[Transcript] Lex Fridman Podcast / #380 – Neil Gershenfeld: Self-Replicating Robots and the Future of Fabrication

The following is a conversation with Neil Gershenfeld,
the director of MIT's Center for Bits and Atoms,
an amazing laboratory that is breaking down boundaries
between the digital and physical worlds,
fabricating objects and machines at all scales of reality,
including robots and automata
that can build copies of themselves
and self-assemble into complex structures.
His work inspires millions across the world
as part of the maker movement to build cool stuff,
to create the very act that makes life so beautiful
and fun.
And now, a quick few second mention of each sponsor.
Check them out in the description.
It's the best way to support this podcast.
We've got Element 4, Zero Sugar Hydration,
NetSuite for Business Management Software
and BetterHelp for Mental Health.
Choose wisely, my friends.
Also, if you want to work with our amazing team
where I was hiring,
go to lexfreedman.com slash hiring.
And now, onto the full ad reads.
As always, no ads in the middle.
I try to make this interesting,
but if you must skip them,
please still check out our sponsors.
I enjoy their stuff, maybe you will too.
This episode is brought to you by Element,
spelled L-M-N-T.
It's an electrolyte drink mix
that I'm currently drinking,
that I drink throughout the day.
I drink a huge amount of it.
My favorite flavor is the watermelon salt flavor.
Doesn't mean it'll be your favorite flavor,
but I'm pretty sure it's gonna be your favorite flavor.
For fasting, for low carb diets,
for all kinds of diets really,
but certainly for low carb keto carnivore,
you have to get the electrolytes right.
They call it the keto flu
if you don't get the electrolytes right.

You know, for doing all kinds of crazy exercise
that I do, all of that,
you have to get the sodium, the potassium, magnesium,
and element as a great job of balancing all of that.
Makes it delicious, makes it really easy
to make sure that you're getting just the water intake right.
Because it basically makes water taste great
and balances out the electrolytes, the hydration,
everything for you.
I bring it, when I travel, I bring it anywhere I go.
I have to have element as part of my life.
It makes me happy.
It makes me feel like I got my stuff together.
Anyway, get a simple pack for free with any purchase.
Try it at drinkelement.com slash Lex.
This show is also brought to you by Netsuite,
an all-in-one cloud business management system.
You can manage financials, HR, inventory, e-commerce,
if you do that kind of thing,
and many business-related details.
Running a company is really complicated.
There's a lot of people involved,
a lot of tasks involved, a lot of roles involved.
It's not just engineering.
It's not just design, idea, strategy, vision,
all those kinds of things.
It's all of the glue, the thing that makes the thing
a cohesive, singular system
that works efficiently and flawlessly.
And so you have to hire the right people.
You have to use the best tools for the job.
Netsuite does that in the cloud
to manage all kinds of messy business details
and make it super easy.
You can start now with no payment
or interest for six months.
Go to netsuite.com slash Lex
to access their one-of-a-kind financing program.
That's netsuite.com slash Lex.
This episode is also brought to you by BetterHelp,
spelled H-E-L-P, help.
They figure out what you need
to match you with a licensed professional therapist

in under 48 hours, talking to somebody
through all the difficult things
that are going through your mind.
Taking those thoughts, taking those memories,
converting them into words
and letting those words leave your mouth, sounds,
somehow transforms your own understanding
of what those memories, those thoughts mean.
It allows you to move past them,
to integrate them into a healthier,
a deeper understanding of the world around you,
your own emotions of your own way of being.
It's kind of amazing that talk therapy, that talking,
talking with the person that knows what they're doing
is a powerful way to do that kind of growth.
It's funny and it's amazing when simple things like therapy,
can have a big profound impact, results on your life.
Anyway, one of the big things about BetterHelp
is just how easy it is to get started.
I think that's a barrier for a lot of people.
So it's really important to say that it's easy to screen.
It is affordable and it's available everywhere, worldwide.
Check them out at betterhelp.com slash Lex
and save on your first month.
That's betterhelp.com slash Lex.
This is a Lex Friedman podcast.
To support it, please check out our sponsors
in the description.
And now, dear friends, here's Neil Gershwinfeld.
You have spent your life working at the boundary
between bits and atoms.
So the digital and the physical.
What have you learned about engineering
and about nature of reality from working at this divide,
trying to bridge this divide?
I learned why Van Neumann and Turing
made fundamental mistakes.
That's good stuff.
I learned the secret of life.
Yeah.
I learned how to solve many of the world's
most important problems, which all sound presumptuous,
but all of those are things I learned at that boundary.

OK, so Turing and Van Neumann, let's start there.
Some of the most impactful, important humans
who have ever lived in computing, why were they wrong?
So I worked with Andy Gleeson, who is Turing's counterpart.
So just for background, if anybody doesn't know,
Turing is credited with the modern architecture
of computing, among many other things.
Andy Gleeson was his US counterpart.
And you might not have heard of Andy Gleeson,
but you might have heard of the Hilbert Problems.
And Andy Gleeson solved the fifth one.
So he was a really notable mathematician.
During the war, he was Turing's counterpart.
Then Van Neumann is credited with the modern architecture
of computing, and one of his students was Marvin Minsky.
So I could ask Marvin what Johnny was thinking,
and I could ask Andy what Alan was thinking.
And what came out from that, what
I came to appreciate as background,
I never understood the difference between computer
science and physical science.
But Turing's machine, that's the foundation
of modern computing, has a simple physics mistake, which
is the head is distinct from the tape.
So in the Turing machine, there's a head
that programmatically moves and reads and writes a tape.
The head is distinct from the tape,
which means persistence of information
is separate from interaction with information.
Then Van Neumann wrote deeply and beautifully
about many things, but not computing.
He wrote a horrible memo called the first draft
of a report on the EDVAC, which is how you program
a very early computer.
In it, he essentially roughly took Turing's architecture
and built it into a machine.
So the legacy of that is the computer somebody's
using to watch this is spending much of its effort
moving information from storage transistors
to processing transistors, even though they
have the same computational complexity.
So in computer science, when you learn about computing,
there's a ridiculous taxonomy of about 100 different models

of computation.
But they're all fictions.
In physics, a patch of space occupies space.
It stores state.
It takes time to transit, and you can interact.
That is the only model of computation that's physical.
Everything else is a fiction.
So I really came to appreciate that a few years back
when I did a keynote for the annual meeting
at the supercomputer industry, and then went into the halls
and spent time with the supercomputer builders
and came to appreciate, oh, let's see,
if you're familiar with the movie, The Metropolis,
people would frolic upstairs in the gardens,
and down in the basement, people would move levers.
And that's how computing exists today,
that we pretend software is not physical.
It's separate from hardware.
And the whole canon of computer science
is based on this fiction that bits aren't constrained by atoms.
But all sorts of scaling issues in computing
come from that boundary, but all sorts of opportunities
come from that boundary.
And so you can trace it all the way back
to Turing's machine making this mistake between the head
and the tape.
Van Neumann, he never called it Van Neumann's architecture.
He wrote about it in this dreadful memo,
and then he wrote beautifully about other things
we'll talk about.
Now, to end a long answer, Turing and Van Neumann
both knew this.
So all of the canon of computer scientists credits them
for what was never meant to be a computer architecture.
Both Turing and Van Neumann ended their life
studying exactly how software becomes hardware.
So Van Neumann studied self-reproducing automata,
how a machine communicates its own construction.
Turing studied morphogenesis, how genes give rise to form.
They ended their life studying the embodiment of computation,
something that's been forgotten by the canon of computing,
but developed sort of off to the sides
by a really interesting lineage.

So there's no distinction between the head and the tape,
between the computer and the computation.
It is all computation.
Right, so I never understood the difference
between computer science and physical science.
And working at that boundary helped lead to things like,
my lab was part of doing with a number of interesting
collaborators, the first Fester than classical quantum
computations, we were part of a collaboration creating
the minimal synthetic organism where you design life
in a computer.
Those both involve domains where you just can't separate
hardware from software.
The embodiment of computation is embodied
in these really profound ways.
So the first quantum computations, synthetic life,
so in the space of biology, the space of physics
at the lowest level and the space of biology
at the lowest level.
So let's talk about CBA, Center of Bits and Atoms.
What's the origin story of this MIT legendary MIT center
that you're a part of creating?
In high school, I really wanted to go to vocational school
where you learned to weld and fix cars and build houses.
And I was told, no, you're smart.
You have to sit in a room.
And nobody could explain to me why I couldn't go
to vocational school.
I then worked at Bell Labs, this wonderful place
before deregulation, legendary place.
And I would get union grievances because I would go
into the workshop and try to make something.
And they would say, no, you're smart.
You have to tell somebody what to do.
And it wasn't until MIT, and I'll explain how CBA started,
but I could create CBA, that I came to understand
this is a mistake that dates back to the Renaissance.
So in the Renaissance, the liberal arts emerged.
And liberal doesn't mean politically liberal.
This was the path to liberation, birth of humanism.
And so the liberal arts were the trivium,
quadrivium, roughly language, natural science.
And at that moment, what emerged was this dreadful concept

of the illiberal arts.
So anything that wasn't the liberal arts
was for commercial gain and was just making stuff
and wasn't valid for serious study.
And so that's why we're left with learning to weld
wasn't a subject for serious study.
But the means of expression have changed
since the Renaissance.
So micromachining or embedded coding
is every bit as expressive as painting a painting
or writing a sonnet.
So never understanding this difference
between computer science and physical science.
The path that led me to create CBA with colleagues
was I was what's called the junior fellow at Harvard.
I was visiting MIT through Marvin
because I was interested in the physics
of musical instruments.
This will be another slight digression.
And Cornell, I would study physics
and then I would cross the street
and go to the music department where I played the bassoon
and I would trim reads and play the reads.
And they'd be beautiful, but then they'd get soggy.
And then I discovered in the basement
of the music department at Cornell was David Borden,
who you might not have heard of,
but his legend during electronic music
because he was really the first electronic musician.
So Bob Mogue, who invented Mogue synthesizers
was a physics student at Cornell,
like me crossing the street.
And eventually he was kicked out
and invented electronic music.
David Borden was the first musician
who created electronic music.
So he's legendary for people like Phil Glass
and Steve Reich.
And so that got me thinking about,
I would behave as a scientist in the music department,
but not in the physics department,
but not in the music department.
Got me thinking about what's the computational capacity

