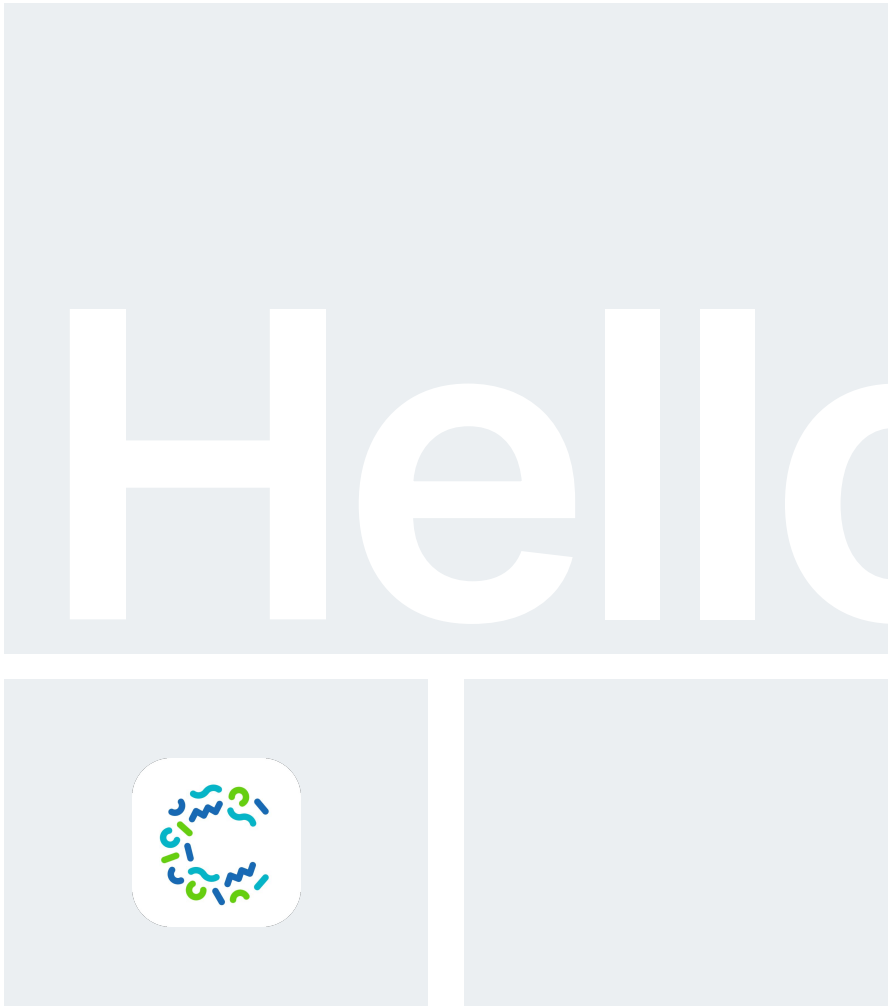
PEGS 2025

Lessons in building a platforms for engineering biologics



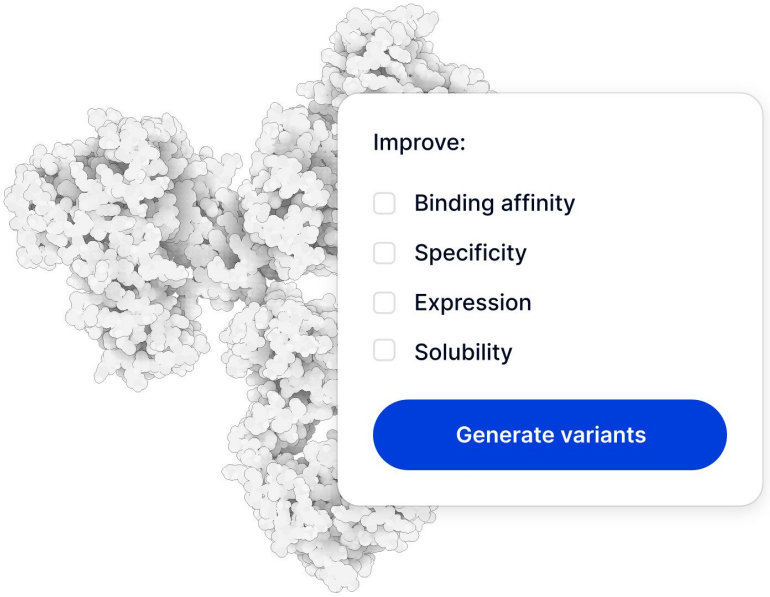
Stef van Grieken (stef@cradle.bio)

Stef van Grieken

Now and before

Cradle Co-Founder and CEO	All the time
Operator Exchange Angel investing together with cool people	Sometimes
Google (AI, X, Flights, Android Auto) Group product manager	2014 - 2021
OpenState Foundation Founder and Chairman	Before that
Google Software Engineering Intern	Long time ago
European Parliament Policy Advisor	Very long time ago

Cradle makes engineering biologics easy & accessible.



Think Dall·E, ChatGPT, Stable Diffusion or Github Copilot – but for engineering proteins.

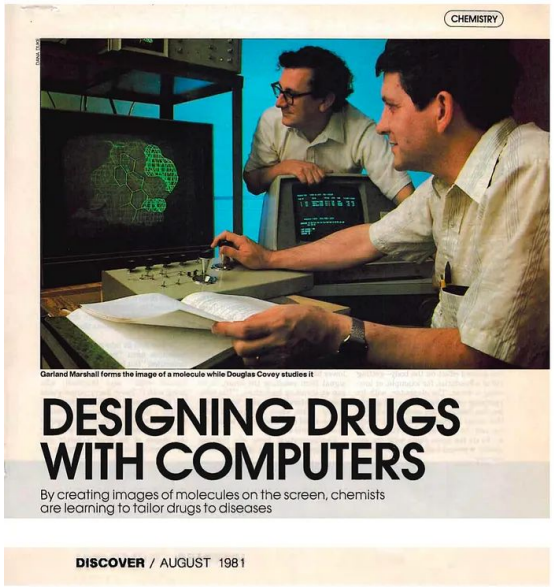
26 Customers

7 of top 20 pharma, 2 of top 5 industrial bio and 1 of top 3 agribusinesses

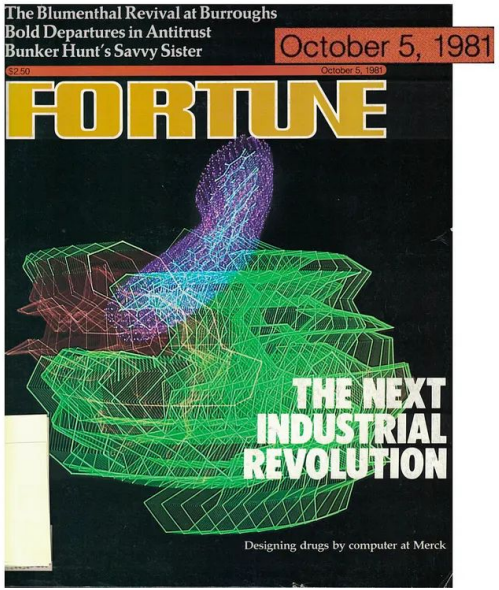
+45 Programs

Tech has promised to solve biology since 1981

A



B



Setting the stage

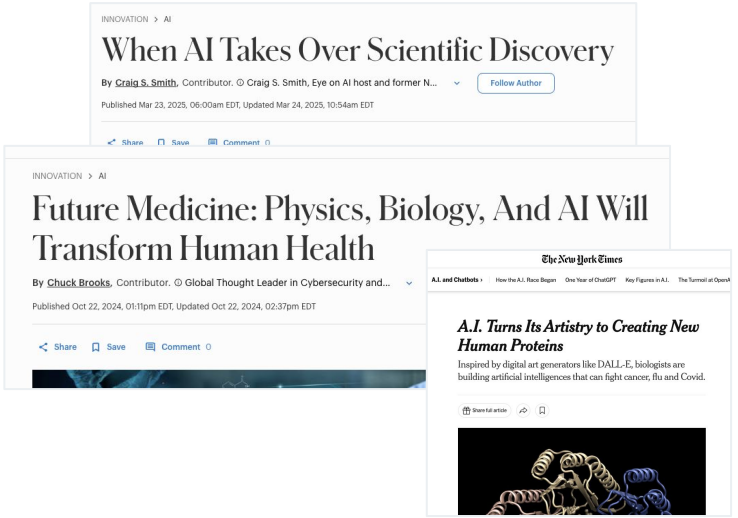
We’re watching the same movie… again

Then (1981):

