Generators as "Platforms"

Gabriel Perdue // @gnperdue
Fermi National Accelerator Laboratory // @Fermilab
ECT, Trento, July 2018

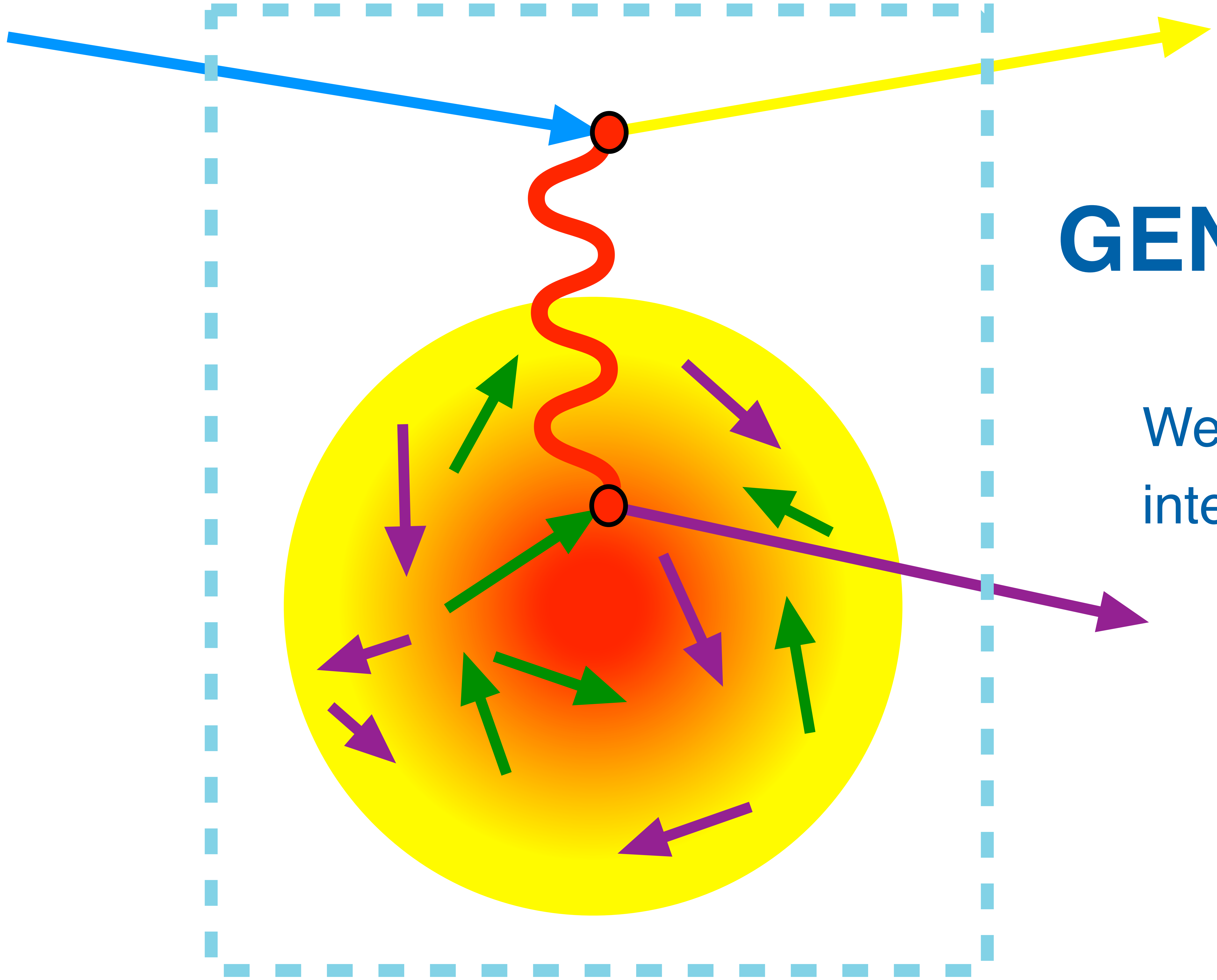
Outline

- What would a generator as a "platform" look like? (What the heck is Gabe talking about?) What problems are we trying to solve?
- How could we feasibly hope to achieve this "vision"? What sort of resources would be required?
  - For both of these sets of questions, **I will offer my opinions - they are just that - my personal opinions and not the position of FNAL/MINERvA/GENIE/etc. (or any of the groups I could feasibly represent here).** They are also not definitive in any way - a project like this will require a great deal of input and engagement with the whole community.

