Ensembl Fungi, JGI and FungiDB: Community-driven manual gene curation

1.0 Introduction

With easy access to affordable sequencing technologies, the volume of genomic data (particularly, for microbial species) has grown exponentially. A majority of these undergo automated gene annotation before going into public circulation. Despite advances in gene prediction algorithms, they still cannot automatically resolve all complexities surrounding the precise location and structure of the genome elements. It is not uncommon, therefore, for there to be significant differences in the quality of these automated gene annotations compared to carefully manually curated gene sets. It is also the case that for certain pathogens there are different gene sets used by communities owing to preferred tools and protocols. Given that a high-quality, unified gene set is the key to enabling further inferences, this is an important problem to address.

2.0 Approaches to manual gene model curation

Although beneficial, manual gene curation is a laborious undertaking and often unfunded except for model species. There are several approaches to doing manual gene curation ranging from dedicated teams looking deeply at gene structures to enabling Wikipedia-style community editing.

In this course, we would like to draw your attention to approaches adopted by the Joint Genome Institute (JGI), Ensembl Fungi and FungiDB to capture changes to existing gene models. You can find detailed instructions on how to use these platforms in appendices at the end of this tutorial.

2.1 JGI MycoCosm: Gene models open to editing by collaborating scientists

The JGI Fungal annotation pipeline uses several gene prediction algorithms, including ab-initio, homology, and EST-based gene modelers to produce multiple overlapping gene models for a given locus. A heuristic filtering process chooses the “best” model at each locus according to specific weights given to each model based on evidence, completeness, homology, presence of known domains and structures. These filtered models are stored in the “FilteredModels” track on the JGI browser. A copy of the FilteredModels is stored as the GeneCatalog. Users with specific privileges (collaborating scientists) can modify, add and remove models from the GeneCatalog using available manual curation tools. These corrected gene models eventually become the reference list of gene models for this organism. More on how to use JGI’s MycoCosm platform can be found in Appendix A.

2.2 Ensembl Fungi: Annotation projects with species-specific communities

Ensembl Fungi facilitates collaborations between research groups interested in the same species to work together to redefine the *de facto* gene set. The process usually begins with an interested community approaching Ensembl Fungi (or vice versa). Ensembl Fungi sets up an Apollo instance, provides training and user support and collates evidence from this community necessary for the curation into Apollo tracks.

Apollo (http://genomearchitect.github.io/) is a web-based collaborative gene-editing plugin for the JBrowse genome viewer (doi.org/10.1371/journal.pcbi.1006790). With it, a team of researchers spread across the globe can work on the same sequences at the same time using just their internet browser.

Once the curation effort is complete, the gene sets undergo a QC process and get integrated into Ensembl Fungi. Ensembl Fungi also helps distribute the work across the members, typically generating gene lists or chromosome regions for each user to examine. All the gene models in a typical fungal genome can be manually curated in months using this approach. For instance, a recent curation project for *Botrytis cinerea* involved close to 50 members spread across several countries and took around six months, resulting in a completely revised gene set. A similar project was completed for *Blumeria graminis* and one is underway for *Zymoseptoria tritici.* More about this process can be found in Ensembl Fungi's publication at doi:10.3389/fmicb.2019.02477.

We have included remarks below from two participants of community annotation projects: From Dr Jan van Kan (Wageningen University, Netherlands) who lead the gene editing effort for *Botytris cinerea*:

> *"After annotating several thousands of genes, I can tell you it is fun and not (always) difficult. In fact, in most cases the situation is obvious and simple. For most of the genes, you will easily be able to verify whether the exon structure is OK, and you will be able to adjust the UTRs to a reasonable position."*

From Dr Alice Feurtey (Max Planck Institute) involved in the *Zymoseptoria tritici* curation project highlighting the need for common guidelines to standardise the annotation process:

> *"We need common, detailed guidelines that annotators can follow. In our review of the test manual annotations, we have found that different people make different decisions in similar scenarios."*

2.3 FungiDB: Curation open to any account holder for many species

FungiDB allows account holders to make structural and functional changes to gene models (using Apollo) across all organisms integrated into FungiDB. This is also enabled for other species in VEuPathDB, including AmoebaDB, CryptoDB, PlasmoDB, PiroplasmaDB, VectorBase and ToxoDB. Users need an account (which is free) to log into Apollo. All changes made in Apollo are collated and are subject to automated and manual QC checks

before integration into the gene set. All user-submitted tracks become available on the live site and are visible on gene records pages and in the genome browser JBrowse. Specific instructions on how to use the Apollo interface through FungiDB/VEuPathDB are in Appendix C.

3.0 Data to support manual gene curation

All manual gene curation approaches make use of additional 'evidence' to help curators make decisions about gene models. Some supporting data types are listed below - not all of them will exist for all species. These are often added as "tracks" to the editing interface such as Apollo and appear alongside the gene models.

1. Transcriptome data
    a. RNA-Seq coverage (BigWig files)
    b. RNA-Seq read mapping (BAM files)
    c. PacBio Iso-Seq data
    d. Expressed Sequence Tag (EST) and cDNA data
2. Short read intron data
3. Homology and conservation data
4. Polyadenylation site data (polyA seq)
5. Protein sequences
6. Long read and short read sequencing data
7. Other gene sets available for the species
8. Alignments to gene sets and proteins of *closely related* species

4.0 Structural curation of gene models

Most of the discussions in this tutorial focus on **structural curation.** This can involve the following:

- Verifying the accuracy of **splice junctions**
- Deciding whether (biologically meaningful) **splice variants** can be detected
- Adjusting the transcription start sites (**5'-UTR**) and polyadenylation sites (**3'-UTR**) to the most likely position
- Adjusting transcript boundaries
- Splitting gene models or merging fragments
- Adding new genes
- Removing incorrect gene models (false positives)
- Choosing the best from a proposed set of gene models for a given locus

4.1 Examples of structural gene curation

This section contains examples and suggestions for the structural curation tasks listed above. These should be treated as broad guidelines as there will inevitably be subtle species-specific/community-specific differences in the ways that supporting data is interpreted. The screenshots below show Apollo displaying either *Botrytis cinerea* or *Zymoseptoria tritici* data from the group curation projects conducted by Ensembl Fungi. See Appendix B for a description of the Apollo interface.

4.1.1 Verifying the accuracy of splice junctions

Errors in splice junction prediction can occur when the splice donor site is GC instead of GT; this non-canonical splice site is not easily detected by gene prediction tools. Occasionally you will see a GT splice junction in a predicted gene, whereas a GC junction nearby is the correct one. Another error that sometimes occurs is the wrong prediction of the translation start site if an in-frame start codon is nearby. Sometimes the gene prediction then calls the second ATG as the likely start codon, while the first (upstream) ATG codon is correct.

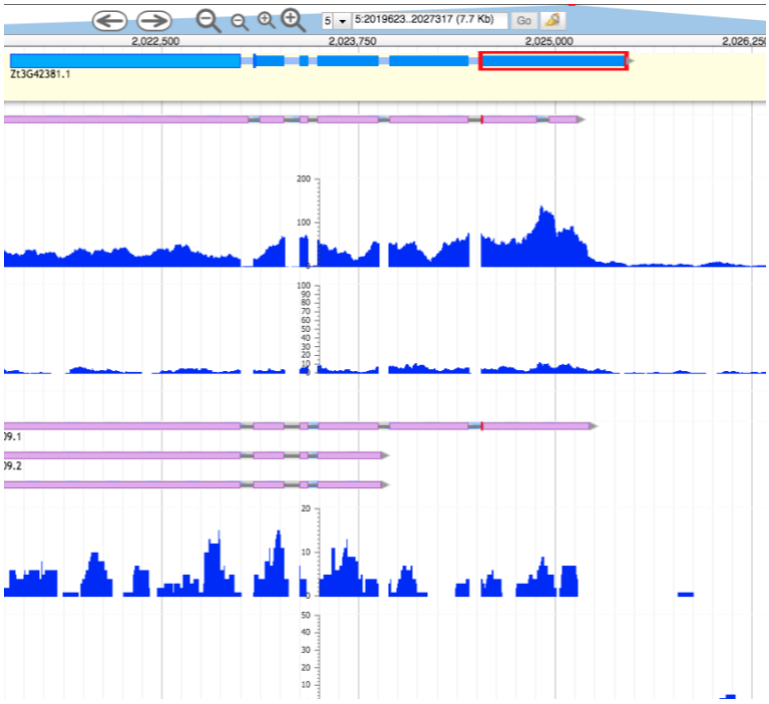


Figure 1 - No evidence to support last intron in the gene model in pink at the top

To assist in the decision to modify a splice site, you can also download the translated sequences (menu option available in Apollo) and use them to search well-curated protein databases, such as UniProt, to see if you can resolve the question using protein alignments. Incorrect splice sites would likely cause gaps in the alignments. Keep in mind that the best alignment may be the exact prediction from which you initiated your annotation; you should not consider the identical protein from your organism as external evidence supporting the annotation. Instead, look at alignments to proteins from other organisms. If there does not appear to be any way to resolve the non-canonical splice, leave it as is and add a comment.

4.1.2 Detecting new splice variants

Another important thing to look out for is splice variants. Sometimes you can see from the read mapping track that in a certain region, a proportion of reads is spliced whereas another proportion is unspliced. This is not always biologically relevant, such as below:

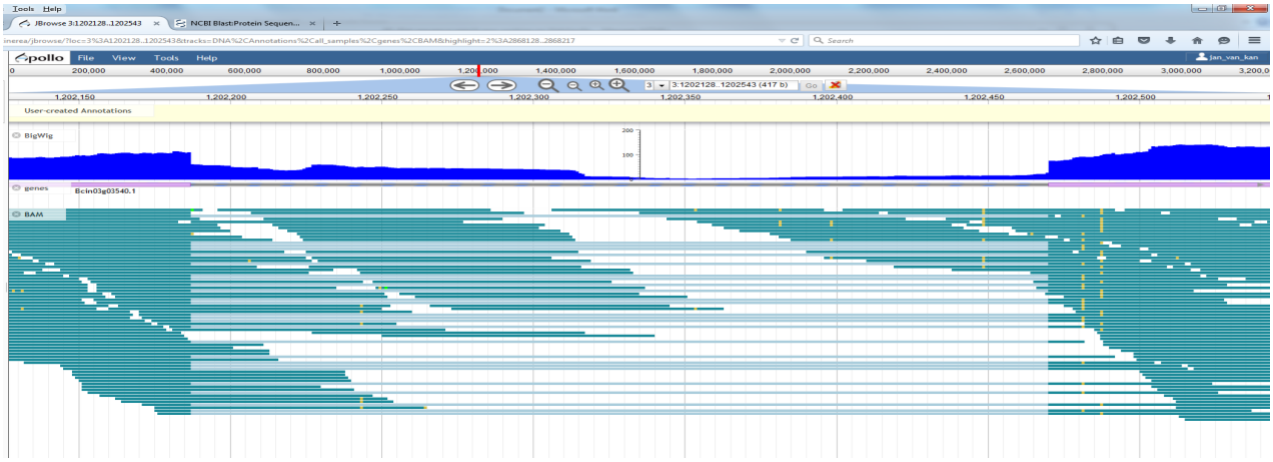


Figure 2 - Read splicing does not always point to a new, biologically meaningful splice event

The reads that map across the intron above are most likely derived from unspliced or partially spliced RNAs. If the intron is not spliced, merging the two flanking exons leads to frameshifts and stop codons, and would never result in a functional protein. The proportion of unspliced or partially spliced RNAs can in some cases be high, up to 10%, as compared to the real exons. In the particular case illustrated in the figure, the *Botrytis cinerea* annotators decided that this was not alternative splicing. In other cases, alternative splicing may be meaningful, but it is up to your personal judgement to decide this as there are no general rules.

If alternative splicing occurs, and only one gene model is provided, you can duplicate the gene model (in Apollo by a double click) and make the adjustment to the splice junctions in the duplicate to create the splice variant. The splice variants must be numbered as different mRNAs (usually the mRNA gets the gene name followed by a suffix such as .1 , .2 , .3 and so on for different splice variants). An example of alternative splicing can be seen below in the hydrophobin gene Bhp1 (gene is in right to left orientation) in *Botrytis cinerea*. A small proportion of the transcripts have a shorter second intron, leading to an insertion of 13 amino acids in the third exon. The ratio of the major variant and minor variant is ~20:1; nevertheless this may be biologically meaningful.

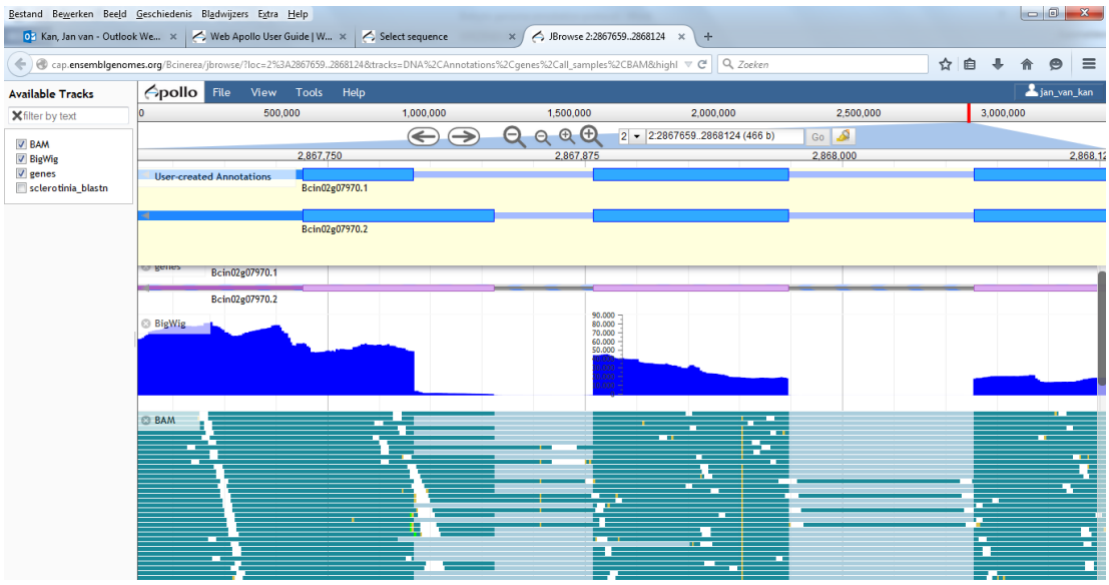


Figure 3 - Identification of a new splice variant (shown in blue at the top)

4.1.3 Adjusting untranslated regions (UTRs)

In some cases, UTRs may not have been predicted by the gene prediction software that was used and, therefore, most gene models will not have any UTR information. Even if the gene models you are looking at contain UTR information, this is still worth checking. RNA-Seq coverage can be used to identify the start and end of transcription. Sometimes the place is fairly obvious, such as in the image below. The UTRs in the figure below can be extended to the approximate positions where the BigWig coverage track reaches the baseline.

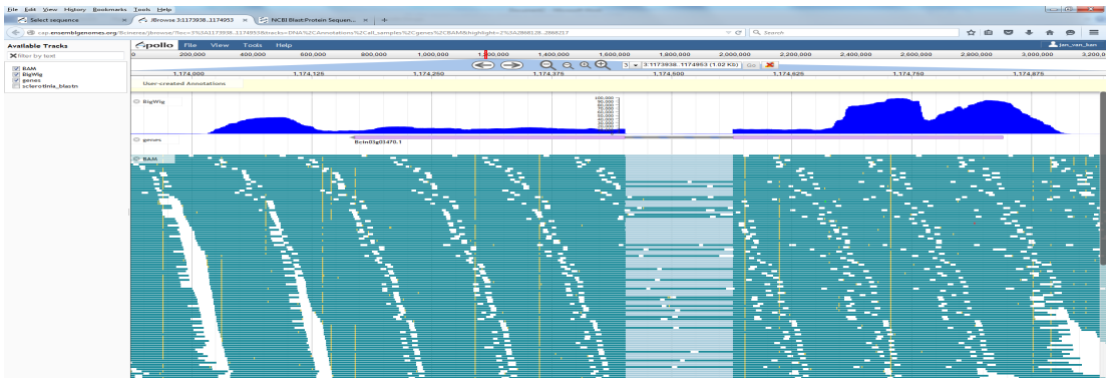


Figure 4 - The UTRs in the gene model in pink can be extended to the point where the RNA-Seq coverage reaches the baseline

Here are two rules to be considered for UTRs:

- When you adjust the 5’-UTR, make sure to always set it at a G. Many fungal transcription start sites are at the end of C/T rich regions and start with a G (preferentially with GA or GG). So, choose an appropriate G that follows a C/T rich stretch.
- A 3’-UTR ends at the site of polyadenylation, which is ~10-30 nucleotides downstream of a polyadenylation signal. In mammals, the polyadenylation signal is very easy and straightforward: AATAAA. In fungi, it is not conserved so well. Often the AAT is present, but the next three nucleotides have many options, some of which are preferred. Frequently, motifs around putative polyadenylation sites are AATAAT, AATATT, AATATC and AATACT (Gs in the last three nucleotides of the motif are less frequent). Be aware that the polyadenylation itself occurs 10-30 bases downstream, and there is a slight preference for the motif CA, where the poly(A)-tail is added to the A. That is a good place to end the 3’-UTR.

The gene model shown below is an example from the *Botrytis cinerea* annotation project. The users adjusted the 5’-UTR (gene is in inverse orientation) to a site where the RNA-Seq coverage drops <500 (the peak around the start codon has coverage ~100,000). The transcript starts with GACCTCG.

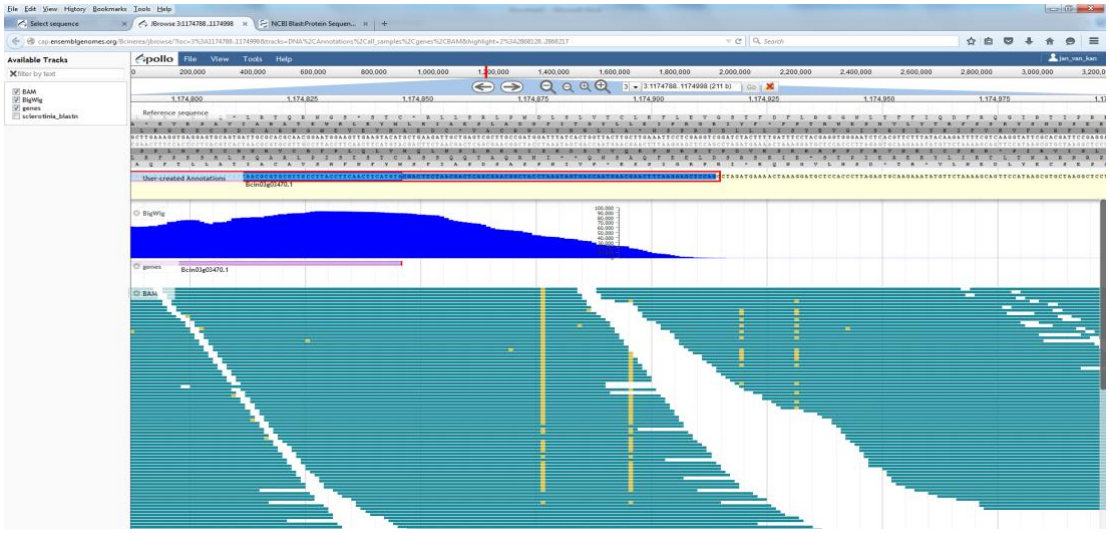


Figure 5 - 5'-UTR adjustment. The users adjusted the 5’-UTR (gene is in inverse orientation) to a site where the RNA-Seq coverage drops <500 (the peak around the start codon has coverage ~100,000). The transcript starts with GACCTCG

The 3’-UTR of the same gene was adjusted as follows: the sequence motifs AATGAT and AATCTA occur shortly after one another. About 10 nucleotides downstream there is a sequence GCTA where coverage drops to <500. This is where the curators considered was the

most likely place for the polyadenylation site and the end of the 3’-UTR. For such highly expressed genes, coverage is unlikely to drop entirely to zero.

