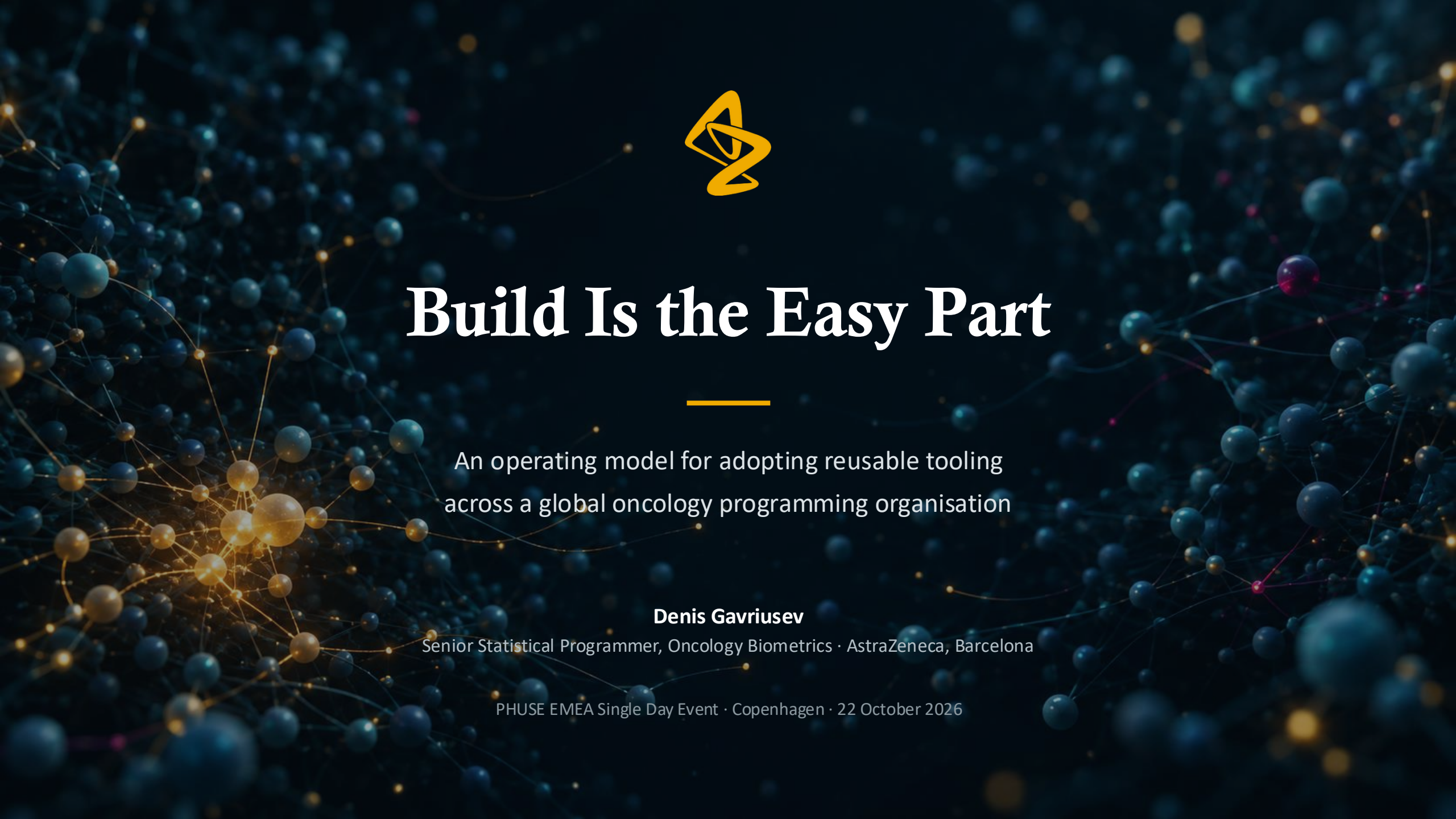


# Build Is the Easy Part

An operating model for adopting reusable tooling across a global oncology programming organisation

**Denis Gavriusev**
Senior Statistical Programmer, Oncology Biometrics · AstraZeneca, Barcelona

PHUSE EMEA Single Day Event · Copenhagen · 22 October 2026

WHERE THIS STARTS

# We had a catalogue. We had no evidence anyone was using it.



**1 of 5 champions rated our communication of tool updates as effective**

December 2025 · n = 5 · our own champions, not end users

**270+** tools in the catalogue

**70+** recommended for oncology

**300–400** programmers to reach

> *"Most programmers don't check the tool library when they need a tool."*
>
> a champion, same survey

**Every tool behind those numbers already worked. This was never a build problem.**

# Three toolkits, organised by task

**9** tools

**Study set-up & delivery oversight**

*What should be in place?*

→

**7** tools

**SDTM & ADaM**

*Submission-ready datasets*

→

**10** tools

**TLF production + 61 templates**

*Formatted outputs, ready to deliver*

## Choose tools by task, not by name.

**Champions own the toolkits**

- 5 part-time champions own the toolkits.
- They do this alongside study delivery.
- The central tools team contributes.
- Toolkit ownership is in personal goals.