- *Note: these proposals/suggestions are in large part based on a cloud of thoughts originating with Stefan Prestel (Pythia) and Minerba Betancourt (MINERvA, ICARUS) and contain threads of "folk history" from the evolution of event generators at the energy frontier from Stefan.*

https://indico.fnal.gov/event/15600/session/0/material/0/0.pdf

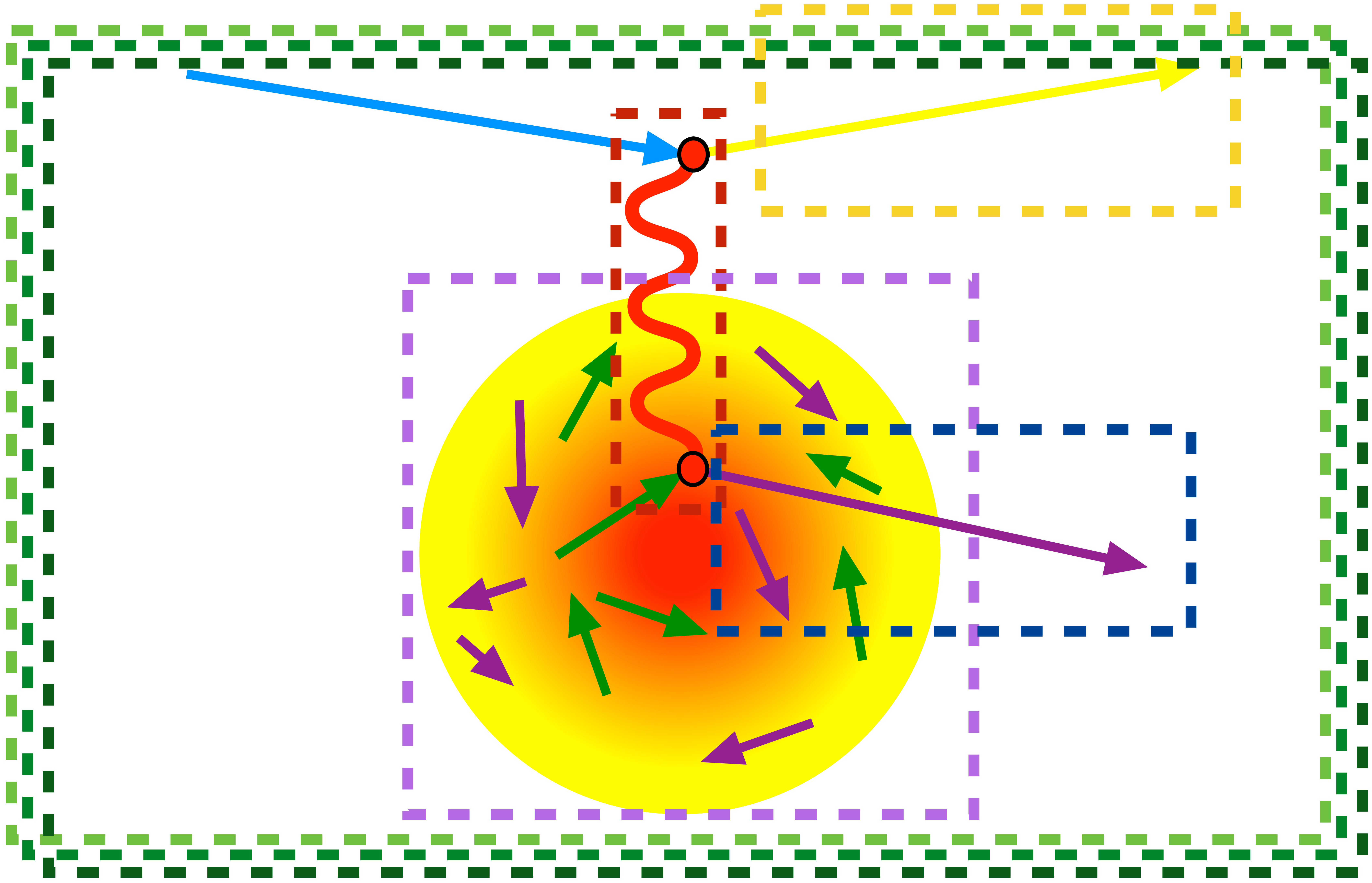
How things work today in neutrino physics...



