

Will AI R&D Automation Cause a Software Intelligence Explosion?

Daniel Eth & Tom Davidson

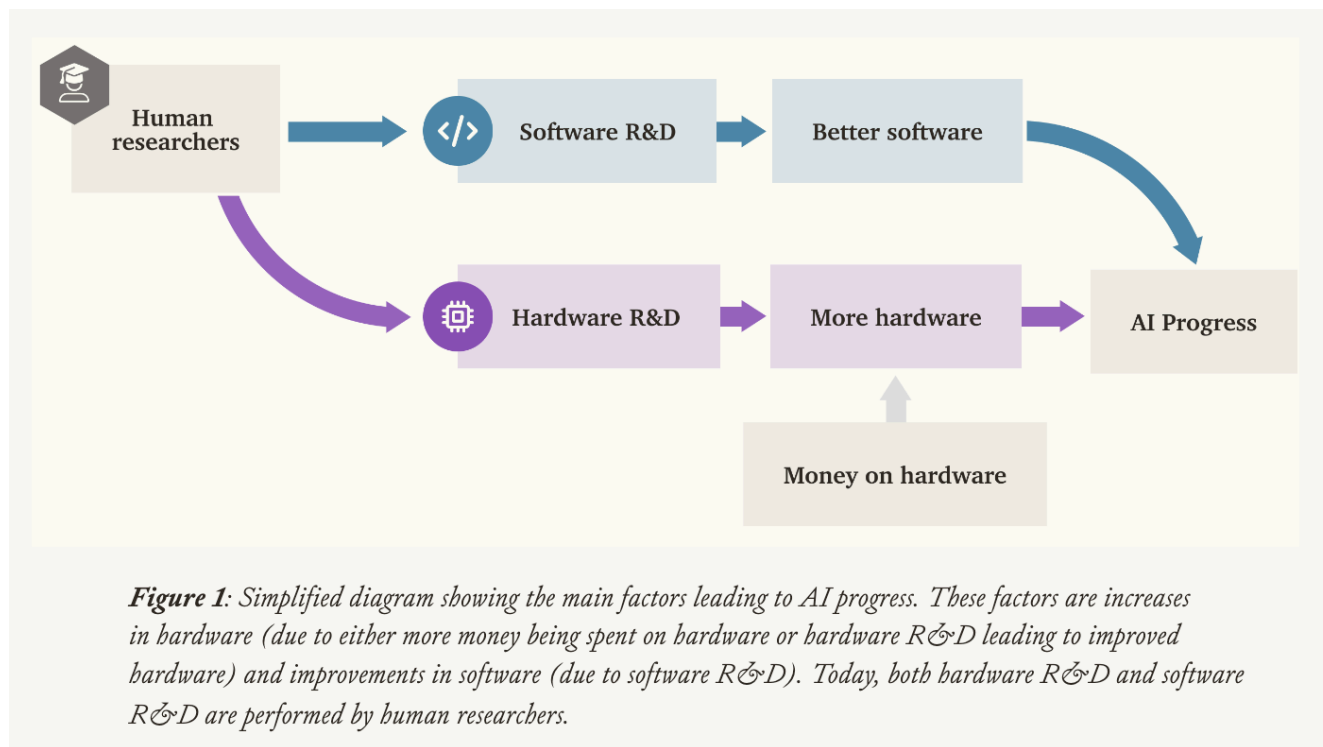
26th March 2025

<https://www.forethought.org/research/will-ai-r-and-d-automation-cause-a-software-intelligence-explosion>

Slides created by Yegon Kim
Presented on 2026/10/06

Question

- **After AI research becomes fully automated**, can software progress keep accelerating?
- Fixed stock of computing hardware (hypothetical scenario)
- **ASARA**: AI Systems for AI R&D Automation



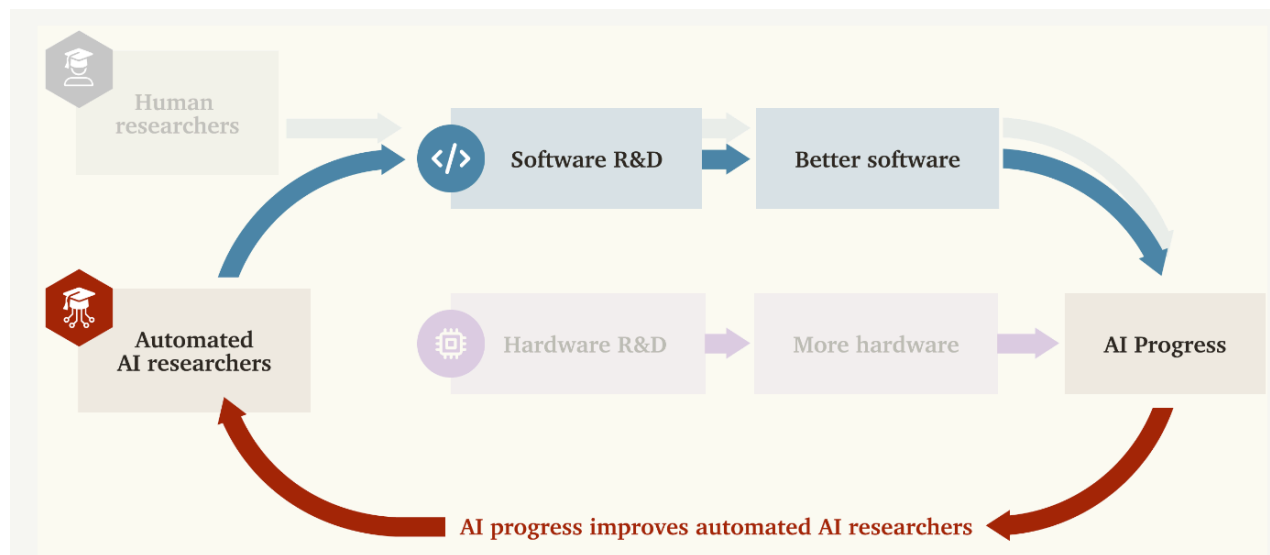


Figure 2: Diagram similar to Figure 1, but modified in two ways for an SIE. First, the hardware path is nixed, allowing for examining a scenario where only software improves. Second, ASARA systems replace human researchers in performing R&D, yielding faster AI progress, which further enhances ASARA systems' abilities, enabling them to conduct even better and/or faster software R&D.

