# The Day the Internet Stood Still

A peculiar thing happened a few weeks ago. On March 22nd, thousands of [JavaScript developers were faced with broken builds and failed installations](http://www.theregister.co.uk/2016/03/23/npm_left_pad_chaos/) due to a missing piece of code, 11 lines in length. Much like the events in the *Day the Earth Stood Still*, a single superior force brought the much larger but far more primitive press of humanity to a grinding stop. But unlike that iconic movie, the motive in the internet crisis wasn’t moral but rather economic (although there is certainly a moral aspect to this story as well – as there is in all things economic).

The timeline of events is disclosed in detail elsewhere. The key features for the sake of this argument are simply these. There exists a common JavaScript code repository called NPM which dubs itself as the place to “Build amazing things” and describes itself as:

<npm is the package manager for JavaScript. Find, share, and reuse packages of code from hundreds of thousands of developers — and assemble them in powerful new ways.>

One such developer, by the name of Azer Koçulu, had provided to all of humanity, the 250 JavaScript Modules. Of these, the reader must focus on only two of them. The first was named kik, which is also the common short form name of [Kik Messenger](https://en.wikipedia.org/wiki/Kik_Messenger), a messaging app for smartphones. The second, called left-pad, was the 11-line piece of code that brought much of the internet to its knees and opened lots of new horizons in the ownership of intellectual property.

As might be predicted by the common name, a clash developed between Azer and Kik’s corporate office. The latter requested that Koçulu surrender the name of the module since they legally owned the trademark. When he refused their less than polite request, they went to NPM to force the issue and, when NPM management complied, Koçulu unpublished all his modules. The resulting elimination of “left-pad” broke the systems that depended on it, precipitated NPM’s unprecedented step of restoring “left-pad” (so-called un-unpublising), and launched a controversy that is likely to become a watershed event discussed for decades to come.

Now I’m not going to weigh in on the various legal points that have been raised, such as did Kik have the right to the name, did NPM have the right to give it away or to un-unpublish the “left-pad”, or did Koçulu have the right to unpublish the code in the first place. As interesting as these questions are there is a much more interesting question. Was “left-pad” a public good?

To appreciate this question one must first understand how economists place good into the four categories of private, club, common pool resources, and public. Each good is judged in terms of two attributes: excludability and rivalry.

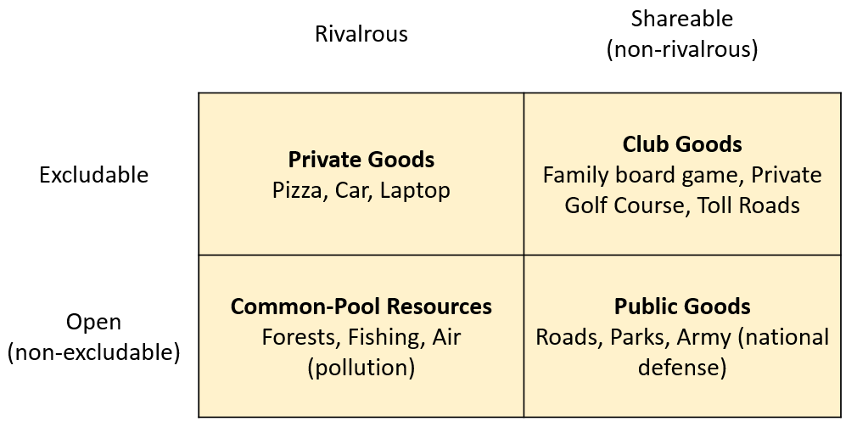
A good is termed excludable if a person or entity possesses legal rights that enable them to prevent others from using it. A good is non-excludable if no one either possess such a right or if the right is effective non-enforceable. The term open is synonymous with non-excludable in what follows.

A good is rivalrous if the use of the good by one entity precludes it use by all others. A good is non-rivalrous if it can be used by many entities without harm being done to any of them. The term shareable is synonymous with non-rivalrous in what follows. Note that only intangible things likes ideas and concepts can be truly shareable but that in many cases some goods are so much closer to shareable than not that the idealization is useful.

