a project could be better, they fail to account for the costs of coordination and bringing new developers up to speed. The industry is littered with the failed projects of those who failed to heed this law.

Hofstadter's Law

Douglas Hofstadter popularized the concept of recursion in his classic Pulitzer Prize winning book *Gödel, Escher, Bach: An Eternal Golden Braid*. Although not strictly about computing, the book is influential within the computing industry. Hofstadter came up with his own humorous law, known as [Hosftadter's Law](https://en.wikipedia.org/wiki/Hofstadter%27s_law): "It always takes longer than you expect, even when you take into account Hofstadter's Law."

This law uses recursion, referring to Hofstadter's Law within itself, reminding us that no matter what we do, sometimes things go wrong anyway.

Even in a heavily abstract field like computing, there are some observable "laws" - just like in mathematics. By closely studying these laws, we can build on our understand of computing and build even better things by expanding innovation.