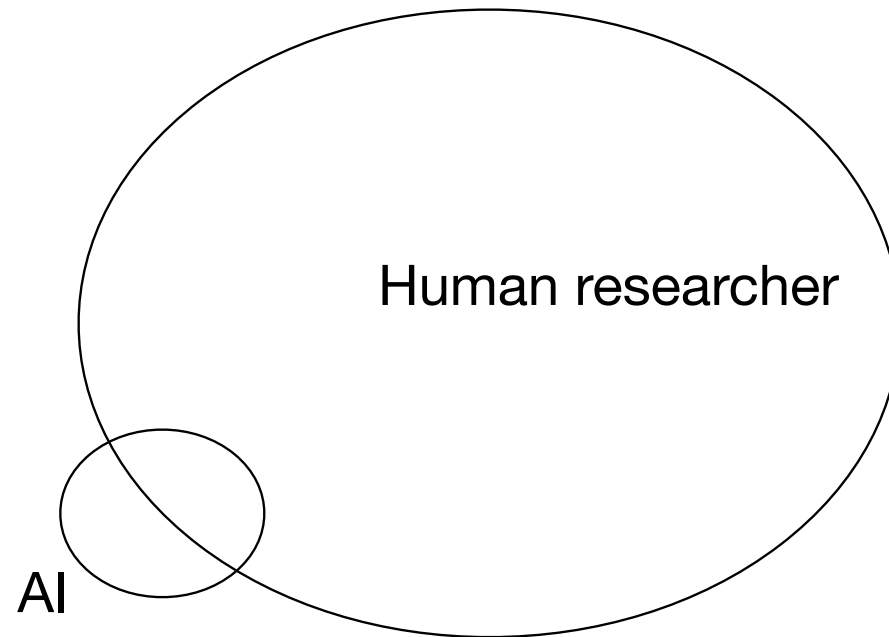
How do we measure progress

- Efficiency is easier to quantify than capabilities

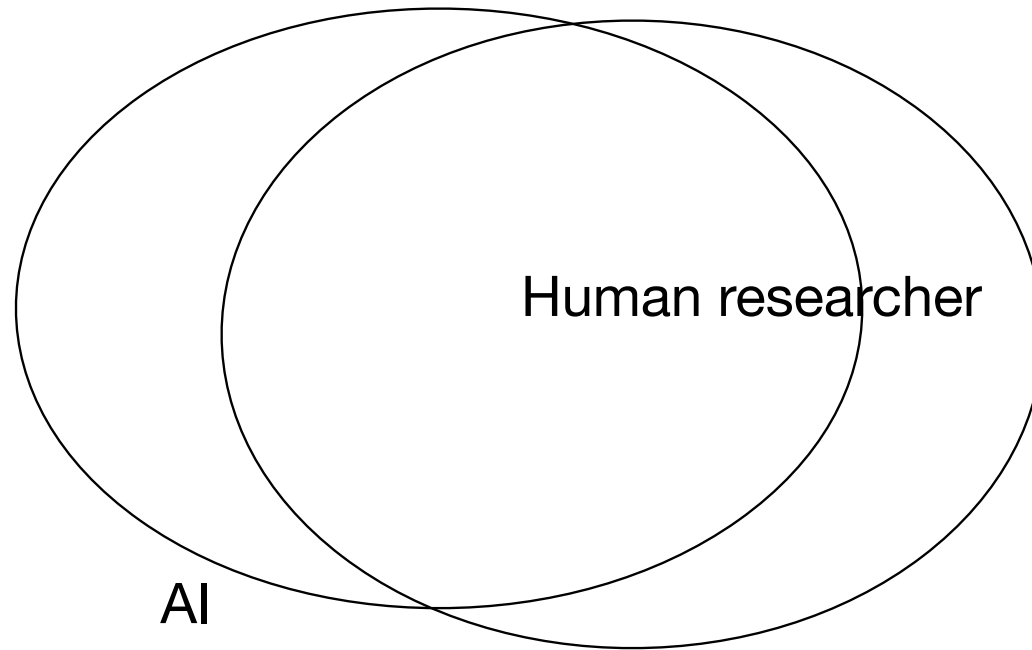
	Image recognition	Language translation	Game playing	Large language models	Algorithms writ large (with large datasets)
Result #1 Doubling time	15 months (runtime efficiency)	4 months (training efficiency)	4 months (Go, training efficiency)	8 months (base models, training efficiency)	Basically no gains (close to half of algorithm classes)
Result #2 Doubling time	16 months (training efficiency)	6 months (training efficiency)	25 days (Dota, training efficiency)	3.6 months (runtime cost efficiency)	~1-3 years (around a quarter of algorithm classes)
Result #3 Doubling time	9 months (training efficiency)		10-18 months (Atari games, training data efficiency)		< 1 year (around a quarter of algorithm classes)

Table 1: Summary of results from various studies investigating the efficiency doubling times of AI systems in several different domains. Note that most of these studies investigated training efficiency (how much computing resources are needed to train an AI system to a specific level of capabilities) instead of runtime efficiency (how much computing power is needed to run the resultant system).