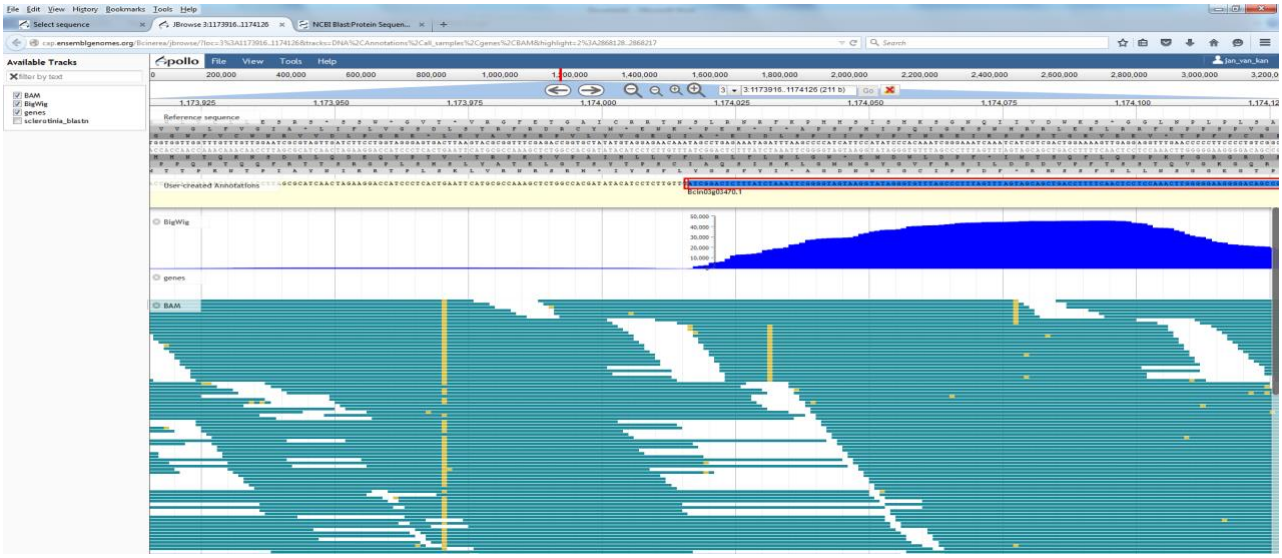


Figure 6 - 3'-UTR adjustment

4.1.4 Adjusting transcript boundaries

It is not always clear where a transcript ends, as shown below (gene going from right to left). The coding sequence ends in the middle of the image, yet there is RNA-Seq coverage all the way to the left, suggesting a 3’-UTR that would be at least 1.5 kb in length! Also, the 3’-UTR would have an intron around position 1,198,300 which seems an unlikely situation. In this case, it may be a long non-coding RNA. In this example, the *Botrytis cinerea* curators decided to end the 3’-UTR at a position where a reasonable polyadenylation consensus is detected, and the length of the 3’-UTR is in the range of 50-200 nt. It would be meaningless (and probably misleading) to extend it to 1.5 kb.

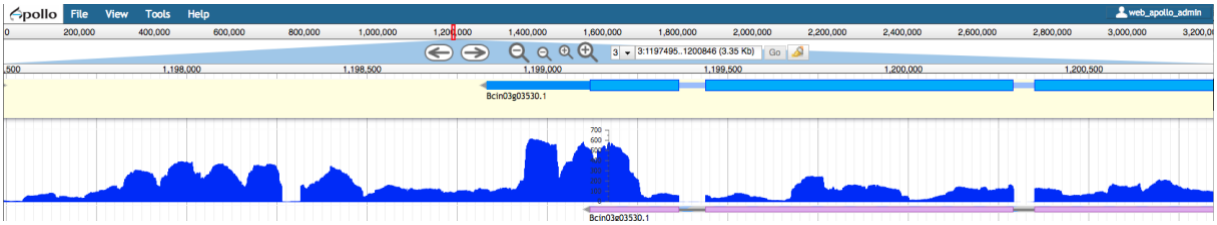


Figure 7 - Confusing transcript end (Gene Bcin03g03530 in *Botrytis cinerea*)

4.1.5 Splitting gene models and merging fragments

For neighbouring genes with short intergenic regions, it is not always obvious where one gene ends and the next gene begins (as shown in the conflicting gene annotations below). The transcripts may even overlap if the neighbouring genes are transcribed convergently towards each other making it hard to determine intron and UTR limits. Aligning stranded data (in the evidence tracks) can help resolve this.

Furthermore, for compact genomes like *Zymoseptoria tritici* data from polyA studies can greatly help clear up confusion over merged gene models and rare read throughs.

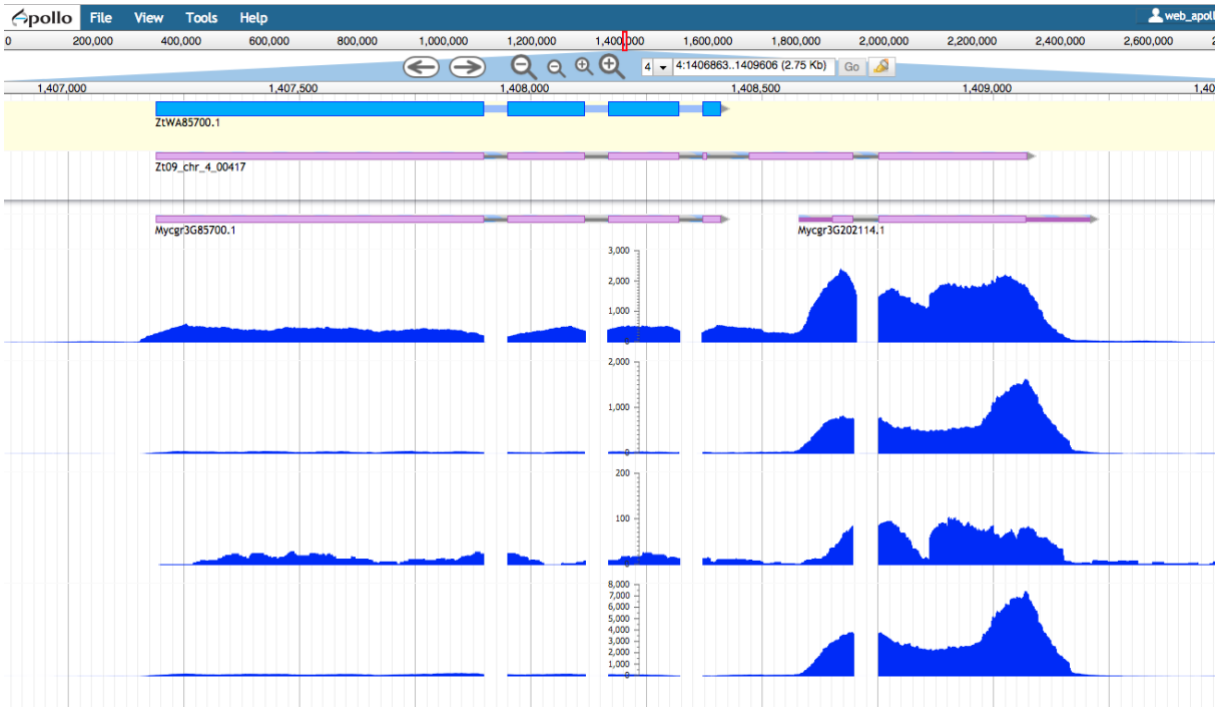


Figure 8 - Two neighbouring genes wrongly merge together in the top pink gene model. These are likely to be two genes expressed differently (shown by RNA-Seq coverage)

In contrast, sometimes gene models that have been predicted as being separate might need merging. The image below shows an example of two gene models from a predicted set being merged together based on RNA-Seq data and models of closely related species.

Figure 9: The annotation in pink shows two genes but, based on alignments to other species, the curators have decided to merge these two genes into one model (Bcin14g05150, *Botrytis cinerea*)

4.1.6 Adding a new gene

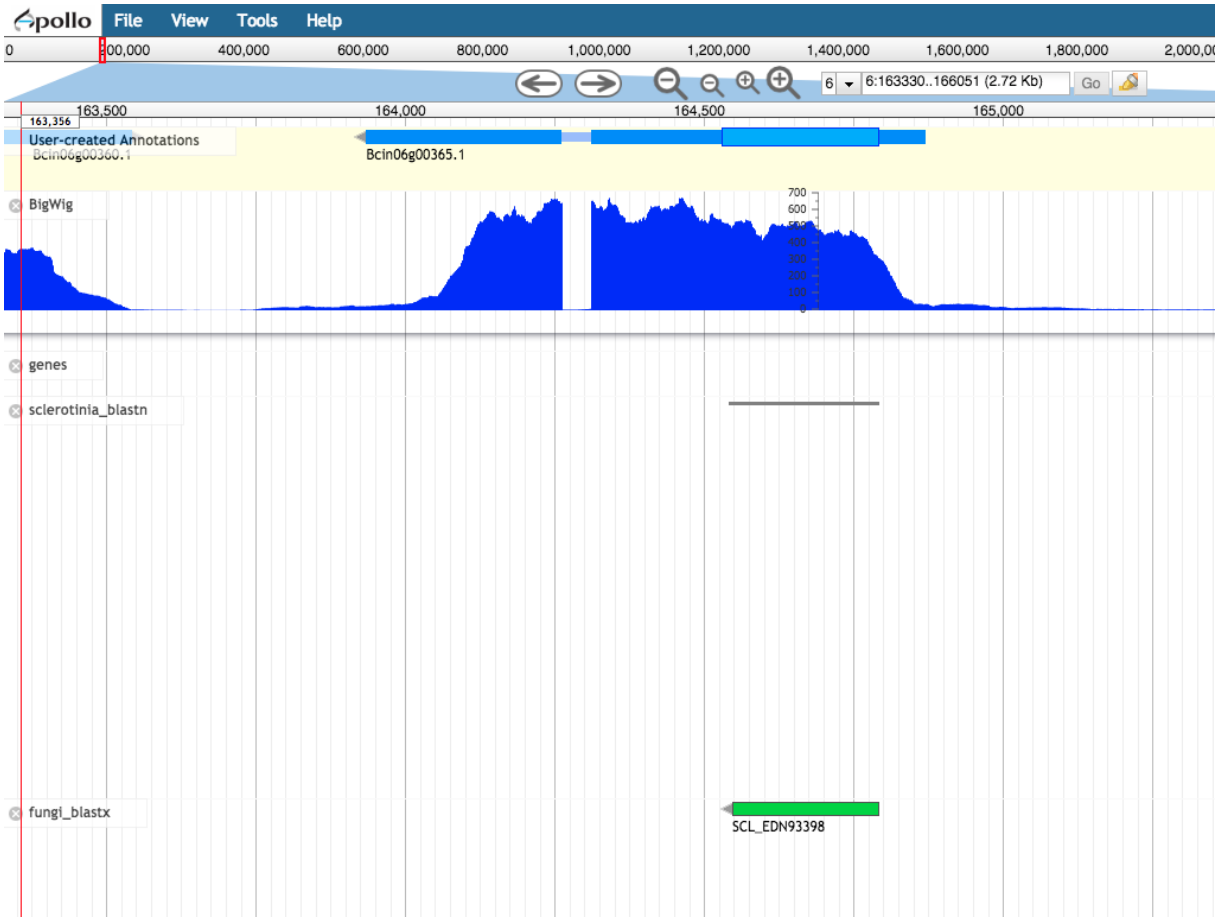


Figure 10: Adding a new gene

The screenshot above shows a new gene that was added in *Botrytis cinerea* based on RNA-Seq data *and* matches to a closely related species. See section below on the 'Absence of a gene' for situations where RNA-Seq alignments alone may not always inform the presence of a new gene.

4.1.7 Choosing the best from a proposed set of gene models

If there are multiple gene predictions for a species, it is likely that there will be conflicting genes for a given locus. Often the curator's job will be to choose the most representative and accurate model from the set as shown below.

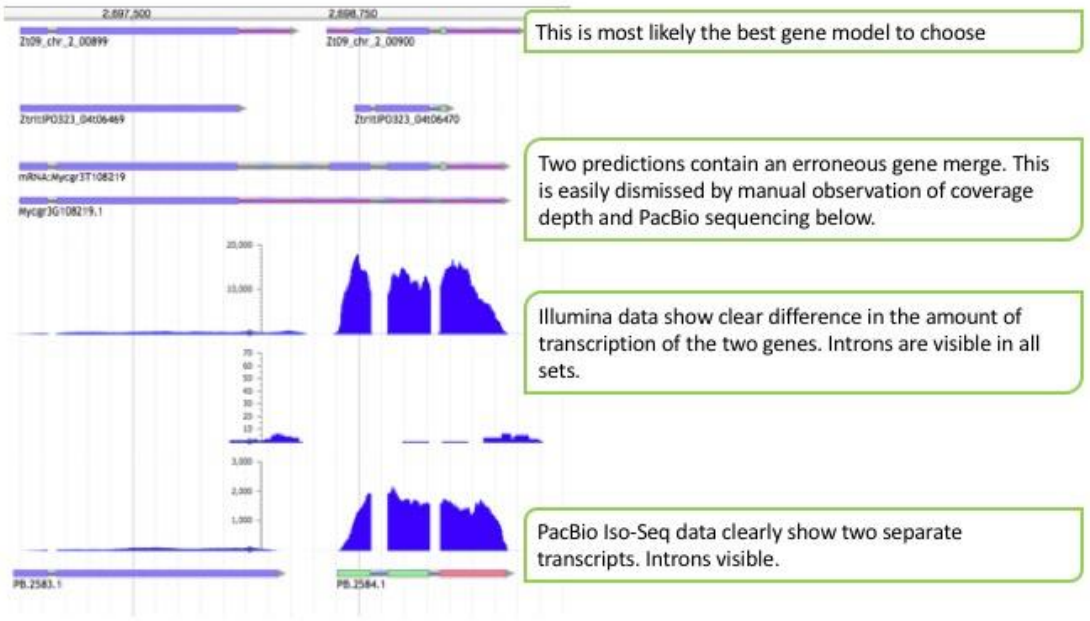


Figure 11: The figure shows four alternative gene models at the top, followed by Illumina RNA-Seq and PacBio Iso-Seq data underneath. Two of the gene predictions contain a gene model merge even though the evidence points to two separate genes. In this instance, the curators have chosen the first model as the best representation for this locus.

4.2 Other interesting observations

4.2.1 Very small exons

If you examine the picture below (*Botrytis cinerea*), you will see that the third tiny exon appears not to be supported by RNA-Seq data. This exon is only 5 nucleotides in length and it is difficult/impossible to map RNA-Seq reads if the corresponding match is too short. This explains the gap in the read coverage. The gene model, which encodes a GH10 endoxylanase, is perfect as it is here, but required quite a bit of tweaking. There are several other mind-boggling gene models that have been seen (for instance, exons of 2 nucleotides).

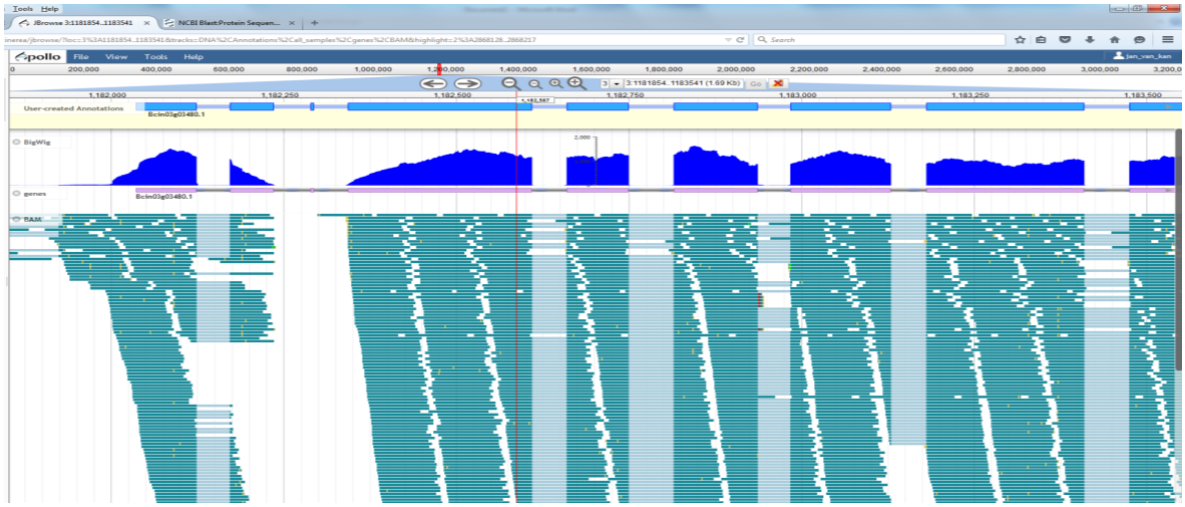


Figure 12- Reads not mapping to very small third exon

4.2.2 Absence of a gene

It is entirely possible for there to be regions in the genome where you will see RNA aligned to a locus (coverage sometimes high) but with *no* gene model predicted. This can happen for many reasons. For instance, the RNA could be derived from transposons or pseudogenes. The assembly itself may contain errors or the RNA alignments could be wrong or misleading specially over highly-similar loci such as repeats. It is not advisable to insert entirely new gene models if uncertain and if there is no other clear evidence (in addition to RNA) to support it (for instance, strong conservation).

4.2.3 Spliced antisense transcripts

There are also examples of spliced, antisense transcripts where the read mapping suggests that an intron is spliced, but the gene model does not show it. In some cases, inspection of the splice junction will show you that in the orientation of the gene model, the splice junctions read CT……AG, totally against all consensus rules. In reality, such a situation reflects an intron in an antisense transcript with splice junctions GT…AG, which is perfectly normal. These antisense transcripts may be non-coding.

4.2.4 Surprising variation

More data can occasionally uncover surprising variation. In the example below, showing multiple gene sets and RNA-Seq data for *Z. tritici,* all the Illumina tracks show a very small intron that is absent in the gene models and RNA from other laboratories. In theory, the strains should be the same but are maintained in different labs and possibly different conditions. One could imagine that mutations could accumulate and lead to such a difference.

