ND Genotypes Documentation

Release 7.x-2.0-beta1

Lacey-Anne Sanderson et al., University of Saskatchewan, Pulse I

May 27, 2022

Resources:

This module provides support and visualization of genotypic data stored in a modified GMOD Chado schema. The 3.x branch of this module represents a shift towards support for large scale genotypic datasets through backwards compatible improvements to the Chado schema including a new gathering table for genotypes (genotype_call) modelled after the chado phenotype table, optimized queries and well-choosen indexes.

Note: Easy Data loading is available via the Genotypes Loader which supports VCF files!

CHAPTER 1

Features

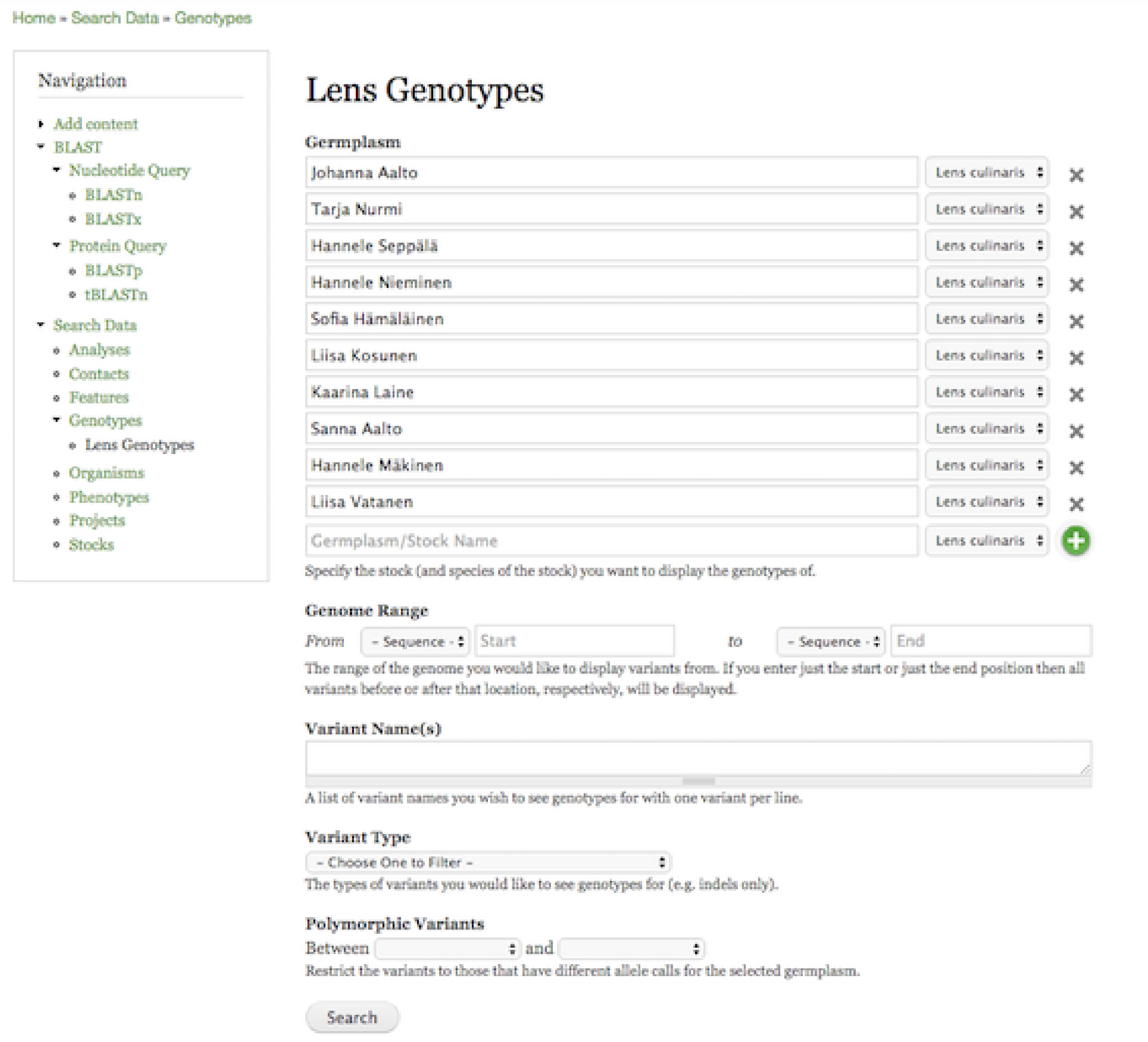
- Extensive configuration allowing for flexiblity in ontology terms used, as well as, colours and wording used in visualizations.
- **Multiple Tripal 3 Fields which provide flexible, configurable summaries of genotypic data.**
  - Marker/Variant Genotype Summary: a pie chart showing the ratio of alleles recorded per marker.
  - Marker/Variant Flanking Sequence: a FASTA record showing flanking sequence with all known variants indicated via IUPAC codes (useful in marker design).
  - Marker List: provides links to the markers assaying a given variant.
  - **Genotype Matrix Quick Link: provides a quick link to a pre-filtered genotype matrix. How it is filtered is depen**
    - On Marker/Variant pages: restricted to specific variant
    - On Germplasm pages: germplasm is pre-selected
    - On Project pages: project is pre-selected if genus is a property of the project.
- Genotype Matrix search allowing users to extract genotypes for a user-defined set of germplasm. Includes filtering by marker/variant type, variant location, and pairwise polymorphism. Filtering by quality is coming soon.
- Integration of all fields with Tripal 3 web services allowing you to share your genotypic data with other groups.

Note: If ND Genotypes fields are not automatically attached to the genetic marker and sequence variant content types, go to the "Manage Fields" page for each and click "Find new fields". Also, go to the "Manage Display" page and ensure they are not hidden.

1.1 Genotype Matrix

This module provides genotype search functionality that allows users to select which germplasm and variants they are interested in and be shown a colour-coded variant by germplasm table which can be further filtered by marker/variant type and to only show polymorphic variants (pairwise comparison choosen by the user). After filtering to their desired dataset, the user can download the table as a tab-delimited file.

As you can see in the following screenshot, the user can enter any number of germplasm depending upon their needs. Additionally, the filter criteria is well-defined including helpful descriptions under each one.



Lens Genotypes

Germplasm

Johanna Aalto	Lens culinaris
Tarja Nurmi	Lens culinaris
Hannele Seppälä	Lens culinaris
Hannele Nieminen	Lens culinaris
Sofia Hämäläinen	Lens culinaris
Liisa Kosunen	Lens culinaris
Kaarina Laine	Lens culinaris
Sanna Aalto	Lens culinaris
Hannele Mäkinen	Lens culinaris
Liisa Vatanen	Lens culinaris
Germplasm/Stock Name	Lens culinaris

Specify the stock (and species of the stock) you want to display the genotypes of.

Genome Range

From - Sequence - Start to - Sequence - End

The range of the genome you would like to display variants from. If you enter just the start or just the end position then all variants before or after that location, respectively, will be displayed.

Variant Name(s)

A list of variant names you wish to see genotypes for with one variant per line.

Variant Type

- Choose One to Filter -

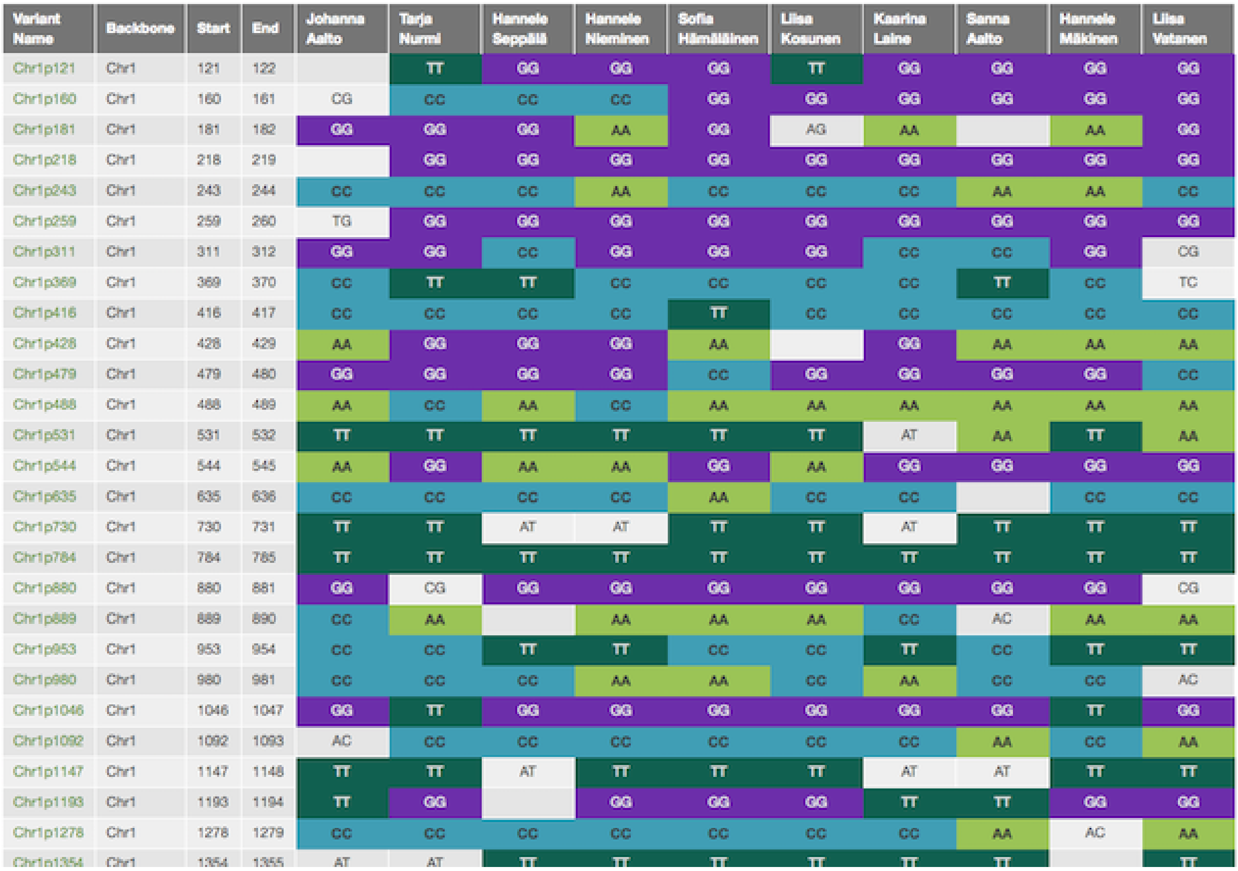
The types of variants you would like to see genotypes for (e.g. indels only).

Polymorphic Variants

Between and

Restrict the variants to those that have different allele calls for the selected germplasm.

This is the matrix resulting from the above filter criteria. As you can see, each column represents one of the chosen germplasm and each row represents a specific variant.



Variant Name	Backbone	Start	End	Johanna Aalto	Tarja Nurmi	Hannele Seppälä	Hannele Nieminen	Sofia Hämäläinen	Liisa Kosunen	Kaarina Laine	Sanna Aalto	Hannele Mäkinen	Liisa Vatanen
Chr1p121	Chr1	121	122		TT	GG	GG	GG	TT	GG	GG	GG	GG
Chr1p160	Chr1	160	161	CG	CC	CC	CC	GG	GG	GG	GG	GG	GG
Chr1p181	Chr1	181	182	GG	GG	GG	AA	GG	AG	AA		AA	GG
Chr1p218	Chr1	218	219		GG	GG	GG	GG	GG	GG	GG	GG	GG
Chr1p243	Chr1	243	244	CC	CC	CC	AA	CC	CC	CC	AA	AA	CC
Chr1p259	Chr1	259	260	TG	GG	GG	GG	GG	GG	GG	GG	GG	GG
Chr1p311	Chr1	311	312	GG	GG	CC	GG	GG	GG	CC	CC	GG	CG
Chr1p369	Chr1	369	370	CC	TT	TT	CC	CC	CC	CC	TT	CC	TC
Chr1p416	Chr1	416	417	CC	CC	CC	CC	TT	CC	CC	CC	CC	CC
Chr1p428	Chr1	428	429	AA	GG	GG	GG	AA		GG	AA	AA	AA
Chr1p479	Chr1	479	480	GG	GG	GG	GG	CC	GG	GG	GG	GG	CC
Chr1p488	Chr1	488	489	AA	CC	AA	CC	AA	AA	AA	AA	AA	AA
Chr1p531	Chr1	531	532	TT	TT	TT	TT	TT	TT	AT	AA	TT	AA
Chr1p544	Chr1	544	545	AA	GG	AA	AA	GG	AA	GG	GG	GG	GG
Chr1p635	Chr1	635	636	CC	CC	CC	CC	AA	CC	CC		CC	CC
Chr1p730	Chr1	730	731	TT	TT	AT	AT	TT	TT	AT	TT	TT	TT
Chr1p784	Chr1	784	785	TT	TT	TT	TT	TT	TT	TT	TT	TT	TT
Chr1p880	Chr1	880	881	GG	CG	GG	GG	GG	GG	GG	GG	GG	CG
Chr1p889	Chr1	889	890	CC	AA		AA	AA	AA	CC	AC	AA	AA
Chr1p953	Chr1	953	954	CC	CC	TT	TT	CC	CC	TT	CC	TT	TT
Chr1p980	Chr1	980	981	CC	CC	CC	AA	AA	CC	AA	CC	CC	AC
Chr1p1046	Chr1	1046	1047	GG	TT	GG	GG	GG	GG	GG	GG	TT	GG
Chr1p1092	Chr1	1092	1093	AC	CC	CC	CC	CC	CC	CC	AA	CC	AA
Chr1p1147	Chr1	1147	1148	TT	TT	AT	TT	TT	TT	AT	AT	TT	TT
Chr1p1193	Chr1	1193	1194	TT	GG		GG	GG	GG	TT	TT	GG	GG
Chr1p1278	Chr1	1278	1279	CC	CC	CC	CC	CC	CC	CC	AA	AC	AA
Chr1p1354	Chr1	1354	1355	AT	AT	TT	TT	TT	TT	TT	TT		TT

1.2 Marker/Variant Genotype Summary Fields

This field adds a summary pie chart figure to marker or variant pages. It shows the ratio of alleles saved for the given marker/variant and can be used to give the researcher an idea of what alleles to expect when using the marker, as well as, how rare a given result might be.

