On long-term research vision

Peter Sewell

University of Cambridge

With thanks to Robin Milner – and all my colleagues

RTFM 2026 faculty mentoring workshop: Read the Faculty Manual, Rennes, 2026-01-13

1. This is a tricky topic to speak to: there's huge survivorship bias, and everyone's context and experience will be different. It's unclear how much I can say that isn't either vapid or obvious or over-specific to my world. So my goal is not to say what you should do, but just raise awareness of some things that I've found important, in the hope that it sparks some thought and discussion.

1. Today is mostly about pragmatics – grant writing and so on, which are all really important. But with all the pressures we have have to focus on those things can lead us to lose sight of the fundamental reasons we do what we do, so I thought I'd contextualise the day with this (with apologies to JFK):

We choose. We choose to advance science. We choose to advance science not because it is easy, but because it is hard, because that goal will serve to organise and measure the best of our energies and skills, because that challenge is one that we are willing to accept, one we are unwilling to postpone, and one which we intend to win.

We set sail upon this new sea because there is new knowledge to be gained, and it must be won and used for the progress of all people.

[with apologies to JFK, 1962]

1. Today is mostly about pragmatics – grant writing and so on, which are all really important. But with all the pressures we have have to focus on those things can lead us to lose sight of the fundamental reasons we do what we do, so I thought I'd contextualise the day with this (with apologies to JFK):

What is a "long-term research vision" anyway? And do you even need one?

1. I have a dream that computing will become less terrible. And a bad dream that it won't... But that alone is not an actionable research vision.
2. These stories each need to be subtly different. They each have to be articulated in well-chosen words. And they have to be *true* stories. Some optimism is called for, but not made-up pseudomotivation – otherwise you're deceiving all those people, and especially. yourself. This is not *advertising*.

What is a "long-term research vision" anyway? And do you even need one?

1. a dream

1. I have a dream that computing will become less terrible. And a bad dream that it won't... But that alone is not an actionable research vision.
2. These stories each need to be subtly different. They each have to be articulated in well-chosen words. And they have to be *true* stories. Some optimism is called for, but not made-up pseudomotivation – otherwise you're deceiving all those people, and especially. yourself. This is not *advertising*.

What is a “long-term research vision” anyway? And do you even need one?

1. a dream
2. a dream *and* at least part of a plan for how to get there from here

1. I have a dream that computing will become less terrible. And a bad dream that it won't... But that alone is not an actionable research vision.
2. These stories each need to be subtly different. They each have to be articulated in well-chosen words. And they have to be *true* stories. Some optimism is called for, but not made-up pseudomotivation – otherwise you're deceiving all those people, and especially. yourself. This is not *advertising*.

What is a "long-term research vision" anyway? And do you even need one?

1. a dream
2. a dream *and* at least part of a plan for how to get there from here
3. a story you tell to inspire potential PhD students and postdocs (and others)

1. I have a dream that computing will become less terrible. And a bad dream that it won't... But that alone is not an actionable research vision.
2. These stories each need to be subtly different. They each have to be articulated in well-chosen words. And they have to be *true* stories. Some optimism is called for, but not made-up pseudomotivation – otherwise you're deceiving all those people, and especially. yourself. This is not *advertising*.

What is a “long-term research vision” anyway? And do you even need one?

1. a dream
2. a dream *and* at least part of a plan for how to get there from here
3. a story you tell to inspire potential PhD students and postdocs (and others)
4. a story you tell paper readers, to contextualise what you’ve done and help the reader understand why it matters

1. I have a dream that computing will become less terrible. And a bad dream that it won’t... But that alone is not an actionable research vision.
2. These stories each need to be subtly different. They each have to be articulated in well-chosen words. And they have to be *true* stories. Some optimism is called for, but not made-up pseudomotivation – otherwise you’re deceiving all those people, and especially. yourself. This is not *advertising*.

What is a "long-term research vision" anyway? And do you even need one?

1. a dream
2. a dream *and* at least part of a plan for how to get there from here
3. a story you tell to inspire potential PhD students and postdocs (and others)
4. a story you tell paper readers, to contextualise what you've done and help the reader understand why it matters
5. a story you tell grant reviewers and faculty appointment/promotion panels

1. I have a dream that computing will become less terrible. And a bad dream that it won't... But that alone is not an actionable research vision.
2. These stories each need to be subtly different. They each have to be articulated in well-chosen words. And they have to be *true* stories. Some optimism is called for, but not made-up pseudomotivation – otherwise you're deceiving all those people, and especially. yourself. This is not *advertising*.

What is a "long-term research vision" anyway? And do you even need one?

1. a dream
2. a dream *and* at least part of a plan for how to get there from here
3. a story you tell to inspire potential PhD students and postdocs (and others)
4. a story you tell paper readers, to contextualise what you've done and help the reader understand why it matters
5. a story you tell grant reviewers and faculty appointment/promotion panels
6. a story you learn about yourself, from retrospective introspection

1. I have a dream that computing will become less terrible. And a bad dream that it won't... But that alone is not an actionable research vision.
2. These stories each need to be subtly different. They each have to be articulated in well-chosen words. And they have to be *true* stories. Some optimism is called for, but not made-up pseudomotivation – otherwise you're deceiving all those people, and especially. yourself. This is not *advertising*.

What is a "long-term research vision" anyway? And do you even need one?

1. a dream
2. a dream *and* at least part of a plan for how to get there from here
3. a story you tell to inspire potential PhD students and postdocs (and others)
4. a story you tell paper readers, to contextualise what you've done and help the reader understand why it matters
5. a story you tell grant reviewers and faculty appointment/promotion panels
6. a story you learn about yourself, from retrospective introspection
7. a hopefully faithful representation of your team's ideas

1. I have a dream that computing will become less terrible. And a bad dream that it won't... But that alone is not an actionable research vision.
2. These stories each need to be subtly different. They each have to be articulated in well-chosen words. And they have to be *true* stories. Some optimism is called for, but not made-up pseudomotivation – otherwise you're deceiving all those people, and especially. yourself. This is not *advertising*.

What is a "long-term research vision" anyway? And do you even need one?

1. a dream
2. a dream *and* at least part of a plan for how to get there from here
3. a story you tell to inspire potential PhD students and postdocs (and others)
4. a story you tell paper readers, to contextualise what you've done and help the reader understand why it matters
5. a story you tell grant reviewers and faculty appointment/promotion panels
6. a story you learn about yourself, from retrospective introspection
7. a hopefully faithful representation of your team's ideas
8. a tool to help you decide what in detail to do

1. I have a dream that computing will become less terrible. And a bad dream that it won't... But that alone is not an actionable research vision.
2. These stories each need to be subtly different. They each have to be articulated in well-chosen words. And they have to be *true* stories. Some optimism is called for, but not made-up pseudomotivation – otherwise you're deceiving all those people, and especially. yourself. This is not *advertising*.

What is a "long-term research vision" anyway? And do you even need one?

1. a dream
2. a dream *and* at least part of a plan for how to get there from here
3. a story you tell to inspire potential PhD students and postdocs (and others)
4. a story you tell paper readers, to contextualise what you've done and help the reader understand why it matters
5. a story you tell grant reviewers and faculty appointment/promotion panels
6. a story you learn about yourself, from retrospective introspection
7. a hopefully faithful representation of your team's ideas
8. a tool to help you decide what in detail to do
9. a tool to help you (in sum) do something really worth doing – to keep an eye on, in tension against short-term pressures

1. I have a dream that computing will become less terrible. And a bad dream that it won't... But that alone is not an actionable research vision.
2. These stories each need to be subtly different. They each have to be articulated in well-chosen words. And they have to be *true* stories. Some optimism is called for, but not made-up pseudomotivation – otherwise you're deceiving all those people, and especially. yourself. This is not *advertising*.

What is a "long-term research vision" anyway? And do you even need one?

1. a dream
2. a dream *and* at least part of a plan for how to get there from here
3. a story you tell to inspire potential PhD students and postdocs (and others)
4. a story you tell paper readers, to contextualise what you've done and help the reader understand why it matters
5. a story you tell grant reviewers and faculty appointment/promotion panels
6. a story you learn about yourself, from retrospective introspection
7. a hopefully faithful representation of your team's ideas
8. a tool to help you decide what in detail to do
9. a tool to help you (in sum) do something really worth doing – to keep an eye on, in tension against short-term pressures
10. a tool to let yourself not pay attention to too many parts of the subject – intentional narrowing of scope

1. I have a dream that computing will become less terrible. And a bad dream that it won't... But that alone is not an actionable research vision.
2. These stories each need to be subtly different. They each have to be articulated in well-chosen words. And they have to be *true* stories. Some optimism is called for, but not made-up pseudomotivation – otherwise you're deceiving all those people, and especially. yourself. This is not *advertising*.

What is a "long-term research vision" anyway? And do you even need one?

1. a dream
2. a dream *and* at least part of a plan for how to get there from here
3. a story you tell to inspire potential PhD students and postdocs (and others)
4. a story you tell paper readers, to contextualise what you've done and help the reader understand why it matters
5. a story you tell grant reviewers and faculty appointment/promotion panels
6. a story you learn about yourself, from retrospective introspection
7. a hopefully faithful representation of your team's ideas
8. a tool to help you decide what in detail to do
9. a tool to help you (in sum) do something really worth doing – to keep an eye on, in tension against short-term pressures
10. a tool to let yourself not pay attention to too many parts of the subject – intentional narrowing of scope
11. a recurringly fruitful area, in which there are a bunch of good paper-sized steps to be taken - fertile ground where you have a lot of parts and skills to hand

1. I have a dream that computing will become less terrible. And a bad dream that it won't... But that alone is not an actionable research vision.
2. These stories each need to be subtly different. They each have to be articulated in well-chosen words. And they have to be *true* stories. Some optimism is called for, but not made-up pseudomotivation – otherwise you're deceiving all those people, and especially. yourself. This is not *advertising*.

What is a "long-term research vision" anyway? And do you even need one?