The four possible combinations of excludable/open with rivalrous/shareable give the four categories of goods:

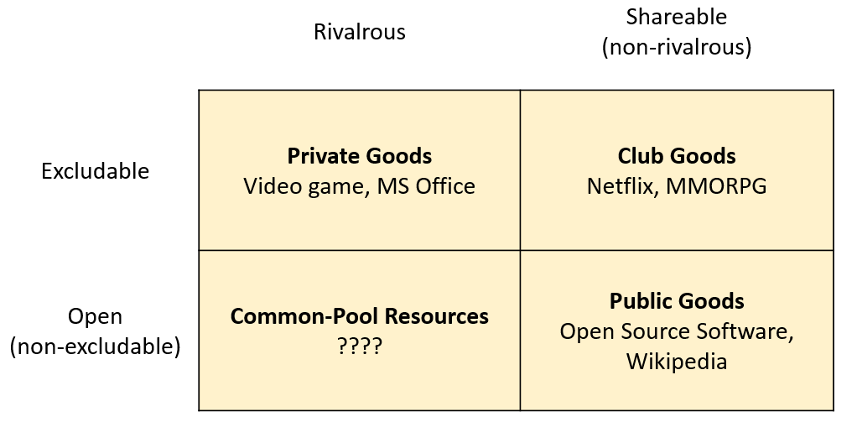
* Private property - excludable and rivalrous
* Club good – excludable and shareable
* Common pool resource – open and rivalrous
* Public good – open and shareable

These definitions are abstract and difficult to think about so a common tool is to construct a 2x2 table with instances of each type. Common tangible goods can be placed in such a table and on such version is



The next step is to create an analogous table for digital goods and then, using the resulting categorization, conclude in which of these cells the innocuous but vital “left-pad” module should live.

Adapting the 2x2 table to cyberspace is a bit more challenging than tangible goods precisely because of the blurred lines that exist in the digital world between ownership and right-to-use. For example, when one buys a videogame, one is really buying the right-to-use the game on a game console and not the game itself. Unlike Monopoly, where the owner really owns the matter/hardware that goes into the game and can transform it as he sees fit, the owner of Halo really owns the ability to interact with that particular copy of the game he purchased. The situation is further complicated by the fact that there is a fundamental difference between the embodiment of the game (the pit and blanks on the DVD, the DVD itself, the game console, etc.) and the code that makes up the game.

Nonetheless, after some thought, it is possible to come up with good examples in three of the four categories; the common pool resource being the only one that seems to lack a digital analog. One such instance is 

The only step that remains is to determine where in this table “left-pad” finds a home. The natural first reaction is that “left-pad” is a public good and [Nadia Eghbal](https://medium.com/@nayafia/there-is-no-my-in-open-source-c3e5555390fa#.af4sj6urr) certainly thinks so. But this question isn’t really well-defined enough to answer.

Certainly, the code concept itself, taken as an abstract entity, is a public good. Koçulu neither claimed copy-right nor did he regulate (exclude) use. But the embodiment that he maintained on NPM was more like a club good, where for much of its life the club was everyone. Then after the debacle with Kik, Koçulu simply redefined the club to be no one.

As time progresses and society, in general, and economists, in particular, have a chance to analyze the fallout from this event and others like it, I suspect that whole new modes of thought will have to be developed about who owns what in digital realm.

[Klaatu barada nikto!](https://en.wikipedia.org/wiki/Klaatu_barada_nikto)

# Physical Limits of Computing

Do It Yourself Colorist

Walk into any Barnes and Noble bookstore and you’ll find, prominently displayed on a table near the front of the store, a host of adult coloring books and supplies of brush markers. It seems that adult coloring is a new pastime. I’ve tried it and it is a lot of fun. To be clear, I never gave up on coloring the old way (crayons and kids coloring books), so maybe I am bias but whatever.

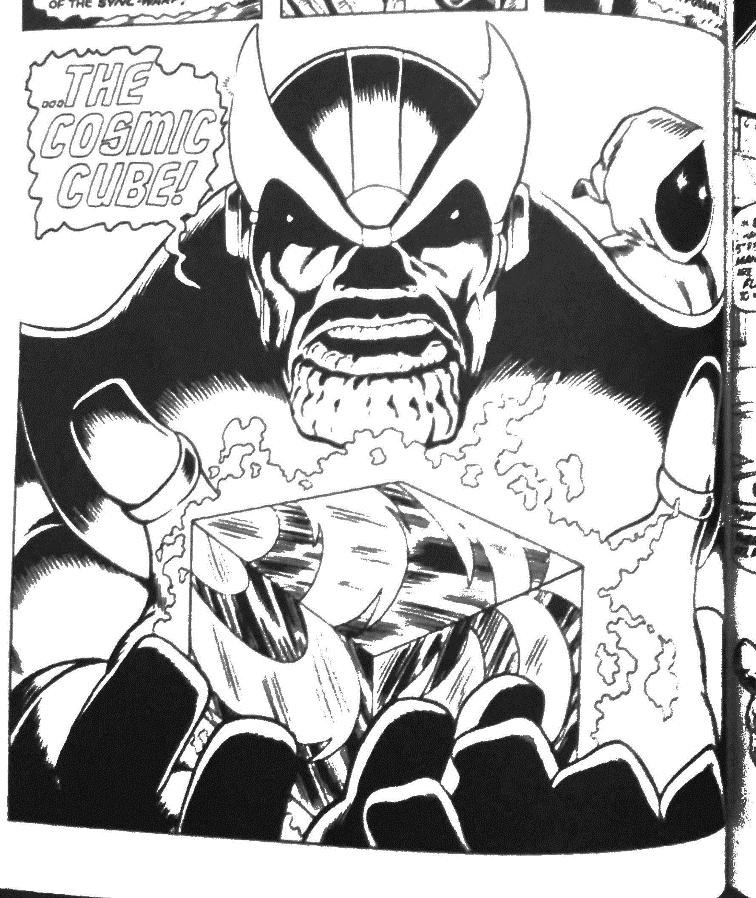
At first, the only available books featured complex geometric patterns or stylized animal prints or nature scenes – you know… adult things. Fortunately, comics publishers have been jumping on the bandwagon and one can now get coloring books with content ranging from EC’s horror line to everyone’s favorite moral reprobate: Deadpool.