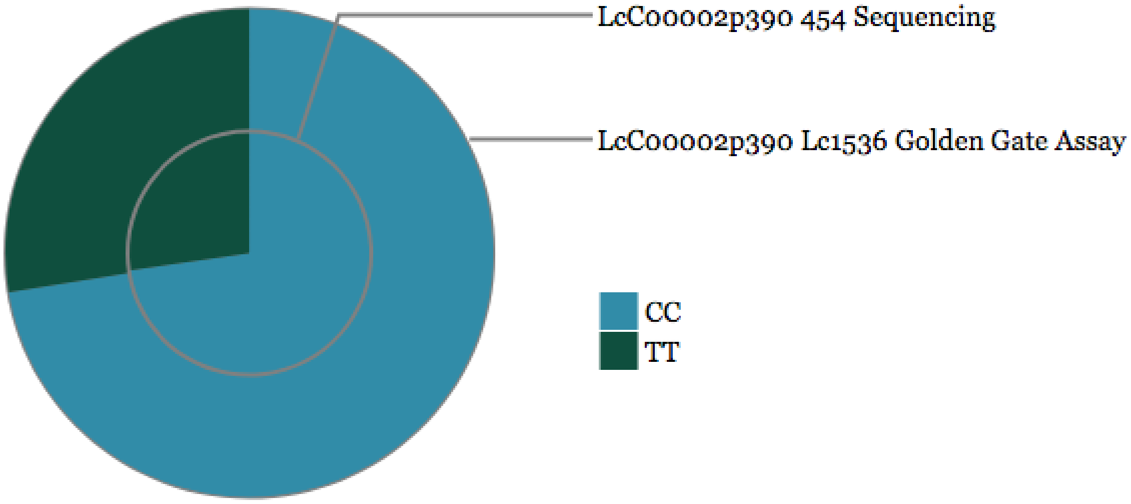
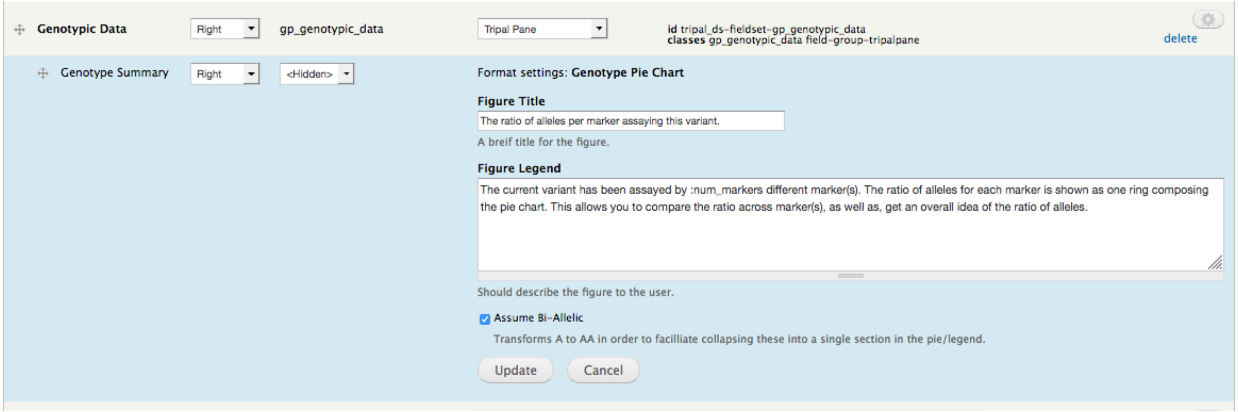


Figure: The ratio of alleles per marker assaying this variant. The current variant has been assayed by 2 different marker(s). The ratio of alleles for each marker is shown as one ring composing the pie chart. This allows you to compare the ratio across marker(s), as well as, get an overall idea of the ratio of alleles.

Both the title and description of the figure legend can be configured by going to Administration » Structure » Tripal Content Types » [Variant/Marker] » Manage Display and clicking on the gear beside the genotype summary field.



Genotypic Data | Right | gp_genotypic_data | Tripal Pane | **id** tripal_ds-fieldset-gp_genotypic_data **classes** gp_genotypic_data field-group-tripalpane | delete

Genotype Summary | Right | <Hidden>

Format settings: **Genotype Pie Chart**

Figure Title

The ratio of alleles per marker assaying this variant.

A breif title for the figure.

Figure Legend

The current variant has been assayed by :num_markers different marker(s). The ratio of alleles for each marker is shown as one ring composing the pie chart. This allows you to compare the ratio across marker(s), as well as, get an overall idea of the ratio of alleles.

Should describe the figure to the user.

☑ Assume Bi-Allelic

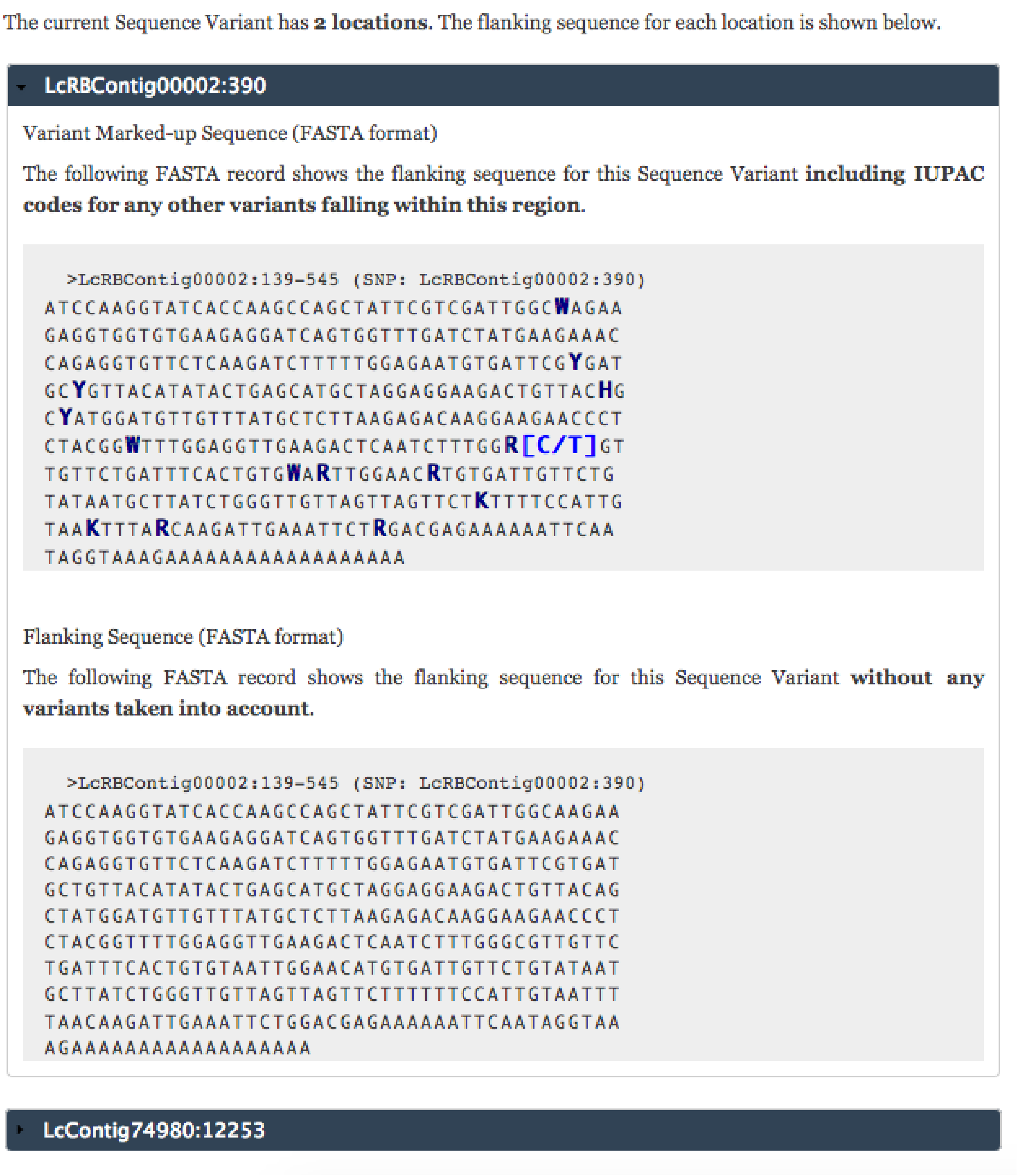
Transforms A to AA in order to facilliate collapsing these into a single section in the pie/legend.

Update | Cancel

Warning: Make sure to click "Update" in the blue settings pane; as well as, "Save" at the bottom of the page.

1.3 Marker/Variant Flanking Sequence Field

This field adds a FASTA record showing the flanking sequence for the current marker variant. It also highlights the variants in the flanking region with their IUPAC codes. This field seamlessly handles variants with multiple locations by showing each one with the first one by rank expanded.



The current Sequence Variant has **2 locations**. The flanking sequence for each location is shown below.

Variant Marked-up Sequence (FASTA format)

The following FASTA record shows the flanking sequence for this Sequence Variant **including IUPAC codes for any other variants falling within this region.**

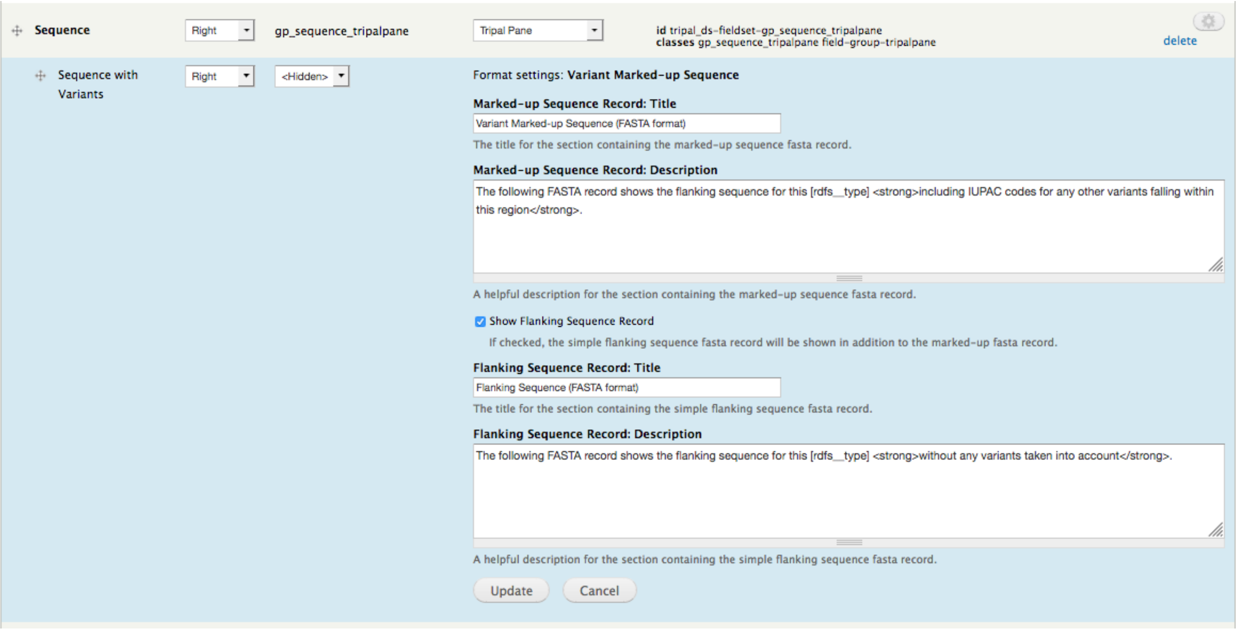
```
  >LcRBContig00002:139-545 (SNP: LcRBContig00002:390)
ATCCAAGGTATCACCAAGCCAGCTATTCGTCGATTGGCWAGAA
GAGGTGGTGTGAAGAGGATCAGTGGTTTGATCTATGAAGAAAC
CAGAGGTGTTCTCAAGATCTTTTTGGAGAATGTGATTCGYGAT
GCYGTTACATATACTGAGCATGCTAGGAGGAAGACTGTTACHG
CYATGGATGTTGTTTATGCTCTTAAGAGACAAGGAAGAACCCT
CTACGGWTTTGGAGGTTGAAGACTCAATCTTTGGR[C/T]GT
TGTTCTGATTTCACTGTGWARTTGGAACRTGTGATTGTTCTG
TATAATGCTTATCTGGGTTGTTAGTTAGTTCTKTTTTCCATTG
TAAKTTTARCAAGATTGAAATTCTRGACGAGAAAAAATTCAA
TAGGTAAAGAAAAAAAAAAAAAAAAAA
```

Flanking Sequence (FASTA format)

The following FASTA record shows the flanking sequence for this Sequence Variant **without any variants taken into account.**

```
  >LcRBContig00002:139-545 (SNP: LcRBContig00002:390)
ATCCAAGGTATCACCAAGCCAGCTATTCGTCGATTGGCAAGAA
GAGGTGGTGTGAAGAGGATCAGTGGTTTGATCTATGAAGAAAC
CAGAGGTGTTCTCAAGATCTTTTTGGAGAATGTGATTCGTGAT
GCTGTTACATATACTGAGCATGCTAGGAGGAAGACTGTTACAG
CTATGGATGTTGTTTATGCTCTTAAGAGACAAGGAAGAACCCT
CTACGGTTTTGGAGGTTGAAGACTCAATCTTTGGGCGTTGTTC
TGATTTCACTGTGTAATTGGAACATGTGATTGTTCTGTATAAT
GCTTATCTGGGTTGTTAGTTAGTTCTTTTTTCCATTGTAATTT
TAACAAGATTGAAATTCTGGACGAGAAAAAATTCAATAGGTAA
AGAAAAAAAAAAAAAAAAAA
```

Both the title and description of the figure legend can be configured by going to Administration » Structure » Tripal Content Types » [Variant/Marker] » Manage Display and clicking on the gear beside the genotype summary field.



Sequence | Right | gp_sequence_tripalpane | Tripal Pane | id tripal_ds-fieldset-gp_sequence_tripalpane
classes gp_sequence_tripalpane field-group-tripalpane | delete

Sequence with Variants | Right | <Hidden>

Format settings: **Variant Marked-up Sequence**

Marked-up Sequence Record: Title

Variant Marked-up Sequence (FASTA format)

The title for the section containing the marked-up sequence fasta record.

Marked-up Sequence Record: Description

The following FASTA record shows the flanking sequence for this [rdfs__type] <strong>including IUPAC codes for any other variants falling within this region</strong>.

A helpful description for the section containing the marked-up sequence fasta record.

☑ Show Flanking Sequence Record

If checked, the simple flanking sequence fasta record will be shown in addition to the marked-up fasta record.

Flanking Sequence Record: Title

Flanking Sequence (FASTA format)

The title for the section containing the simple flanking sequence fasta record.

Flanking Sequence Record: Description

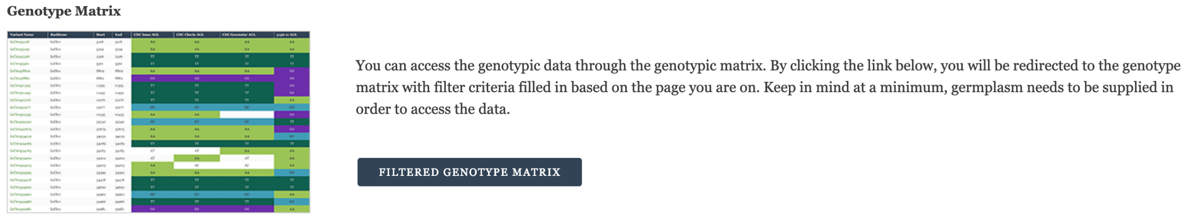
The following FASTA record shows the flanking sequence for this [rdfs__type] <strong>without any variants taken into account</strong>.

A helpful description for the section containing the simple flanking sequence fasta record.

Warning: Make sure to click "Update" in the blue settings pane; as well as, "Save" at the bottom of the page.

1.4 Genotype Matrix Quick Link

This field provides a quick link to the genotype matrix from project, germplasm, marker and variant pages. It pre-filters the genotype matrix to data relating to the page it's on. For example, on a germplasm page (any content type storing data in the Chado stock table) the user will be taken to a genotype matrix of the correct genus already displaying genotypes for the germplasm they were looking at.