Figure 13 - More data shows interesting variation (*Zymoseptoria tritici*): an intro absent in all gene models and RNA from other laboratories

5.0 Functional curation of gene models

There is also **functional curation** that can be done. This can involve the following:

- Assigning or changing a gene name
- Deciding on the biotype of a gene (based on strong conservation and/or very large ORF)
- Adding Gene Ontology (GO) terms
- Adding or changing the function or description of a gene or product
- Adding publications

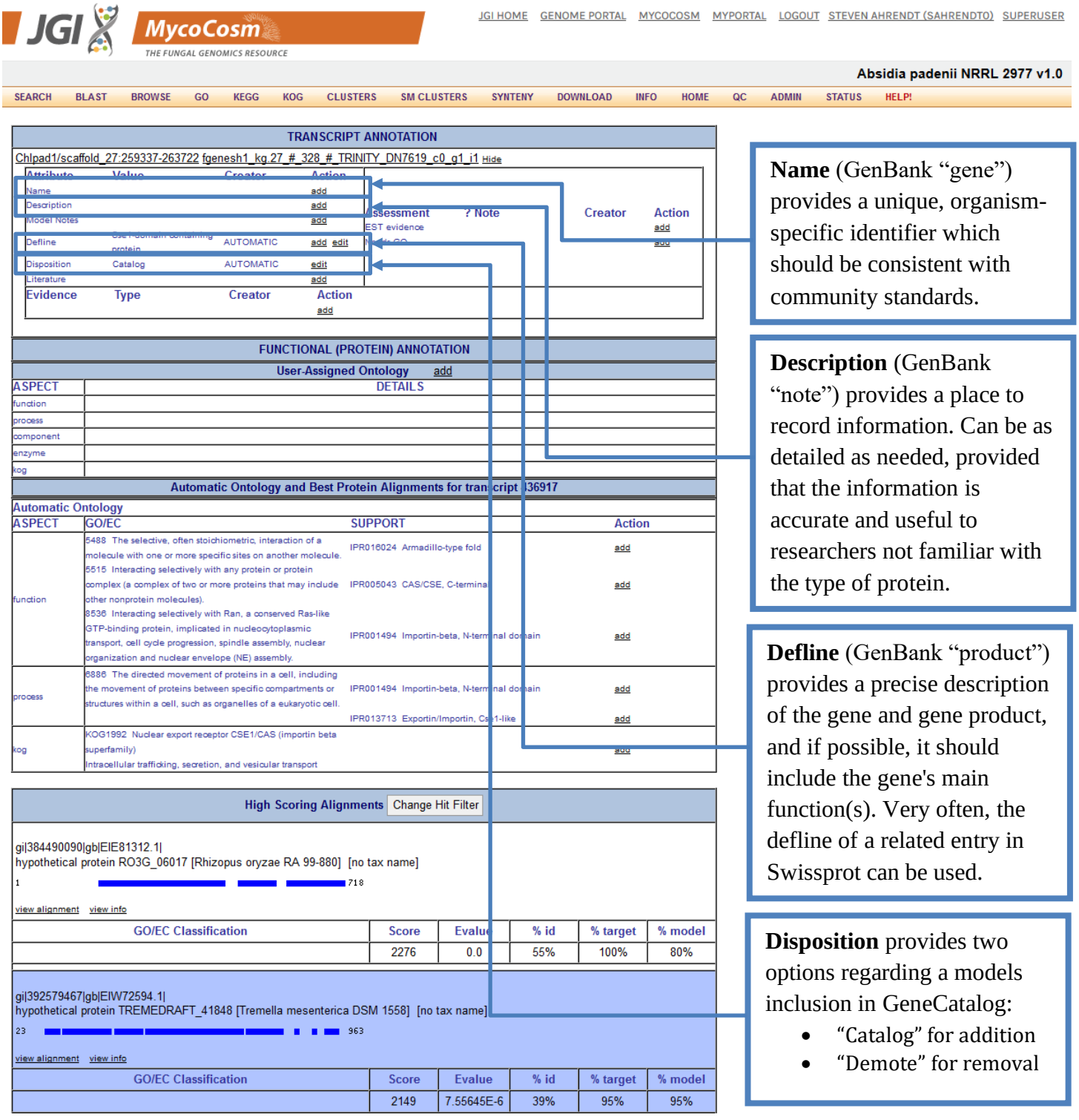
Functional annotation is the process of attaching biological information to sequences of genes or proteins. When updating functional annotation in Apollo, you can update protein descriptions, PubMed IDs, alternative product descriptions and previous identifies/aliases, EC

numbers, GO terms using Biological Process, Cellular Component, and Molecular Function ontologies, and add comments (e.g. phenotypes and other relevant data).

Appendix A: How to use the JGI MycoCosm platform

The Transcript Annotation Page

If you are a registered user, you can annotate a genome with information about the gene you are viewing. This is accomplished via the Transcript Annotation tool, which displays annotation information for the gene, and allows a user to modify several fields, including a model's Disposition by promotion (or demotion) to (or from) GeneCatalog.



Name (GenBank "gene") provides a unique, organism-specific identifier which should be consistent with community standards.

Description (GenBank "note") provides a place to record information. Can be as detailed as needed, provided that the information is accurate and useful to researchers not familiar with the type of protein.

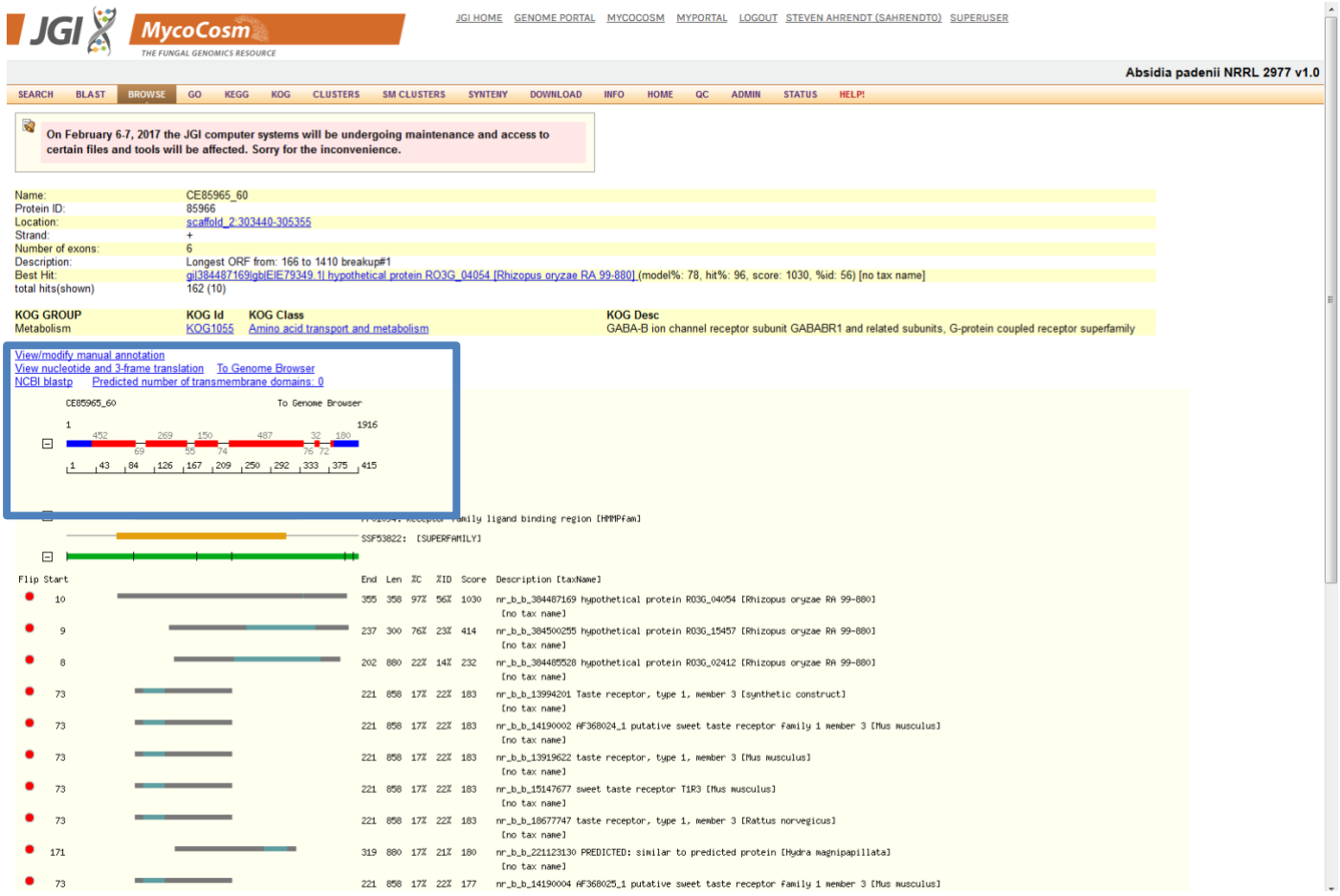
Defline (GenBank "product") provides a precise description of the gene and gene product, and if possible, it should include the gene's main function(s). Very often, the defline of a related entry in Swissprot can be used.

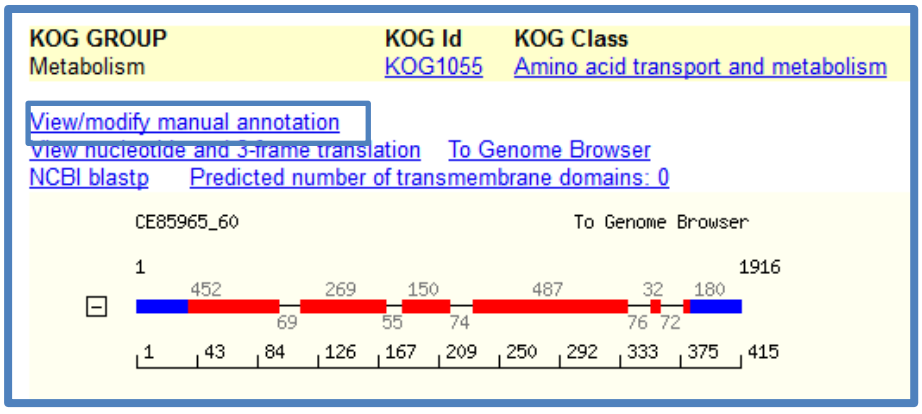
Disposition provides two options regarding a models inclusion in GeneCatalog:

- "Catalog" for addition
- "Demote" for removal

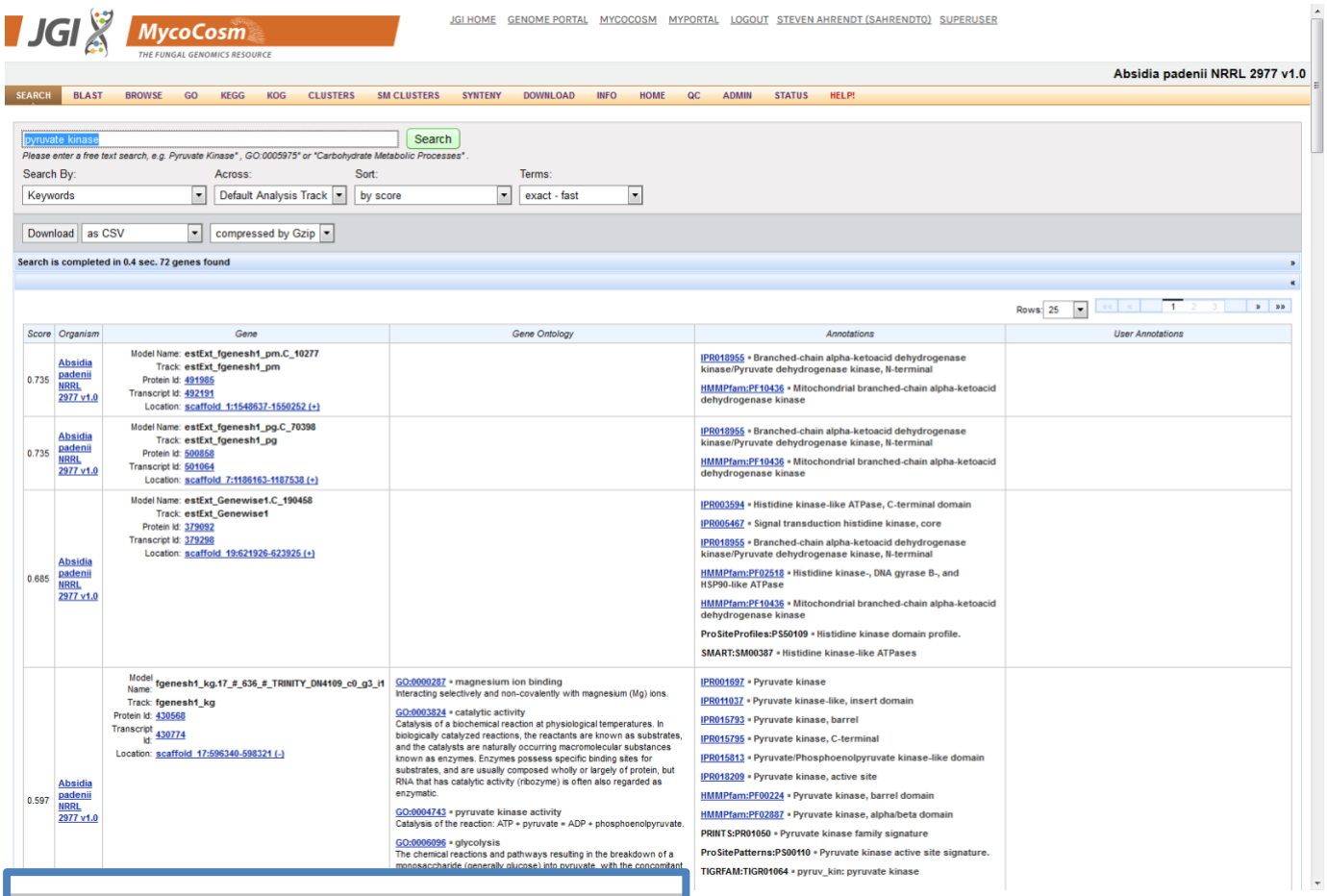
There are multiple ways of accessing the transcript annotation page for a given gene model:

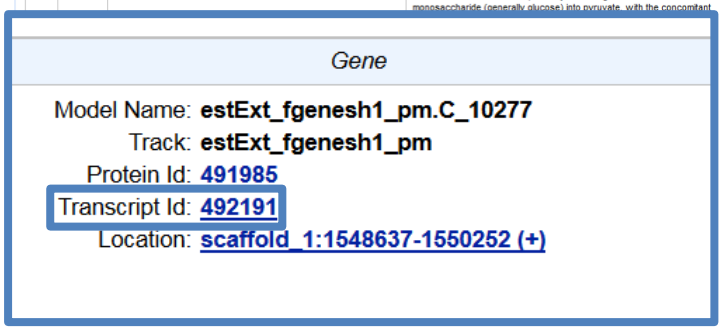
1. Via the View/Modify manual annotation link on the gene model's Protein page:





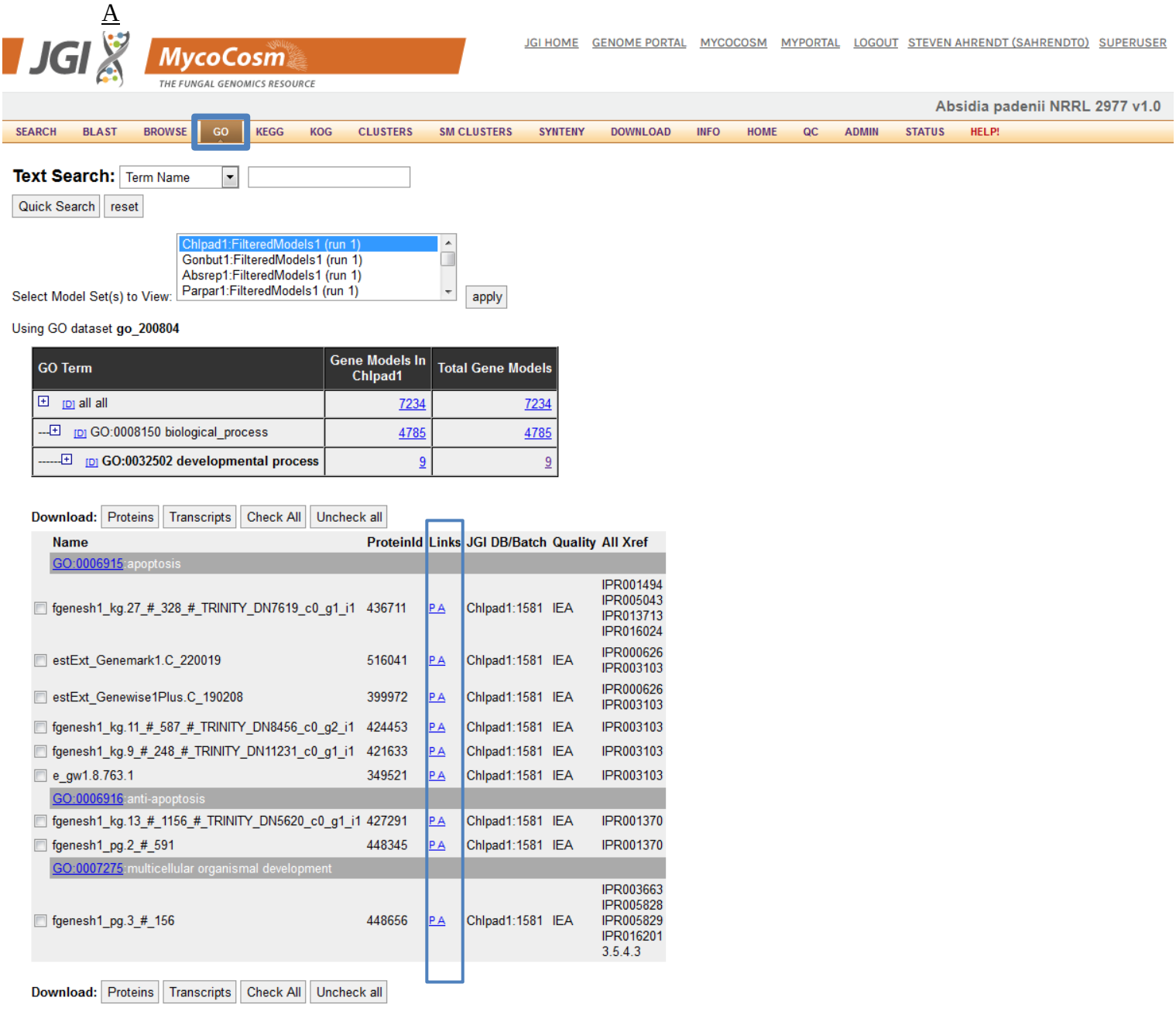
2. Via Advanced Searching directly against annotations
    a. Gene models which match the specified search criteria are returned as a table, sorted by relevance score. The Gene column provides the following links:
        - Protein id: Link to the Protein page
        - Transcript id: Link to the Transcript Annotation page
        - Location: Link to the genome browser, zoomed on the gene model



