Invisible networks 2023

Julien Kirch

2023-04-01

03 — Invisible networks

04 — Day 01: how to align friends and optimize people

05 — Day 02: wikinomicon

06 — Day 03:

07 — Day 04:

08 — Day 05:

09 — Day 06:

10 — Day 07:

11 — Day 08:

12 — Day 09:

13 — Day 10:

14 — Day 11:

15 — Day 12:

16 — Day 13:

Invisible networks

Invisible networks is a writing jam organized by ctrlcreep. The topic is "Invent a weird/magical/deeply sinister social network every day", and each day provides a prompt (and three bonus prompts in case you're not inspired by the default ones).

Thanks to them for it.

Day 01: how to align friends and optimize people

How to align your friends depends on their attributes and topologies.

If your friends all implement the `IAlignable` interface and thus provide an `alignWith(Friend otherFriend)` method, aligning them can be done within a constrained time and memory envelope. With a high-end machine or specialized hardware, it can be done at 60fps (friends per second).

If it's not the case, it will be determined by your friends' topology. Using your friends' real shape only works if you're okay with static aligning.

Instead you have to approximate their shape, and the common way is to use solid boxes (aka rectangular cuboids).

The main issue is how to deal with friends with unusual topologies (FwUT). An unusual topology means that using a solid box would not be good enough.

The canonical example of FwUT is the torus-shaped friend, or doughnut-shaped friend, because:

- the visible surface is round
- the hole in the middle

At scale, the best solution is often to regroup your FwUTs in special areas separated by their type, so each can use a specially tuned algorithm. These algorithms are available off the shelf, often for a nominal fee, depending on the friend management engine you use.

There's a growing hype around deformable friends: new research suggests that they could be dealt with in real time unless they have specific characteristics like being vapor-based. We hope to cover this topic in a future installment.

Day 02: wikinomicon

Everybody knows the old trope of the danger of a lone occultist finding something they shouldn't have and publishing it, leading to a lot of bad stuff.

Dealing with this issue required recruiting scholars to read papers and books before they are put in print, thus the establishment of the "peer review" system.

But nowadays the risky sources are not scholarly related but are :

- Personal information management systems (aka note-taking software)
- Wikis, and especially enterprise and video games wikis

Compared to ancient times, monitoring their content doesn't require secret cooperation amoung lots of specialized people any more. A few people, a smart usage of modern computing trends and lots of money are enough: by leveraging VCs model and the desire of people and organizations to offload the administration of servers to third parties, if good-enough subsidized softwares are available, people will happily put their sensible data on external servers, thus enabling content scanning at scale.

But as prevention is better than cure, pushing people around dangerous topics would be even better than assessing content after the fact. The aggressive inclusion of content suggestion is a step in this direction: a few nudges here and there can significantly lower the risks.

Day 03:

Day 04:

Day 05:

Day 06:

Day 07:

Day 08:

Day 09:

Day 10:

Day 11:

Day 12:

Day 13: