Governance

i290M Open Collaboration and Peer Production

Sebastian Benthall

Cooperation evolves

Governments are made

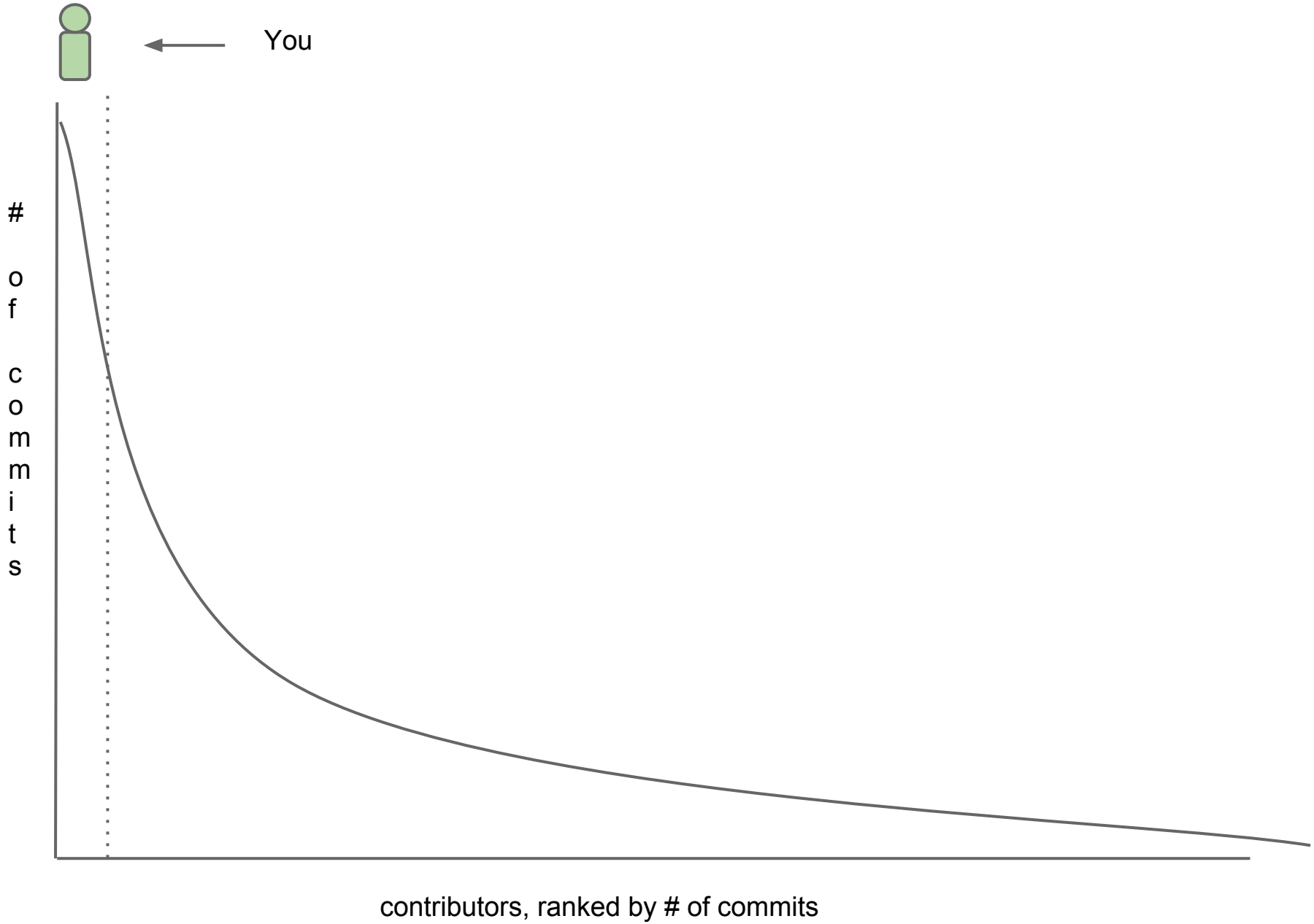
Why this lecture?

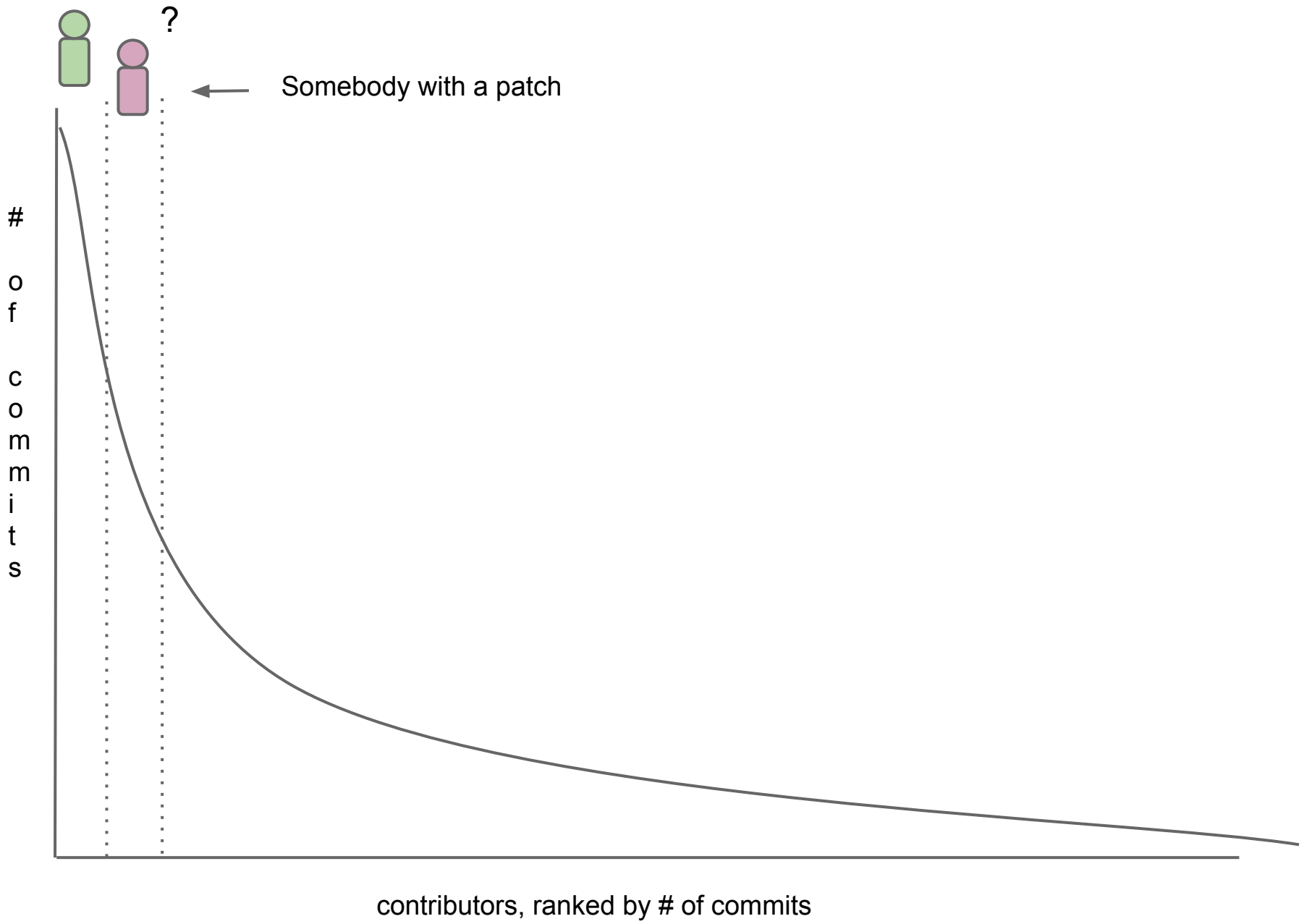
This is something you have a choice in as a participant and leader

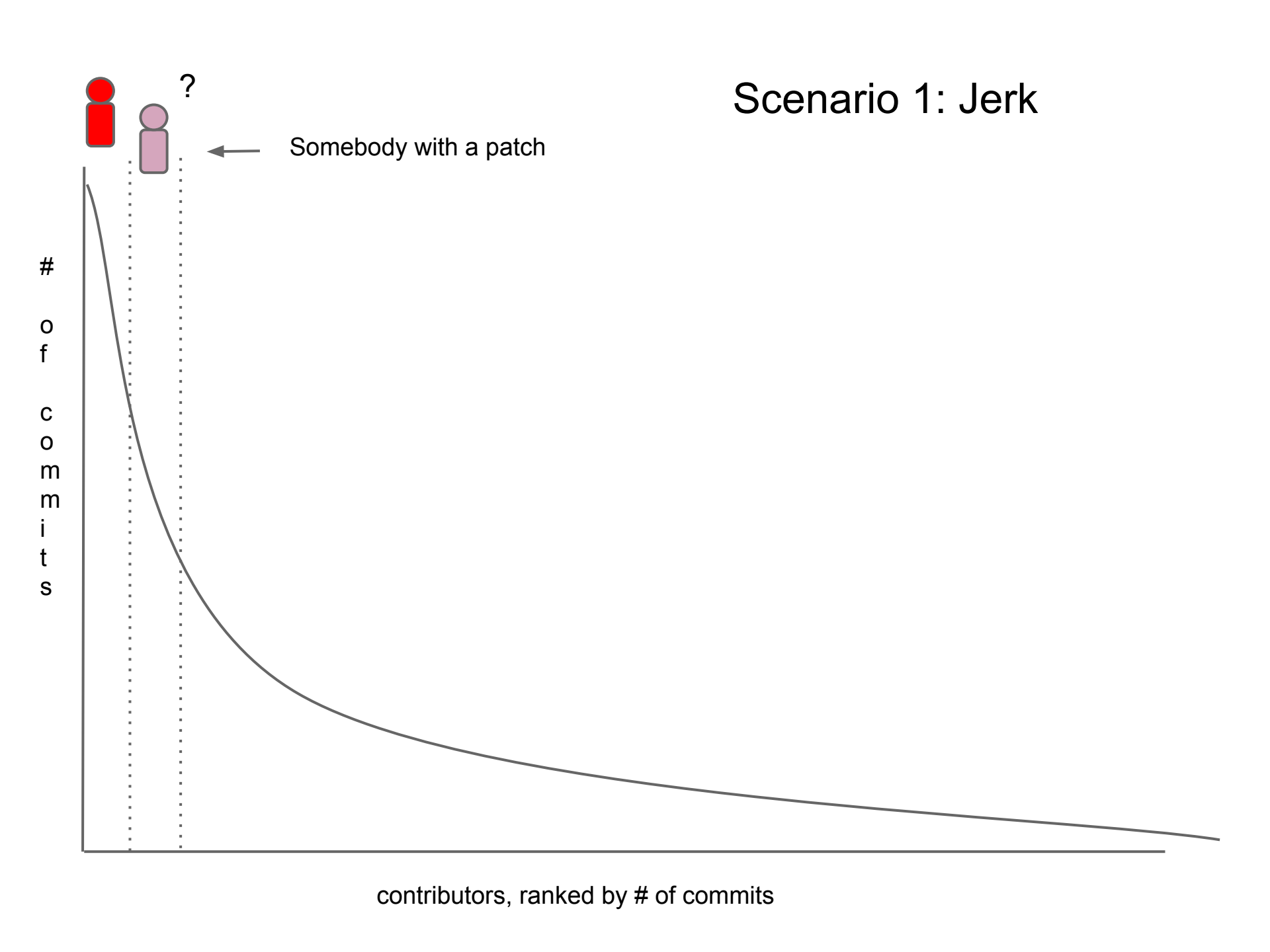
1) to understand open collaboration's governance aspect

2) to understand governance using open collaboration as laboratories

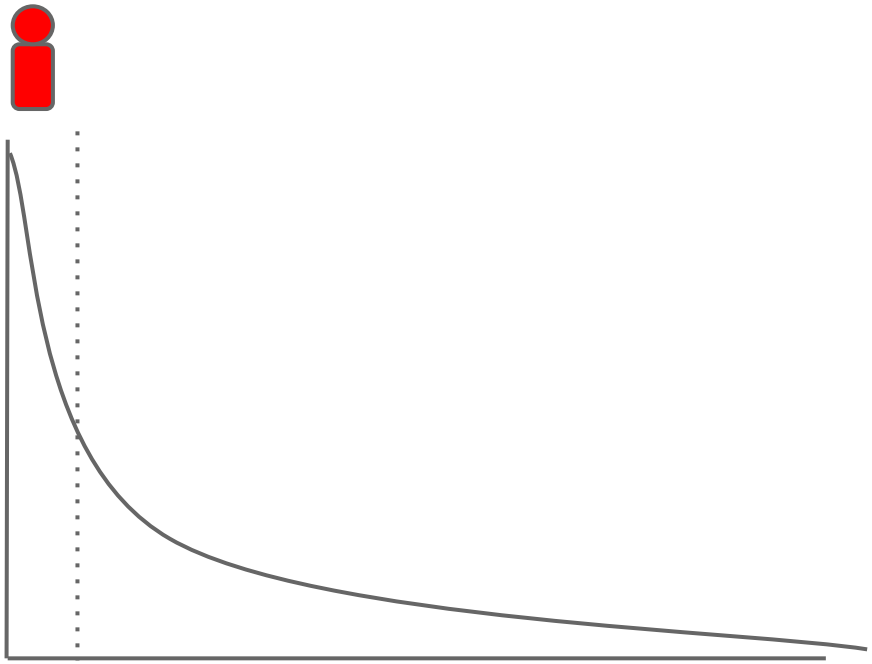
the trajectory of an open source project


You
# of commits
contributors, ranked by # of commits

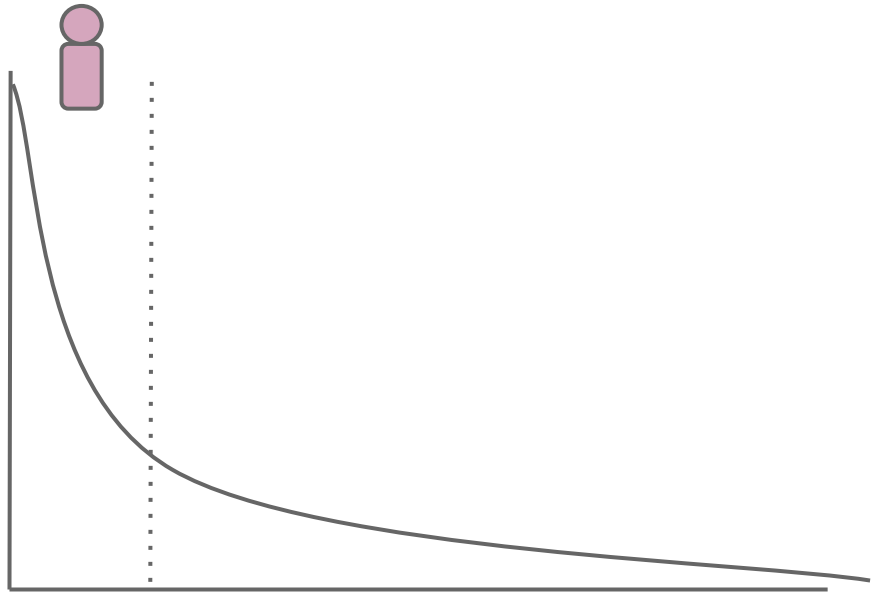

?
Somebody with a patch
# of commits
contributors, ranked by # of commits


Scenario 1: Jerk
?
Somebody with a patch
# of commits
contributors, ranked by # of commits

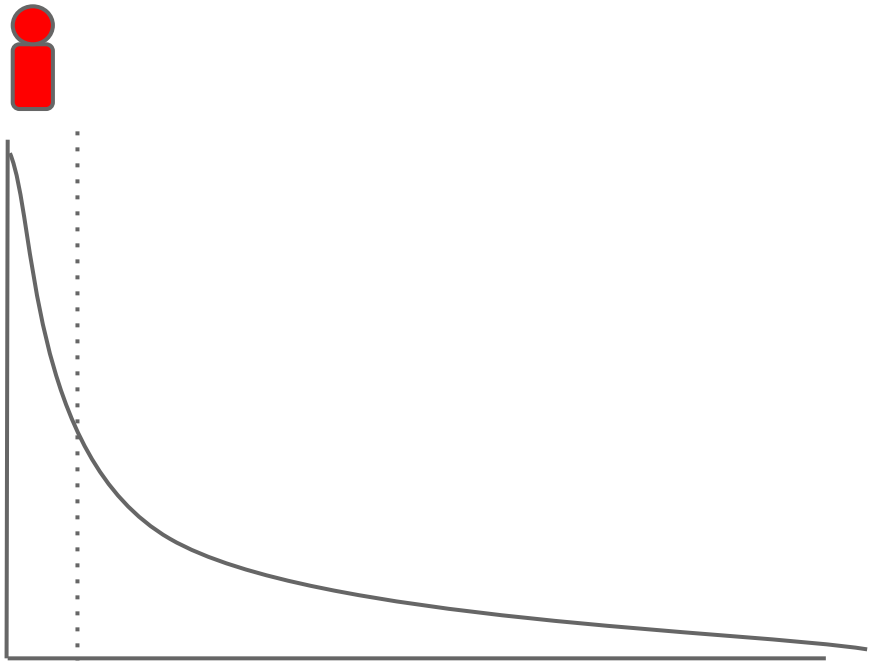
Scenario 1: Jerk



FORK!

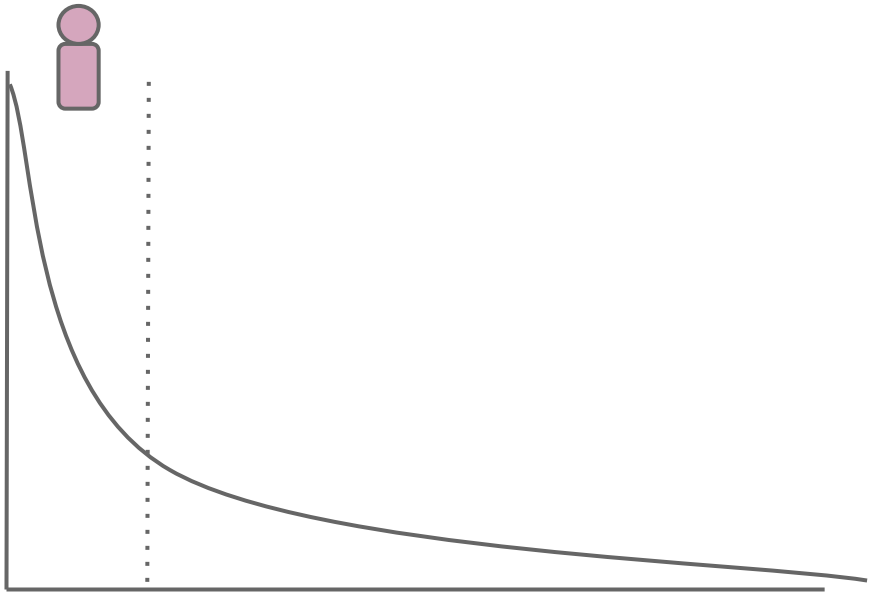


Scenario 1: Jerk

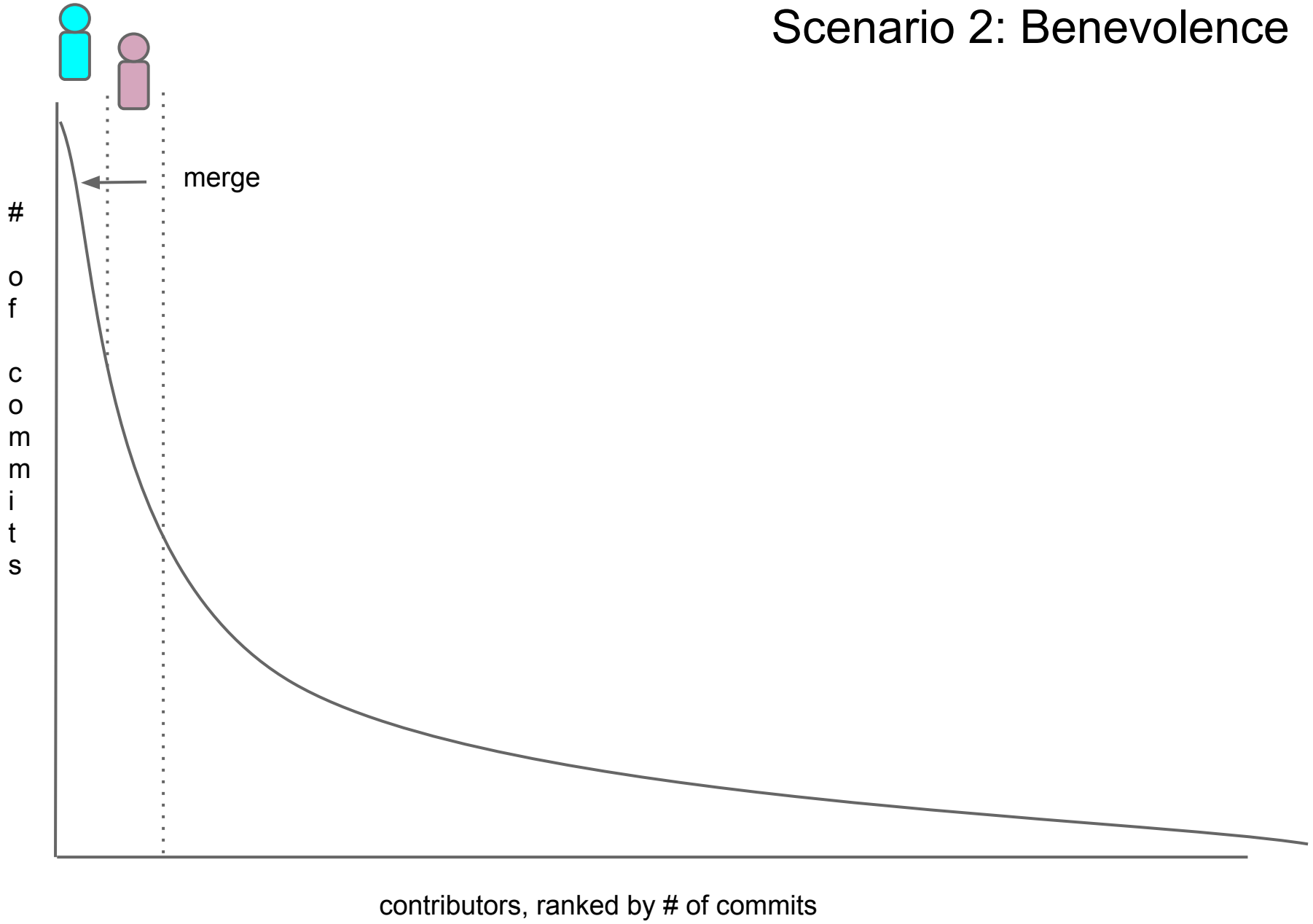


FORK!

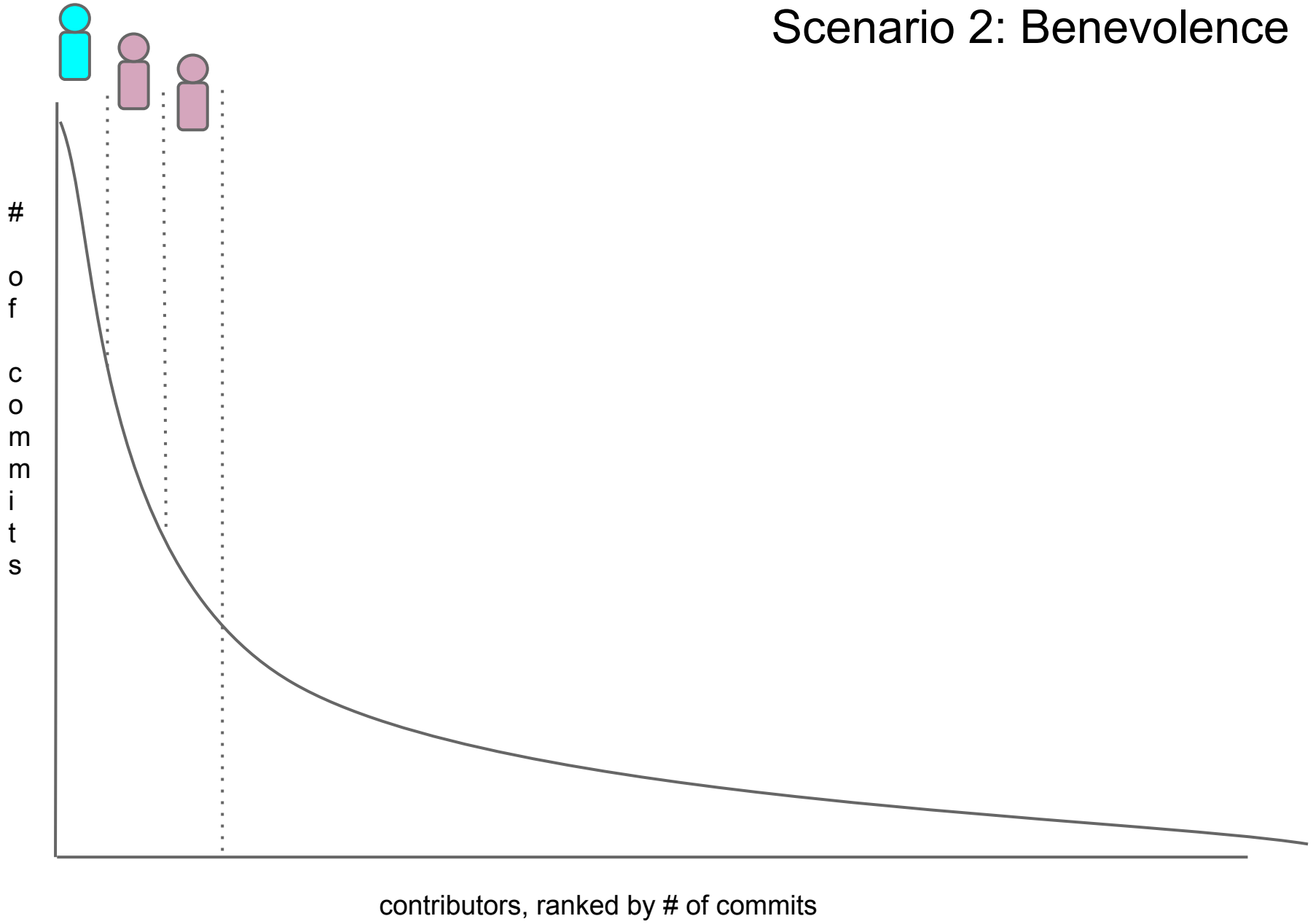
(the fork may be *better*. You now are competing with a better project, potentially not lead by a jerk)



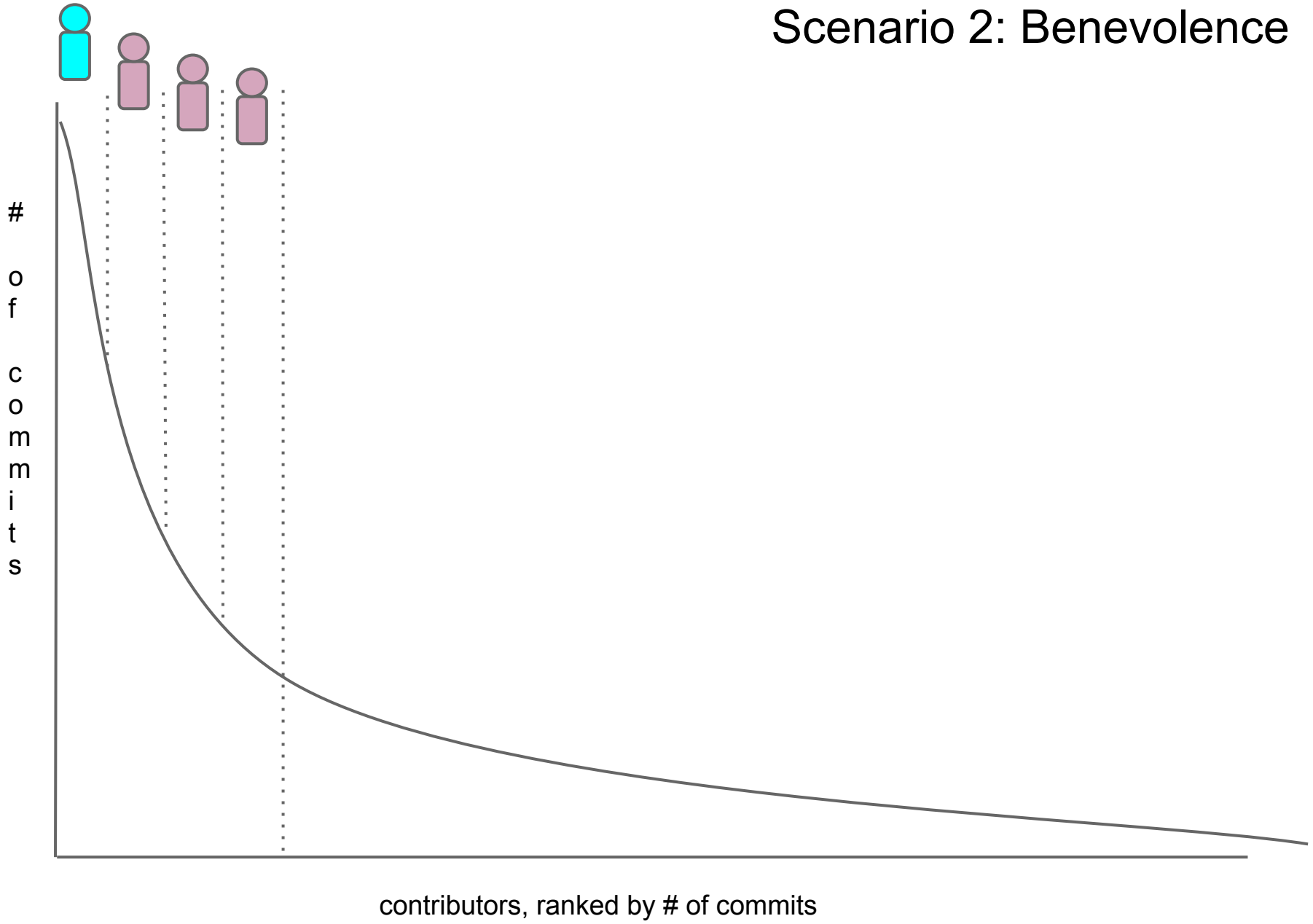
Scenario 2: Benevolence



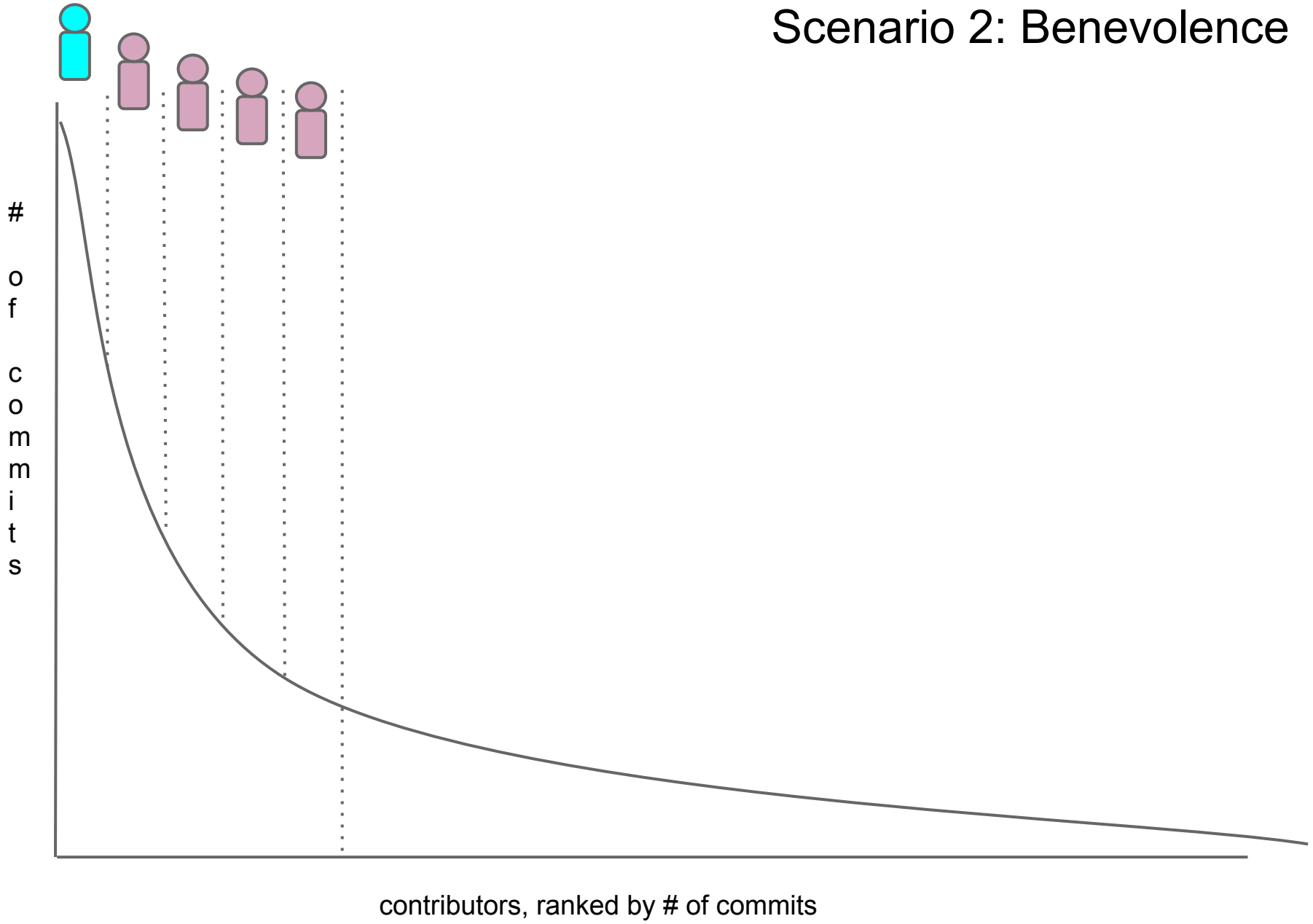
Scenario 2: Benevolence



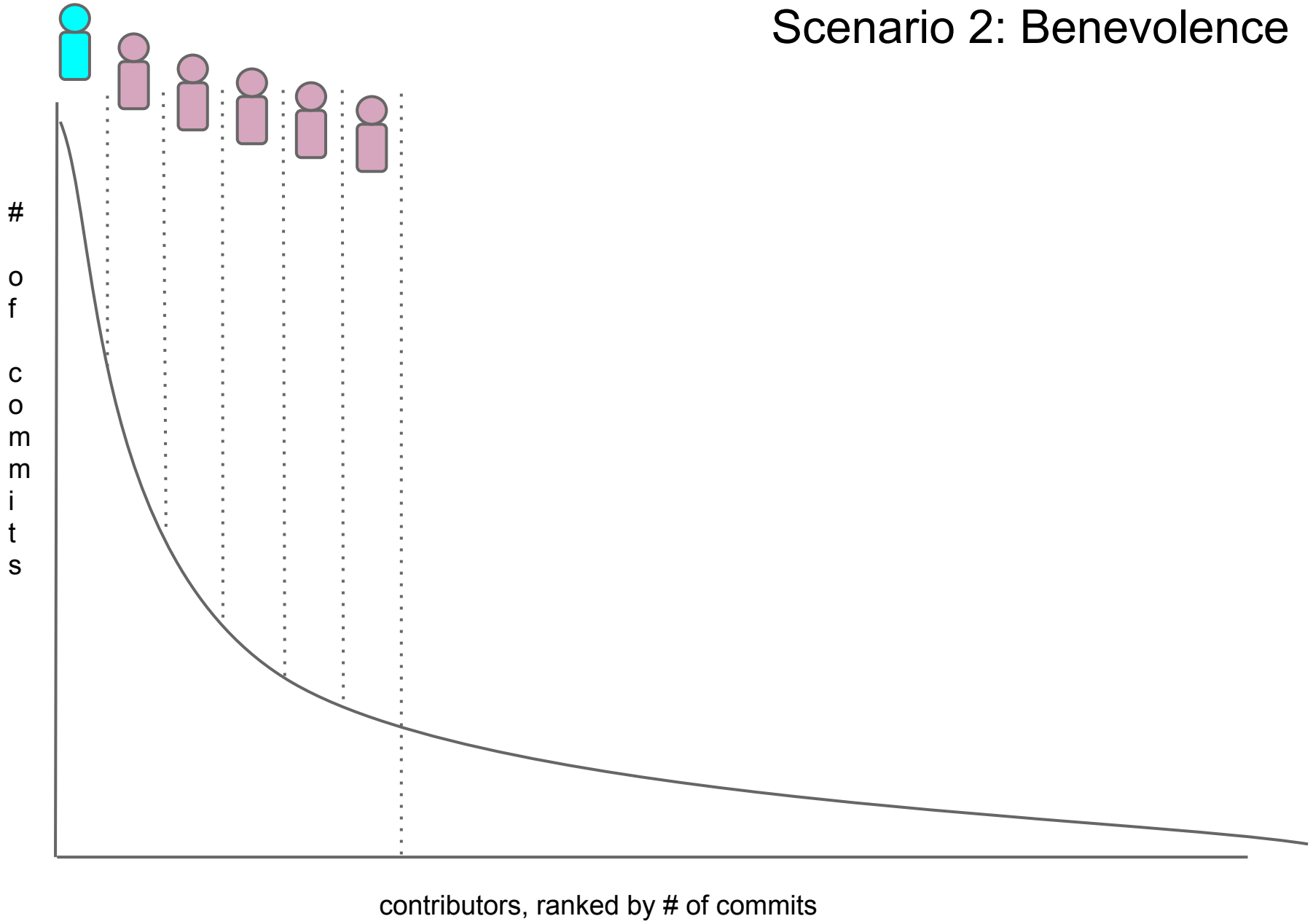
Scenario 2: Benevolence

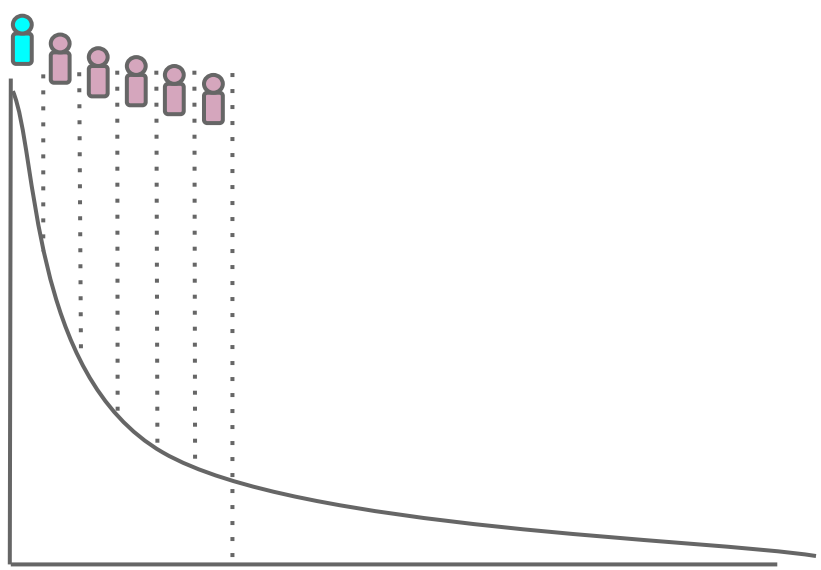


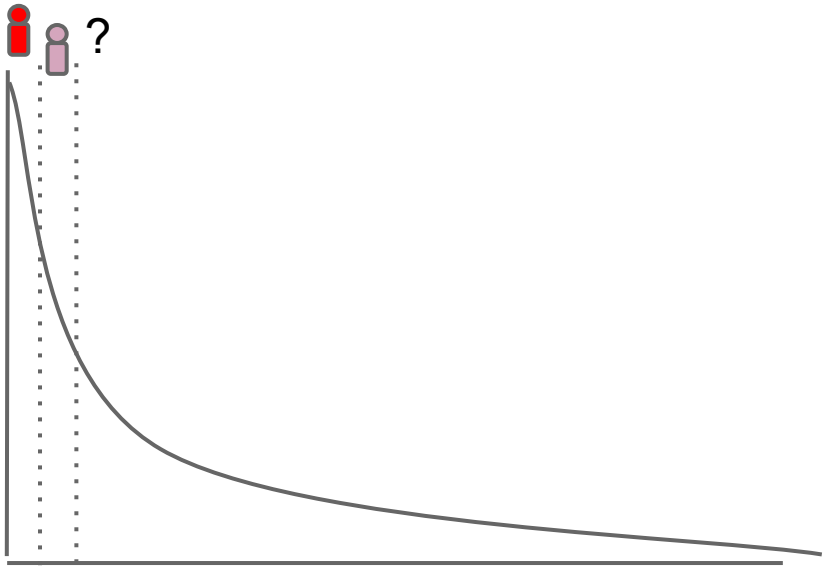
Scenario 2: Benevolence

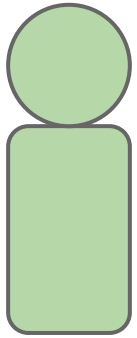


Scenario 2: Benevolence

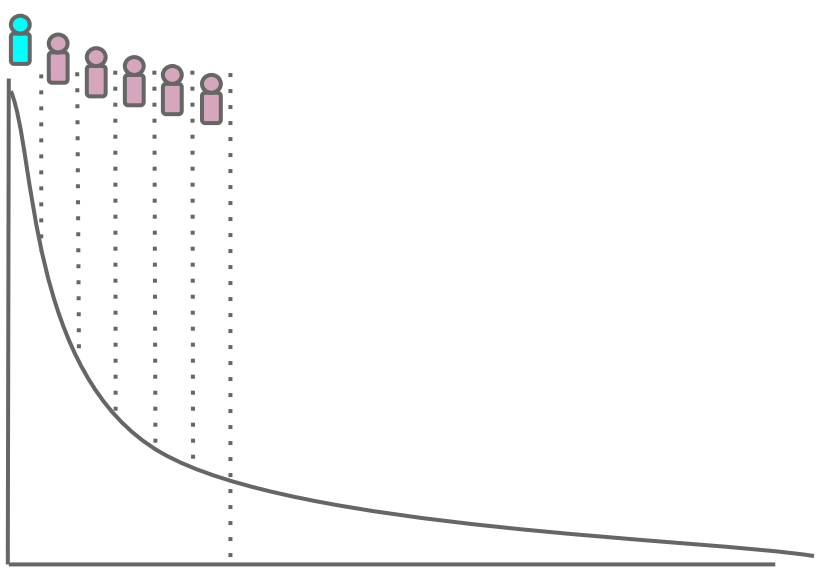


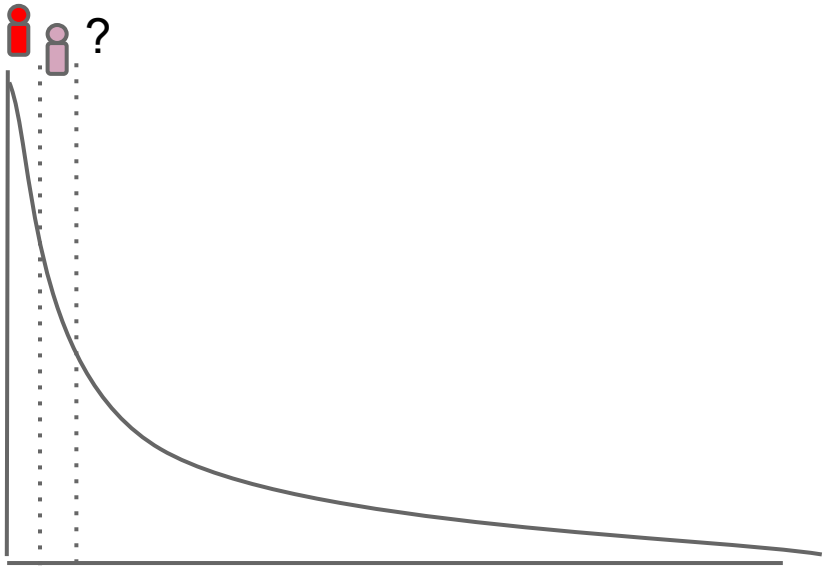


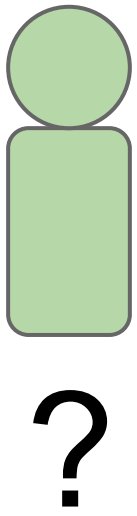




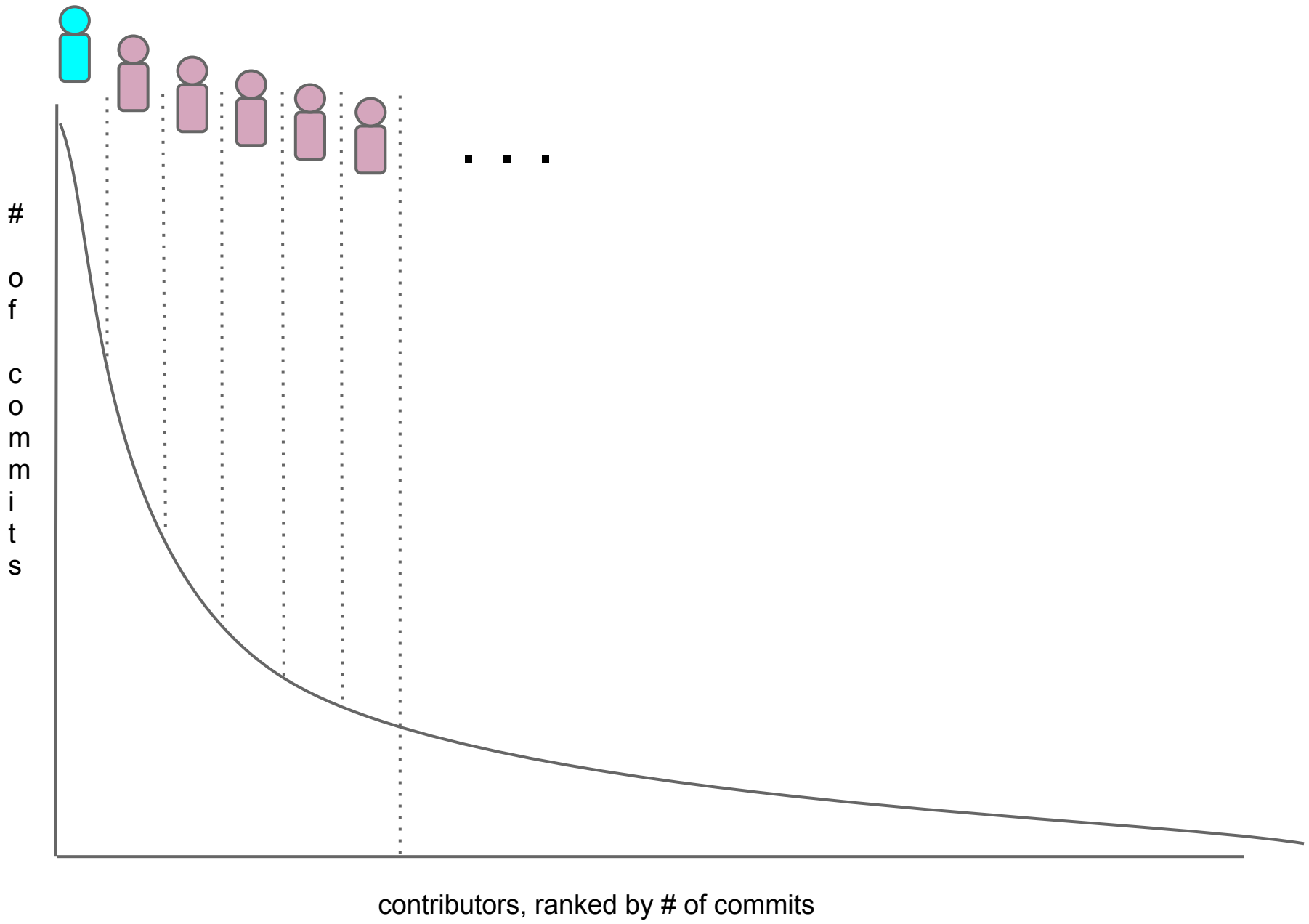
?