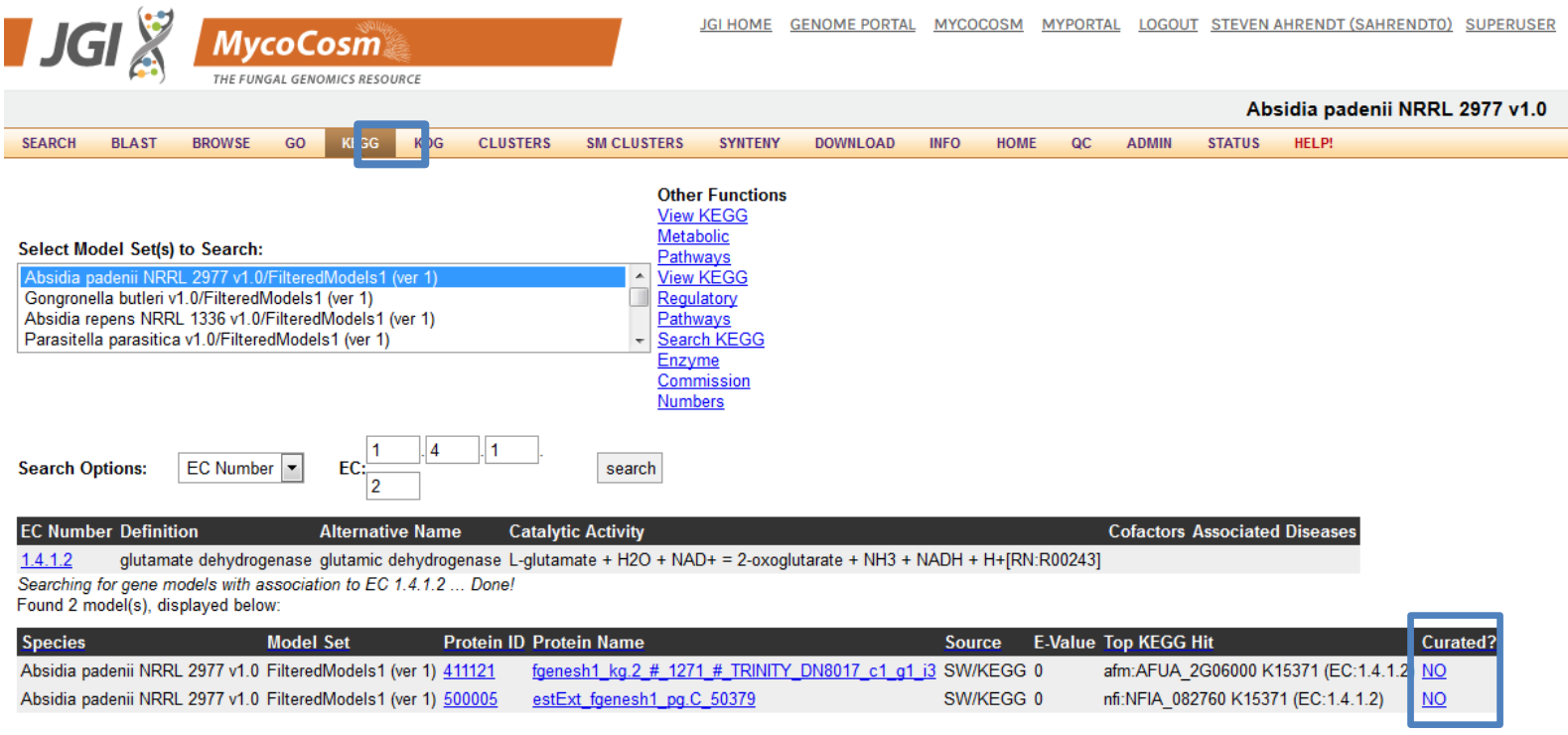


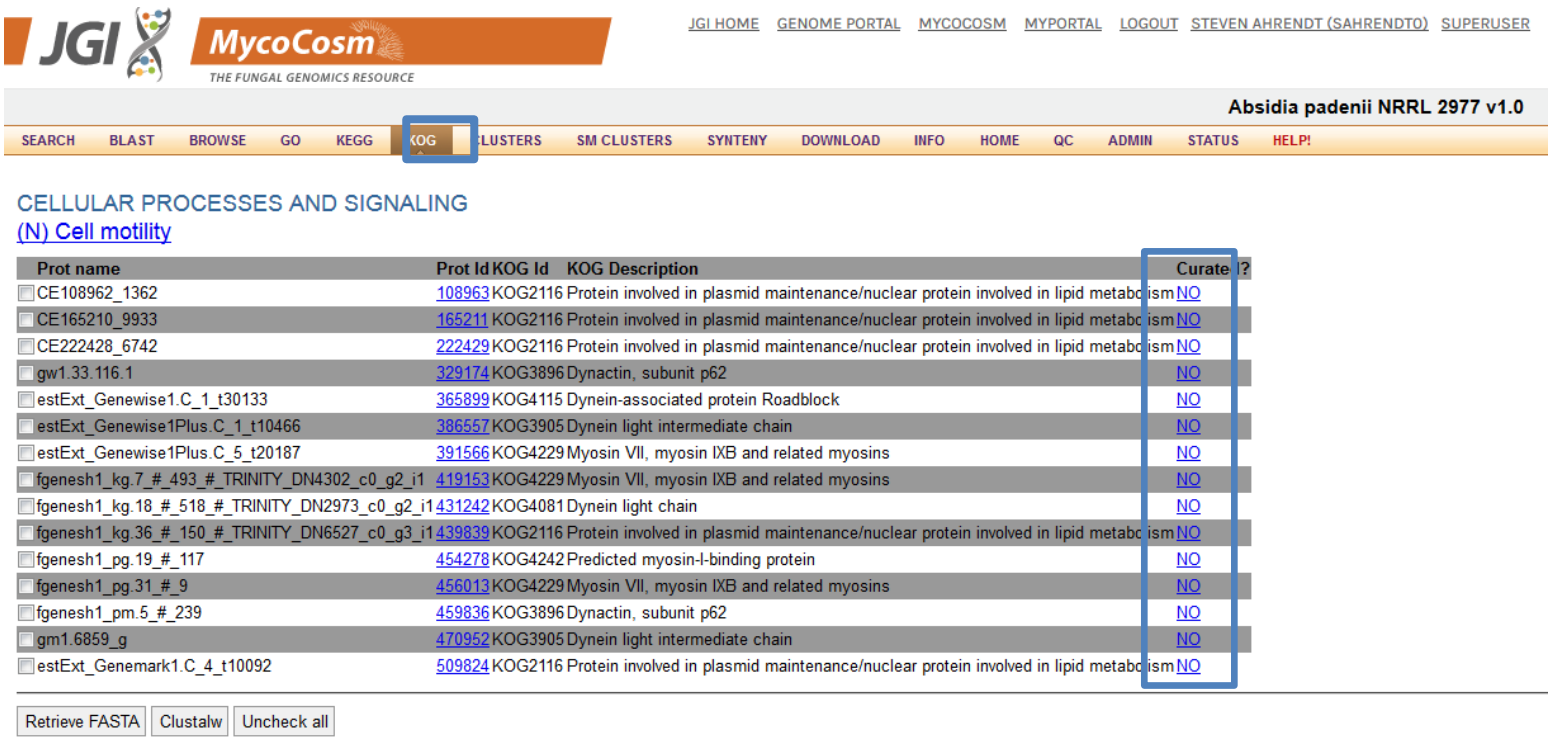
3. Via the GO/KEGG/KOG functional tools
    a. These utilities provide dynamic lists of gene models which match functional search criteria specific to the particular functional category
    b. (GO) For gene models belonging to a particular GO category, the Links column contains the following:
        - P: Link to the Protein page
        - A: Link to the Transcript Annotation page

A


JGI
MycoCosm
THE FUNGAL GENOMICS RESOURCE
JGI HOME GENOME PORTAL MYCOCOSM MYPORTAL LOGOUT STEVEN AHRENDT (SAHRENDTO) SUPERUSER
Absidia padenii NRRL 2977 v1.0
SEARCH BLAST BROWSE GO KEGG KOG CLUSTERS SM CLUSTERS SYNTENY DOWNLOAD INFO HOME QC ADMIN STATUS HELP!
Text Search: Term Name
Quick Search reset
Chlpad1:FilteredModels1 (run 1)
Gonbut1:FilteredModels1 (run 1)
Absrep1:FilteredModels1 (run 1)
Parpar1:FilteredModels1 (run 1)
Select Model Set(s) to View:
apply
Using GO dataset go_200804
GO Term
Gene Models In Chlpad1
Total Gene Models
[D] all all 7234 7234
[D] GO:0008150 biological_process 4785 4785
[D] GO:0032502 developmental process 9 9
Download: Proteins Transcripts Check All Uncheck all
Name ProteinId Links JGI DB/Batch Quality All Xref
GO:0006915:apoptosis
fgenesh1_kg.27_#_328_#_TRINITY_DN7619_c0_g1_i1 436711 P A Chlpad1:1581 IEA IPR001494 IPR005043 IPR013713 IPR016024
estExt_Genemark1.C_220019 516041 P A Chlpad1:1581 IEA IPR000626 IPR003103
estExt_Genewise1Plus.C_190208 399972 P A Chlpad1:1581 IEA IPR000626 IPR003103
fgenesh1_kg.11_#_587_#_TRINITY_DN8456_c0_g2_i1 424453 P A Chlpad1:1581 IEA IPR003103
fgenesh1_kg.9_#_248_#_TRINITY_DN11231_c0_g1_i1 421633 P A Chlpad1:1581 IEA IPR003103
e_gw1.8.763.1 349521 P A Chlpad1:1581 IEA IPR003103
GO:0006916:anti-apoptosis
fgenesh1_kg.13_#_1156_#_TRINITY_DN5620_c0_g1_i1 427291 P A Chlpad1:1581 IEA IPR001370
fgenesh1_pg.2_#_591 448345 P A Chlpad1:1581 IEA IPR001370
GO:0007275:multicellular organismal development
fgenesh1_pg.3_#_156 448656 P A Chlpad1:1581 IEA IPR003663 IPR005828 IPR005829 IPR016201 3.5.4.3
Download: Proteins Transcripts Check All Uncheck all

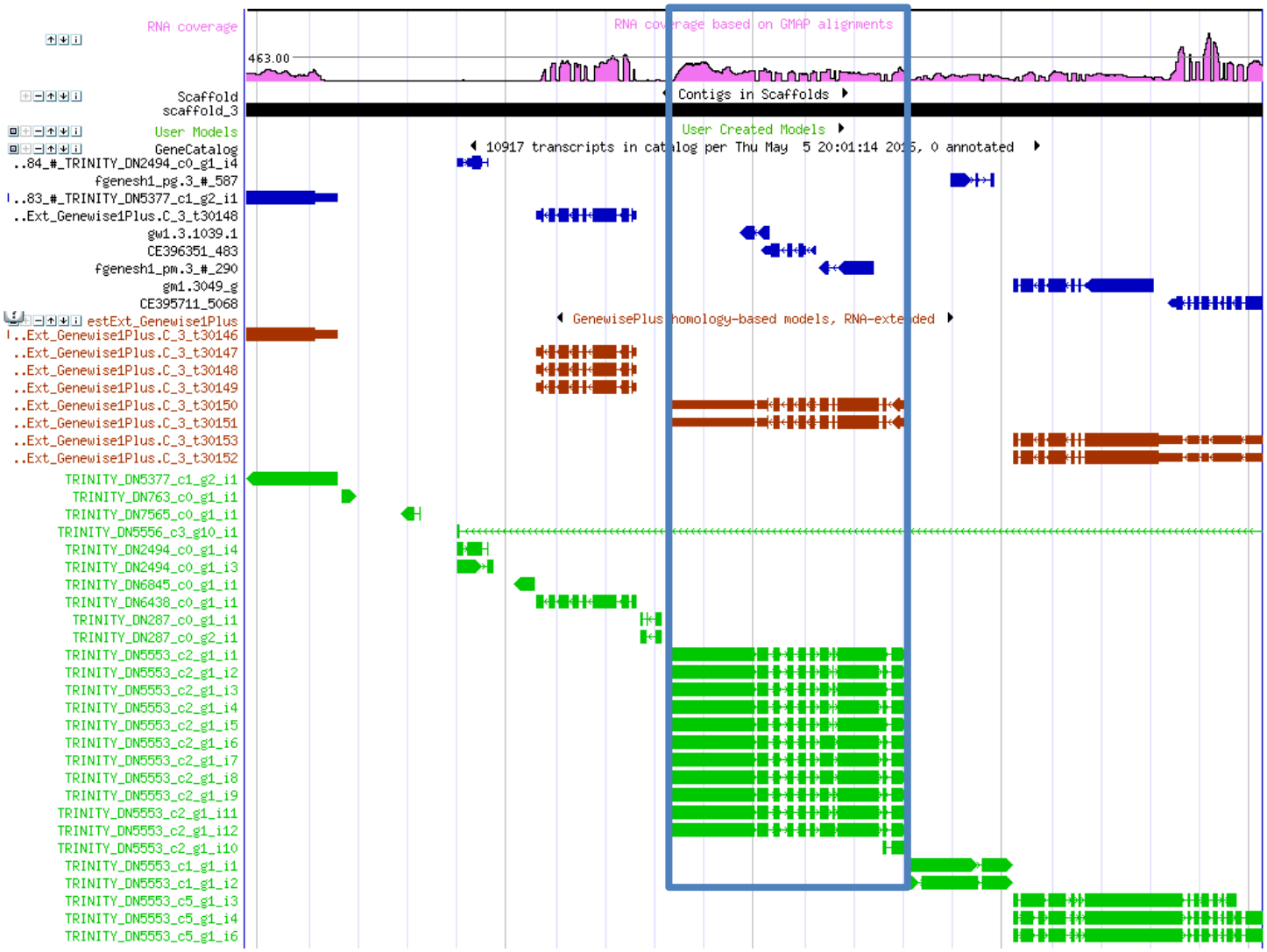
c. (KEGG/KOG) For gene models belonging to a particular KEGG metabolic pathway (EC designation) or KOG functional group (KOG id), the Curated? column contains a YES/NO link to the Transcript Annotation page





Model Promotion

To search for and evaluate alternative models at a given locus, expand all model tracks (red) and EST tracks (green). In many cases, a better model has already been generated by one of the gene predictors but was not promoted to GeneCatalog. For example, below is a view of select tracks displaying a long model covering three short fragment models, with EST and RNA coverage:



If an alternative model exists and is determined to be more accurate than the current model, it should be promoted to GeneCatalog. Use the Disposition field on the Transcript Annotation page to promote a model to GeneCatalog by setting the value to “Catalog”.

Model Creation

If none of the alternative models are of acceptable quality, it will be necessary to create a model using the Track Editor tool: http://genome.jgi.doe.gov/help/track_editor.html

Using the Track Editor, it is possible to:

- Create a new model by copying an existing model
- Edit a new model
- Add existing exons to a new model
- Create an ab initio model

Once editing is finished, the model should be released in order to initiate protein analysis. However, since releasing a model does not automatically add it to the GeneCatalog, the model's Disposition must also be set to "Catalog" via the model's Transcript Annotation page (similar to Model Promotion).

Model Demotion

Regardless of whether an existing model was promoted or a new model was created, the old/incorrect model should be demoted; otherwise it will appear concurrently with the new/correct model. Similar to Model Promotion, use the Disposition field on the Transcript Annotation page to demote a model from GeneCatalog by setting the value to "Demote". This option does not delete the model or its annotation from the database. It simply removes it from the Catalog track.

Appendix B: Brief guide to the Apollo gene editing interface

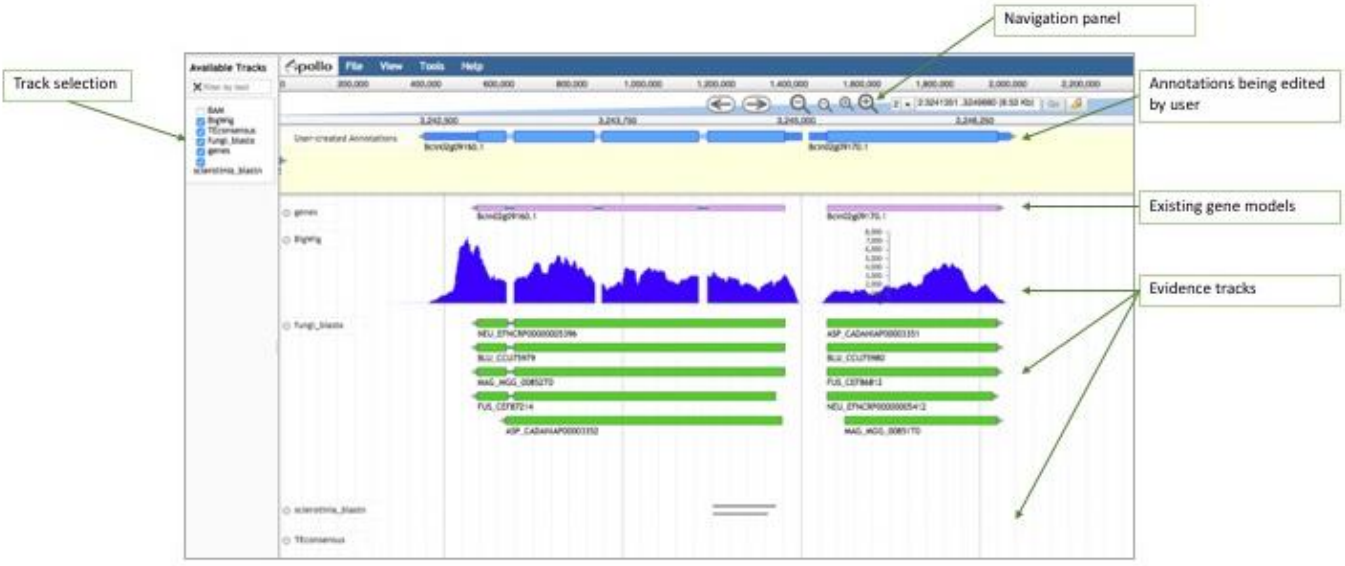


Figure B:1 - Anatomy of the Apollo gene editing interface

Apollo (http://genomearchitect.github.io/) allows annotators to modify and refine the precise location and structure of the genome elements that predictive algorithms cannot yet resolve automatically. Using Apollo, annotators may corroborate or modify the structures of coding genes, pseudogenes, repeat regions, transposable elements, and non-coding RNAs (i.e: snRNA, snoRNA, rRNA, tRNA, and miRNA).

In order to make any changes to gene models, you must first move them to the "user annotation track" highlighted in yellow above. You can click on a gene model in the "gene model" track and simply drag the feature up into the "user annotation track" while holding the mouse button. The gene model will appear in the track with different boxes for coding and non-coding sequences. Clicking the right mouse button while the cursor is on a feature opens a z-menu. Clicking the left mouse button while the cursor is on a feature enables you to move the boundaries of this feature (moving the splice junction or the UTR boundary).

You may reveal or hide any of the data tracks listed in tabular form by ticking/unticking the track selection checklist.

The blue bar at the top holds top-level menus with the following functions:

- 'File':
    - Allows users to add data files (e.g. GFF3, BAM, BigWig, etc.) by opening sequence and track files, as well as loading tracks via URLs. Apollo automatically suggests tracks to display their contents.

    - It is possible to combine the information from quantitative tracks into a ‘Combination Track’. Data from tracks containing graphs may be compared and combined in an additive, subtractive, or divisive arithmetic operation. The resulting track highlights the differences between the data.
    - The third option allows users to ‘Add sequence search track’. This tool creates tracks showing regions of the reference sequence (or its translations) that match a given string of nucleotides or amino acids residues.
- ‘View’:
    - Allows users to colour all exons in display according to CDS frame.
    - Users may choose between light and dark options for their working environment by changing the ‘Colour Scheme.’
    - Toggle the view of the plus and minus strands, and reveal or hide the labels for each track.
    - It is also possible to highlight a region using the ‘Set highlight’ option and marking the region. The highlight option will automatically be turned ‘On’ when inspecting the results from a BLAT search.
    - Annotators will also use this menu when resizing the scale of quantitative tracks.
- ‘Tools’ leads users to perform BLAT searches
- The ‘Help’ tab includes links to a list of helpful commands for Apollo, details about the version of Apollo in use and about JBrowse, as well as a link to explore Apollo Web Services options.
- On the upper right corner, a box with the username offers the option to logout. When logged out, the word ‘Login’ will be displayed instead of the username.

The Apollo user guide can be found here: http://genomearchitect.github.io/users-guide/

Appendix C: How to use the VEuPathDB/FungiDB Apollo interface for structural changes to gene models

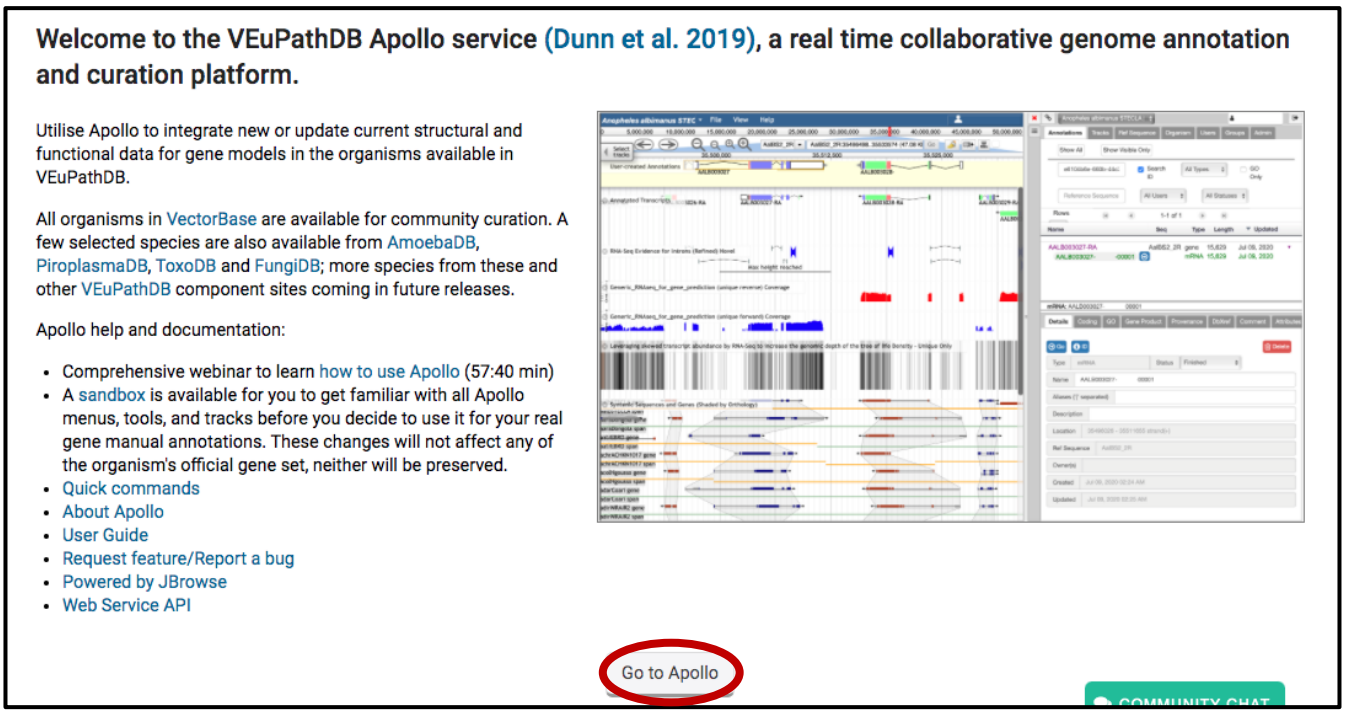
1. Structural annotation in VEuPathDB Apollo

In this short tutorial we are showing you step-by-step how to change genes structures in Apollo.