1. a dream
2. a dream *and* at least part of a plan for how to get there from here
3. a story you tell to inspire potential PhD students and postdocs (and others)
4. a story you tell paper readers, to contextualise what you've done and help the reader understand why it matters
5. a story you tell grant reviewers and faculty appointment/promotion panels
6. a story you learn about yourself, from retrospective introspection
7. a hopefully faithful representation of your team's ideas
8. a tool to help you decide what in detail to do
9. a tool to help you (in sum) do something really worth doing – to keep an eye on, in tension against short-term pressures
10. a tool to let yourself not pay attention to too many parts of the subject – intentional narrowing of scope
11. a recurringly fruitful area, in which there are a bunch of good paper-sized steps to be taken - fertile ground where you have a lot of parts and skills to hand
12. a domain within which you can make a fair guess about whether some project (for specific people) will work out, and roughly how big it is.

1. I have a dream that computing will become less terrible. And a bad dream that it won't... But that alone is not an actionable research vision.
2. These stories each need to be subtly different. They each have to be articulated in well-chosen words. And they have to be *true* stories. Some optimism is called for, but not made-up pseudomotivation – otherwise you're deceiving all those people, and especially. yourself. This is not *advertising*.

What is a "long-term research vision" anyway? And do you even need one?

1. a dream
2. a dream *and* at least part of a plan for how to get there from here
3. a story you tell to inspire potential PhD students and postdocs (and others)
4. a story you tell paper readers, to contextualise what you've done and help the reader understand why it matters
5. a story you tell grant reviewers and faculty appointment/promotion panels
6. a story you learn about yourself, from retrospective introspection
7. a hopefully faithful representation of your team's ideas
8. a tool to help you decide what in detail to do
9. a tool to help you (in sum) do something really worth doing – to keep an eye on, in tension against short-term pressures
10. a tool to let yourself not pay attention to too many parts of the subject – intentional narrowing of scope
11. a recurringly fruitful area, in which there are a bunch of good paper-sized steps to be taken - fertile ground where you have a lot of parts and skills to hand
12. a domain within which you can make a fair guess about whether some project (for specific people) will work out, and roughly how big it is.
13. a trap you live within? "I'm interested (only) in X"

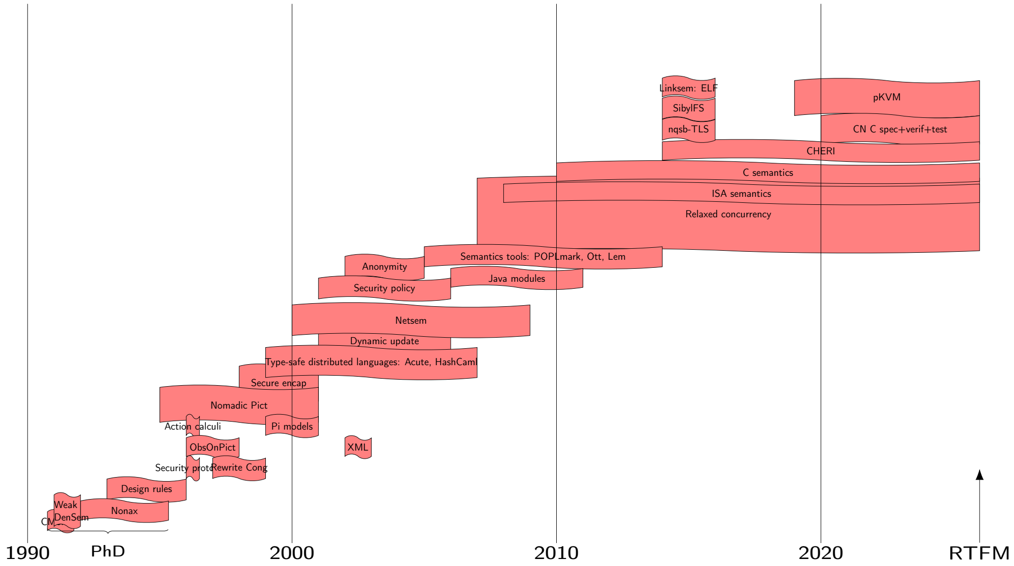
1. I have a dream that computing will become less terrible. And a bad dream that it won't... But that alone is not an actionable research vision.
2. These stories each need to be subtly different. They each have to be articulated in well-chosen words. And they have to be *true* stories. Some optimism is called for, but not made-up pseudomotivation – otherwise you're deceiving all those people, and especially. yourself. This is not *advertising*.

"My research aims to put the engineering of real-world computer systems on better foundations, developing techniques (both mathematically rigorous and pragmatically useful) to make systems that are better-understood, more robust, and more secure.

This applied semantics needs tightly integrated theoretical and practical research, spanning a range of Computer Science: architecture specification, programming languages, verification, security, and networking; developing and using techniques from semantics, type systems, automated reasoning, and concurrency theory.

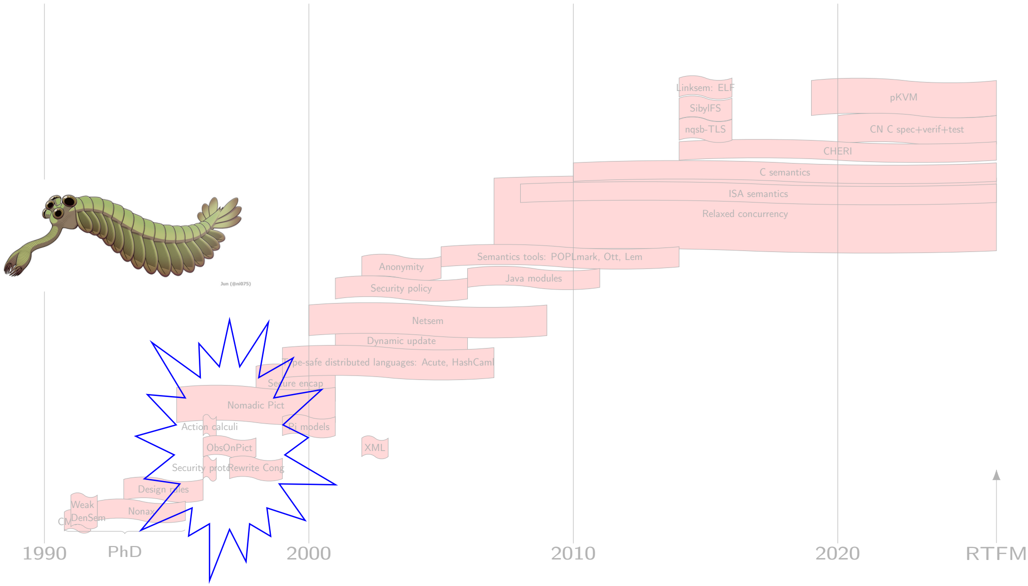
Our REMS group (Rigorous Engineering for Mainstream Systems) does this, in collaboration with [...]"

1. I looked at the web pages of the speakers here to see what they say about themselves. Here's one:
2. That's not totally useless: it does tell you something fairly distinctive about what my colleagues and I focus on. But it's about the *kind* of research, not really telling you much about the actual content. It's the introduction to those stories, but neither actionable nor sufficiently descriptive to be very useful. It's also confused about its pronouns – is this *my* research, or a *group's* research? Option B.
   To go further, even abstractly, we need a little more detail, so I drew a little picture.



1. That's rather a complicated picture. It's rough rendering of the timeline of topics from my web page. We don't need to read it or go into detail – I'm not here to talk about my research agenda or history – but there are some things maybe worth noticing just from the *shape* of it

mini Cambrian explosion



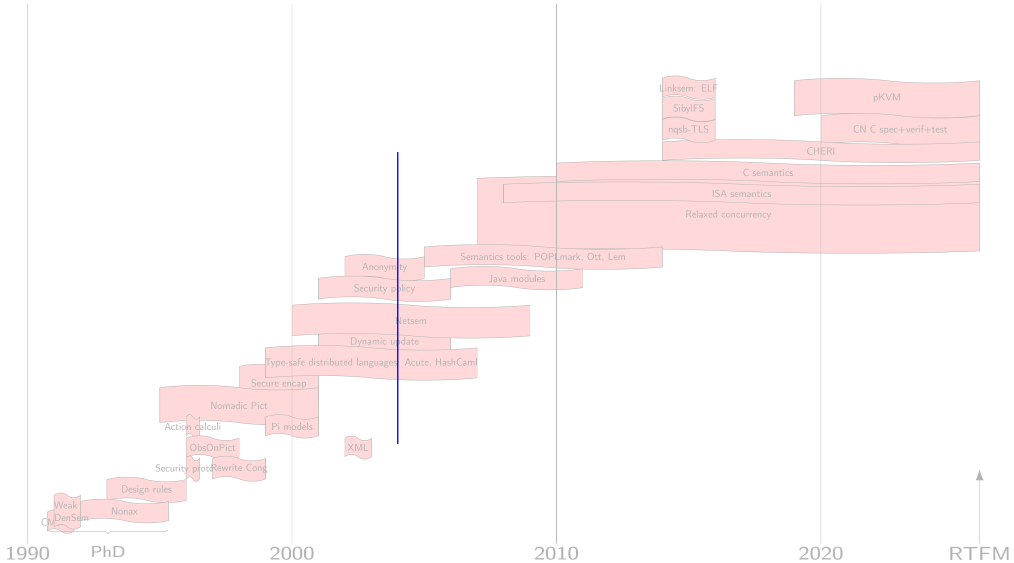
1. **Opabinia Regalis**

   post-PhD, as a young postdoc with Robin, I had this mini-Cambrian-explosion of topics – I was interested in lots of things, and (thanks to Robin) had the space to try and explore. Benjamin Pierce (also in Cambridge at that time) said something about "research agendas" and I was a bit mystified by the concept – I thought of myself as just doing whatever seemed interesting.

   They mostly turned into single papers; some didn't lead to anything published – which was fine. I was also writing various small and medium-scale grant proposals.

