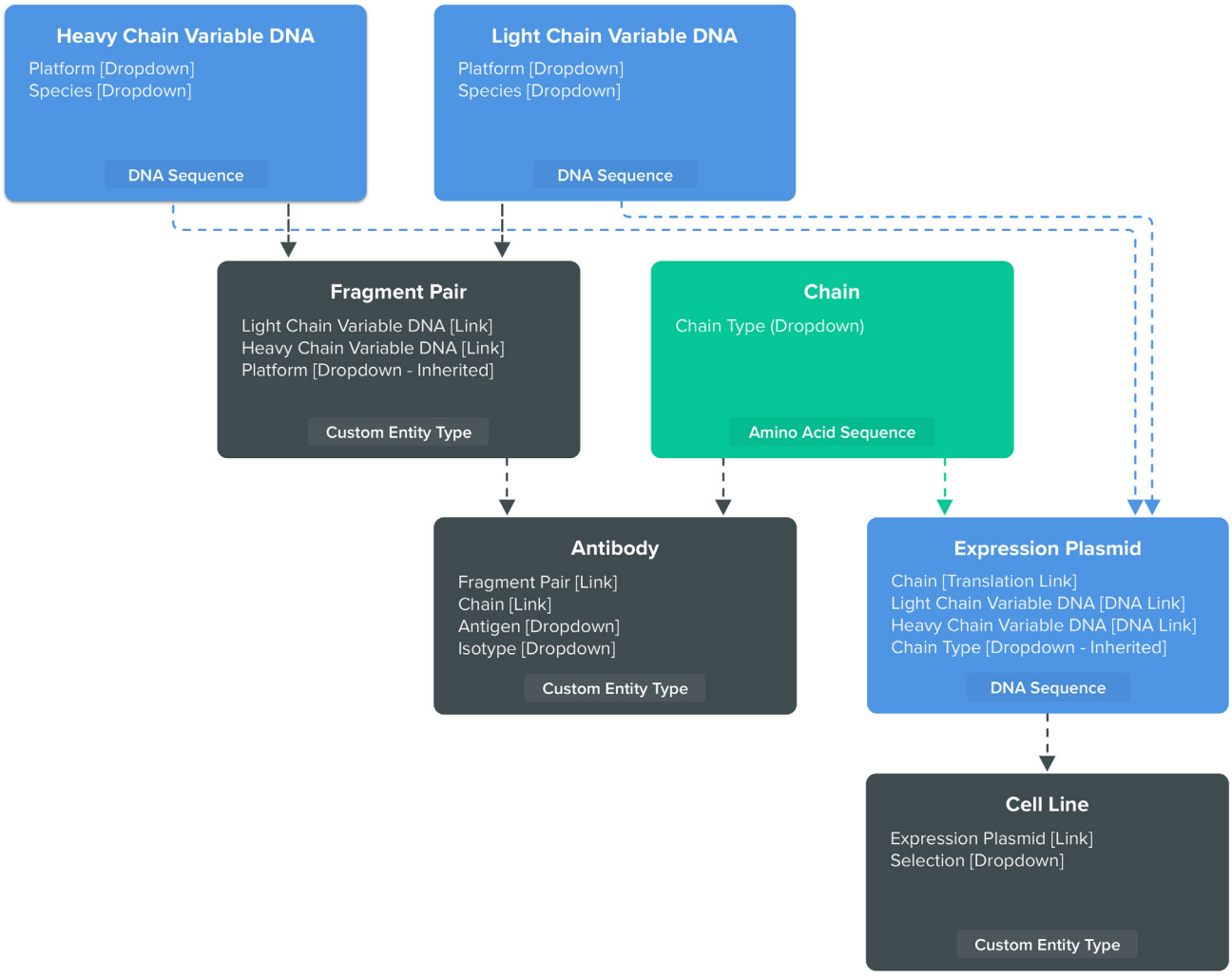
Benchling

How to Configure a Bioregistry for Antibody R&D

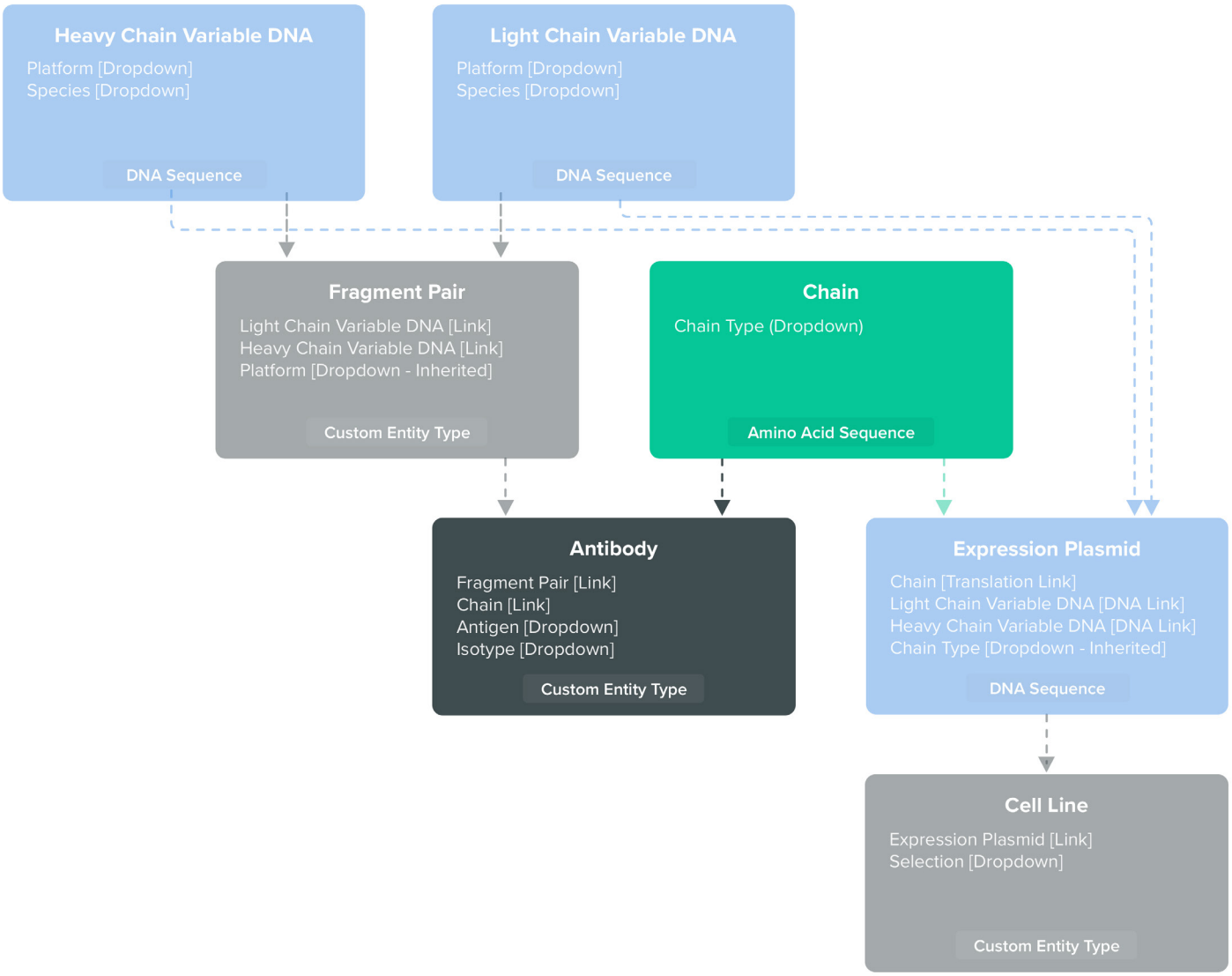
From heavy and light chains, to plasmids and cell lines, antibody discovery involves many moving parts that have to be digitally modeled and interlinked. Without a robust registration system that's configured to track all of these pieces, you end up with data that's inaccurate, redundant, and insufficient. But if you configure your Bioregistration system properly, you can answer essentially any question about your large molecules. With these answers at your fingertips, you can drive research decisions with data and results that are reliable and comprehensive.

Coming up with a final data model for your antibody discovery ultimately depends on your particular research needs. Here's how, illustrated in Benchling's point-and-click interface, you can configure a **foundational Bioregistry for Antibody Discovery.**

TABLE OF CONTENTS

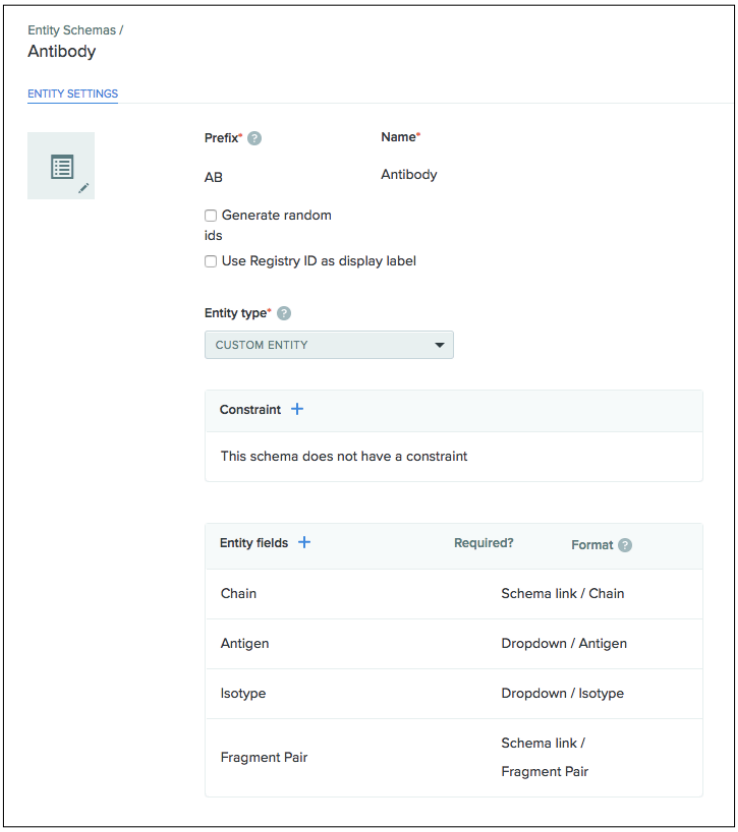


The Basics:
Antibodies and Chains



If you're configuring a Bioregistry for antibody discovery, then of course you'll need to track antibodies. The first question to ask yourself is: What do you want to track on your **Antibody** entity? By thoroughly addressing this single question, we can develop a complete network of entities and fields that we want to track.

Modeling the Antibody Schema



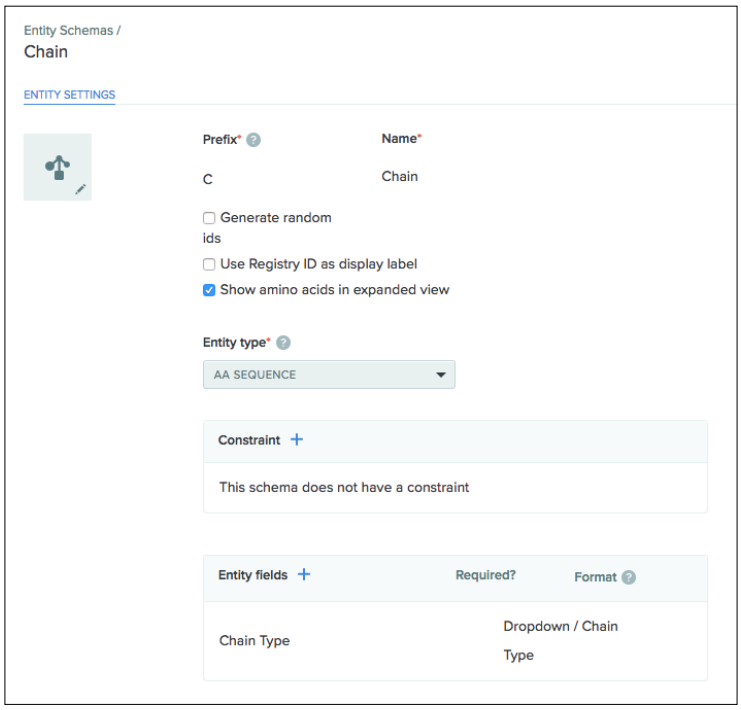
For starters, we obviously want to track the Antigen that each **Antibody** targets, so we add that as a field in our **Antibody** schema. We recommend giving this field a “dropdown” data type to enforce specific antigen values and naming conventions (ex. CD19, VEGF).

Next, every antibody needs an Isotype. This can also be a dropdown field (ex. IgG, IgM).

Lastly, every monoclonal antibody needs a Light Chain and a Heavy Chain. Unlike the Antigen and Isotype fields that we just configured, we want to track additional information about **Chains**, such as their amino acid sequences.

So rather than just make **Chains** a field on the Antibody entity, we’ll make them full-fledged entities of their own.