Structural annotation can involve:
- modifying a gene model, i.e. extending exons, adding exons, deleting exons, adding UTRs (see step 4.1)
- merging two or more gene models (see step 4.2)
- splitting a gene model (see step 4.3)
- adding alternative transcripts (see step 4.4)
- adding new genes (see step 4.5)

1) Accessing Apollo

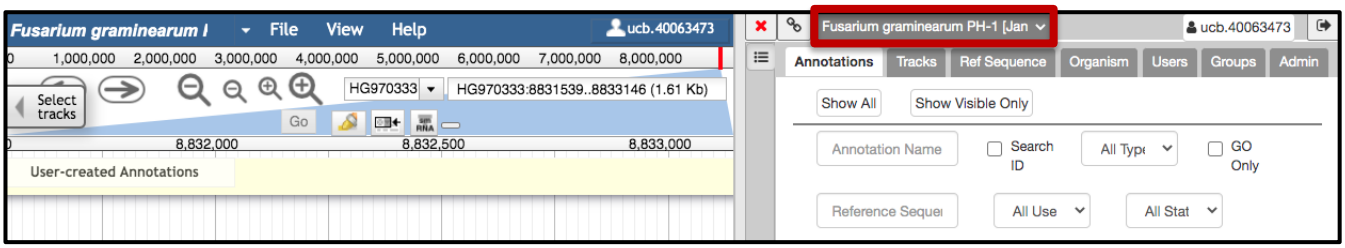
To access Apollo go to the following page:
https://fungidb.org/fungidb/app/static-content/apollo_help.html
Select **Go to Apollo.**



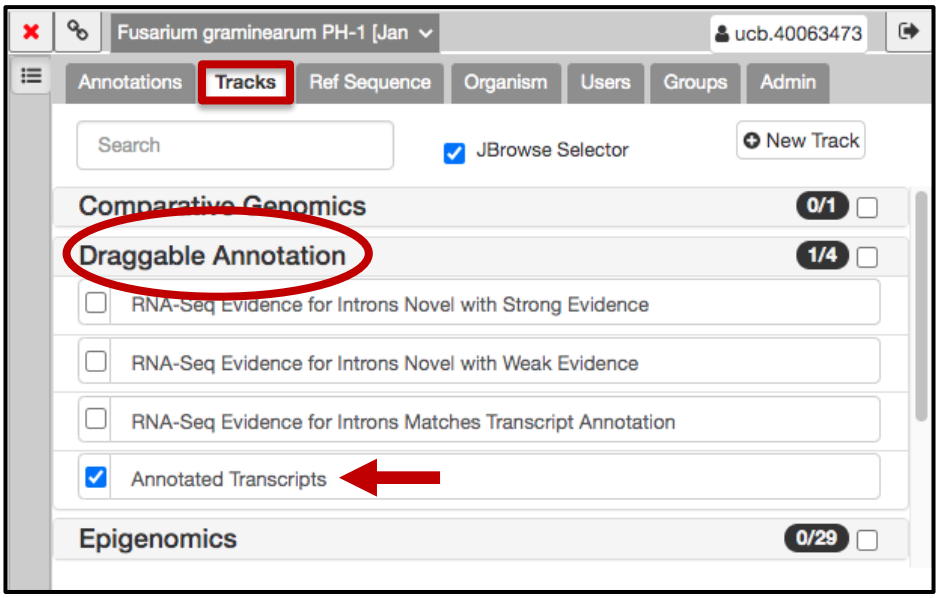
To use Apollo you need to be logged into VEuPathDB. If you have not done so yet log now into Apollo with your VEuPathDB user ID and password.

2) Navigating to the gene or genome coordinates

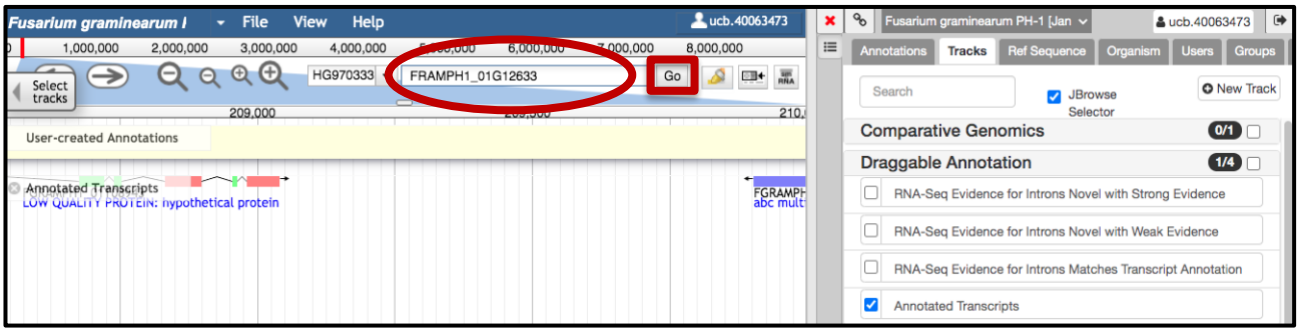
Select on the right-hand side from the drop-down menu the genome that you would like to annotate.



Select the tab **Tracks**, click on the menu item **Draggable Annotation** and select **Annotated Transcripts**.

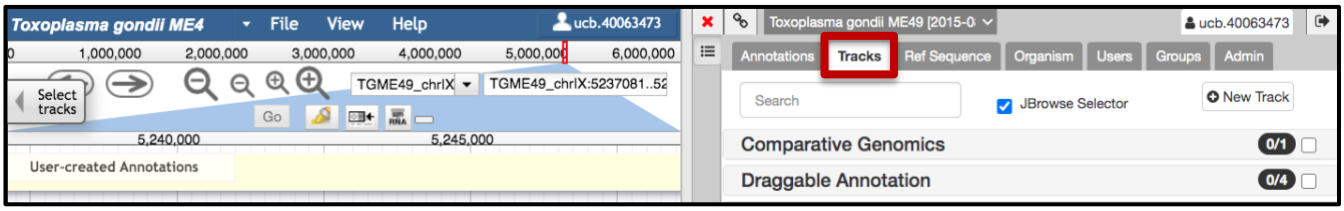


Type in the search box the gene ID of your gene of interest, wait a few seconds until the gene ID has been found and then click on Go.

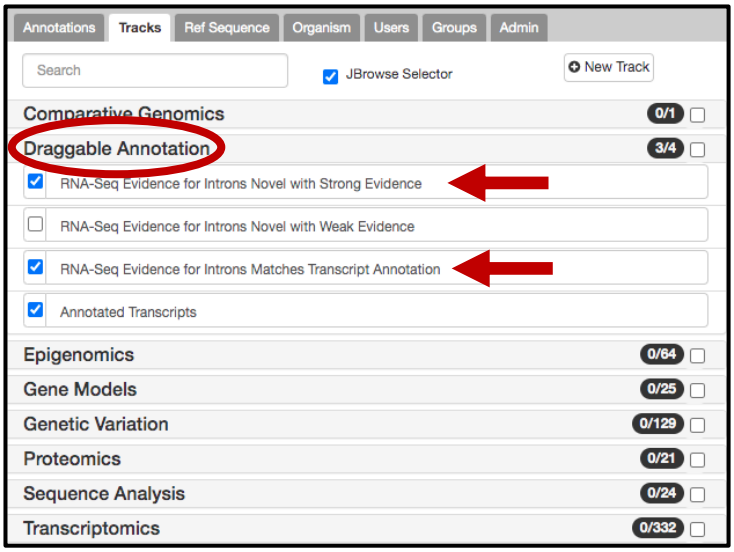


3) Adding draggable annotation and supporting evidence

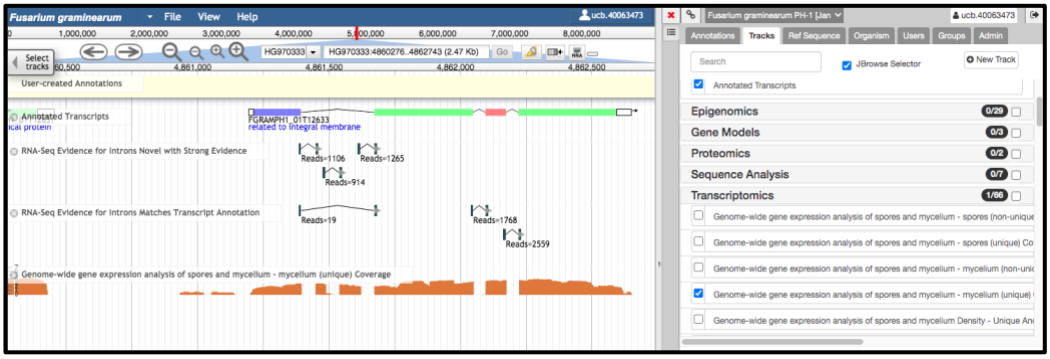
Select on the right-hand side the tab **Tracks.**



Click on the menu item **Draggable Annotation** select in addition to the **Annotated Transcripts**, **RNA-Seq Evidence for Introns Novel with Strong Evidence** and **RNA-Seq Evidence for Introns Matches Transcript Annotation**.



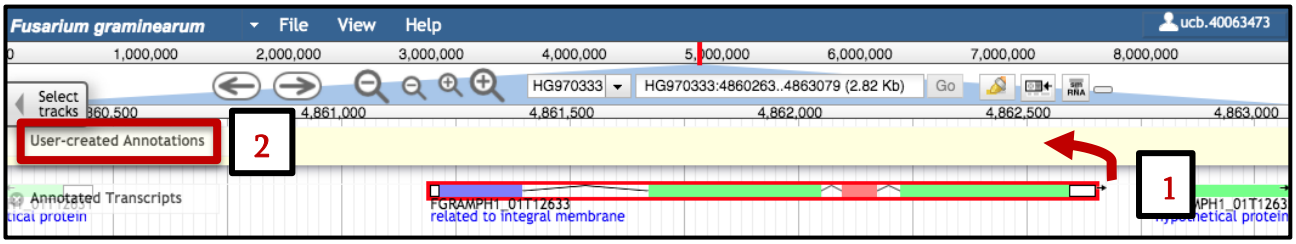
Select additional evidence, i.e. RNAseq plots.



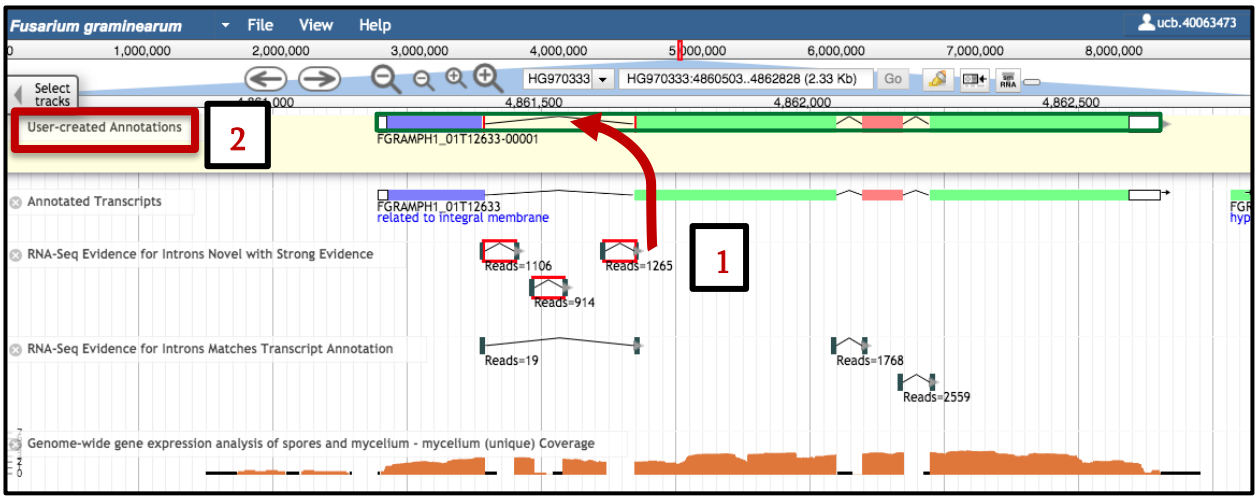
4) Changing the gene structure

4.1) Modifying gene structures - adding additional exons

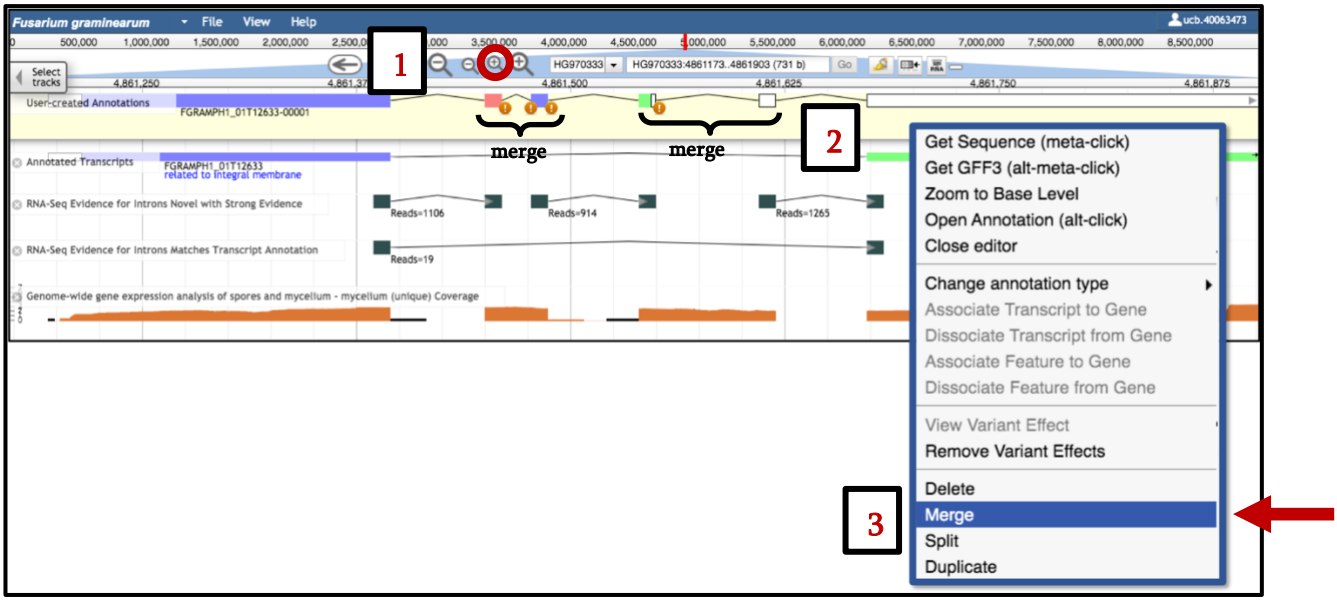
Select the gene model by clicking on one of the introns or by double clicking on the gene model (**1**). The gene model will show up with red boundaries. Drag and drop the gene into the User-created Annotations track (**2**).



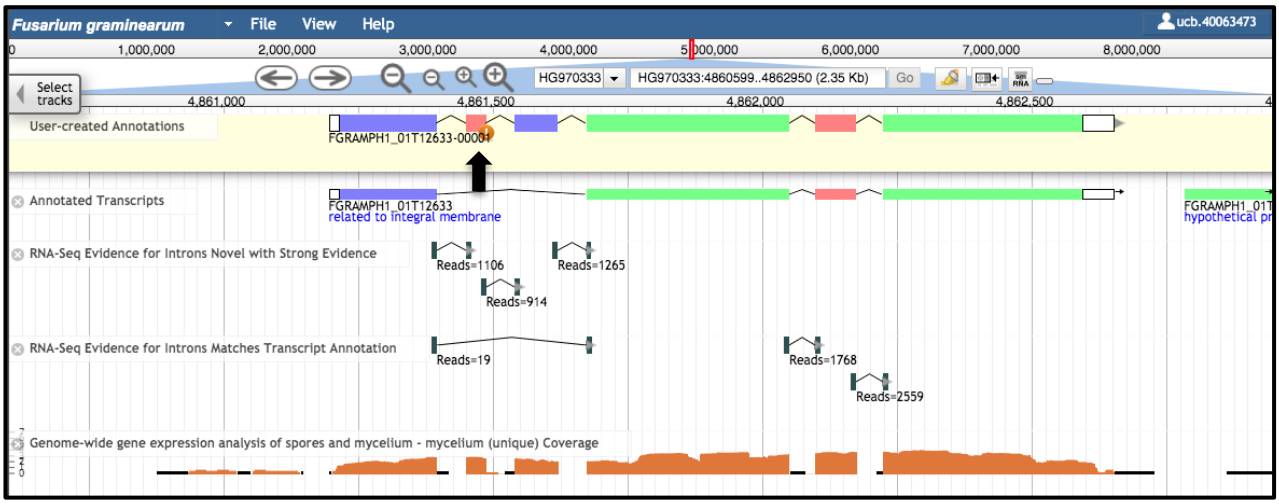
To create new exons drag and drop the intron junctions into the User-created Annotations area. You can either select the intron junctions individually, or hold down the shift key and select all intron junctions with strong evidence (**1**), drag and drop them into the gene model (**2**). The gene boundaries will show up in green when dragging and dropping.



Zoom in by clicking on the + sign on the top (**1**). Press the Shift key and select the exons that should be merged (**2**). With a right-click open the drop-down menu and choose **Merge** (**3**). Alternatively, select one of the exons you would like to merge, go to the edge of the feature until a little arrow appears and extend the exon until it overlaps with the second exon.

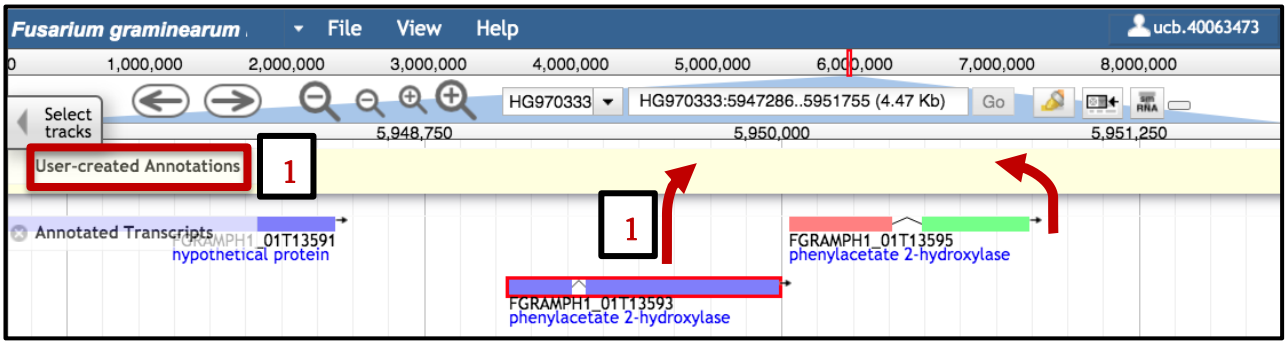


Please note, the exclamation mark informs about non-canonical splice sites, i.e. GC splice sites.