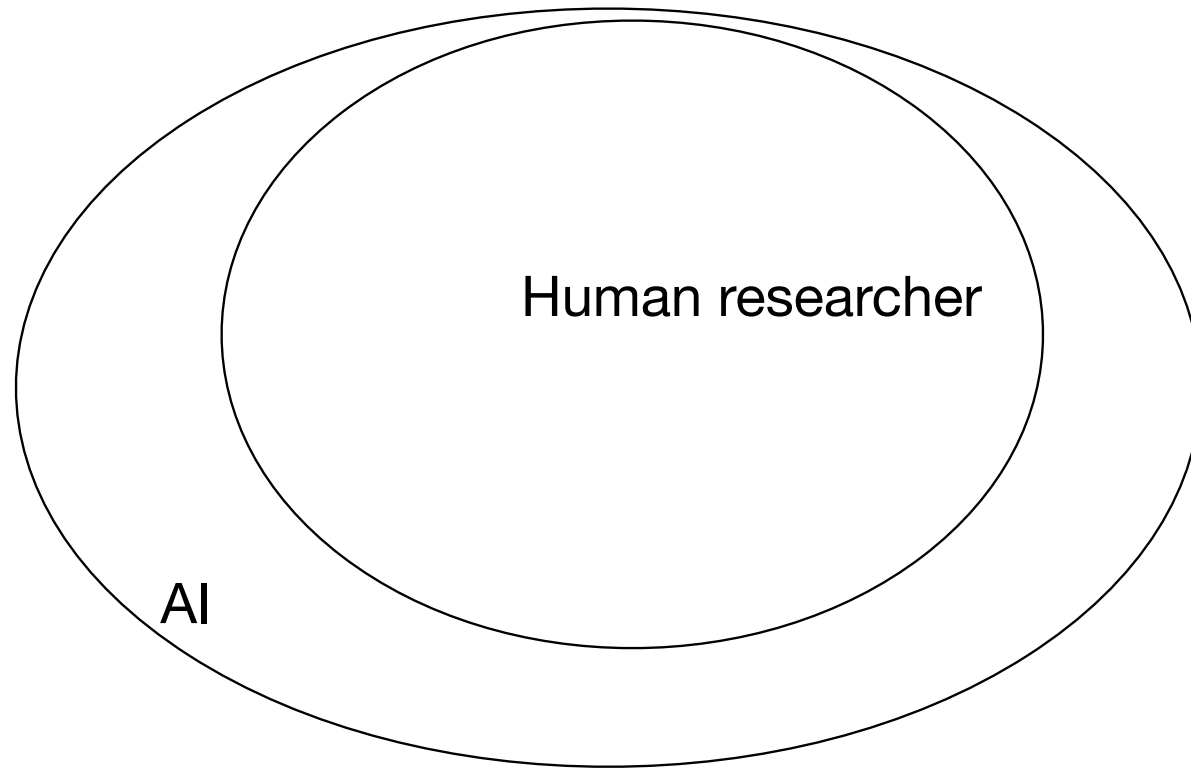
Range of tasks in R&D



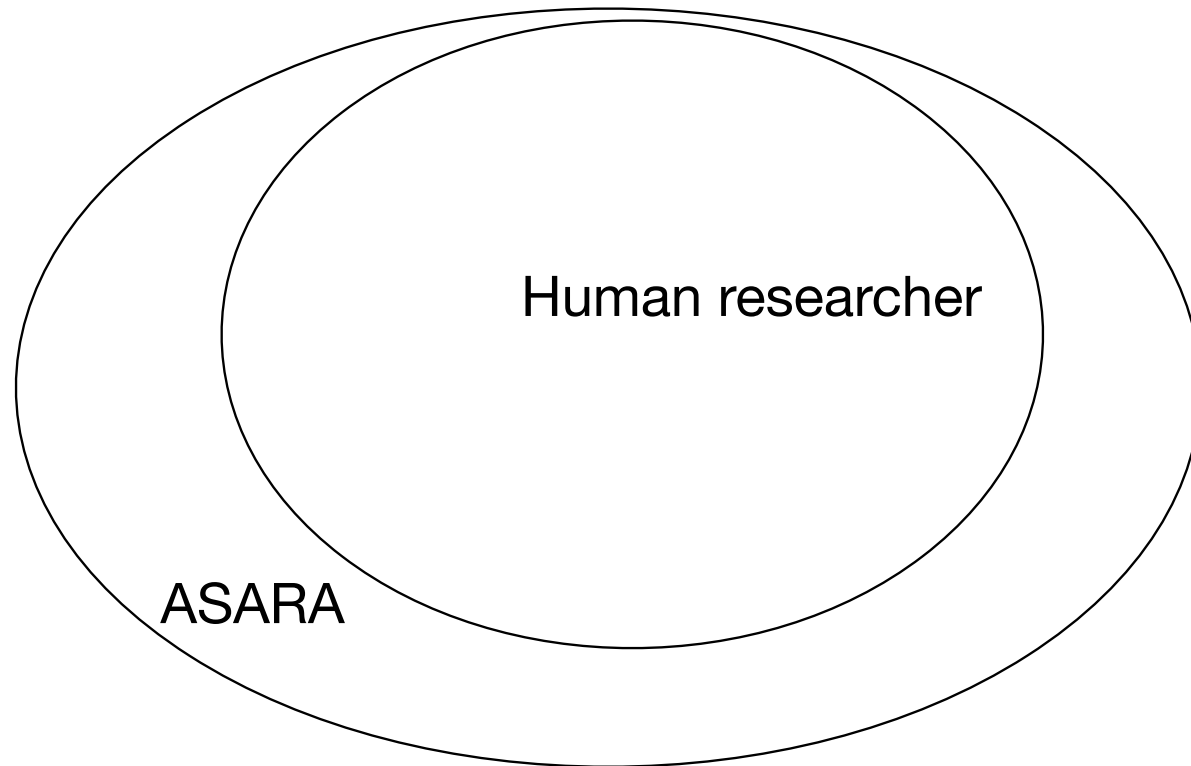
Range of tasks in R&D



Range of tasks in R&D

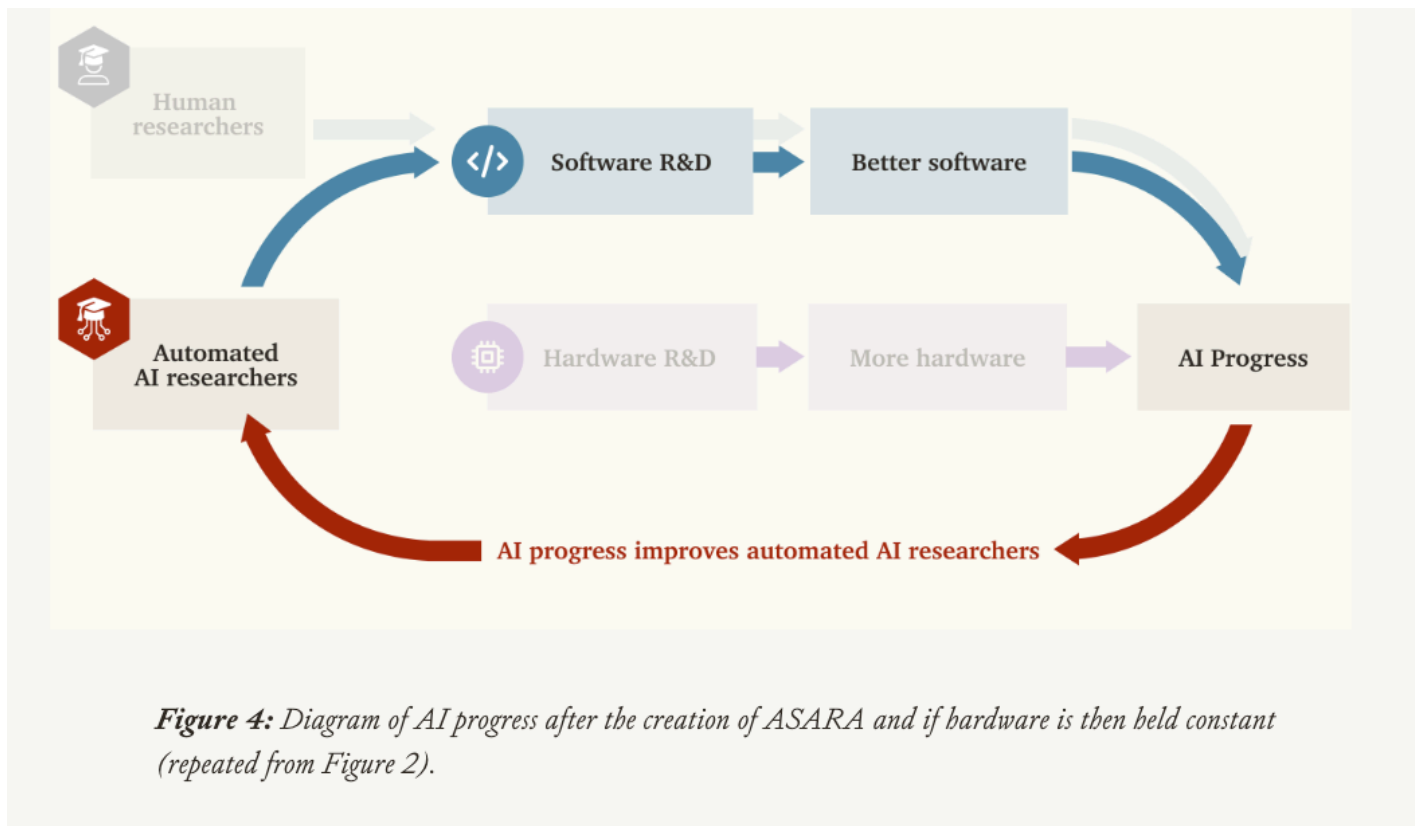


Range of tasks in R&D

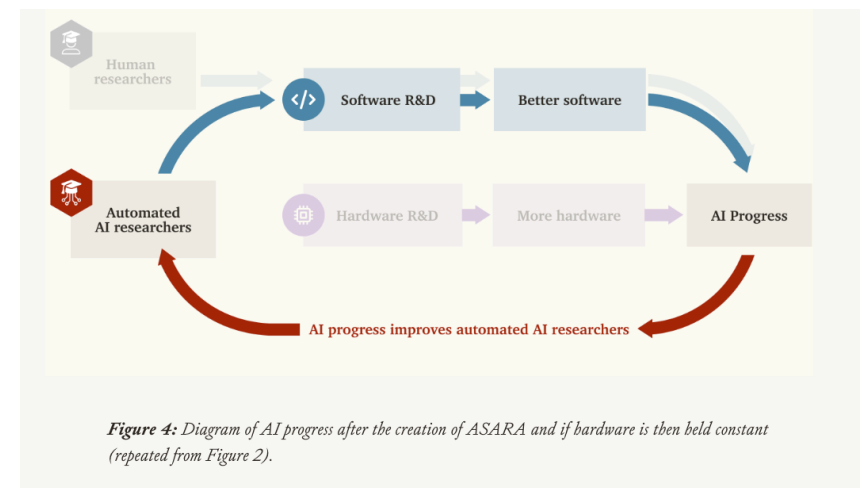


What happens when we reach ASARA

- Training >> Inference gap will increase
- Training compute can be repurposed for ASARA inference
 - > mass “cognitive output”, ~millions of agents running in parallel
- **However**, it might take 1,000 seconds of AI “thinking” to equal 1 second of human expert’s thought
 - > millions of agents ~= only thousands of humans
- So how fast would we then expect AI progress to be?



- Two competing dynamics:
- 1. *Diminishing returns* to software R&D, as software improvements get harder and harder to find.
- 2. The positive feedback from *increasingly powerful ASARA systems*.



Toy model

- Assumptions:
- ASARA is made up of “AGIs”. Each can do all tasks in software R&D
- AI progress = number of cumulative “papers” written (each of which represents an incremental amount of progress)
- Each AGI has the same productivity = number of papers per unit time
- AGIs cannot be made more productive over time, but more “computationally efficient” (thus we ignore capability improvements)