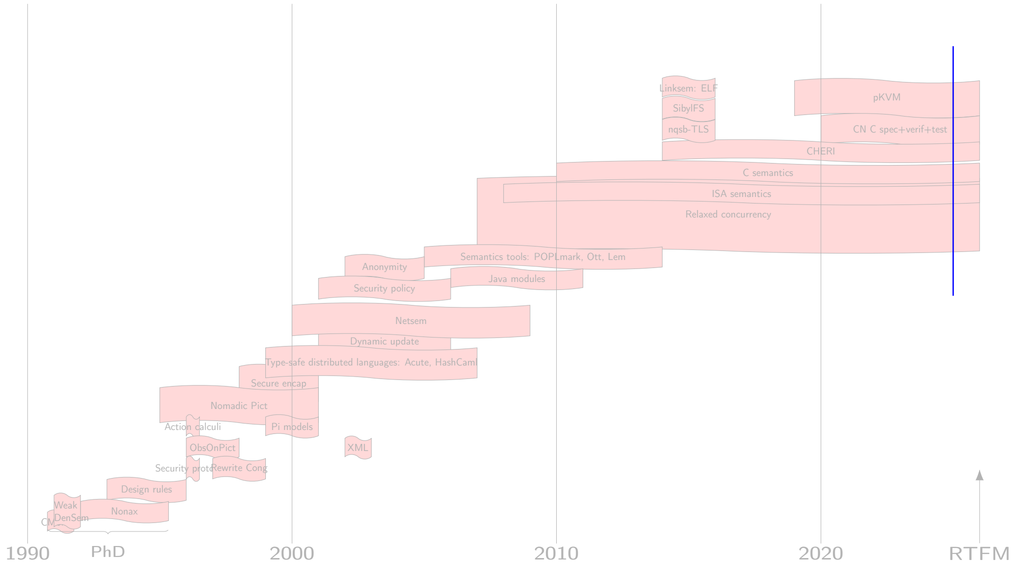
   In hindsight, that time for independent exploration was really important, as was the grant writing – forcing me to articulate what I was interested, with reasons why it mattered and a plausible plan. Though obviously pressure-to-publish also makes it very hard. It's a delicate balance – both for you as individuals and your team and other colleagues. Try to ignore pressure-to-publish *as much as possible*...

Vertical structure – Concurrency – how many things to do at once?



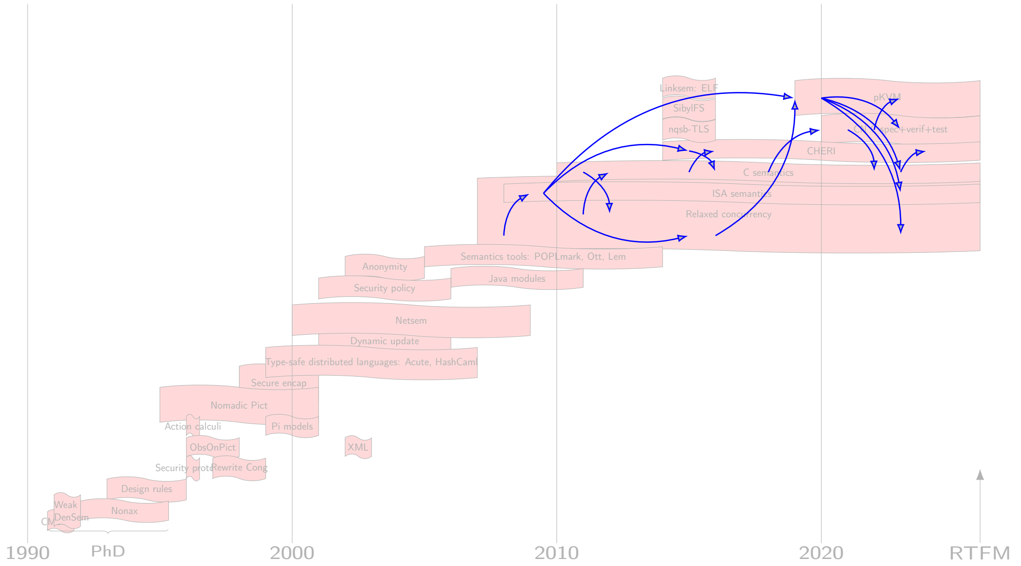
1. Look at vertical slices. Here, that was five things going on at once, with several students and two overlapping small groups all doing different things. That was great – but also, the context switching kills you.

Vertical structure – Concurrency – how many things to do at once?



1. But then what about now? If that was bad, why am I doing it again? (a) I'm stupid, but (b) all these things feed into each other...

Vertical structure – Concurrency – how many things to do at once?



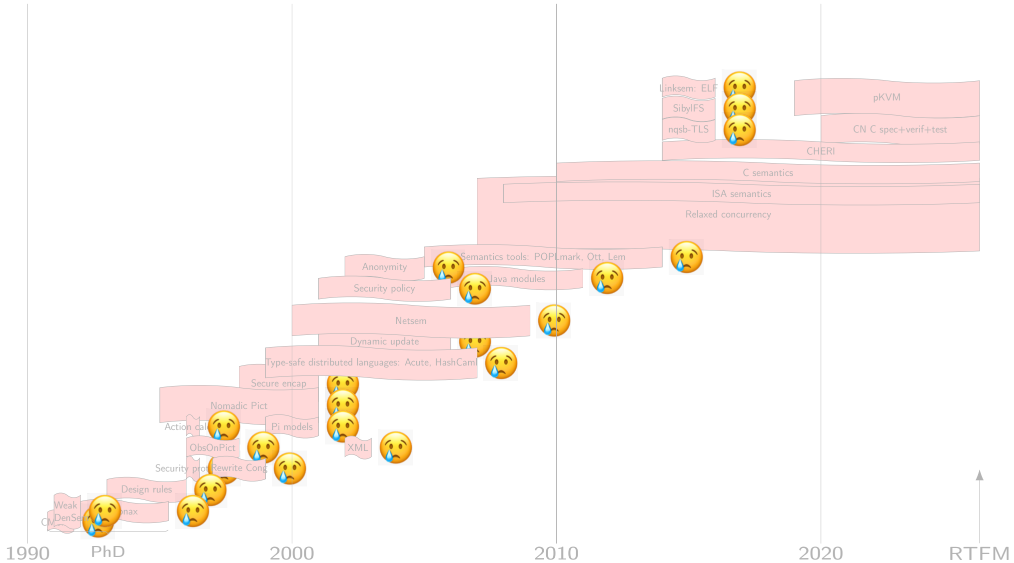
1. – relaxed concurrency needs ISA semantics
   – ISA semantics enables CHERI design-time ISA semantics, CHERI ISA semantics enables RISC-V ISA semantics
   – architecture concurrency and language-level concurrency inform each other (and need to be tensioned against each other)
   – C semantics and CHERI C/C++ inform each other
   – C semantics enables CN verification and testing of C
   – all these things are needed for pKVM hypervisor verification (which serves as a test-lab for what we can do to improve real-world systems software)

and it's fun and (I like to think) useful...

They're individual papers and individual projects and all one whole, all at the same time (you thought the particle vs wave thing was bad...)

We'll get back to people later, but the other big difference is the groups of people I was working with. Both were/are great – but now it's a larger and more distributed group – not just "my" group, and with much more accumulated experience

Horizontal structure: endings

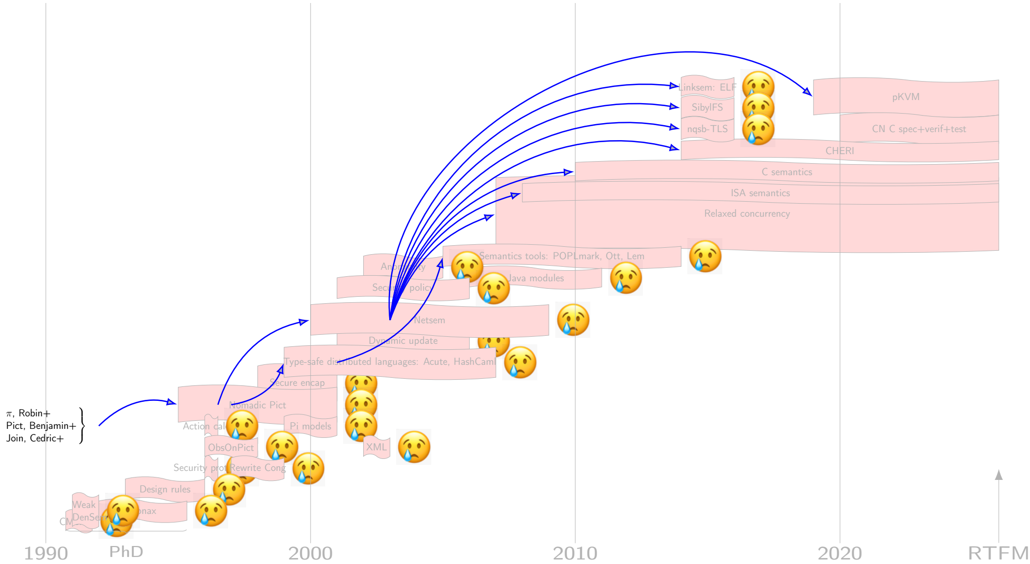


1. Lots of projects also means lots of project endings – why is that? Isn't it sad? Isn't it wasteful (of all that expertise you and your colleagues have built up)?

   Why do they end?

   - Sometimes the research line is properly done. For example, the nonaxiomatisability results of my thesis – of debatable interest in the first place – closed down rather than opening up new directions. Now, CPU architectural relaxed concurrency (speaking just of defining the whole-system semantics) is not done, but one can see that we're getting there.
   - Sometimes we didn't get funding, and the people moved on – E.g. Security policy
   - Sometimes people moved on and we got tired – eg Netsem
   - Sometimes there was a combination of the people moving on, and realising that this was not the best idea. Sometimes that's because the research motivation was predicated on a belief about the way the subject would go which turned out not to be the case
     - for example, there was much excitement about mobile computation which didn't really eventuate (well, maybe it did, but in the form of VM migration within datacenters, and functions-as-a-service, which didn't pay any attention to what we did)
     - for example, we did a lot of work on type-safe distributed computing, and made a version of OCaml supporting that, but (a) there was no prospect of upstreaming, and (b) industry didn't (and still hasn't) gone that way. Though arguably it should...