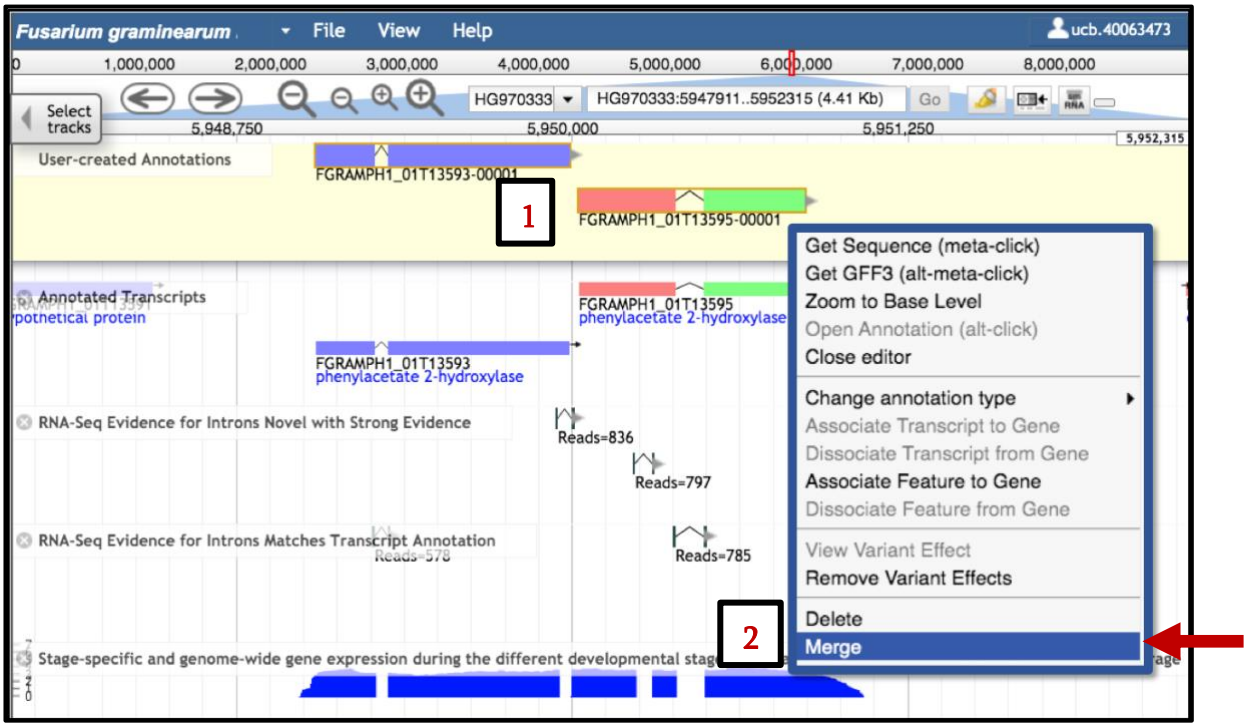


4.2) Merging two genes

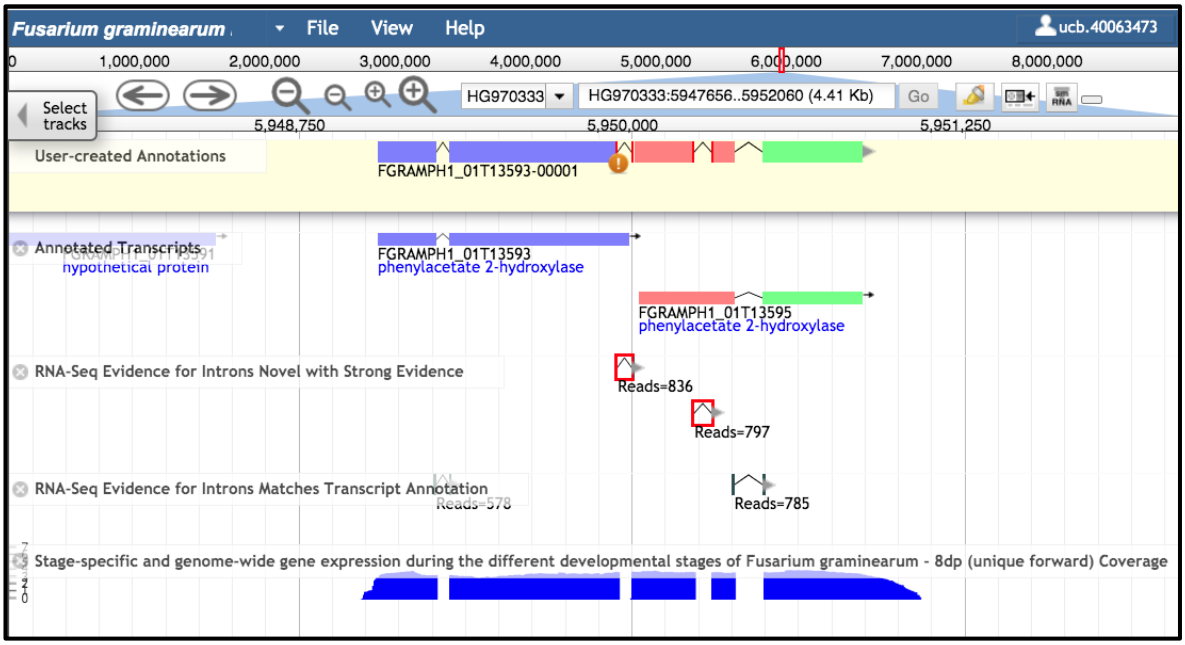
Select the gene models that you would like to merge (**1**). Drag and drop the genes into the User-created Annotations track (**2**).



Hold down the shift key and select both gene models (**1**). With a right-click open the drop-down menu and choose **Merge** (**2**).

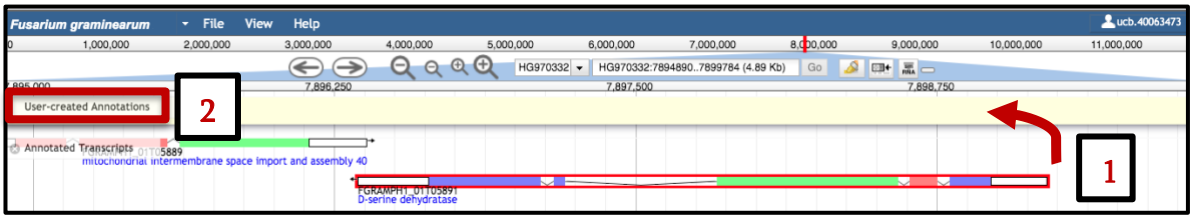


Drag and drop the intron junctions into the User-created Annotations area and merge them with the gene. If the exon is longer than the new intron, you need to shorten the exon.

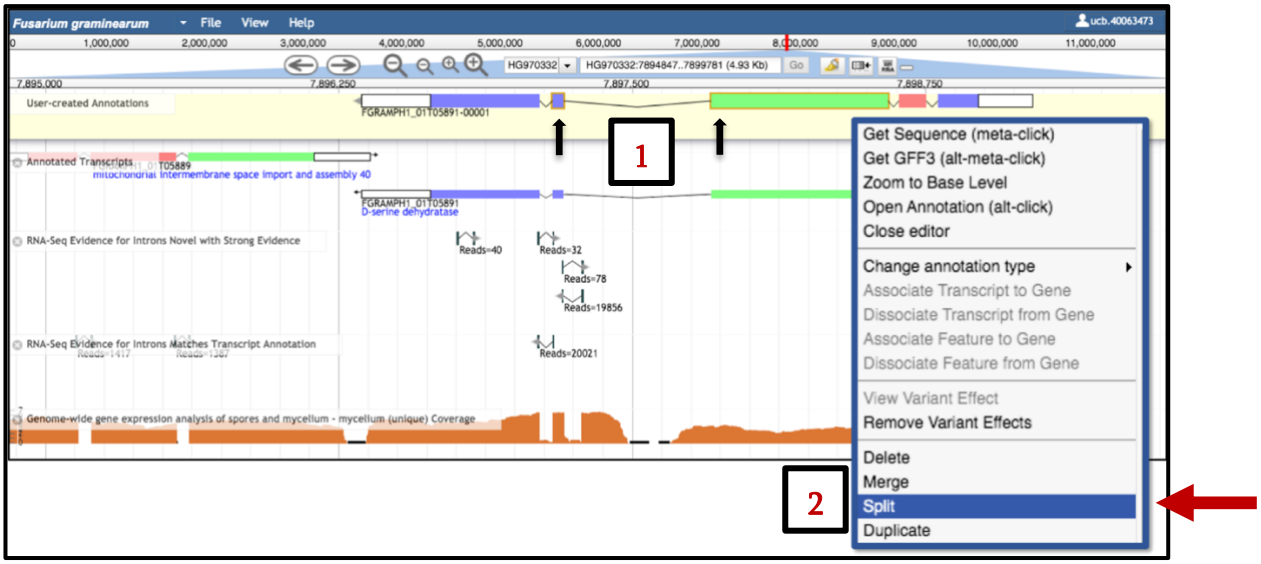


4.3) Splitting genes

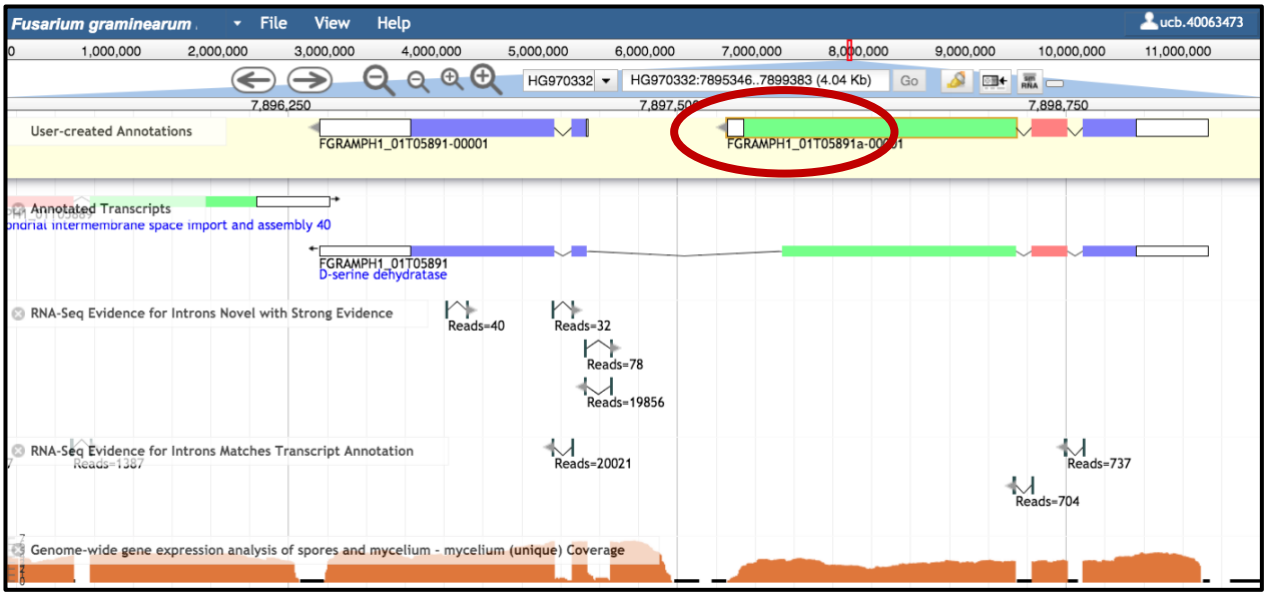
Select the gene model by clicking on one of the introns or by double clicking on the gene model (**1**). The gene model will show up with red boundaries. Drag and drop the gene into the User-created Annotations track (**2**).



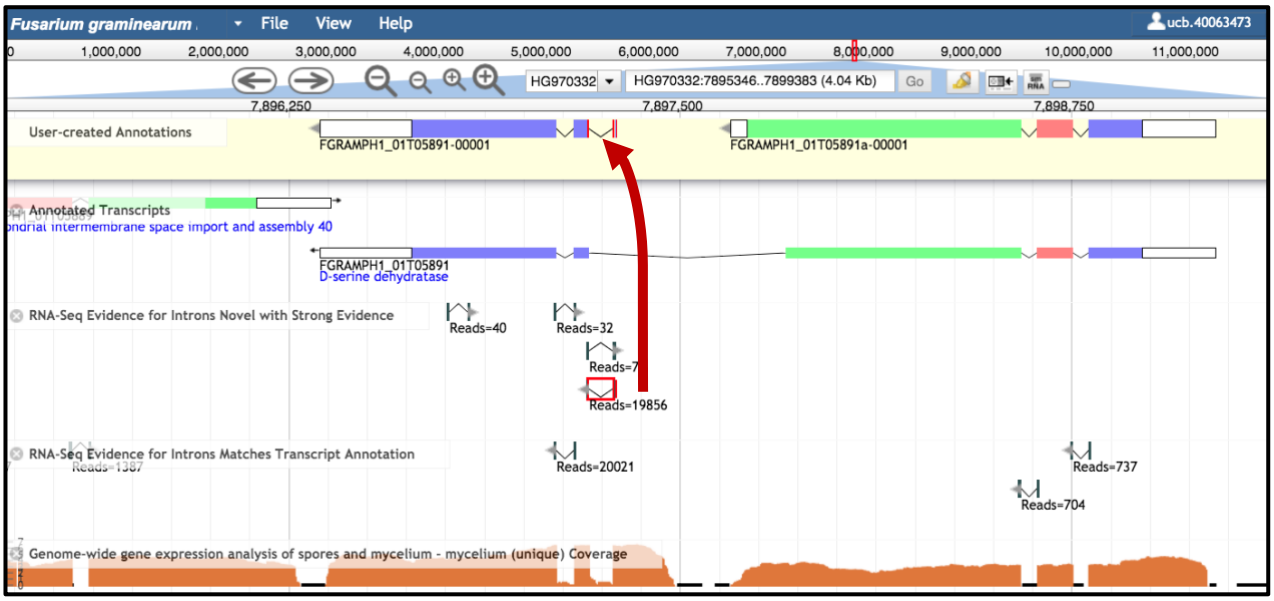
To split the gene model hold down the shift key, mark the two exons that border the intron that should be split (**1**). With a right-click open the annotation drop-down menu and select **Split** (**2**).



Now that the gene model has been split, the newly created genes need a correct start codon and stop codon. To create the stop, click with your mouse at the end of the gene model and extend it. The 3'UTR will be created automatically!

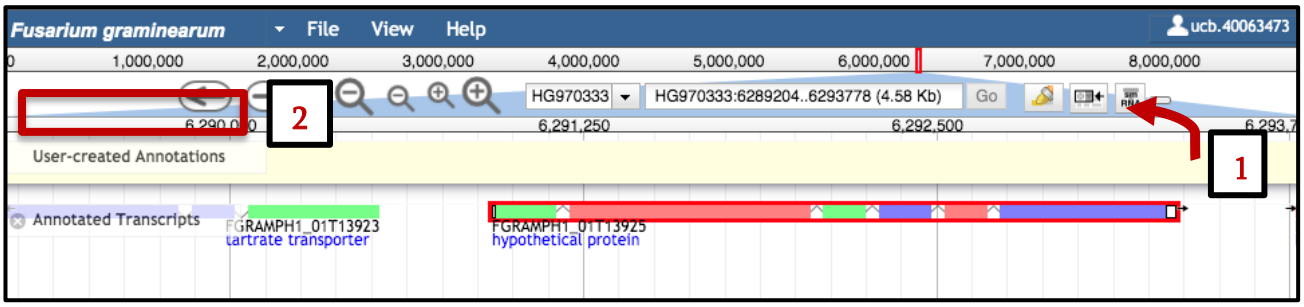


The gene model on the left, needs a start codon. Drag and drop the splice junction into the annotation area. Move it up and hover the splice junction over the gene. A green border will show up which indicates that the intron junction has been merged with the gene. Now extend the first exon to include a UTR.

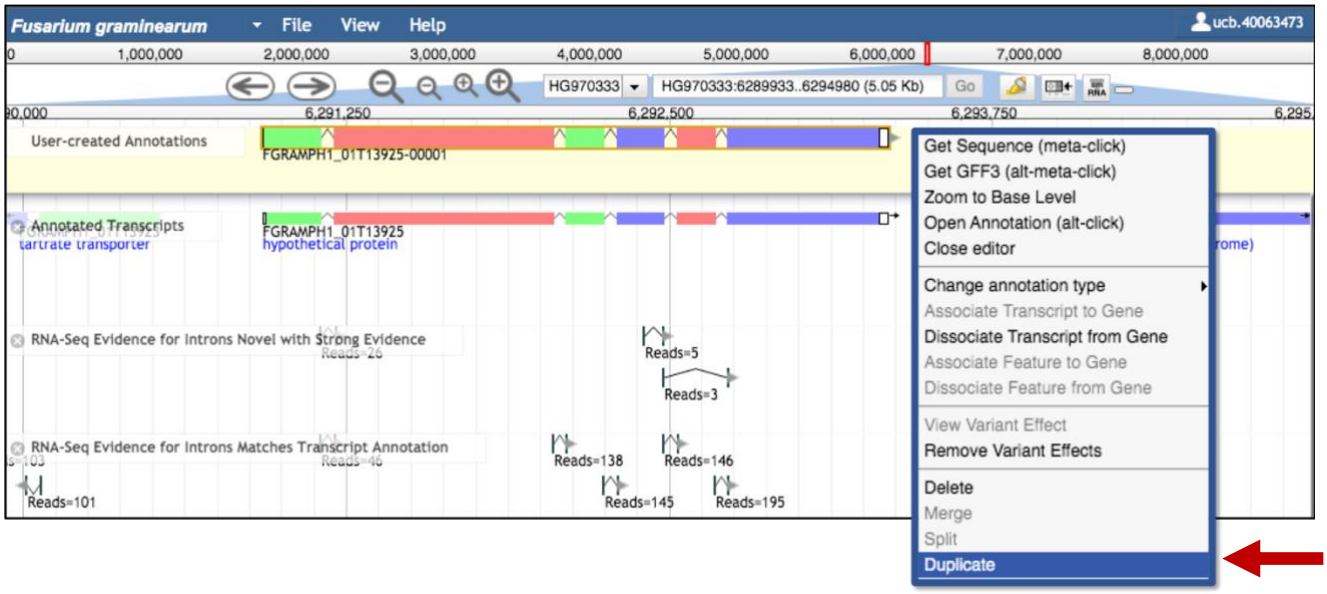


4.4) Adding alternative transcripts

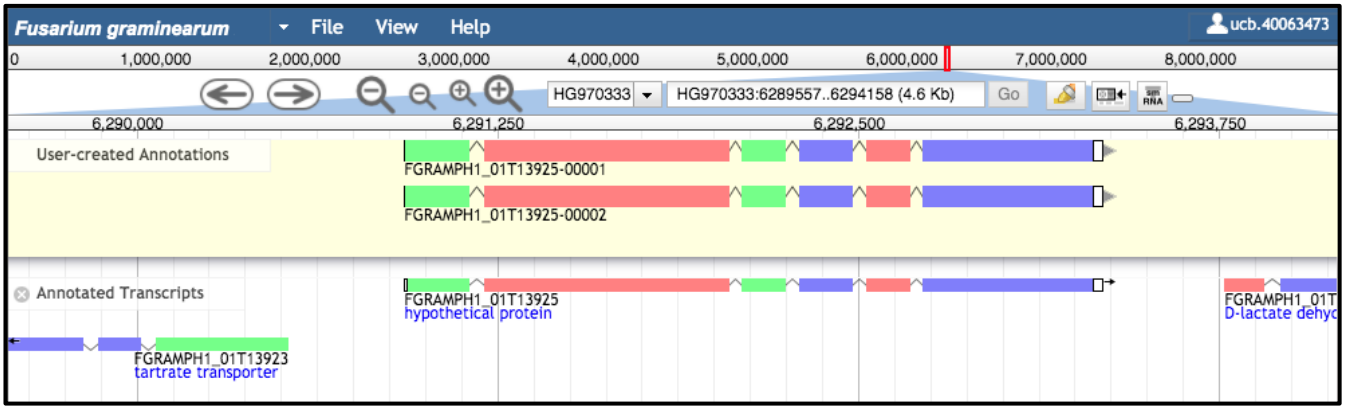
Select the gene model by clicking on one of the introns or by double clicking on the gene model (**1**). The gene model will show up with red boundaries. Drag and drop the gene into the User-created Annotations track (**2**).