Toy model

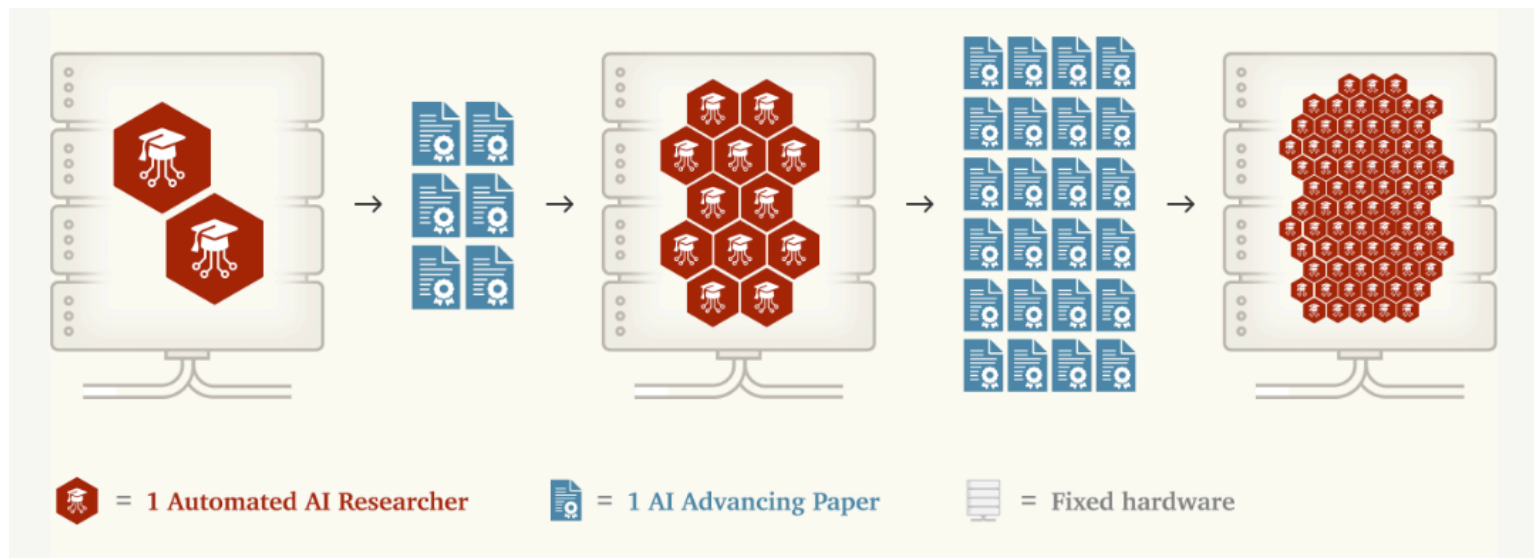


Figure 5: Illustration of the toy model. In the toy model, we're assuming that AGIs (represented in this figure by neural nets in graduation caps) perform software R&D, resulting in "papers," and these papers allow for increasing the efficiency of our AGIs, enabling the fixed amount of hardware to house more AGIs, resulting in more total papers per unit time, and so on.

Fizzle

- Initially 1 AGI.
Productivity = 1 paper per month.
- Computational efficiency is *doubled* after 2 papers.
So 2 months pass, and efficiency doubles.
Now we can run 2 AGIs.
- Both AGIs write 1 paper per month, so 2 papers per month in total.
Due to *diminishing returns*, we now need 3x higher = 6 papers to double the efficiency
- It takes 3 months to write 6 papers.
Now we have 4 AGIs.
- Again we need 3x higher = 18 papers to double efficiency.
This takes 4.5 months with 4 AGIs.



A fizzle

AI Progress Fizzles

SCENARIO

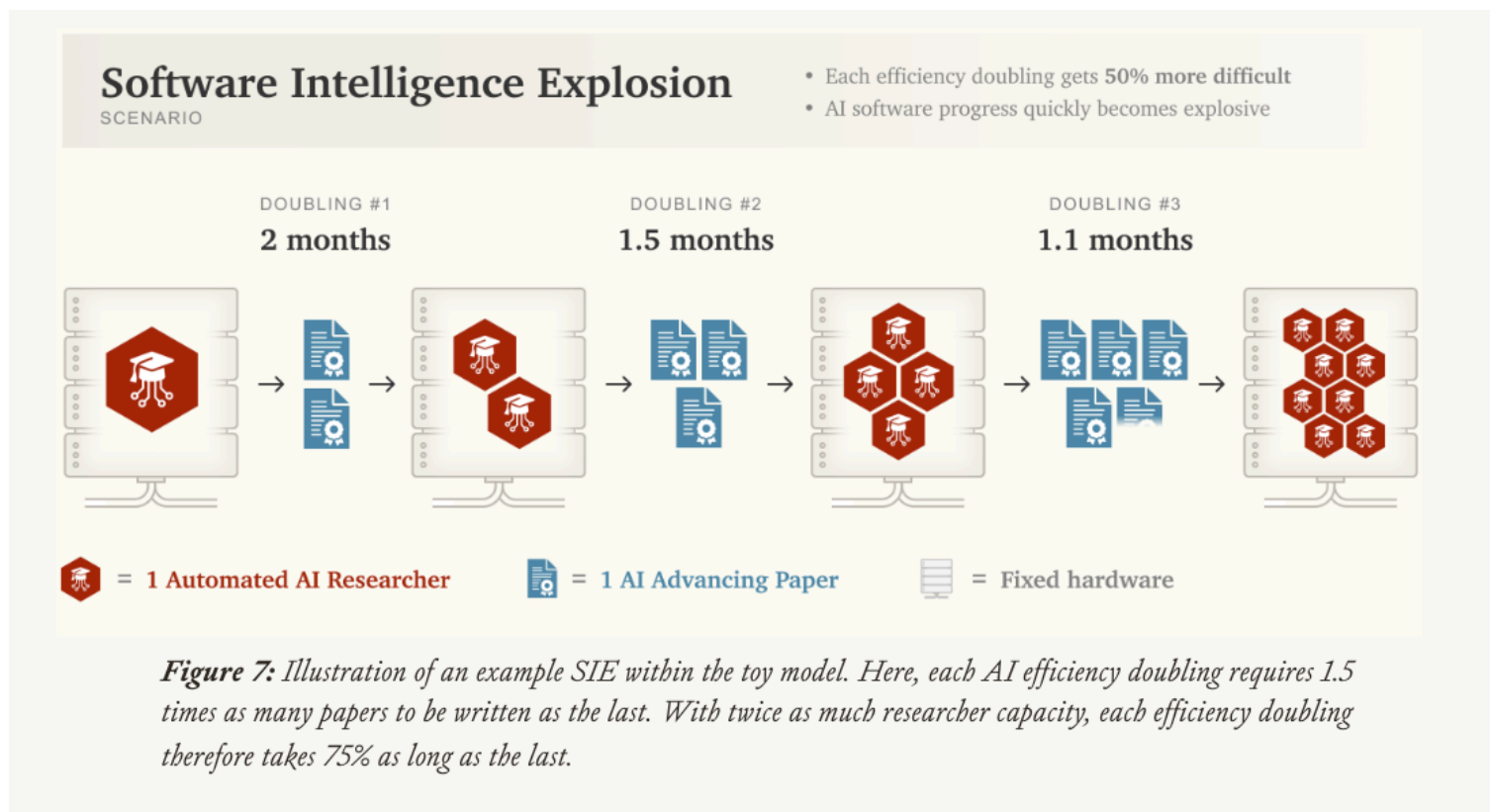
- Each efficiency doubling gets 3x more difficult
- AI software progress gradually slows down