GENIE

We have a full, vertically integrated solution.

How things would look today if we were at the LHC...



Fits:

Groups A, B, C

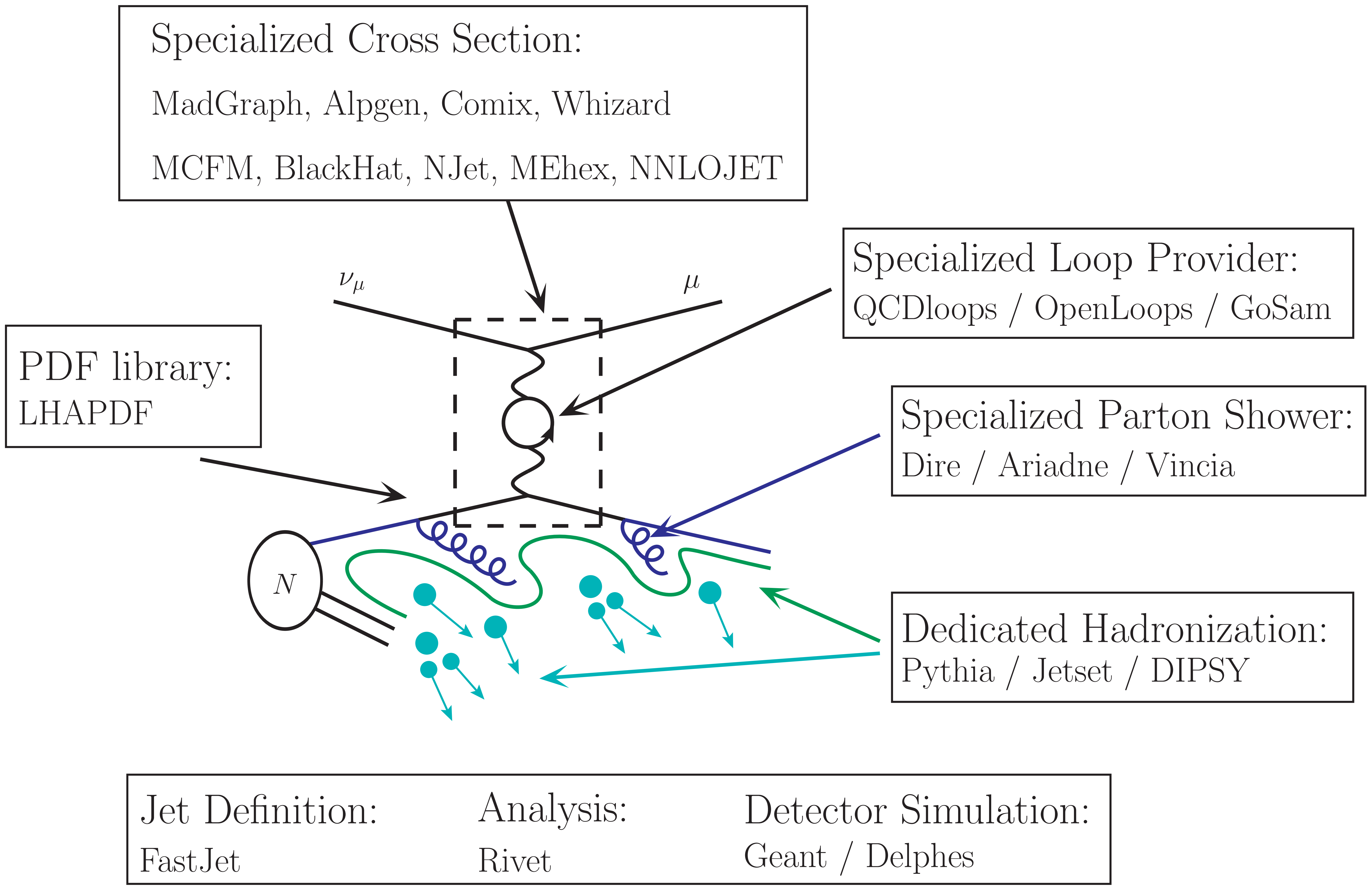
Groups D, E, F

Groups G, H

Groups I, J

Groups K, L, M

Factorized of collider event generation



More factorized – i.e. more codes contributing – where it matters most (precision hard scattering)

S. Prestel

Why do these look different?

- Excuse my diversion into "business theory"...
- Vertically integrated solutions are the right *business model* for new and small markets (usually).
  - Deep engagement with each customer - customization, etc.