Select the gene in the User-created Annotations area, with a right-click open the drop-down menu and choose duplicate.

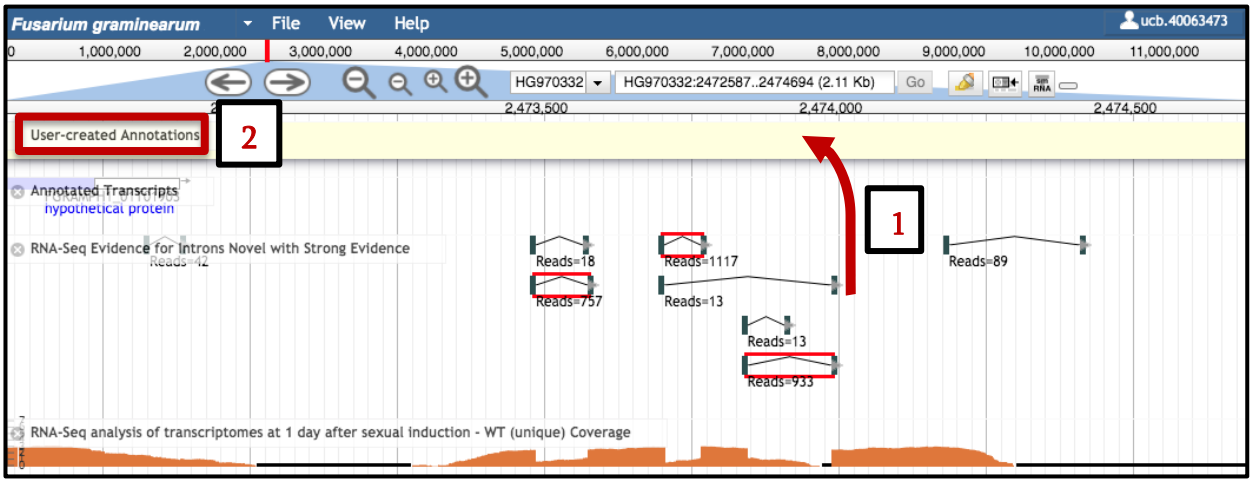


You can now see two transcripts with the different transcript ID extensions: 00001 and 00002.

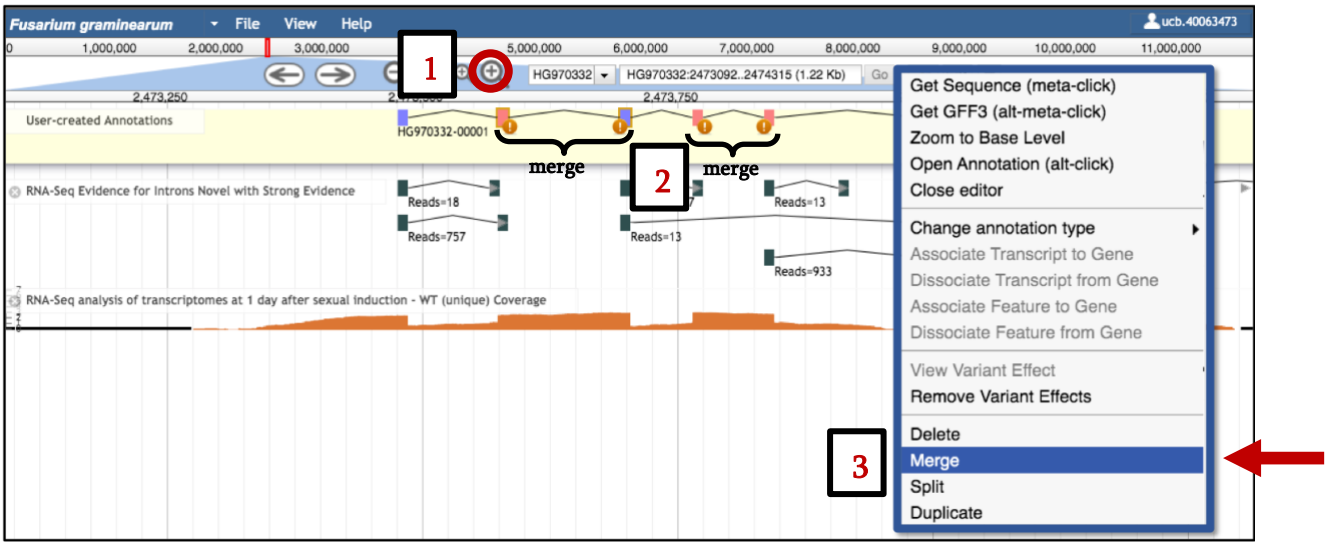


4.5) Creating a new gene

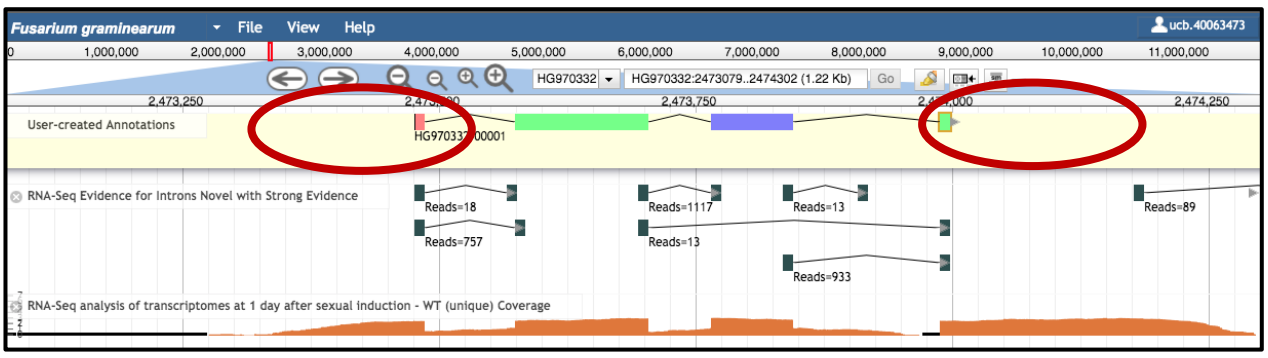
Select the supporting evidence, i.e. intron junctions (**1**). Use the shift key to select more than one intron junction. The selected intron junctions will show up with a red border. Drag and drop them into the User-created Annotations track (**2**).



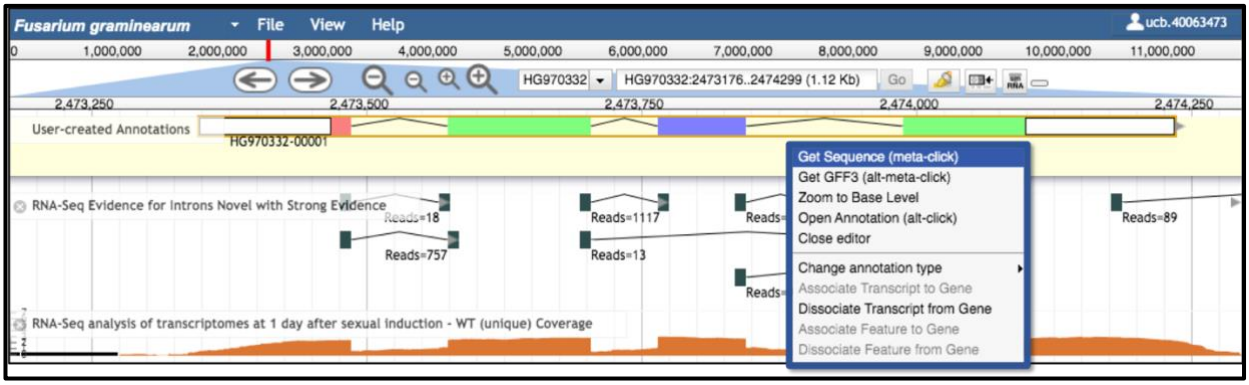
Zoom in by clicking on the + sign on the top (**1**). Press the Shift key and select the exons that should be merged (**2**). With a right-click open the drop-down menu and choose **Merge** (**3**). Alternatively, select one of the exons you would like to merge, go to the edge of the feature until a little arrow appears and extend the exon until it overlaps with the second exon.



Select the first and the last exon, go to the edge of the exon until a little arrow appears and extend it to the start/end. UTRs will be created automatically.

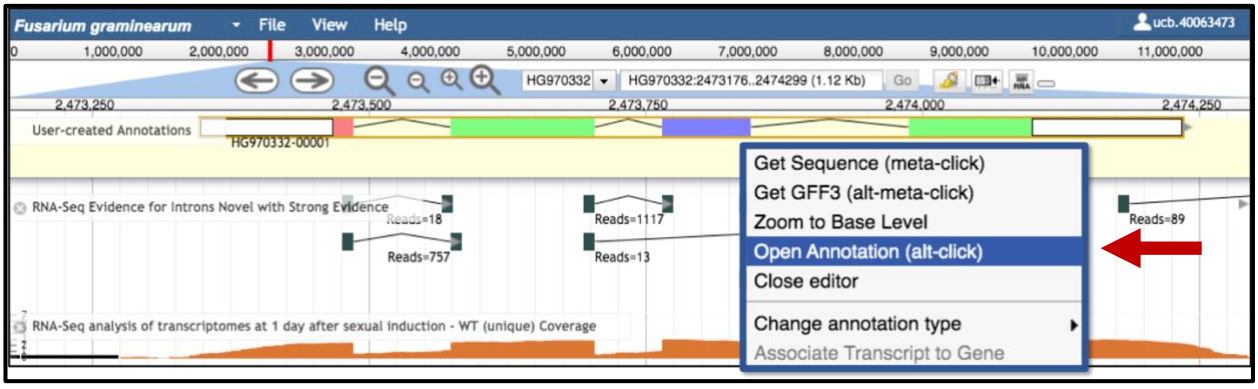


Select the new gene model, with a right-click open the annotation drop-down menu and select **Get Sequence**. Copy the sequence, run blast (https://blast.ncbi.nlm.nih.gov) and Interpro (https://www.ebi.ac.uk/interpro) to get more information about the new gene.

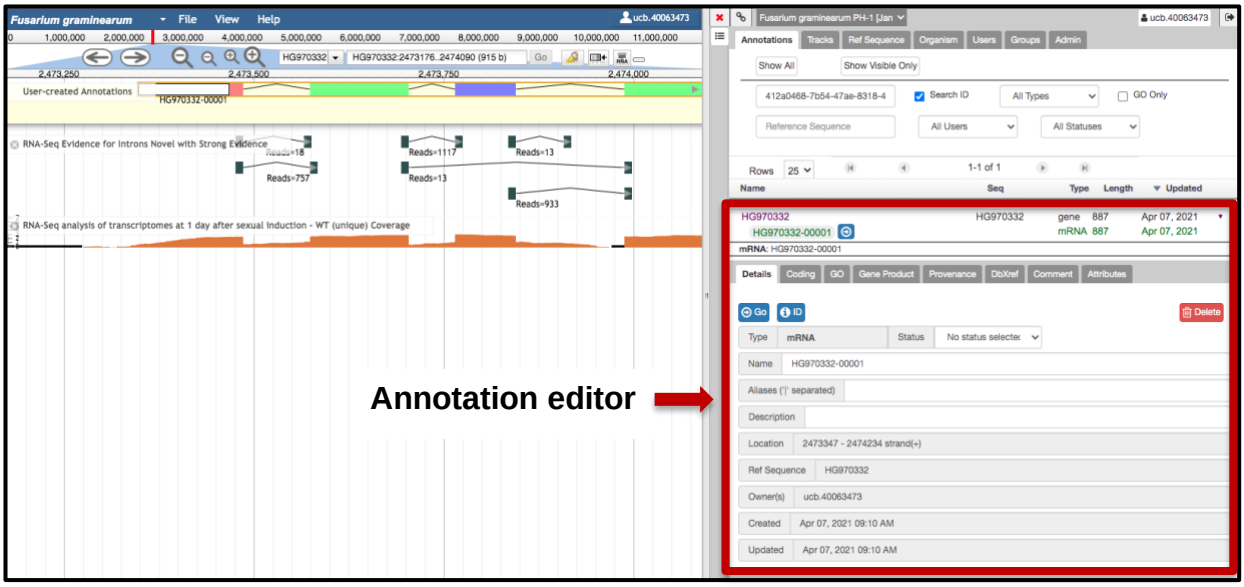


5) Opening of the Annotation editor window

Select the gene in the User-created Annotation track and with a right-click open the drop-down menu and choose **Open Annotation**. Alternatively, you can use the short cut **alt-click**.

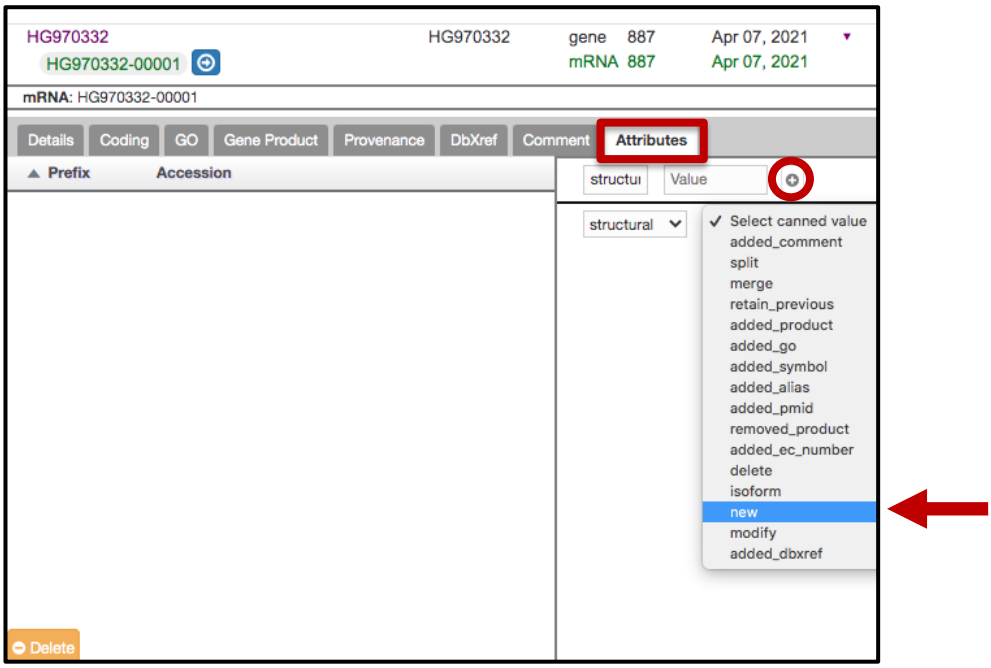


The annotation editor window is now shown on the right-hand side.

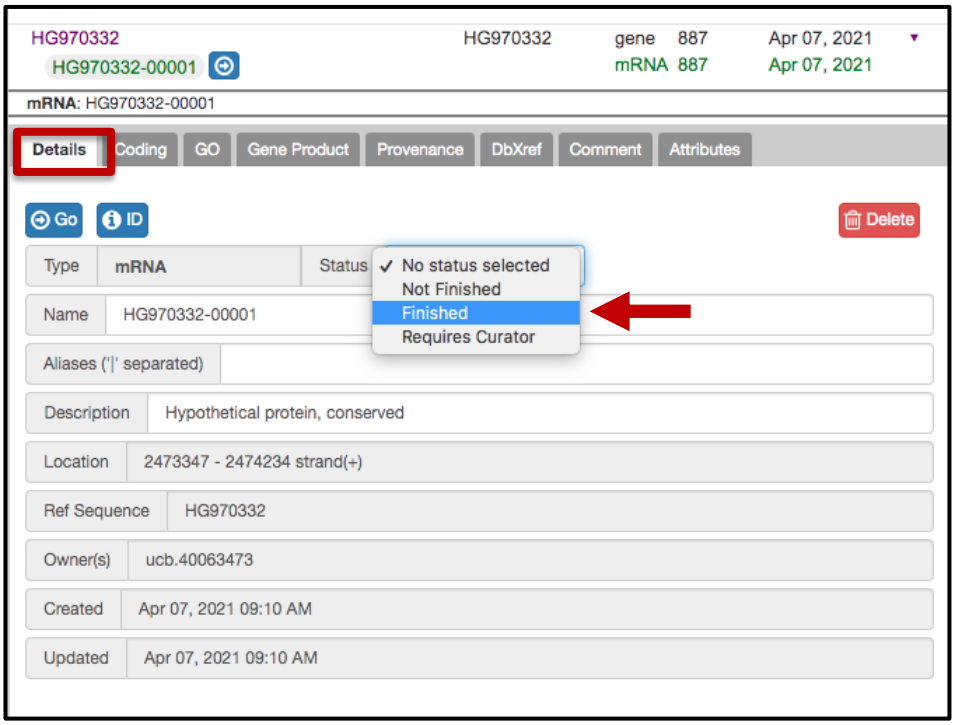


6) Finalising the structural annotation

Once the annotations panel is open click on the Attributes tab, select from the canned tag **structural** and from the canned value the structural annotation type, i.e. new, modify, merge, split or isoform. Click on the + sign.



Add a product description for newly created genes, split and merged genes. Finally go the Details tab and select the status **Finished** on the gene and mRNA.



Done! For additional questions, please get in touch with the FungiDB help desk help@fungidb.org

Appendix D: How to use the VEuPathDB/FungiDB Apollo interface for functional changes to gene models

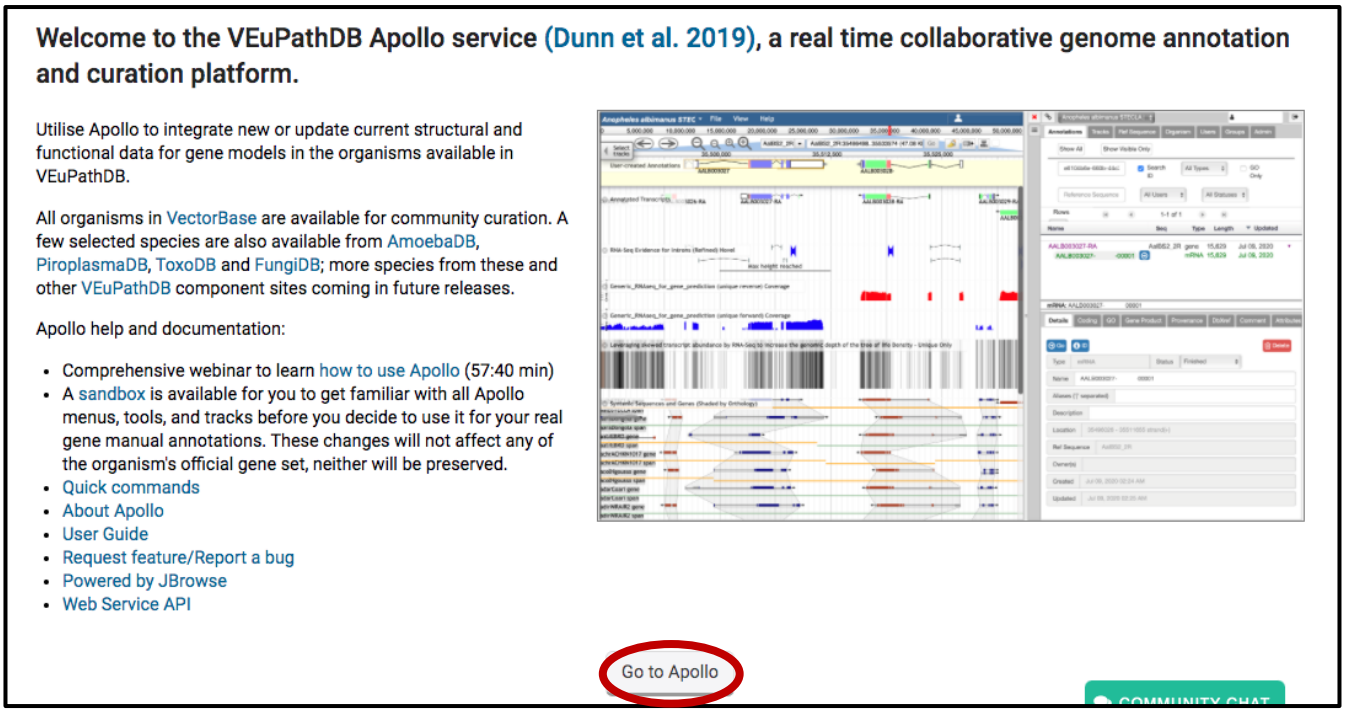
There are two options to add functional annotation in VEuPathDB:
1) Adding a user comment on the gene record page
2) Using the community annotation tool Apollo

Functional annotation can involve:
- Adding or changing the description of a gene or product
- Assigning or changing a gene name/symbol
- Adding Gene Ontology (GO) terms
- Adding a publication
- Adding an EC number

In this manual we are showing you step by step how to add a gene name, product description and GO term to a gene in Apollo.