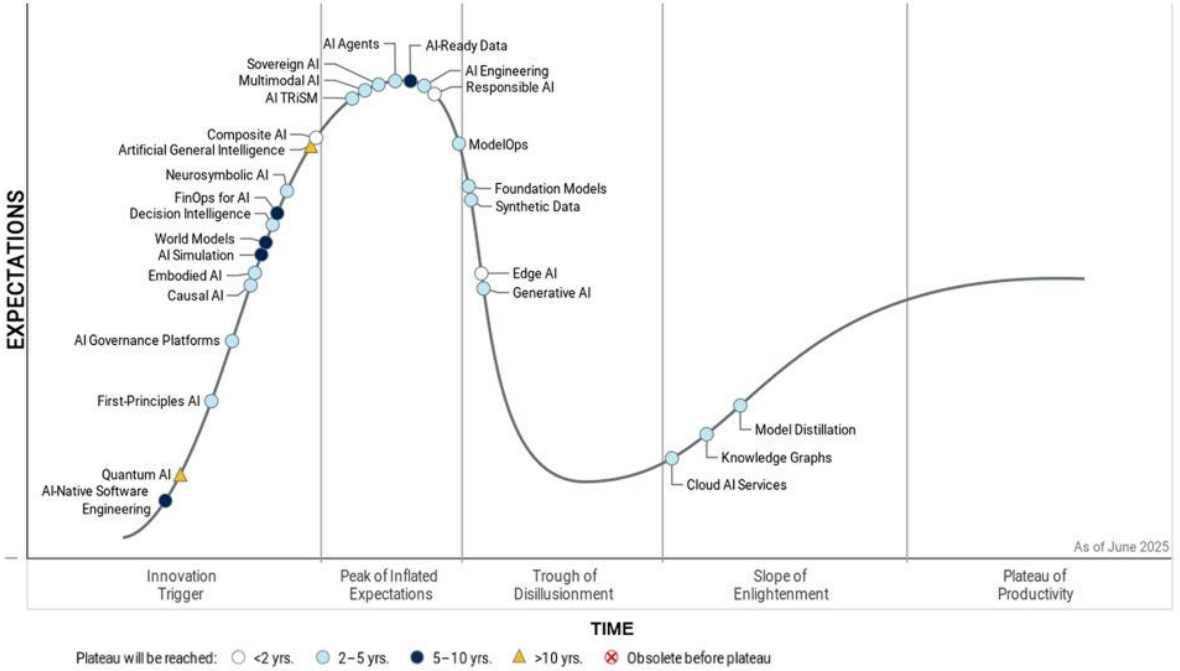
"Computers will design drugs"
CADD (Computer-Aided Drug Design)

Now (2025):

"We can just zero-shot our way to the perfect protein"
Foundation models for proteins



This time the focus is on Foundational Models

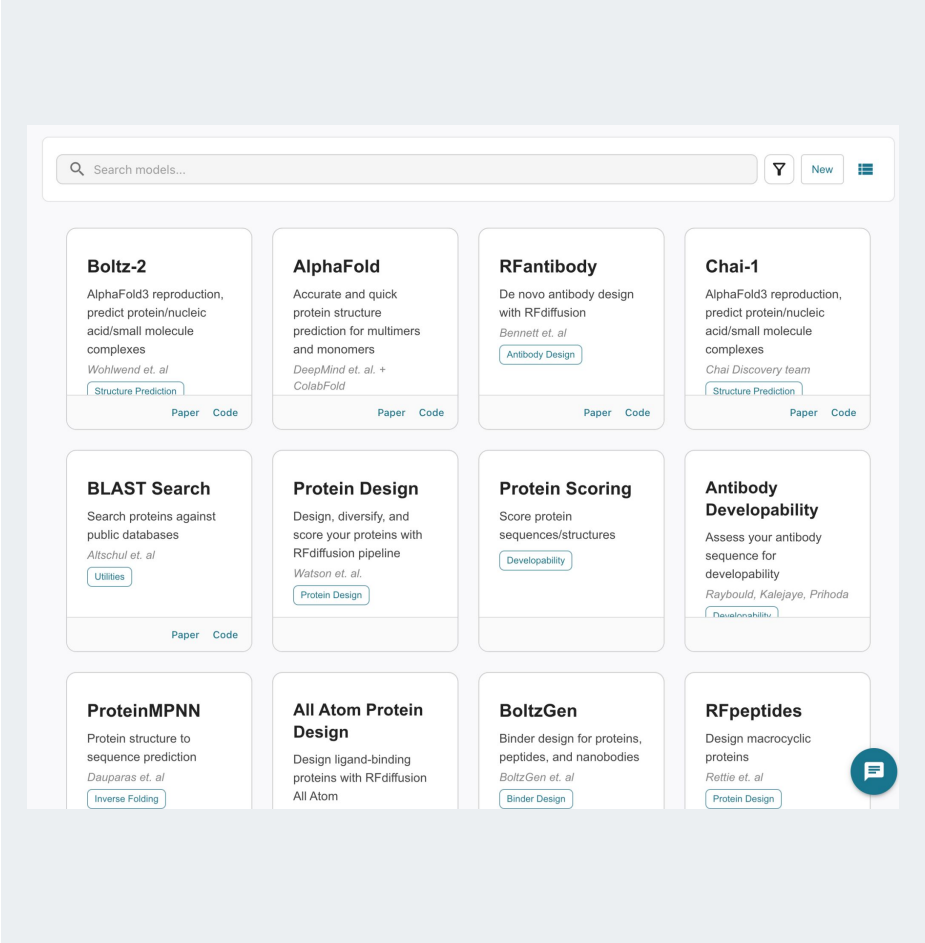


Which has lead to most companies building ‘model zoos’ to enable teams

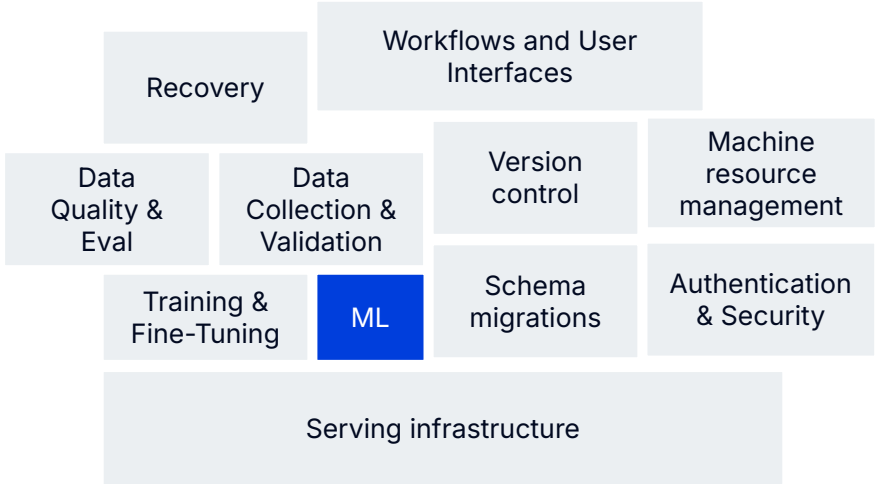
What many companies are building:

- Dozens of separate, task-specific models (binding predictor, structure predictor, inverse folding model, stability predictor, homology search model, peptide generator)
- Requires teams of computational PhD to build protocols and consult with therapeutic teams

This doesn’t scale



At Google, models are only 5% of the code base



Sculley, D., Holt, G., Golovin, D., Davydov, E., Phillips, T., Ebner, D., Chaudhary, V., Young, M., Crespo, J.-F., & Dennison, D. (2015). **Hidden Technical Debt in Machine Learning Systems.** In *Advances in Neural Information Processing Systems 28 (NIPS 2015)*, pp. 2503-2511.