Genotype Matrix

You can access the genotypic data through the genotypic matrix. By clicking the link below, you will be redirected to the genotype matrix with filter criteria filled in based on the page you are on. Keep in mind at a minimum, germplasm needs to be supplied in order to access the data.

The link is consistent across content types and does not need to be configured. It automatically detects the type of content it is on and adds information to link to pre-filter the genotype matrix accordingly.

1.4.1 Project Pages

Project pages are any Tripal Content which stores it's base data in the Chado project table including "Study", "Genome Project" and "Project" default Tripal Content Types. The genus is determined based on a Chado property with cvterm TAXRANK:genus and the genotype matrix link with simply not appear on content without this property. The unique project identifier is used to pre-filter the genotype matrix to data from the project the researcher was viewing. Once clicking through to the genotype matrix, the researcher still needs to select which germplasm they want to see the data for.

1.4.2 Variant Pages

Variant pages are any Tripal Content which stores it's base data in the Chado Feature table and are of type SO:sequence_variant including the default Tripal Content Type "Sequence Variant". The genus is determined based on the associated organism and the variant name is used to pre-filter the genotype matrix to data specific to the variant being viewed by the researcher. Once clicking through to the genotype matrix, the researcher still needs to select which germplasm they want to see the data for.

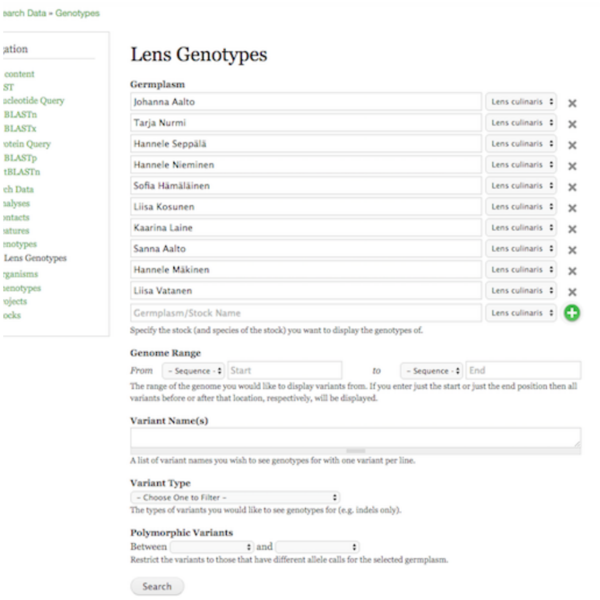
1.4.3 Genetic Marker Pages

Genetic Marker pages are any Tripal Content which stores it's base data in the Chado Feature table and are of type SO:genetic_marker including the default Tripal Content Type "Genetic Marker". The genus is determined based on the associated organism. The Genotype Matrix will be pre-filtered to any sequence variants related to the current genetic marker. Once clicking through to the genotype matrix, the researcher still needs to select which germplasm they want to see the data for.

1.4.4 Germplasm Pages

Germplasm pages are any Tripal Content which stores it base data in the Chado stock table including "Germplasm Accession" and "Cultivar (germplasm Variety)" and "Generated Germplasm (breeding Cross)" default Tripal Content Types. The genus is determined based on the associated organism and the unique germplasm identifier is used to ensure the pre-filtered matrix is showing the correct germplasm to the user. This provides a great way for researchers to access the genotypic data quickly and intuitively from the germplasm page.

The following screenshots are meant to visually summarize the features. For more detail, please click on one of the features above.



Lens Genotypes

Germplasm

Johanna Aalto	Lens culinaris ✕
Tarja Nurmi	Lens culinaris ✕
Hannele Seppälä	Lens culinaris ✕
Hannele Nieminen	Lens culinaris ✕
Sofia Hämäläinen	Lens culinaris ✕
Liisa Kosunen	Lens culinaris ✕
Kaarina Laine	Lens culinaris ✕
Sanna Aalto	Lens culinaris ✕
Hannele Mäkinen	Lens culinaris ✕
Liisa Vatanen	Lens culinaris ✕
Germplasm/Stock Name	Lens culinaris ⊕

Specify the stock (and species of the stock) you want to display the genotypes of.

Genome Range

From - Sequence - Start *to* - Sequence - End

The range of the genome you would like to display variants from. If you enter just the start or just the end position then all variants before or after that location, respectively, will be displayed.

Variant Name(s)

A list of variant names you wish to see genotypes for with one variant per line.

Variant Type

- Choose One to Filter -

The types of variants you would like to see genotypes for (e.g. indels only).

Polymorphic Variants

Between and

Restrict the variants to those that have different allele calls for the selected germplasm.

Search

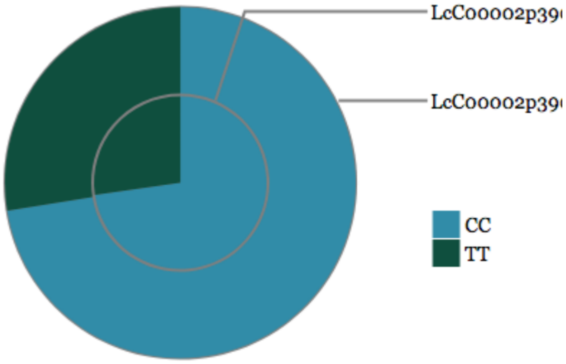
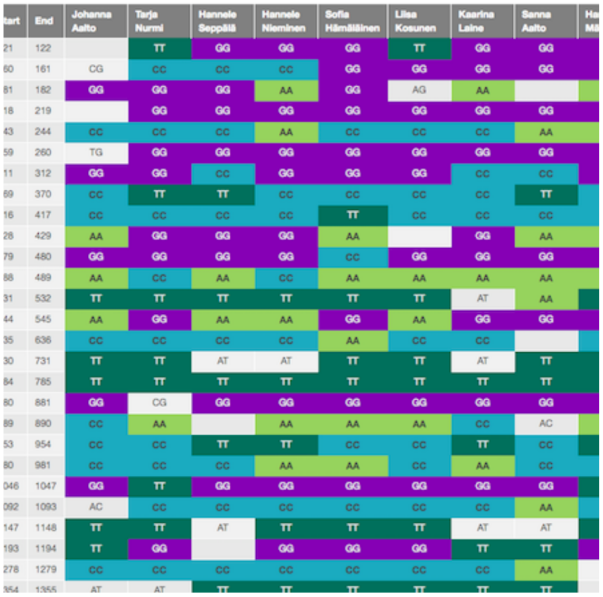
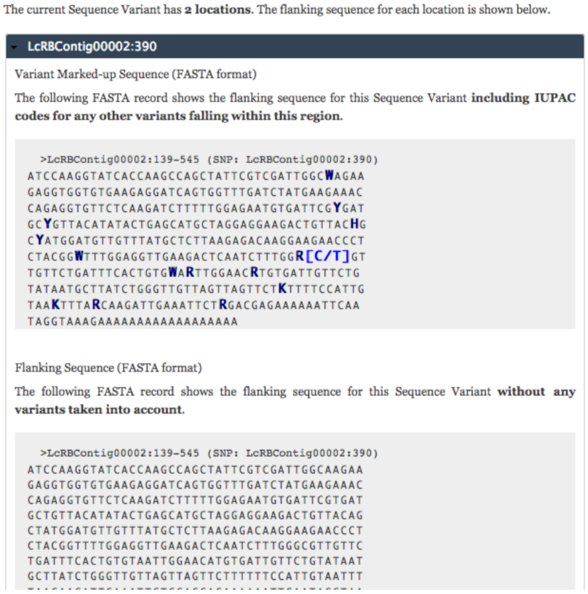


Figure: The ratio of alleles per marker assaying this variant. The
by 2 different marker(s). The ratio of alleles for each marker is shown as o
This allows you to compare the ratio across marker(s), as well as, get an ov



tart	End	Johanna Aalto	Tarja Nurmi	Hannele Seppälä	Hannele Nieminen	Sofia Hämäläinen	Liisa Kosunen	Kaarina Laine	Sanna Aalto
21	122		TT	GG	GG	GG	TT	GG	GG
60	161	CG	CC	CC	CC	GG	GG	GG	GG
81	182	GG	GG	GG	AA	GG	AG	AA	
18	219		GG	GG	GG	GG	GG	GG	GG
43	244	CC	CC	CC	AA	CC	CC	CC	AA
59	260	TG	GG	GG	GG	GG	GG	GG	GG
11	312	GG	GG	CC	GG	GG	GG	CC	CC
69	370	CC	TT	TT	CC	CC	CC	CC	TT
16	417	CC	CC	CC	CC	TT	CC	CC	CC
28	429	AA	GG	GG	GG	AA		GG	AA
79	480	GG	GG	GG	GG	CC	GG	GG	GG
88	489	AA	CC	AA	CC	AA	AA	AA	AA
31	532	TT	TT	TT	TT	TT	TT	AT	AA
44	545	AA	GG	AA	AA	GG	AA	GG	GG
35	636	CC	CC	CC	CC	AA	CC	CC	
30	731	TT	TT	AT	AT	TT	TT	AT	TT
84	785	TT	TT	TT	TT	TT	TT	TT	TT
80	881	GG	CG	GG	GG	GG	GG	GG	GG
89	890	CC	AA		AA	AA	AA	CC	AC
53	954	CC	CC	TT	TT	CC	CC	TT	CC
80	981	CC	CC	CC	AA	AA	CC	AA	CC
046	1047	GG	TT	GG	GG	GG	GG	GG	GG
092	1093	AC	CC	CC	CC	CC	CC	CC	AA
147	1148	TT	TT	AT	TT	TT	TT	AT	AT
193	1194	TT	GG		GG	GG	GG	TT	TT
278	1279	CC	CC	CC	CC	CC	CC	CC	AA
354	1355	AT	AT	TT	TT	TT	TT	TT	TT



The current Sequence Variant has **2 locations**. The flanking sequence for each location is shown below.

LcRBContig00002:390

Variant Marked-up Sequence (FASTA format)

The following FASTA record shows the flanking sequence for this Sequence Variant **including IUPAC codes for any other variants falling within this region.**

```
  >LcRBContig00002:139-545 (SNP: LcRBContig00002:390)
ATCCAAGGTATCACCAAGCCAGCTATTCGTCGATTGGCWAGAA
GAGGTGGTGTGAAGAGGATCAGTGGTTTGATCTATGAAGAAAC
CAGAGGTGTTCTCAAGATCTTTTTGGAGAATGTGATTCGYGAT
GCYGTTACATATACTGAGCATGCTAGGAGGAAGACTGTTACHG
CYATGGATGTTGTTTATGCTCTTAAGAGACAAGGAAGAACCCT
CTACGGWTTTGGAGGTTGAAGACTCAATCTTTGGR[C/T]GT
TGTTCTGATTTCACTGTGWARTTGGAACRTGTGATTGTTCTG
TATAATGCTTATCTGGGTTGTTAGTTAGTTCTKTTTTCCATTG
TAAKTTTARCAAGATTGAAATTCTRGACGAGAAAAAATTCAA
TAGGTAAAGAAAAAAAAAAAAAAAAAA
```

Flanking Sequence (FASTA format)

The following FASTA record shows the flanking sequence for this Sequence Variant **without any variants taken into account.**

```
  >LcRBContig00002:139-545 (SNP: LcRBContig00002:390)
ATCCAAGGTATCACCAAGCCAGCTATTCGTCGATTGGCAAGAA
GAGGTGGTGTGAAGAGGATCAGTGGTTTGATCTATGAAGAAAC
CAGAGGTGTTCTCAAGATCTTTTTGGAGAATGTGATTCGTGAT
GCTGTTACATATACTGAGCATGCTAGGAGGAAGACTGTTACAG
CTATGGATGTTGTTTATGCTCTTAAGAGACAAGGAAGAACCCT
CTACGGTTTTGGAGGTTGAAGACTCAATCTTTGGGCGTTGTTC
TGATTTCACTGTGTAATTGGAACATGTGATTGTTCTGTATAAT
GCTTATCTGGGTTGTTAGTTAGTTCTTTTTTCCATTGTAATTT
```

CHAPTER 2

Installation

2.1 Quickstart

This installation assumes you have Tripal 3.x and PostgreSQL 9.3+.

1. Install the following dependencies: Drupal Libraries API, Tripal D3.js, Tripal Donwload API.

```
drush pm-download libraries
drush pm-enable libraries -y
cd [drupal root]/sites/all/modules
git clone https://github.com/tripal/tripald3
git clone https://github.com/tripal/trpdownload_api
cd [drupal root]/sites/all/libraries
mkdir d3 && cd d3
wget https://github.com/d3/d3/releases/download/v3.5.17/d3.zip
unzip d3.zip
drush pm-enable trpdownload_api tripald3 -y
```

2. Install this module as you would any Drupal module.

```
cd [drupal root]/sites/all/modules
git clone https://github.com/UofS-Pulse-Binfo/nd_genotypes.git
drush en nd_genotypes -y
```

3. Load data using the genotype loader. Since the Genotype loader is not yet released, we highly suggest test loading each dataset on a development site.

4. Configure this module by going to Administration » Tripal » Extensions » Natural Diversity Genotypes » Settings.

5. Once data is available make sure to sync it (Administration » Tripal » Extensions » Natural Diversity Genotypes » Sync)

Note: If you do not have data and you want to try the module out, you can use the Tripal Test Suite Database Seeder provided with this module. See *Manual Testing (Demonstration)*.

- You can access the genotype matrix at `[your drupal site]/chado/genotype/[genus]`.
- You should see a “Genotypes” and updated “Sequences” pane on Genetic Marker and Variant pages.
  - You may need to go to Administration > Structure > Tripal Content Types > Genetic Marker > Manage Fields and click “Find new fields”.
  - Then go to “Manage Display” and enable the field by dragging it into the display area.

Note: If ND Genotypes fields are not automatically attached to the genetic marker and sequence variant content types, go to the “Manage Fields” page for each and click “Find new fields”. Also, go to the “Manage Display” page and ensure they are not hidden.

2.2 Dependencies

1. Tripal 3.x
2. Drupal Libraries API
3. Tripal Download API
4. Tripal D3.js
5. PostgreSQL 9.3 (9.4+ recommended; tested with 11.3)

2.3 Installation

1. Install the following dependencies: Drupal Libraries API, Tripal D3.js, Tripal Donwload API.
   - First we install the Drupal Libraries API which is required for Tripal D3.

   ```
   drush pm-download libraries
   drush pm-enable libraries -y
   ```

   - Next we grab the latest version of the remaining dependencies from Github.

   ```
   cd [drupal root]/sites/all/modules
   git clone https://github.com/tripal/tripald3
   git clone https://github.com/tripal/trpdownload_api
   ```

   - The charts for the module are drawn using D3.js v3 . As such we need to download it and place it in our libraries folder.

   ```
   cd [drupal root]/sites/all/libraries
   mkdir d3 && cd d3
   wget https://github.com/d3/d3/releases/download/v3.5.17/d3.zip
   unzip d3.zip
   ```

   - Finally we can enable the last of our dependencies.