of a musical instrument.
And through Marvin, he introduced me to Todd Backover
at the Media Lab,
who was just about to start a project with Yo-Yo Ma
that led to a collaboration to instrument a cello,
to extract Yo-Yo's data
and bring it out into computational environments.
What is the computational capacity of a musical instrument?
Does it continue on this tangent
and will we share a return to CBA?
Yeah, so one part of that is to understand the computing.
And if you look at like the finest timescale
and length scale you need to model the physics,
it's not heroic.
A good GPU can do teraflops today.
That used to be a national class supercomputer,
now it's just a GPU.
And that's about,
if you take the timescales and length scales
relevant for the physics,
that's about the scale of the physics computing.
For Yo-Yo, what was really driving it
was he's completely unsentimental about the Strad.
It's not that it makes some magical wiggles
in the sound wave.
It's performance as a controller,
how he can manipulate it as an interface device.
Interface between what and what exactly?
Him and sound.
Okay, him and sound.
And so what it led to was,
I had started by thinking about ops per second,
but Yo-Yo's question was really resolution and bandwidth.
It's how fast can you measure what he does?
And the bandwidth and the resolution of,
detecting his controls and then mapping them into sounds.
And what we found, what he found was,
if you instrument everything he does
and connect it to almost anything,
it sounds like Yo-Yo,
that the magic is in the control,
not in ineffable details in how the wood wiggles.
And so with Yo-Yo and Todd that led to a piece in,

towards the end I asked Yo-Yo,
what it would take for him to get rid of his Strad
and use our stuff.
And his answer was just logistics.
It was at that time, our stuff was like a rack of electronics
and lots of cables and some grad students to make it work.
Once the technology becomes as invisible as the Strad,
then sure, absolutely he would take it.
And by the way, as a footnote on the footnote,
an accident in the sensing of Yo-Yo's cello
led to $100 million a year auto safety business
to control airbags and cars.
How did that work?
I had to instrument the bow without interfering with it.
So I set up local electromagnetic fields
where I would detect how those fields interact
with the bow he's playing.
But we had a problem that his hand,
whenever his hand got near these sensing fields,
I would start sensing his hand
rather than the materials on the bow.
And I didn't quite understand what was going on
with those, that interference.
So my very first grad student ever, Josh Smith,
did a thesis on tomography with electric fields,
how to see in 3D with electric fields.
Then through Todd and at that point,
research scientist in my lab, Joe Paradiso,
it led to a collaboration with Penn and Teller
who, where we did a magic trick in Las Vegas
to contact Houdini.
And sort of these fields are sort of like,
contacting spirits.
So we did a magic trick in Las Vegas.
And then the crazy thing that happened after that
was Phil Rittmueller came running into my lab.
He worked with, this became with Honda and NEC,
airbags were killing infants in rear-facing child seats.
Cars need to distinguish a front-facing adult
where you'd save the life versus a bag of groceries
where you don't need to fire the airbag
versus a rear-facing infant where you would kill it.
And so the seat need to, in effect,

see in 3D to understand the occupants.
And so we took the Penn and Teller magic trick
derived from Josh's thesis from Yo-Yo's cello
to an auto show.
And all the car companies said, great, when can we buy it?
And so that became Ellisissa.
And it was $100 million a year business making sensors.
There wasn't a lot of publicity
because it was in the car, so the car didn't kill you.
So they didn't sort of advertise,
we have nice sensors, so the car doesn't kill you.
But it became a leading auto safety sensor.
And that started from the cello
and the question of the computational capacity,
the musical instrument.
Right.
So now to get back to MIT,
I was spending a lot of outside time at IBM research
that had gods of the foundations of computing.
There's just amazing people there.
And I'd always expected to go to IBM to take over a lab,
but at the last minute pivoted and came to MIT
to take a position in the Media Lab
and start what became the predecessor to CBA.
Media Lab is well known for Nicholas Negroponte.
What's less well known is the role of Jerry Wiesner.
So Jerry was MIT's president
before that Kennedy science advisor,
grand old man of science.
At the end of his life, he was frustrated
by how knowledge was segregated.
And so he wanted to create a department
of none of the above,
a department for work that didn't fit in departments.
And the Media Lab in a sense was a cover story
for him to hide a department.
As MIT's president towards the end of his tenure,
if he said, I'm gonna make a department
for things that don't fit in departments,
the departments would have screamed.
But everybody was sort of paying attention
to Nicholas creating the Media Lab.
And Jerry kind of hid in it a department

called Media Arts and Sciences.
It's really the department of none of the above.
And Jerry explaining that and Nicholas then confirming,
it is really why I pivoted and went to MIT.
Because my students who helped create quantum computing
or synthetic life get degrees from Media Arts and Sciences,
this department of none of the above.
So that led to coming to MIT
with Todd and Joe Paradiso and my colleague.
We started a consortium called Things That Think.
And this was around the birth of Internet of Things
and RFID.
But then we started doing things like work we can discuss
that became the beginnings of quantum computing
and cryptography and materials
and logic and microfluidics.
And those needed much more significant infrastructure
and we're much longer research arcs.
So with a bigger team of about 20 people,
we wrote a proposal to the NSF
to assemble one of every tool to make anything of any size
was roughly the proposal.
One of any tool to make anything of any size.
Yeah, so they're usually nanometers, micrometers,
millimeters, meters are segregated,
input and output is segregated.
We wanted to look just very literally
how digital becomes physical and physical becomes digital.
And fortunately, we got NSF on a good day
and they funded this facility of one of almost every tool
to make anything.
And so with a group of core colleagues
that included Joe Jacobson, Ike Tryang, Scott Minales,
we launched CBA.
And so you're talking about nanoscale, microscale,
nanostructures, microstructures, macrostructures,
electron microscopes and focus time beam probes
for nanostructures, laser, micro machining
and X-ray microtomography for microstructures,
multi-axis machining and 3D printing
for macrostructures, just some examples.
What are we talking about in terms of scale?
How can we build tiny things and big things

all in one place?
Yeah, so a well-equipped research lab
has the sort of tools we're talking about,
but they're segregated in different places.
They're typically also run by technicians
where you then have an account and a project and you charge.
All of these tools are essentially
when you don't know what you're doing,
not when you do know what you're doing,
in that they're when you need to work across length scales
where we don't, once projects are running in this facility,
we don't charge for time,
you don't make a formal proposal to schedule
and the users really run the tools and it's for work
that's kind of in Co8 that needs to span
these disciplines and length scales.
And so work in the project today,
work in CBA today ranges from developing
Zeptojoule electronics for the lowest power computing
to micromachining diamond to take 10 million RPM bearings
for molecular spectroscopy studies
up to exploring robots to build
100 meter structures in space.
Okay, can we, the three things you just mentioned,
let's start with the biggest.
What are some of the biggest stuff you attempted
to explore how to build in a lab?
Sure, so viewed from one direction,
what we're talking about is a crazy random seeming
of almost unrelated projects,
but if you rotate 90 degrees,
it's really just a core thought over and over again,
just very literally how bits and atoms relate,
how digital and just going from digital to physical
in many different domains,
but it's really just the same idea over and over again.
So to understand the biggest things,
let me go back to bring in now Shannon
as well as Van Neumann.
Claude Shannon.
Yeah, so what is digital?
The casual obvious answer is digital in one and zero,
but that's wrong.

There's a much deeper answer,
which is Claude Shannon at MIT
wrote the best master's thesis ever.
In his master's thesis,
he invented our modern notion of digital logic.
Where it came from was Vannevar Bush
was a grand old man at MIT.
He created the post-war research establishment
that led to the National Science Foundation
and he made an important mistake,
which we can talk about,
but he also made the differential analyzer,
which was the last grade analog computer.
So it was a room full of gears and pulleys
and the longer it ran, the worse the answer was.
And Shannon worked on it as a student
and he got so annoyed in his master's thesis,
he invented digital logic,
but he then went on to Bell Labs
and what he did there was communication
was beginning to expand,
there was more demand for phone lines.
And so there's a question about how many phone lines
you could, phone messages you could send down a wire.
And you could try to just make it better and better.
He asked a question nobody had asked,
which is rather than make it better and better,
what's the limit to how good it can be?
And he proved a couple of things,
but one of the main things he proved
was a threshold theorem for channel capacity.
And so what he showed was my voice to you right now
is coming as a wave through sound
and the further you get, the worse it sounds.
But people watching this are getting it
as packets of data in a network.
When they get, when the computer,
they're watching this gets the packet of information,
it can detect and correct an error.
And what Shannon showed is if the noise in the cable
to the people watching this
is above a threshold they're doomed,
but if the noise is below a threshold

for a linear increase in the energy
representing our conversation,
the error rate goes down exponentially.
Exponentials are fast,
there's very few of them in engineering,
and the exponential reduction of error
below a threshold if you restore state
is called a threshold theorem.
That's what led to digital,
that means unreliable things can work reliably.
So Shannon did that for communication.
Then Van Neumann was inspired by that
and applied it to computation
and he showed how an unreliable computer
can operate reliably
by using the same threshold property of restoring state.
It was then forgotten many years,
we had to rediscover it in effect
in the quantum computing era
when things are very unreliable again.
But now to go back to,
how does this relate to the biggest things I've made?
So in fabrication,
MIT invented computer-controlled manufacturing in 1952.
Jet aircraft were just emerging.
There was a limit to turning cranks on a machine,
on a milling machine to make parts for jet aircraft.
Now, this is a messy story.
MIT actually stole computer-controlled machining
from an inventor who brought it to MIT,
wanted to do a joint project with the Air Force
and MIT effectively stole it from him.
So it's kind of a messy history.
But that sounds like the birth
of computer-controlled machining, 1952.
There are a number of inventors of 3D printing.
One of the companies spun off my lab
by Max Lobowski's Forum Labs,
which is now a billion dollar 3D printing company.
That's the modern version.
But all of that's analog,
meaning the information is in the control computer,
there's no information in the materials.

And so it goes back to Vannevar Bush's analog computer.
If you make a mistake in printing or machining,
just the mistake accumulates.
The real birth of computerized digital manufacturing
is four billion years ago.
That's the evolutionary age of the ribosome.
So the way you're manufactured
is there's a code that describes you, the genetic code.
It goes to a micro machine, the ribosome,
which is this molecular factory
that builds the molecules that are you.
The key thing to know about that is
there are about 20 amino acids that get assembled.
And in that machinery,
it does everything Shannon and Van Neumann taught us.
You detect and correct errors.
So if you mix chemicals, the error rate
is about a part in a hundred.
When you make elongated protein in the ribosome,
it's about a part in 10 to the four.
When you replicate DNA,
there's an extra level of error correction.
It's a part in 10 to the eight.
And so in the molecules that make you,
you can detect and correct errors
and you don't need a ruler to make you.
The geometry comes from your parts.
So now compare a child playing with Lego
and a state of the art 3D printer
or a computerized milling machine.
The tower made by a child is more accurate
than their motor control
because the act of snapping the bricks together
gives you a constraint on the joints.
You can join bricks made out of dissimilar materials.
You don't need a ruler for Lego
because the geometry locally gives you the global parts
and there's no Lego trash.
The parts have enough information to disassemble them.
Those are exactly the properties of a digital code.
The unreliable is made reliable.
Yes, absolutely.
So what the ribosome figured out four billion years ago

is how to embody these digital properties
but not for communication or computation in effect
but for construction.
So a number of projects in my lab
have been studying the idea of digital materials
and think of a digital material just as Lego bricks.
The precise meaning is a discrete set of parts
reversibly joined with global geometry
determined from local constraints.
And so it's digitizing the materials.
And so I'm coming back to
what are the biggest things I've made.
My lab was working with the aerospace industry.
So Spirit Arrow was Boeing's factories.
They asked us for how to join composites.
When you make a composite airplane
you make these giant wing and fuselage parts
and they asked us for a better way to stick them together
because the joints were a place of failure.
And what we discovered was
instead of making a few big parts
if you make little loops of carbon fiber
and you reversibly link them in joints
and you do it in a special geometry
that balances being under constrained and over constrained
with just the right degrees of freedom.
We set the world record
for the highest modulus ultralight material
just by in effect making carbon fiber Lego.
So lightweight materials are crucial for energy efficiency.
This let us make the lightest weight
high modulus material.
We then showed that with just a few part types
we can tune the material properties
and then you can create really wild robots
that instead of having a tool the size of a jumbo jet
to make a jumbo jet,
you can make little robots
that walk on these cellular structures
to build the structures
where they error correct their position on the structure
and they navigate on the structure.
And so using all of that with NASA

we made morphing airplanes,
a former student Kenny,
Chang and Ben Jeanette made a morphing airplane
the size of NASA Langley's biggest wind tunnel.
With Toyota we've made super efficiency race cars.
We're right now looking at projects with NASA
to build these for things like space telescopes
and space habitats where the ribosome
I who I mentioned a little while back
can make an elephant one molecule at a time.
Ribosomes are slow.
They run at about one molecule a second
but ribosomes make ribosomes.
So you have thousands of them,
trillions of them and that makes an elephant.
In the same way these little assembly robots I'm describing
can make giant structures at heart
because the robot can make the robot.
So more recently to my students Amira and Miana
had a nature communication paper showing
how this robot can be made out of the parts it's making
so the robots can make the robots
so you build up the capacity of robotic assembly.
They can self-replicate.
Can you linger on what that robot looks like?
What is a robot that can walk along
and do error correction and what is a robot
that can self-replicate from the materials it is given?
What does that look like?
What are we talking about?
This is fascinating.
Yeah, the answer is different at different length scales.
So to explain that in biology,
primary structure is the code in the messenger RNA
that says what the ribosome should build.
Secondary structure are geometrical motifs
or things like helices or sheets.
Tertiary structures are functional elements
like electron donors or acceptors.
Quatenary structure is things like molecular motors
that are moving my mouth
or making the synapses work in my brain.
So there's that hierarchy of primary,

secondary, tertiary, quatenary.
Now what's interesting is if you wanna buy electronics today
from a vendor, there are hundreds of thousands of types
of resistors or capacitors or transistors, huge inventory.
All of biology is just made
from this inventory of 20 parts of the amino acids.
And by composing them, you can create all of life.
And so as part of this digitization of materials,
we're in effect trying to create something
like amino acids for engineering,
creating all of technology from 20 parts.
Let's see, as another discretion,
I helped start an office for science in Hollywood.
And there was a fun thing for the movie, The Martian,
where I did a program with Bill Nye and a few others
on how to actually build a civilization on Mars
that they described in a way that I like
as I was talking about how to go to Mars without luggage.
And at heart, it's sort of how to create life
in non-living materials.
So if you think about this primary secondary tertiary
quaternary structure, in my lab, we're doing that,
but on different length scales for different purposes.
So we're making micro robots out of like nanobricks
and to make the robots to build large scale structures
in space, the elements of the robots now are centimeters
rather than micrometers.
And so the assembly robots for the bigger structures
are there are the cells that make up the structure,
but then we have functional cells.
And so cells that can process and actuate,
each cell can like move one degree of freedom
or attach or detach or process.
Now, those elements I just described,
we can make out of the still smaller parts.
So eventually there's a hierarchy of the little parts
make little robots that make bigger parts of bigger robots
up through that hierarchy.
And that way you can move up the length scale.
Right, early on I tried to go in a straight line
from the bottom to the top,
and that ended up being a bad idea.
Instead, we're kind of doing all of these in parallel

and then they're growing together.
And so to make the larger scale structures,
like there's a lot of hype right now
about 3D printing houses
where you have a printer the size of the house.
We're now working on using swarms of these table scale
robots that walk on the structures to place the parts
much more efficiently.
That's amazing.
But you're saying you can't for now go from the very small
to the very large.
That'll come.
That'll come in stages.
Can we just link around this idea,
starting from Von Neumann's self-replicating automata
that you mentioned, it's just a beautiful idea.
So that's at the heart of all of this.
In the stack I described,
so one student will Langford made these micro robots
out of little parts that then we're using for me
on as bigger robots up through this hierarchy.
And it's really realizing this idea
of the self-reproducing automata.
So Von Neumann, when I complained about
the Von Neumann architecture,
it's not fair to Von Neumann
because he never claimed it as his architecture.
He really wrote about it in this one fairly dreadful memo
that led to all sorts of lawsuits and fights
and about the early days of computing.
He did beautiful work on reliable computation
and unreliable devices.
And towards the end of his life,
what he studied was how,
and I have to say this precisely,
how a computation communicates its own construction.
Yeah, so beautiful.
So a computation can store a description
of how to build itself.
But now there's a really hard problem,
which is if you have that in your mind,
how do you transfer it and wake up a thing
that then can contain it?

So how do you give birth to a thing
that knows how to make itself?
And so with Stan Ulam,
he invented cellular automata as a way to simulate these.
But that was theoretical.
Now the work I'm describing in my lab
is fundamentally how to realize it,
how to realize self-reproducing automata.
And so this is something Von Neumann thought very deeply
and very beautifully about theoretically.
And it's right at this intersection.
It's not communication or computation or fabrication.
It's right at this intersection
where communication and computation meets fabrication.
Now the reason self-reproducing automata
intellectually is so important
because this is the foundation of life.
This is really just understanding the essence of how to life.
And in effect, we're trying to create life
in non-living material.
The reason it's so important technologically
is because that's how you scale capacity.
That's how you can make an elephant from a ribosome
because the assemblers make assemblers.
So simple building blocks
that inside themselves contain the information
how to build more building blocks
and between each other construct arbitrarily complex objects.
Now let me give you the numbers.
So let me relate this to right now
we're living in AI mania explosion time.
Let me relate that to what we're talking about.
A hundred petaflop computer,
which is a current generation supercomputer,
not quite the biggest ones,
does 10 to the 17 ops per second.
Your brain does 10 to the 17 ops per second.
It has about 10 to the 15 synapses
and they run at about a hundred Hertz.
So as of a year or two ago,
the performance of a big computer matched a brain.
So you could view AI as a breakthrough,
but the real story is within about a year or two ago,

and let's see that the supercomputer has about 10
to the 15 transistors in the processors,
10 to the 15 transistors in the memory,
which is the synapses in your brain.
So the real breakthrough was the computers match
the computational capacity of a brain.
And so we'd be sort of derelict
if they couldn't do about the same thing.
But now the reason I'm mentioning that
is the chip fab making the supercomputer
is placing about 10 to the 10 transistors a second.
While you're digesting your lunch right now,
you're placing about 10 to the 18 parts per second.
There's an eight order of magnitude difference.
So in computational capacity, it's done, we've caught up,
but there's eight orders of magnitude difference
in the rate at which biology can build
versus state-of-the-art manufacturing can build.
And that distinction is what we're talking about.
That distinction is not analog,
but this deep sense of digital fabrication,
of embodying codes in construction.
So a description doesn't describe a thing,
but the description becomes the thing.
So you're saying, I mean, this is one of the cases
you're making and that this is this third revolution.
We've seen the Moore's Law in communication,
we've seen the Moore's Law-like type of growth
in computation and you're anticipating,
we're going to see that in digital fabrication.
Can you actually first of all describe
what you mean by this term, digital fabrication?
So the casual meaning is the computer controls
the tool to make something.
And that was invented when MIT stole it in 1952.
There's the deep meaning of what the ribosome does,
of a digital description doesn't describe a thing,
a digital description becomes the thing.
That's the path to the Star Trek replicator,
and that's the thing that doesn't exist yet.
Now, I think the best way to understand
what this roadmap looks like is to now bring in FAB Labs
and how they relate to all of this.

What are FAB Labs?
So here's a sequence.
With colleagues, I accidentally started a network
of what's now 2,500 digital fabrication community labs
called FAB Labs right now in 125 countries,
and they double every year and a half.
That's called Lassa's Law after Sherry Lasseter,
who I'll explain.
So here's the sequence.
We started Center for Bits and Atoms
to do the kind of research we're talking about.
We had all of these machines,
and then had a problem,
it would take a lifetime of classes
to learn to use all the machines.
So with colleagues who helped start CBA,
we began a class modestly called
How to Make Almost Anything.
And there's no big agenda.
It was just, it was aimed at a few research students
to use the machines.
And it was completely unprepared
for the first time we taught it.
We were swamped by every year
since hundreds of students tried to take the class.
It's one of the most oversubscribed classes at MIT.
Students would say things like,
can you teach this at MIT?
It seems too useful.
It's just how to work these machines.
And the students in the class,
I would teach them all the skills to use all these tools,
and then they would do projects integrating them.
And they're amazing.
So Kelly was a sculptor, no engineering background.
Her project was,
she made a device that saves up screams when you're mad
and plays them back later.
And saves up screams when you're mad
and plays them back later.
You scream into this device
and it deadens the sound, records it,
and then when it's convenient releases your scream.

Can we just pause on the brilliance of that invention?
Creation, the art, I don't know.
The brilliance, who is this that created this?
Kelly Dobson.
Kelly Dobson.
Gone on to do a number of interesting things.
Median, who's gone on to do a number of interesting things,
made a dress instrumented with sensors and spines.
And when somebody creepy comes close,
it would defend your personal space.
Also very easy.
Another project early on was a web browser for parrots,
which have the cognitive ability of a young child
and lets parrots surf the internet.
You know, another was an alarm clock you wrestle with
and prove you're awake.
And what connects all of these is,
so MIT made the first real-time computer, the whirlwind.
That was transistorized as the TX.
The TX was spun off from MIT as the PDP.
PDPs were the mini computers that created the internet.
So outside MIT was DEC, Prime, Wang, Data General,
the whole mini computer industry.
The whole computing industry was there
and it all failed when computing became personal.
Ken Olson, the head of digital,
famously said you don't need a computer at home.
There's a little background to that,
but DEC completely missed computing became personal.
So I mentioned all of that because I was asking
how to do digital fabrication, but not really why.
The students in this how to make class were showing me
that the killer app of digital fabrication
is personal fabrication.
Yeah, how do you jump to the personal fabrication?
So Kelly didn't make the screen body
because it was for a thesis.
She wasn't writing a research paper.
It wasn't a business model.
It was because she wanted one.
It was personal expression going back
to me in vocational school.
It was personal expression in these new means of expression.

So that's happened every year since.
It literally is called,
the course is literally called
how to make almost anything.
A legendary course at MIT every year.
And it's grown to multiple labs at MIT
with as many people involved in teaching as taking it.
And there's even a Harvard lab for the MIT class.
What have you learned about humans
colliding with the fab lab,
about what the capacity was to be creative and to build?
I mentioned Marvin.
Another mentored MIT sadly no longer living
is Seymour Papert.
So Papert studied with Piaget.
He came to MIT to get access to the early computer.
Piaget was a pioneer in how kids learn.
Papert came to MIT to get access to the early computers
with the goal of letting kids play with them.
Piaget helps show kids are like scientists.
They learn as scientists
and it gets kind of throttled out of them.
Seymour wanted to let kids have a broader landscape to play.
Seymour's work led with Mitch Resnick
to Lego, Logo, Mindstorms, all of that stuff.
As fab lab spread,
and we started creating educational programs
for kids in them,
Seymour said something really interesting.
He made a gesture.
He said, it was a thorn in his side
that they invented what's called the turtle,
a early robot kids could program
to connect it to a mainframe computer.
Seymour said, the goal was not for the kids
to program the robot,
it was for the kids to create the robot.
And so in that sense, the fab labs,
which for me were just this accident,
he described this sort of this fulfillment
of the arc of kids learned by experimenting.
It was to give them the tools to create,
not just assemble things and program things,

but actually create.
So coming to your question,
what I've learned is MIT,
a few years back, somebody added up businesses
from spun off from MIT
and it's the world's 10th economy.
It falls between India and Russia.
And I view that in a way as a bad number
because it's only a few thousand people
and these aren't uniquely the 4,000 brightest people,
it's just a productive environment for them.
And what we found is in rural Indian villages
in African shantytowns and Arctic hamlets,
I find exactly precisely that profile.
So Ling sighed at a few hours above Tromsø,
way above the Arctic circles,
it's so far north the satellite dishes,
look at the ground, not the sky.
Hans Christian in the lab was considered a problem
in the local school
because they couldn't teach him anything.
I showed him a few projects.
Next time I came back,
he was designing and building little robot vehicles.
And in South Africa,
in I mentioned Soshengovie,
in this apartheid township,
the local technical institute taught kids
how to make bricks and fold sheets, it was punitive.
But Chepiso in the fab lab
was actually doing all the work of my MIT classes.
And so over and over we found precisely
the same kind of bright invent of creativity.
And historically the answer was,
go, you're smart, go away.
It's sort of like me in vocational school.
But in this lab network,
what we could then do is in effect bring the world to them.
Now let's look at the scaling of all of this.
So there's one earth,
1,000 cities, a million towns,
a billion people, a trillion things.
There was one whirlwind computer,

MIT made the first real-time computer.
There were thousands of PDPs.
There were millions of hobbyist computers
that came from that,
billions of personal computers,
trillions of internet of things.
So now if we look at this fab lab story,
1952 was the NC mill.
There are now thousands of fab labs.
And the fab lab costs exactly the same
cost and complexity of the mini computer.
So on the mini computer, it didn't fit in your pocket.
It filled the room.
But video games, email, word processing,
really anything you do at the internet,
anything you do with a computer today happened at that era
because it got on the scale of a work group,
not a corporation.
In the same way, fab labs are like the mini computers
inventing how does the world work
if anybody can make anything.
Then if you look at that scaling,
fab labs today are transitioning from buying a machine
to make machines, making machines.
So we're transitioning to you can go to a fab lab,
not to make a project, but to make a new machine.
So we talked about the deep sense of self-replication.
There's a very practical sense of fab lab machines
making fab lab machines.
And so that's the equivalent of the hobbyist computer era,
what it's called the Altair historically.
Then the work we spent a while talking about
about assemblers and self-assemblers,
that's the equivalent of smartphones
and internet of things.
That's when, so the assemblers are like the smartphone
where a smartphone today has the capacity
of what used to be a supercomputer in your pocket.
And then the smart thermostat on your wall
has the power of the original PDP computer,
not metaphorically, but literally,
and now there's trillions of those in the same sense
that when we finally merge materials

with the machines in the self-assembly,
that's like the internet of things stage.
But here's the important lesson.
If you look at the computing analogy,
computing expanded exponentially,
but it really didn't fundamentally change.
The core things happened in that transition
in the mini-computer era.
So in the same sense, the research now,
we spent a while talking about
is how we get to the replicator.
Today you can do all of that
if you close your eyes and view the whole fab lab
as a machine, in that room you can make almost anything,
but you need a lot of inputs.
Bit by bit, the inputs will go down
and the size of the room will go down
as we go through each of these stages.
So how difficult is it to create a self-replicating assembler,
self-replicating machine that builds copies of itself
or builds more complicated version of itself,
which is kind of the dream towards which you're pushing
in a generic arbitrary sense?
I had a student Nadia Peek with Jonathan Ward,
who for me started this idea of how do we use
the tools in my lab to make the tools in the lab?
In a very clear sense,
they are making self-reproducing machines.
So one of the really cool things that's happened
is there's a whole network of machine builders
around the world.
So there's Danielle and now in Germany
and Jens in Norway.
And each of these people has learned the skills
to go into a fab lab and make a machine.
And so we've started creating a network of super fab,
so the fab lab can make a machine,
but it can't make a number of the precision parts
of the machine.
So in places like Bhutan or Kerala in the south of India,
we started creating super fab labs
that have more advanced tools to make the parts
of the machines so that the machines themselves

become even cheaper.
So that is self-reproducing machines,
but you need to feed it things like bearings
or microcontrollers, they can't make those parts.
But other than that, they're making their own things.
And I should note as a footnote,
the stack I described of computers controlling machines
to machine making machines to assemblers to self-assemblers,
view that as fab one, two, three, four.
So we're transitioning from fab one to fab two,
and the research in the lab is three and four.
At this fab two stage, a big component of this
is sustainability in the material feedstocks.
So Alicia, colleague in Chile is leading a great effort
looking at how you take forest products
and coffee grounds and seashells
and a range of locally available materials
and produce the high tech materials that go into the lab.
So all of that is machine building today.
Then back in the lab, what we can do today
is we have robots that can build structures
and can assemble more robots that build structures.
We have finer resolution robots
that can build micromechanical systems.
So robots that can build robots that can walk and manipulate.
And we're just now, we have a project
at the layer below that where there's endless attention today
to billion dollar chip fab investments.
But a really interesting thing we pass through
is today the smallest transistors you can buy
as a single transistor, just commercially for electronics
is actually the size of an early transistor
in an integrated circuit.
So we're using these machines making machines
making assemblers to place those parts
to not use a billion dollar chip fab
to make integrated circuits
but actually assemble little electronic components.
So have a fine enough, precise enough actuators
and manipulators that allow you to place these transistors.
Right, that's a research project in my lab
called DICE on Discrete Assembly of Integrated Electronics.
And we're just at the point

to really start to take seriously this notion
of not having a chip fab make integrated electronics
but having not a 3D printer
but a thing that's a cross between a pick in place
makes circuit boards in 2D.
The 3D printer extrudes in 3D.
We're making sort of a micromanipulator
that acts like a printer
but it's placing to build electronics in 3D.
But this micromanipulator is distributed
so there's a bunch of them or is this one centralized thing?
So that's why that's a great question.
So I have a prize that's almost but not been claimed
for the students whose thesis can walk out of the printer.
Oh, nice.
So you have to print the thesis with the means
to exit the printer and it has to contain
its description of the thesis that says how to do that.
It's a really good, I mean, it's a fun example
of exactly the thing we're talking about.
And I've had a few students almost get to that.
And so in what I'm describing, there's the stack
where we're getting closer
but it's still quite a few years to really go from us.
So there's a layer below the transistors
where we assemble the base materials
that become the transistor.
We're now just at the edge of assembling the transistors
to make the circuits.
We can assemble the micro parts to make the micro robots.
We can assemble the bigger robots.
And in the coming years,
we'll be patching together all of those scales.
So do you see a vision of just endless billions of robots
at the different scales, self-assembling,
self-replicating and building complicated structures?
Yes.
And the but to the yes, but is let me clarify two things.
One is that immediately raises King Charles' fear
of gray goo, of runaway mutant self-reproducing things.
The reason why there are many things
I can tell you to worry about,
but that's not one of them,

is if you want things to autonomously self-reproduce
and take over the world,
that means they need to compete with nature
on using the resources of nature, of water and sunlight.
And in light of everything I'm describing,
biology knows everything I told you.
Every single thing I explained,
biology already knows how to do.
What I'm describing isn't new for biology,
it's new for non-biological systems.
So in the digital era, the economic win
ended up being centralized, the big platforms,
in this world of machines that can make machines.
I'm asked, for example, what's the killer opportunity?
Who's gonna make all the money?
Who to invest in?
But if the machine can make the machine,
it's not a great business to invest in the machine.
In the same way that if you can think globally
but produce locally,
then the way the technology goes out into society
isn't a function of central control
but is fundamentally distributed.
Now, that raises an obvious kind of concern,
which is, well, doesn't this mean
you could make bombs and guns and all of that?
The reason that's much less of a problem than you would think
is making bombs and guns and all of that
is a very well-met market need.
Anywhere we go, there's a fine supply chain for weapons.
Now, hobbyists have been making guns for ages
and guns are available just about anywhere.
So you could go into the lab and make a gun.
Today, it's not a very good gun
and guns are easily available.
And so generally, we've run these lab in war zones.
What we find is people don't go to them to make weapons,
which you can already do anyway.
It's an alternative to making weapons.
Coming back to your question,
I'd say the single most important thing I've learned
is the greatest natural resource of the planet
is this amazing density of bright invent of people

whose brains are underused.
And you could view the social engineering of this lab work
as creating the capacity for them.
And so in the end,
the way this is gonna impact society
isn't gonna be command and control.
It's how the world uses it.
And it's been really gratifying for me to see just how it does.
Yeah, but what are the different ways
the evolution of the exponential scaling
of digital fabrication can evolve?
So you said, yeah, self-replicating nanobots, right?
This is the gray goo fear.
It's a caricature of a fear,
but nevertheless, there's interesting,
just like you said, spam and all these kinds of things
that came with the scaling of communication and computation.
What are the different ways
that malevolent actors will use this technology?
Yeah, well, first, let me start with a benevolent story,
which is trash is an analog concept.
There's no trash in a forest.
All the parts get disassembled and reused.
Trash means something doesn't have enough information
to tell you how to reuse it.
It's as simple as there's no trash in a Lego room.
When you assemble Lego,
the Lego bricks have enough information to disassemble them.
So as you go through this Fab1234 story,
one of the implications of this transition
from printing to assembling.
So the real breakthrough technologically
isn't additive versus subtractive,
which is a subject of a lot of attention and hype.
3D printers are useful.
We spun off companies like Form Labs,
led by Max for 3D printing.
But in a Fab lab, it's one of maybe 10 machines.
It's used, but it's only part of the machines.
The real technological change is when we go from printing
and cutting to assembling and disassembling.
But that reduces inventories of hundreds of thousands
of parts to just having a few parts