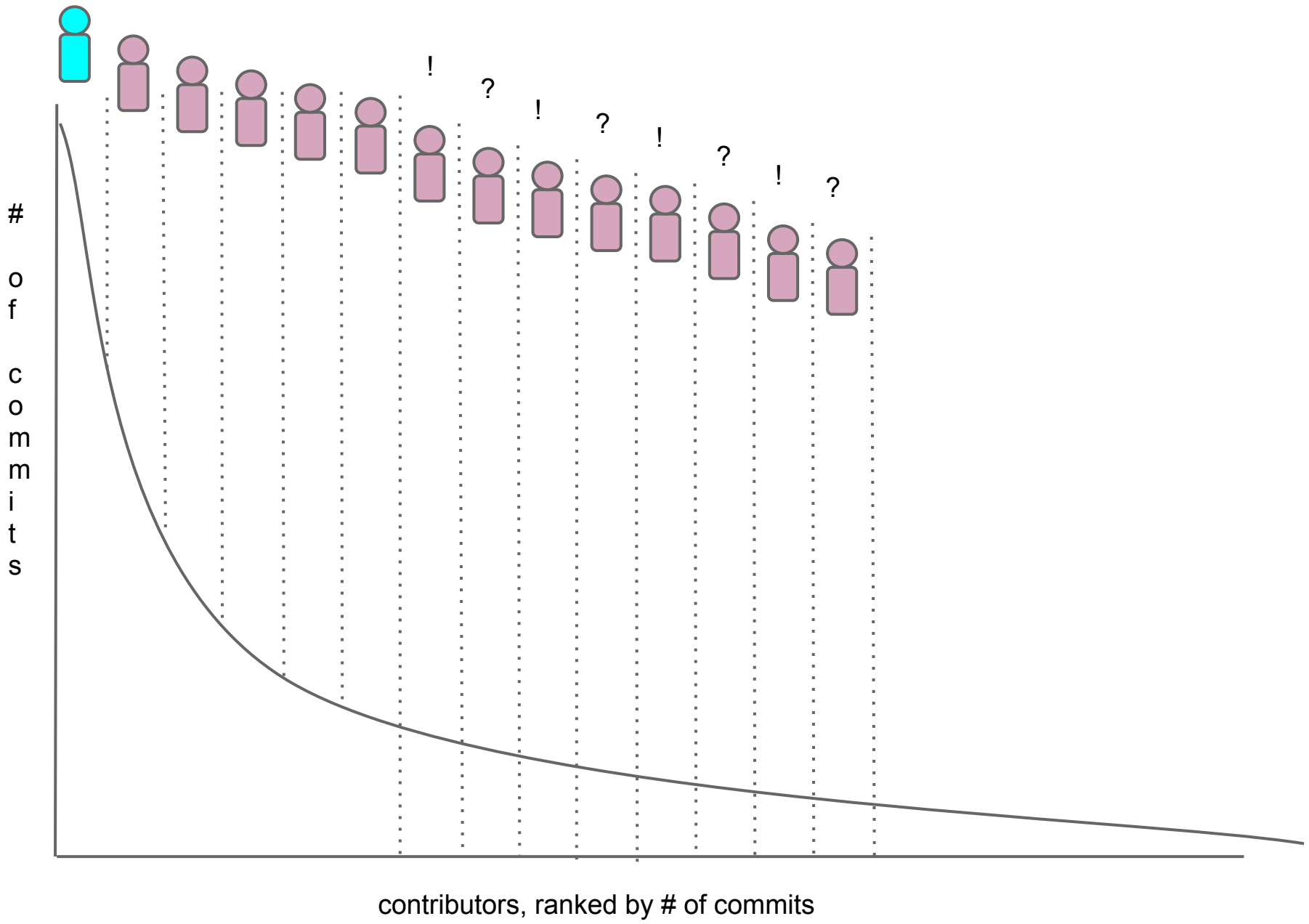


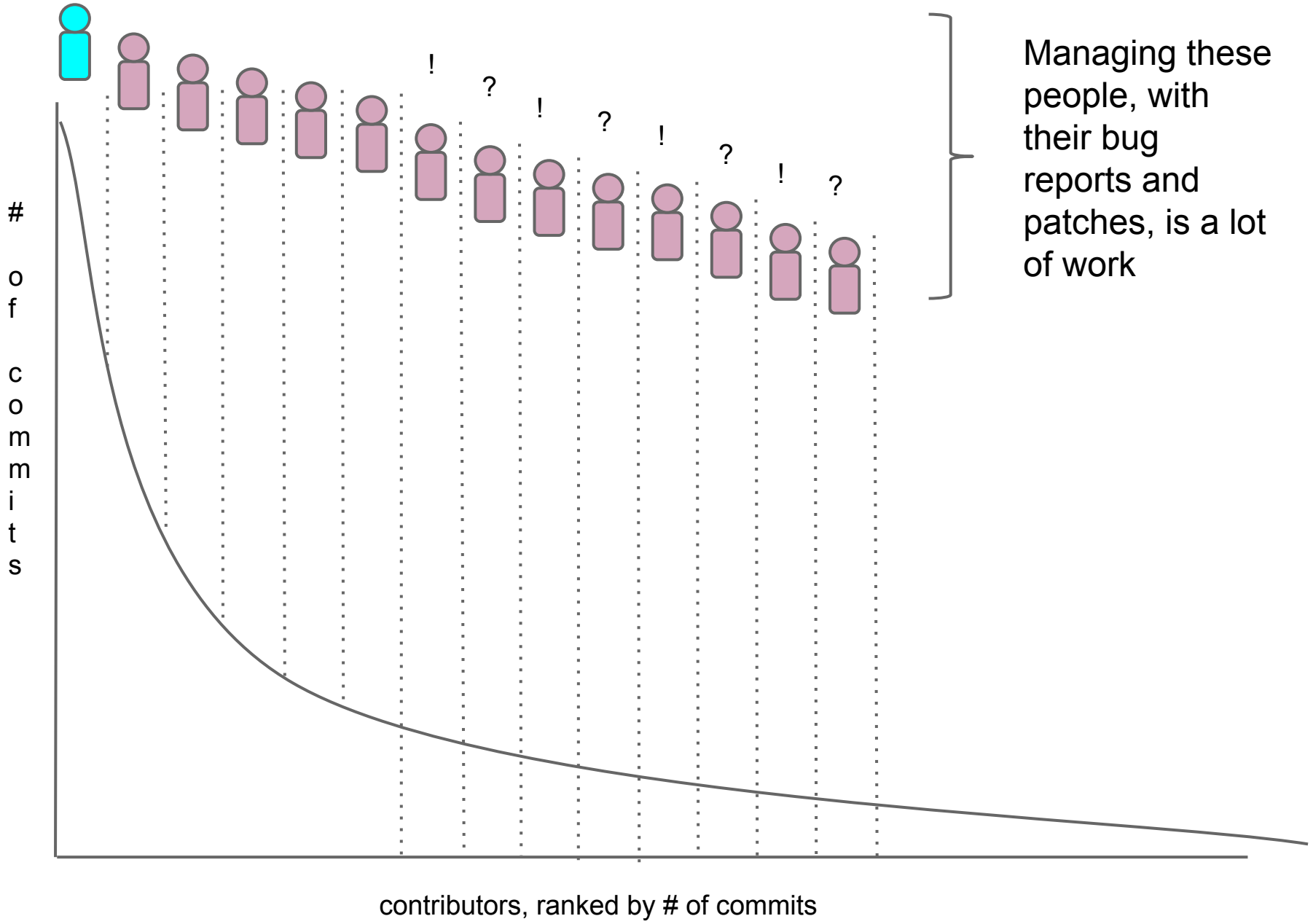




“Sometimes [the participation decision] isn’t [all] about technical considerations. There’s another OS project whose technology I use and I want to develop further, but the “benevolent dictator” is simply a dictator the few developers who stick around are like that too who needs that? This project is flexible. I’ m watching for interesting stuff to do here.” - from Shah, 2006


# of commits
contributors, ranked by # of commits


! ? ! ? ! ? ! ?
# of commits
contributors, ranked by # of commits


! ? ! ? ! ? ! ?
Managing these people, with their bug reports and patches, is a lot of work
# of commits
contributors, ranked by # of commits

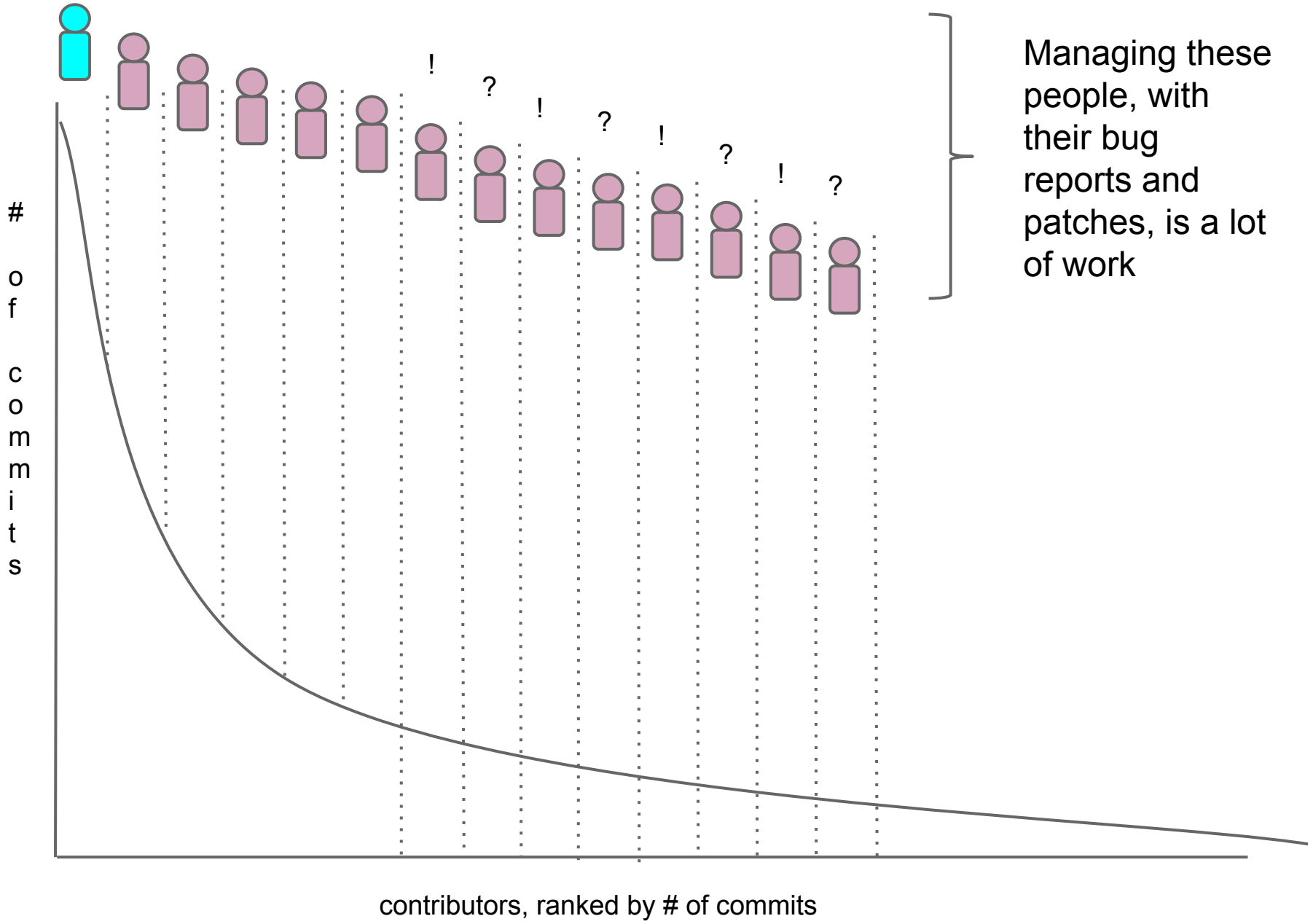
Rather than merge
in all the patches
yourself,
you need to trust
others to do it

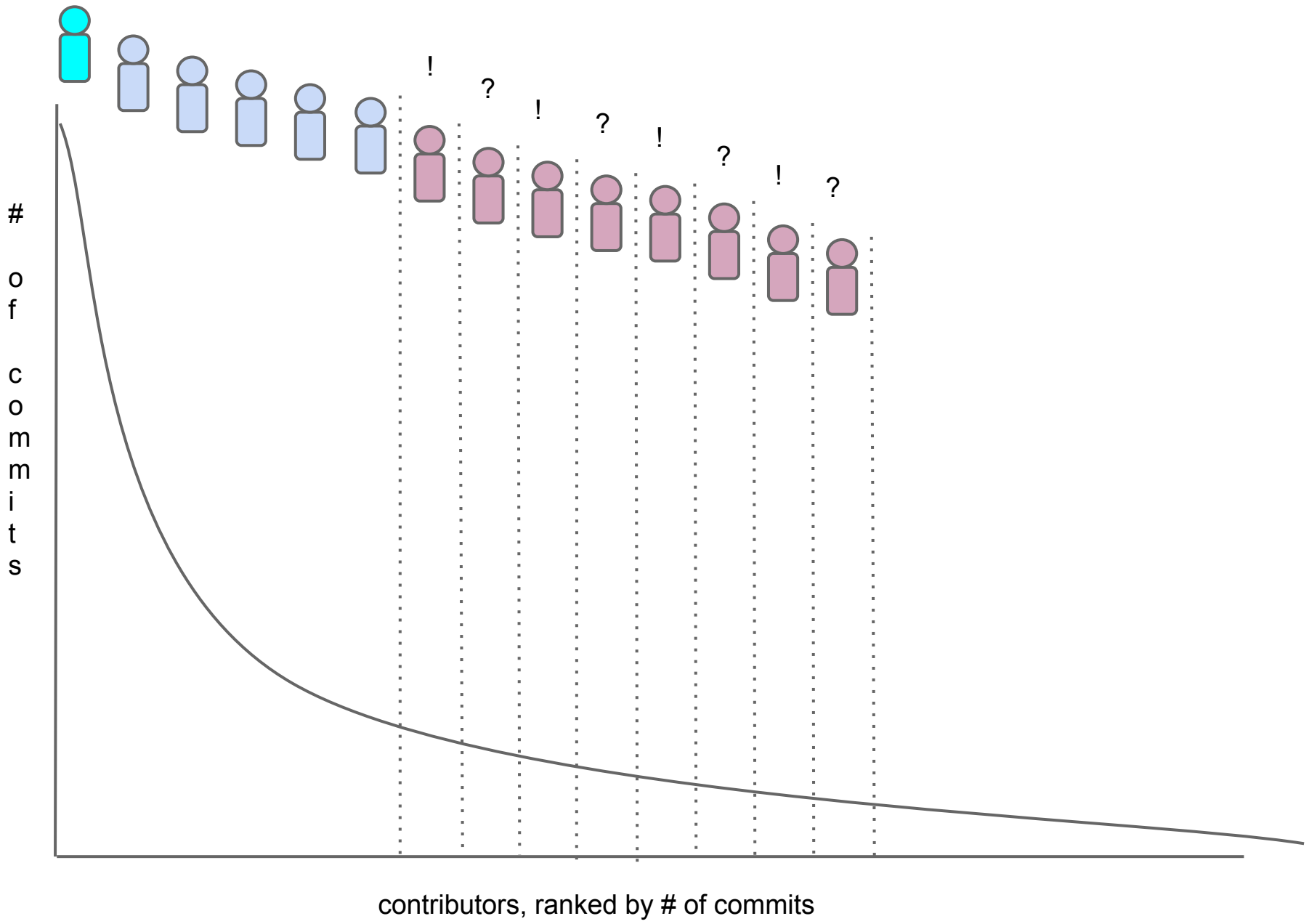
com·mit priv·i·lege

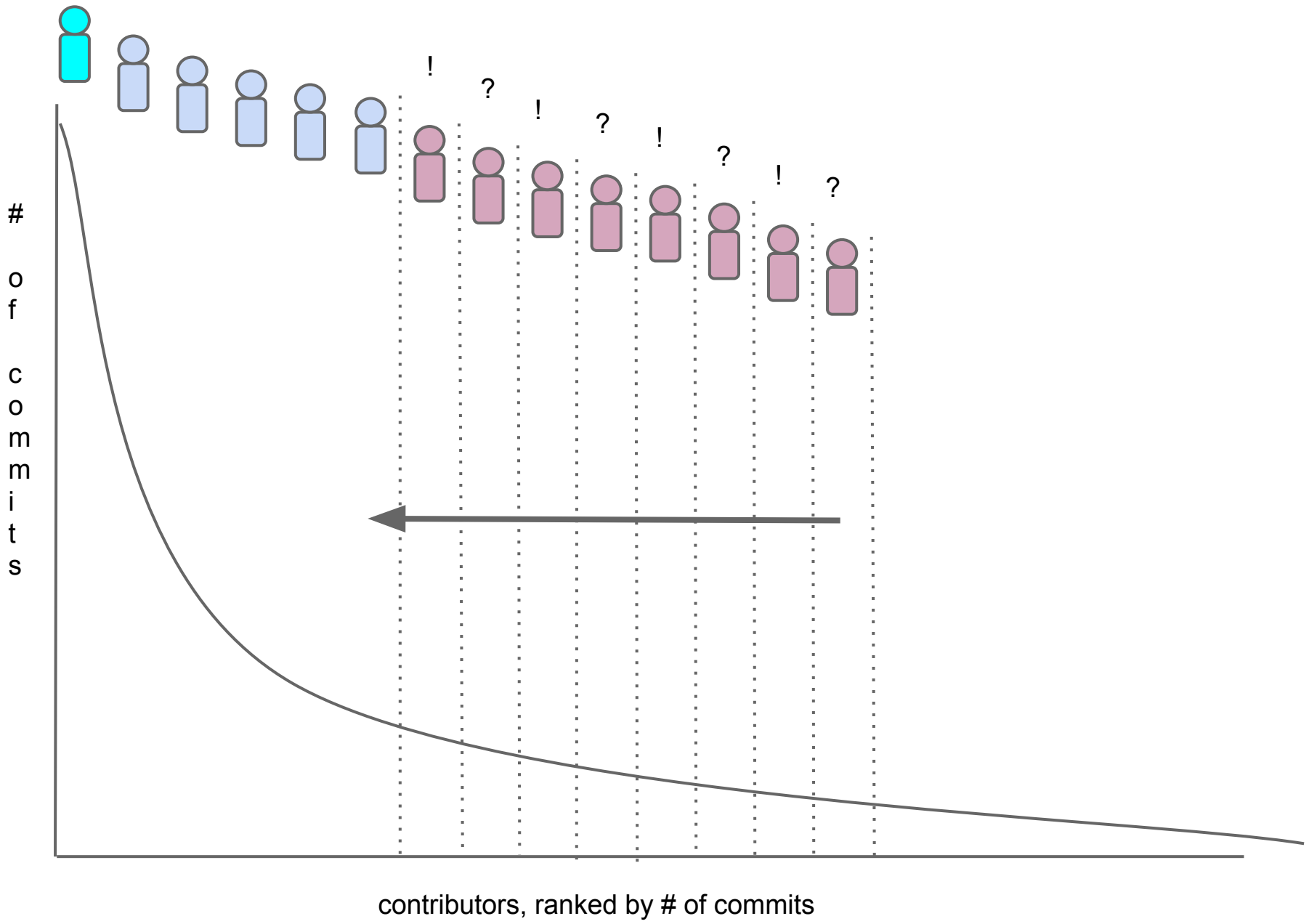
‘write access’ to the project’s central repository

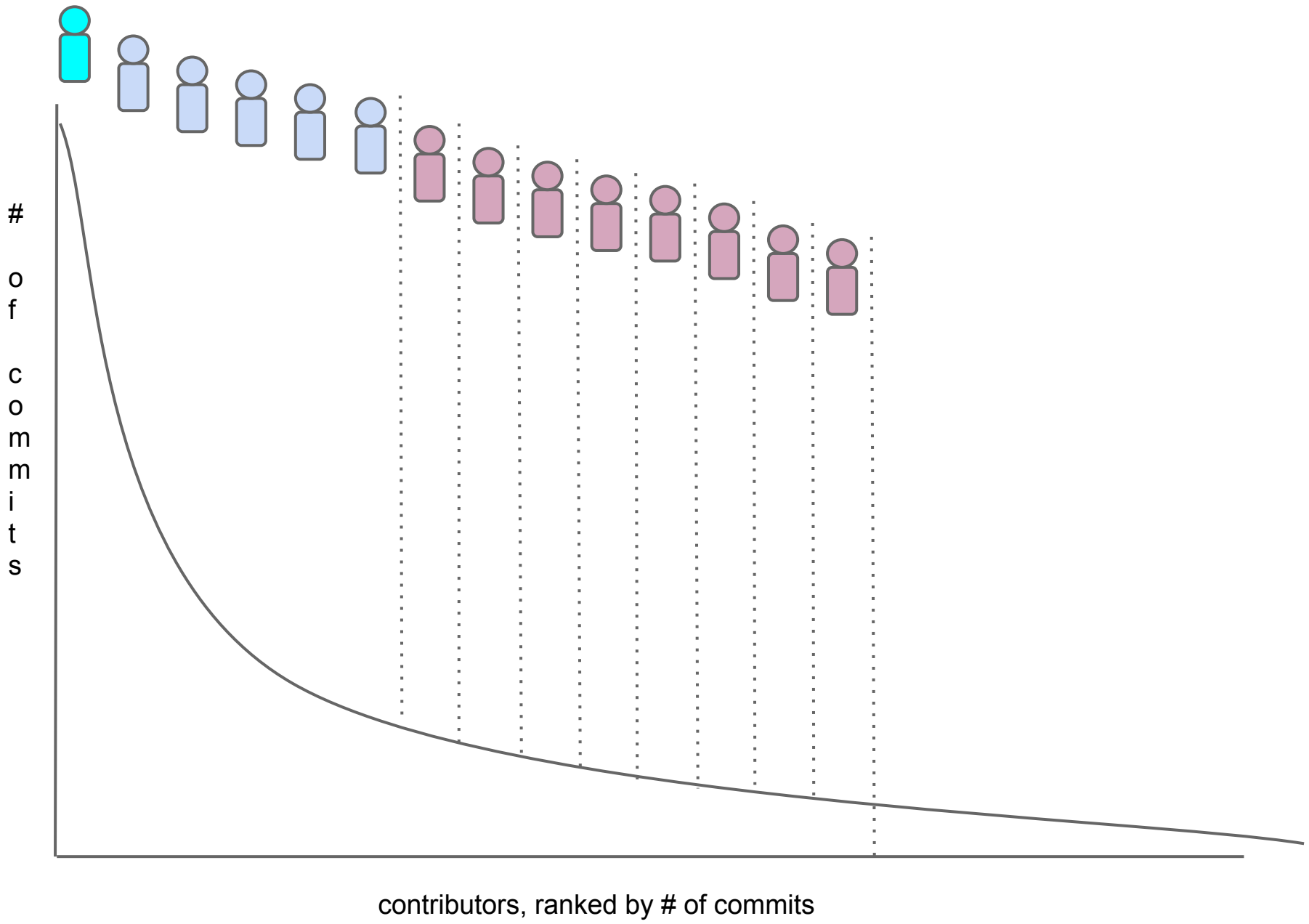
Somebody with
commit privilege
is called a
“committer”