   ```
   drush pm-enable trpdownload_api tripald3 -y
   ```

2. Install this module as you would any Drupal module.

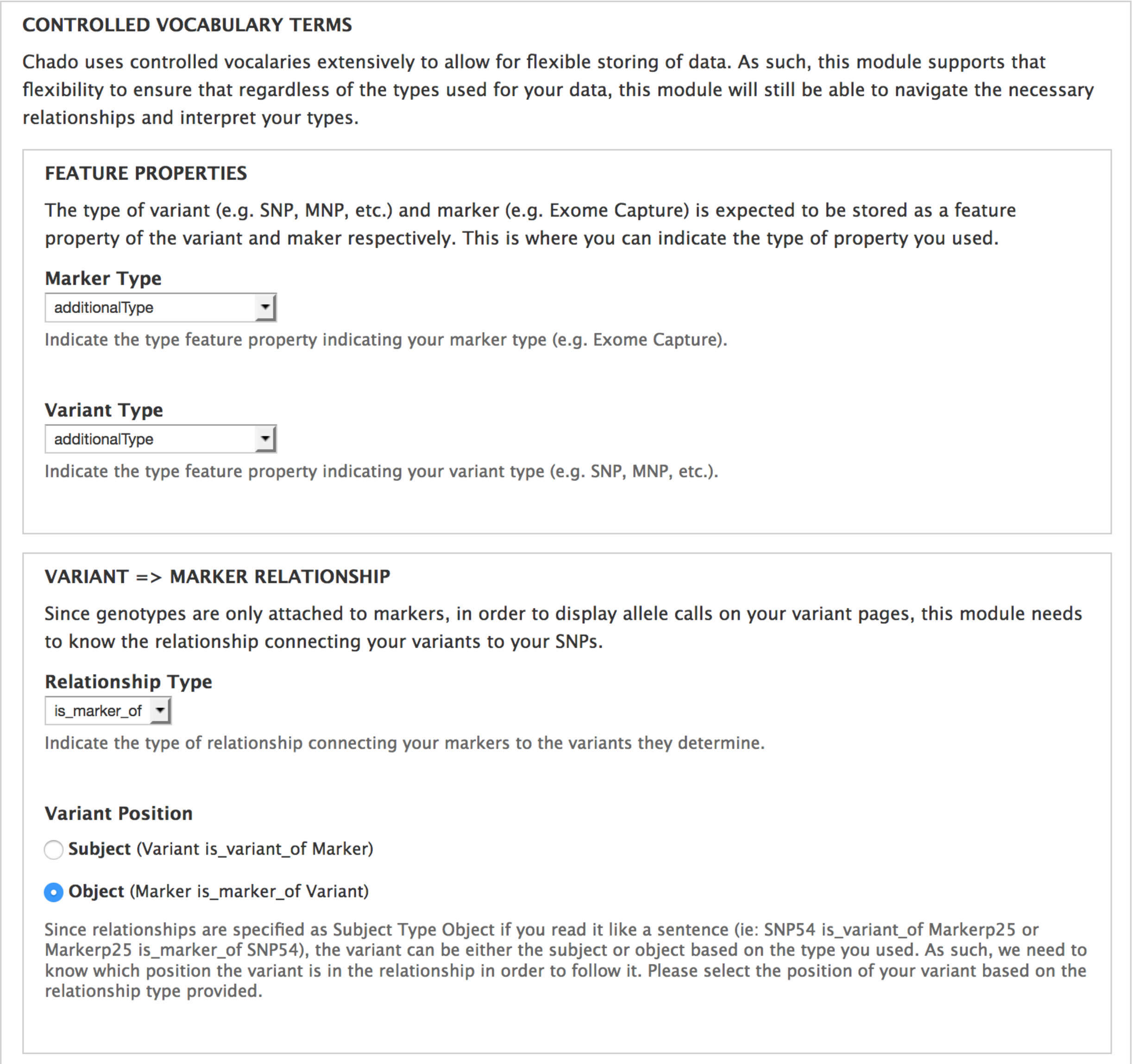
```
cd [drupal root]/sites/all/modules
git clone https://github.com/UofS-Pulse-Binfo/nd_genotypes.git
drush en nd_genotypes -y
```

CHAPTER 3

Configuration

3.1 Set Controlled Vocabulary Terms

1. Navigate to Administration » Tripal » Extensions » Natural Diversity Genotypes » Settings
2. Under "Controlled Vocabulary Terms" you will see a number of drop-downs. Simply set these to the terms you use in your chado database. This allows ND Genotypes to better support the flexibility of Chado and allows you to use the terms most fitting for your data.



CONTROLLED VOCABULARY TERMS

Chado uses controlled vocalaries extensively to allow for flexible storing of data. As such, this module supports that flexibility to ensure that regardless of the types used for your data, this module will still be able to navigate the necessary relationships and interpret your types.

FEATURE PROPERTIES

The type of variant (e.g. SNP, MNP, etc.) and marker (e.g. Exome Capture) is expected to be stored as a feature property of the variant and maker respectively. This is where you can indicate the type of property you used.

Marker Type

additionalType

Indicate the type feature property indicating your marker type (e.g. Exome Capture).

Variant Type

additionalType

Indicate the type feature property indicating your variant type (e.g. SNP, MNP, etc.).

VARIANT => MARKER RELATIONSHIP

Since genotypes are only attached to markers, in order to display allele calls on your variant pages, this module needs to know the relationship connecting your variants to your SNPs.

Relationship Type

is_marker_of

Indicate the type of relationship connecting your markers to the variants they determine.

Variant Position

○ **Subject** (Variant is_variant_of Marker)

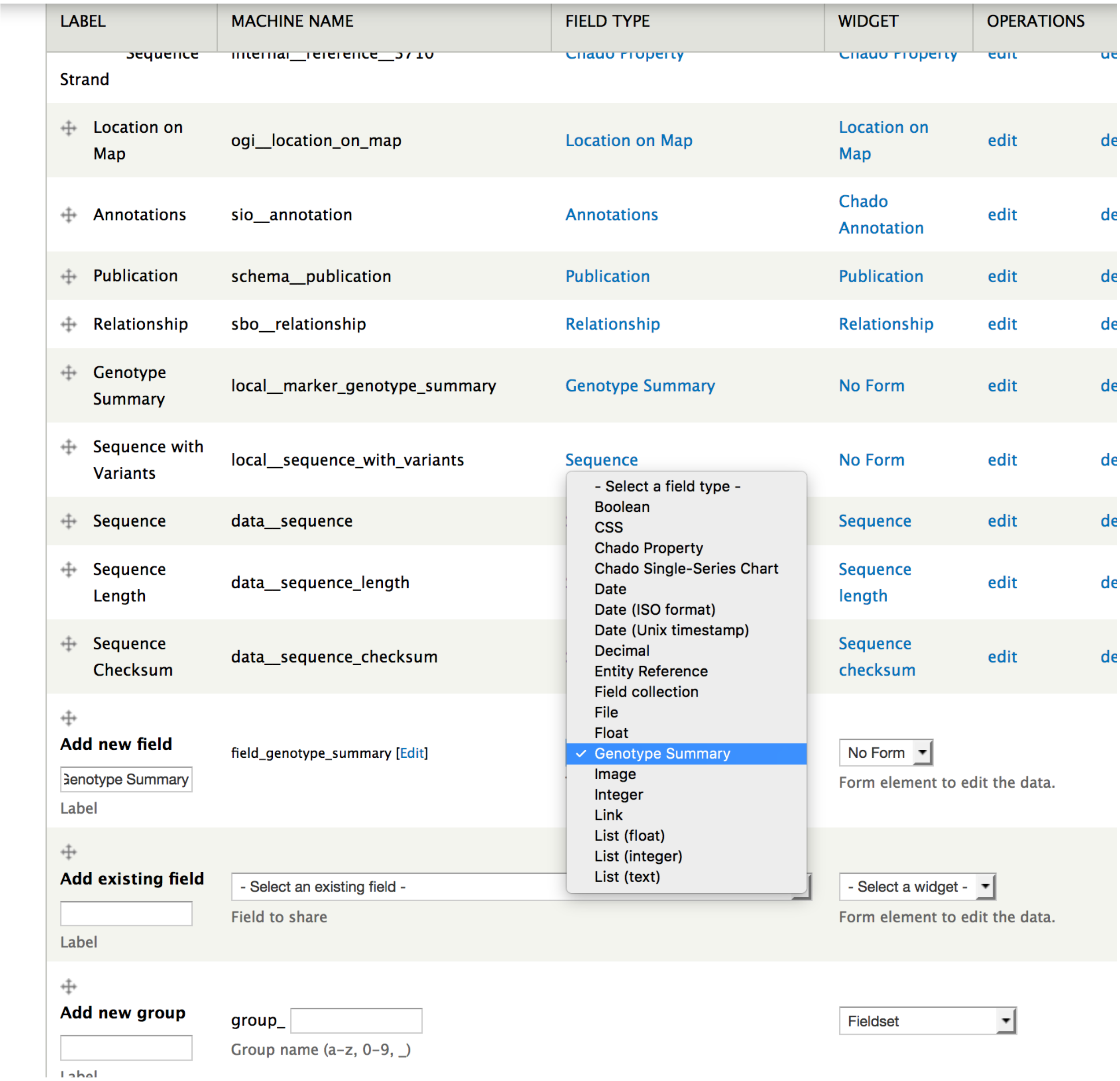
● **Object** (Marker is_marker_of Variant)

Since relationships are specified as Subject Type Object if you read it like a sentence (ie: SNP54 is_variant_of Markerp25 or Markerp25 is_marker_of SNP54), the variant can be either the subject or object based on the type you used. As such, we need to know which position the variant is in the relationship in order to follow it. Please select the position of your variant based on the relationship type provided.

3. Click “Save Terms” once you’ve set them all appropriately.

3.2 Add Genotype Summaries to Variant/Marker Pages

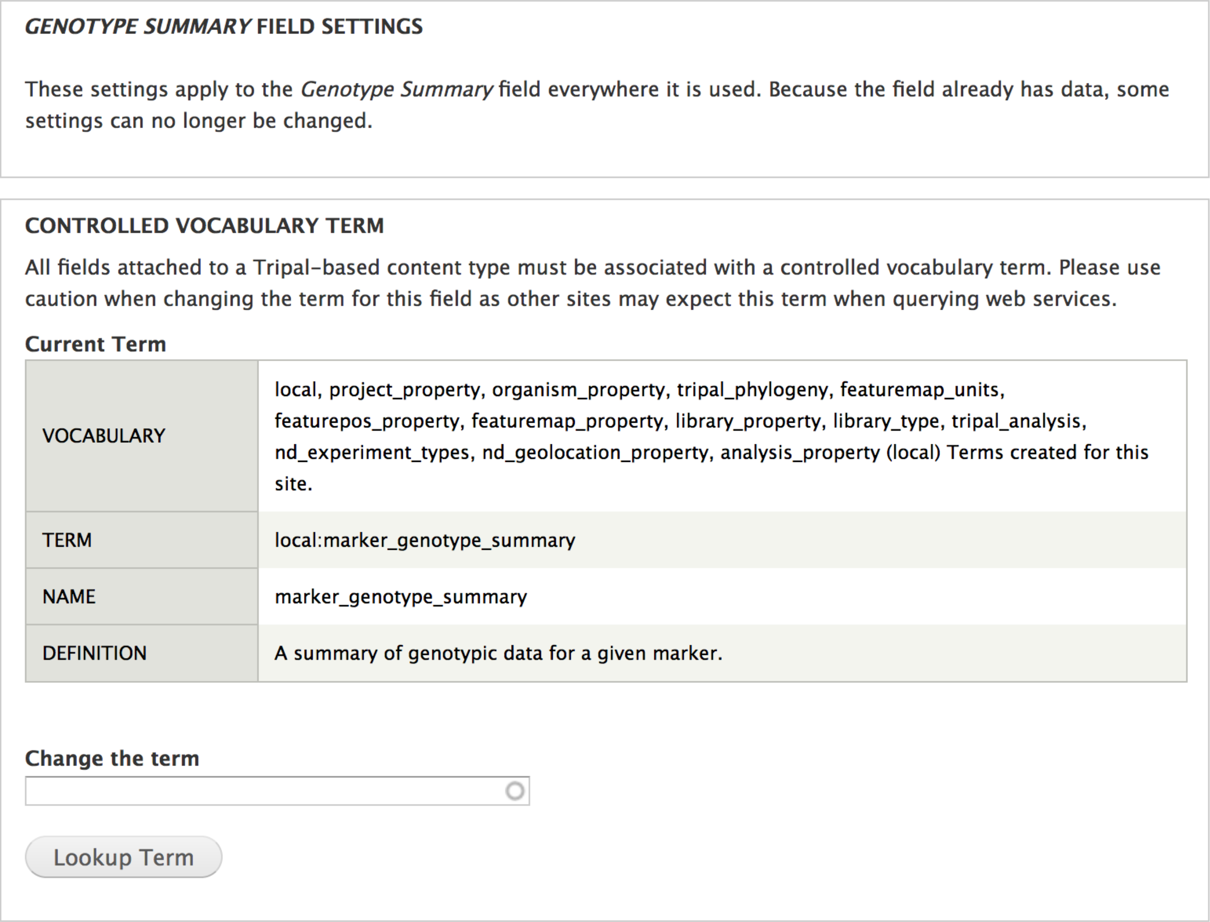
1. Navigate to Administration » Structure » Tripal Content Types » [Variant/Marker] » Manage Fields
2. Scroll down to “Add a new field”, enter a label and select “Genotype Summary” from the first drop-down.



LABEL	MACHINE NAME	FIELD TYPE	WIDGET	OPERATIONS	
Sequence Strand	internal__reference__3710	Chado Property	Chado Property	edit	de
Location on Map	ogi__location_on_map	Location on Map	Location on Map	edit	de
Annotations	sio__annotation	Annotations	Chado Annotation	edit	de
Publication	schema__publication	Publication	Publication	edit	de
Relationship	sbo__relationship	Relationship	Relationship	edit	de
Genotype Summary	local__marker_genotype_summary	Genotype Summary	No Form	edit	de
Sequence with Variants	local__sequence_with_variants	Sequence	No Form	edit	de
Sequence	data__sequence		Sequence	edit	de
Sequence Length	data__sequence_length		Sequence length	edit	de
Sequence Checksum	data__sequence_checksum		Sequence checksum	edit	de
Add new field Genotype Summary Label	field_genotype_summary [Edit]		No Form Form element to edit the data.		
Add existing field Label	- Select an existing field - Field to share		- Select a widget - Form element to edit the data.		
Add new group Label	group_ Group name (a-z, 0-9, _)		Fieldset		

- Select a field type -
Boolean
CSS
Chado Property
Chado Single-Series Chart
Date
Date (ISO format)
Date (Unix timestamp)
Decimal
Entity Reference
Field collection
File
Float
✓ Genotype Summary
Image
Integer
Link
List (float)
List (integer)
List (text)

3. Choose a term for the field or create a local one



GENOTYPE SUMMARY **FIELD SETTINGS**

These settings apply to the *Genotype Summary* field everywhere it is used. Because the field already has data, some settings can no longer be changed.

CONTROLLED VOCABULARY TERM

All fields attached to a Tripal-based content type must be associated with a controlled vocabulary term. Please use caution when changing the term for this field as other sites may expect this term when querying web services.