  - Customers don't know what they want or what they need - best served by one product that fills an entire "need" (often one they didn't know they had).
- Platforms are the right *business model* for large, mature markets (usually - counterexample: Apple).
  - Large customer base creates a wide variety and large number of needs.
  - Customers very familiar with the product, with strong opinions about it.
  - Pieces of the stack can be commoditized because the market is big enough to support economies of scale.
  - Single provider cannot serve the entire market - to achieve scale, need a full *value chain*.

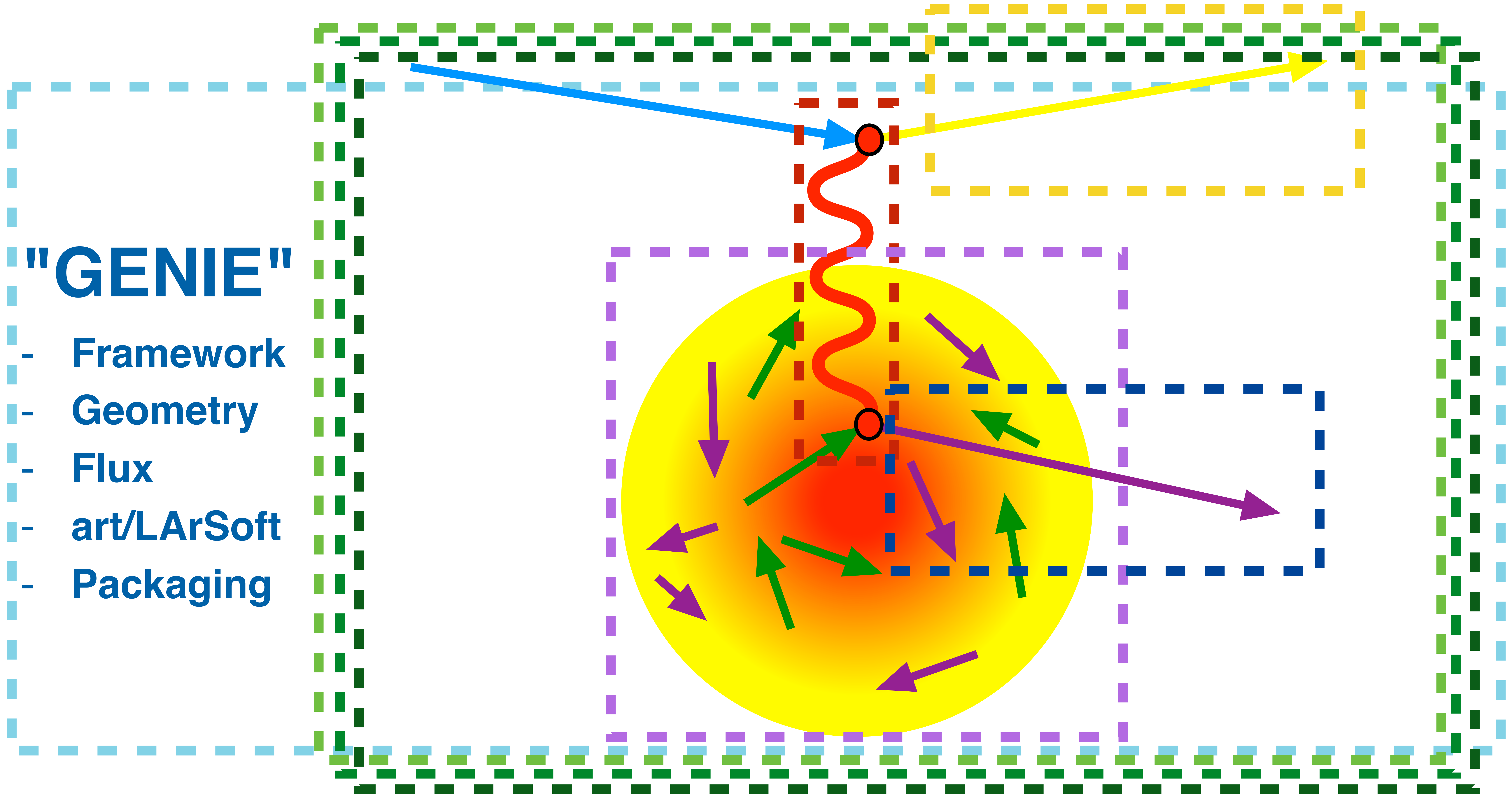
History (many, many examples...)