All of that is well and good and I have purchased coloring books of both stripes – sober adult content and cheeky adolescent fare – and have enjoyed coloring in each kind. Up to a point.

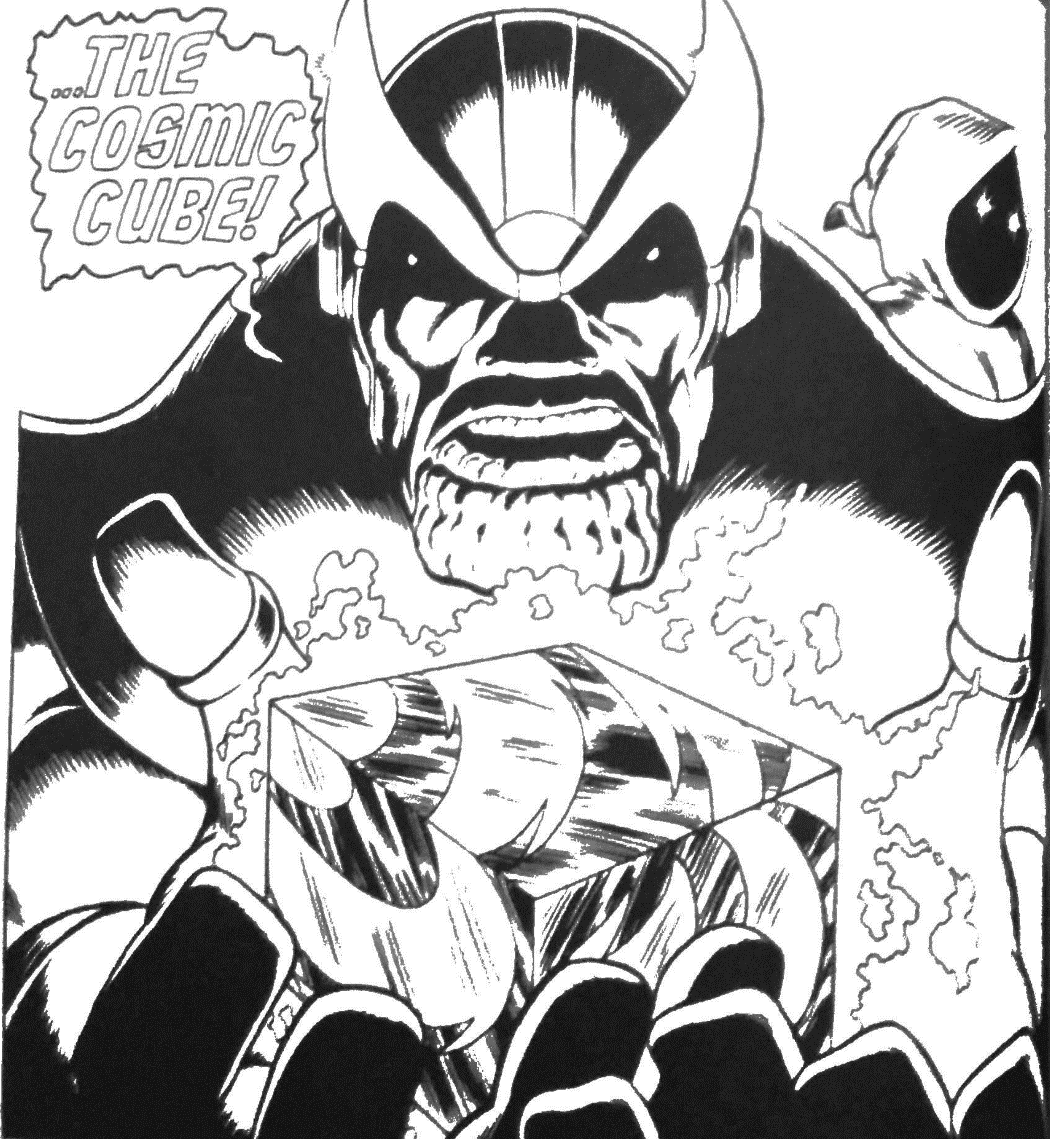
Of course coloring the image has its drawbacks. The most notable one is that the original unfilled image is lost forever once you start coloring unless you buy another copy or you photocopy, scan, or otherwise digitally reproduce the original. Personally, I don’t want to indulge in the former and much prefer the reproduction route. But if one is going to do that, why limit oneself to what the publishers deem appropriate. Branch out.