Horizontal structure: endings – and the invisible flows

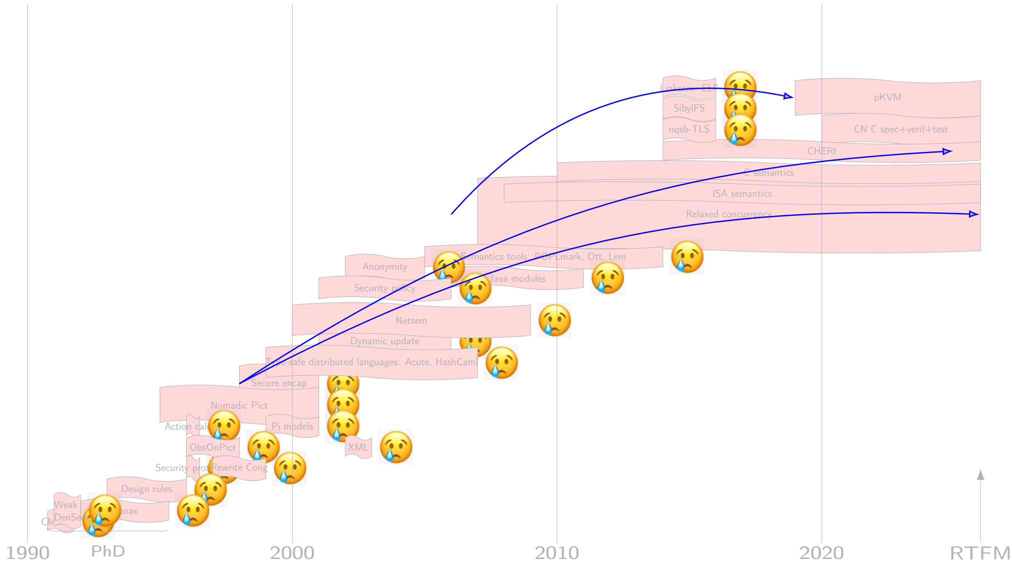


1. And sometimes it looks like they end – but actually there are invisible indirect flows that are more important than what you see in the published papers
   - pi-pict-join → nomadicpict (wanting more realistic failure semantics)
   - nomadicpict → type-safe-distributed (wanting more realistic programming language)
   - {nomadicpict,type-safe-distributed} → netsem (again wanting more realistic failure semantics)
   - type-safe-distributed languages → semantic-tools (to make large semantic definitions viable to work with)
   - netsem → {relaxed-concurrency, ISAsemantics, C semantics, CHERI, CN, nqsb-TLS, sibylFS, linksem, pkvm} (learning that (and how) we could do semantics for real-world abstractions)

In hindsight, there's a general trend: to try to "do" semantics for more real things (where "do" is figure out how to create *and make useful*). Is that my retrospective "research agenda"? Certainly it's one of them.

There is no "do", only "try"? In the course of trying, one finds new problems – which can be the biggest payoff.

Horizontal structure: endings – and the invisible flows



1. Sometimes one has, not a technical flow, but related motivation recurring over a disturbing long time interval. For example, in the late 1990s Jan Vitek and I worked on secure encapsulation of untrusted code, in a highly idealised setting – and last year and this, we've proved secure encapsulation results about real systems, for Morello (by Thomas Bauereiss and Angus Hammond et al., and Arm virtual memory, by Thibaut Perami and Thomas again). Our work on relaxed memory actually arose from thinking about hypervisor verification, with Susmit Sarkar in 2006 – and now we're seriously looking at the pKVM hypervisor.

Trajectories

maths
LICS
SIGPLAN (theory-end)
SIGPLAN (practice-end)
systems research (OS, CompArch, Security, Networks)
mainstream PL impl, std cttes, industry architects, Linux kernel devs, etc.
back-end application developers
web developers
spreadsheet users
consumers

1. I reckon Computing has a diameter of about 9 – where the metric is one unit for one community thinking that another is within reach but pretty hard-core, or pretty theoretical.

   It's fairly common for people's interests to gradually move towards the more applied – mine has certainly done that, and also broadened. Why? After doing some theory, it's natural to want to see whether it actually does what one claimed, or to do whatever is necessary for others to take it up. And it's probably easier for people to move in this direction than the other one. Though I certainly don't want to argue that the more applied is in general "better" – there are good things to do at all these points.

Trajectories

maths
LICS
SIGPLAN (theory-end)
SIGPLAN (practice-end)
systems research (OS, CompArch, Security, Networks)
mainstream PL impl, std cttes, industry architects, Linux kernel devs, etc.
back-end application developers
web developers
spreadsheet users
consumers

1. I reckon Computing has a diameter of about 9 – where the metric is one unit for one community thinking that another is within reach but pretty hard-core, or pretty theoretical.

   It's fairly common for people's interests to gradually move towards the more applied – mine has certainly done that, and also broadened. Why? After doing some theory, it's natural to want to see whether it actually does what one claimed, or to do whatever is necessary for others to take it up. And it's probably easier for people to move in this direction than the other one. Though I certainly don't want to argue that the more applied is in general "better" – there are good things to do at all these points.

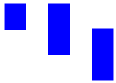
Trajectories

maths
LICS
SIGPLAN (theory-end)
SIGPLAN (practice-end)
systems research (OS, CompArch, Security, Networks)
mainstream PL impl, std cttes, industry architects, Linux kernel devs, etc.
back-end application developers
web developers
spreadsheet users
consumers

1. I reckon Computing has a diameter of about 9 – where the metric is one unit for one community thinking that another is within reach but pretty hard-core, or pretty theoretical.

   It's fairly common for people's interests to gradually move towards the more applied – mine has certainly done that, and also broadened. Why? After doing some theory, it's natural to want to see whether it actually does what one claimed, or to do whatever is necessary for others to take it up. And it's probably easier for people to move in this direction than the other one. Though I certainly don't want to argue that the more applied is in general "better" – there are good things to do at all these points.

Trajectories

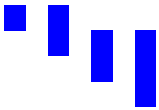


maths
LICS
SIGPLAN (theory-end)
SIGPLAN (practice-end)
systems research (OS, CompArch, Security, Networks)
mainstream PL impl, std cttes, industry architects, Linux kernel devs, etc.
back-end application developers
web developers
spreadsheet users
consumers

1. I reckon Computing has a diameter of about 9 – where the metric is one unit for one community thinking that another is within reach but pretty hard-core, or pretty theoretical.

   It's fairly common for people's interests to gradually move towards the more applied – mine has certainly done that, and also broadened. Why? After doing some theory, it's natural to want to see whether it actually does what one claimed, or to do whatever is necessary for others to take it up. And it's probably easier for people to move in this direction than the other one. Though I certainly don't want to argue that the more applied is in general "better" – there are good things to do at all these points.

Trajectories

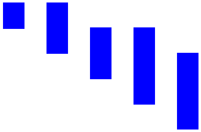


maths
LICS
SIGPLAN (theory-end)
SIGPLAN (practice-end)
systems research (OS, CompArch, Security, Networks)
mainstream PL impl, std cttes, industry architects, Linux kernel devs, etc.
back-end application developers
web developers
spreadsheet users
consumers

1. I reckon Computing has a diameter of about 9 – where the metric is one unit for one community thinking that another is within reach but pretty hard-core, or pretty theoretical.

   It's fairly common for people's interests to gradually move towards the more applied – mine has certainly done that, and also broadened. Why? After doing some theory, it's natural to want to see whether it actually does what one claimed, or to do whatever is necessary for others to take it up. And it's probably easier for people to move in this direction than the other one. Though I certainly don't want to argue that the more applied is in general "better" – there are good things to do at all these points.

Trajectories



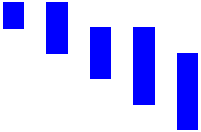
maths
LICS
SIGPLAN (theory-end)
SIGPLAN (practice-end)
systems research (OS, CompArch, Security, Networks)
mainstream PL impl, std cttes, industry architects, Linux kernel devs, etc.
back-end application developers
web developers
spreadsheet users
consumers

Different kinds of contribution
More direct and more widely appreciable motivational argument
If you want to show something works, you often have to do it yourself
More diverse (more fun?)

1. I reckon Computing has a diameter of about 9 – where the metric is one unit for one community thinking that another is within reach but pretty hard-core, or pretty theoretical.

   It's fairly common for people's interests to gradually move towards the more applied – mine has certainly done that, and also broadened. Why? After doing some theory, it's natural to want to see whether it actually does what one claimed, or to do whatever is necessary for others to take it up. And it's probably easier for people to move in this direction than the other one. Though I certainly don't want to argue that the more applied is in general "better" – there are good things to do at all these points.

Trajectories



maths
LICS
SIGPLAN (theory-end)
SIGPLAN (practice-end)
systems research (OS, CompArch, Security, Networks)
mainstream PL impl, std cttes, industry architects, Linux kernel devs, etc.
back-end application developers
web developers
spreadsheet users
consumers

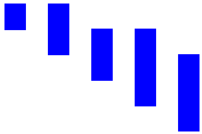
Different kinds of contribution
More direct and more widely appreciable motivational argument
If you want to show something works, you often have to do it yourself
More diverse (more fun?)

Consequences: greater friction + inertia, needs more people, needs more kinds of people (esp collaborations with the right industry people), harder to bring audience along *("please explain all of relaxed memory to make this self-contained")*, referees get confused by research which is not of a kind that they're used to *("what's the simple idea?" "why should we care about actual computers?")*

1. I reckon Computing has a diameter of about 9 – where the metric is one unit for one community thinking that another is within reach but pretty hard-core, or pretty theoretical.

   It's fairly common for people's interests to gradually move towards the more applied – mine has certainly done that, and also broadened. Why? After doing some theory, it's natural to want to see whether it actually does what one claimed, or to do whatever is necessary for others to take it up. And it's probably easier for people to move in this direction than the other one. Though I certainly don't want to argue that the more applied is in general "better" – there are good things to do at all these points.
2. This larger-scale stuff is what I've chosen to do – and I hope and think many others should, but I'm talking about it because it's what I know, not because everyone should do it.

Trajectories



maths
LICS
SIGPLAN (theory-end)
SIGPLAN (practice-end)
systems research (OS, CompArch, Security, Networks)
mainstream PL impl, std cttes, industry architects, Linux kernel devs, etc.
back-end application developers
web developers
spreadsheet users
consumers

Different kinds of contribution
More direct and more widely appreciable motivational argument
If you want to show something works, you often have to do it yourself
More diverse (more fun?)