Current Term

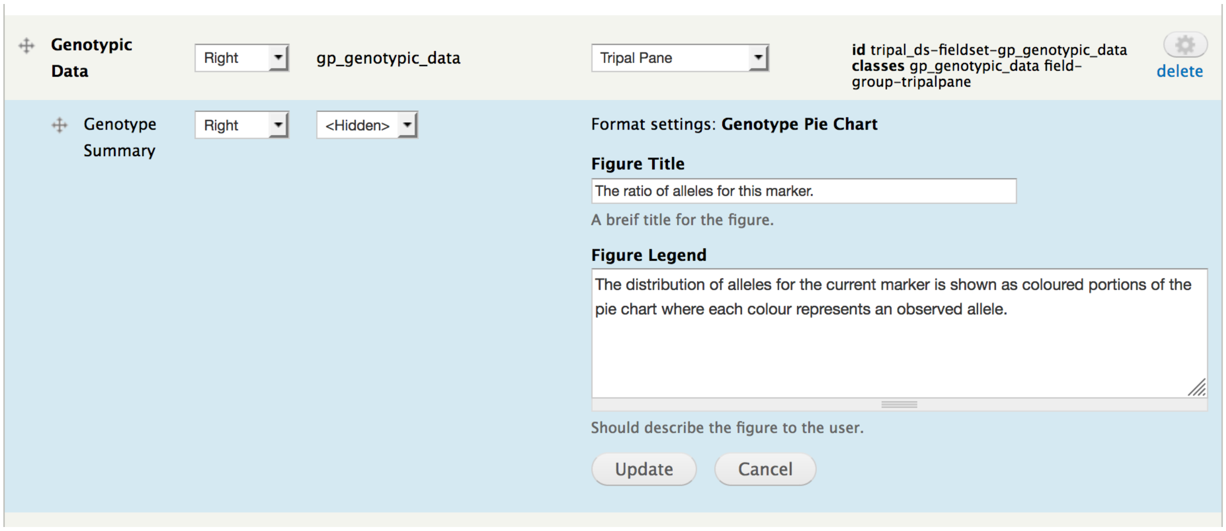
VOCABULARY	local, project_property, organism_property, tripal_phylogeny, featuremap_units, featurepos_property, featuremap_property, library_property, library_type, tripal_analysis, nd_experiment_types, nd_geolocation_property, analysis_property (local) Terms created for this site.
TERM	local:marker_genotype_summary
NAME	marker_genotype_summary
DEFINITION	A summary of genotypic data for a given marker.

Change the term

4. Navigate to "Manage Display" for the same content type and ensure the field you just created is placed where you would like it to be.

Warning: Ensure that the field is not in the "Disabled" section under "Manage Display"; otherwise, it will not appear on the page.

5. You can also configure the figure legend. On the "Manage Display" page, click the gear icon at the far right of the Genotype Summary field.



Format settings: **Genotype Pie Chart**

Figure Title

The ratio of alleles for this marker.

A breif title for the figure.

Figure Legend

The distribution of alleles for the current marker is shown as coloured portions of the pie chart where each colour represents an observed allele.

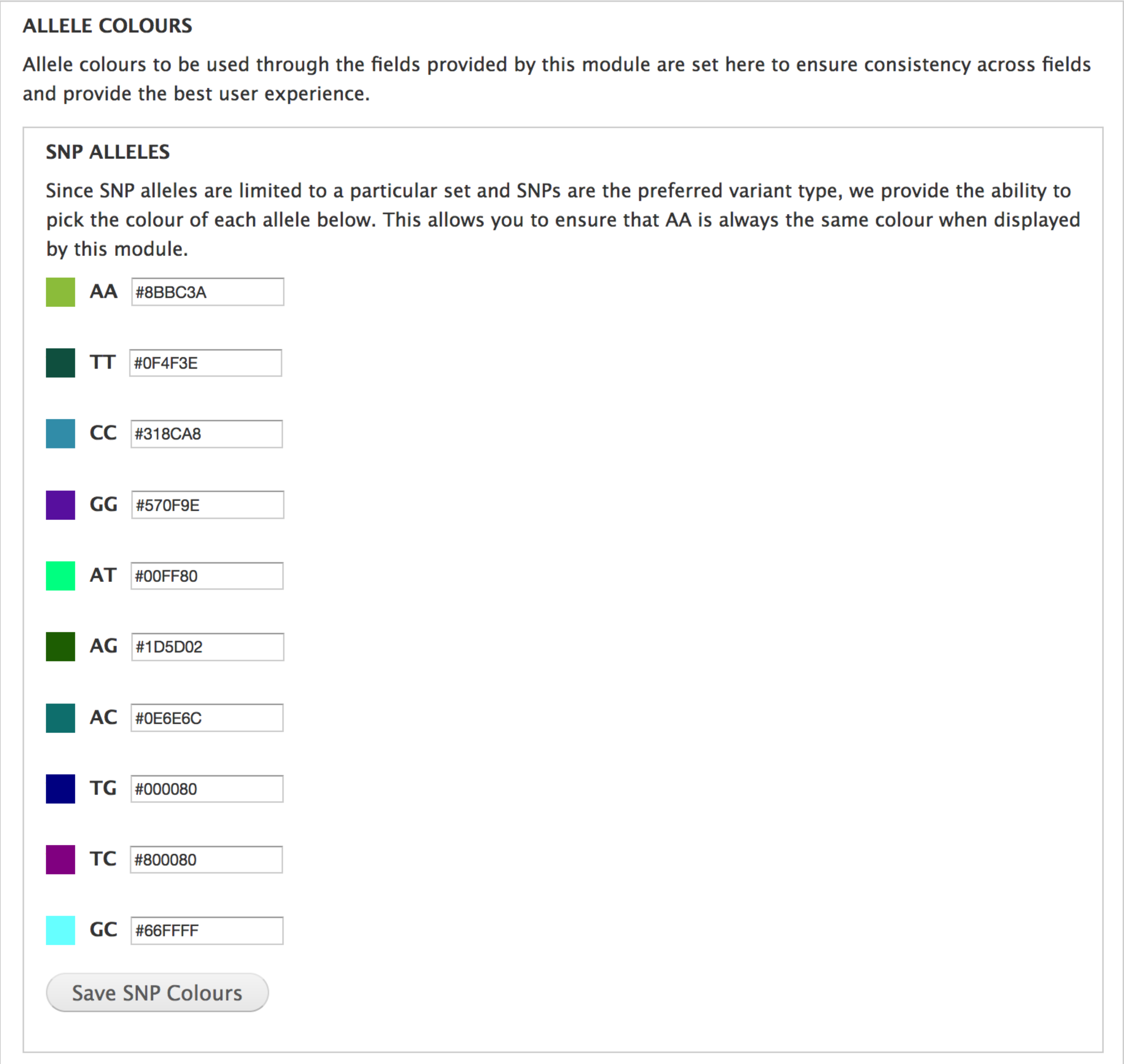
Should describe the figure to the user.

Warning: Make sure to click “Update” in the blue settings pane; as well as, “Save” at the bottom of the page.

3.3 Set Preferred Allele Colours (Optional)

You can also change the colours used for the genotype matrix and summary charts:

1. Navigate to Administration » Tripal » Extensions » Natural Diversity Genotypes » Settings
2. Under “Allele Colours” enter the HEX code for the colours you would like to use. Once you save the colours, you will see your choice demonstrated in front of the allele.



ALLELE COLOURS

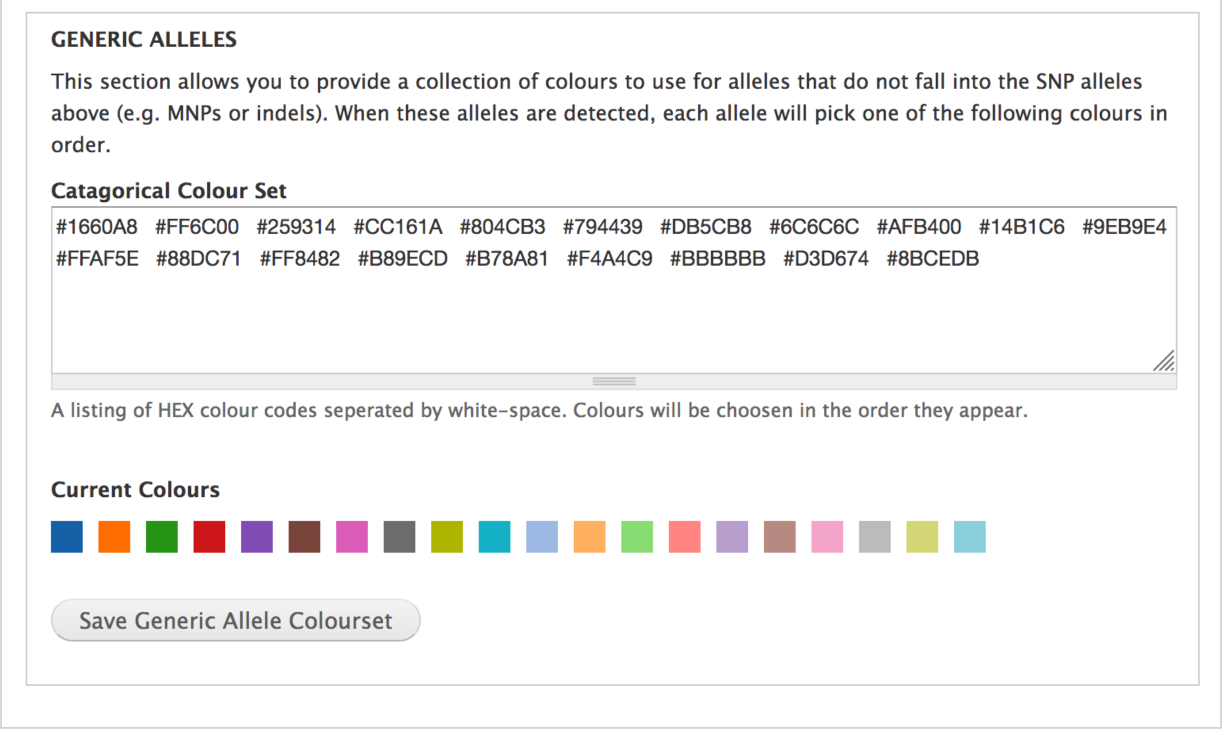
Allele colours to be used through the fields provided by this module are set here to ensure consistency across fields and provide the best user experience.

SNP ALLELES

Since SNP alleles are limited to a particular set and SNPs are the preferred variant type, we provide the ability to pick the colour of each allele below. This allows you to ensure that AA is always the same colour when displayed by this module.

🟩	AA	#8BBC3A
🟩	TT	#0F4F3E
🟦	CC	#318CA8
🟪	GG	#570F9E
🟩	AT	#00FF80
🟩	AG	#1D5D02
🟦	AC	#0E6E6C
🟦	TG	#000080
🟪	TC	#800080
🟦	GC	#66FFFF

3. You can also indicate a collection of colours you would like to be used for alleles that don't fall into the typical SNP categories such as MNPs.



GENERIC ALLELES

This section allows you to provide a collection of colours to use for alleles that do not fall into the SNP alleles above (e.g. MNPs or indels). When these alleles are detected, each allele will pick one of the following colours in order.

Catagorical Colour Set

```
#1660A8 #FF6C00 #259314 #CC161A #804CB3 #794439 #DB5CB8 #6C6C6C #AFB400 #14B1C6 #9EB9E4
#FFAF5E #88DC71 #FF8482 #B89ECD #B78A81 #F4A4C9 #BBBBBB #D3D674 #8BCEDB
```

A listing of HEX colour codes seperated by white-space. Colours will be choosen in the order they appear.

Current Colours

CHAPTER 4

Use Cases

The following tutorials walk researchers through how these tools can be used to answer common research questions.

4.1 Find a variant in a trait-implicated region

Research Question:

Through other analysis you have a region of the genome which likely contributes to a specific phenotype for your trait of interest. Now you would like to find a causative or at least correlated sequence variant. For this purpose you know at least two germplasm with differing phenotypes which you have genotypic data available for.

Fictional Example:

- Trait: `FAIRness`
- Region of interest: `non:150-300`
- Germplasm with FAIRness: `placeat libero cupiditate et`
- Germplasm without FAIRness: `omnis fuga molestiae et`

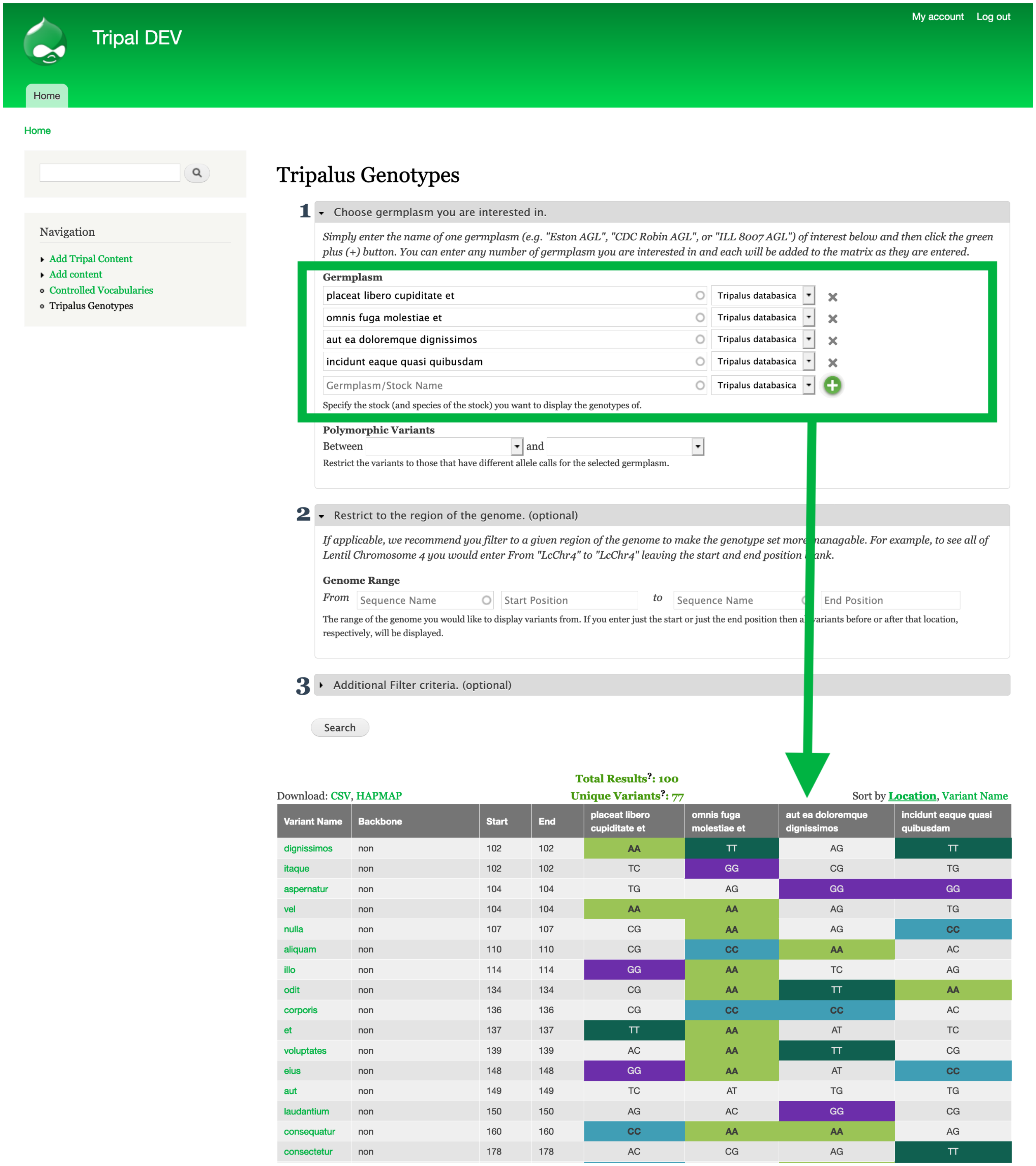
Data:

This example uses simulated data for the fictional species Tripalus databasica. You can generate similar using the Tripal Test Suite as described here: *Manual Testing (Demonstration)*. You can also use your own data by importing it into your Tripal site using the genotype loader.