Figure 6: Illustration of an example “fizzle” within the toy model. In this example, each AI efficiency doubling requires three times as many papers to be written as the last. With twice as much researcher capacity, each efficiency doubling therefore takes 50% longer than the last.

Software intelligence explosion (foom)

What if we only need 1.5x papers to double efficiency



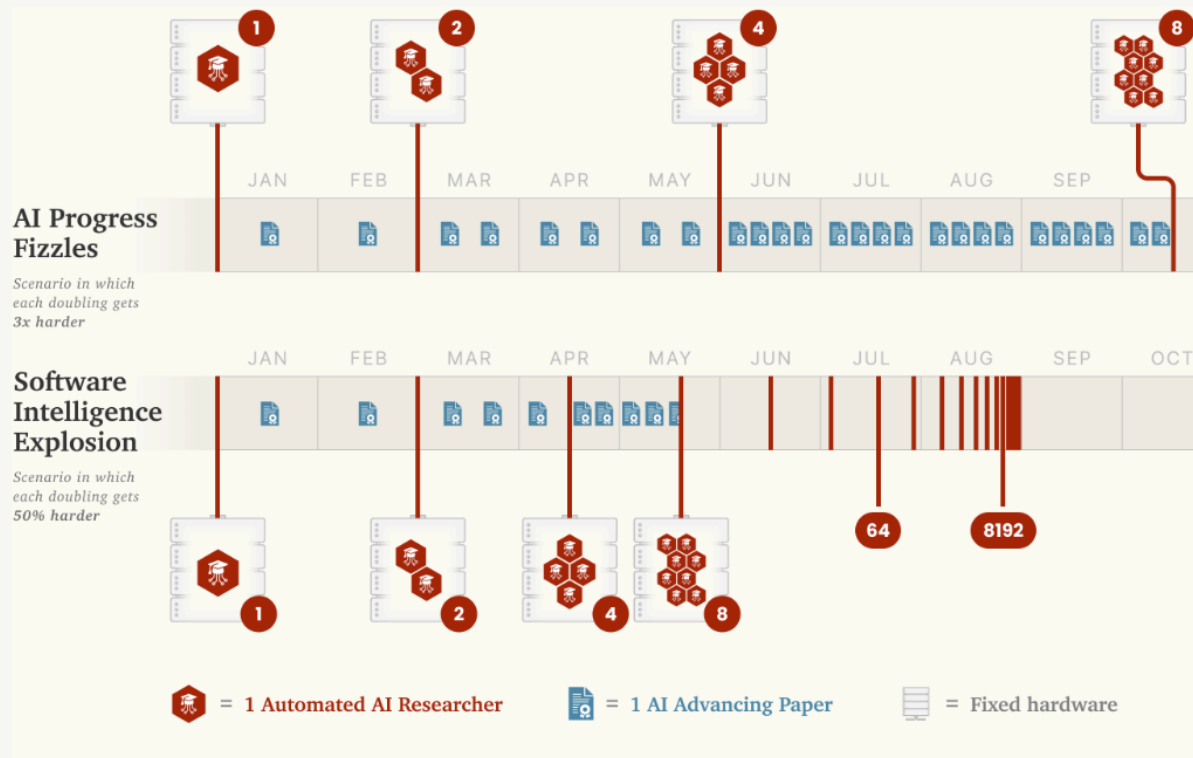


Figure 8: Comparison of our fizzle scenario and our SIE scenario. While progress gradually slows down in the fizzle, in an SIE it speeds up without limit. Given the parameters of our toy model, the SIE has an asymptote at 8 months after AI R&D is fully automated.

	Papers for first efficiency doubling	Papers for second efficiency doubling	Relative increase for each subsequent doubling
SCENARIO #1 AI Progress Fizzles	2	6	3x (from 6 / 2 = 3)
SCENARIO #2 Software Intelligence Explosion	2	3	1.5x (from 3 / 2 = 1.5)
SCENARIO #3 Exponential growth	2	4	2x (from 4 / 2 = 2)

Table 2: Chart demonstrating the differentiator between a fizzle and an SIE in our toy model. Note the specific number of papers listed for each doubling in each scenario is simply illustrative; the important point is that the tipping point between a fizzle and an SIE is when the papers required for a subsequent doubling in efficiency is twice as large as the papers required for the current doubling. Colored font used to help readers follow along.

Returns to software R&D

- $r = \frac{\Delta \log S}{\Delta \log W}$ Returns to software R&D
- S = software capacity
- W = software R&D (cumulative)
- One *doubling* of cumulative effort gives r *doublings* of software capacity

	Papers for first efficiency doubling	Papers for second efficiency doubling	Relative increase for each subsequent doubling
SCENARIO #1 AI Progress Fizzles	2	6	3x (from 6 / 2 = 3)
SCENARIO #2 Software Intelligence Explosion	2	3	1.5x (from 3 / 2 = 1.5)
SCENARIO #3 Exponential growth	2	4	2x (from 4 / 2 = 2)

Table 2: Chart demonstrating the differentiator between a fizzle and an SIE in our toy model. Note the specific number of papers listed for each doubling in each scenario is simply illustrative; the important point is that the tipping point between a fizzle and an SIE is when the papers required for a subsequent doubling in efficiency is twice as large as the papers required for the current doubling. Colored font used to help readers follow along.