to make almost anything.
It reduces global supply chains
to locally sourcing these building blocks.
But one of the key implications
is it gets rid of technological trash
because you can disassemble and reuse the parts,
not throw them away.
And so initially that's of interest for things
at the end of long supply chains,
like satellites on orbit.
But one of the things coming is eliminating technical trash
through reuse of the building blocks.
So like when you think about 3D printers,
you're thinking about addition and subtraction
when you think about the other options available to you
in that parameter space as you call it.
That's going to be assembly,
disassembly, cutting, you said.
So the 1952 NC mill was subtractive, your movement here.
And 3D printing additive,
and there's a couple of claims
to the invention of 3D printing,
that's closer to what's called net shape,
which is you don't have to cut away the material
you don't need, you just put material where you do need it.
And so that's the 3D printing revolution.
But there are all sorts of limitations on 3D printing
to the kinds of materials you can print,
the kind of functionality you can print.
We're just not going to get to making a,
everything in a cell phone on a single printer.
But I do expect to make everything in a cell phone
with an assembler.
And so instead of printing and cutting,
technologically it's this transition
to assembling and disassembling,
going back to Shannon and Van Neumann,
going back to the ribosome four billion years ago.
Now you come to malevolent.
Let me tell you a story about,
I was doing a briefing for the National Academy
of Sciences group that advises the intelligence communities.
And I talked about the kind of research we do.

And at the very end, I showed a little video clip
of Valentina and Ghana making a local girl
making surface mount electronics in the fab lab.
And I showed that to this room full of people.
One of the members of the intelligence community
got up, livid and said,
how dare you waste our time showing us a young girl
in an African village making surface mount electronics.
We're looking at, we need to know about disruptive threats
to the future of the United States.
And somebody else got up in the room and yelled at him
and you idiot, I can't think of anything more important
than this, but for two reasons.
One reason was because if we rely on like
informational superiority in the battlefield,
it means other people could get access to it.
But this intelligence person's point, bless him,
wasn't that it was getting at the root causes of conflict
is if this young girl in an African village
could actually master surface mount electronics,
it changes some of the most fundamental things
about recruitment for terrorism,
impact of economic migration,
basic assumptions about an economy.
It's just existential for the future of the planet.
But we've just lived through a pandemic.
I would love to linger on this
because the possibilities that are positive are endless,
but the possibilities are negative
are still nevertheless extremely important.
What's both positive and negative?
What do you do with a large number of general assemblers?
With the fab lab, you could roughly make a bio lab
then learn biotechnology.
Now that's terrifying because making self-reproducing
gray goo that outcompetes biology,
I consider doom because biology knows everything
I'm describing and it's really good at what it does.
How to grow almost anything,
you learn skills in biotechnology
that would let you make serious biological threats.
And when you combine some of the innovations
you see with large language models,

some of the innovations you see with AlphaFold,
so applications of AI for designing biological systems
for writing programs,
which you can with large language models increasingly.
So there seems to be an interesting dance here
of automating the design stage of complex systems using AI
and then that's the bits.
And you can leap now, the innovations you're talking about,
you can leap from the complex systems in the digital space
to the printing, to the creation, to the assembly
at scale of complex systems in the physical space.
Yeah, so something to be scared about is
a fab lab can make a bio lab,
a bio lab can make biotechnology,
somebody could learn to make a virus.
That's scary.
That's unlike some of the things I said,
I don't worry about that's something
I really worry about that is scary.
Now, how do you deal with that?
Prior threats we dealt with command and control.
So like, early color copiers had unique codes
and you could tell which copier made them,
eventually you couldn't keep up with that.
There was a famous meeting at a Silamar
in the early days of recombinant DNA
where that community recognized the dangers
of what it was doing and put in place a regime
to help manage it.
And so that led to the kind of research management.
MIT has an office that supervises research
and it works with the national office.
That works if you can identify who's doing it and where.
It doesn't work in this world we're describing.
So anybody could do this anywhere.
And so what we found is you can't contain this,
it's already out.
You can't forbid because there isn't command and control.
The most useful thing you can do
is provide incentives for transparency.
But really the heart of what we do is
you could do this by yourself in a basement
for nefarious reasons,

or you could come into a place in the light
where you get help and you get community
and you get resources.
And there's an incentive to do it in the open,
not in the dark.
And that might sound naive,
but in the sort of places we're working,
again, bad people do bad things in these places already,
but providing openness and providing transparency
is a key part of managing these.
And so it transitions from regulating risks
as regulation to soft power to manage them.
So there's so much potential for good,
so much capacity for good that Fab Labs
and the ability and the tools of creation
really unlock that potential.
Yeah, and I don't say that as sort of dewy-eyed naive.
I say that empirically from just years
of seeing how this plays out in communities.
I wonder if it's the early days of personal computers though
before we get spam, right?
In the end, most fundamentally,
literally the mother of all problems
is who designed us.
So assume success
in that we're gonna transition to the machines,
making the machines,
and all of these new sort of social systems we're describing
will help manage them and curate them and democratize them.
If we close the gap I just let off with
of 10 to the 10 to 10 to the 18 between Chip Fab and you,
we're ultimately in marrying communication,
computation, and fabrication
gonna be able to create unimaginable complexity.
And how do you design that?
And so I'd say the deepest of all questions
that I've been working on is,
goes back to the oldest part of our genome.
So in our genome, what are called Hox genes,
and these are morphogenes,
and nowhere in your genome is the number five.
It doesn't store the fact that you have five fingers.
And what it stores is what's called a developmental program.

It's a series of steps.
And the steps have the character of like,
grow up a gradient or break symmetry.
And at the end of that developmental program,
you have five fingers.
So you are stored not as a body plan,
but as a growth plan.
And there's two reasons for that.
One reason is just compression.
Billions of genes can place trillions of cells.
But the much deeper one is evolution
doesn't randomly perturb.
Almost anything you did randomly in the genome
would be fatal or inconsequential, but not interesting.
But when you modify things in these developmental programs,
you go from like webs for swimming to fingers,
or you go from walking to wings for flying.
It's a space in which search is interesting.
So this is the heart of the success of AI.
In part, it was the scaling we talked about a while ago.
And in part, it was the representations
for which search is effective.
AI has found good representations.
It hasn't found new ways to search,
but it's found good representations of search.
You're saying that's what biology,
that's what evolution has done
is creative representations,
structures, biological structures
through which search is effective.
And so the developmental programs in the genome
beautifully encapsulate the lessons of AI.
And it's molecular intelligence.
It's AI embodied in our genome.
It's every bit as profound as the cognition in our brain,
but now this is sort of thinking in molecular thinking
in how you design.
And so I'd say the most fundamental problem we're working on
is it's kind of tautological
that when you design a phone, you design the phone.
You represent the design of the phone.
But that actually fails when you get
to the sort of complexity that we're talking about.

And so there's this profound transition to come.
Once I can have self-reproducing assemblers
placing 10 to the 18 parts,
you need to not sort of metaphorically, but create life
in that you need to learn how to evolve.
But evolutionary design
has a really misleading, trivial meaning.
It's not as simple as you randomly mutate things.
It's this much more deep embodiment of AI and morphogenesis.
Is there a way for us to continue
the kind of evolutionary design that led us to this place
from the early days of bacteria, single cell organism,
to ribosomes, and the 20 amino acids?
You mean for human augmentation or?
For life, augment.
I mean, what would you call assemblers
that are self-replicating and placing parts?
What is that?
The dynamic complex things built with digital fabrication,
what is that?
That's life.
Yeah, so ultimately, absolutely,
if you add everything I'm talking about,
it's building up to creating life in non-living materials.
And I don't view this as copying life.
I view it as driving life.
I didn't start from how does biology work
and then I'm gonna copy it.
I start from how to solve problems
and then it leads me to, in a sense, rediscover biology.
So if you go back to Valentina in Ghana
making her circuit board,
she still needs a chip fab very far away
to make the processor on her circuit board.
For her to make the processor locally,
for all the reasons we described,
you actually need the deep things we were just talking about.
And so it really does lead you.
So there's a wonderful series of books by Gingery.
Book one is how to make a charcoal furnace
and at the end of book seven, you have a machine shop.
So it's sort of how you do your own
personal industrial revolution.

ISRU is what NASA calls in situ resource utilization.
And that's how do you go to a planet
and create a civilization?
ISRU has essentially assumed Gingery.
You go through the industrial revolution
and you create the inventory of 100,000 resistors.
What we're finding is the way,
the minimum building blocks for a civilization
is roughly 20 parts.
So what's interesting about the amino acids
is they're not interesting.
They're hydrophobic or hydrophilic, basic or acidic.
They have typical but not extremal properties
but they're good enough you can combine them to make you.
What this is leading towards is technology
doesn't need enormous global supply chains.
It just needs about 20 properties you can compose
to create all technology as the minimum building blocks
for a technological civilization.
So there's going to be 20 basic building blocks
based on which the self-replicating assemblers can work.
Right, and I say that not philosophically,
just empirically, that's where it's heading.
And I like thinking about how you bootstrap
a civilization on Mars, that problem.
There's a fun video on bonus material for the movie
where we're with a neat group of people we talk about it
because it has really profound implications
back here on earth about how we live sustainably.
What is that civilization on Mars looks like
that's using ISRU,
that's using these 20 building blocks and does self-assembly?
Yeah, go through primary secondary tertiary, quaternary.
You extract properties like conducting, insulating,
semi-conducting, magnetic, dielectric, flexural.
These are the kind of roughly 20 properties.
With those, those are enough for us to assemble logic
and they're enough for us to assemble actuation.
With logic and actuation, we can make micro-robots.
The micro-robots can build bigger robots.
The bigger robots can then take the building block materials
and make the structural elements
that you then do to make construction

and then you boot up through the stages
of a technological civilization.
By the way, where in the span of logic and actuation
did the sensing come in?
Oh, I skipped over that.
But my favorite sensor is a step response.
So if you just make a step
and measure the response to the electric field,
that ranges from user interfaces
to positioning to material properties.
And if you do it at higher frequencies, you get chemistry.
And you can get all of that
just from a step in an electric field.
So for example, once you have time resolution in logic,
something as simple as two electrodes
let you do amazingly capable sensing.
So we've been talking about all the work I do.
There's a story about how it happens.
Where do ideas come from and...
That's an interesting story.
Where do ideas come from?
So I had mentioned Vannevar Bush
and he wrote a really influential thing
called the Endless Frontier.
So science won World War II.
The more known story is Nuclear Bombs.
The less well-known story is the Rad Lab.
So at MIT, an amazing group of people invented radar
which is really credited as winning the war.
So after the war, a grand old man from MIT
and it was charged with science won the war,
how do we maintain that edge?
And the report he wrote led to
the National Science Foundation
and the modern notion we take for granted
but didn't really exist before then
of public funding of research or research agencies.
In it, he made again,
what I consider an important mistake, which is
he described basic research leads to applied research,
leads to applications, leads to commercialization,
leads to impact.
And so we need to invest in that pipeline.