Modeling the Chain Schema and linking it to the Antibody Schema

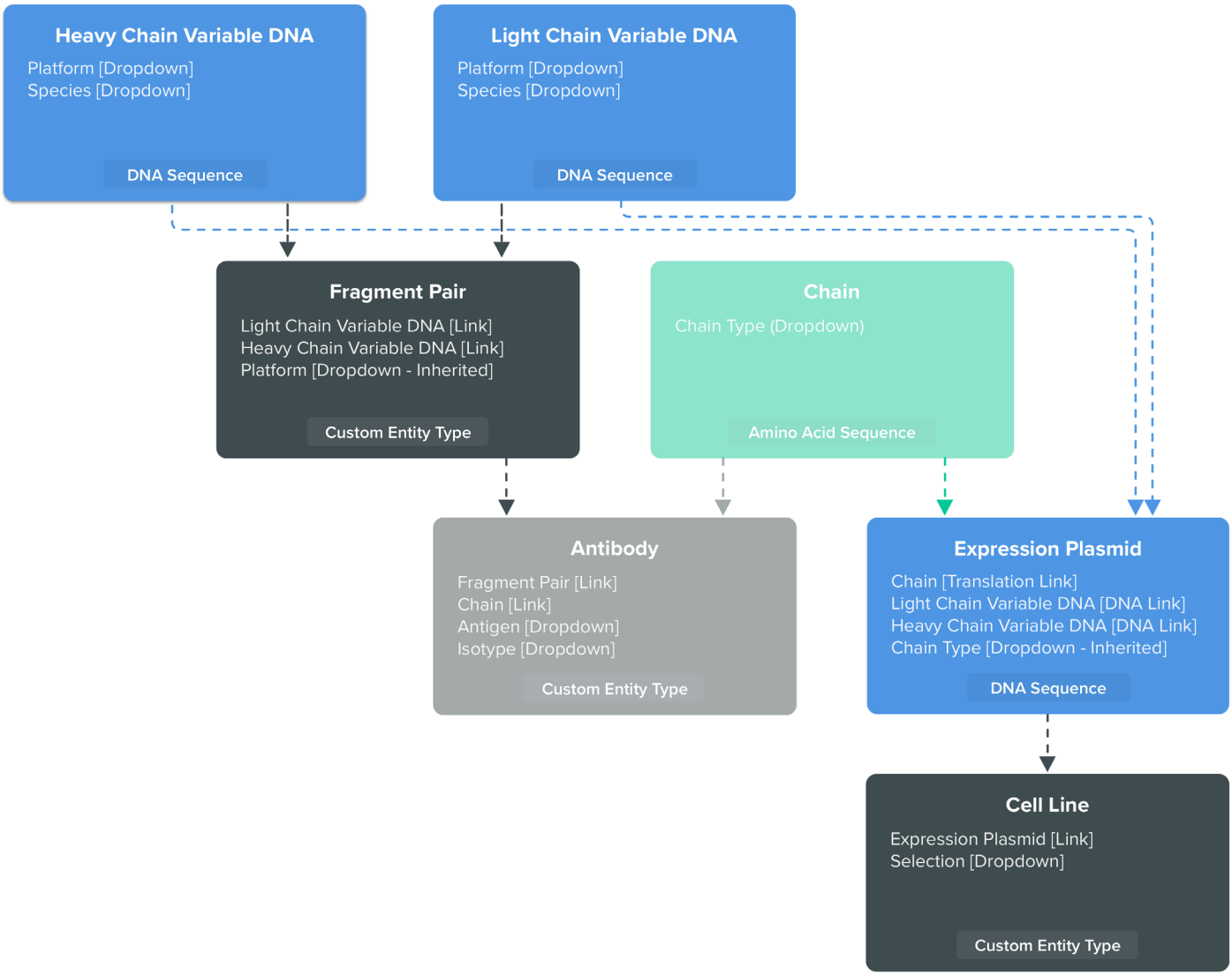


What do we want to track on a **Chain** entity? We've already partially answered this question: its amino acid sequence. In Benchling, we do this by declaring the **Chain** entity to be an "Amino Acid"-type entity. This means that **Chains** are amino acid sequences (the type that you can view in Benchling's Molecular Biology Tools) that also have additional data fields that we can define.

In this case, other than the amino acid sequence, all we want to know is whether a **Chain** is a Light Chain or a Heavy Chain, so we'll create a dropdown field on **Chains** called "Chain Type", with those two options.

Now that we've defined our **Chain** entity, we can go back to the **Antibody** entity. We'll create a new field on our **Antibodies** called "Chain". Unlike the other fields that we've created so far (Antigen, Isotype, and Chain Type), all of which were dropdowns, we'll define this "Chain" field as a link to the **Chain** entity that we just defined. This is our first interlinkage between two entities – and as you'll see, it's the first of many.

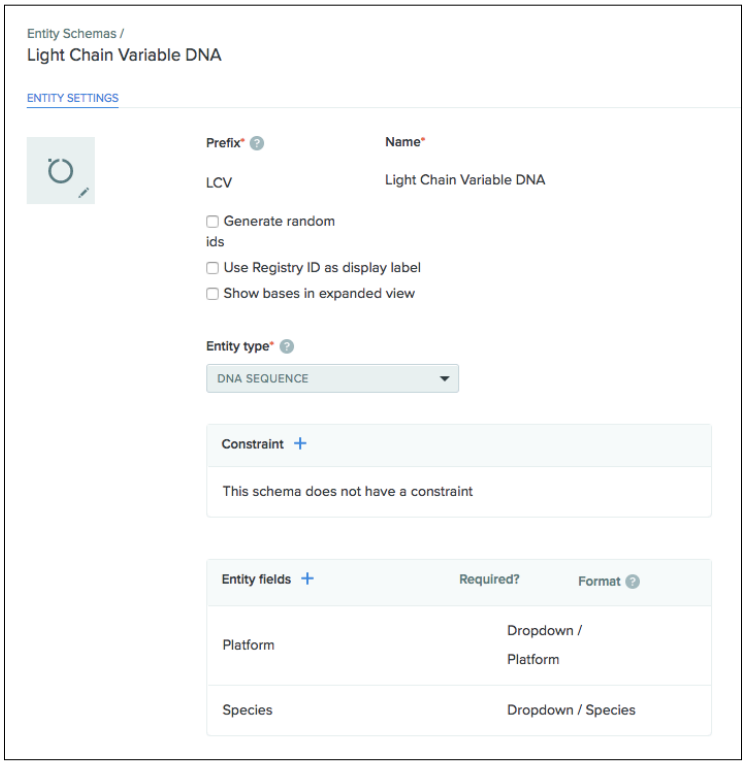
DNA Sequences:
Chain Variables, Fragment Pairs, Plasmids, and Cell Lines



Having modeled **Antibodies** and **Chains**, we can take a step back and drill down into the upstream DNA entities that connect to them: **Heavy and Light Chain Variables** and **Fragment Pairs**.