	Relative increase in papers for each subsequent doubling	Doublings in papers required for each additional efficiency doubling	Returns to software R&D (r)
SCENARIO #1 AI Progress Fizzles	3x	1.58 (from: $\log_2(3) = 1.58$)	0.63 (from: $1/1.58 = 0.63$)
SCENARIO #2 Software Intelligence Explosion	1.5x	0.58 (from: $\log_2(1.5) = 0.58$)	1.72 (from: $1/0.58 = 1.72$)
SCENARIO #3 Exponential growth	2x	1 (from: $\log_2(2) = 1$)	1 (from: $1/1 = 1$)

Table 3: Continuation from Table 2, updated to demonstrate the derivation of r. Again, this table uses numbers that are illustrative of each scenario in our toy model, and colored font is used to help readers follow along. The tipping-point condition between an SIE and a fizzle is $r = 1$. Note the first column in this table is copied over from the last column in Table 2.

- $r > 1$: SIE
- $r = 1$: exponential
- $r < 1$: fizzle

- Is $r = 1$ unrealistic?
e.g. COVID

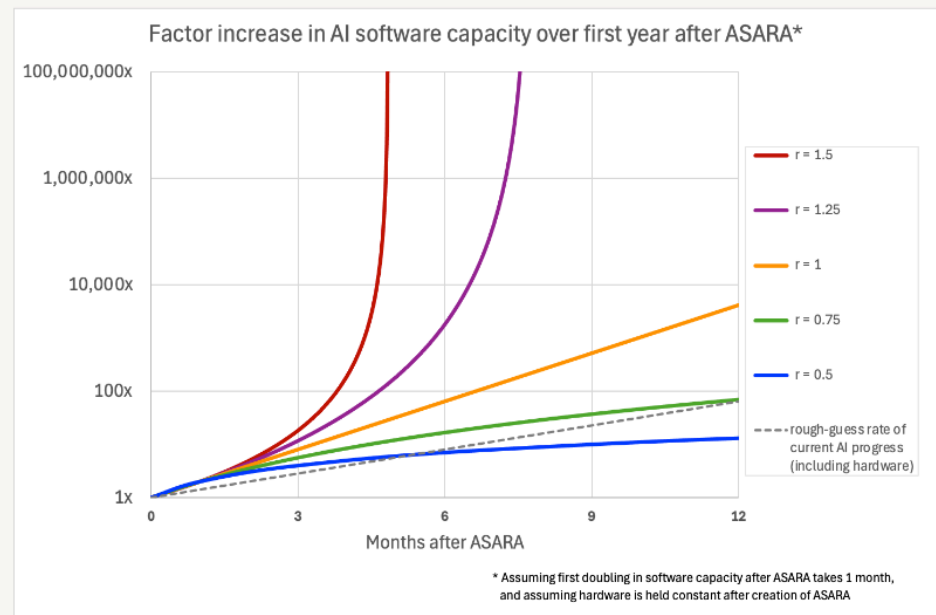


Figure 9: Graph showing AI software progress over the first year after ASARA, as a function of r , assuming the first doubling in software progress takes 1 month and hardware is held constant upon the creation of ASARA. Note that for $r > 1$ (red and purple lines above), representing an SIE, progress becomes faster and faster, without limit (in reality, physical limits would eventually reduce r and slow progress, but it's unclear whether those physical limits would start biting before tremendous progress was achieved – see Figure 10 for more). For $r < 1$ (green and blue lines above), representing a fizzle, progress slows down over time, but could still be quite fast for a period of time (for reference, the dashed line on the graph corresponds to a very rough guess of the current rate of progress in AI capacity, including gains derived from hardware growth). Note that this graph makes simplifying assumptions, such as ignoring the “stepping on toes” effect.²⁷

- Eventually we expect $r = 0$. (Theoretical limit with fixed hardware)
- But when?

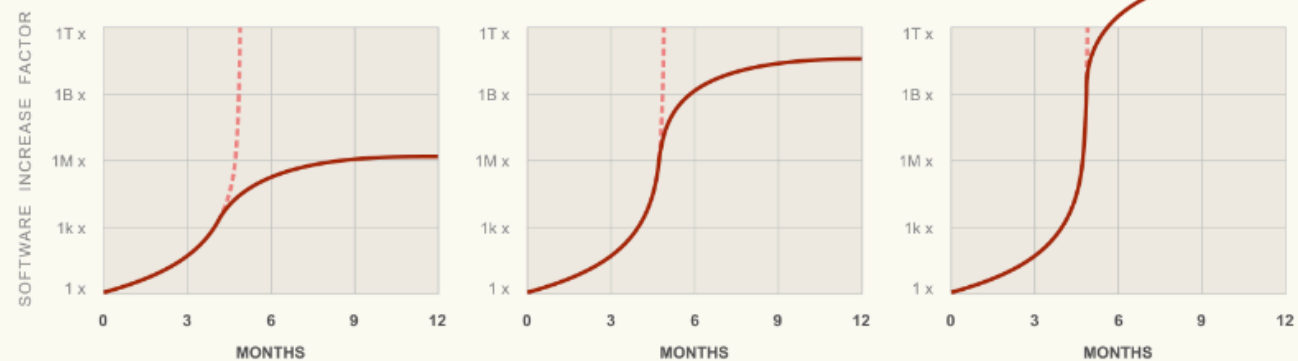
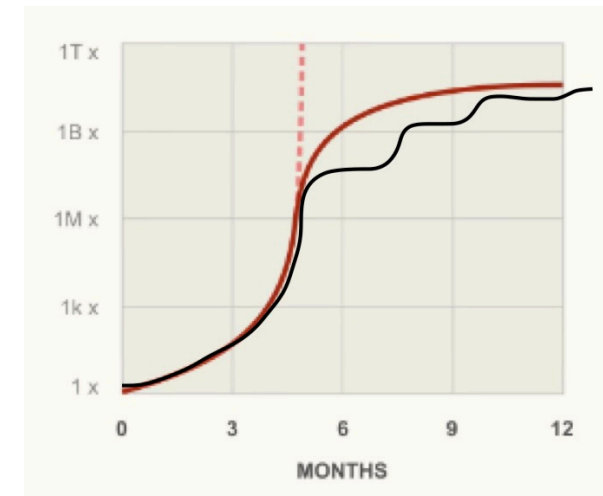


Figure 10: Even if there is an SIE, physical limits (as described in footnote 26) would eventually slow progress, if progress doesn't slow for other reasons first. There is tremendous uncertainty about how high these limits are above the first ASARA systems, though, and we may see extraordinary progress before these limits are approached.