And so that is just what I did. Using a smart phone (or digital camera or a scanner), some photo-editing software, and some of those DC Showcase or Marvel Essentials black-and-white reprint volumes, you can make your own custom coloring book and get started practicing as a do-it-yourself colorist.

For this post, I photographed an image from *Essential Captain Marvel, Volume 2* in which Thanos first reveals that he possesses the cosmic cube. To capture the image, I used Genius Scan on my Samsung Galaxy Note 3 (old but powerful and much beloved). I like Genius Scan since it corrects for some curvature and rotation automatically. The page was in the middle of the volume which is a thick as an old-fashioned phone book from a mid-size city. The raw image looked like this



I then loaded the image into the GIMP and cropped it to

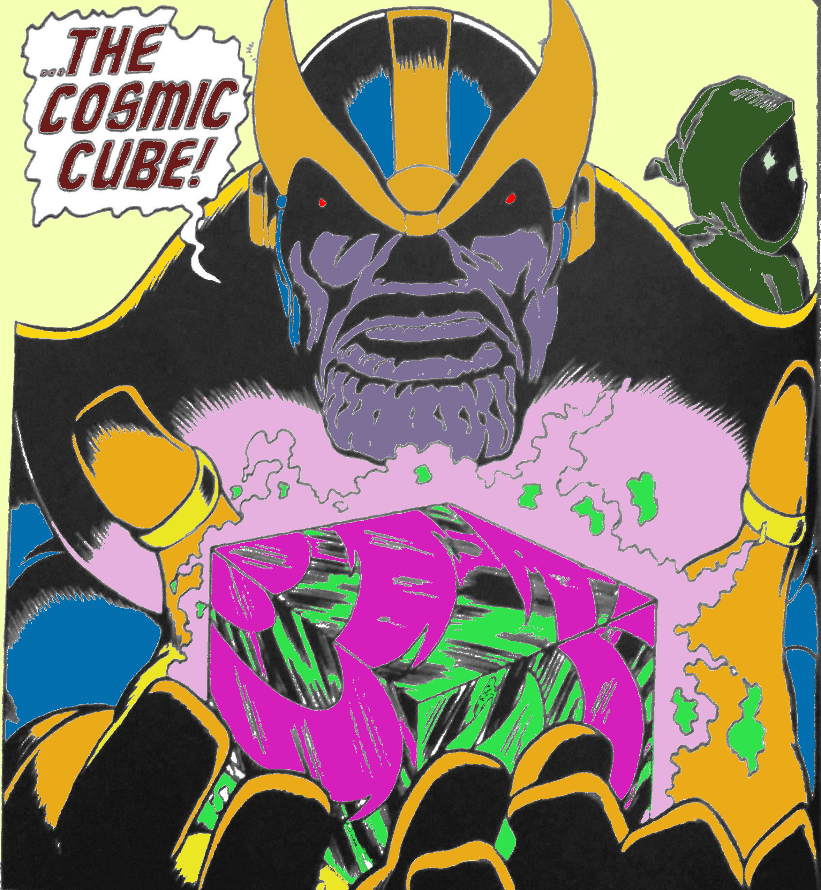


The next step was to start coloring. For this quick and dirty approach I didn’t use layers nor did I try any tools except the bucket fill. This was a bit clumsy as certain regions that looked closed often had small gaps due to either the publisher reproduction, the printing process, or the image capture. Whatever the reason, when a small, undesired gap was present the bucket fill would sometimes over fill as in this image



At times like this, the undo (ctrl-Z) I your friend. Repairs involving the eye-dropper (color-picker) tool and the brush. Simply grab a gray/black from some point nearby and close the gap and color again.

Using this rather primitive process, it only took me about 10-15 minutes to color in the image to



Clearly the image needs work, especially around the crenellated chin of the Big-T. But all, told it didn’t turn out too bad given that layers and brushes and other sophisticated tools were totally ignored.

So there you have it – a three-fold win: 1) a fast way to create your own digital coloring books using your favorite art, 2) a new use/justification for buying the cheap B&W reprint volumes, and 3) a path to learn to be a real colorist without the need to find an artist and inker.