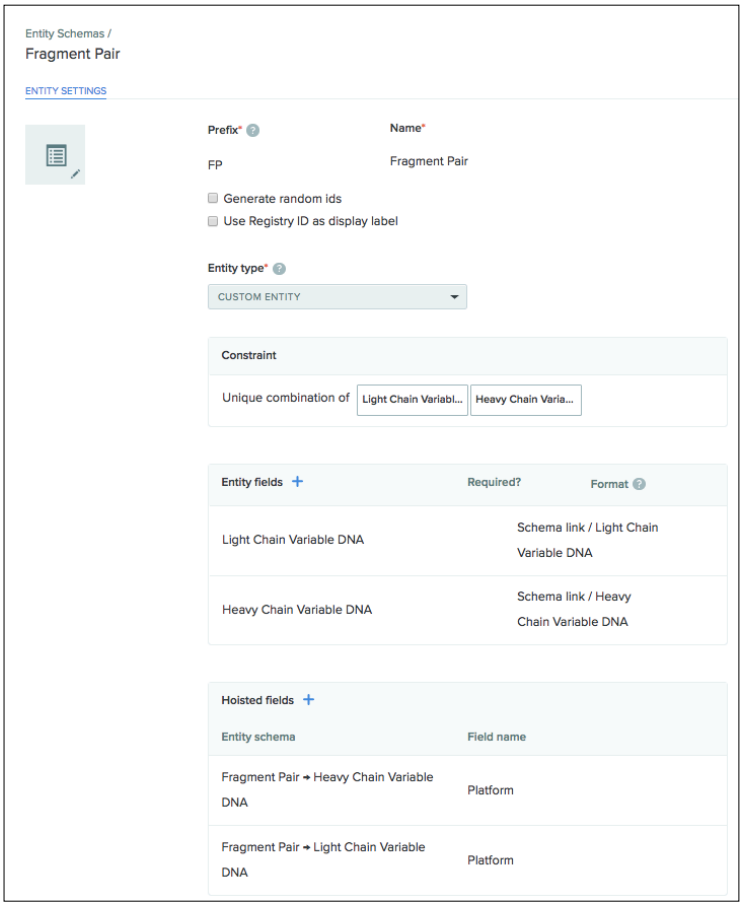
After that, we'll model our two most downstream entities: **Expression Plasmids** and **Cell Lines**.

Modeling the Light and Heavy Chain Variable DNA Schemas



Our most upstream entities are **Light Chain Variable DNA** and **Heavy Chain Variable DNA**. Similarly to how we wanted **Chains** to have amino acid sequences, we clearly want these **Variable DNA** entities to have DNA sequences. Thus, we'll define them as "DNA"-type entities. This means that they have DNA sequences (again, viewable in the Molecular Biology Tools) and additional data fields that we can define. We'll also give **Light Chain Variable DNA** and **Heavy Chain Variable DNA** dropdown fields for "Platform" (ex. phage, hybridoma) and "Species" (ex. human, mouse).

Modeling the Fragment Pair Schema and Linking it to the Variable Chain DNA Schemas, as well as to the Antibody Schema



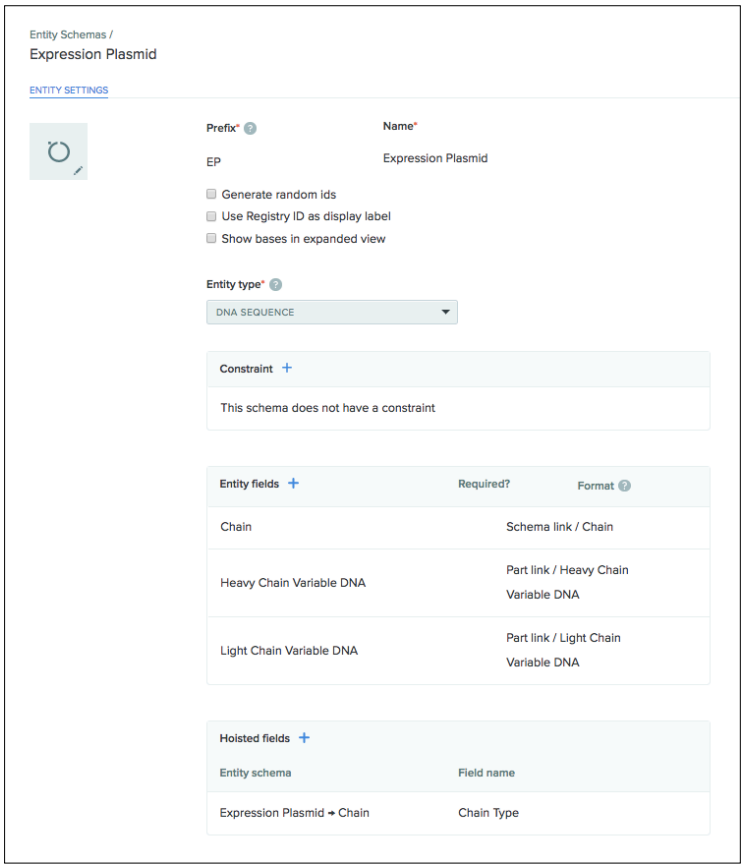
These variable DNA sequences alone, however, would tell scientists nothing about how the light and heavy chains actually pair. To do this, we'll create a new entity called **Fragment Pair**. Just like how we gave **Antibody** entities a link to **Chain** entities, we'll give **Fragment Pair** entities a link to the **Light Chain Variable DNA** and **Heavy Chain Variable DNA** entities.