- ASARA is, by definition, as good as humans at coming up with new useful paradigms, e.g. GOFAL
- Paradigms might still be a bottleneck, like hardware



- Measurements of r

	Image recognition	Computer chess	RL training data efficiency	SAT solvers	Linear programming	Algorithms in general
r EST.	1.4	0.8	1.6	3.5	1.1	~4?

Table 4: Summary of estimates of r from various domains related to AI. Note that post-training enhancement and capability advancements would further increase our estimate of r .

- Something to think about: how much of this is due to increase in compute?

- “The software intelligence explosion debate needs experiments”

Ho and Whitfill

Nov. 14, 2025

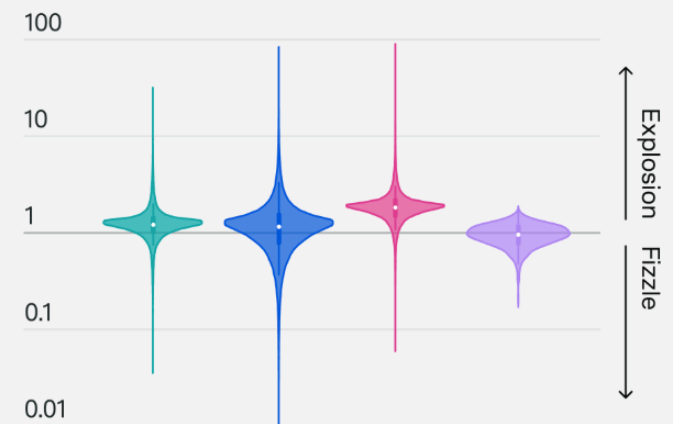
- Median r between 0.96 and 1.9

<https://epoch.ai/gradient-updates/the-software-intelligence-explosion-debate-needs-experiments>

AI software R&D returns may be high enough for a software-only singularity

- Computer vision
- Reinforcement learning
- Language models (Approach 1)
- Language models (Approach 2)

Returns to AI software R&D

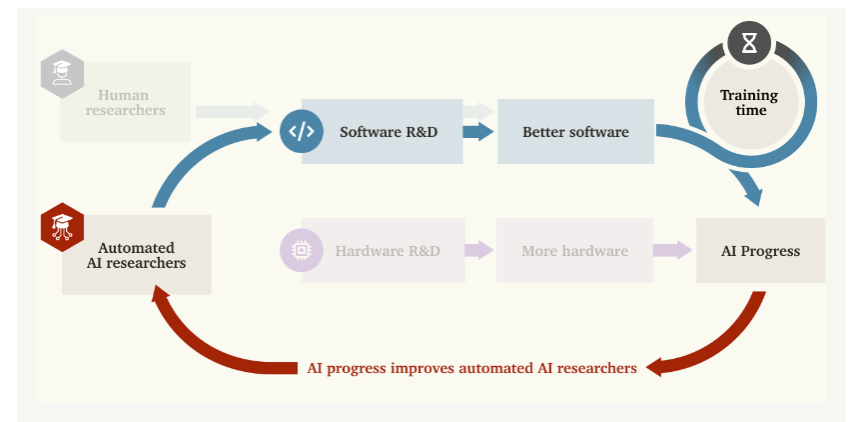
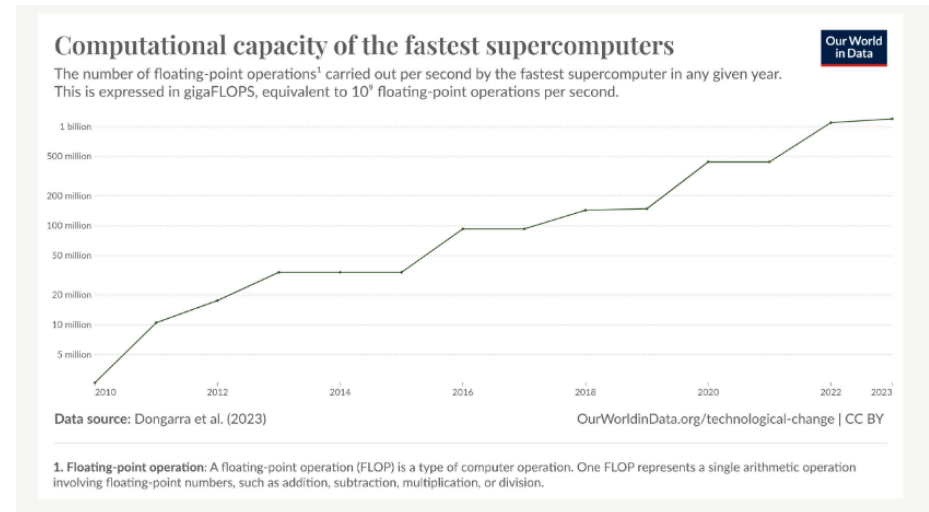


EPOCH AI | CC-BY

epoch.ai

Bottlenecks

- Most discoveries could have been driven by large compute
- *Training* the next (more efficient) ASARA can take a long time
- But:
- Software efficiency -> more efficient experiments
- Small experiments can give insights about large experiments
- Methods that are inherently cheap to experiment with
- GOF AI?



Implications

- Even if hardware is held constant, the creation of ASARA could lead to very fast capability gains, e.g. doubling *every few days or faster* (c.f. few months)

Implications

- Even a fizzle, e.g. with $r = 0.7$, can still be very fast!
This yields *five* doublings in the first year (30, 41, 55, 74, 100 days), *starting from a very capable AI system*



A large fizzle

Implications

- Hardware will most likely *not* be held constant