The reason I considered a mistake
is almost all of the examples we've been talking about
in my lab went backwards,
that the basic research came from applications.
And further, almost all of the examples
we've been talking about came fundamentally from mistakes.
So essentially everything I've ever worked on has failed,
but in failing, something better happened.
So the way I like to describe it is ready, aim, fire
is you do your homework, you aim carefully
at a target you want to accomplish.
And if everything goes right,
you then hit the target and succeed.
What I do you can think of as ready, fire, aim.
So you do a lot of work to get ready,
then you close your eyes
and you don't really think about where you're aiming,
but you look very carefully at where you did aim,
where you aim after you fire.
And the reason that's so important is
if you do ready, aim, fire,
there's the best you can hope is hit what you aim at.
So let me give you some examples.
Because this is a source of great-
You're full of good lines today.
Source of great frustration.
So I mentioned the early quantum computing.
So quantum computing is this power of using quantum mechanics
to make computers that for some problems
are dramatically more powerful than classical computers.
Before it started,
there was a really interesting group of people
who knew a lot about physics and computing
that were inventing what became quantum computing
before it was clear anything.
There was an opportunity there.
It was just studying how those relate.
Here's how it fits to the ready, fire, aim.
In that I was doing really short-term work in my lab
on shoplifting tags.
On, this was really before there was modern RFID.
And so how you put tags in objects to sense them.
Something we just take for granted commercially.

And there was a problem of how you can sense
multiple objects at the same time.
And so I was studying how you can remotely sense materials
to make low-cost tags
that could let you distinguish multiple objects simultaneously.
To do that, you need non-linearity
so that the signal is modulated.
And so I was looking for material sources of non-linearity
and that led me to look at how nuclear spins interact.
Just for spin resonance.
The sort of things you use when you go in an MRI machine.
And so I was studying how to use that.
And it turns out that it was a bad idea.
You couldn't remotely use it for shoplifting tags.
But I realized you could compute.
And so with a group of colleagues thinking
about early quantum computing,
like David D. Vincenzo and Charlie Bennett,
was articulating what are the properties you need to compute.
And then looking at how to make the tags.
It turns out the tags were a terrible idea
for sensing objects in a supermarket checkout.
But I realized they were computing.
So with Ike Chuang and a few other people,
we realized we could program nuclear spins to compute.
And so that's what we use to do Grover's search algorithm.
And then it was used for a Shor's factoring algorithm.
And it worked out the systems we did it in nuclear magnetic
resonance don't scale beyond a few qubits.
But the techniques have lived on.
And so all the current quantum computing techniques
grew out of the ways we would talk to these spins.
But I'm telling this whole story
because it came from a bad way to make a shoplifting tag.
Starting with an application mistakes
led to the fundamental science, fundamental science.
I mean, can you just link on that?
I mean, just using nuclear spins to do computation
and that like what gave you the guts to try to think through this?
From a digital fabrication perspective, actually,
how to leap from one to the other?
I wouldn't call it guts.
I would call it collaboration.

So at IBM, there was this amazing group of like I mentioned
Charlie Bennett and David DiVincenzo and Ralph Landau
and Nabil Amir.
And these were all gods of thinking about physics
and computing.
So I yelled at the whole computer industry
being based on a fiction, metropolis.
Programmers frolicking in the garden
while somebody moves levers in the basement.
There's a complete parallel history
of Maxwell de Boltzmann to Zillard to Landau or to Bennett.
Most people won't know most of these names.
But this whole parallel history thinking deeply
about how computation and physics relate.
So I was collaborating with that whole group of people.
And then at MIT, I was in this high traffic environment.
I wasn't deeply inspired to think about better ways
to detect shoplifting tags, but stumbled
across companies that needed help with that
and was thinking about it.
And then I realized those two worlds intersected.
And we could use the failed approach
for the shoplifting tags to make early quantum computing
algorithms.
And this kind of stumbling is fundamental to the fab lab
idea, right?
Right.
Here's one more example.
With a student, Manu, we talked about ribosomes.
And I was trying to build a ribosome that
worked on fluids so that I could place the little parts we're
talking about.
And it kept failing because bubbles would come into our system
and the bubbles would make the whole thing stop working.
And we spent about half a year trying to get rid of the bubbles.
Then Manu said, wait a minute.
The bubbles are actually better than what we're doing.
We should just use the bubbles.
And so we invented how to do universal object
logic with little bubbles and fluid.
OK.
You have to explain this microfluidic bubble logic,
please.

How does this work?
So.
It's super interesting.
Yeah.
And so I'll come back and explain it.
But what it led to was we showed fluids could do.
It'd been known fluid could do logic,
like your old automobile transmissions do logic.
But that's macroscopic.
It didn't work at little scales.
We showed with these bubbles we could do it at little scales.
Then I'm going to come back and explain it.
But what came out of that is Manu then
showed you could make a 50-cent microscope using little bubbles.
And then the techniques we developed
are what we used to transplant genomes to make synthetic life.
All came out of the failure of trying to make the genome,
the ribosome.
Now, so the way the bubble logic works
is in a little channel, fluid at small scales
is fairly viscous.
It's sort of like pushing jello, think of it as.
If a bubble gets stuck, the fluid has to detour around it.
So now imagine a channel that has two wells and one bubble.
If the bubble is in one well, the fluid
has to go in the other channel.
If the fluid is in the other well,
it has to go in the first channel.
So the position of the bubble can switch.
It's a switch.
It can switch the fluid between two channels.
So now we have one element, a switch.
And it's also a memory because you
can detect whether or not a bubble is stored there.
Then if two bubbles meet, if you have two channels crossing,
a bubble can go through one way or a bubble
can go through the other way.
But if two bubbles come together,
then they push on each other and one goes one way
and one goes the other way.
That's a logic operation.
That's a logic gate.
So we now have a switch.

We have a memory and we have a logic gate.
And that's everything you need to make a universal computer.
I mean, the fact that you did that with bubbles
and microfluid just kind of brilliant.
Well, so to stay with that example,
what we proposed to do was to make a fluidic ribosome.
And the project crashed and burned.
It was a disaster.
This is what came out of it.
And so it was precisely ready fire aim
in that we had to do a lot of homework
to be able to make these microfluidic systems.
The fire part was we didn't think too hard
about making the ribosome.
We just tried to do it.
The aim part was we realized the ribosome failed
but something better had happened.
And if you look all across research funding,
research management, it doesn't anticipate this.
So fail fast is familiar,
but fail fast tends to miss ready and aim.
You can't just fail.
You have to do your homework before the fail part
and you have to do the aim part after the fail part.
And so the whole language of research
is about like milestones and deliverables.
That works when you're going down a straight line,
but it doesn't work for this kind of discovery.
And to leap to something you said that's really important
is I view part of what the fab lab network is doing
is giving more people the opportunity to fail.
You've said that geometry is really important in biology.
What does fabrication biology look like?
Why is geometry important?
So molecular biology is dominated by geometry.
That's why the protein folding is so important
that the geometry gives the function.
And there's this hierarchical construction
of as you go through primary secondary tertiary quaternary,
the shapes of the molecules
make the shape of the molecular machines.
And they really are exquisite machines.
If you look at how your muscles move,

if you were to see a simulation of it,
it would look like a improbable science fiction cyborg world
of these little walking robots that walk on a discrete lattice
that they're really exquisite machines.
And then from there, there's this whole hierarchical stack
of once you get to the top of that,
you then start making organelles that make cells
that make organs through the stack of that hierarchy.
Just stepping back, does it amaze you
that from small building blocks
where amino acids you mentioned, molecules,
let's go to the very beginning of hydrogen and helium
at the start of this universe,
that we're able to build up such complex
and beautiful things like our human brain?
So, studying thermodynamics,
which is exactly the question of,
that batteries run out and need recharging,
equipment, cars get old and fail,
yet life doesn't.
And that's why there's a sense
in which life seems to violate thermodynamics,
although of course it doesn't.
It seems to resist the March towards entropy somehow.
Right, and so Maxwell,
who helped give rise to the science of thermodynamics,
posited a problem that was so infuriating,
it led to a series of suicides.
There was a series of advisors and advisees,
three in a row that all ended up committing suicide
that happened to work on this problem.
And Maxwell's demon is this simple but infamous problem
where right now in this room we're surrounded by molecules
and they run at different velocities.
Imagine a container that has a wall
and it's got gas on both sides and a little door.
And if the door is a molecular sized creature
and it could watch the molecules coming,
and when a fast molecule is coming it opens the door,
when a slow molecule is coming it closes the door.
After it does that for a while,
one side is hot, one is cold.
When something is hot and is cold, you can make an engine.

And so you close that, you make an engine and you make energy.
So the demon is violating thermodynamics
because it's never touching the molecule,
yet by just opening and closing the door
it can make arbitrary amounts of energy and power a machine.
And in thermodynamics you can't do that.
So that's Maxwell's demon.
That problem is connected to everything
we just spoke about for the last few hours.
So Leo Zalard around early 1900s
was a deep physicist who then had a lot to do
with also post-war anti-nuclear things,
but he reduced Maxwell's demon to a single molecule.
So the molecule, there's only one molecule
and the question is which side of the partition is it on.
That led to the idea of one bit of information.
So Shannon credited Zalard's analysis of Maxwell's demon
for the invention of the bit.
For many years, people tried to explain Maxwell's demon
by like the energy in the demon looking at the molecule
or the energy to open and close the door
and nothing ever made sense.
Finally, Ralph Landauer,
one of the colleagues I mentioned at IBM,
finally solved the problem.
He showed that you can explain Maxwell's demon
by you need the mind of the demon.
When the demon open and closes the door,
as long as it remembers what it did,
you can run the whole thing backwards.
But when the demon forgets,
then you can't run it backwards
and that's where you get dissipation
and that's where you get the violation of thermodynamics.
And so the explanation of Maxwell's demon
is that it's in the demon's brain.
So then Ralph's colleague, Charlie at IBM,
then shock Ralph by showing you can compute
with arbitrarily low energy.
So one of the things that's not well covered
is the big computers used for big machine learning,
the data centers use tens of megawatts of power,
they use as much power as a city.