4.1.1 Step #1: Find genotypic data for your reference germplasm

- Go to `[yourtripalsite]/chado/genotype/[Genus]` (e.g. `http://localhost/tripal-DEV/chado/genotype/Tripalus`) to access the genotype matrix tool for the genus of the germplasm you are interested in.

- Enter the name of each germplasm you are interested in by typing it in the textfield labelled germplasm. Then check the correct species is selected to the right of the textbox. To add more then one germplasm click the green + button.
- Each time you click the green + button or search, the genotypic data for the listed germplasm will be shown.



Tripalus Genotypes

1 ▾ Choose germplasm you are interested in.

Simply enter the name of one germplasm (e.g. "Eston AGL", "CDC Robin AGL", or "ILL 8007 AGL") of interest below and then click the green plus (+) button. You can enter any number of germplasm you are interested in and each will be added to the matrix as they are entered.

Germplasm

placeat libero cupiditate et ○	Tripalus databasica ▾	✕
omnis fuga molestiae et ○	Tripalus databasica ▾	✕
aut ea doloremque dignissimos ○	Tripalus databasica ▾	✕
incidunt eaque quasi quibusdam ○	Tripalus databasica ▾	✕
Germplasm/Stock Name ○	Tripalus databasica ▾	⊕

Specify the stock (and species of the stock) you want to display the genotypes of.

Polymorphic Variants

Between ▾ and ▾

Restrict the variants to those that have different allele calls for the selected germplasm.

2 ▾ Restrict to the region of the genome. (optional)

If applicable, we recommend you filter to a given region of the genome to make the genotype set more managable. For example, to see all of Lentil Chromosome 4 you would enter From "LcChr4" to "LcChr4" leaving the start and end position blank.

Genome Range

From Sequence Name ○ Start Position *to* Sequence Name ○ End Position

The range of the genome you would like to display variants from. If you enter just the start or just the end position then all variants before or after that location, respectively, will be displayed.

3 ▸ Additional Filter criteria. (optional)

Search

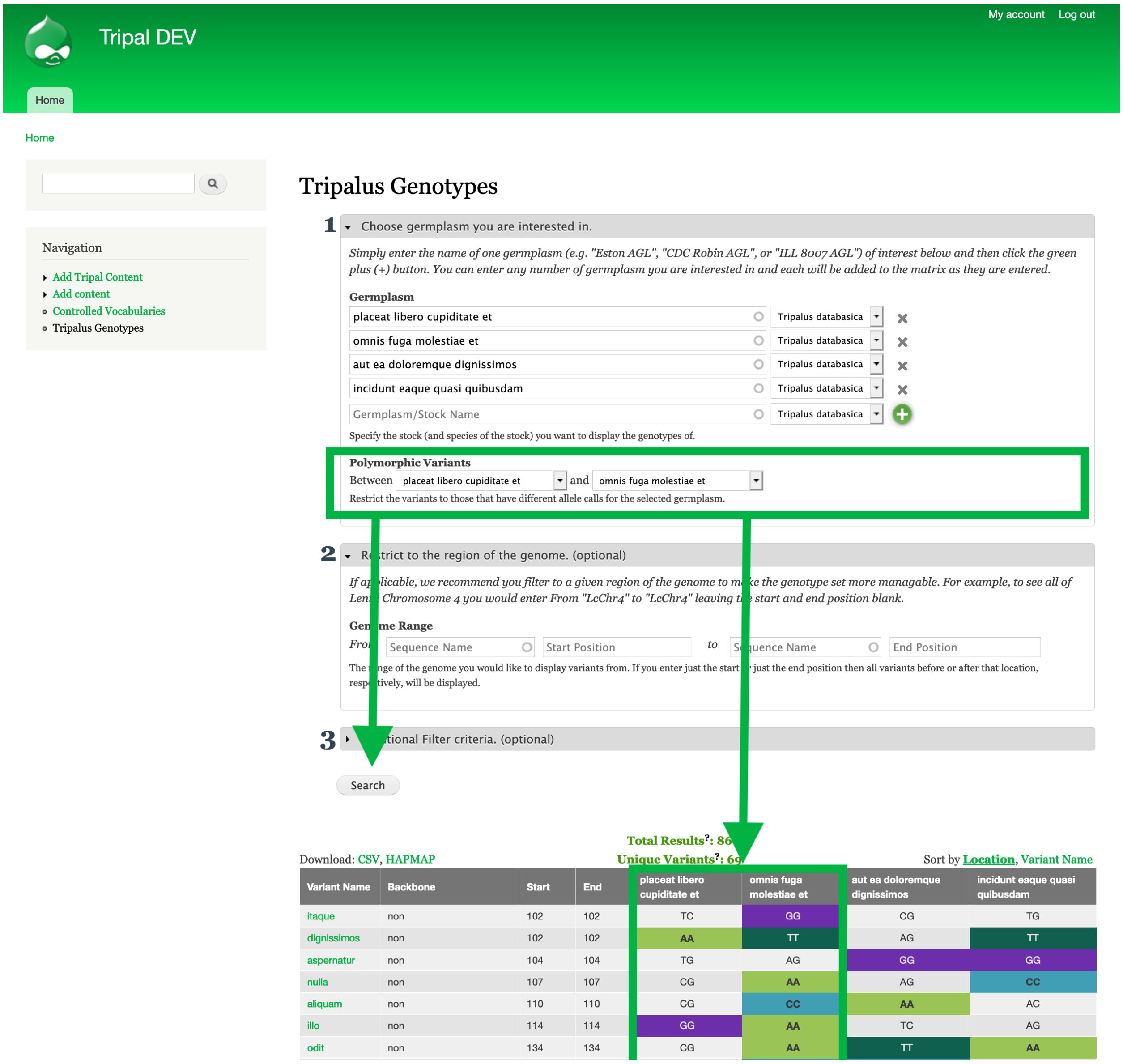
Total Results?: 100

Download: CSV, HAPMAP **Unique Variants?: 77** Sort by Location, Variant Name

Variant Name	Backbone	Start	End	placeat libero cupiditate et	omnis fuga molestiae et	aut ea doloremque dignissimos	incidunt eaque quasi quibusdam
dignissimos	non	102	102	AA	TT	AG	TT
itaque	non	102	102	TC	GG	CG	TG
aspernatur	non	104	104	TG	AG	GG	GG
vel	non	104	104	AA	AA	AG	TG
nulla	non	107	107	CG	AA	AG	CC
aliquam	non	110	110	CG	CC	AA	AC
illo	non	114	114	GG	AA	TC	AG
odit	non	134	134	CG	AA	TT	AA
corporis	non	136	136	CG	CC	CC	AC
et	non	137	137	TT	AA	AT	TC
voluptates	non	139	139	AC	AA	TT	CG
eius	non	148	148	GG	AA	AT	CC
aut	non	149	149	TC	AT	TG	TG
laudantium	non	150	150	AG	AC	GG	CG
consequatur	non	160	160	CC	AA	AA	AG
consectetur	non	178	178	AC	CG	AG	TT

4.1.2 Step #2: Restrict the Sequence Variants to polymorphic between your germplasm

- Underneath germplasm, there is a filter to restrict to polymorphic variants. This filter compares two germplasm and only shows variants with different genotypic calls.
- For our example, we would select `placeat libero cupiditate et` in the first drop down and `omnis fuga molestiae et` in the second drop-down to see only sequence variants with differing genotypes (i.e polymorphic variants) between these two germplasm.
- Click Search to see the results.



Tripalus Genotypes

1 ▾ Choose germplasm you are interested in.

Simply enter the name of one germplasm (e.g. "Eston AGL", "CDC Robin AGL", or "ILL 8007 AGL") of interest below and then click the green plus (+) button. You can enter any number of germplasm you are interested in and each will be added to the matrix as they are entered.

Germplasm

placeat libero cupiditate et	Tripalus databasica
omnis fuga molestiae et	Tripalus databasica
aut ea doloremque dignissimos	Tripalus databasica
incidunt eaque quasi quibusdam	Tripalus databasica
Germplasm/Stock Name	Tripalus databasica

Specify the stock (and species of the stock) you want to display the genotypes of.

Polymorphic Variants

Between placeat libero cupiditate et and omnis fuga molestiae et

Restrict the variants to those that have different allele calls for the selected germplasm.

2 ▾ Re trict to the region of the genome. (optional)

If ap licable, we recommend you filter to a given region of the genome to m ke the genotype set more managable. For example, to see all of Len Chromosome 4 you would enter From "LcChr4" to "LcChr4" leaving t e start and end position blank.

Gen me Range

Fro Sequence Name ○ Start Position *to* S quence Name ○ End Position

The ange of the genome you would like to display variants from. If you enter just the start just the end position then all variants before or after that location, resp tively, will be displayed.

3 ▸ tional Filter criteria. (optional)

Search

Total Results?: 86

Unique Variants?: 69

Download: CSV, HAPMAP

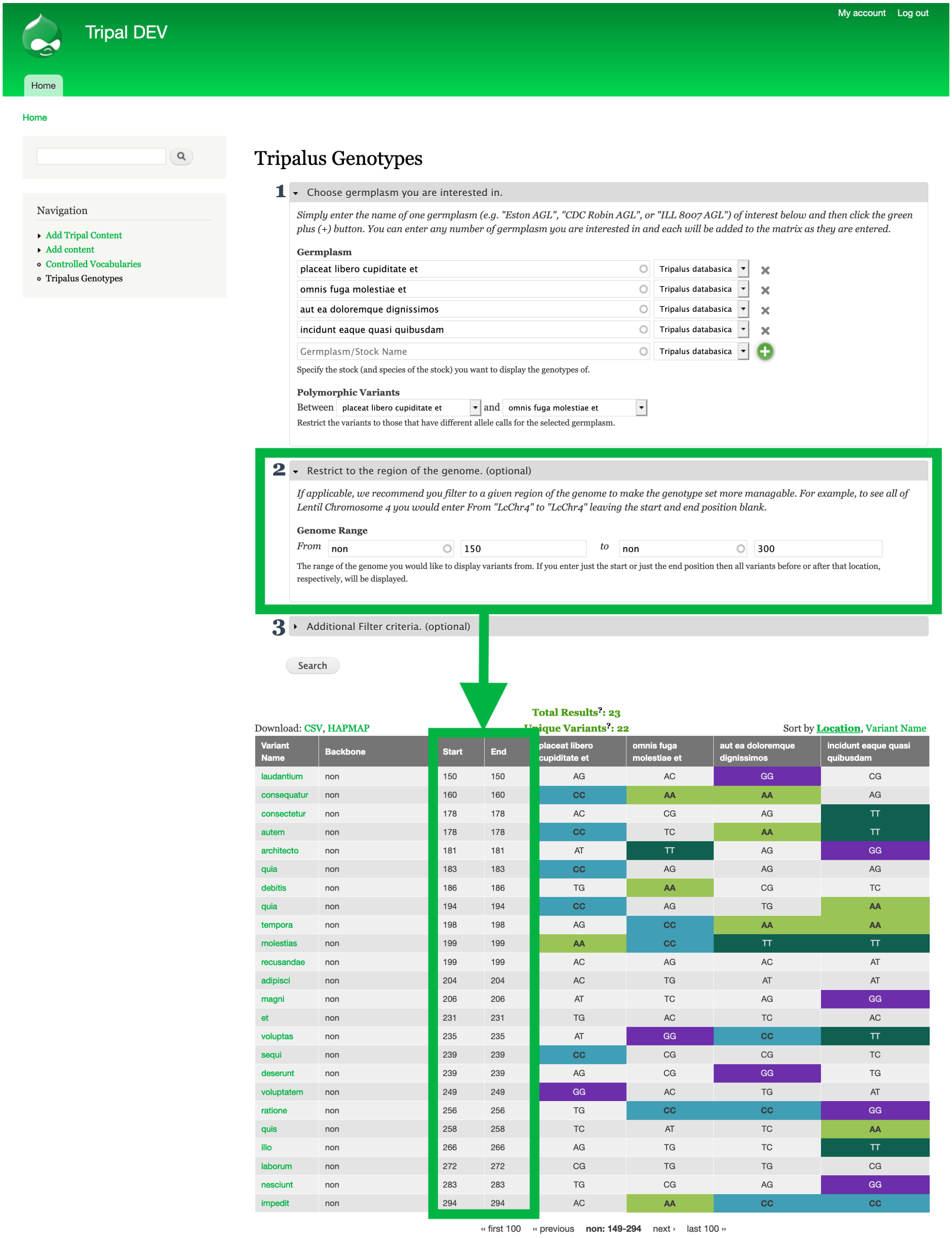
Sort by Location, Variant Name

Variant Name	Backbone	Start	End	placeat libero cupiditate et	omnis fuga molestiae et	aut ea doloremque dignissimos	incidunt eaque quasi quibusdam
itaque	non	102	102	TC	GG	CG	TG
dignissimos	non	102	102	AA	TT	AG	TT
aspernatur	non	104	104	TG	AG	GG	GG
nulla	non	107	107	CG	AA	AG	CC
aliquam	non	110	110	CG	CC	AA	AC
illo	non	114	114	GG	AA	TC	AG
odit	non	134	134	CG	AA	TT	AA

4.1.3 Step #3: Restrict to you trait-implicated Region of the Genome.

- The second section of the filter criteria available for the genotype matrix allows you to enter the region of the genome you are interested in. Once you click search, the genotype matrix will only show sequence variants found in this region.

- In our example, the region of interest is non:150-300. To enter this we place `non` for the `Sequence Name`, `150` for the start position and `300` for the end position.



Navigation

- Add Tripal Content
- Add content
- Controlled Vocabularies
- Tripalus Genotypes

Tripalus Genotypes

1 Choose germplasm you are interested in.

Simply enter the name of one germplasm (e.g. "Eston AGL", "CDC Robin AGL", or "ILL 8007 AGL") of interest below and then click the green plus (+) button. You can enter any number of germplasm you are interested in and each will be added to the matrix as they are entered.

Germplasm

placeat libero cupiditate et	Tripalus databasica
omnis fuga molestiae et	Tripalus databasica
aut ea doloremque dignissimos	Tripalus databasica
incidunt eaque quasi quibusdam	Tripalus databasica
Germplasm/Stock Name	Tripalus databasica

Specify the stock (and species of the stock) you want to display the genotypes of.

Polymorphic Variants

Between placeat libero cupiditate et and omnis fuga molestiae et

Restrict the variants to those that have different allele calls for the selected germplasm.

2 Restrict to the region of the genome. (optional)

If applicable, we recommend you filter to a given region of the genome to make the genotype set more managable. For example, to see all of Lentil Chromosome 4 you would enter From "LcChr4" to "LcChr4" leaving the start and end position blank.

Genome Range

From non 150 *to* non 300

The range of the genome you would like to display variants from. If you enter just the start or just the end position then all variants before or after that location, respectively, will be displayed.