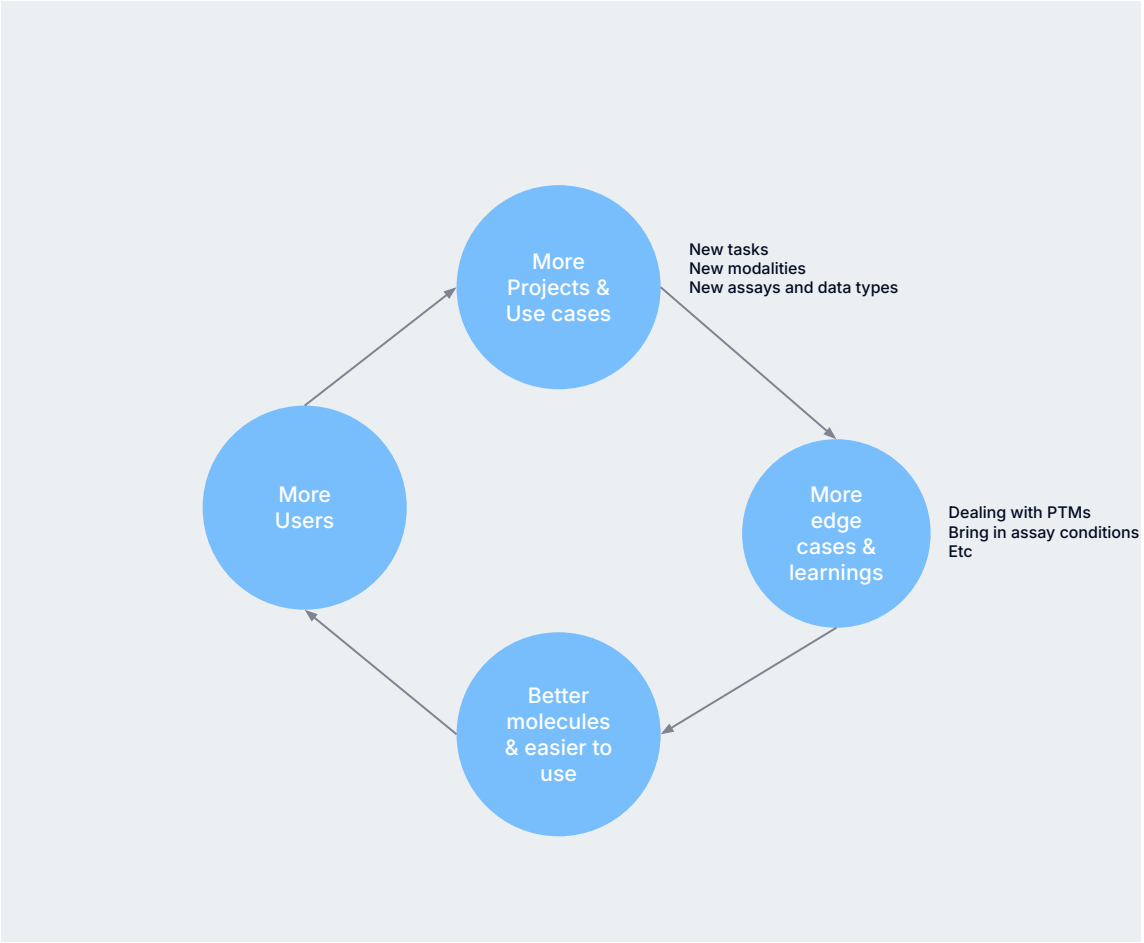
Setting the stage

Platforms beat models

Moving from models to platforms.

- Bespoke, one-off models put together by smart phd's initially win on local use cases (i.e. language translation, search ranking)
- After launch, within months the automated system started to outperform the expert teams.
- Why centralized won: bespoke teams face time and pressure constraints, centralized systems encode best practices, robustness, and scalability improvement compound at scale as they are forced to solve diverse problems