Charlie showed you can actually compute
with arbitrarily low amounts of energy
by making computers that can go backwards
as well as forwards.
And what limits the speed of the computer
is how fast you want an answer
and how certain you want the answer to be.
But we're orders of magnitude away from that.
So I have a student Cameron working with Lincoln Labs
on making superconducting computers
that operate near this land hour limit
that are orders of magnitude more efficient.
So stepping back to all of that,
that whole tour was driven by your question about life.
And right at the heart of it is Maxwell's demon.
Life exists because it can locally violate thermodynamics.
It can locally violate thermodynamics
because of intelligence.
And it's molecular intelligence
that I would even go out on a limb to say,
we can already see we're beginning to come to the end
of this current AI phase.
So depending on how you count,
this is I'd say the fifth AI boom bus cycle.
And you can already, it's exploding,
but you can already see where it's heading,
how it's going to saturate what happens on the far side.
The big thing that's not yet on horizons
is embodied AI, molecular intelligence.
So to step back to this AI story,
there was automation and that was gonna change everything.
Then there were expert systems.
There was then the first phase of the neural network systems.
There have been about five of these.
In each case on the slope up,
it's gonna change everything.
Each case what happens is on the slope down,
we sort of move the goal posts
and it becomes sort of irrelevant.
So a good example is going up,
computer chess was gonna change everything.
Once computers could play chess,
that fundamentally changes the world.

Now on the downside, computers play chess.
Winning at chess is no longer seen as a unique human thing,
but people still play chess.
This new phase is gonna take a new chunk of things
that we thought computers couldn't do.
Now computers will be able to do,
they have roughly our brain capacity,
but we'll keep thinking as well as computers.
And as I described,
while we've been going through these five boom busts,
if you just look at the numbers of ops per second,
bits storage, bits of IO,
that's the more interesting one.
That's been steady and that's what finally caught up to people.
But as we've talked about a couple of times,
there's eight orders of magnitude to go,
not in the intelligence in the transistors or in the brain,
but in the embodied intelligence,
in the intelligence in our body.
So the intelligent constructions of physical systems
that would embody the intelligence
versus contain it within the computation.
Right, and there's a brain centrism
that assumes our intelligence is centered in our brain.
And in endless ways in this conversation,
we've been talking about molecular intelligence.
Our molecular systems do a deep kind
of artificial intelligence.
All the things you think of as artificial intelligence does
in representing knowledge, storing knowledge,
searching over knowledge, adapting to knowledge,
our molecular systems do,
but the output isn't just a thought, it's us.
It's the evolution of us.
And the real horizon to come is now embodying AI,
of not just a processor and a robot,
but building systems that really can grow and evolve.
So we've been speaking about this boundary
between bits and atoms.
So let me ask you about one of the big mysteries
of consciousness.
Do you think it comes from somewhere between that boundary?
I won't name names, but if you know who I'm talking about,

it's probably clear.
I once did a drive, in fact, up to the Mussolini era villa
outside Torino in the early days
of what became quantum computing
with a famous person who thinks about quantum mechanics
and consciousness.
And we had the most infuriating conversation
that went roughly along the lines of consciousness is weird,
quantum mechanics is weird,
therefore quantum mechanics explains consciousness.
That was roughly the logical process.
And you're not satisfied with that process?
No, and I say that very precisely in the following sense.
I was a program manager, somewhat by accident,
in a DARPA program on quantum biology.
And so biology trivially uses quantum mechanics
and that were made out of atoms,
but the distinction is in quantum computing,
quantum information, you need quantum coherence.
And there's a lot of muddled thinking
about collapse of the wave function
and claims of quantum computing
that garbles just quantum coherence.
That you can think of it as a wave
that has very special properties,
but these wave-like properties.
And so there's a small set of places
where biology uses quantum mechanics in that deeper sense.
One is how light is converted to energy in photo systems.
It looks like one is all faction,
how your nose is able to tell different smells.
Probably one has to do with how birds navigate,
how they sense magnetic fields.
That involves a coupling between a very weak energy
with a magnetic field, coupling into chemical reactions.
And there's a beautiful system.
Standard in chemistry is magnetic fields
like this can influence chemistry,
but there are biological circuits
that are carefully balanced with two pathways
that become unbalanced with magnetic fields.
So each of these areas are expensive for biology.
It has to consume resources

to use quantum mechanics in this way.
So those are places where we know
there's quantum mechanics in biology.
In cognition, there's just no evidence.
There's no evidence of anything quantum mechanical going on
in how cognition works.
Consciousness.
Well, I'm saying cognition, I'm not saying consciousness,
but to get from cognition to consciousness.
So McCulloch and Pitts made a model of neurons.
That led to perceptrons
that then through a couple of boom busts
led to deep learning.
One of the interesting things about that sequence is
it diverged off so deep neural networks
used in machine learning,
diverged from trying to understand how the brain works.
What makes them work, what's emerged is,
it's a really interesting story.
This may be too much of a technical detail,
but it has to do with function approximation.
We talked about exponentials.
A deep network needs an exponentially larger shallow network
to do the same function.
And that exponential is what gives the power
to deep networks.
But what's interesting is the sort of lessons
about building these deep architectures
and how to train them have really interesting echoes
to how brains work.
And there's an interesting conversation
that's sort of coming back of neuroscientists
looking over the shoulder of people training
these deep networks, seeing interesting echoes
for how the brain works, interesting parallels with it.
And so I didn't say consciousness, I just said cognition,
but I don't know any experimental evidence
that points to anything in neurobiology
that says we need quantum mechanics.
And I view the question about whether a large language model
is conscious as silly in that biology is full of hacks
and it works.
There's no evidence we have

that there's anything deeper going on
than just this sort of stacking up of hacks in the brain.
And somehow consciousness is one of the hacks
or an emergent property of the hacks.
Absolutely.
And just numerically I said big computations
now have the degrees of freedom of the brain.
And they're showing a lot of the phenomenology
of what we think is properties of what a brain can do.
And I don't see any reason to invoke anything else.
That makes you wonder what kind of beautiful stuff
a digital fabrication will create.
If biology created a few hacks
on top of which consciousness and cognition,
some of the things we love about human beings was created.
It makes you wonder what kind of beauty in the complexity
can create with a digital fabrication.
There's an early peak at that,
which is there's a misleading term, which is generative design.
Generative design is where you don't tell a computer
how to design something,
you tell the computer what you want it to do.
That doesn't work, that only works in limited subdomains.
You can't do really complex functionality that way.
The one place it's mature though
is topology optimization for structure.
So let's say you wanted to make a bicycle or a table.
You describe the loads on it
and it figures out how to design it.
And what it makes are beautiful organic looking things.
These are things that look like they grew in a forest
and they look like they grew in a forest
because that's sort of exactly what they are,
that they're solving the ways of how you handle loads
in the same way biology does.
And so you get things that look like trees and shells
and all of that.
And so that's a peak at this transition
to from we design to we teach the machines how to design.
What can you say about,
because you mentioned cellular automata earlier,
about from this example you just gave
and in general the observation you can make

by looking at cellular automata that there's a,
from simple rules and simple building blocks
can emerge arbitrary complexity.
Do you understand what that is, how that can be leveraged?
So understand what it is is much easier than it sounds.
I complained about Turing's machine
making a physics mistake,
but Turing never intended it to be a computer architecture.
He used it just to prove results about uncomputability.
What Turing did on what his computation is exquisite
is gorgeous.
He gave us our notion of computational universality
and something that sounds deep
and turns out to be trivial is it's really easy
to show almost everything is computationally universal.
So Norm Margolis wrote a beautiful paper
with Tom Toffoli showing in a cellular,
a cellular automata world is like the game of life
where you just move tokens around.
They showed that modeling billiard balls
on a billiard table with cellular automata
is a universal computer.
To be universal, you need a persistent state,
you need a nonlinear operation to interact them
and you need connectivity.
So that's what you need to show computational universality.
So they showed that a CA modeling billiard balls
is a universal computer.
Chris Moore went on to show that instead of chaos,
let's see, Turing showed there are computable,
there are problems in computation that you can't solve,
that they're harder than you can't predict,
they're actually in a deep reason that they are unsolvable.
Chris Moore showed it's very easy to make physical systems
that are uncomputable, that what the physics system does,
just bouncing balls and surfaces,
you can make systems that solve uncomputable problems.
And so almost any non-trivial physical system
is computationally universal.
So the first part of the answer to your question is,
this comes back to my comment
about how do you bootstrap a civilization?
You just don't need much to be computationally universal.

So then there isn't today a notion
of like fabricational universality
or fabricational complexity.
The sort of numbers I've been giving you
about you eating lunch versus the chip fab,
sort of that, that's in the same spirit of what Shannon did.
But once you connect computational universality
to kind of fabricational universality,
you then get the ability to grow and adapt and evolve.
Because that evolution happens in the physical space.
Yeah, and so that's why, for me,
the heart of this whole conversation is morphogenesis.
So just to come back to that,
what Turing ended his sadly cut short life studying
was how genes give rise to form.
So how the small amount of it,
relatively it affects small amount of information
in the genome can give rise to the complexity
of who you are.
And that's where what resides is this molecular intelligence,
which is first how to describe you,
but then how to describe you such that you can exist
and you can reproduce and you can grow and you can evolve.
And so that's the seat of our molecular intelligence.
The maker of revolution in biology.
Yeah, it really is, it really is.
And that's where you can't separate
communication, computation and fabrication.
You can't separate computer science and physical science.
You can't separate hardware and software.
They all intersect right at that place.
Do you think of our universe as just one giant computation?
I would even kind of say quantum computing is overhyped
in that there's a few things quantum computing
is gonna be good at.
One is breaking crypto systems,
how to make new crypto systems.
What it's really good at is modeling other quantum systems.
So for studying nanotechnology, it's gonna be powerful,
but quantum computing is not going to disrupt
and change everything.
But the reason I say that is this interesting group
of strange people who helped invent quantum computing

before it was clear anything was there.
One of the main reasons they did it
wasn't to make a computer that can break a crypto system.
It was you could turn this backwards.
You could be surprised quantum mechanics can compute
or you can go in the opposite direction
and say if quantum mechanics can compute,
that's a description of nature.
So physics is written in terms of
partial differential equations.
That is an information technology
from two centuries ago.
The equations of physics are not,
this would sound very strange to say,
but the equations of physics,
Schrodinger's equations and Maxwell's equations
and all of them are not fundamental.
They're a representation of physics
that was accessible to us in the era
of having a pencil and a piece of paper.
They have a fundamental problem
which is if you make a dot on a piece of paper
in traditional physics theory,
there's infinite information in that dot.
A point has infinite information.
That can't be true because
information is a fundamental resource
that's connected to energy.
And in fact, one of my favorite questions
you can ask a cosmologist to trip them up
is ask is information a conserved quantity in the universe?
Was all the information created in the Big Bang
or can the universe create information?
And I've yet to meet a cosmologist
who doesn't stutter and not clearly know
how to handle that existential question.
But sort of putting that to a side,
in physics theory, the way it's taught,
information comes late.
You're taught about X, a variable,
which can contain infinite information,
but physically that's unrealistic.
And so physics theories have to find ways to cut that off.