But we don't want to make **Fragment Pairs** out of just any pair of **Light and Heavy Chain Variables**. For example, we don't want two **Fragment Pairs** to have the same pair of **Light and Heavy Chain Variables**, because that would create duplicated data. This is where Benchling's data constraints come in handy. We can structure a rule that says, "Only allow a **Fragment Pair** to be registered if it has a unique pair of **Light and Heavy Chain Variables**."

We'll also add a dropdown field for "Platform" to the **Fragment Pair** entity. Benchling will automatically populate this **Fragment Pair** "Platform" field based on the "Platform" values of the linked **Light Chain and Heavy Chain Variables**.

As a final step, we'll go back to our **Antibody** entity and link it to the **Fragment Pair** entity, so we can see the pairings for every antibody.

Modeling the Expression Plasmid Schema and Linking it to the Variable Chain DNA Schemas and the Chain Schema



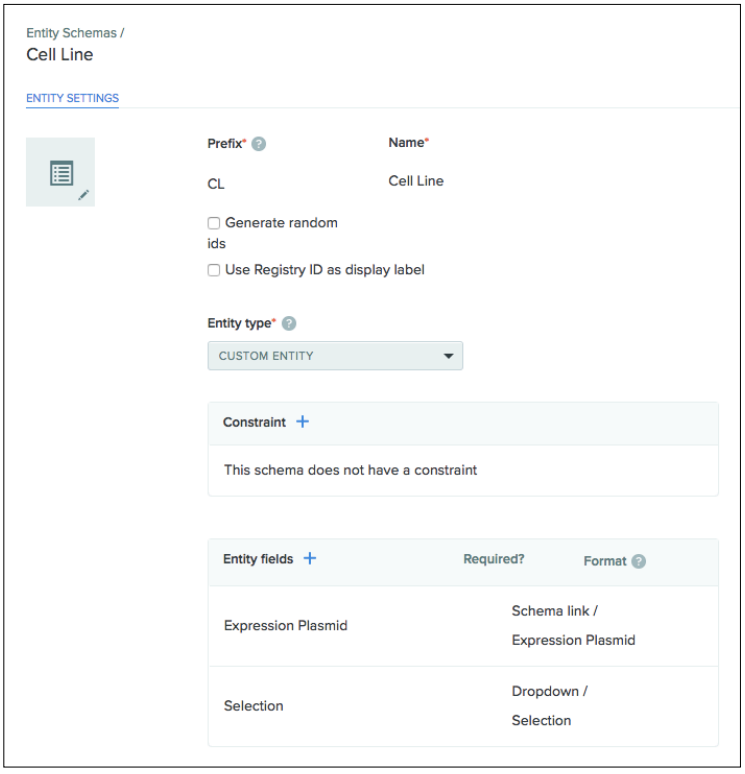
Having modeled **Chains**, **Variables**, and **Fragment Pairs**, we can move on to a more daunting entity that brings them all together: the **Expression Plasmid**. Like **Light Chain and Heavy Chain Variables**, the **Expression Plasmid** will be a “DNA”-type entity, which means it has a DNA sequence. As you can probably guess, the **Expression Plasmid** will encode a **Light Chain Variable** and a **Heavy Chain Variable**. So, we’ll link from the **Expression Plasmid** to one **Light Chain Variable DNA** entity and one **Heavy Chain Variable DNA** entity.

However, this won’t be a plain linkage. Because the **Expression Plasmid** is a “DNA”-type, and because the **Light and Heavy Chain Variables** are also “DNA”-types, Benchling can automatically detect when **Chain Variable** sequences are encoded within the **Plasmid‘s** sequence. In other words, when you click “Register” on an **Expression Plasmid** sequence, Benchling will automatically check, “Out of all the previously registered **Light and Heavy Chain Variables**, do any of their DNA sequences exist within this **Expression Plasmid** sequence?” If they do exist, Benchling will automatically identify them and link them to your newly-registered **Expression Plasmid**, taking the burden off the user.