- Ford ...
  - "You can have any color you want as long as it is black."
  - In-house production of tires, windows, etc.
- -> GM
  - "Any color you like..."
  - Large, diverse supplier chain - whole industries built up around making windshield wipers.
- Amazon ...
  - Sell books in a one-stop shop.
  - Completely solve exactly one problem.
- -> Amazon
  - Hosts a market for retailers to sell anything.
  - Manages fulfillment, logistics, etc. as a service.
  - Web infrastructure for store built out into cloud platform that supports thousands of different businesses.

How I think we want to look...

Fits:

GENIE, Theory Group X, Experiment Y



"GENIE"

- Framework
- Geometry
- Flux
- art/LArSoft
- Packaging

Groups D, E, F

Groups G, H

Groups I, J

Groups K, L, M

What does this mean?

- "Substrate" layer defines interfaces to geometry, flux, physics, I/O, etc.
- Theory-friendly "physics API" allows factorization of the full calculation to a diverse group:
  - any group should be able to offer (and support!) a model for a part of the whole event calculation,
  - groups are allowed to specialize (Ulrich is relieved of responsibility for the detector geometry),
  - subsets of the calculation need not pass through review at the substrate layer - enforce APIs to enforce behavior, suppliers are responsible for their models ("GENIE" gets out of the business of translating Theory Fortran into C++),
  - development model scales to the whole community, with specializations naturally emerging,
  - groups emerge even for combining/fitting the pieces together.
- Obvious downside: assembling an "event" (or some unit of exposure) is much more complex.

What problems are we trying to solve?

- Number one problem is *scale*.
  - Current model worked well for "early" neutrino experiments.
  - If the community is small, say ~O(100) people working, then O(5-10) generator developers/ experts is plenty.
  - If the community is large, say O(1000) people working, then the structure changes. Many contributors (esp. theorists, but also fitters, model-builders, software developers, etc.) are not able to contribute.
- Number two problem is *efficiency*.
  - We should not be spending resources re-implementing the same calculation in many generators, or re-implementing flux drivers, etc., or re-writing unit tests, validation and plotting frameworks, etc.
- Number three problem is *accuracy*.
  - Effectively utilizing currently marginalized resources (esp. theories) makes everything better.

Examples in the spirit of this

- See S. Dolan's and G. Miegas's work implementing SuSA in GENIE.
  - They were able to leverage a reasonably clean API to build a model with minimal guidance.
  - Not every model needs to work the same way (provide hadron tensors, etc.), but many could and many could do conceptually similar things based on structure functions or response functions.
    - "mini-platforms" inside a larger platform...
- GiBUU in NOvA
  - Leverages GENIE flux and geometry driver, and hooks into GENIE event loop structure.
  - Looks very promising, but a bit of hack and took a lot of effort from very technical people.
    - Can we make this easier, and "official"?