So instead, there are a number of people
who start with a theory of the universe
should start with information and computation
as the fundamental resources that explain nature.
And then you build up from that to something
that looks like throwing baseballs down a slope.
And so in that sense, the work on physics and computation
has many applications that we've been talking about,
but more deeply, it's really getting at new ways
to think about how the universe works.
And there are a number of things
that are hard to do in traditional physics
that make more sense when you start
with information and computation
as the root of physical theory.
So information and computation
being the real fundamental thing in the universe.
Right, that information is a resource.
You can't have infinite information in finite space.
Information propagates and interacts.
And from there, you erect the scaffolding of physics.
Now it happens, the words I just said
look a lot like quantum field theories.
But there's an interesting way
where instead of starting with differential equations
to get to quantum field theories
and quantum field theories, you get to quantization.
If you start from computation and information,
you begin sort of quantized and you build up from there.
And so that's the sense in which
absolutely I think about the universe as a computer.
The easy way to understand that is
just almost anything is computationally universal,
but the deep way is it's a real fundamental way
to understand how the universe works.
Let me go a little bit to the personal
and the center bits and atoms.
You have worked with the students you've worked with
have gone on to do some incredible things in this world,
including build supercomputers that power Facebook
and Twitter and so on.
What advice would you give to young people?
What advice have you given them?

How to have one heck of a great career,
one heck of a great life?
What one important one is
if you look at junior faculty
trying to get tenure at a place like MIT,
the ones who try to figure out how to get tenure
are miserable and don't get tenure.
And the ones who don't try to figure it out
are happy and do get it.
I mean, you have to love what you're doing and believe in it
and nothing else could possibly be
what you want to be doing with your life
and it gets you out of bed in the morning.
And again, it sounds naive,
but within the limited domain I'm describing now
of getting tenure at MIT, that's a key attribute to it.
And in the same sense, if you take the sort of outliers
students were talking about 99 out of 100 come to me
and say, your work is very fascinating,
I'd be interesting to work for you.
And one out of 100 come and say, here you're wrong.
Here's your mistake, here's what you should have been doing.
They just sort of say, I'm here and get to work.
And again, that's, I don't know how far this resource goes.
So I've said, I consider the world's greatest resource,
this engine, a bright invent of people
of which we only see a tiny little iceberg of it.
And everywhere we open these labs,
they come out of the woodwork.
They come, we didn't create all these educational programs,
all these other things I'm describing.
We tried to partner everywhere with local schools
and local companies and kept tripping over dysfunction
and find we had to create the environment
where people like this can flourish.
And so I don't know if this is everyone,
if it's 1% of society, what the fraction is,
but it's so many orders of magnitude bigger
than we see it today.
We've been racing to keep up with it
to take advantage of that resource.
Something tells me it's a very large fraction
of the population.

I mean, the thing that gives me most hope for the future
is that population.
Once a year, this whole lab network meets.
And it's my favorite gathering, it's in Bhutan this year,
because it's every body shape, it's every language,
every geography, but it's the same person
in all those packages.
It's the same sense of bright invent of joy and discovery.
If there's people listening to this
and they're just overwhelmed with how exciting this is,
which I think they would be, how can they participate,
how can they help, how can they encourage
young people or themselves to build stuff, to create stuff?
Yeah, that's a great question.
So this is part of a much bigger maker movement
that has a lot of embodiments.
The part I've been involved in, this fab lab network,
you can think of as a curated part
that works as a network.
So you don't benefit in a gym
if somebody exercises in another gym,
but in the fab network, you do in a sense benefit
when somebody works in another lab
in the way it functions as a network.
So you can come to cba.mit.edu
to see the research we're talking about.
There's a fab foundation run by Sherry Lasseter
at fabfoundation.org.
Fab Labs I.O. is a portal into this lab network.
Fabacademy.org is this distributed hands-on
educational program.
Fab.city is the platform of cities
producing what they consume.
Those are all nodes in this network.
So you can learn with Fabacademy
and you can perhaps launch or help launch
or participate in launching a fab lab.
Well, in particular, from one to 1,000,
we carefully counted labs.
Now we're going from 1,000 to a million
where it ceases to become interesting to count them.
And in the 1,000 to the million,
what's interesting about that stage

is technologically, you go to a lab
not to get access to the machine,
but you go to the lab to make the machine.
But the other thing interesting in it
is we have an interesting collaboration
on a fab lab in a box.
And this came out of a collaboration
with SolidWorks on how you can put a fab lab in a box
which is not just the tools, but the knowledge.
So you open the box and the box contains the knowledge
of how to use it as well as the tools within it
so that the knowledge can propagate.
And so we have an interesting group of people
working on the original fab labs,
we have a whole team to get involved
in the setting up and training.
And the fab academy is a real in-depth,
deep technical program in the training.
But in this next phase, how sort of the lab itself
knows how to do the lab.
We've talked deeply about the intelligence in fabrication,
but in a much more accessible one
about how the AI in the lab in effect
becomes a collaborator with you in this nearer term
to help get started.
And for people wanting to connect,
it can seem like a big step, a big threshold,
but we've gotten to thousands of these
and they're doubling exactly that way
just from people opting in.
And in so doing, driving towards this kind of idea
of personal digital fabrication.
And it's not utopia, it's not free,
but come back to today, we separately have education,
we have big business, we have startups,
we have entertainment, sort of each of these things
are segregated.
When you have global connection
to one of these local facilities,
in that you can do play and art and education
and create infrastructure.
You can make many of the things you consume.
You could make it for yourself,

it could be done on a community scale,
it could be done on a regional scale.
It really, I'd say the research we spent
the last few hours talking about, I thought was hard.
And in a sense, I mean, it's non-trivial
but in a sense, it's just sort of playing out
we're turning the crank.
What I didn't think was hard is
if anybody can make almost anything anywhere.
How do you live, how do you learn?
How do you work, how you play?
These very basic assumptions about how society functions.
There's a way in which it's kind of back to the future
in that this mode where work is money is consumption
and consumption is shopping by selecting
is only a kind of a few decade old stretch.
In some ways, we're getting back
to a Asami village in North Norway is deeply sustainable.
But rather than just reverting to living
the way we did a few thousand years ago,
being connected globally, having the benefits
of modern society, but connecting it back
to older notions of sustainability.
I hadn't remotely anticipated just how fundamentally
that challenges how a society functions
and how interesting and how hard it is to figure out
how we can make that work.
And it's possible that this kind of process
will give a deeper sense of meaning to each person.
Let me violently agree in two ways.
One way is this community making
crosses many sensitive sectarian boundaries
in many parts of the world where there's just
implicit or explicit conflict,
but sort of this act of making seems to transcend
a lot of historical divisions.
I don't say that philosophically,
I just say that as an observation.
And I think there's something really fundamental
in what you said, which is deep in our brain
is shaping our environment.
A lot of what's strange about our society
is the way that we can't do that.

The act of shaping our environment touches something
really, really deep that gets to the essence of who we are.
That's again why I say that in a way
the most important thing made in these labs
is making itself.
What do you think if the shaping of our environment
gets to something deep, what do you think is the meaning
of it all?
What's the meaning of life, Neil?
I can tell you my insights into how life works.
I can tell you my insights in how to make life
meaningful and fulfilling and sustainable.
I have no idea what the meaning of life is,
but maybe that's the meaning of life.
The uncertainty, the confusion.
Because there's a magic to it all.
Everything you've talked about from starting
from the basic elements with the big bang
that somehow created the sun,
that somehow set a few to thermodynamics and created life
and all the ways that you've talked about
from ribosomes that created the machinery
that created the machine
and then now the biological machine creating
through digital fabrication,
more complex artificial machines,
all of that, there's a magic to that creative process.
And we humans are smart enough
to notice the magic.
So you haven't said the S word yet.
Which one is that?
Singularity.
Yeah, I'm not sure if Ray Kurzweil is listening
if he is high Ray, but I have a complex relationship
with Ray because a lot of the things he projects,
I find annoying, but then he does his homework
and then somewhat annoyingly he points out
how almost everything I'm doing fits on his roadmaps.
And so, the question is,
are we heading towards a singularity?
So I'd have to say I lean towards sigmoids
rather than exponentials.
But we've done pretty well with sigmoids.

Yeah, so sigmoids are things grow and they taper
and then there can be one after it and one after it.
So I'll pass on whether there's enough of them
that they diverge.
But the selfish gene answer to the meaning of life
is the meaning of life is the propagation of life.
And so it was a step for atoms to assemble into a molecule.
For atoms to assemble into a molecule.
For molecules to assemble into a proto-cell.
For the proto-cell to form, to then form organelles.
For the organ cells to form organs,
the organs to form an organism.
Then it was a step for organisms to form family units.
Then family units to form villages.
You can view each of those as a stack
in the level of organizations.
You could view everything we've spoken about
as the imperative of life,
just the next step in the hierarchy of that
in the fulfillment of the inexorable drive
of the violation of thermodynamics.
So you could view, I'm an embodiment of the will
of the violation of thermodynamics speaking.
The two of us having an old chat, yes.
And so it continues and even then the singularity
is just a transition up the ladder.
There's nothing deeper to consciousness
than it's a derived property of distributed problem solving.
There's nothing deeper to life
than embodied AI in morphogenesis.
So why so much of this conversation in my life
is involved in these fab labs.
And initially it just started as outreach.
Then it started as keeping up with it.
Then it turned to, it was rewarding.
Then it turned to we're learning as much from these labs
in as goes out to them.
It began as outreach, but now more knowledge
is coming back from the labs that is going into them.
And then finally it ends with what I described
as competing with myself at MIT,
but a better way to say that is tapping
the brain power of the planet.

And so I guess for me personally,
that's the meaning of my life.
And maybe that's the meaning for the universe too.
It's using us humans and our creations to understand itself
in the way it's a whatever the creative process
that created earth is competing with itself.
Yeah, so you could take morphogenesis
as a summary of this whole conversation
or you could take recursion that in a sense,
what we've been talking about is recursion all the way down.
And in the end, I think this whole thing is pretty fun.
It's short life is, but it's pretty fun.
And so is this conversation.
I mentioned to you offline them,
going through some difficult stuff personally
and your passion for what you do is just really inspiring.
And it just lights up my mood and lights up my heart.
And you're an inspiration for, I know thousands of people
that work with you at MIT and millions of people across the world.
It's a big honor to use it with me today.
This is really fun.
This was a pleasure.
Thanks for listening to this conversation
with Neil Gershenfeld.
To support this podcast,
please check out our sponsors in the description.
And now, let me leave you with some words from Pablo Picasso.
Every child is an artist.
The challenge is staying an artist when you grow up.
Thank you for listening and hope to see you next time.