Benchling can also take this biologically-aware linking a step further. On **Expression Plasmids**, it also makes sense to link a **Chain** entity, so that scientists can determine which amino acid sequences are encoded in the plasmid. (As a reminder, we modeled the **Chain** entity as an “Amino Acid”-type, so it has a sequence associated with it.) This link will be a “Translation Link”, which means that when you link a **Chain** on an **Expression Plasmid**, Benchling will check, “Is there a region of this **Expression Plasmid** sequence that translates to the amino acid of the **Chain**?” The registration will succeed only if there is a valid translation. This ensures that you don’t accidentally link a chain to the wrong plasmid.

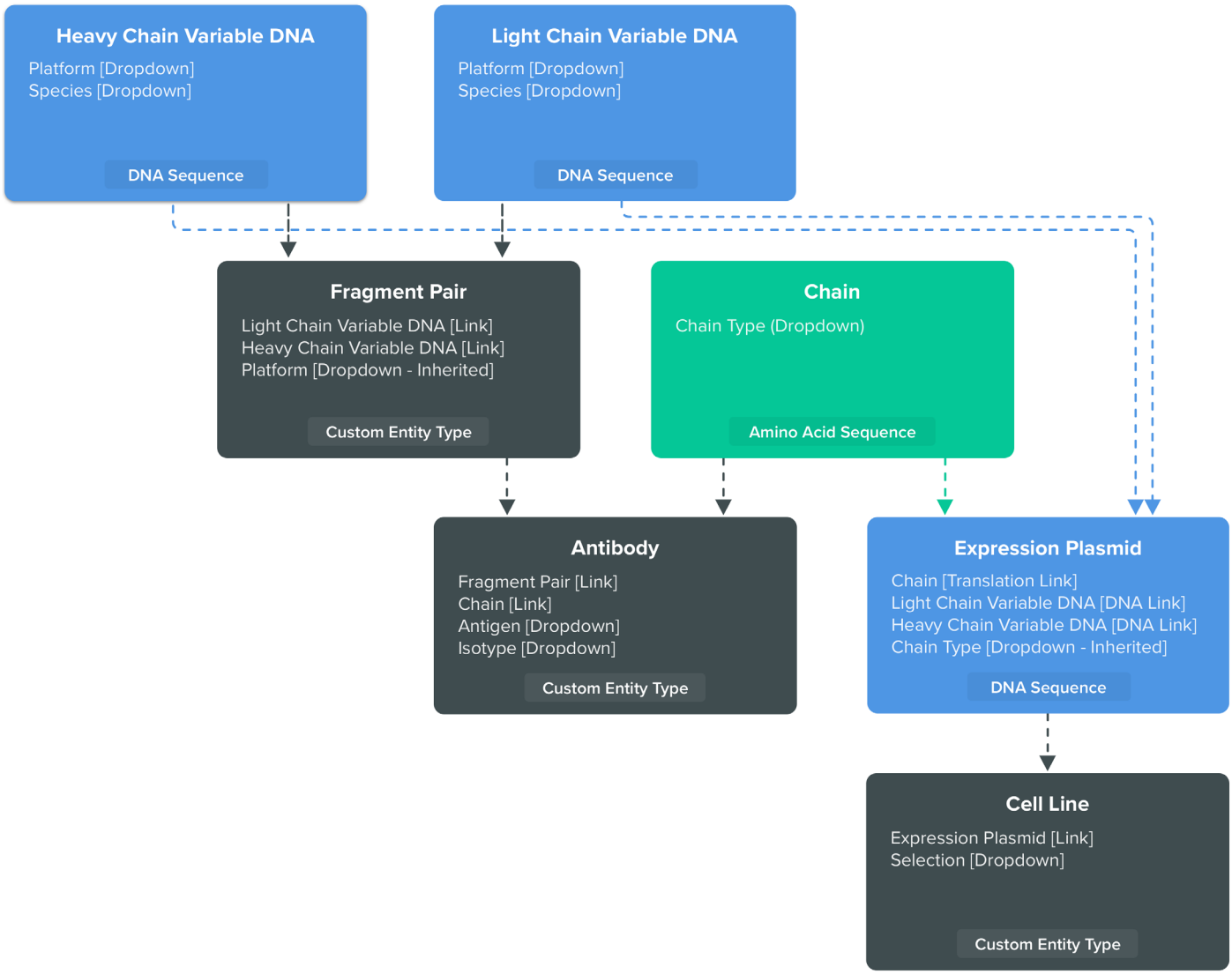
We can also structure a “Chain Type” field on the **Expression Plasmid** that’s automatically inherited from the linked **Chain**, which will make it easier to search for plasmids later on.