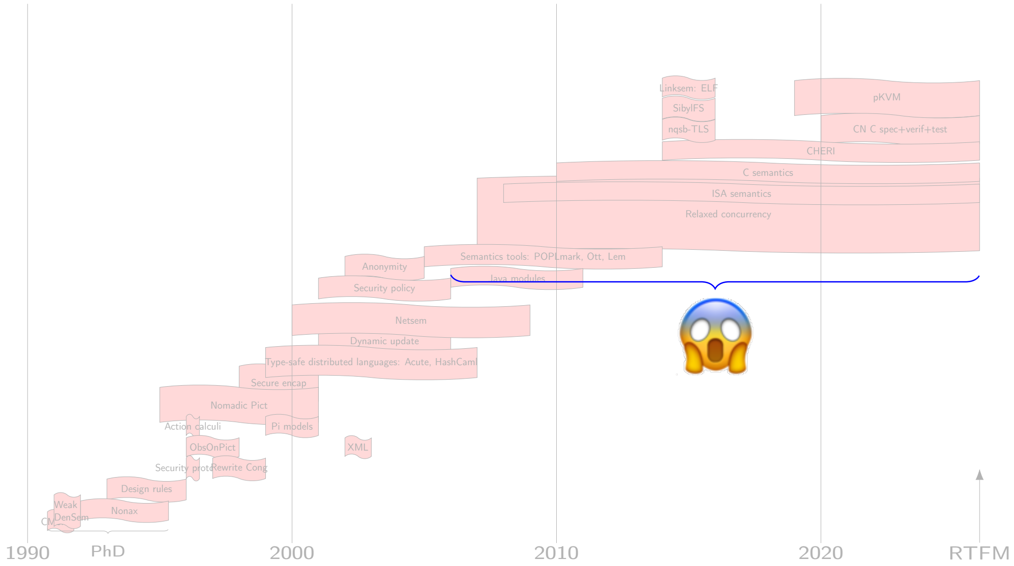
Consequences: greater friction + inertia, needs more people, needs more kinds of people (esp collaborations with the right industry people), harder to bring audience along *("please explain all of relaxed memory to make this self-contained")*, referees get confused by research which is not of a kind that they're used to *("what's the simple idea?" "why should we care about actual computers?")*

Trajectories within a project, between them, of a person, of the area – want to intentionally position

1. I reckon Computing has a diameter of about 9 – where the metric is one unit for one community thinking that another is within reach but pretty hard-core, or pretty theoretical.

   It's fairly common for people's interests to gradually move towards the more applied – mine has certainly done that, and also broadened. Why? After doing some theory, it's natural to want to see whether it actually does what one claimed, or to do whatever is necessary for others to take it up. And it's probably easier for people to move in this direction than the other one. Though I certainly don't want to argue that the more applied is in general "better" – there are good things to do at all these points.
2. This larger-scale stuff is what I've chosen to do – and I hope and think many others should, but I'm talking about it because it's what I know, not because everyone should do it.

Horizontal structure – Length and dependencies



1. Substantial impact (if at all – which is rare) is often from a whole sequence of papers, or a whole community, not just one paper. Or much longer-term than you might imagine (or wish)

   But that means you have to maintain:
   - skills
   - culture
   - research software

   over many years, much longer than the single-PhD duration.

   We still routinely use software tools that Francesco, Scott, and I wrote 20 years ago. And we still depend on tools (Lem) that we can't maintain any more, even though they don't do exactly what we need.

   You also need to make solid building blocks: if you're going to make a difference to practice – and also if you're going to make advances solid enough for you yourselves to build on – they have to be done better than one might "get away with" just for one paper. For example, it's one thing to make a semantics for a more-or-less C-like language – but if you want to engage with the C community, or to use it as the basis for a verification tool, it has to be the semantics (of a fragment, with limitations) of actual C. Etc etc. Making something that's good enough to apply to real systems (even if still imperfect) can make a huge qualitative difference in what you can do with – though it costs.

   Some kinds of research lead one to be centered around – and tied to – a single software tool, e.g. a verification tool or solver.

Horizontal structure – Length and dependencies

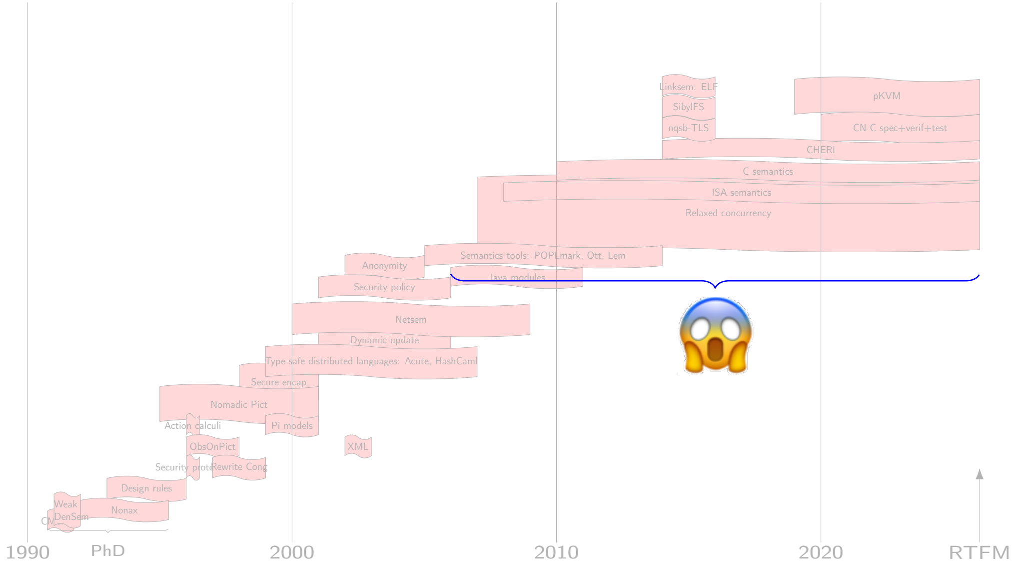
Mundane mechanics:

- record the working notes for future-you!

  `notes01-2026-01-13-rtfm-scribble.md`

  - sequence number – in-order and good for auto-complete
  - creation date
  - useful title
  - long-term-usable format (txt, md, jpg, tex, pdf, html)
  - authors in the file
- write down the reasons for choices right then
- stable luddite tooling - text or markdown notes. Archived snapshots. Latex. Makefiles. slides in beamer so can paste bits around. cvs-svn-{bitbucket,git}-git. Stable PLs

Horizontal structure – Length and dependencies – and users(!)



1. Depending on the topic, papers may not be the end game – whatever impact you want might *need* you to maintain and support artifacts that people actually use, or standards documents, or educational materials – or inspire and nurture a new community

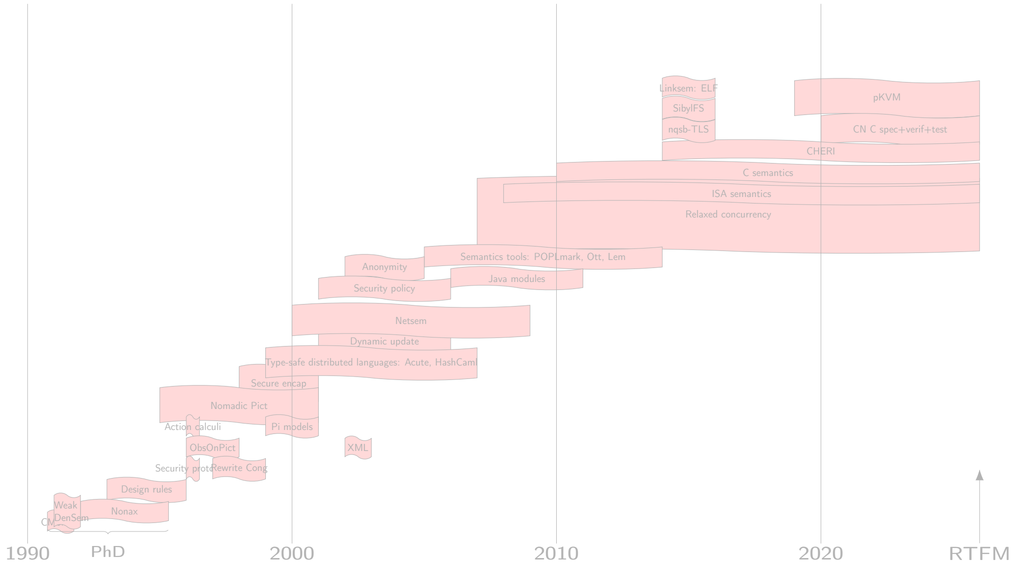
   - Users are great!
   - Users are terrible!

   And... you have to teach and support them.

   All this can require big investments of effort.

   And... as those can be at odds with conventional publication, you have to make sure it's valued, by whatever/whoever matters in your local environment.

Wait, how is that even possible?



Wait, how is that even possible?

The people!

1. the people as individuals, and as groups and subgroups, and the local cultures

   all unique, with different talents and attributes and potential roles, and all changing

Jigsaw

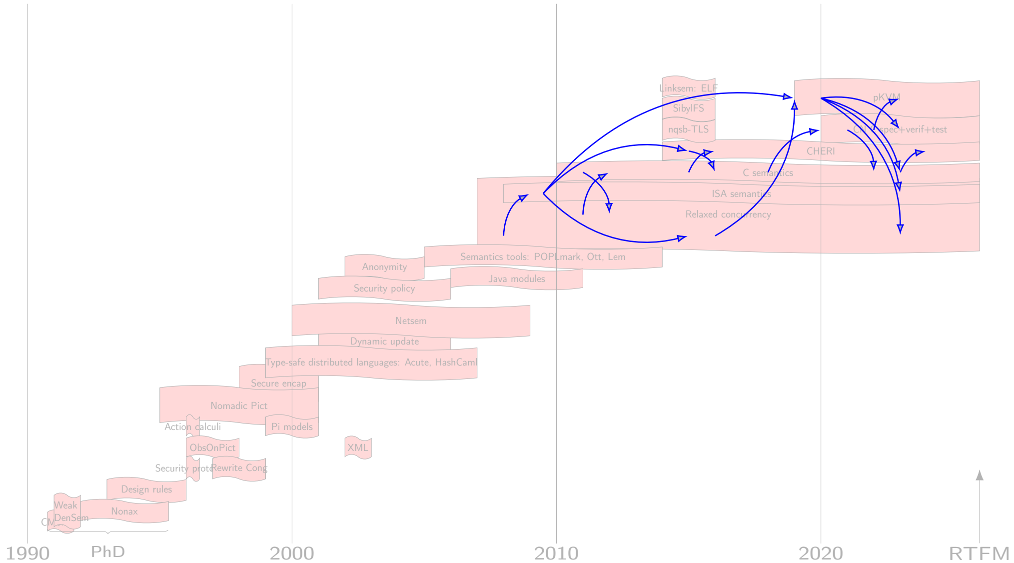
N-dimensional jigsaw. Of pieces of all kinds of shapes, some flexible, some rigid. With no picture on the front of the box except what you (collectively) hallucinate.