Here are some of our lessons



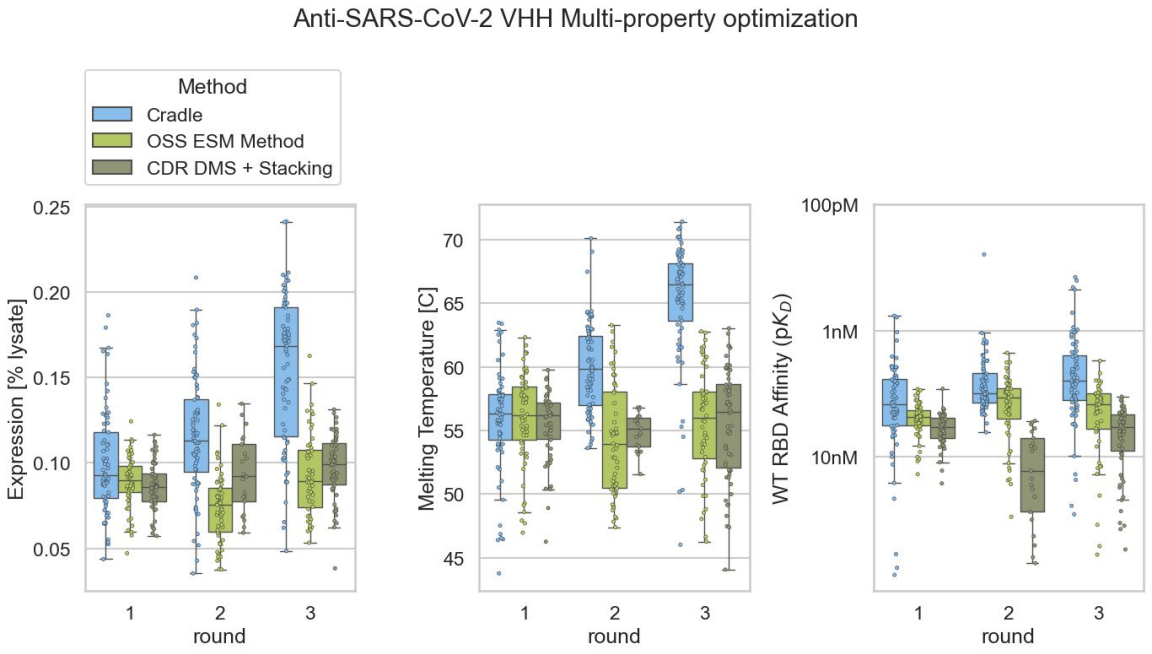
Setting the stage

Benchmarking 3 approaches

Benchmarking Cradle on a VHH against SARS-COV-2

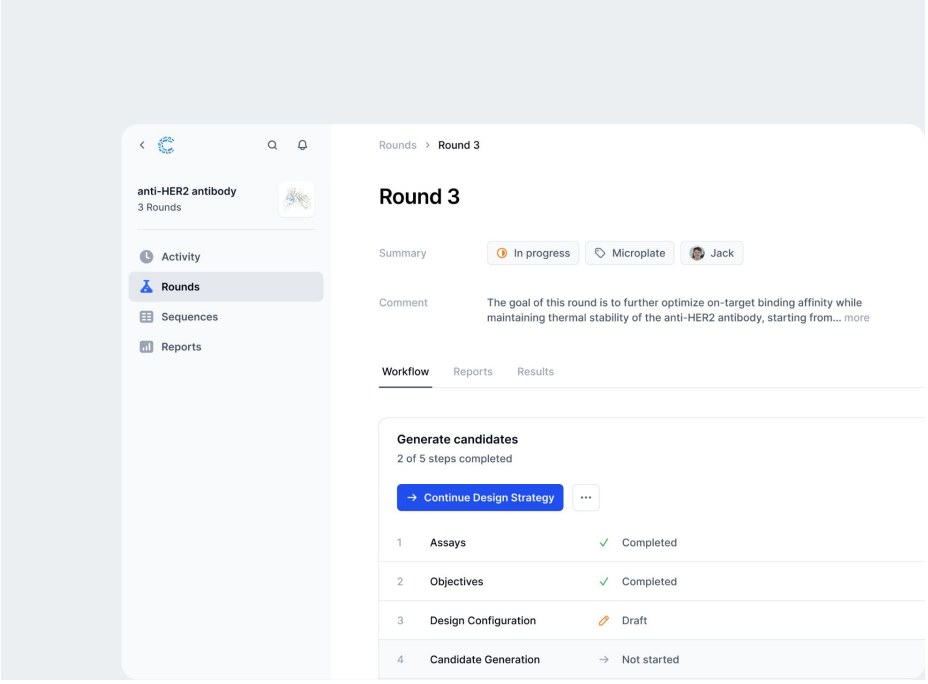
- **Baselines**: we benchmark against Evolutionary Scale (ESM) and a more conventional mutation scan + mutation stacking strategy
- **Rounds**: diversification from a know starting point with 2 optimisation rounds

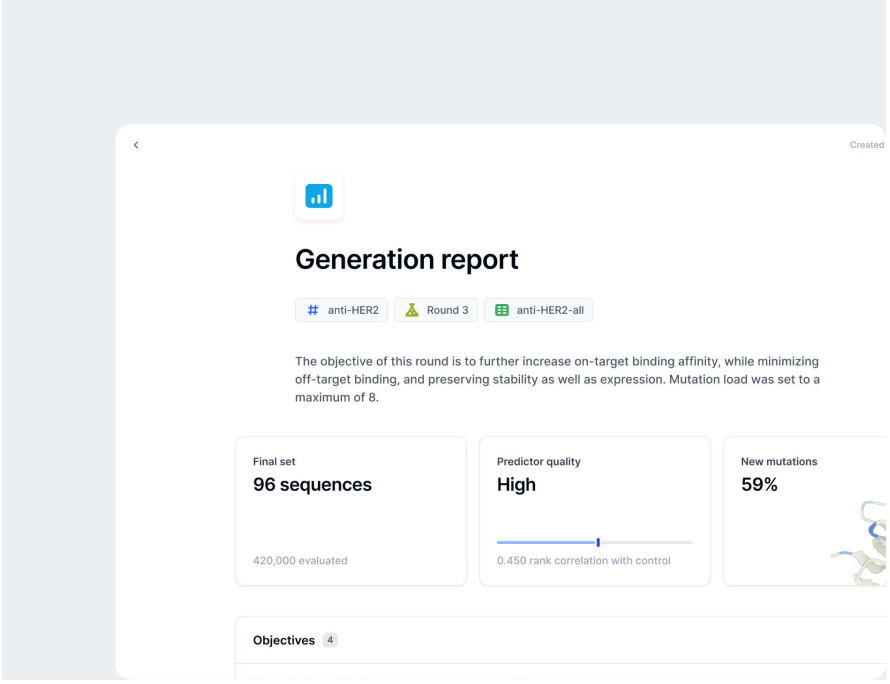
Here are some of our lessons



Workflows and interfaces

You want to fit in a scientists workflow





Setting the stage

Prompting

Moving from models to platforms.

- Bespoke, one-off models put together by smart phd’s initially win on local use cases (i.e. language translation, search ranking)
- After launch, within 2 quarters the automated system started to outperform the expert teams.
- Why centralized won: bespoke teams face time and pressure constraints, centralized systems encode best practices, robustness, and scalability improvement compound at scale from solving diverse problems

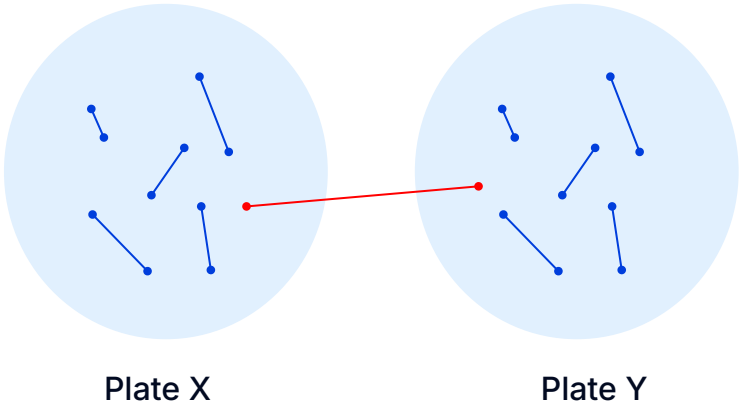
Assay	Scale type	Objective/constraint
Thermostability max	+ Additive	Optimize · Increase
Thermostability min	+ Additive	Predict only
Expression	+ Additive	Constrain · Keep above selected sequence
Dissociation Constant (wildtype epitope)	* Multiplicative	Constrain · Keep below selected sequence
Dissociation Rate (wildtype epitope)	* Multiplicative	Predict only
Association Rate (wildtype epitope)	* Multiplicative	Predict only

Data Collection, Data Quality & Evaluation

Experimental data is often of low quality

Round to Round, Day to Day, and Plate to Plate lab data varies in quality. GenAI need to be robust to the messiness of biology.

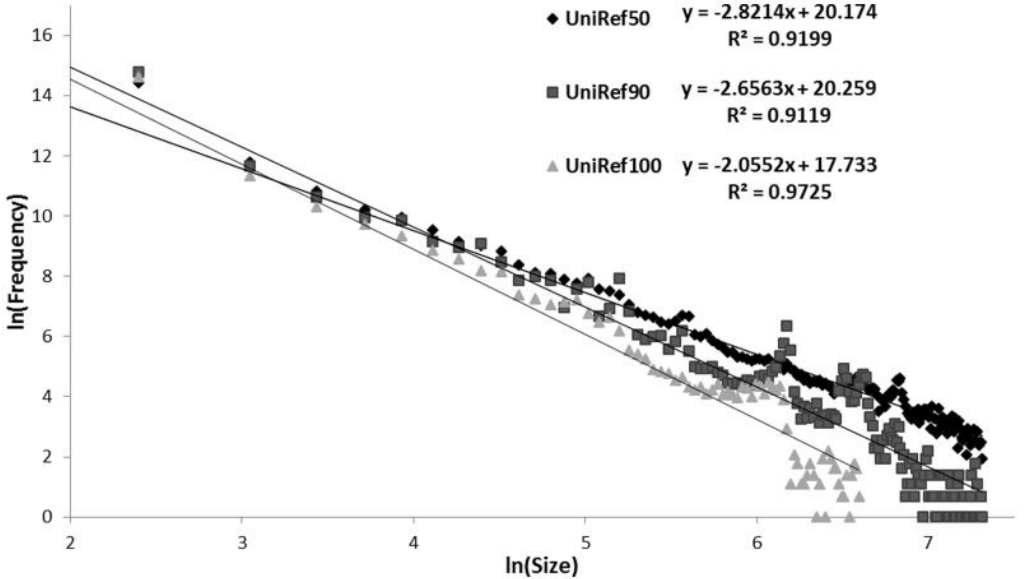
Learning across assays, plates and rounds using contrastive learning



Contrastive Learning allows models to learn relationships between lab data points that cross experiments. The nature of biological research is such that experiments are so different you can only compare pairs within the same experiment.