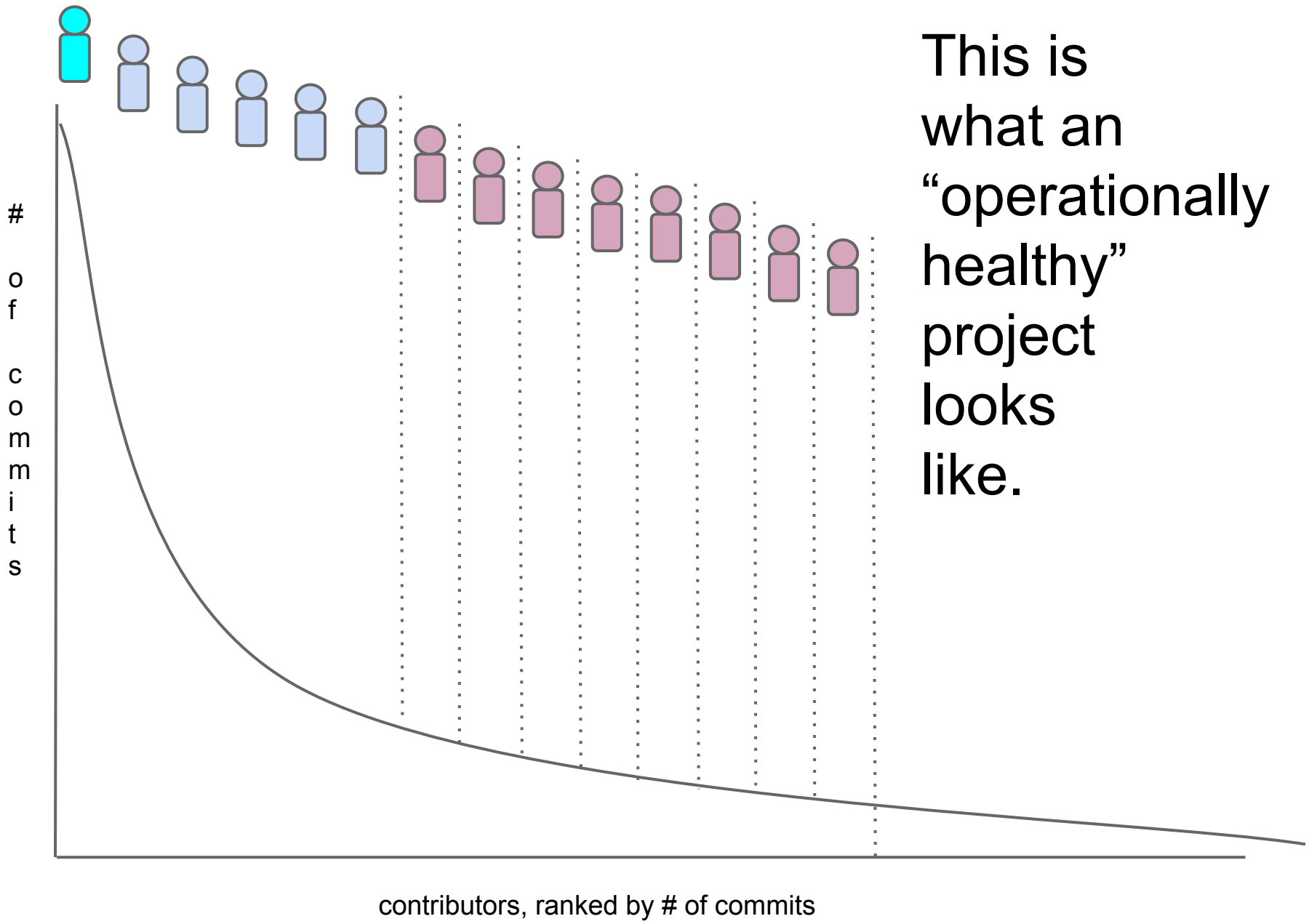
Committers
can take on some of
the work of
reviewing patches


! ? ! ? ! ? ! ?
Managing these people, with their bug reports and patches, is a lot of work
# of commits
contributors, ranked by # of commits

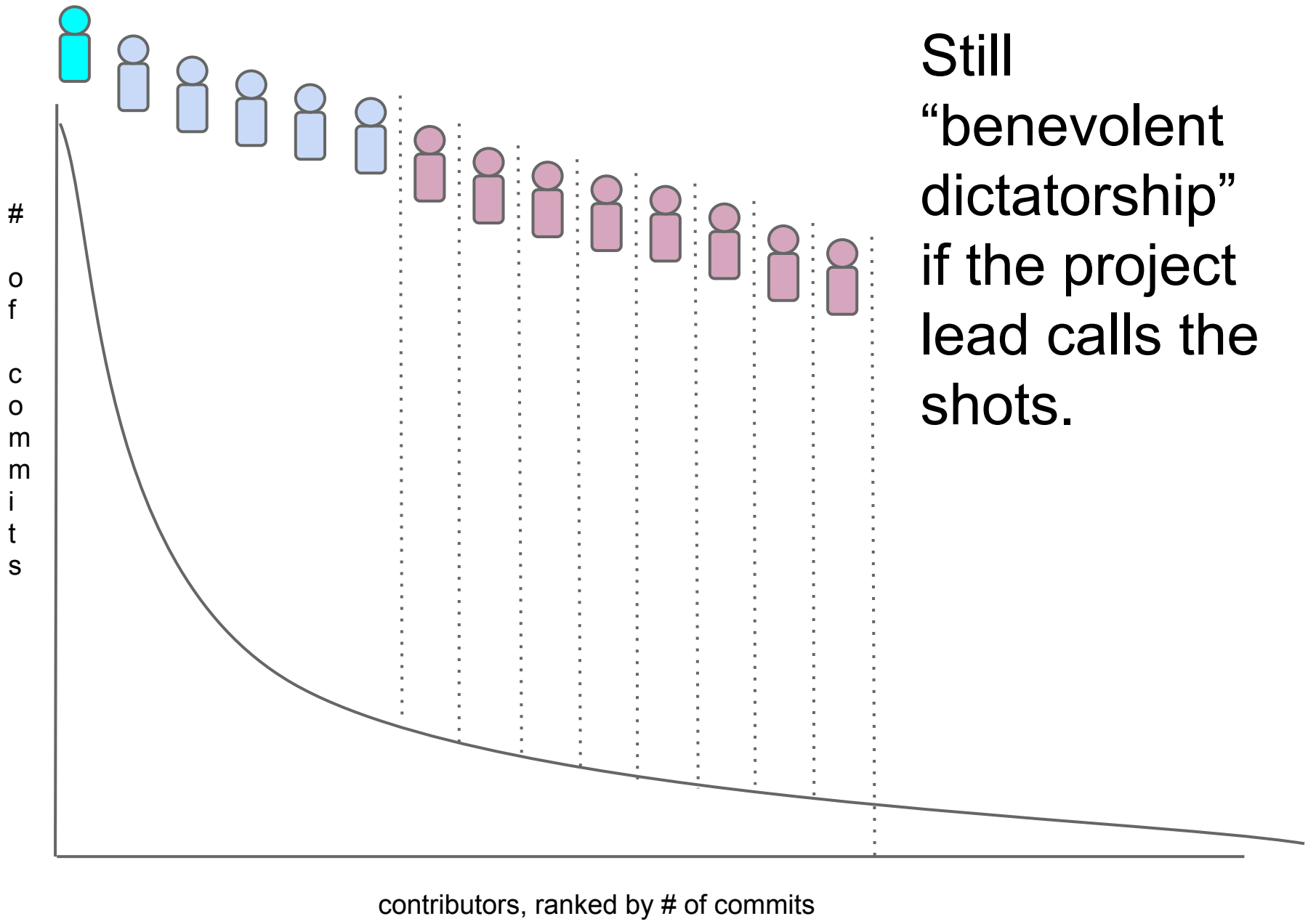

!
?
!
?
!
?
!
?
# of commits
contributors, ranked by # of commits


! ? ! ? ! ? ! ?
# of commits
contributors, ranked by # of commits

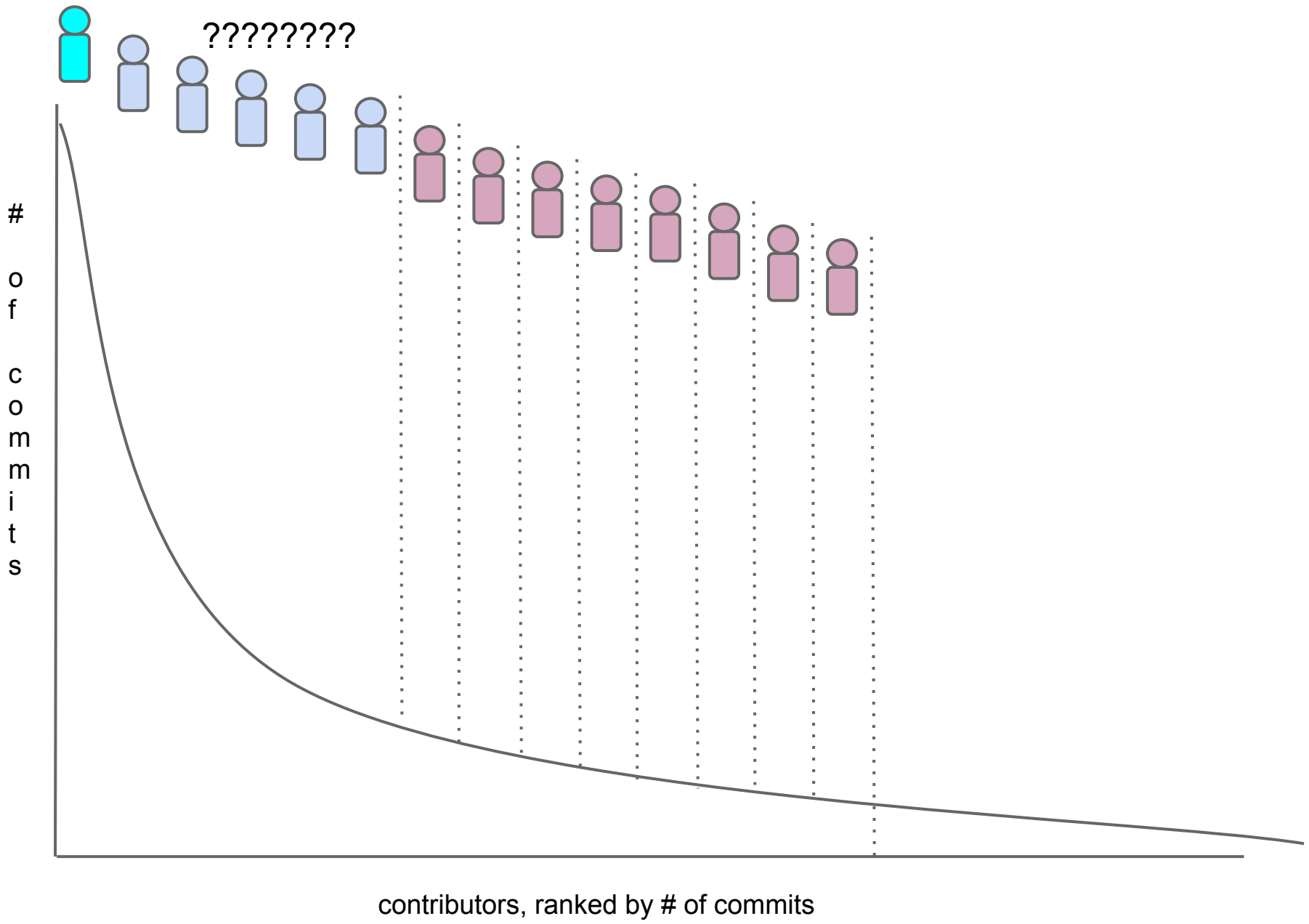

# of commits
contributors, ranked by # of commits

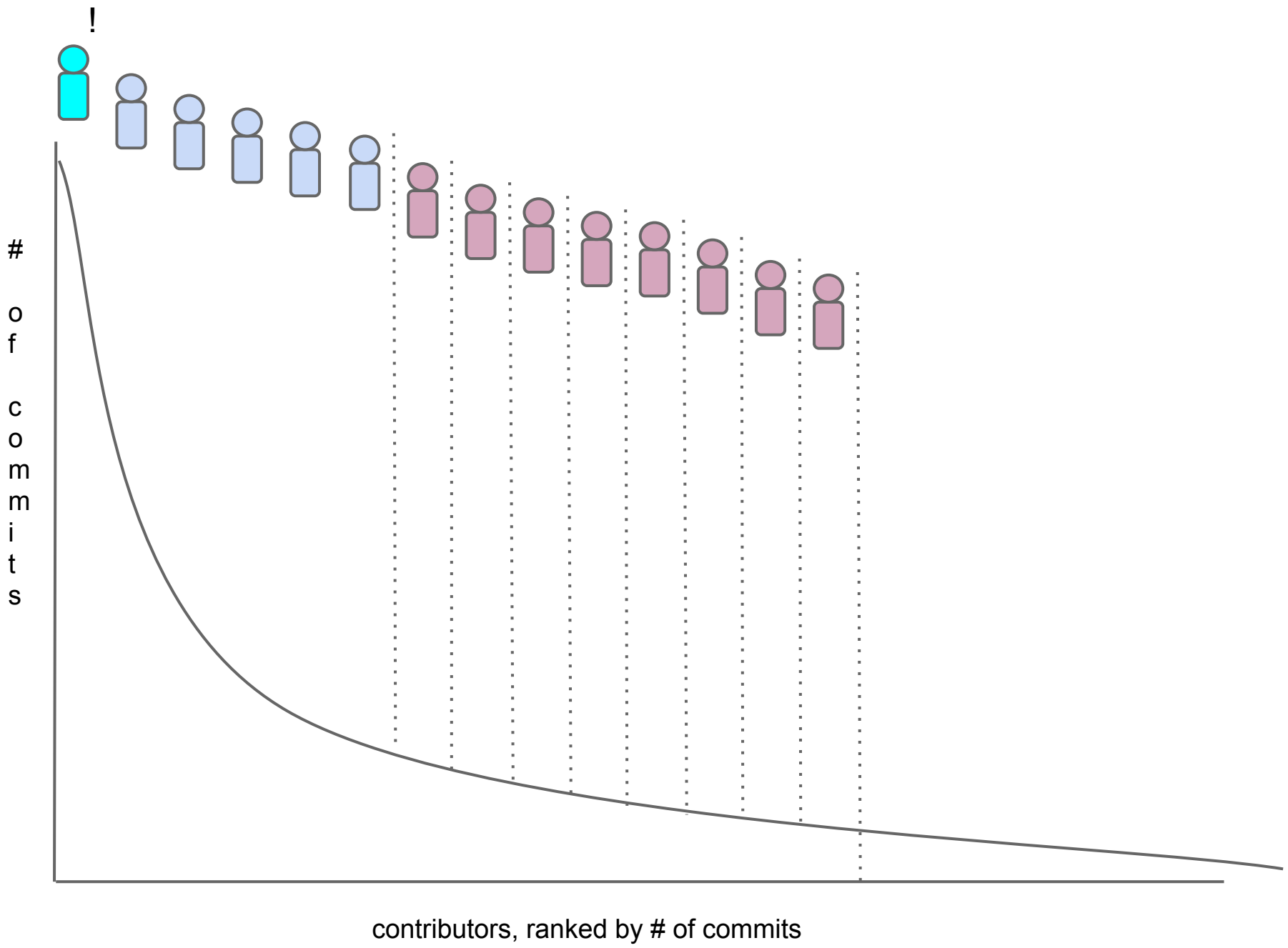


This is what an "operationally healthy" project looks like.

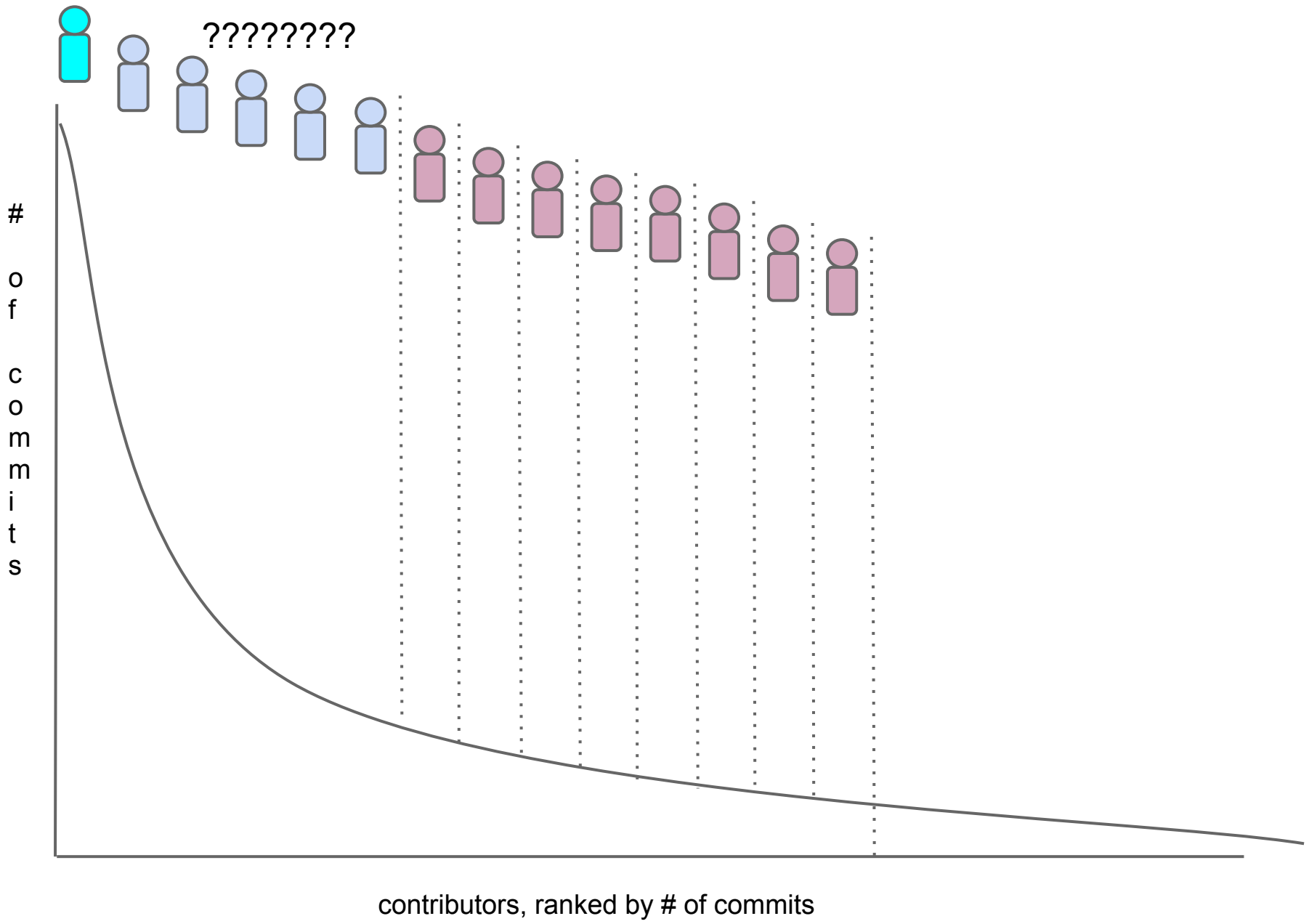


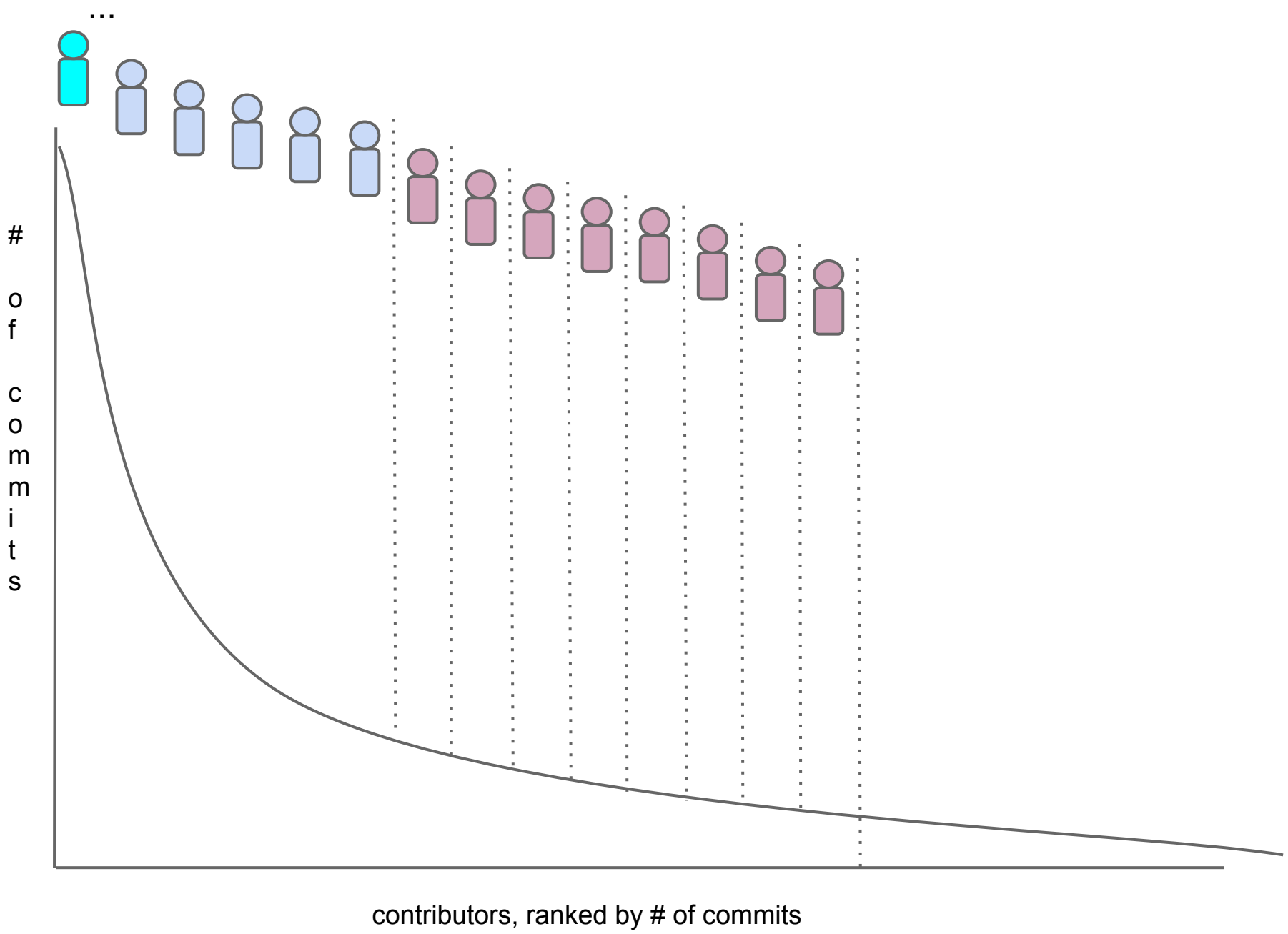
Still “benevolent dictatorship” if the project lead calls the shots.

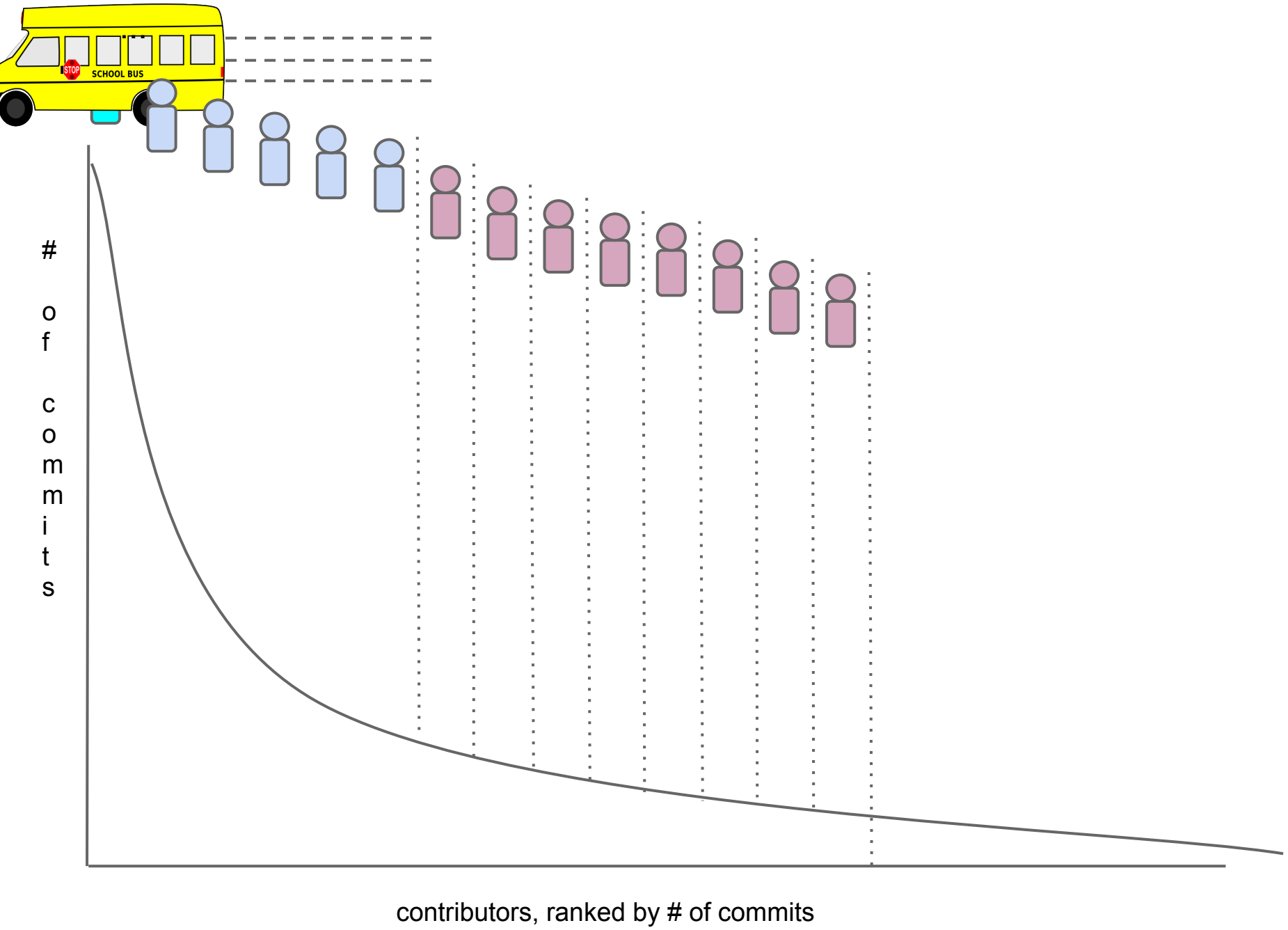

????????
# of commits
contributors, ranked by # of commits

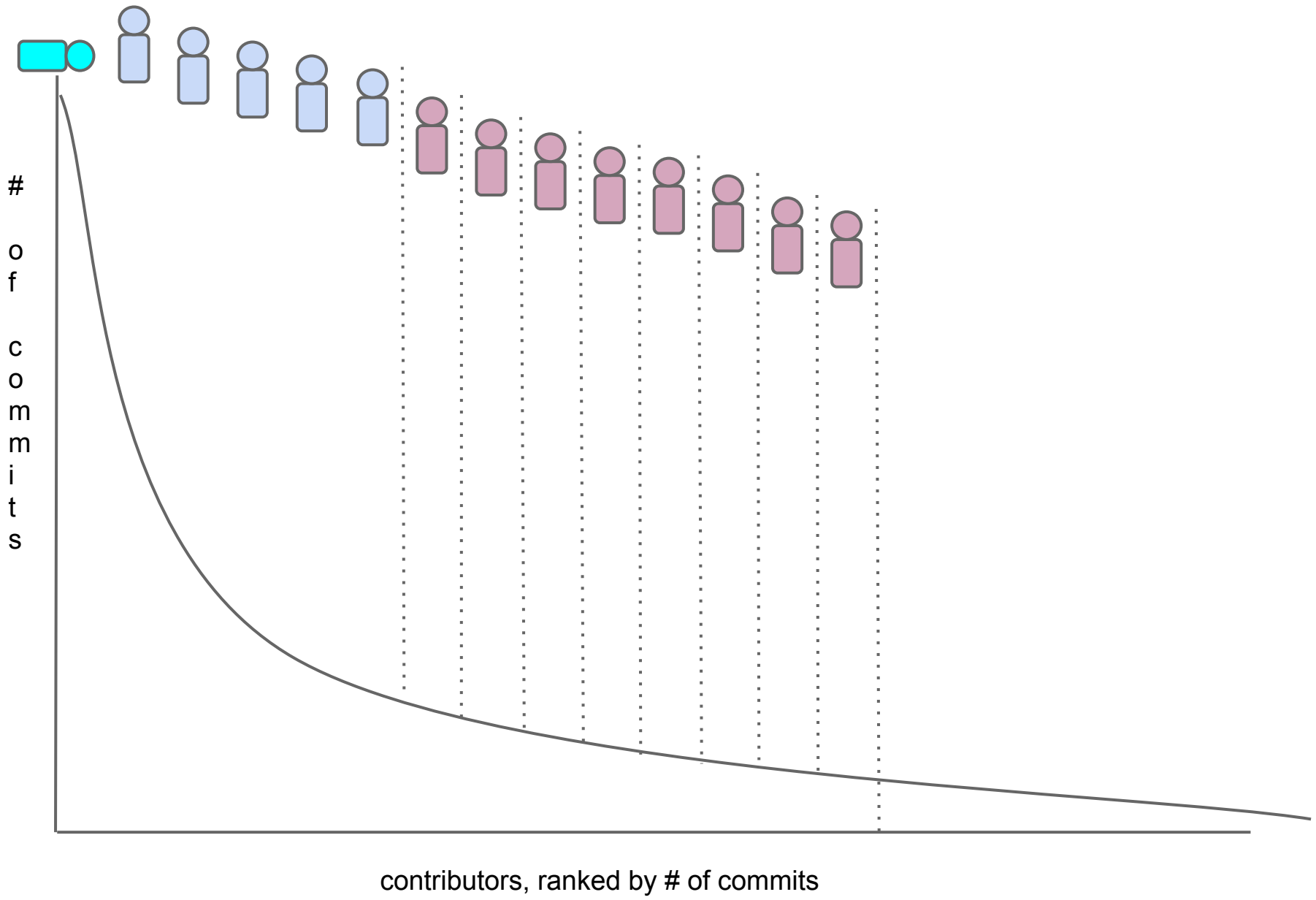

!
# of commits
contributors, ranked by # of commits

There are some problems with benevolent dictators


????????
# of commits
contributors, ranked by # of commits

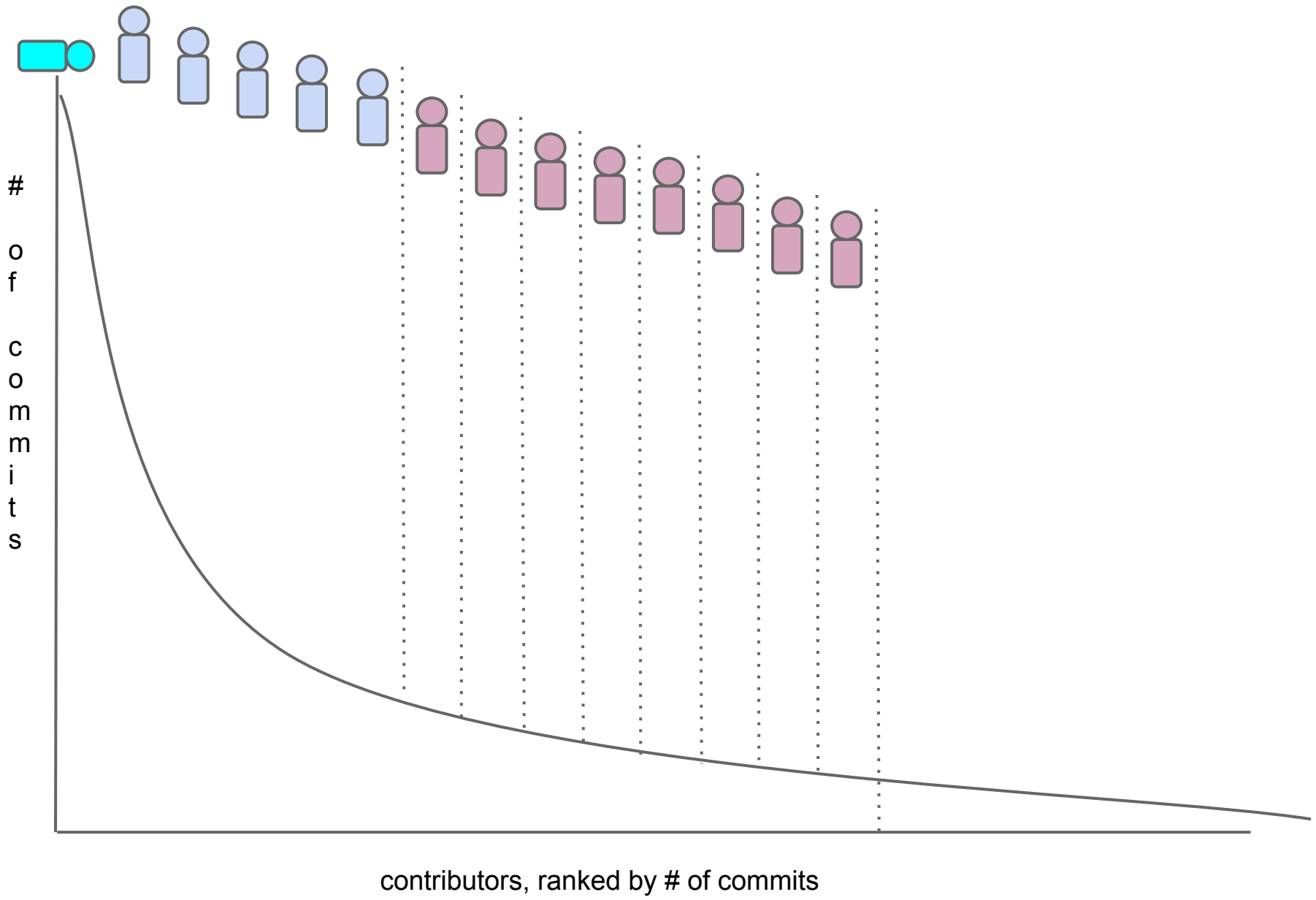

...
# of commits
contributors, ranked by # of commits

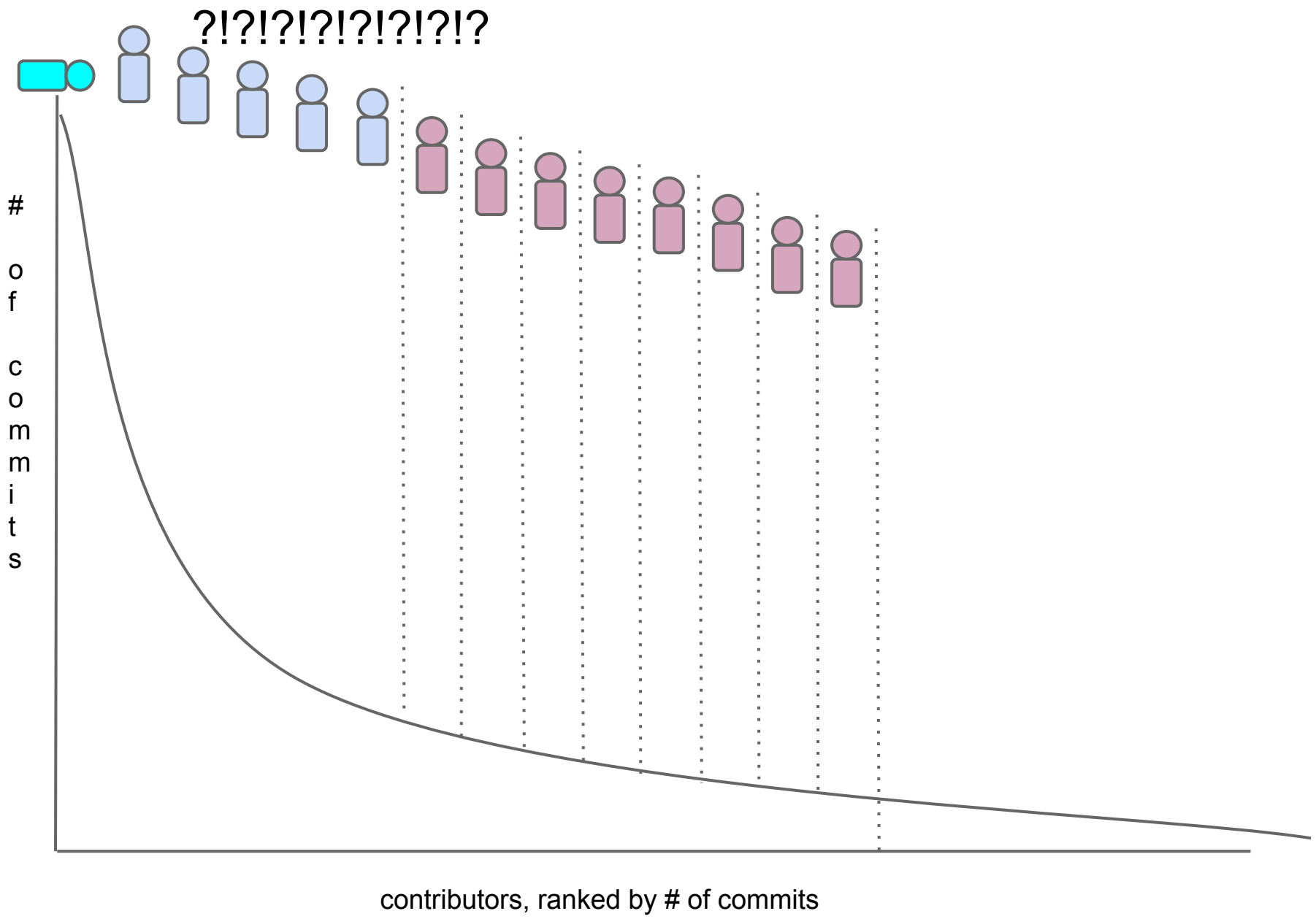

SCHOOL BUS
STOP
# of commits
contributors, ranked by # of commits


# of commits
contributors, ranked by # of commits

bus fac·tor

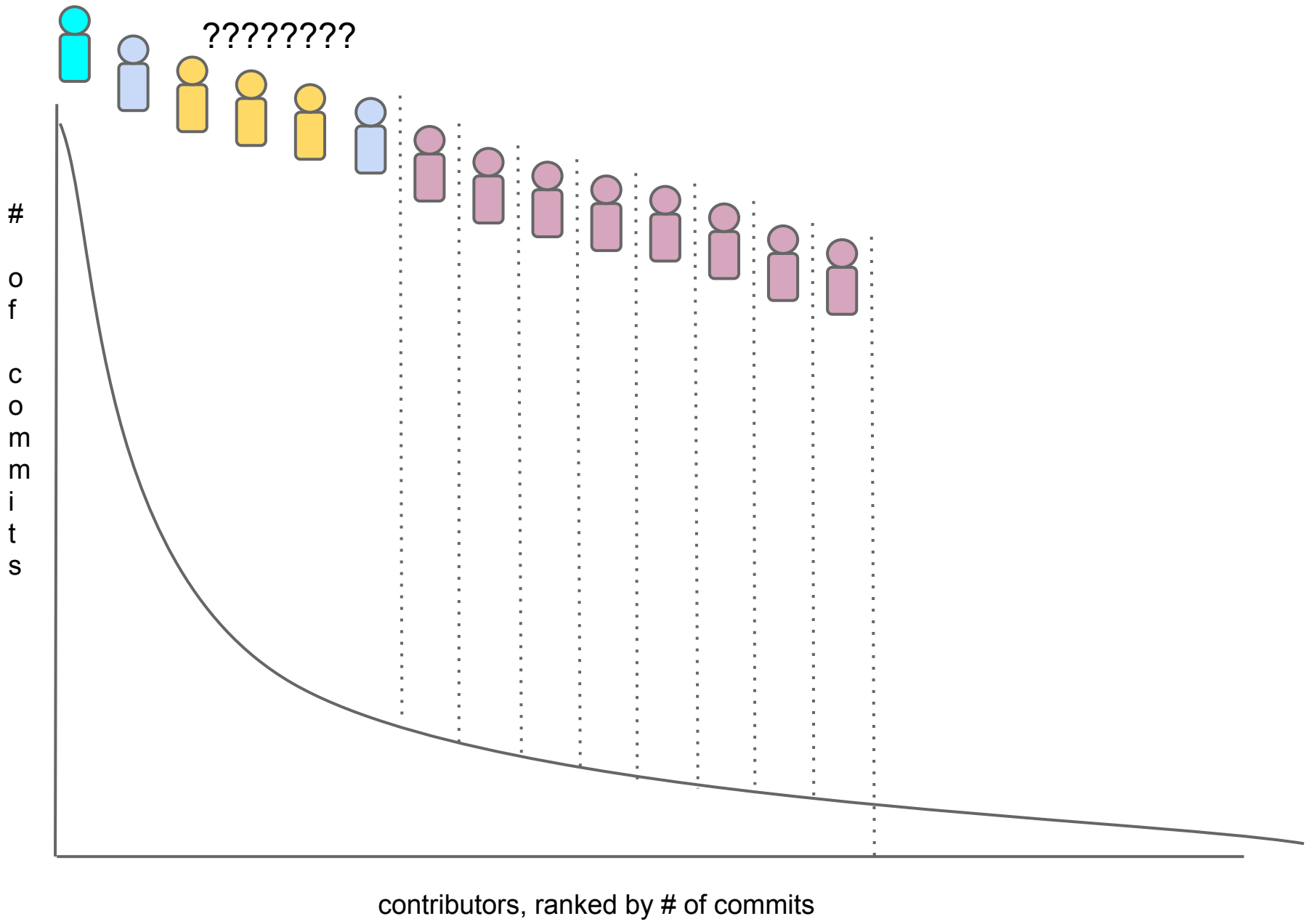
The bus factor is the total number of key developers who would need to be incapacitated (as by getting hit by a bus) to send the project into such disarray that it would not be able to proceed; the project would retain information with which no remaining team member is familiar.