Fit together:

- the people (postdocs/students/other collaborators). Different people have different skills, different inclinations, and different long-term goals and plans
- the technical direction(s) and the choice of what (and how big) a step to take
- the funding
- the social interactions
- the external interactions, with users or industry folk or whoever

It's your job (well, really everyone's job) to make projects come out well for everyone involved – which means different things for different people

A research vision? more like a braid



A research vision? more like a braid

For example:

ArchSem: Reusable rigorous semantics of relaxed architectures. Thibaut Pérami, Thomas Bauereiss, Brian Campbell, Zongyuan Liu, Nils Lauermann, Alasdair Armstrong, and Peter Sewell. POPL 2026.

The concrete results:

- prover integration of architecture semantics for instruction-set and relaxed-concurrent systems behaviour
- proof that the Arm-A architecture can, in a simple scenario, provide a sound illusion of virtual memory

Research agendas that that contributes to, from short-term and narrow to long-term and broad:

- clean up some of our accumulated technical debt from software-tool-based relaxed-memory semantics
- identify and solve new problems that arise in doing that (partiality, intra-instruction de-ordering)
- provide a setting to develop more rigorous analogues of existing relaxed-memory tooling
- enable foundational reasoning about semantics of real-world architectures
  - state and prove implicit fundamental theorems that computing relies on
  - provide a basis for soundness proofs of architecture program logics, compilers, translation validation...
  - and thus enable proofs of concurrent hypervisors and other systems code
- improve conventional practice in architecture specification
- weave executable rigorous semantics, testing, and proof tightly together, and inspire others to do likewise
- *"Our research aims to put the engineering of real-world computer systems on better foundations, developing techniques (both mathematically rigorous and pragmatically useful) to make systems that are better-understood, more robust, and more secure."*

Different social-group modes of doing research

1. single-author papers
2. support one or two PhD students or postdocs, from either their or your initial idea
3. an ensemble-cast production, with several main players, and supporting cast as needed
4. several of the above, with overlapping people and varying leads
5. support one of your PhD students in de facto supervising students and/or faculty in other groups
6. line up multiple groups in roughly the same direction
7. stand back and let them get on with it (and ideally publish without you)
8. inspire and enable others to do good things

1. All are good – though for some things, the ensembles are more efficient (and perhaps more fun?) Again it's quite common for people to move down this.

   Quite a lot of what I do is just saying "hey, you and you should talk about that, no? Why don't you do that right now?"

   (Many researchers are *very bad* at actually talking with the others doing related work...)

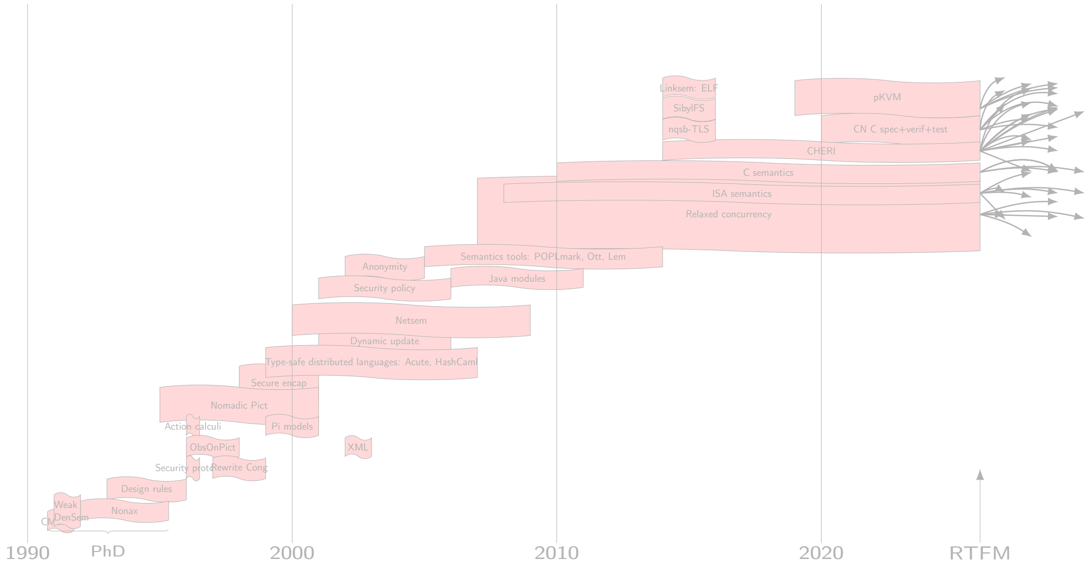
Different wider research environments

1. striking out on a more-or-less unexplored path (hopefully the first steps can be reasonably big steps, and there are no competitors – but also there are no other experts to appreciate and build on what you do, and teach you what they do)
2. working in a buzzing sub-area (with several other groups aiming at similar things)
3. working in the epicentre of some current hype
4. working in a previously well-explored (not enough? too much?) area

You might want or need to deliberately grow and nurture a wider community

Sometimes enabling good comparison can help drive everyone's research, e.g. with challenge problems or competitions or benchmarks – though watch out for tail-wagging-dog

How do we actually decide what to do next?

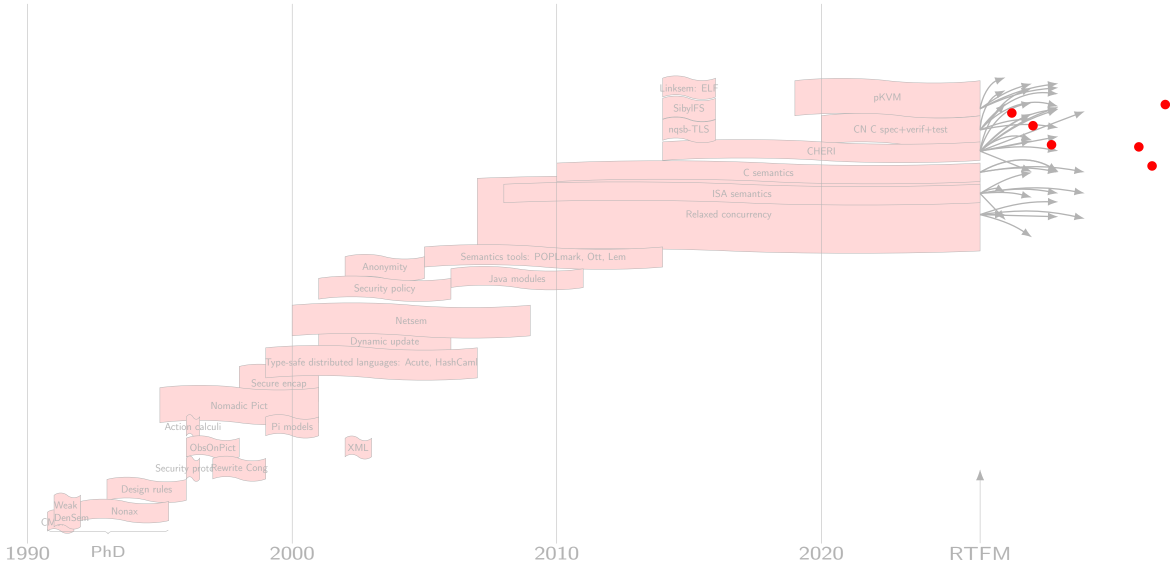


1. **Sometimes the course is clear – but always there are many pending interesting things we *could* do. Every so often (for individuals or groups) brainstorm and lay out options, to *choose*.**

   For example, in 2024 we did some REMS-group planning, producing 150 lines of notes. That's an ineteresting scale: even describing that space wouldn't fit in this talk, or a one-hour meeting.

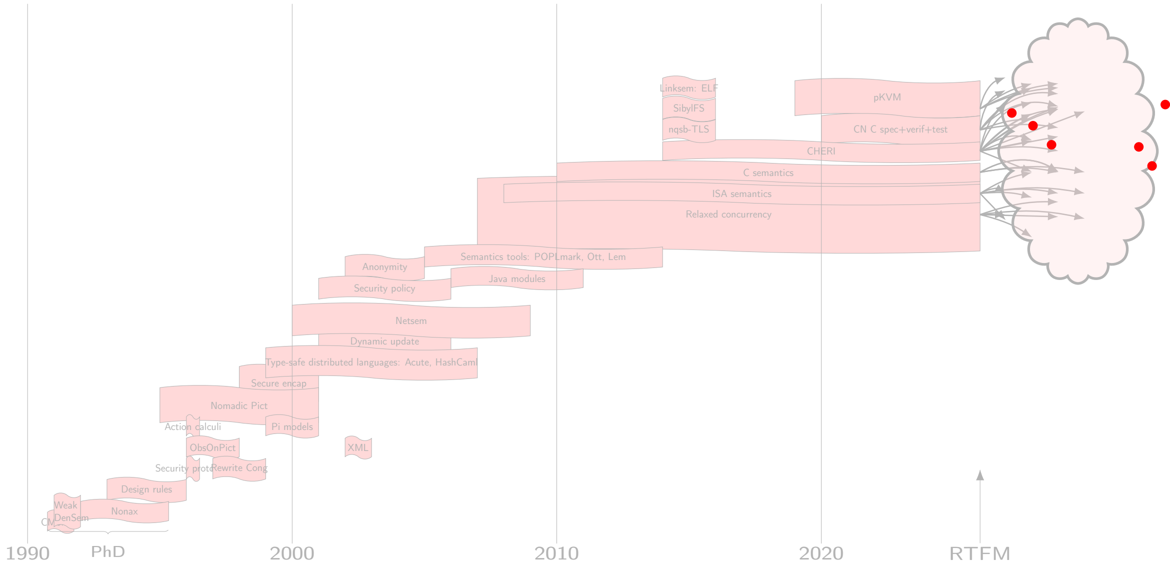
   That was our *actionable* research agenda of the moment. (And did it work out exactly like we planned? Of course not)

How do we actually decide what to do next?