We have very little data in biology, which significantly limits our ability to generalise

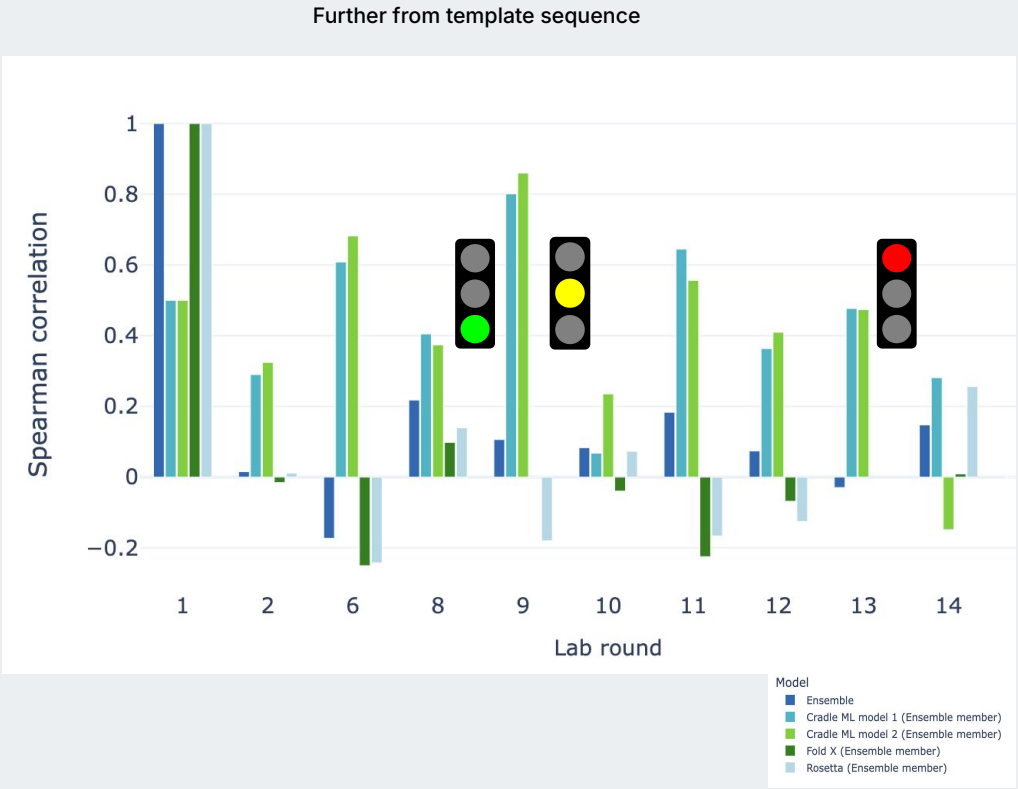
Compared to human language we have only observed very few sequences, and almost negligible measurements



Suzek BE, Wang Y, Huang H, McGarvey PB, Wu CH; UniProt Consortium. UniRef clusters: a comprehensive and scalable alternative for improving sequence similarity searches. Bioinformatics. 2015 Mar 15;31(6):926-32. doi: 10.1093/bioinformatics/btu739. Epub 2014 Nov 13. PMID: 25398609; PMCID: PMC4375400.

This means that how good your predictor may be, you will run out of domain quickly

Many models are hard to update with new data. Given how limited data availability is in biology, active learning will likely be required for many problems. Most advanced organisations realise this and implement ‘lab in the loop’ methods.



Models will only improve if engineers can test them

Most teams only collect data for the purpose of developing a specific protein product and are not allowed to collect data for the purpose of building AI tools.

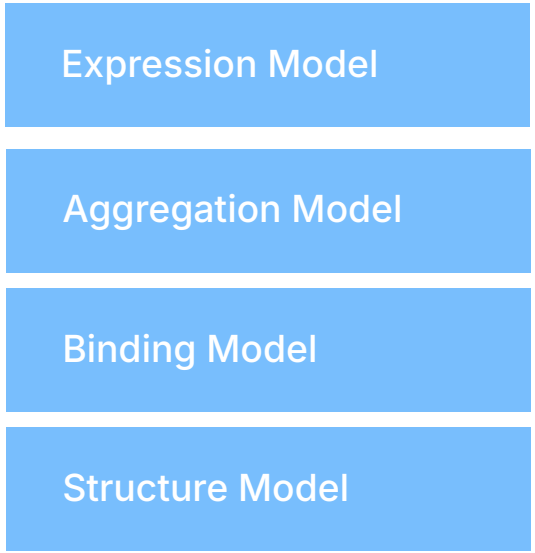
This is a bit like building a self driving car company without cars.

Training & Fine Tuning

Learning across properties & modalities

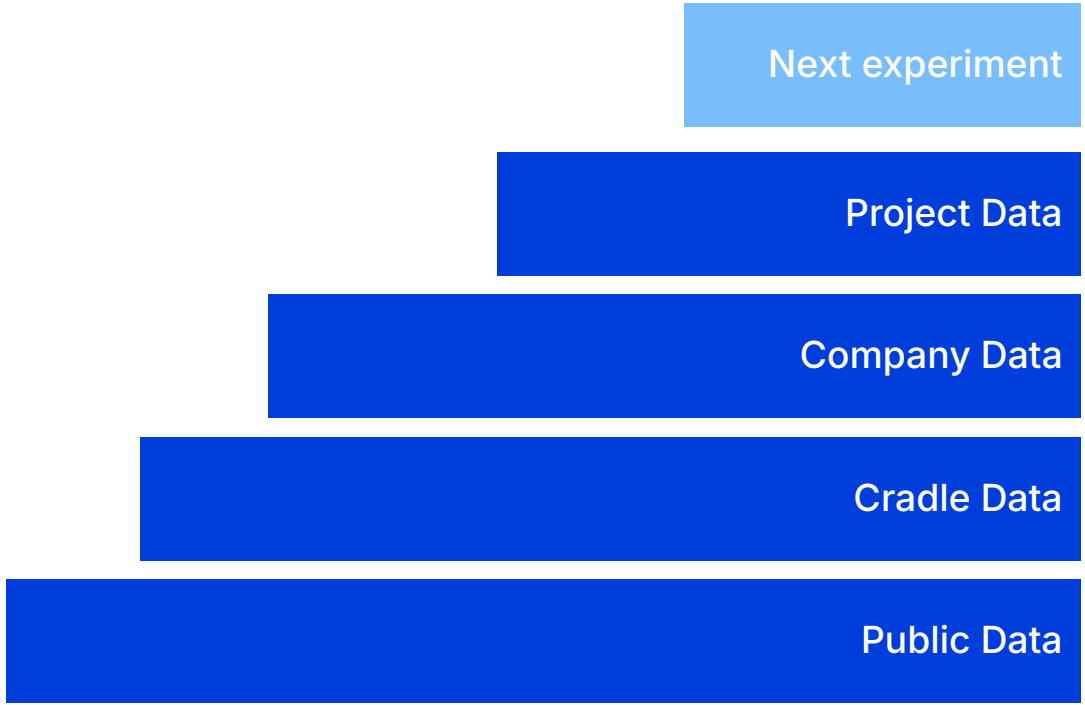
Platforms should be able to support a broad range of functions and assays that are required to get to a fully developed biologic.

- Modelling single properties
- Modelling multiple properties and have humans compare them
- Modelling multiple properties and their relationships to each other



Learning from all data

Platforms should be able to support providing access to diverse sequence and experimental readout datasets and automatically determine what is relevant to fine tune on given a specific experimental round.