3 Additional Filter criteria. (optional)

Search

Total Results?: 23

Unique Variants?: 22

Download: CSV, HAPMAP

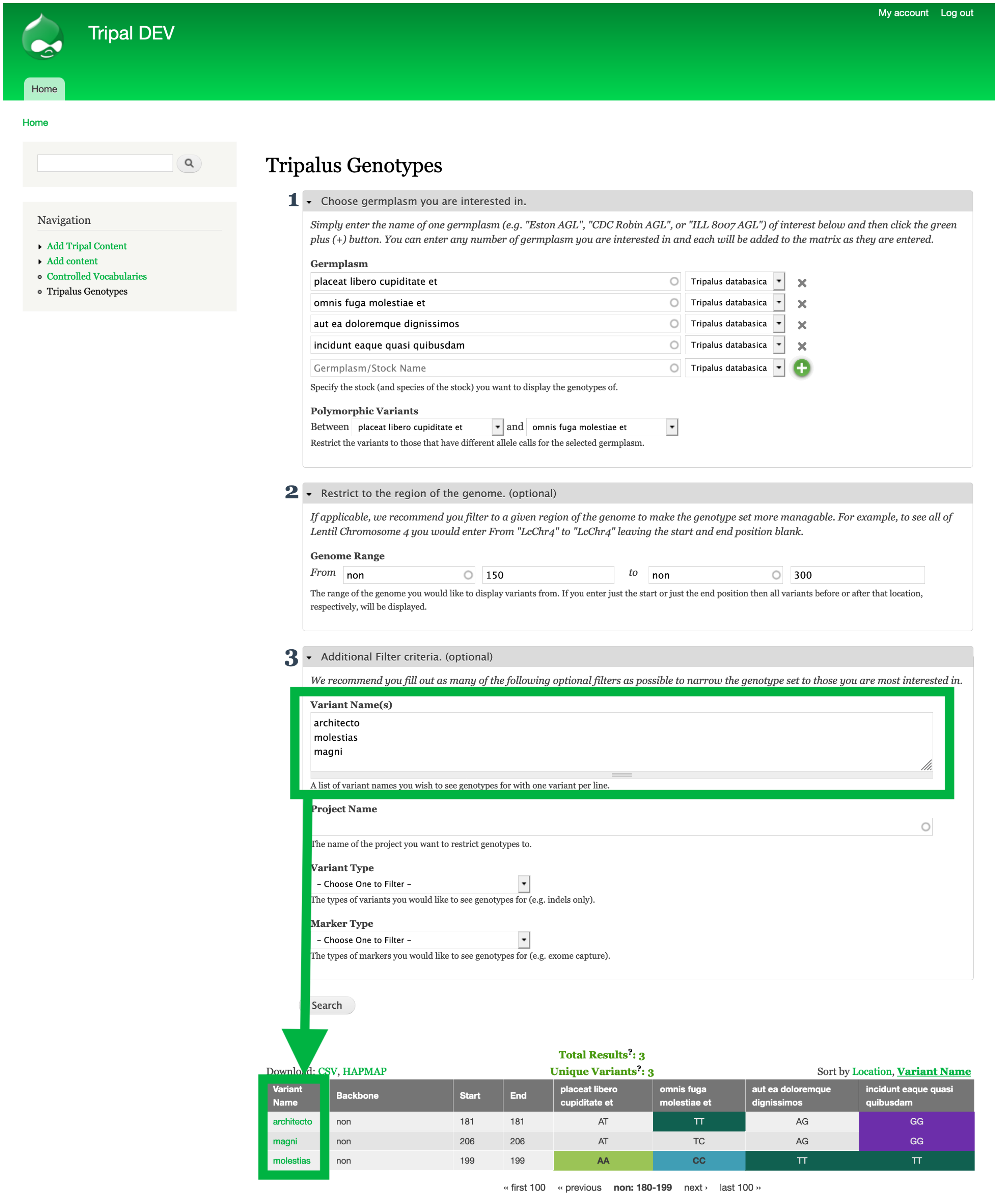
Sort by **Location**, Variant Name

Variant Name	Backbone	Start	End	placeat libero cupiditate et	omnis fuga molestiae et	aut ea doloremque dignissimos	incidunt eaque quasi quibusdam
laudantium	non	150	150	AG	AC	GG	CG
consequatur	non	160	160	CC	AA	AA	AG
consectetur	non	178	178	AC	CG	AG	TT
autem	non	178	178	CC	TC	AA	TT
architecto	non	181	181	AT	TT	AG	GG
quia	non	183	183	CC	AG	AG	AG
debitis	non	186	186	TG	AA	CG	TC
quia	non	194	194	CC	AG	TG	AA
tempora	non	198	198	AG	CC	AA	AA
molestias	non	199	199	AA	CC	TT	TT
recusandae	non	199	199	AC	AG	AC	AT
adipisci	non	204	204	AC	TG	AT	AT
magni	non	206	206	AT	TC	AG	GG
et	non	231	231	TG	AC	TC	AC
voluptas	non	235	235	AT	GG	CC	TT
sequi	non	239	239	CC	CG	CG	TC
deserunt	non	239	239	AG	CG	GG	TG
voluptatem	non	249	249	GG	AC	TG	AT
ratione	non	256	256	TG	CC	CC	GG
quis	non	258	258	TC	AT	TC	AA
illo	non	266	266	AG	TG	TC	TT
laborum	non	272	272	CG	TG	TG	CG
nesciunt	non	283	283	TG	CG	AG	GG
impedit	non	294	294	AC	AA	CC	CC

« first 100 « previous **non: 149-294** next › last 100 »

4.1.4 Step 4: (Optionally): Restrict to specific variants.

- Say further analysis shows that particular sequence variants in that region are more likely to contribute to your phenotype of interest.
- You can enter the specific variant names by expanding the `Additional Filter criteria` section then clicking Search.



Navigation

- Add Tripal Content
- Add content
- Controlled Vocabularies
- Tripalus Genotypes

Tripalus Genotypes

1 Choose germplasm you are interested in.

Simply enter the name of one germplasm (e.g. "Eston AGL", "CDC Robin AGL", or "ILL 8007 AGL") of interest below and then click the green plus (+) button. You can enter any number of germplasm you are interested in and each will be added to the matrix as they are entered.

Germplasm

placeat libero cupiditate et	Tripalus databasica
omnis fuga molestiae et	Tripalus databasica
aut ea doloremque dignissimos	Tripalus databasica
incidunt eaque quasi quibusdam	Tripalus databasica
Germplasm/Stock Name	Tripalus databasica

Specify the stock (and species of the stock) you want to display the genotypes of.

Polymorphic Variants

Between placeat libero cupiditate et and omnis fuga molestiae et

Restrict the variants to those that have different allele calls for the selected germplasm.

2 Restrict to the region of the genome. (optional)

If applicable, we recommend you filter to a given region of the genome to make the genotype set more managable. For example, to see all of Lentil Chromosome 4 you would enter From "LcChr4" to "LcChr4" leaving the start and end position blank.

Genome Range

From non 150 *to* non 300

The range of the genome you would like to display variants from. If you enter just the start or just the end position then all variants before or after that location, respectively, will be displayed.

3 Additional Filter criteria. (optional)

We recommend you fill out as many of the following optional filters as possible to narrow the genotype set to those you are most interested in.

Variant Name(s)

architecto
molestias
magni

A list of variant names you wish to see genotypes for with one variant per line.

Project Name

The name of the project you want to restrict genotypes to.

Variant Type

– Choose One to Filter –

The types of variants you would like to see genotypes for (e.g. indels only).

Marker Type

– Choose One to Filter –

The types of markers you would like to see genotypes for (e.g. exome capture).

Search

Total Results?: 3

Unique Variants?: 3

Download: CSV, HAPMAP

Sort by Location, **Variant Name**

Variant Name	Backbone	Start	End	placeat libero cupiditate et	omnis fuga molestiae et	aut ea doloremque dignissimos	incidunt eaque quasi quibusdam
architecto	non	181	181	AT	TT	AG	GG
magni	non	206	206	AT	TC	AG	GG
molestias	non	199	199	AA	CC	TT	TT

« first 100 « previous **non: 180-199** next › last 100 »

CHAPTER 5

Data Storage

Genotypic data is stored through use of a custom table (genotype_call) created by this module. This table provides a centralized, relational table which pulls all the information for a given genotypic call (marker assay result on a given germplasm for a specific project) together in a single record. It also supports flexible storage for all metadata associated with a genotype assay result through a PostgreSQL JSONB metadata column. We went with this backwards compatible approach to make supporting large genotypic datasets more efficient then chado alone. For more information on our schema and the reasons we went with this approach see *our schema documentation*.

Note: Easy Data loading is available via the Genotypes Loader which supports VCF files!

5.1 Chado Schema and Extensions

All of the tools provided by this module retrieve their data from two question-agnostic materialized views. This provides a significant performance boost, as well as supports flexibility in the ways you can store your data.

There are currently two ways to store your genotypic data in Chado v1.3 with this module providing a third, more efficient way. While this module only supports Method #2, it can easily support data stored using the other two methods via custom queries that populate the materialized views with your data. You can see a comparison of the various methods below which should make it clear why we've gone with the storage method we have. Furthermore, you can see benchmarking for Method #2 here: https://github.com/UofS-Pulse-Binfo/nd_genotypes/wiki/Benchmarking.

5.1.1 Comparison of Methods

Method	Name	Custom Tables	Supports Metadata	# Tables	Comments
1	ND Experiment	No	Yes	14	Not suitable beyond 3 million genotype calls.
2	Genotype Call	Yes	Yes	10	Most efficient; although it touches the same number of tables as Method #3 there are less records per genotype call
3	Stock Genotype	No	No	10	A good alternative if you don't want to use custom tables but have a lot of data. Similar efficiency to Method #2 but less support for meta-data.

All three methods store Markers & Variants in the same way. For the purposes of this module, a variant is a location on the genome where variation has been detected and has a type of SNP, MNP, Indel, etc. A marker then indicates which method the genotype calls associated with it were determined by. For example, you may have a variant on Chromosome 1 at position 45678 that you detected variation through two different methods. Each method would be indicated as a marker and all the genotype calls detected by that method would be attached to the appropriate marker and not directly to the variant. This has been determined necessary since the level of trust and how you interpret any quality meta-data will depend on the method.

5.1.2 Method 1: The Chado Natural Diversity Experiment Tables.

This is the first method that was supported and the only method supported the for the 1.x versions of this module.

To try to give you an idea of the records needed we will consider a single line in a VCF file where there are only three alleles and six stocks:

# Records	Tables	Example	Explanation
2	feature	"LcChr1p555" and "LcChr1p555 GBS Marker"	One each for variant and marker where the variant may already exist.
2	featureloc	Chr1:554-555 for each.	Locate each of the variant and marker on the chromsome.
1	feature_relationship	"LcChr1p555 GBS Marker" is_marker_of "LcChr1p555"	Link the marker and variant.
6	genotype, feature_genotype	"AA", "AC", "CC"	One genotype record per unique allele call. NOTE: the allele call must be unique to the marker in order to be able to trace from marker to stock. Thus there will be a record for "AA" for marker5 and a separate record for "AA" for marker9.
18	nd_experiment_genotype, nd_experiment, nd_experiment_stock	Allele, Foreign Keys	Three records per stock in order to link the stock to it's allele through through the natural diversity tables.
6	nd_experiment_project	Again Foreign Keys	One per nd_experiment to link it to the project. Note there will be one nd_experiment per stock/marker combination.

Total: 35 records per line in a VCF with only 6 stocks and 3 alleles per variant.

Thus if your VCF file has 100,000 lines you will have to create 3,500,000 records across 12 tables to store it. Keep in mind that number doesn't include the records for your chromosomes or for your stocks since the first likely already exists and the second is only entered once per file.

5.1.3 Method 2: Custom Genotype Call Table.

Now, lets consider the same example as in Method 1 (one VCF line with three alleles and six samples):

# Records	Tables	Example	Explanation
2	feature	"LcChr1p555" and "LcChr1p555 GBS Marker"	One each for variant and marker where the variant may already exist.
2	featureloc	Chr1:554-555 for each.	Locate each of the variant and marker on the chromsome.
1	feature_relationship	"LcChr1p555 GBS Marker" is_marker_of "LcChr1p555"	Link the marker and variant.
6	genotype_call	All Foreign Keys with the exception of any quality information you want to store in the meta-data column	This links the marker, variant, allele call, stock and project all in one and stores any addition quality information in the meta-data column.

Total: 11 records per line in a VCF with only 6 stocks and 3 alleles per variant.

Notice how much more efficient this method is. This is because (1) most of the foreign key connections are taking place in a single table (genotype_call) and (2) there now only needs to be a single record in the genotype table for "AA" rather than one record per marker using the previous method. For further comparison, the same 100,000 line VCF file would now only take 1,100,000 records to store not including the records for your chromosomes, which already exist, those for your stocks, only 6 per file, and those for you alleles (genotype table), which likely already exist. Furthermore, storing meta-data doesn't increase the number of records like it would in the first method.

5.1.4 Method 3: via Stock Genotype Table.

Finally, lets consider the last method using the same example (one VCF line with three alleles and six samples):

# Records	Tables	Example	Explanation
2	feature	“LcChr1p555” and “LcChr1p555 GBS Marker”	One each for variant and marker where the variant may already exist.
2 1	feature-loc feature_relationship	Chr1:554-555 for each. “LcChr1p555 GBS Marker” is_marker_of “LcChr1p555”	Locate each of the variant and marker on the chromsome. Link the marker and variant.
6	genotype, feature_genotype	“AA”, “AC”, “CC”	One genotype record per unique allele call. NOTE: the allele call must be unique to the marker in order to be able to trace from marker to stock. Thus there will be a record for “AA” for marker5 and a separate record for “AA” for marker9.
6	stock_genotype	All Foreign Keys	Link each DNA stock to the allele detected using the assay. We are only counting the linking records here since the stocks will only be created once per file.

Total: 17 records per line in a VCF with only 6 stocks and 3 alleles per variant.

This is a good mid-range option that allows you to store genotypes efficiently without the use of any custom tables! The trade-off is that there isn’t a good way to store meta-data related to the assay such as read depth. To complete the comparison, the same 100,000 line VCF file would take 1,700,000 records to store not including the records for your chromosomes, which already exist, those for your stocks, only 6 per file.

5.2 Example Database

The following queries endeavour to show how data used by this module is stored. This is a small peak into a production database and while it’s not perfect (still containing some legacy terms, etc.) it is completely functional with the nd_genotypes module.