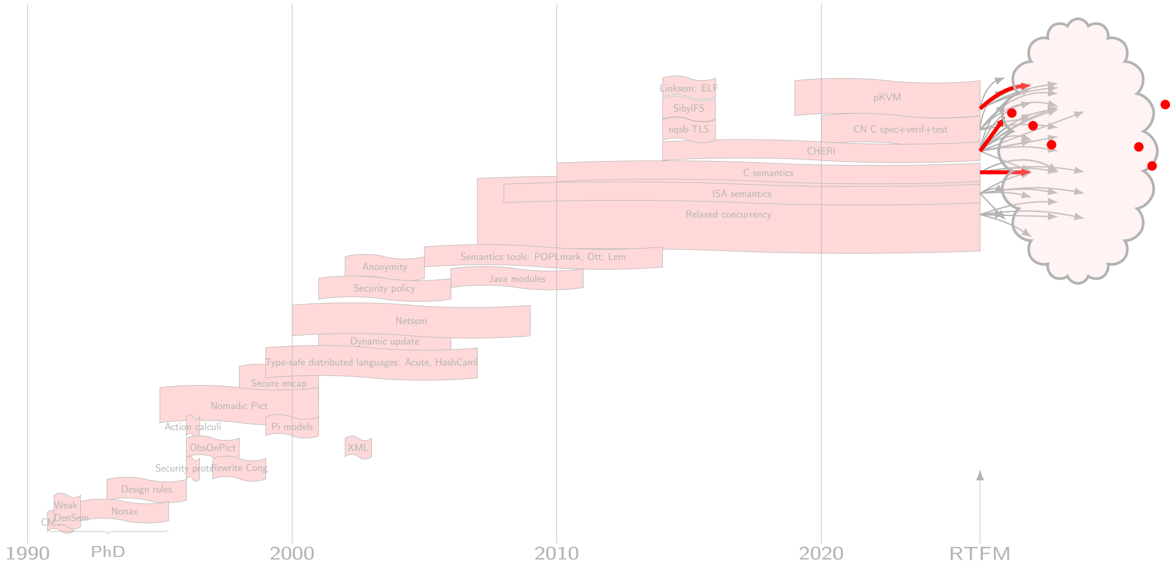
1. We might have many options of what to do right now – the arrows – and then several long-term agendas we're pushing or specific goals we're aiming for – the red blobs. For some, it might be clear how to get there.

How do we actually decide what to do next?



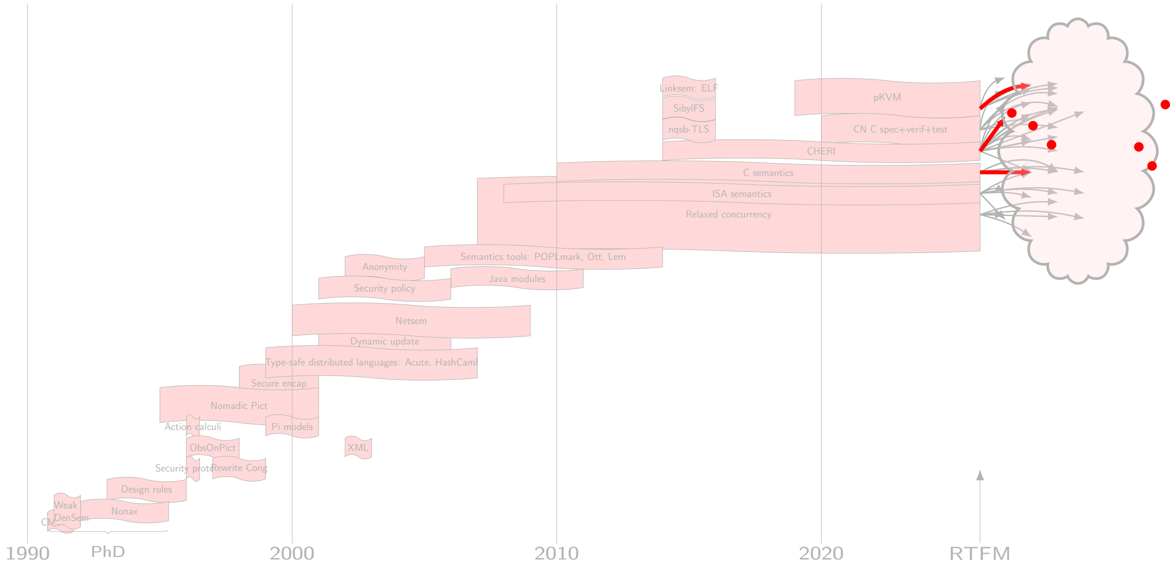
1. We might have many options of what to do right now – the arrows – and then several long-term agendas we're pushing or specific goals we're aiming for – the red blobs. For some, it might be clear how to get there.
2. For others, we can only see a fuzzy path – but still one wants to steer in the right general direction

How do we actually decide what to do next?



1. We might have many options of what to do right now – the arrows – and then several long-term agendas we're pushing or specific goals we're aiming for – the red blobs. For some, it might be clear how to get there.
2. For others, we can only see a fuzzy path – but still one wants to steer in the right general direction
3. Then we choose the specific things to do right now (which inevitably end up not being exactly what we do, but hey

How do we actually decide what to do next?



1. More general comments about this:

   Aim to do something worth doing! And something that people will (if need be, after education by you) will care about. If some research falls in the forest and no-one hears it, did it happen?

   One has to be prepared to invest. Aiming for a POPL paper from nothing ASAP is a recipe to get nothing: sometimes one can see a plan that's immediately actionable with the people, skills, tools, and ideas you have to hand – but many good lines of research have a big start-up cost. For example, we started thinking about relaxed memory around the start of 2007, with the first papers in 2009, and the first good papers for more relaxed architectures in 2010/2011. We started thinking about language design for instruction semantics in 2009, which lead to the Sail paper in 2019 (although with various payoffs along the way).

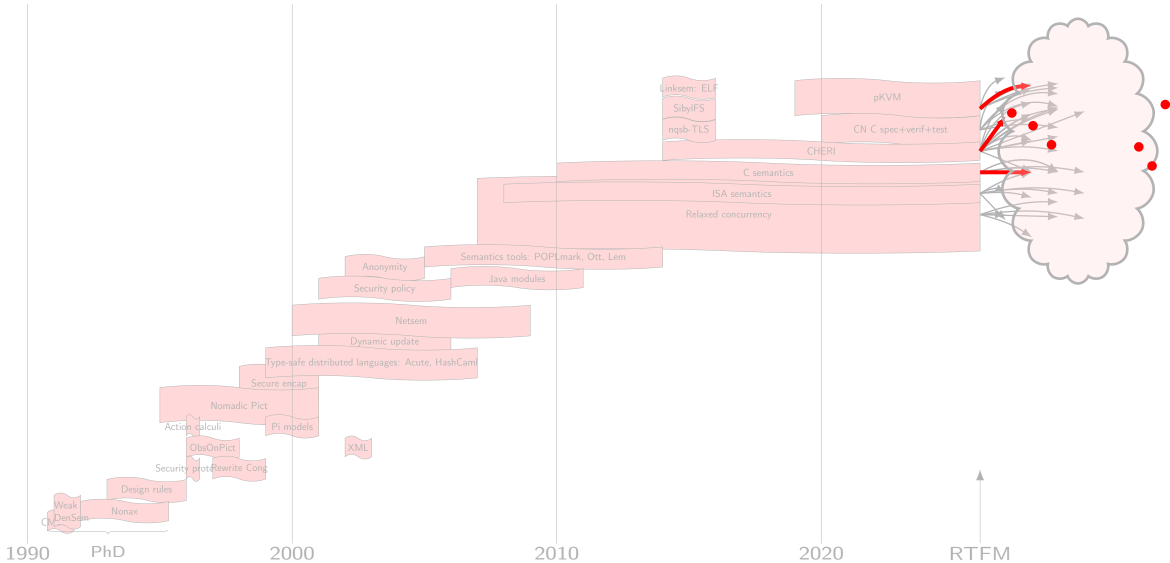
   That said, one doesn't necessarily have to *commit* up-front. Speculatively poking at something that might or might not work can often be worthwhile, for some modest investment. Sometimes it works out years later, when it's the right moment to pounce.

   Scaling the group size and interleaving multiple sub-projects helps mitigate the investment/payoff issues (modulo the context-switching cost).

   Quality not quantity. Ignore all bib metrics stuff as much as possible (in some settings one can't altogether). You and your team want to be known for something(s), not a metric.

   And one has to ensure that each person gets identifiable major contributions, on a timescale that fits their path. (We're not as first-author obsessed as some subjects, but still...).

How do we actually decide what to do next?

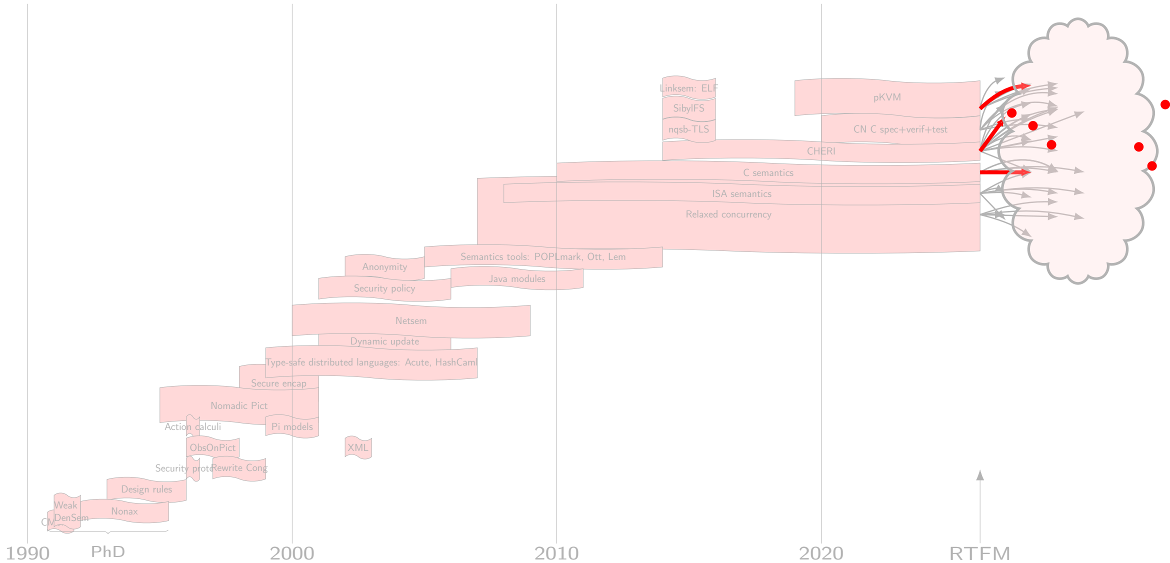


1. Most research takes more (sometimes much more) than one paper to really work out – as Derek likes to emphasise, most good research *is* "incremental", but choosing the step size is crucial.