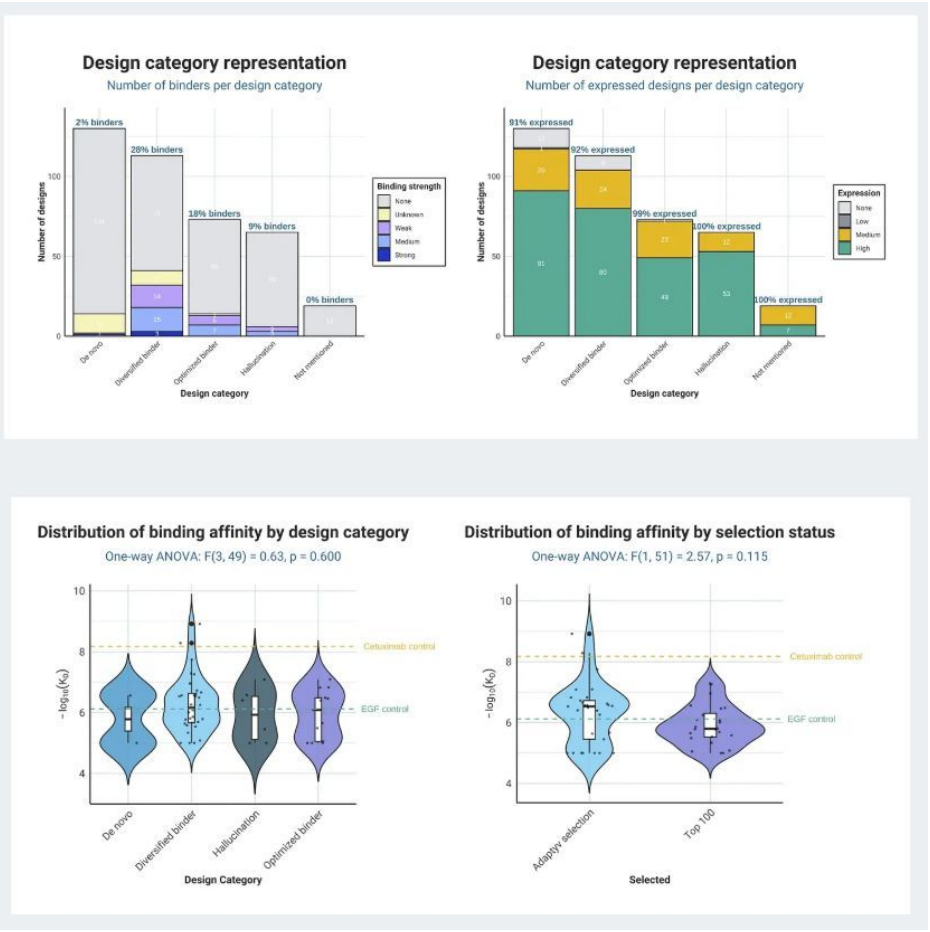


Learning across your project

Platforms should be able to learn longitudinally from assays that are related.

- **Library > Binding > Functional**: many projects start with a library screen, then move to a binding assay to end up with a functional screen at the end. They are correlated, but proxies for each other

Reducing human error through hyperparameter tuning



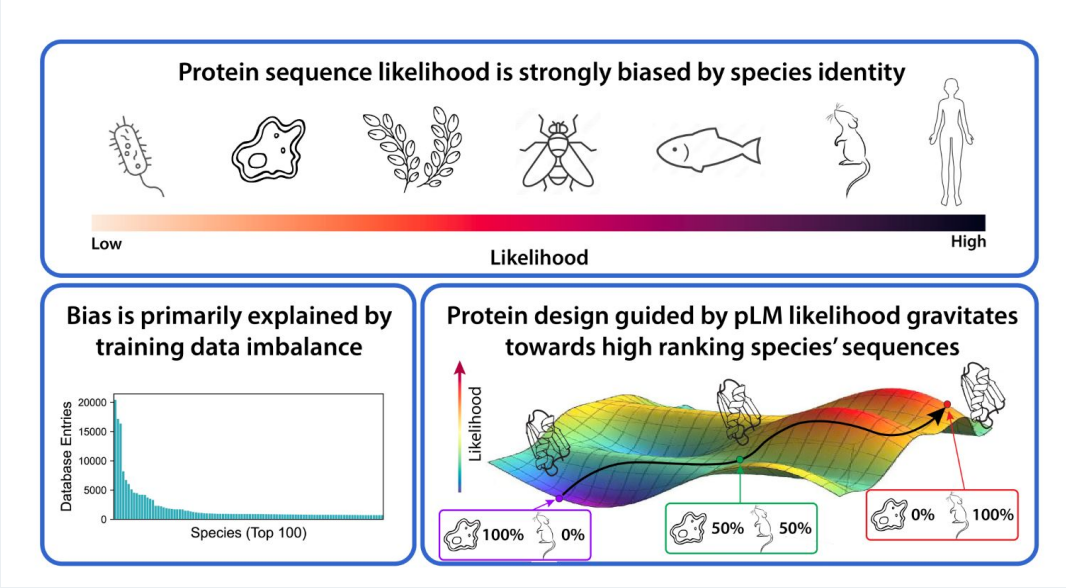
Platforms need to be robust in their results to changes in training & tuning configuration.

- **Human model configuration & evaluation:** many models expose hyper-parameters (configuration variables) to change the models behaviour. These hyperparameters are typically tuned by humans using bespoke protocols on in-silico metrics
- **Competition**: at the Adaptyv competition showed 130 teams trying to find binders against EGFR

Dealing with bias and unbalance in labels

Sequence and experimental data is not collected equally. Models will prefer finding solutions in places where they have more data.

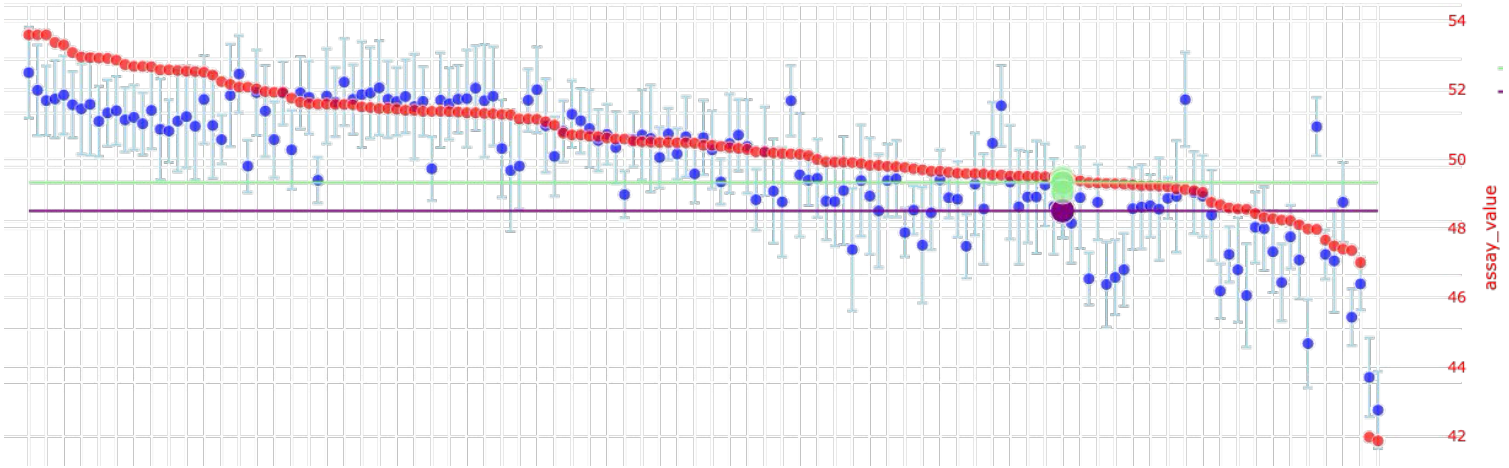
- Species bias for sequence level data is enormous.
- Label bias for experiments is also enormous as some experiments are easier than others