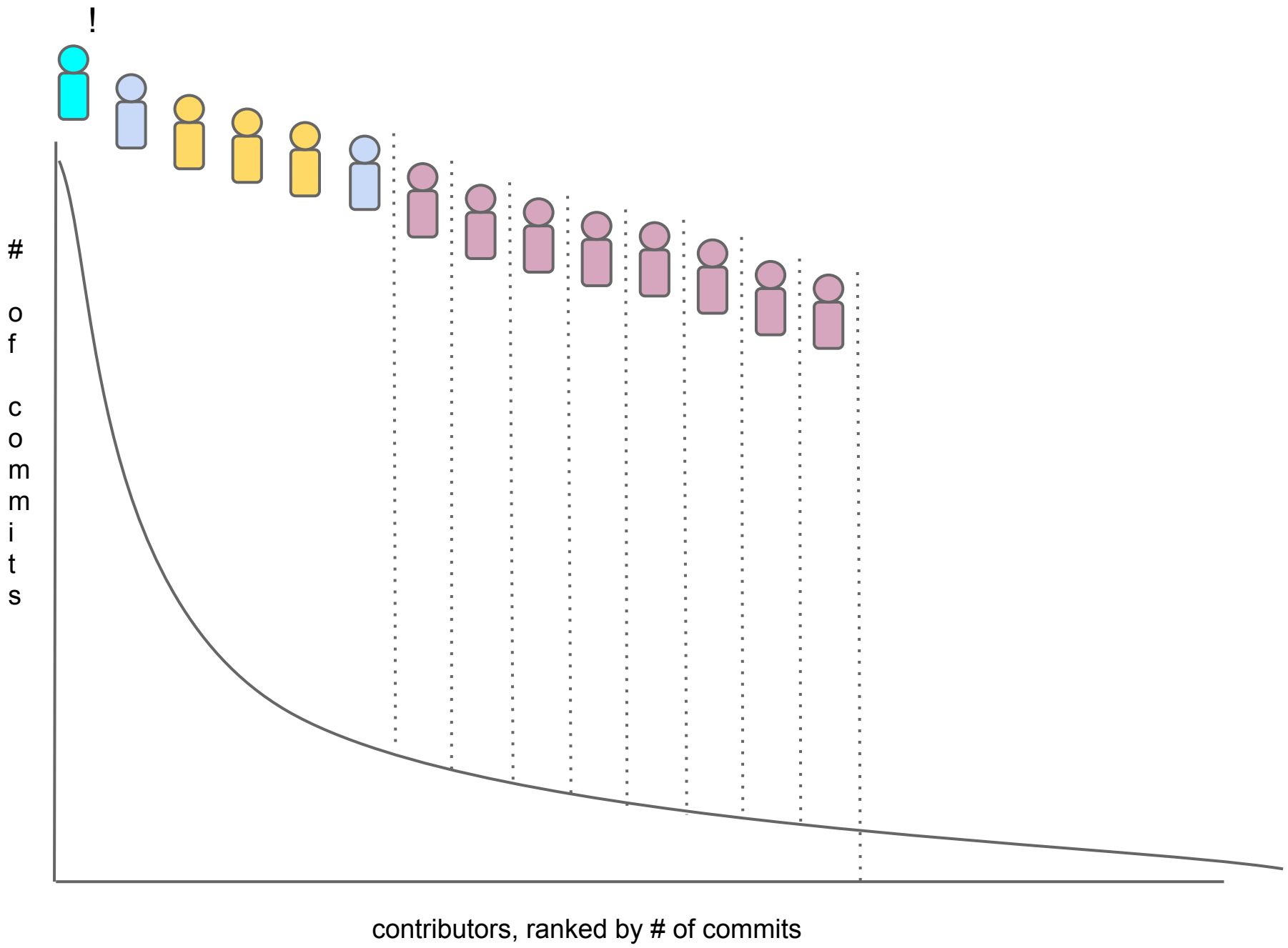

# of commits
contributors, ranked by # of commits

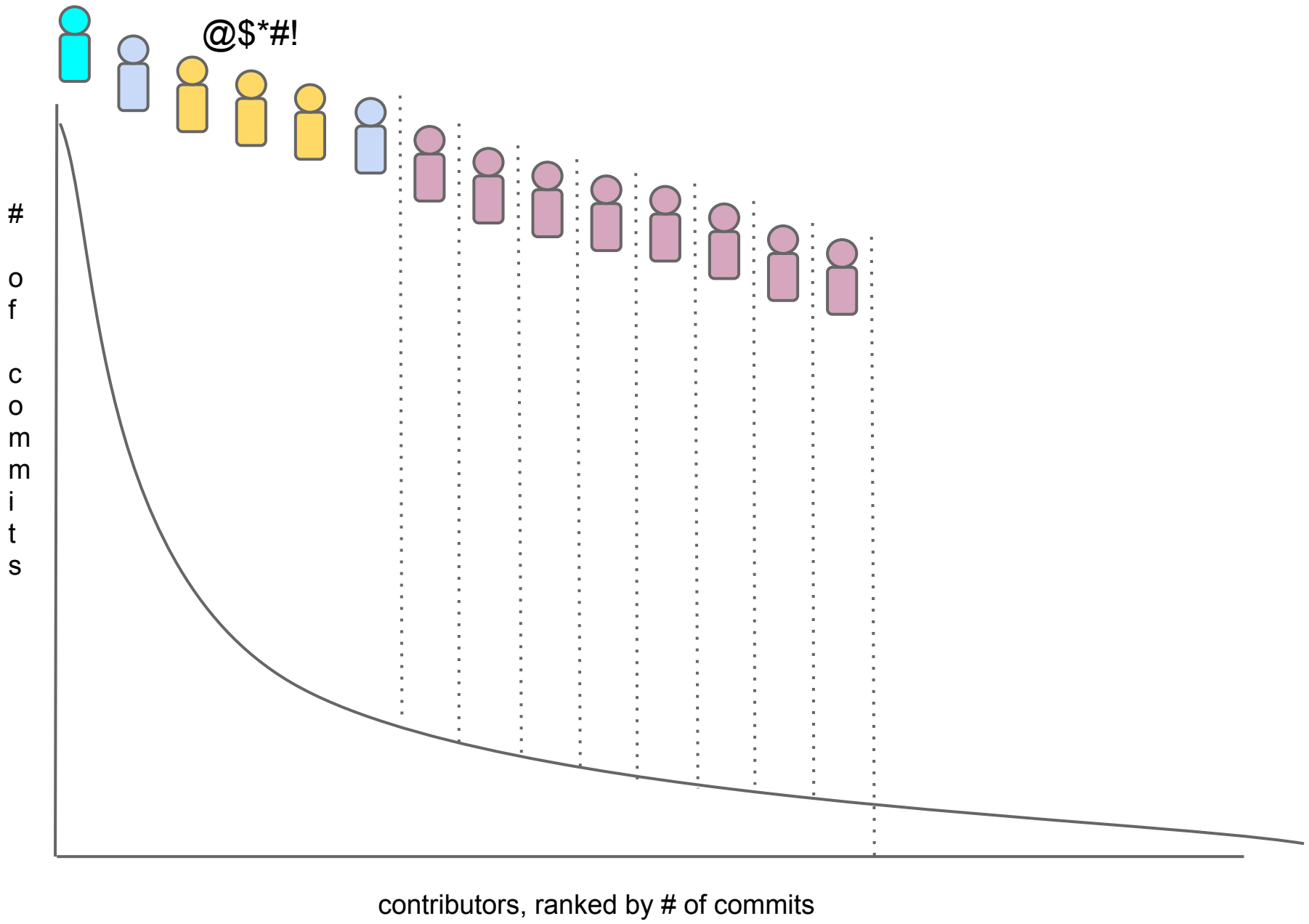

?!?!?!?!?!?!?!?
# of commits
contributors, ranked by # of commits

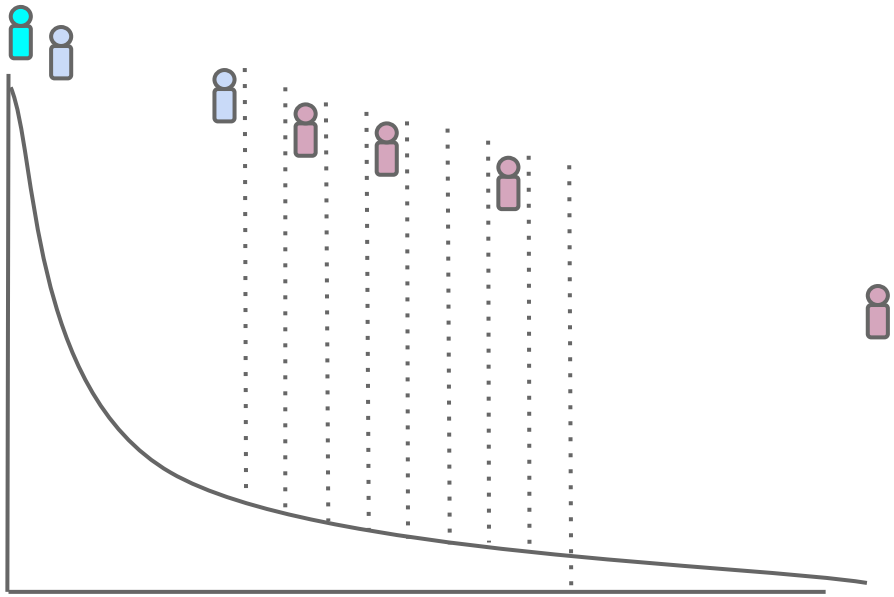
Another problem: factions

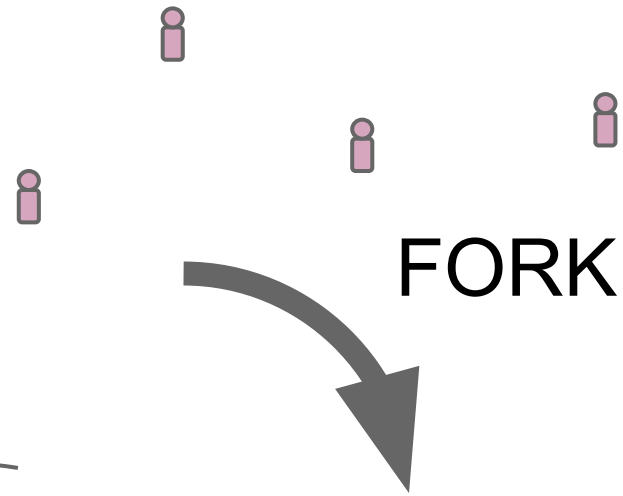
e.g. when two or more organizations pay developers on the project

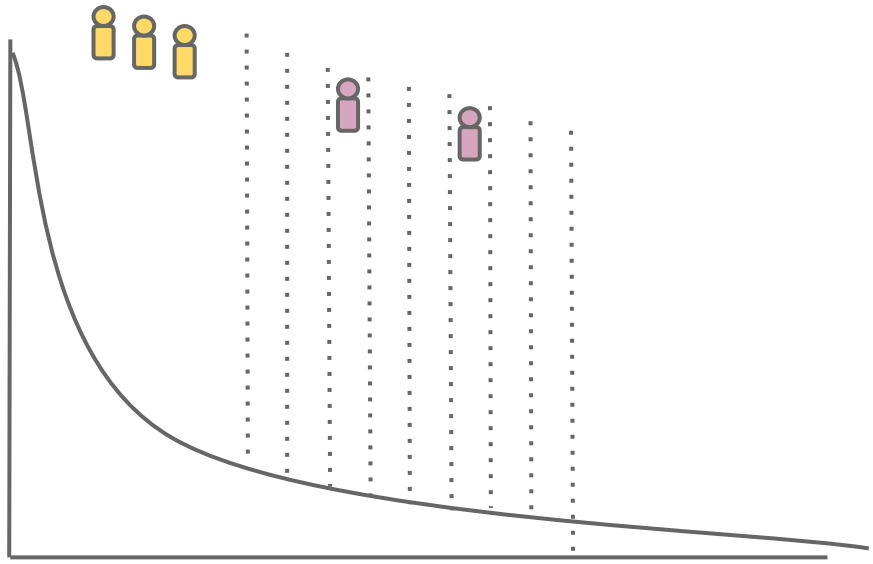

????????
# of commits
contributors, ranked by # of commits

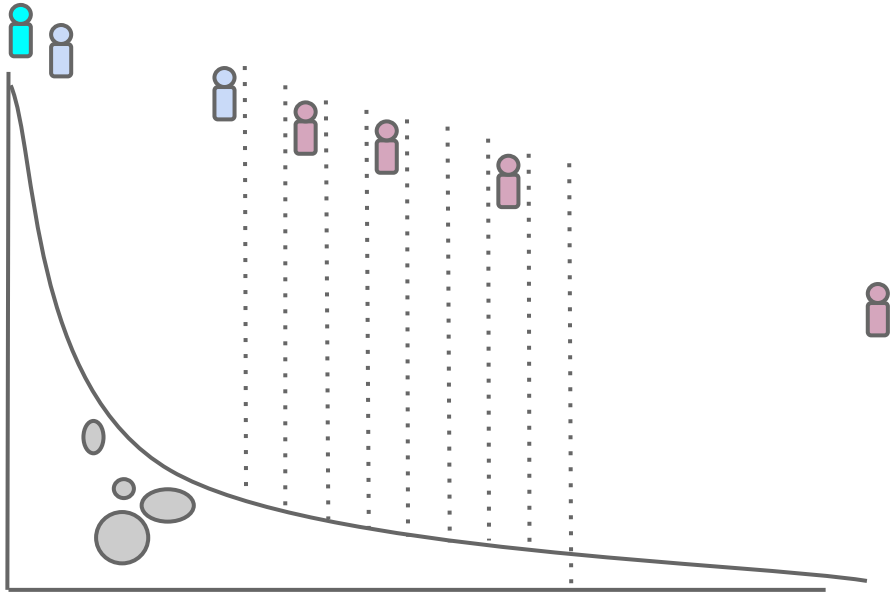

!
# of commits
contributors, ranked by # of commits

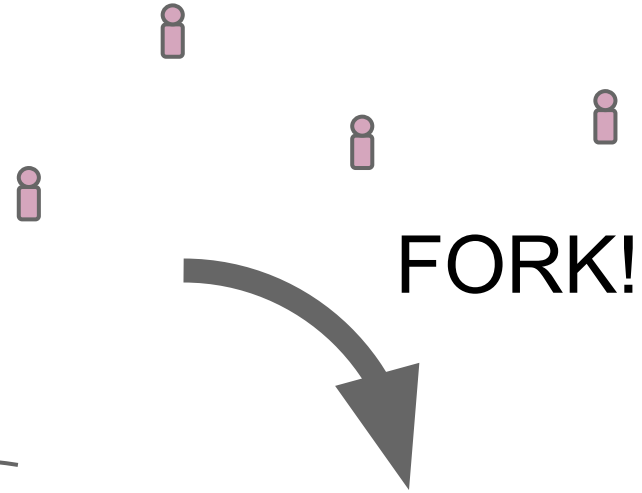

@$*#!
# of commits
contributors, ranked by # of commits



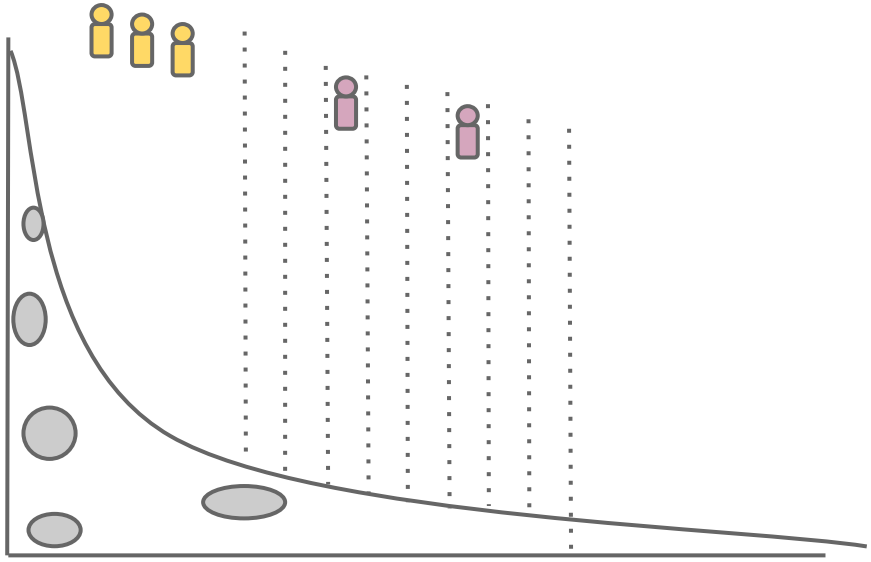

FORK!





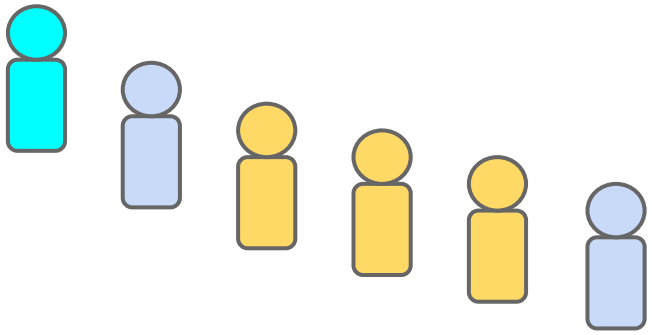


If you lose expertise, your community's code rots.

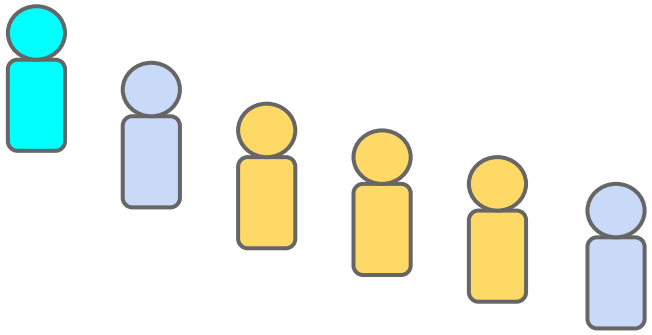


Forks are bad

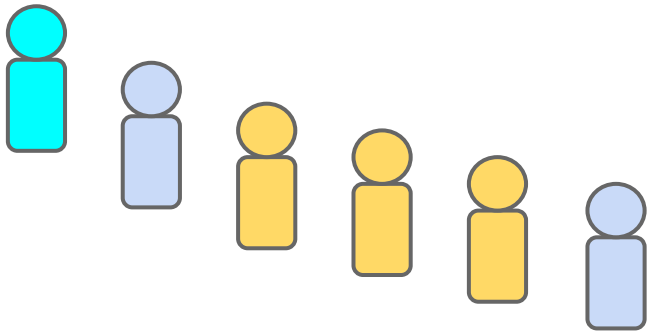
You need to make collective decisions



This is one of the hardest problems of humankind



Let’s get into the problem a little deeper



According to the licenses,
nobody is in charge

“free as in freedom”

structurelessness

the *tyranny* of *structurelessness*

“the *tyranny* of *structurelessness*”

Jo Freeman

Civil rights activist, attorney, writer, political scientist

Originally wrote under “movement name” Joreen

Argument 1

- Structureless is defended as a remedy to structural inequality/elitism
- There will be informal networks--”elites”--who are socially connected outside of the movement/project
- If there is no formal structure, these informal networks will have undue power
- So it is better to have explicit, formal structure to governance that promote equality

Argument 2

- Informal structures have secret power and secret criteria for inclusion.
- This makes it vulnerable to hijacking by other informal groups.
- It is better to formalize the informal structures and criteria.
- This opens the original structure up to formal challenge, which is both *more robust* and *more inclusive* than otherwise.

Principles

- *Delegation* by democratic procedure
- Authority => responsibility
- *Distribute* authority widely
- *Rotate* tasks
- *Allocate* tasks rationally; allow for *apprenticeship* to learn new skills
- *Diffuse information* to *everyone*
- *Equal access* to resources

Principles

- *Delegation* by democratic procedure
- Authority => responsibility
- *Distribute* authority widely
- *Rotate* tasks
- *Allocate* tasks rationally; allow for *apprenticeship* to learn new skills
- *Diffuse information* to *everyone*
- *Equal access* to resources

Somethings on-line communities get for free

Principles

- *Delegation* by democratic procedure
- Authority => responsibility
- *Distribute* authority widely
- *Rotate* tasks
- *Allocate* tasks rationally; allow for *apprenticeship* to learn new skills
- *Diffuse information* to *everyone*
- *Equal access* to resources

These are hallmarks of good project governance

Ok, so we need structure

Why don’t we just vote?

Voting is impossible

Kenneth Arrow

Nobel Prize winning economist

Very significant contributions to social choice theory and general equilibrium analysis.

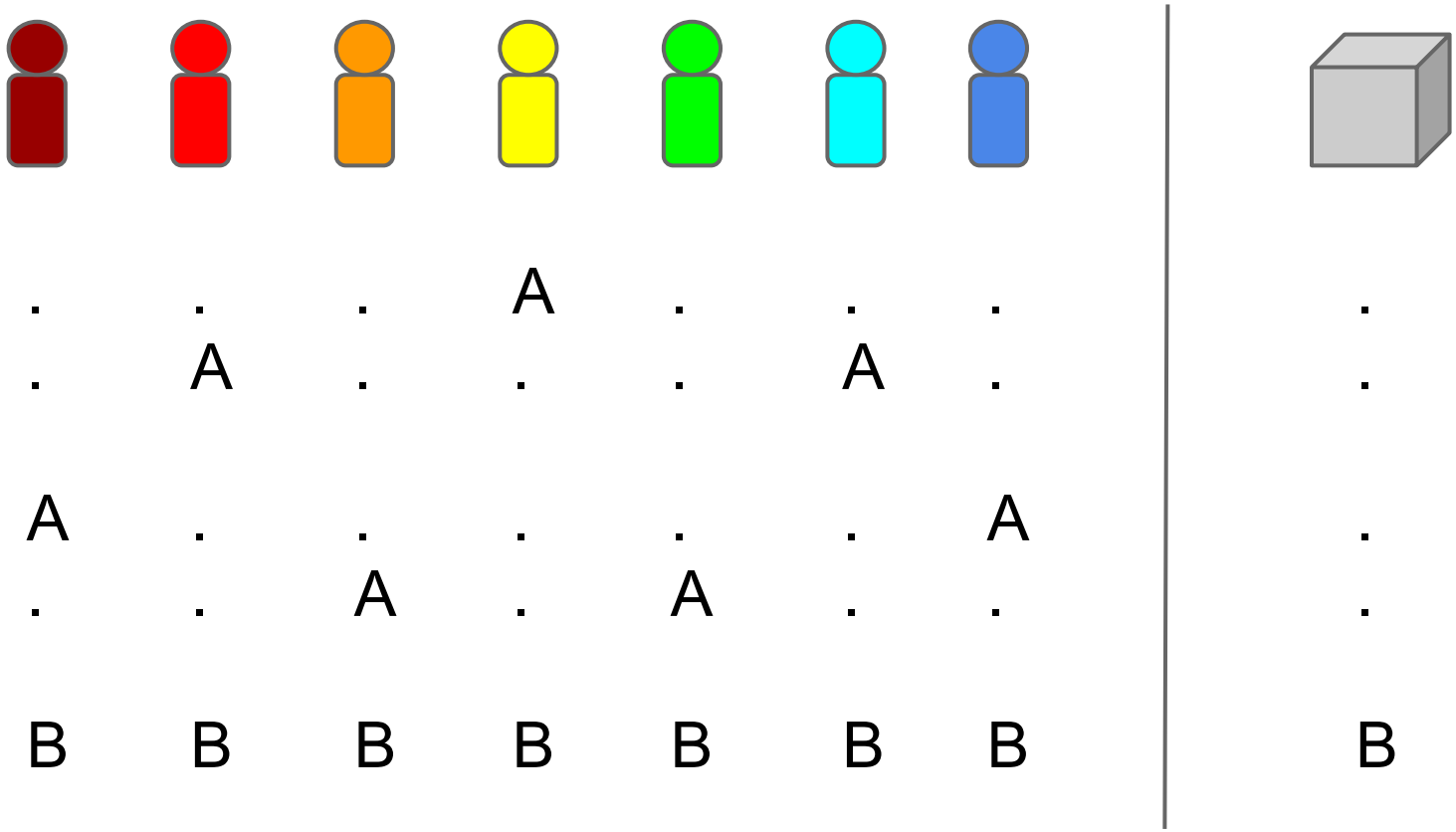
Arrow's Impossibility Theorem

There is no rank-order voting system that satisfies these criteria:

- If every voter prefers X to Y, then the group prefers X to Y
- If every voter's preference between X and Y is unchanged, the group's preference is unchanged
- No single voter can determine the group's preference (be "dictator")

from Wikipedia

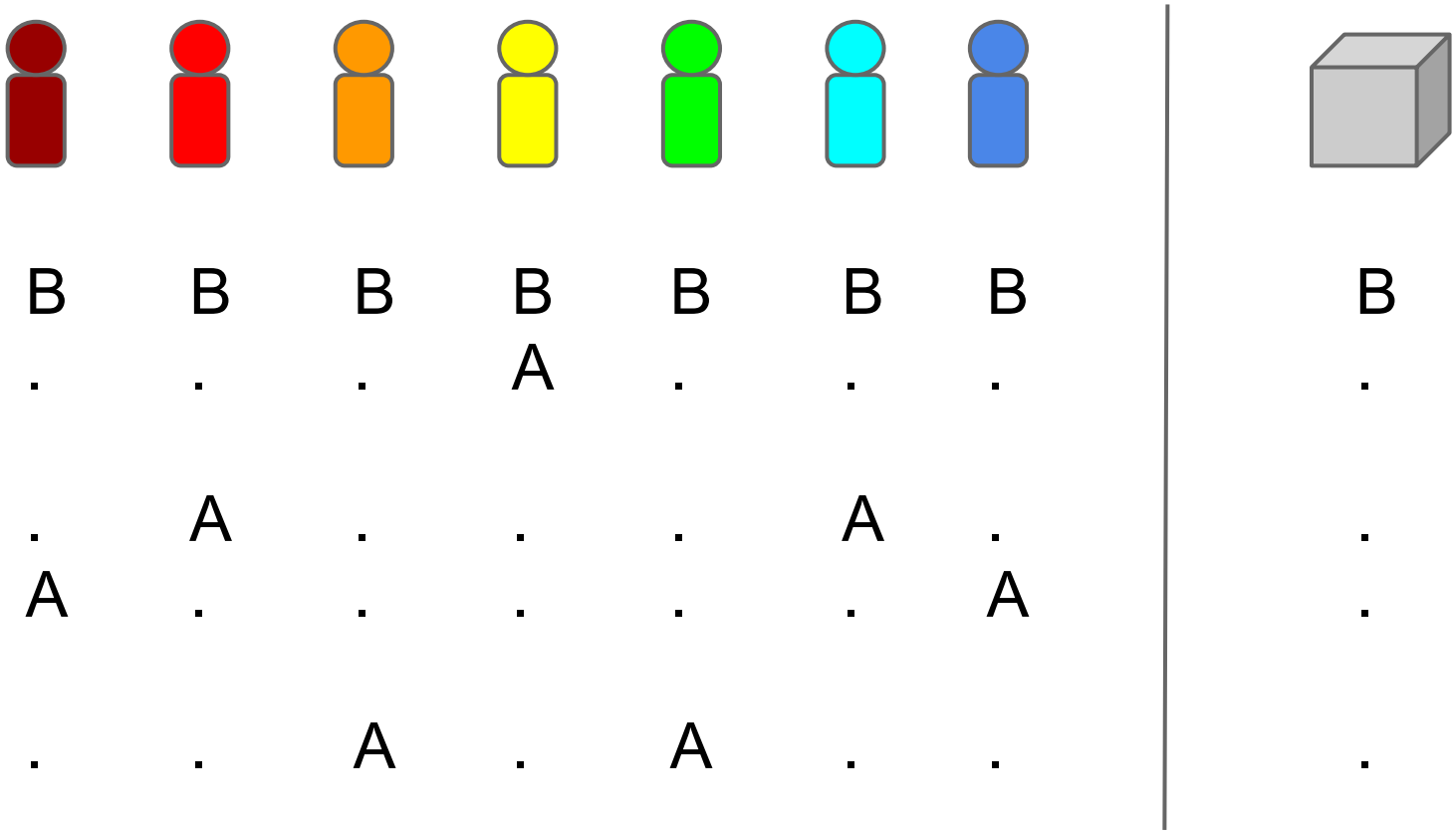
Arrow's Impossibility Theorem



Profile 0: Everyone hates option B

from Wikipedia

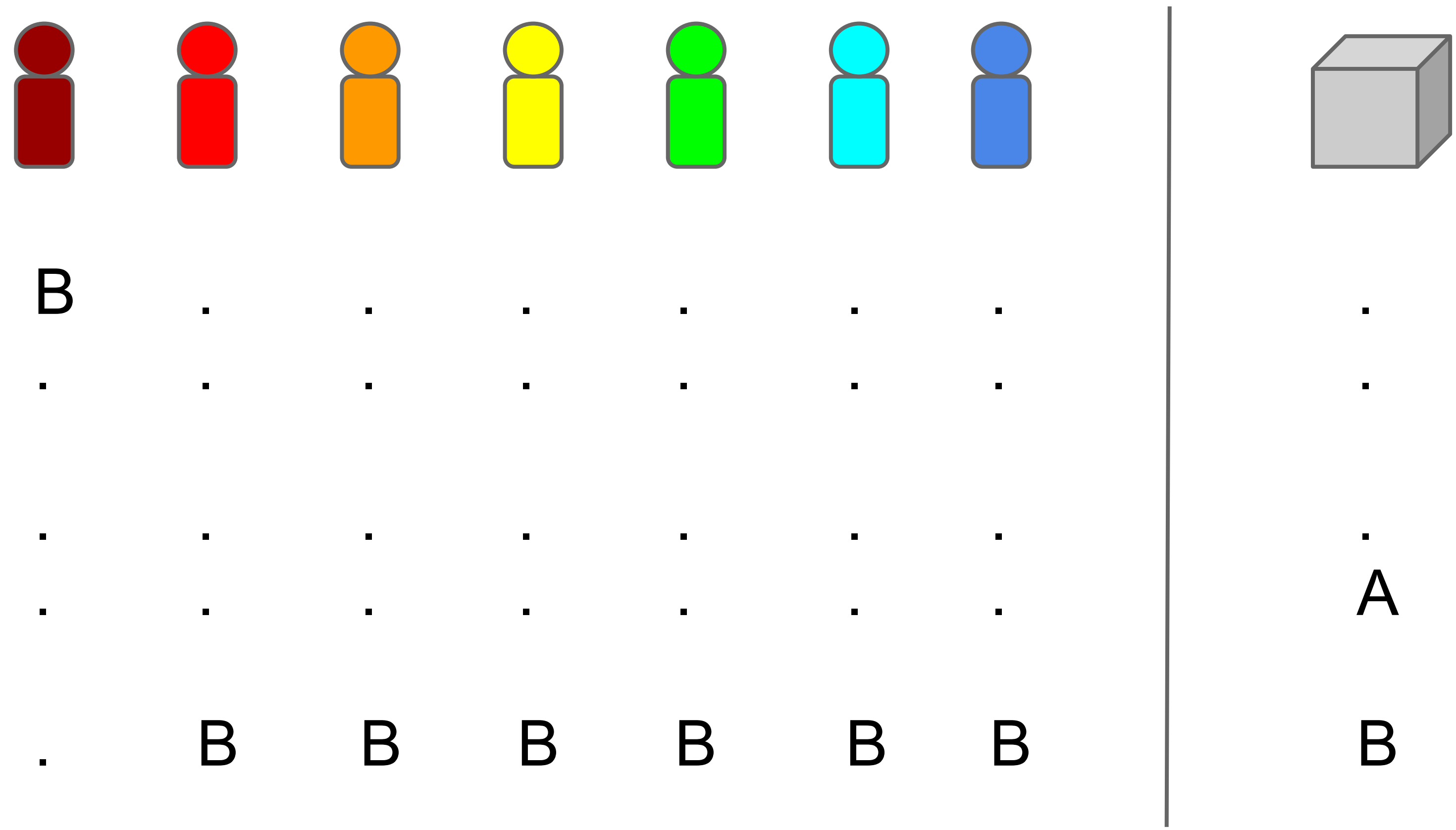
Arrow's Impossibility Theorem



Profile 0: Everyone loves option B

from Wikipedia

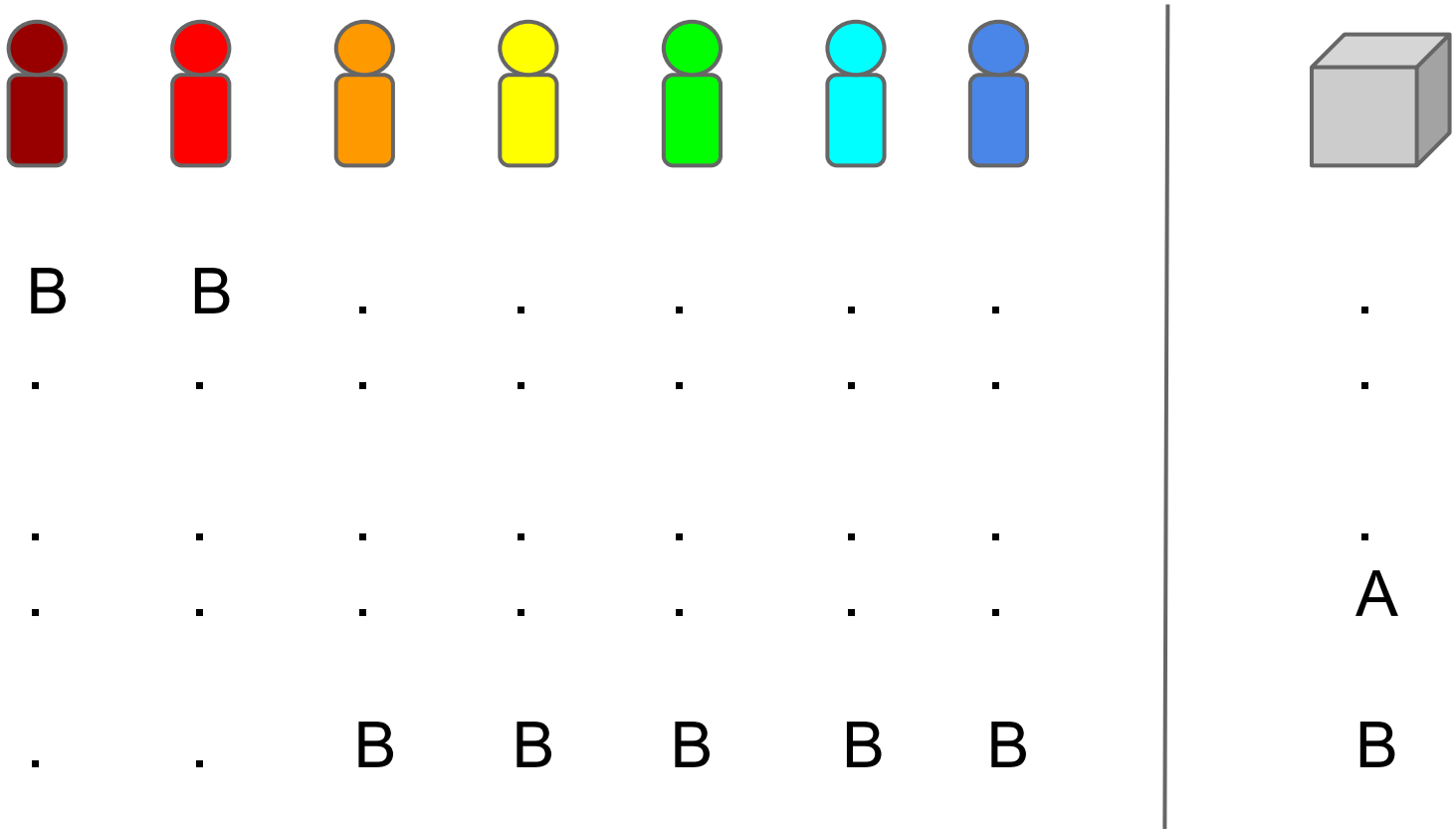
Arrow’s Impossibility Theorem



Profile 1: Only voter 1 loves B

from Wikipedia

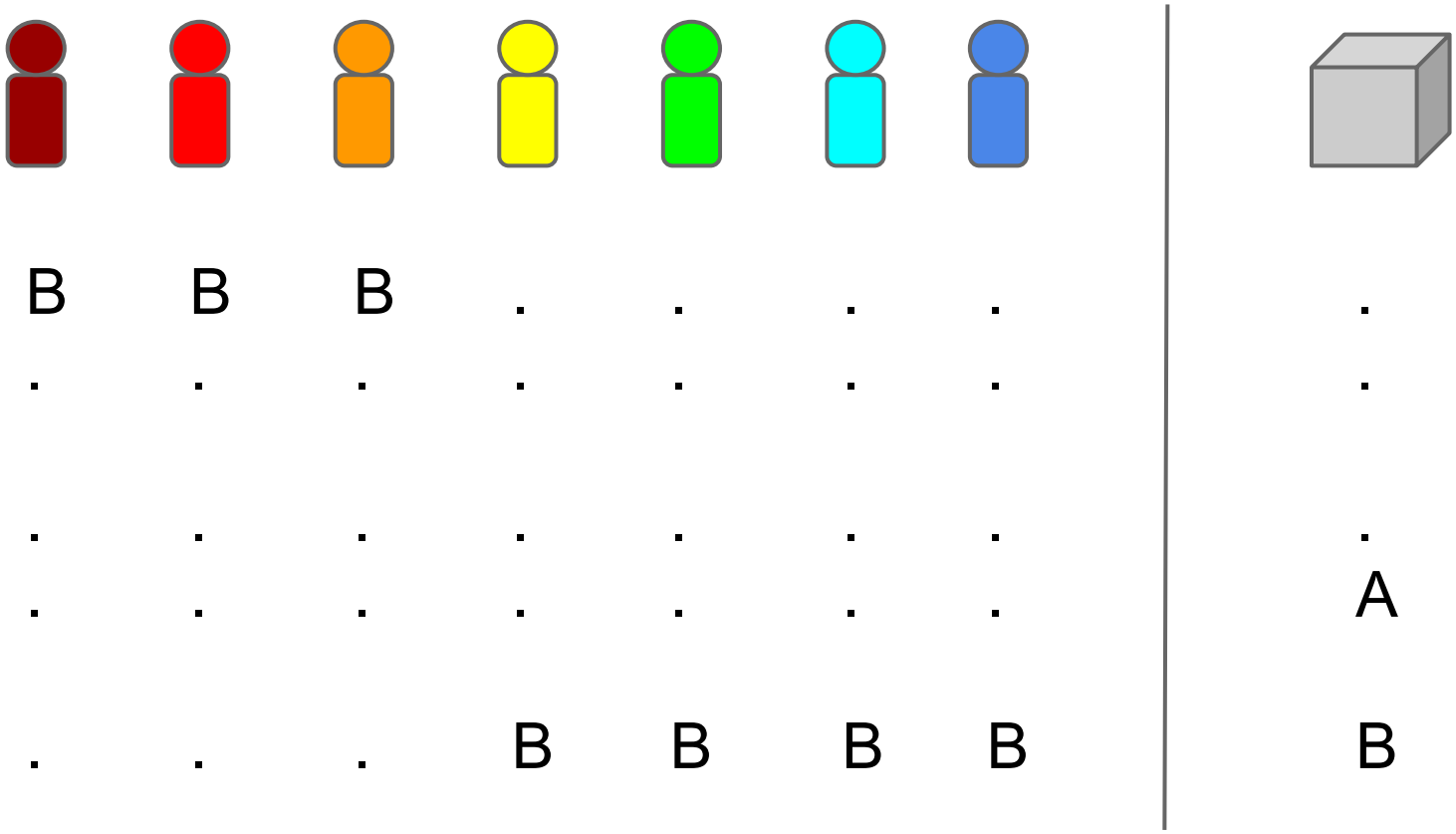
Arrow's Impossibility Theorem



Profile 2: Only voters 1 and 2 love B

from Wikipedia

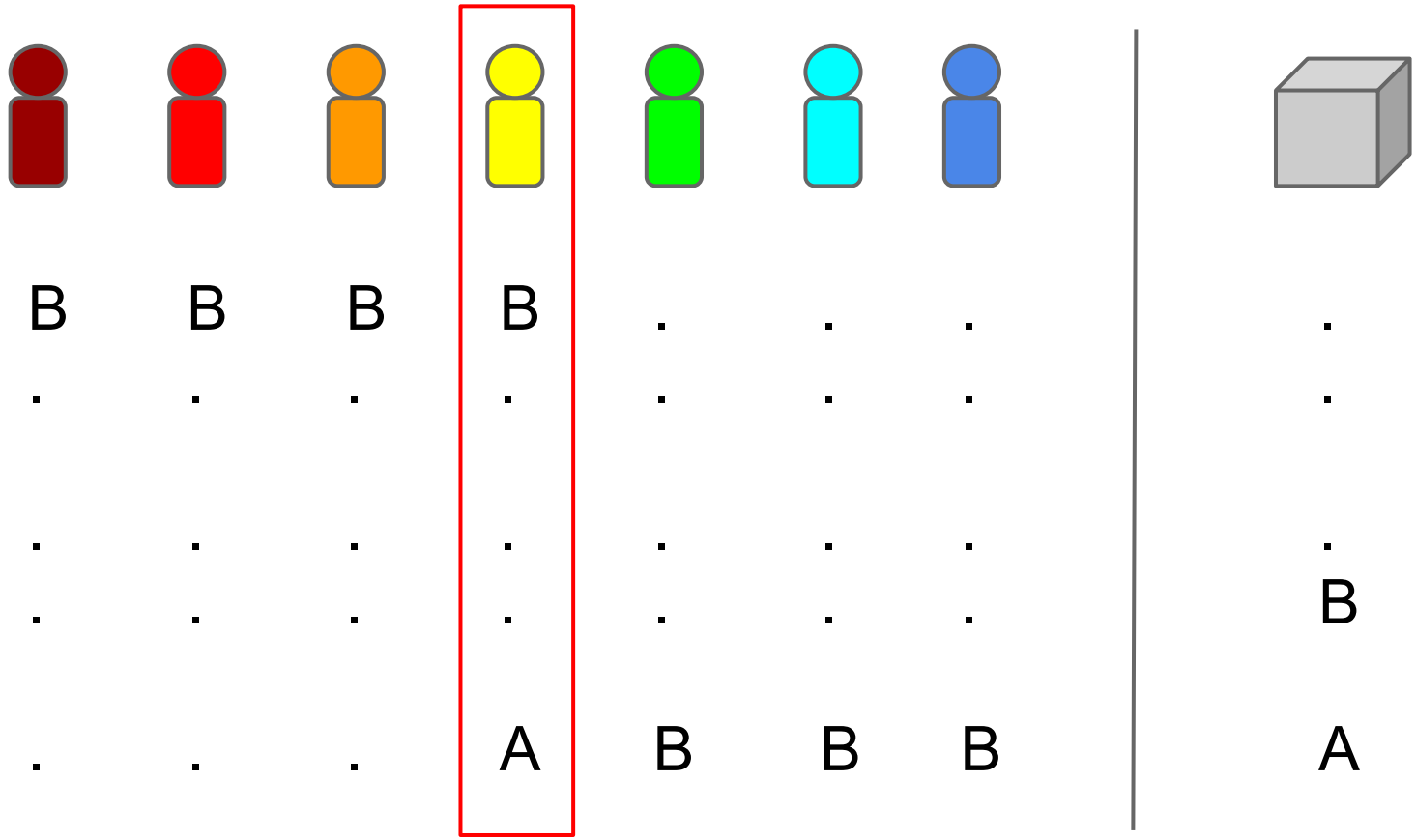
Arrow's Impossibility Theorem



Profile i: Only voters 1 through i love B

from Wikipedia

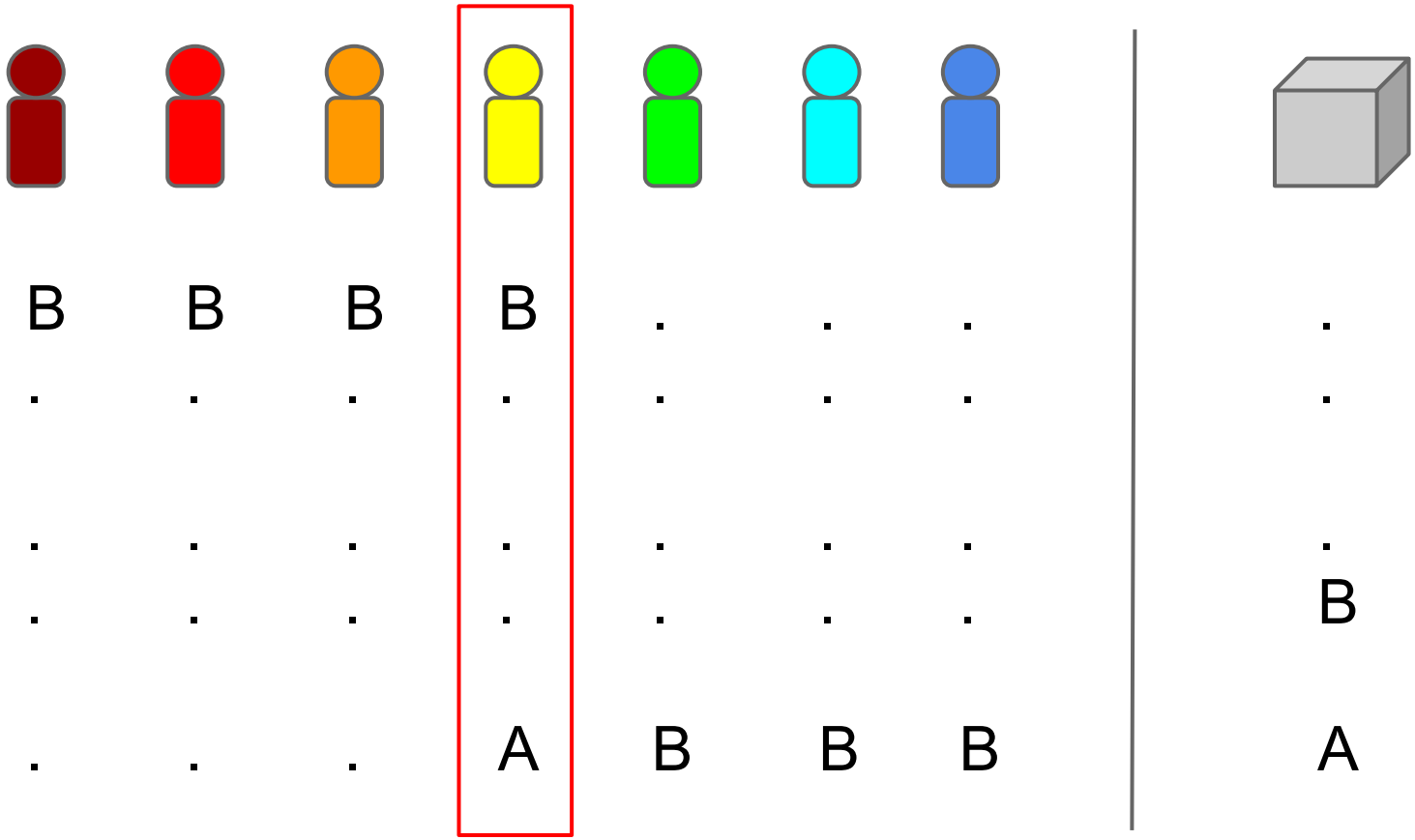
Arrow's Impossibility Theorem



There's got to be some k where B > A.