   Think about the publishable units. People often speak of the minimal publishable unit (usually a terrible idea to aim for that). A bigger issue for us is the maximal publishable unit – the most that one can explain and carry the audience with in a single paper.

   cf Dickens serial publication – each paper has to be standalone, but also (if you want to get bigger things done) fit together. Though perhaps without the cliffhangers.

How do we actually decide what to do next?



1. There are many contradictory tensions:

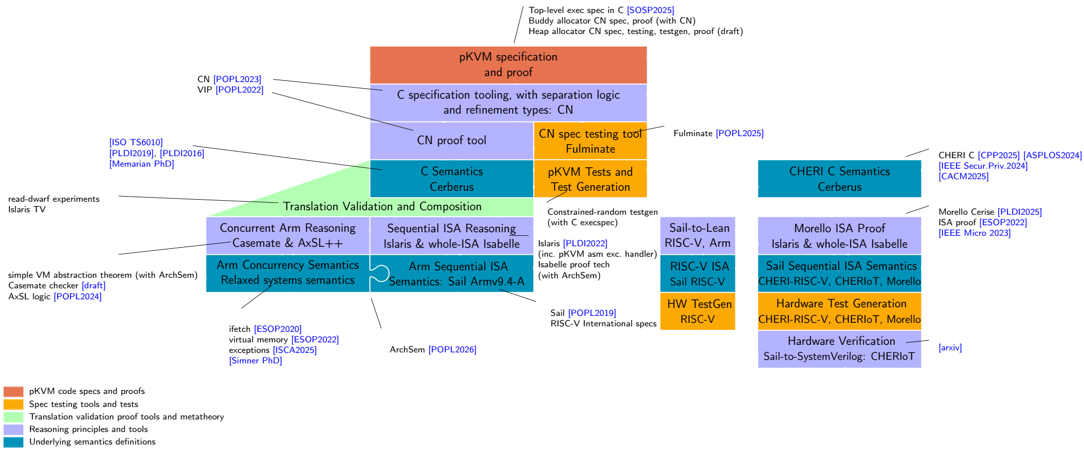
   Be opportunistic ... vs be driven by long-term goal

   Be ambitious ... vs be realistic

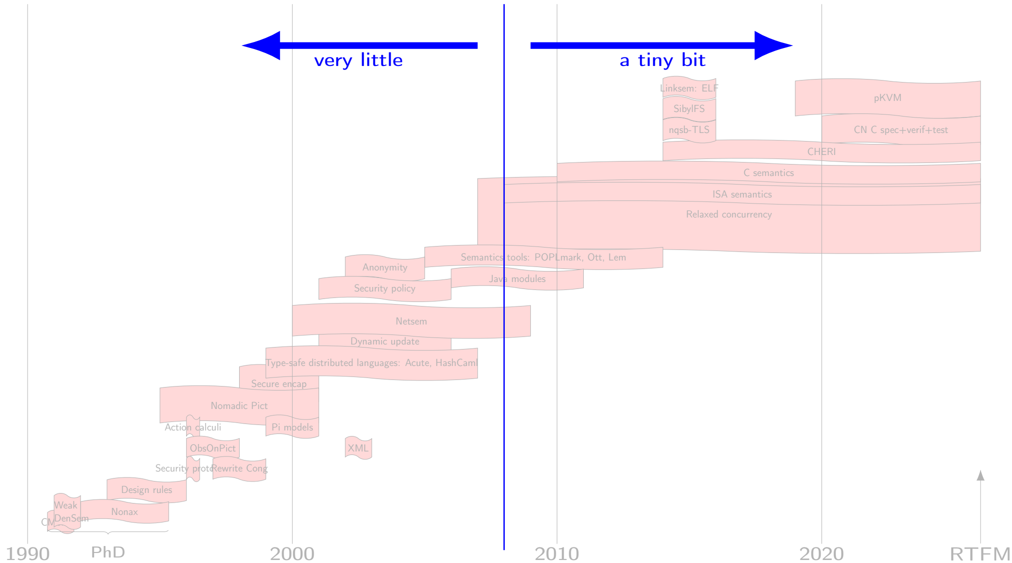
   Pay some up-front thought on how things might fit together – at least to avoid them *not* fitting together for stupid reasons. For example, the choice of proof assistant or other tool is a big issue, and details of definitions.

A different view of the frontier: snapshot of interest areas

Another useful view for explaining what one does: a snapshot of interest areas and recent work. This isn't a research agenda, though one can hang plans off it. For us, it can be structured roughly following bits of the stack, but there are many useful kinds of picture



Did any of that do any good?



Did any of that do any good?

1. That's ok: research is hard, and computing as a whole is a giant rock to try to push up hill. If we can move it just a bit, or smooth the path, that's already great.

Whose vision?

1. you the individual?
2. your research group (either your direct group or a larger collaboration)?
3. your subcommunity?
4. the SIGPLAN community?
5. computing as a whole?

Whose vision?

How focussed does it have to be? Monomaniacal vs follow+enable each student in whatever direction they fancy?

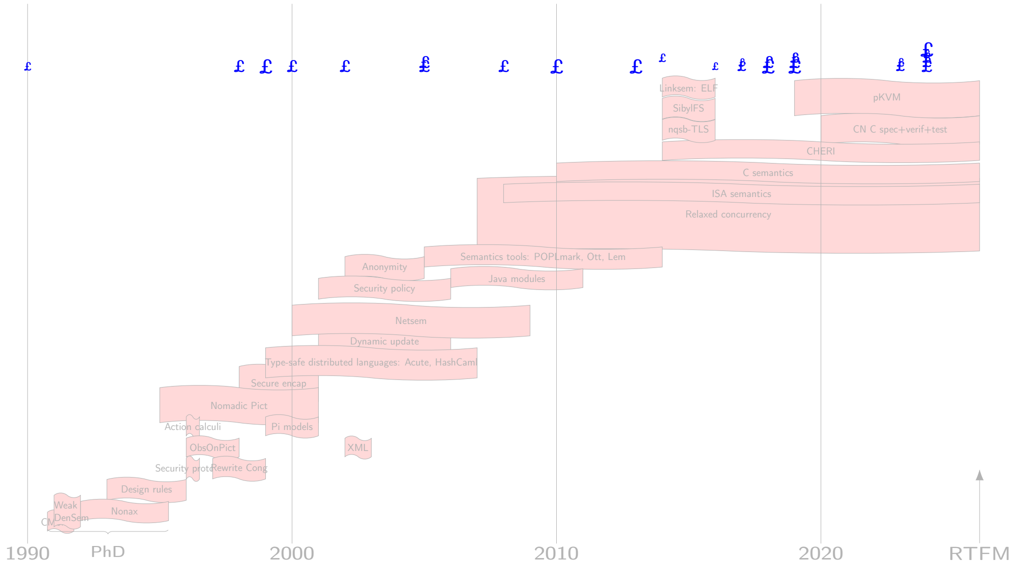
How individual does it have to be? Many high-level goals are shared across the community – perfectly good and proper

How do we as a broader community choose what to value and thus what to do? We're not very good at having conversations about that.

...which is intertwined with the fact that we're not very good at working at scale. PL as a whole miserably fails, compared to our collective potential. (Occasionally people or grant funding bodies try to identify challenges or roadmaps – e.g. Grand Challenges of Robin et al.)

...so make friends and connections, to enable far more than you can do alone. Though... choose your collaborators wisely, to avoid conflict from entanglement with those whose values – or simply whose schedules – don't fit well.

Necessary enablement



Necessary enablement

- funding
  - vision emerges? Think of writing proposals as a useful and creative activity...
  - make funded projects serve you, not vice versa
- people
- (as far as you can arrange) a financially and emotionally secure environment for your team – both because it's right and because it's necessary for everyone to do their best – we ask ourselves and our people to do impossible things...
- get good at writing, and speaking, and graphic design ... and understanding and supporting and persuading and mental-health 101 and ....
- somehow avoid becoming overwhelmed – learn to say no (enough)
- mostly ignore the imposter syndrome (though be humble in the face of reality, and be keenly aware of what you don't know)

Dealing with failure

dealing with PhD students / postdocs / collaborations that didn't work out

be supportive – but clear-eyed. Sometimes (though quite rarely) one needs to just say no.

dealing with rejection

Everyone has to (except Robin?). My anti-CV: UG (Cambridge) x1, Papers x23, Grants x10, Faculty x8

dealing with research failure

dealing with conflict

dealing with indifference

1. Learn from rejection: you didn't convince the readers; why not? Maybe the idea or work wasn't as good as it needed to be; maybe you didn't explain it well – and you need to. But don't over-learn; it's a noisy signal.
2. Persist! (but if need be, be realistic, and cut your losses)

Ultimate-goal motivation

What ultimate-goal motivations do we *choose*?

- get a job
- gain authority
- maximise h-index
- become rich
- become famous (in the SIGPLAN community...?)
- become respected (in the SIGPLAN community...?)
- serve the research community
- learn cool stuff
- solve neat puzzles
- solve hard problems
- solve important problems
- build useful systems
- write some nice-looking papers
- advance our understanding
- improve the way researchers do things
- improve computing as a whole

1. We might value some of those more than others...

Conclusion

hold scientific idealism, the joys of research, and the pragmatics simultaneously in your hand

make choices (and you have to make choices) intentionally and thoughtfully – not accidentally

it's all context-dependent

[Verse 1] I have a dream A song to sing To help me cope With anything If you see the wonder Of a fairy tale You can take the future Even if you fail

[Chorus] I believe in angels Something good in everything I see I believe in angels When I know the time is right for me I'll cross the stream I have a dream

[Verse 2] I have a dream A fantasy To help me through Reality And my destination Makes it worth the while Pushing through the darkness Still another mile

[Verse 3] I have a dream A song to sing To help me cope With anything If you see the wonder Of a fairy tale You can take the future Even if you fail

[ABBA, I have a dream, https://youtu.be/ER_3h03omdE?si=O27tgVdyWl3L0dj2]

“Ever tried. Ever failed. No matter. Try again. Fail again. Fail better.”

[Sam Beckett, Worstward Ho, 1983]