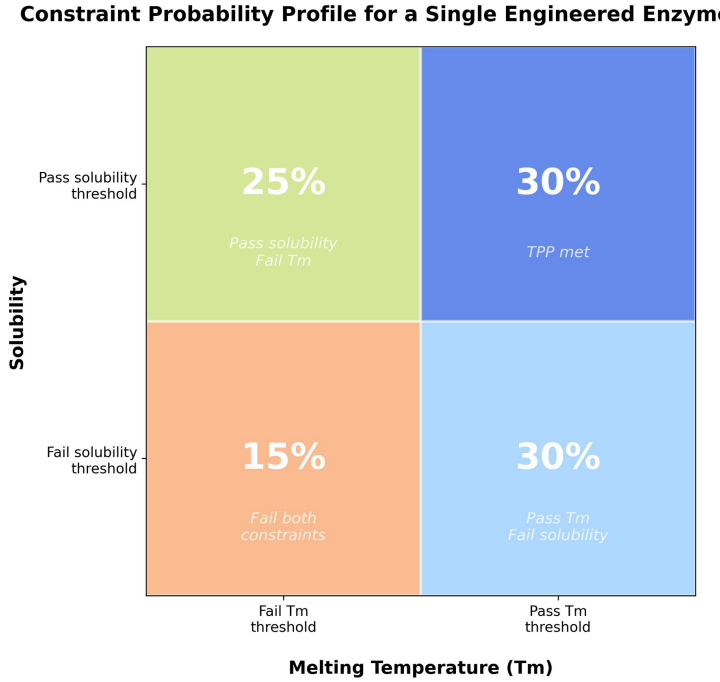
Protein language models are biased by unequal sequence sampling across the tree of life Frances Ding, Jacob Steinhardt
bioRxiv 2024.03.07.584001; doi: https://doi.org/10.1101/2024.03.07.584001

Dealing with uncertainty

95% hit rate
10°C Tm improvement
0.83 Spearman correlation



Designing experiments, not sequences

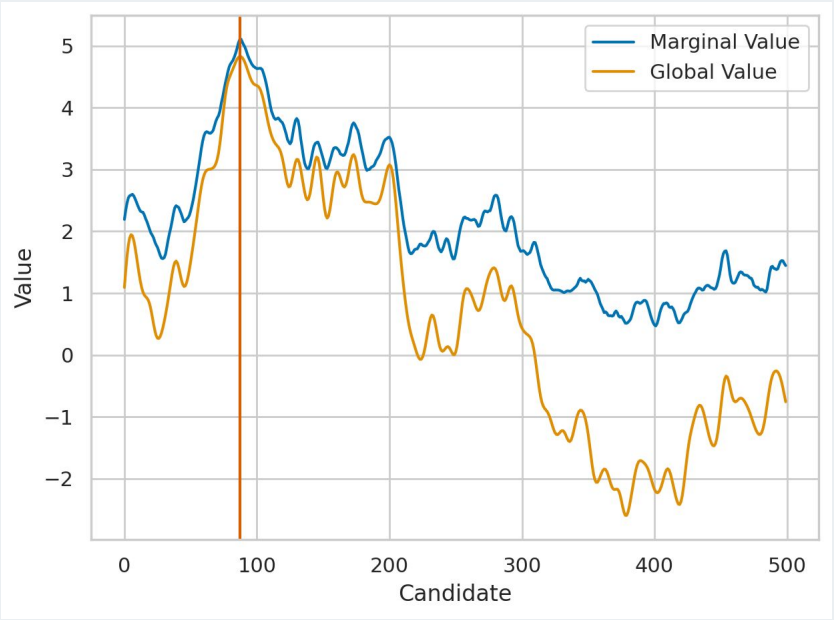


Platforms need to know their own limitations and adjust experiments accordingly.

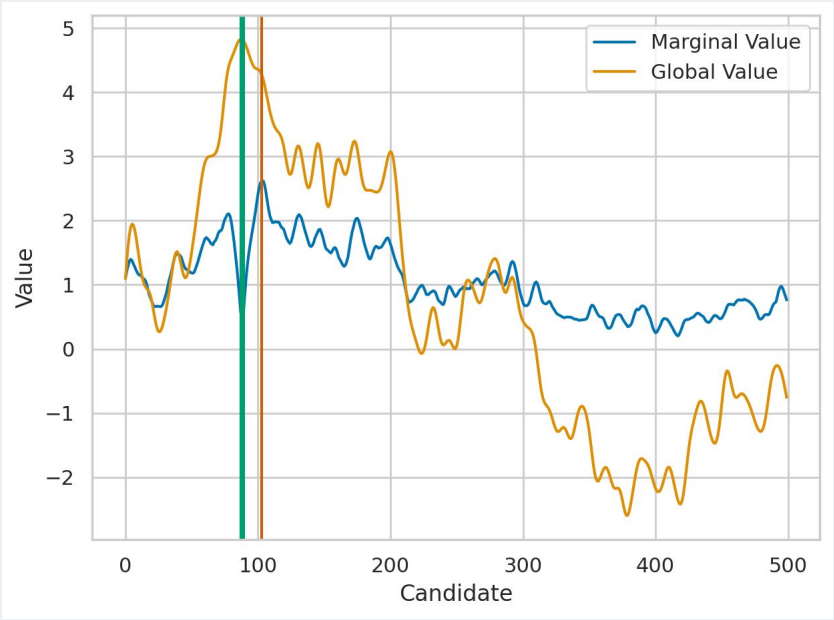
- **Your model will be wrong**: all models will make errors, and they will be wrong in systematic way.
- **Designing plates:** designing plates is about balancing risk and opportunity: because we can risk-manage, we can make higher-risk bets in poorly-characterized regions which can pay of both in performance and learning

Designing experiments, not sequences

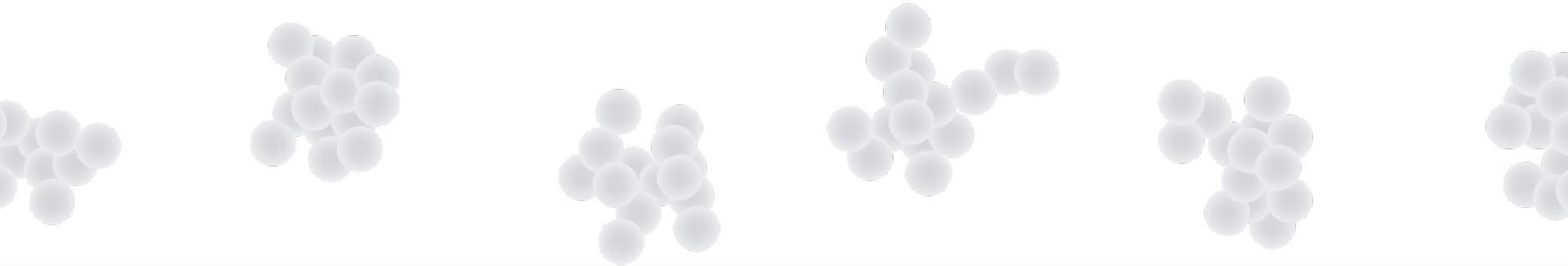
Similar variants “exclude” each-other, as they respond similarly to noise, and contribute relatively little to the expected best value.



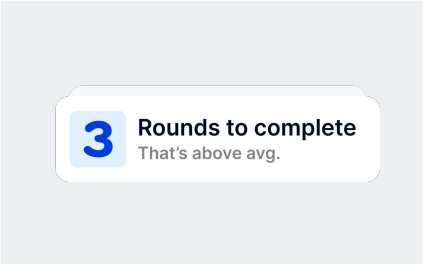
After subsampling, our algorithm reduces the occurrence of strong beneficial mutations from over 10% to less than 2% while maintaining round quality, resulting in significantly increased diversity.