from Wikipedia

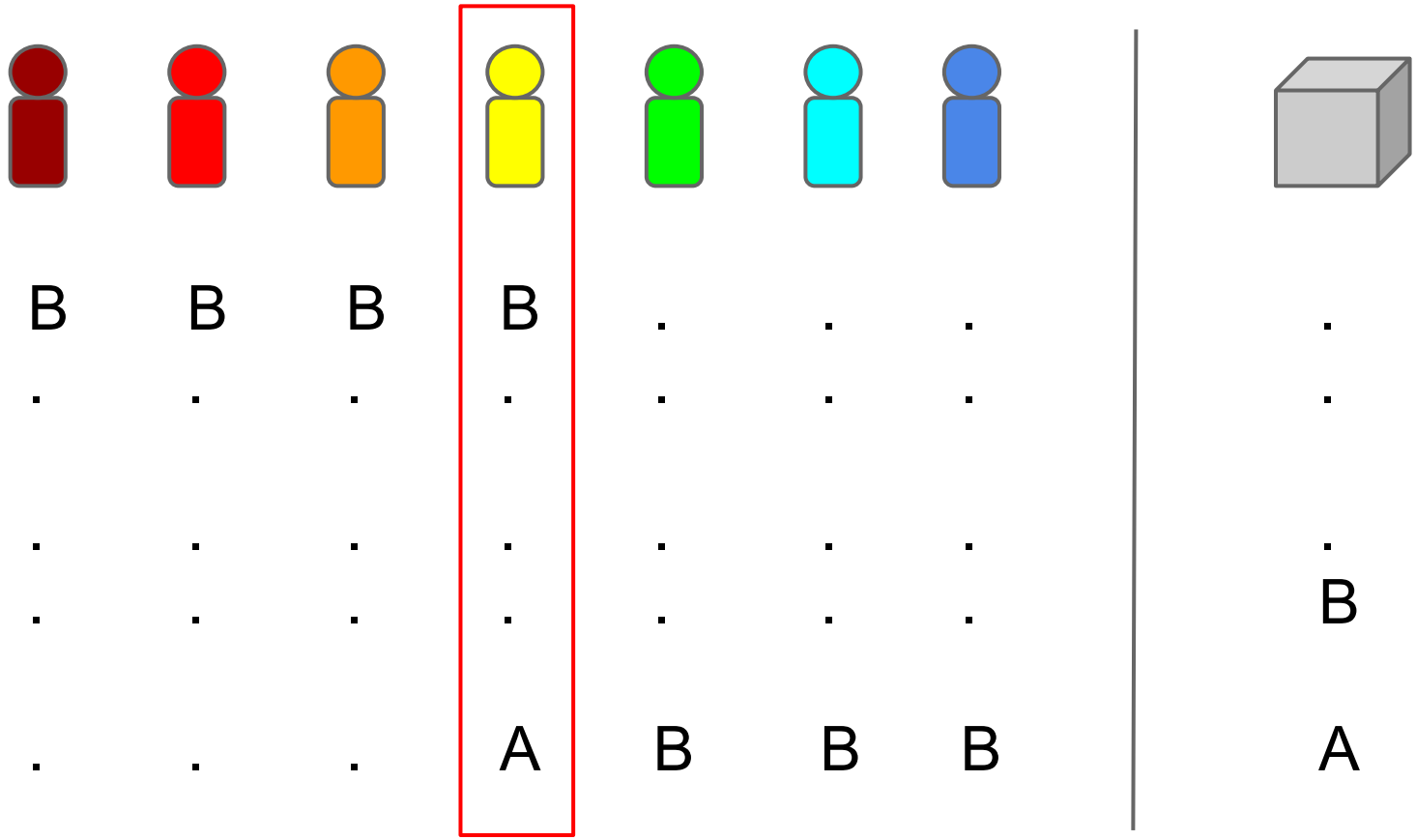
Arrow's Impossibility Theorem



This is the "pivotal voter"

from Wikipedia

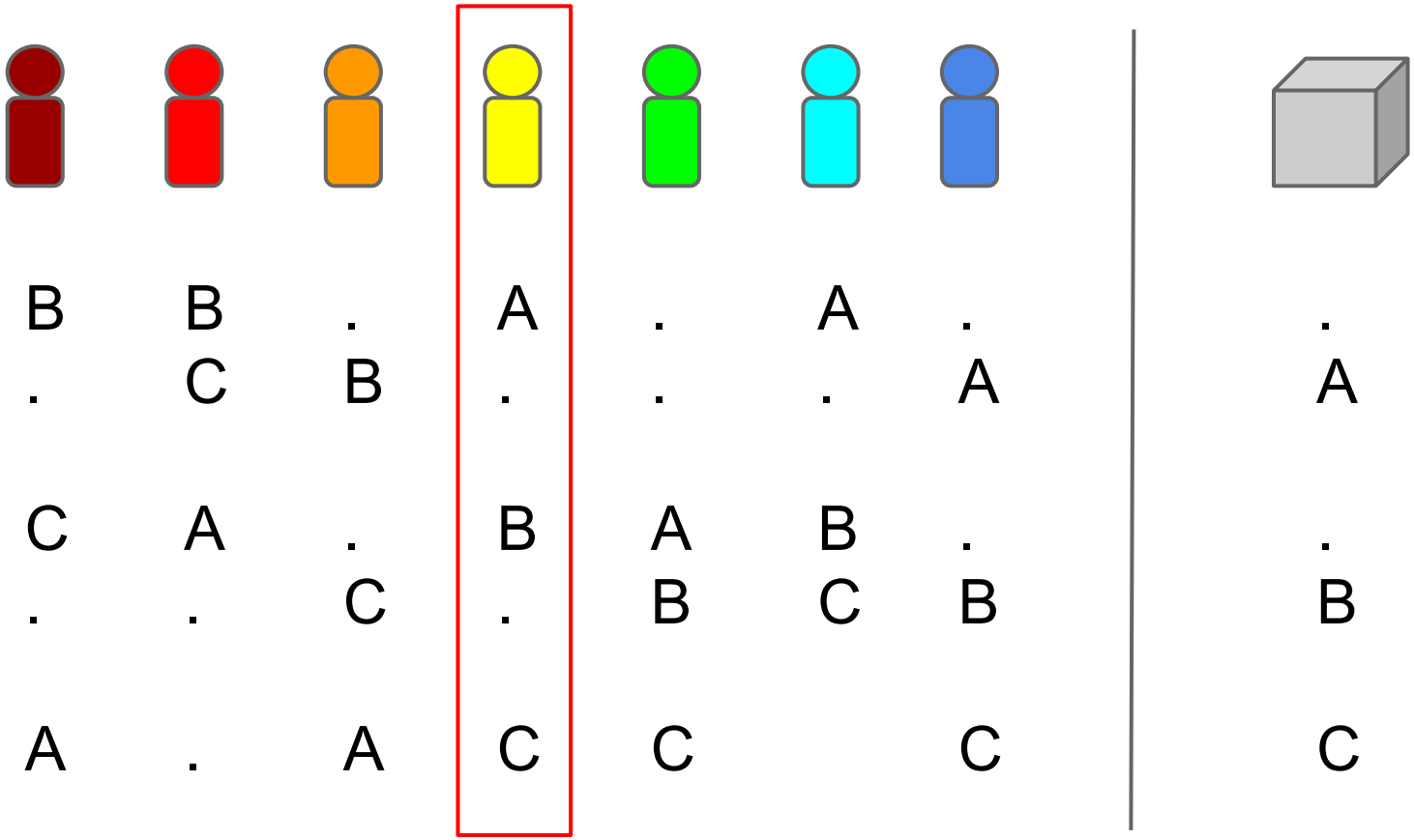
Arrow's Impossibility Theorem



They have great power.

from Wikipedia

Arrow's Impossibility Theorem

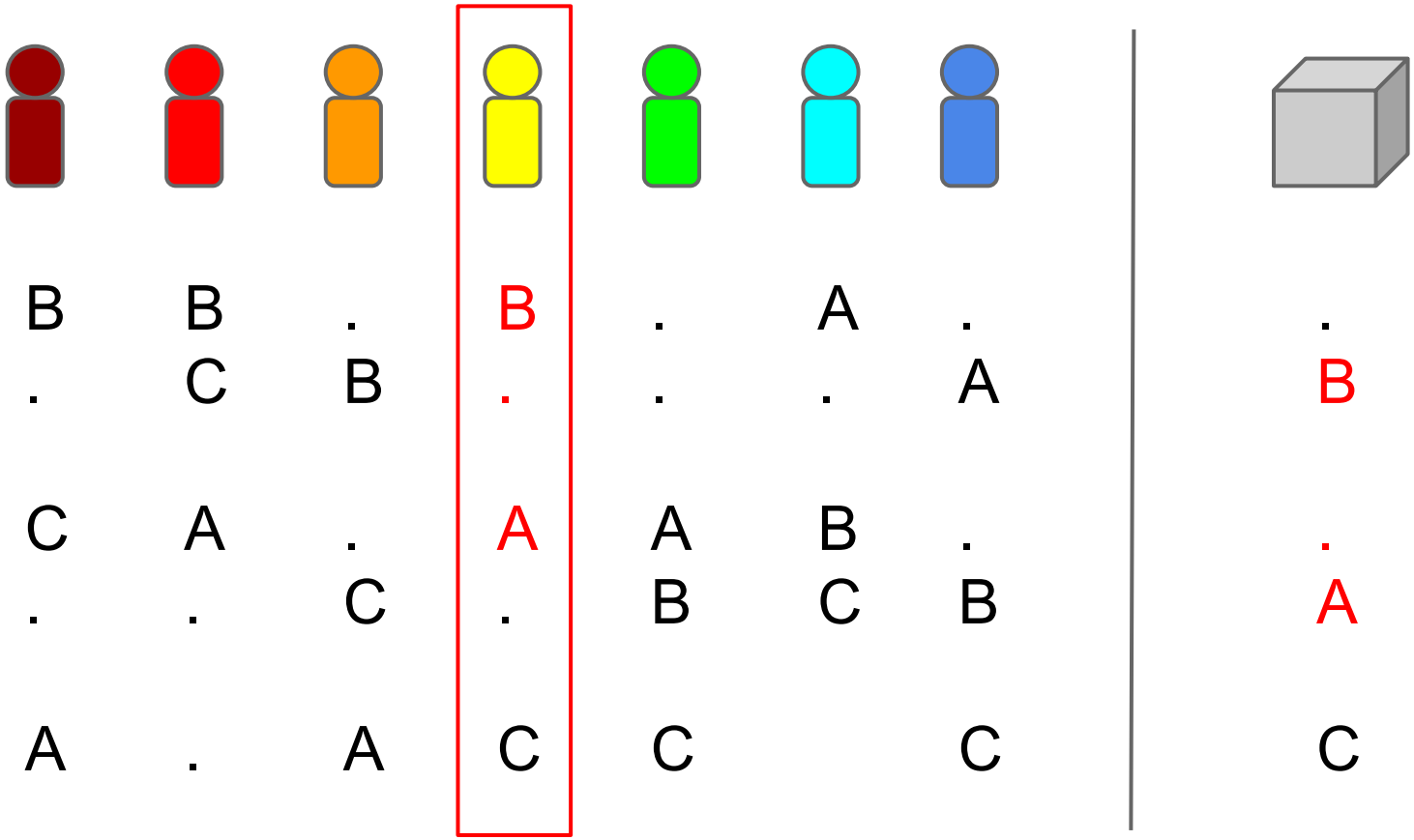


B	B	.	A	.	A	.	.
.	C	B	.	.	.	A	A
C	A	.	B	A	B	.	.
.	.	C	.	B	C	B	B
A	.	A	C	C		C	C

Here they have decided to rank B < A.

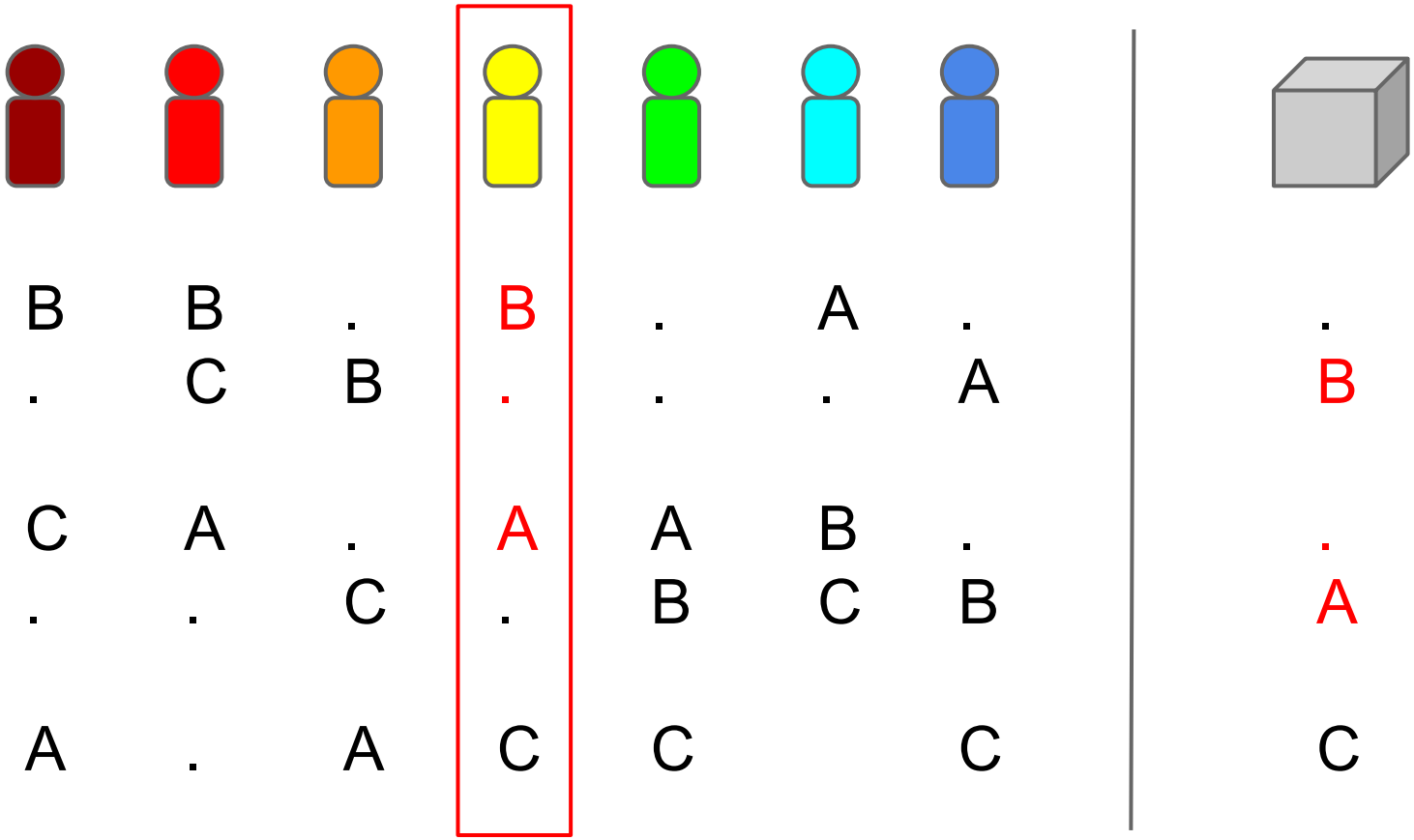
from Wikipedia

Arrow's Impossibility Theorem



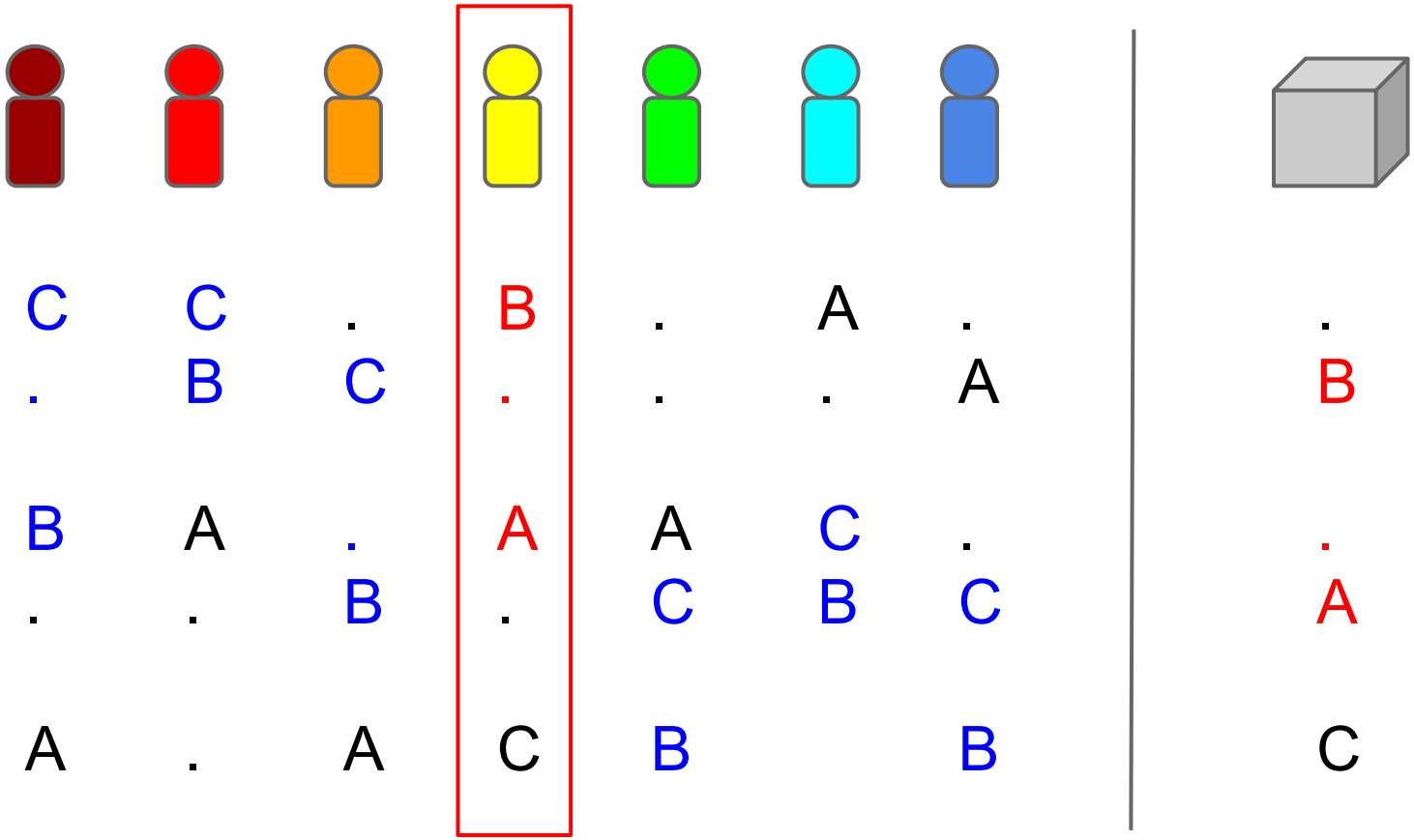
Now they change their mind

Arrow's Impossibility Theorem



Now they change their mind

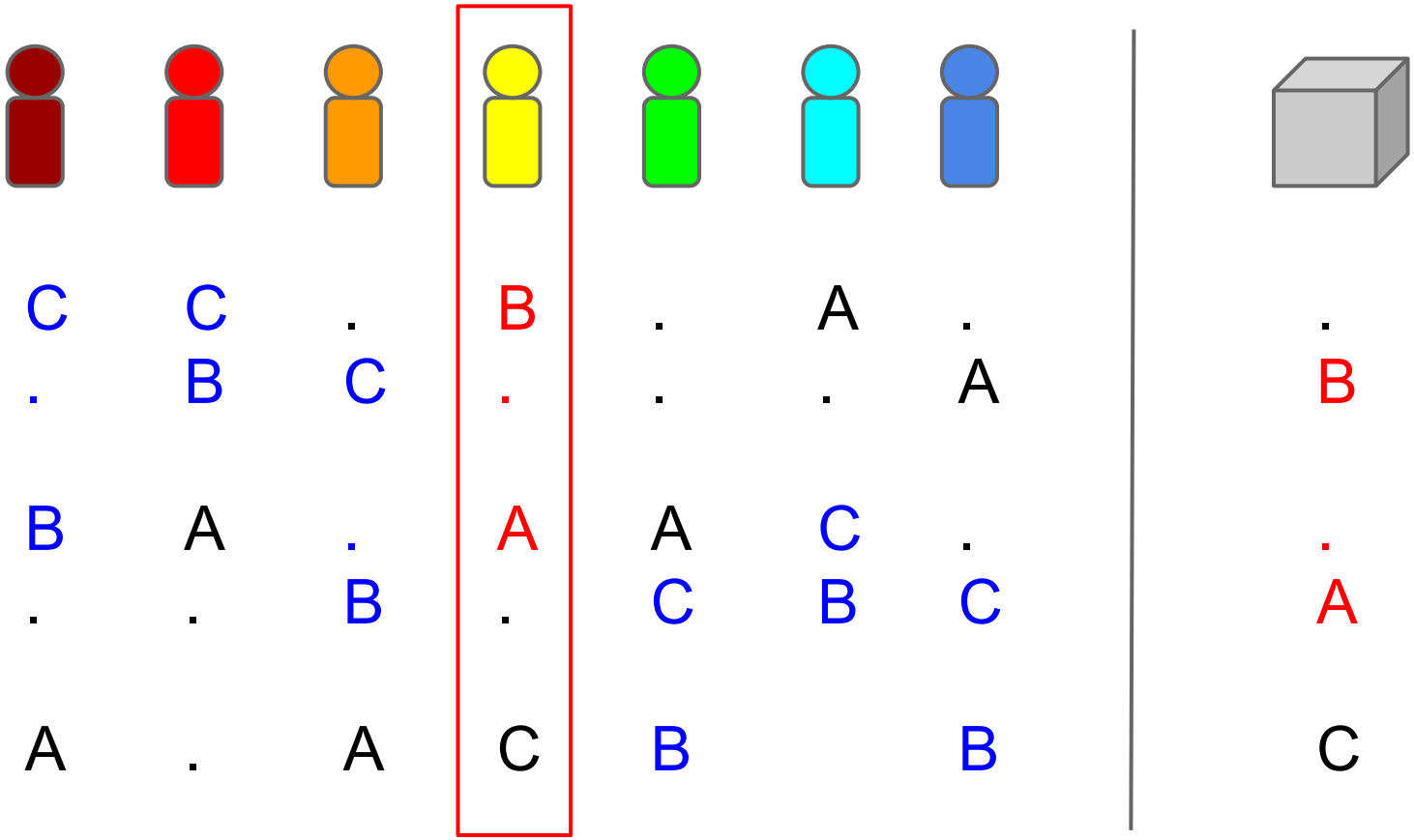
Arrow's Impossibility Theorem



Everybody else can think C > B

from Wikipedia

Arrow's Impossibility Theorem



But B > A > C, the pivotal voter still controls

from Wikipedia

This just scratches the surface of a fascinating body of literature proving the impossibility of fair elections

FOSS communities have converged on structure/procedure

Apache Consensus Voting

- Voting: +1, 0, -1
- “Lazy consensus” -- silence is consent
- Voting implies responsibility: +1 = “I’ll help” or “I reviewed it myself”
- Aim for consensus always, majority vote as a last resort

Who can vote?

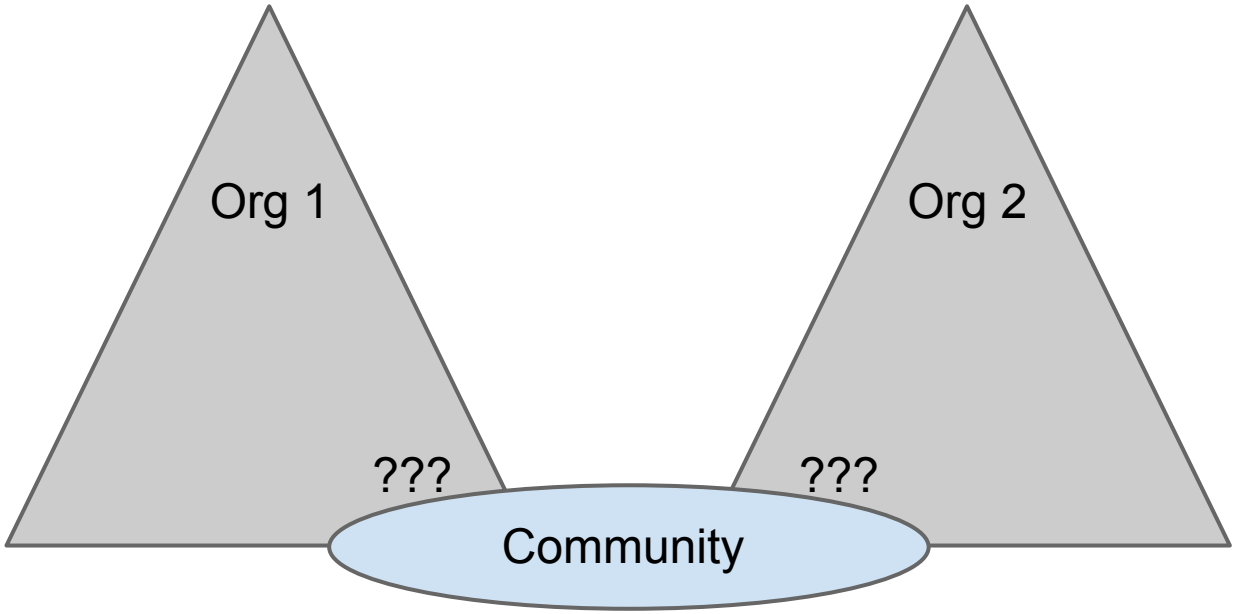
- Often: all committers
- Sometimes: A select group, the Project Steering Committee, balanced to represent stakeholders
- Vote in new members in secret, rotate members from PSC and chair positions to keep things balanced

Benefits of this approach

- Often fast resolution to easy problems
- Engages creative dialog during disagreement
- Privileges actionable proposals: people propose to *do* something
- Maintains buy-in from community members

Issues with this approach

- Can take time
- Difficult to reconcile with an external organizational management structure



Addenda

- Governance takes place within the cyberspaces of the community infrastructure--mailing lists, issue trackers, IRC, etc.
- We are in an early experimental stage of on-line project governance
- Potential for building better institutional patterns and technologies for democracy

Addenda

- There is no global “movement” governance, yet
- Much of the leadership takes place within organizations--businesses, foundations, non-profits--with their own management hierarchy
- Money is power

Next:
How shall ye govern thyselves?