5.2.1 Markers & Variants

The following queries show how markers and variants are stored. The types used for markers and variants can be configured and more then one type can be used for each (e.g. you could use SNP, MNP, Indel types for variants). While the example below shows multiple types for variants, in the future my personal database will be switched to use the SO sequence_variant type for all variants to aid with consistent variant pages in Tripal 3. However, this is a personal choice and both methods have their pro’s and cons.

```
psql=# SELECT f.*, cvt.name as type_name FROM chado.feature f LEFT JOIN chado.cvterm␣
↪cvt ON cvt.cvterm_id=f.type_id WHERE f.name~'LcC09269p298';
 feature_id | dbxref_id | organism_id |                 name                  |      ␣
↪   uniquename           | residues | seqlen | md5checksum | type_id | is_analysis |␣
↪is_obsolete |      timeaccessioned       |      timelastmodified       |   type_name
------------+-----------+-------------+---------------------------------------+-------
↪-----------------------+----------+--------+-------------+---------+-------------+-
↪-----------+----------------------------+----------------------------+-------------
↪---
    327991 |   2513464 |           4 | LcC09269p298                          |␣
↪LcC09269p298                  |          |      1 |             |     796 | f       ␣
↪    | f           | 2011-07-29 16:08:43.515889 | 2011-07-29 16:08:43.515889 | SNP
    372934 |   2649322 |           4 | LcC09269p298 454 Sequencing           |␣
↪LcC09269p298_454              |          |      1 |             |    3969 | f       ␣
↪    | f           | 2011-09-15 11:52:45.943205 | 2011-09-15 11:52:45.943205 |␣
↪genetic_marker
```

(continues on next page)

(continued from previous page)

```
     392501 |    3114923 |           4 | LcC09269p298 Lc1536 Golden Gate Assay |␣
↪LcC09269p298-1_B_F_1890446698 |          |      1 |             |    3969 | f      ␣
↪    | f            | 2011-09-15 12:06:20.86547  | 2011-09-15 12:06:20.86547  |␣
↪genetic_marker
(3 rows)
```

```
psql=# SELECT prop.*, cvt.name as type_name FROM chado.featureprop prop LEFT JOIN␣
↪chado.cvterm cvt ON cvt.cvterm_id=prop.type_id WHERE prop.feature_id IN (327991,␣
↪372934, 392501);
 featureprop_id | feature_id | type_id |           value            | rank |         ␣
↪type_name
----------------+------------+---------+----------------------------+------+----------
↪-------------------
         400633 |     327991 |    1512 | 91 bp                      |    0 | five_
↪prime_flanking_region
         400634 |     327991 |    1513 | 308 bp                     |    0 | three_
↪prime_flanking_region
         525105 |     372934 |    3966 | 454 Sequencing             |    0 | marker_
↪type
         459336 |     392501 |    1891 | 0.909                      |    0 | score
         459337 |     392501 |    1870 | LcRedberry                 |    0 | source
         459338 |     392501 |    3687 | 12/23/2010                 |    0 | design_
↪date
         466357 |     392501 |    3709 | BOT                        |    0 | illumina_
↪strand
         466358 |     392501 |    3710 | BOT                        |    0 |␣
↪reference_sequence_strand
         781915 |     392501 |    3966 | Illumina Golden Gate Assay |    0 | marker_
↪type
(9 rows)
```

```
psql=# SELECT t.* FROM chado.featureloc t WHERE t.feature_id IN (327991, 372934,␣
↪392501);
 featureloc_id | feature_id | srcfeature_id |   fmin    | is_fmin_partial |   fmax    ␣
↪| is_fmax_partial | strand | phase | residue_info | locgroup | rank
---------------+------------+---------------+-----------+-----------------+-----------
↪+-----------------+--------+-------+--------------+----------+------
       3897843 |     372934 |        295264 |       297 | f               |       298␣
↪| f               |      0 |     0 |              |        0 |    0
       3711470 |     392501 |        295264 |       297 | f               |       298␣
↪| f               |      0 |     0 |              |        0 |    0
       3260896 |     327991 |        295264 |       297 | f               |       298␣
↪| f               |        |       |              |        0 |    0
       4562009 |     327991 |       3400411 | 250519947 | f               | 250519948␣
↪| f               |     -1 |       |              |        2 |    0
       4562010 |     327991 |       3400411 | 250136623 | f               | 250136624␣
↪| f               |     -1 |       |              |        2 |    1
       4562011 |     327991 |       3400407 |    501710 | f               |    501711␣
↪| f               |     -1 |       |              |        2 |    2
       4628689 |     372934 |       3400411 | 250519947 | f               | 250519948␣
↪| f               |     -1 |       |              |        2 |    0
       4628690 |     372934 |       3400411 | 250136623 | f               | 250136624␣
↪| f               |     -1 |       |              |        2 |    1
       4628691 |     372934 |       3400407 |    501710 | f               |    501711␣
↪| f               |     -1 |       |              |        2 |    2
(9 rows)
```

```
psql=# SELECT t.*, cvt.name as type_name FROM chado.feature_relationship t LEFT JOIN␣
↪chado.cvterm cvt ON cvt.cvterm_id=t.type_id WHERE t.subject_id IN (327991, 372934,␣
↪392501);
 feature_relationship_id | subject_id | object_id | type_id | value | rank |  type_
↪name
-------------------------+------------+-----------+---------+-------+------+----------
↪----
                 2575387 |     372934 |    327991 |    3685 |       |    0 | is_
↪marker_of
                 2594954 |     392501 |    327991 |    3685 |       |    0 | is_
↪marker_of
(2 rows)
```

5.2.2 Genotypes

The preferred method of storing genotype calls is to use the new genotype_call table created by this module as it is more efficient. As you can see below this results in each unique allele only being stored once in the genotype table with the information of which allele was detected for a given marker/stock combination is recorded in the genotype_call table. This method doesn't use the feature_genotype table.

```
psql=# SELECT t.*, cvt.name as type_name FROM chado.feature_genotype t LEFT JOIN␣
↪chado.cvterm cvt ON cvt.cvterm_id=t.cvterm_id WHERE t.feature_id IN (327991, 372934,
↪ 392501);
 feature_genotype_id | feature_id | genotype_id | chromosome_id | rank | cgroup |␣
↪cvterm_id | type_name
---------------------+------------+-------------+---------------+------+--------+-----
↪------+-----------
(0 rows)
```

```
psql=# SELECT * FROM chado.genotype_call WHERE variant_id=327991 LIMIT 10;
 genotype_call_id | variant_id | marker_id | genotype_id | project_id | stock_id |␣
↪meta_data
------------------+------------+-----------+-------------+------------+----------+----
↪-------
           158529 |     327991 |    372934 |     2625650 |          3 |    27907 |
           158530 |     327991 |    372934 |     2625649 |          3 |    27908 |
           158531 |     327991 |    372934 |     2625649 |          3 |    27911 |
           324755 |     327991 |    372934 |     2625650 |          3 |    27916 |
           324756 |     327991 |    372934 |     2625650 |          3 |    27917 |
           616977 |     327991 |    392501 |     2625652 |         36 |    28283 |
           618223 |     327991 |    392501 |     2625652 |         36 |    28284 |
           619485 |     327991 |    392501 |     2625651 |         36 |    28285 |
           620644 |     327991 |    392501 |     2625651 |         36 |    28286 |
           621871 |     327991 |    392501 |     2625652 |         36 |    28287 |
(10 rows)
```

```
psql=# SELECT g.*, cvt.name as type_name FROM chado.genotype g LEFT JOIN chado.cvterm␣
↪cvt ON cvt.cvterm_id=g.type_id;
 genotype_id | name | uniquename | description | type_id | type_name
-------------+------+------------+-------------+---------+-----------
     2625647 | A    | A          | A           |     796 | SNP
     2625648 | T    | T          | T           |     796 | SNP
     2625649 | C    | C          | C           |     796 | SNP
     2625650 | G    | G          | G           |     796 | SNP
     2625651 | GG   | GG         | GG          |     796 | SNP
```

(continues on next page)

(continued from previous page)

```
     2625652 | CC    | CC           | CC            |      796 | SNP
     2625653 | TT    | TT           | TT            |      796 | SNP
     2625654 | AA    | AA           | AA            |      796 | SNP
(8 rows)
```

5.2.3 Germplasm/Stocks

The DNA source the marker assay was performed on is given a type of DNA with the original germplasm source of this DNA having whichever term is appropriate. The important thing is that the DNA extraction and original germplasm are related consistently through the stock_relationship table.

```
psql=# SELECT s.*, cvt.name as type_name FROM chado.stock s LEFT JOIN chado.cvterm␣
↪cvt ON cvt.cvterm_id=s.type_id WHERE s.stock_id IN (58, 27907);
  stock_id | dbxref_id | organism_id |                    name                      ␣
↪|                uniquename               | description | type_id | is_obsolete |␣
↪type_name
----------+-----------+-------------+-----------------------------------------------+-
↪---------------------------------------+-------------+---------+-------------+------
↪------
       58 |   1901662 |           4 | CDC Redberry                                  |␣
↪KP:GERM58                              |             |    3683 | f           |␣
↪Variety
    27907 |           |           4 | CDC Redberry 454 Extraction                   |␣
↪CDC_Redberry_454                       |             |    3630 | f           | DNA
```

```
psql=# SELECT t.*, cvt.name as type_name FROM chado.stock_relationship t LEFT JOIN␣
↪chado.cvterm cvt ON cvt.cvterm_id=t.type_id WHERE t.subject_id IN (58, 27907) AND␣
↪cvt.name='is_extracted_from';
 stock_relationship_id | subject_id | object_id | type_id | value | rank |     type_
↪name
-----------------------+------------+-----------+---------+-------+------+------------
↪-------
                 43301 |      27907 |        58 |    3712 |       |    0 | is_
↪extracted_from
(1 row)
```

5.2.4 Materialized Views

The following queries show the materialized views created by this module and provide an example of what they should contain. Notice that the variant/markers being demonstrated are located in multiple places on the genotype which explains the multiple records in mview_ndg_lens_variants. If your variants amplify unique regions then there will only be one location per variant in this table.

```
psql=# SELECT * FROM chado.mview_ndg_lens_calls WHERE variant_id=327991 LIMIT 10;
 variant_id | marker_id |              marker_name              |         marker_type ␣
↪       | stock_id |          stock_name          | germplasm_id | germplasm_name |␣
↪project_id | genotype_id | allele_call | meta_data | ndg_call_id
------------+-----------+---------------------------------------+---------------------
↪-------+----------+-----------------------------+--------------+-----------------+---
↪---------+-------------+-------------+-----------+-------------
     327991 |    372934 | LcC09269p298 454 Sequencing           | 454 Sequencing      ␣
↪       |    27908 | 964a-46 454 Extraction      |         6755 | 964a-46        |␣
↪       3 |     2625649 | C           |           |     1223711
```

(continues on next page)

(continued from previous page)

```
     327991 |     372934 | LcC09269p298 454 Sequencing           | 454 Sequencing     ␣
↪        |    27911 | ILL 8006 454 Extraction     |        18809 | ILL 8006       |  ␣
↪        3 |     2625649 | C            |            |     1223712
     327991 |     372934 | LcC09269p298 454 Sequencing           | 454 Sequencing     ␣
↪        |    27907 | CDC Redberry 454 Extraction |           58 | CDC Redberry   |  ␣
↪        3 |     2625650 | G            |            |     1309137
     327991 |     372934 | LcC09269p298 454 Sequencing           | 454 Sequencing     ␣
↪        |    27916 | PI 320937 454 Extraction    |         7832 | PI 320937      |  ␣
↪        3 |     2625650 | G            |            |     1347692
     327991 |     372934 | LcC09269p298 454 Sequencing           | 454 Sequencing     ␣
↪        |    27917 | L01-827A 454 Extraction     |         9727 | L01-827A       |  ␣
↪        3 |     2625650 | G            |            |     1347693
     327991 |     392501 | LcC09269p298 Lc1536 Golden Gate Assay | Illumina Golden␣
↪Gate Assay |     28285 | 1294M-23 Extraction         |         9420 | 1294M-23         ␣
↪|          36 |      2625651 | GG           |            |     1357149
     327991 |     392501 | LcC09269p298 Lc1536 Golden Gate Assay | Illumina Golden␣
↪Gate Assay |     28286 | 2670B Extraction            |         9975 | 2670B            ␣
↪|          36 |      2625651 | GG           |            |     1357418
     327991 |     392501 | LcC09269p298 Lc1536 Golden Gate Assay | Illumina Golden␣
↪Gate Assay |     28288 | 964a-46 Extraction          |         6755 | 964a-46          ␣
↪|          36 |      2625651 | GG           |            |     1357955
     327991 |     392501 | LcC09269p298 Lc1536 Golden Gate Assay | Illumina Golden␣
↪Gate Assay |     28289 | Giftgi Extraction           |         9771 | Giftgi           ␣
↪|          36 |      2625651 | GG           |            |     1358196
     327991 |     392501 | LcC09269p298 Lc1536 Golden Gate Assay | Illumina Golden␣
↪Gate Assay |     28290 | ILL 1704 Extraction         |         8111 | ILL 1704         ␣
↪|          36 |      2625651 | GG           |            |     1358495
(10 rows)

psql=# SELECT * FROM chado.mview_ndg_lens_variants WHERE variant_id=327991;
 variant_id | variant_name | variant_type | srcfeature_id | srcfeature_name |   fmin ␣
↪   |   fmax     |                             meta_data                          ␣
↪| ndg_variants_id
------------+--------------+--------------+---------------+-----------------+---------
↪--+-----------+-----------------------------------------------------------------
↪+-----------------
     327991 | LcC09269p298 | SNP          |        295264 | LcRBContig09269 |       ␣
↪297 |       298 | {"strand": null, "featureloc_id": 3260896, "variant_type_id": 796}
↪ |          396318
     327991 | LcC09269p298 | SNP          |       3400407 | LcChr1          |    ␣
↪501710 |     501711 | {"strand": -1, "featureloc_id": 4562011, "variant_type_id":␣
↪796}   |          396319
     327991 | LcC09269p298 | SNP          |       3400411 | LcChr5          |␣
↪250136623 | 250136624 | {"strand": -1, "featureloc_id": 4562010, "variant_type_id":␣
↪796}   |          396320
     327991 | LcC09269p298 | SNP          |       3400411 | LcChr5          |␣
↪250519947 | 250519948 | {"strand": -1, "featureloc_id": 4562009, "variant_type_id":␣
↪796}   |          396321
(4 rows)
```

CHAPTER 6

Contributing

We're excited to work with you! Post in the issues queue with any questions, feature requests, or proposals.

6.1 Automated Testing

This module uses Tripal Test Suite. To run tests locally:

```
cd MODULE_ROOT
composer up
./vendor/bin/phpunit
```

This will run all tests associated with the ND Genotypes extension module. If you are running into issues, this is a good way to rule out a system incompatibility.

> **Warning:** It is highly suggested you ONLY RUN TESTS ON DEVELOPMENT SITES. We have done our best to ensure that our tests clean up after themselves; however, we do not guarantee there will be no changes to your database.

6.2 Manual Testing (Demonstration)

We have provided a Tripal Test Suite Database Seeder to make development and demonstration of functionality easier. To populate your development database with fake phenotypic data:

1. Install this module according to the instructions in the administration guide.
2. Create an organism (genus: Tripalus; species: databasica)
3. Run the database seeder to populate the database using the following commands:

```
cd MODULE_ROOT
composer up
./vendor/bin/tripaltest db:seed GenotypeDatasetSeeder
```

4. Populate the materialized views by going to Administration » Tripal » Extensions » Natural Diversity Genotypes » Sync and Choose “Tripalus” then click the “Sync” button. Finally run the Tripal jobs submitted.
5. To play with the genotype matrix go to `[your drupal site]/chado/genotype/[genus]`. You can see what germplasm are available by typing a single random letter in the autocomplete box.
6. To play with marker/variant pages, go to Administration » Content » Tripal Content » Publish Tripal Content and then select “Genetic Marker”/”Sequence Variant” and publish to create pages. Remember to run the tripal jobs submitted on the command-line using Drush `trp-job-run`.

Warning: NEVER run database seeders on production sites. They will insert fictitious data into Chado.

Warning: If ND Genotypes fields are not automatically attached to the genetic marker and sequence variant content types, go to the “Manage Fields” page for each and click “Find new fields”. Also, go to the “Manage Display” page and ensure they are not hidden.