Modeling the Cell Line Schema and Linking it to the Expression Plasmid Schema



Having modeled **Antibodies**, **Chains**, **Light and Heavy Chain Variables**, **Fragment Pairs**, and **Expression Plasmids**, only one foundational entity remains to be modeled: the **Cell Line**. Unlike the complex amino acids and DNA sequences that we just modeled, the **Cell Line** will consist of only three data fields: a “Selection” dropdown (ex. G418, hygro), a “Type” dropdown (ex. working, master), and a plain link to an **Expression Plasmid**.

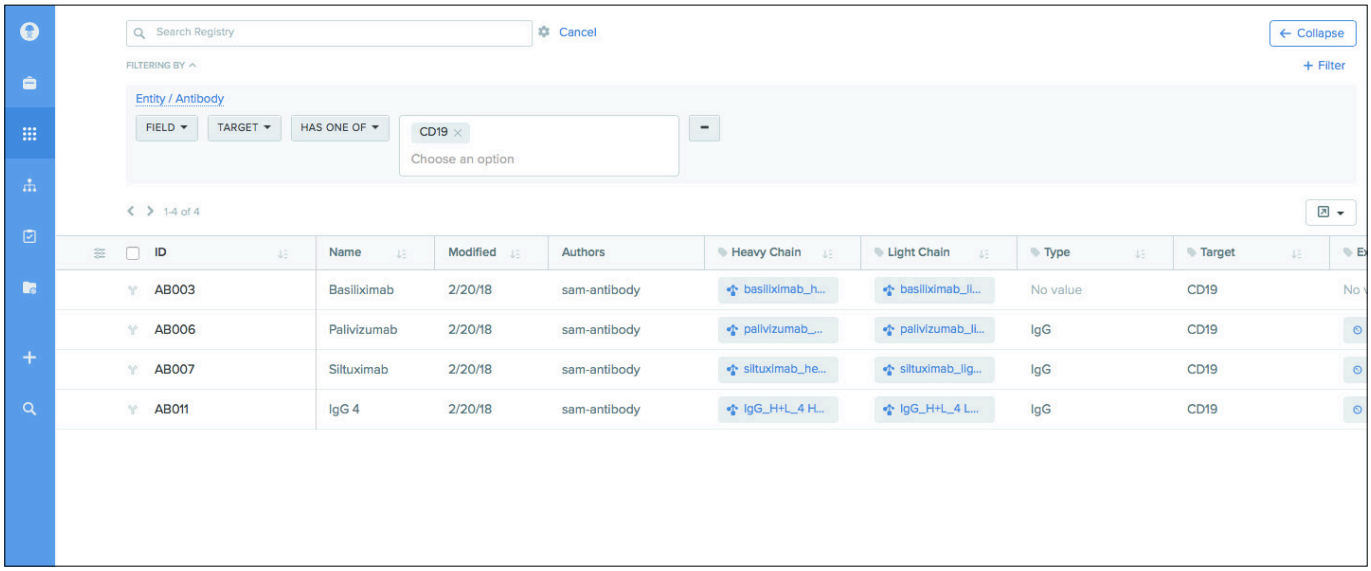
The Big Picture:
Answer Any Question About Your Large Molecules



In short order, we've developed a complex, interlinked registration system for antibody discovery. Needless to say, there are numerous additional fields that you could be tracking on these entities; for example, you could add "Fc Alteration" or "Type" fields to your **Chain** entities.

You could also structure new data constraints based on the "Isotype" field of an **Antibody** entity; for example, you could require that an **Antibody** with an "Isotype" of value "Bispecific" have links to four different **Chains**.

By Tracking Biological Entities and Functional Data, You Can Make Better R&D Decisions



Now that we've created this data model, you're probably wondering what you can do with it. Well, for one, you can run structured queries on any of the entities and fields that we just defined. Find every antibody with one particular target. Or, find every plasmid that encodes a particular chain, and then the relevant cell lines.

As you can see in our final network diagram, the entities that we've modeled are all interconnected. As a result, you can trace a path from the most downstream entity (a cell line) to the most upstream entities (light and heavy chain variables), or to any entities in between. The lineage of any entity is accessible, searchable, and reliable.

Beyond running these queries, Benchling gives you access to all of the functional data produced by your entities. You can trace down from one of your **Antibody** entities to every physical container of it that has ever been produced. You can then answer questions such as, "Which antibody that targets a particular antigen has the highest average binding affinity?"

The flexibility of Benchling's Bioregistry empowers you to do all of this and more, all in a point-and-click interface. Thanks to this flexibility, your Bioregistry can always map perfectly to your large molecules, while Benchling's biologically-aware features ensure that your data is always accurate. With a reliable and complete picture of your large molecules, you can harness every aspect of their multi-dimensional data – from innate characteristics, to decision-quality results.

Ready to learn how Benchling maps to your R&D?
Get a demo at benchling.com

415-980-9932 | contact@benchling.com
555 Montgomery Street, Suite 1100, San Francisco, CA 94111