Other "intermediate" (possibly final) versions of this

- If we could accurately define a "theory API" that could define a clear specification for the sort of information generators consume and output at various stages in the overall calculation, along with conceptual event record interfaces, theory groups could productively target this API and provide code for their model.
- Generator groups could then either (a) wrap this code and target the inputs/outputs to their own specific architecture (treating the implementation as a black box), or (b) write a new implementation and use the theory code for validation.
- This sort of structure would help make it clear to theory groups how generators use their calculations, and it would reduce the burden of "many implementations of the same calculation in different generators" problem.
    - Defining the theory API is, of course, non-trivial...

(Idea courtesy of Y. Hayato and J. Sobczyk)

View from the LHC…

What do we interface? How do we interface?

	Downstream	*Upstream*
Hard cross section	PDF lib, loop integrals	parton shower
Parton shower	PDF lib, hard cross section	MPI, hadronization
Hadronization	parton shower, MPI	analysis, detector sim.
Analysis	data repo^s, ob^s def. tools	results

Need interfaces that are generic and allow cross-talk. We mostly use

- File-based interfaces
- Run-time interfaces

S. Prestel

Fermilab

How could we hope to achieve this?

- Significant obstacles:
  - Coordination,
  - Resource (money, manpower) management does not necessarily get easier,
  - Organizing collaboration?
- Draw lessons from the Energy Frontier:
  - They went through this process (or so I am told),
  - They had all the same problems we have, but the impetus/needs of the LHC was overriding,
  - First steps:
    - Whitepaper exploring the possibilities? - need a variety of inputs and have to consider everything: funding, technical APIs, physics requirements, interfaces to experiments, general ways to use very different types of calculations (e.g. "one event" in GENIE vs "one event" in GiBUU). Need representation from every stakeholder.
      - *Can the output of this workshop cover some of these needs?*
    - Dedicated workshop dedicated to drafting an API and estimating required labor and financial resources.

What resources are required?

- Proper labor, financial, and time estimates to fulfill the effort require a full, proper analysis (watch out - business jargon again) - a real *project*.
- The initial resources required are simpler but non-trivial:
  - Compose a panel of experts and stakeholders to draft a whitepaper.
  - If successful, **arrange a workshop** to process the issues associated with building a general API.
    - We need a common event record.
    - We need a specification of the API.
    - We need to think about exposure accounting, flux, geometry, framework interfaces, etc.
    - Consider the "theory API".
    - We need to think about organization - **how do the existing generator groups react to all of this?** It is important to have deep involvement from all the generator groups in this. Do we have more than one implementation of the spec? Or just one?
      - *What is the right order of operations here? Do we need a whitepaper to have a workshop? Or vice versa?*

Advantages for generator groups in this model

- Get *out of the business* of *translating* Theorist Fortran.
- Get *out of the business* of *maintaining* the translated models.
- Free up time and effort to focus on the really exciting parts. Get more deeply *into the business of* - fitting global models, engineering validation systems, translating theory calculations into usable models, building tools for systematics or usability (e.g., Python wrappers), actually, you know, *analyzing data*, etc.
- Citations - assuming we can drive and utilize new efforts from the theory community, they have to cite our generator papers.
- This is *not* forcing everyone to use one physics model - it is making it easy to efficiently have a multitude of models. Nobody will have their intellectual freedom constrained. The goal is to free up people to work on their favorite parts. I expect lots of exciting and healthy competition and collaboration.

Conclusions

- Interesting possibility to take a more scalable approach towards generator development.
- History suggests the moment is ripe to change our "business model".
- I've doubtlessly forgotten or overlooked many important aspects - what are they?
- **I propose a whitepaper as the next step - is this sensible? How do we draft this? Do we need to organize a workshop to draft the paper? Or should we try to draft the paper to organize a workshop? Could this workshop's output fill the role of a whitepaper and let us skip to a workshop?**
- I really have no idea what the right way to organize this is.
  - But there are some great organizers here!

Fermilab