A ‘full de-novo’ generative model for protein engineering should be able to generate a single protein sequence, that meets required target product profile without relying on new experimental data



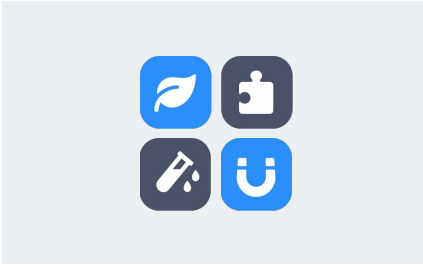
Thinking about generalisation in proteins



Generalising in rounds and throughput

Teams require less data per round and fewer rounds to get to good results.

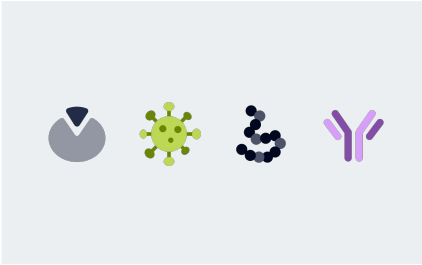
- Multi-round: for a property in a project teams require N rounds (multi-shot) in high throughput
- One-Round: for every property in a project, teams require 1 round (1 shot) in high throughput
- Zero-Round: for every property in a project teams require no experiments (zero-shot)



Generalising in properties and assays

Teams need to measure fewer properties to get good results

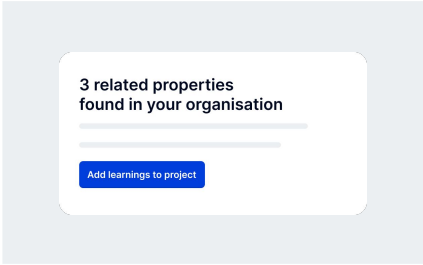
- All properties: for each project, every property needs to be measured
- 'In-Silico' / Zero shot properties: for each project, only some properties need to be measured
- 'De Novo' properties: for each project, no properties need to be measured empirically



Generalising in proteins and formats

Teams need to measure fewer properties for fewer modalities and formats

- Some Formats: round and/or properties generalise for formats
- Some Modalities: rounds and/or properties generalise for all modalities
- All Modalities: rounds and/or properties generalise for all modalities



Generalising in scope

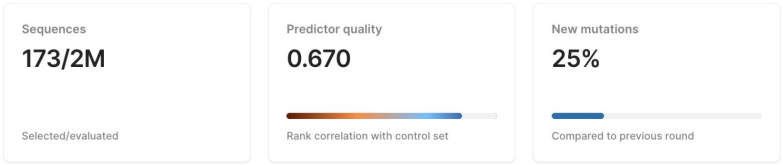
Teams need to measure fewer properties depending on the project, organisation or consortium.

- Project: everything needs to be measured at the project level
- Organisation: some properties, modalities or formats generalise at the organisation level
- Consortium: some properties, modalities/formats generalise at the consortium level
- Global: all properties generalise at every level

Scale, Infra, Security

Scale, Infra & Security

Complexity of the challenge requires scalable infrastructure

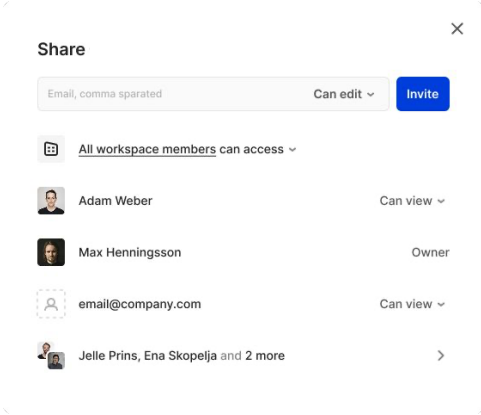


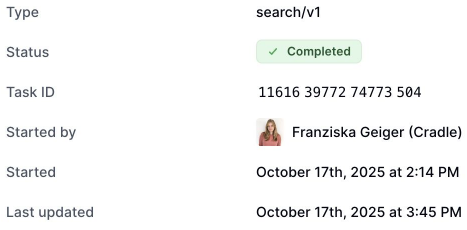
Platforms need to adjust their ability to use compute to the complexity of the task.

- **Millions to billions of variants**: for complex problems models may generate millions to billions of variants that automatically get validated.
- **ML operations:** maintaining this type of infrastructure at scale is complex

Scale, Infra & Security

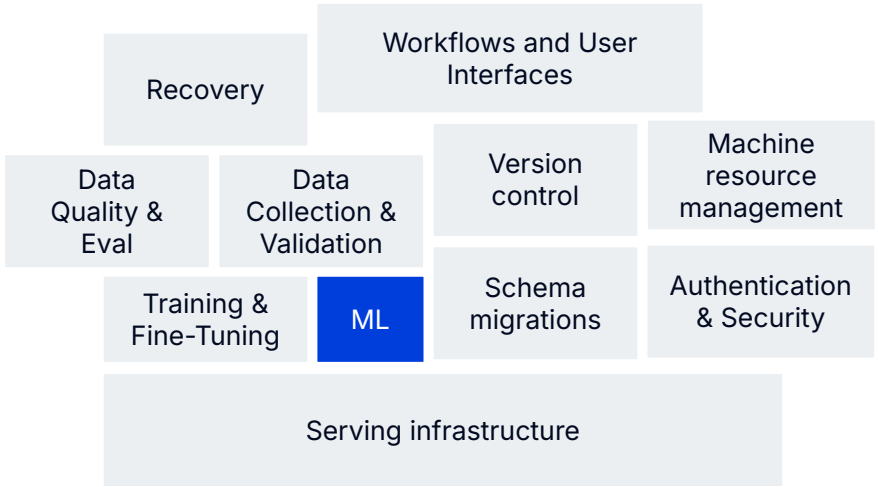
Some of your most valuable intellectual property needs to be protected





- **Access Control**: modern teams comprise of protein engineers, translation scientists, md's, outsourced scientists at CRO's.
- **Audit**: : knowing how a result was produced, who had access and was involved
- **Security:** you need bulletproof security as outside and inside actors are getting ever more sophisticated

Don’t underestimate the 95%



Sculley, D., Holt, G., Golovin, D., Davydov, E., Phillips, T., Ebner, D., Chaudhary, V., Young, M., Crespo, J.-F., & Dennison, D. (2015). **Hidden Technical Debt in Machine Learning Systems.** In *Advances in Neural Information Processing Systems 28 (NIPS 2015)*, pp. 2503-2511.

How to play

Building a minimal in-house platform will cost at least $34-68m

Building an open-source application scientists can use would cost $17 million a year on a shoestring and will likely require 2-4 years to get to completion if you believe you are good at building software and ML models based on open source.

Back of a napkin

MINIMUM BUDGET

Category	Unit	Cost per Unit	Total Co
Expenses			
Machine Learning Researcher	5	$550,000.00	$2,750,000.00
Machine Learning Engineer	6	$430,000.00	$2,580,000.00
Software Engineer	4	$350,000.00	$1,400,000.00
Front-end Engineer	3	$350,000.00	$1,050,000.00
Bioinformatician	2	$350,000.00	$700,000.00
DevOps	1	$350,000.00	$350,000.00
Designer	1	$350,000.00	$350,000.00
Engineering Cost			**$9,180,000.00**
Project Manager	1	$180,000.00	$180,000.00
General & Admin			**$180,000.00**
Compute	12	$300,000.00	$3,600,000.00
Licences	2	$60,000.00	$120,000.00
Other Cost			**$3,720,000.00**
Research Associate	4	$100,000.00	$400,000.00
Scientist	2	$430,000.00	$860,000.00
Reagents & Consumables	12	$500,000.00	$6,000,000.00
Experimental Validation			**$3,720,000.00**
Total Expenses (per year)			$16,800,000.00

THANKS!

Get the slides: cradle.bio/pegs

Psst! We're hiring
cradle.bio/careers

Website	cradle.bio
Email	stef@cradle.bio