*This was the design we tested. Not evidence that it worked.*

# We couldn't measure adoption. So we counted outreach attempts.

- **Adding usage tracking** — blocked on a decision about where code is stored — governance, not engineering
- **The tracking we have** — can't show which tool was used
- **Feedback forms** — few replies, and they don't tell us much
- **Champions** — can't make anyone use a tool - no authority, only relationships

THE ONE THING WE COULD ALWAYS COUNT

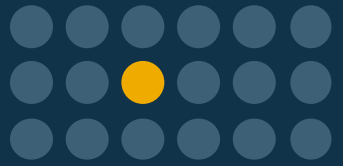
## Our own attempts to reach people.

Not adoption. But we could count it from day one.

SEVEN ATTEMPTS · SIX MONTHS

# One difference: existing meeting or new meeting.

**We used an existing meeting**



## 3 of 3

engaged — every one produced follow-up requests

- A 15-minute slot in a programme leads' meeting
- A slot in a meeting of ~60-70 contract programmers
- A slot in an existing developer meeting

**We asked people to come to something new**



## 0 of 4

progressed — two on capacity, two with no response

- A new monthly tools meeting
- An outreach pilot
- A shared mailbox: no messages in its first month
- Messages sent to 300-400 people

**Use an existing meeting before you create a new one.**

Seven observations. I sorted them afterwards. I judged them. This isn't controlled evidence. Nothing else we ran showed such a clear difference. That is why we acted.

SO WE CHANGED HOW WE ASK

# Same people. Same task. Different request.

BEFORE

*"Who can review this?"*

Open request to everyone. We never said what "done" meant.

**0** replies in 19 days

AFTER — NAME FOUR THINGS

- **the item to review**
- **what "done" means**
- **the time it takes**
- **the deadline**

*"Tell me where you got stuck — getting stuck is the finding, not a request for support."*

**3** reviewers, usable findings - the first request here ever completed

AND THE SIGNALS WE HAD INSTEAD OF A USAGE NUMBER

**A team adopted a macro they didn't build**

and now uses it in their own studies

**A library released this summer**

is used in its first study

**The barrier was the documentation**

the tool worked; the instructions didn't

## None of this is usage data. All of it is change we could see.

# What we still can't prove

**Team sessions**

A short tools slot inside a meeting the team already holds. We agreed to start four times — April, May, July, August. We have not run one yet. Nobody ever refused, study deadlines always took priority.

**Usage data**

Limited. We can't add usage tracking until we decide where code is stored. So I can't tell whether anyone used a tool.

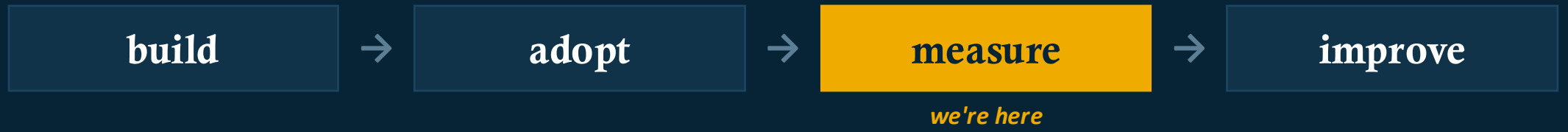
**Updates to teams**

The toolkits are rolled out organisation-wide. One champion has passed them to their team; most haven’t yet. We assumed programme leads would pass updates down — one tried, and the updates stopped one level above the teams.

## “Rolled out” and “adopted” are different words.

I can prove we rolled out. I'd rather say that than show you a chart.

# A loop, not a launch




| | | |
|---|---|---|
| 1 | **Decide how you'll measure usage before you publish.** | Adding it later makes it a governance decision, not an engineering task. |
| 2 | **Can't measure adoption? Count outreach attempts.** | You can count them from day one - and they're the part you control. |
| 3 | **Use an existing meeting, not a new one.** | Best lesson from seven attempts. Not a rule. |
| 4 | **Name four things in every request.** | The item to review, the time it takes, what "done" means, the deadline. |
| 5 | **Don't report rollout as adoption.** | Two different milestones. We have reached the first one. |

**Build is the easy part. Knowing whether anyone uses it — that's the hard part.**

# Thank you

If you've solved the measurement problem,
I'd like to hear how.

Questions welcome.



**Denis Gavriusev**
Senior Statistical Programmer
Oncology Biometrics
AstraZeneca · Barcelona

**denis.gavriusev@astrazeneca.com**
Oncology Programming Tools Champions

AstraZeneca