1) Accessing Apollo

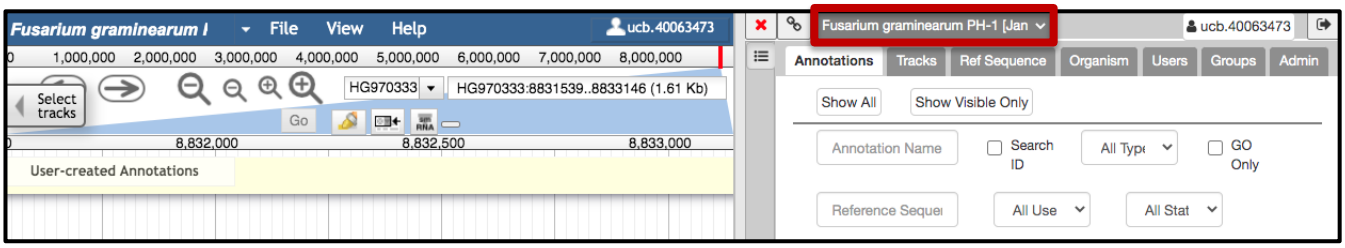
To access Apollo, go to the following page:
https://fungidb.org/fungidb/app/static-content/apollo_help.html
Select **Go to Apollo.**



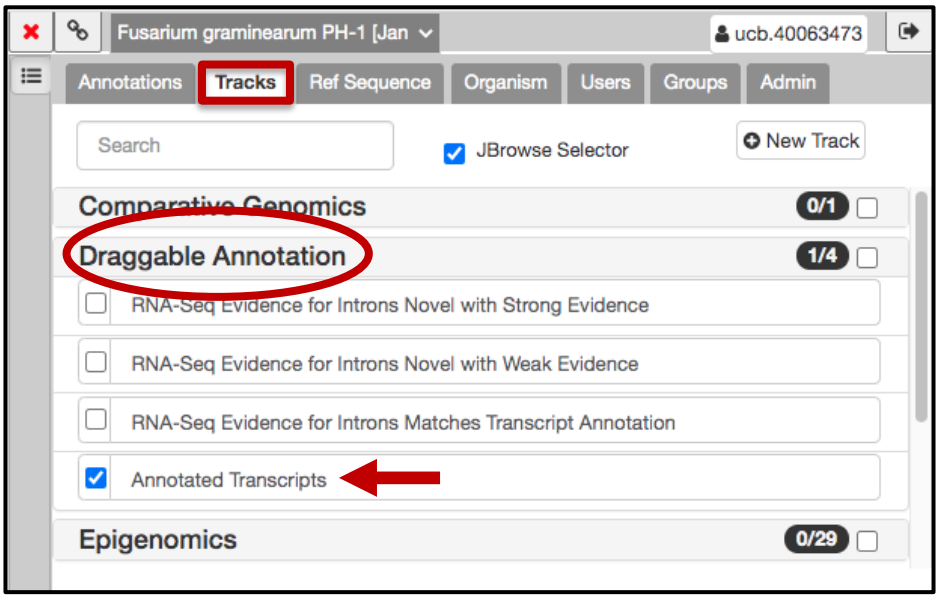
To use Apollo, you need to be logged into VEuPathDB. If you have not done so yet log now into Apollo with your VEuPathDB user ID and password.

2) Open the genome that you want to annotate

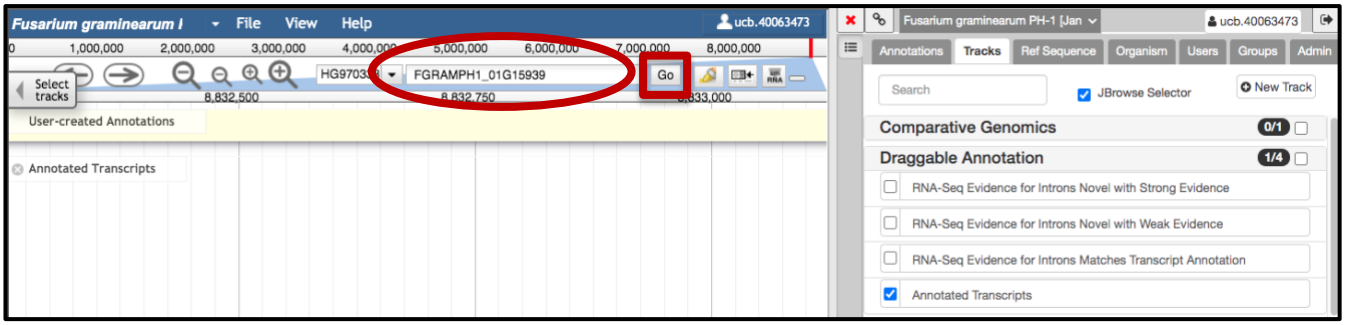
Select on the right-hand side from the drop-down menu the genome that you would like to annotate.



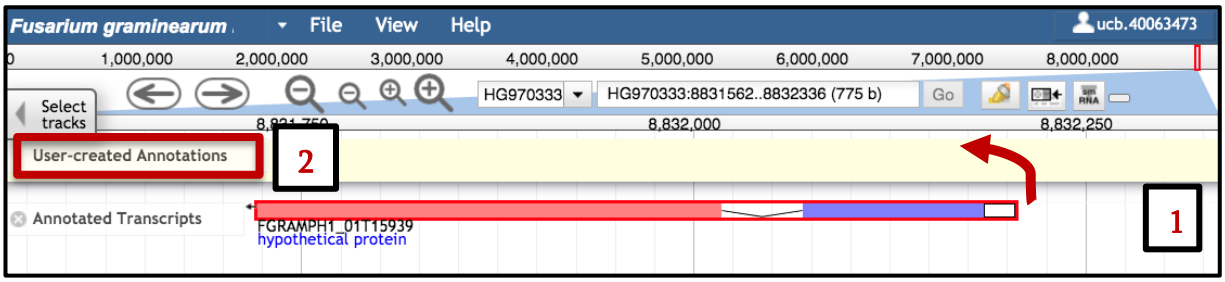
Select the tab **Tracks**, click on the menu item **Draggable Annotation** and select **Annotated Transcripts**.



Type in the search box the gene ID of your gene of interest, wait a few seconds until the gene ID has been found and then click on Go.

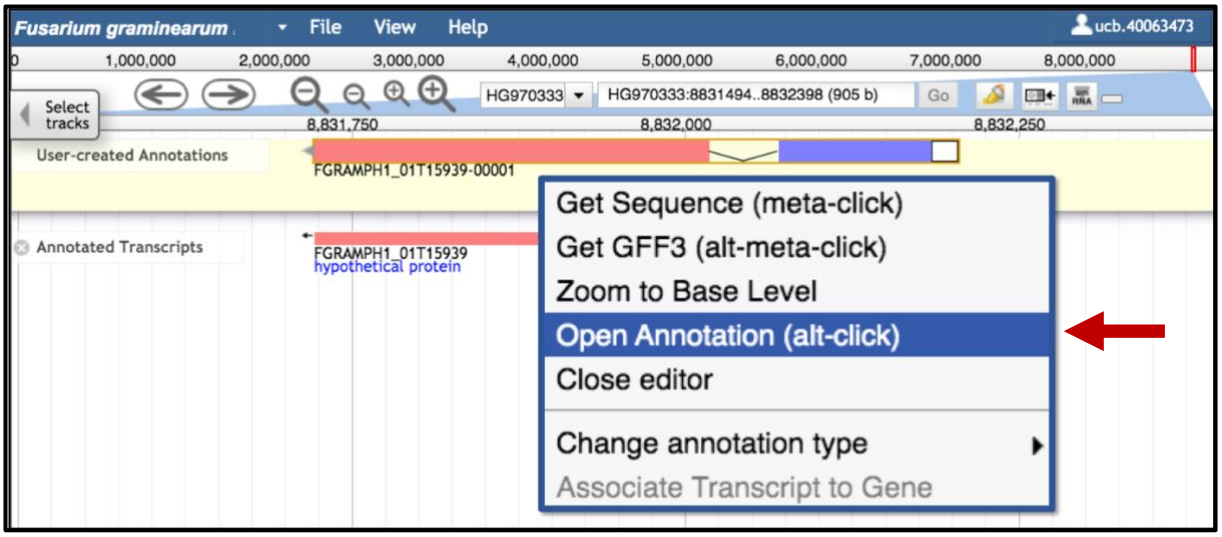


You can now see your gene of interest in the Annotated Transcript track. Select the gene model by clicking on one of the introns or by double clicking on the gene model (**1**). The gene model will show up with red boundaries. Drag and drop the gene into the User-created Annotations area (**2**).

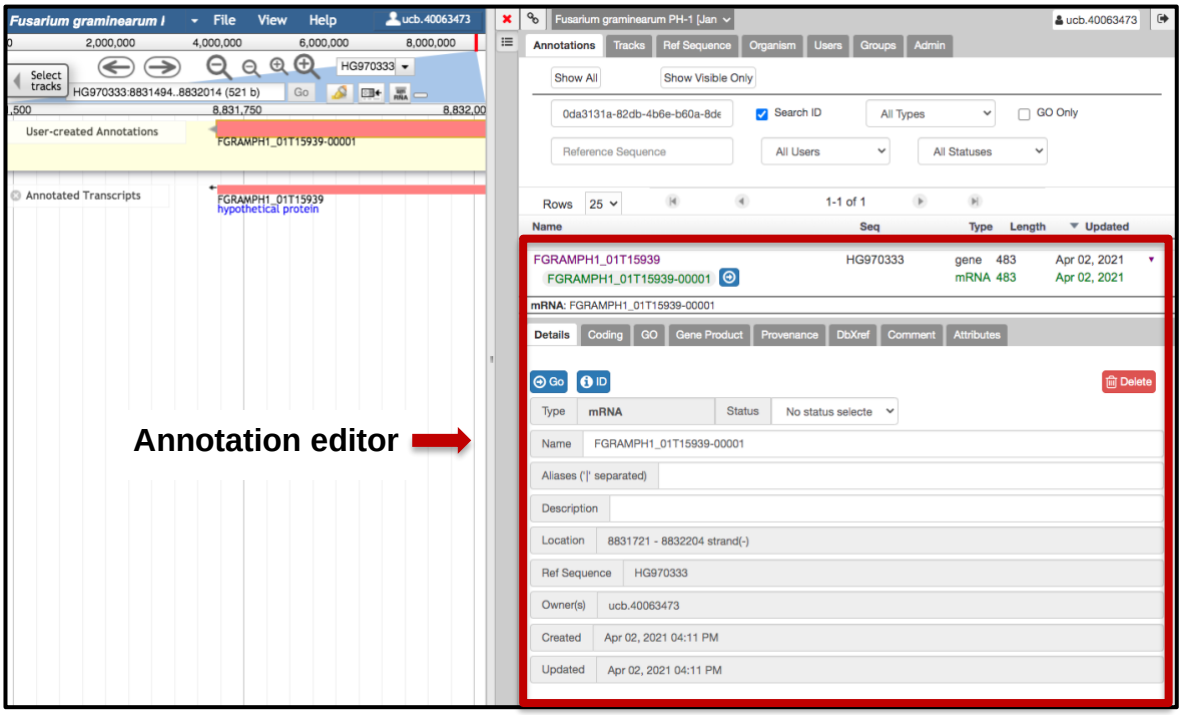


3) Opening of the Annotation editor window

Select the gene in the User-created Annotation track and with a right-click open the drop-down menu and choose **Open Annotation**. Alternatively, you can use the short-cut **alt-click**.



The annotation editor window is now shown on the right-hand side.



You can either selec the gene or the mRNA.

Please note functional annotation shoul be added to the gene. For genes with alternative transcripts add th information to th mRNA.

4) Adding Functional annotation

In this example we are showing you how to improve the functional annotation of FGRAMPH1_01G15939. This gene is currently annotated as hypothetical protein. It has been experimentally characterised in the following publication:

https://pubmed.ncbi.nlm.nih.gov/32873802/

Gene name/symbol: OSP24

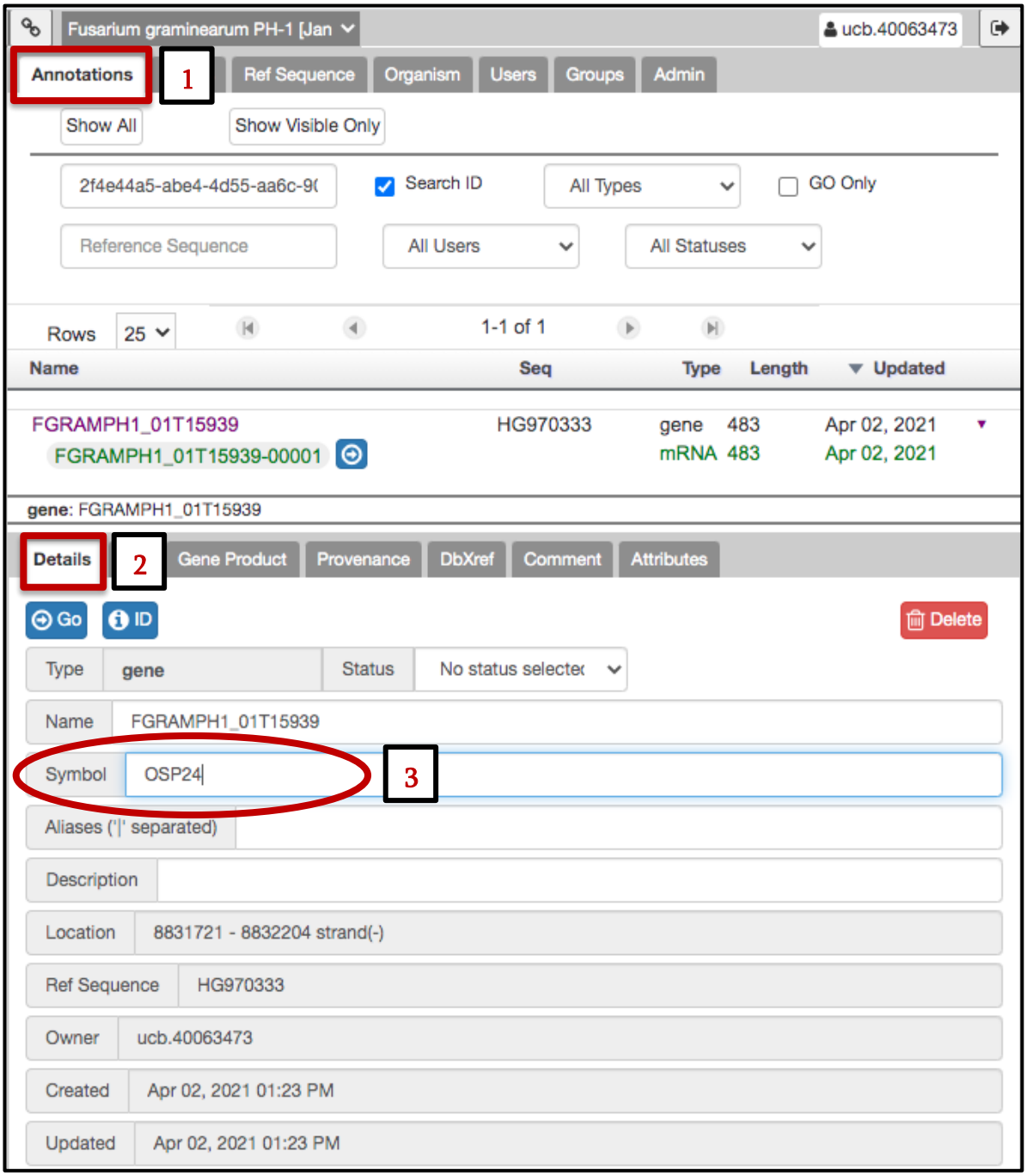
Gene product: Orphan secreted protein 24

GO term: cytoplasm

PMID: 32873802

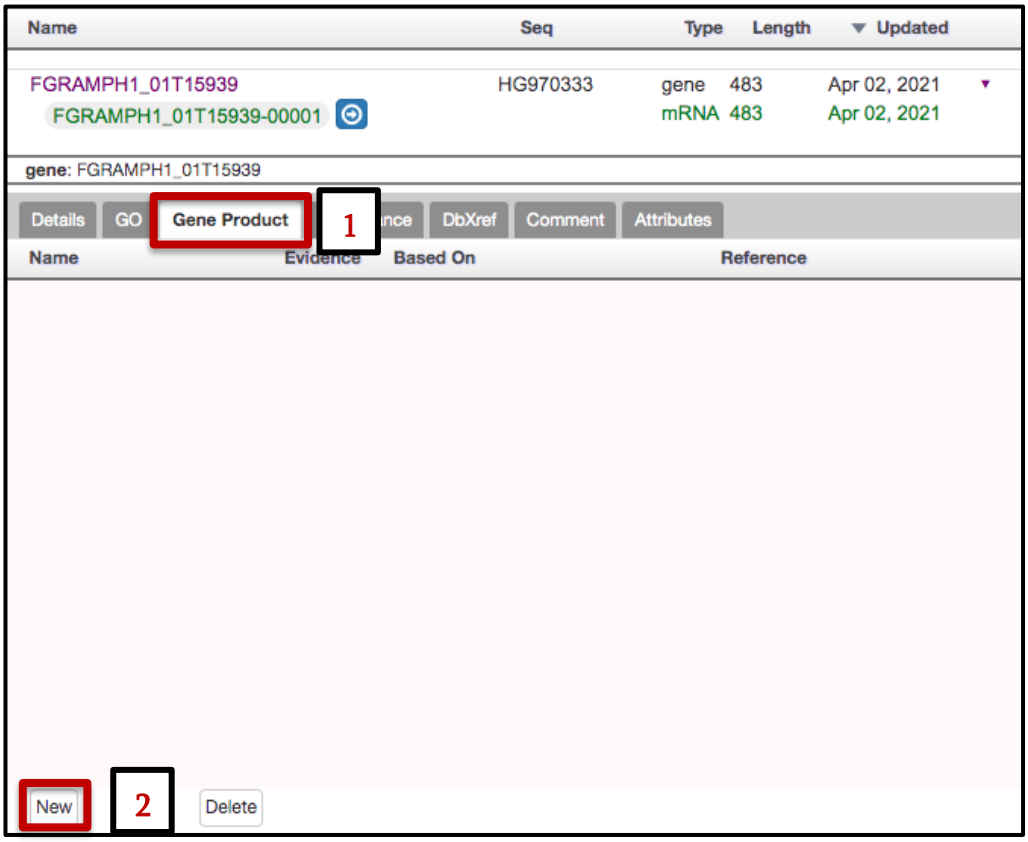
Adding a gene name/symbol

Once the annotations panel (**1**) is open click on the details tab (**2**) and add the gene name in the field Symbol (**3**). In our example the new gene name/symbol is OSP24.

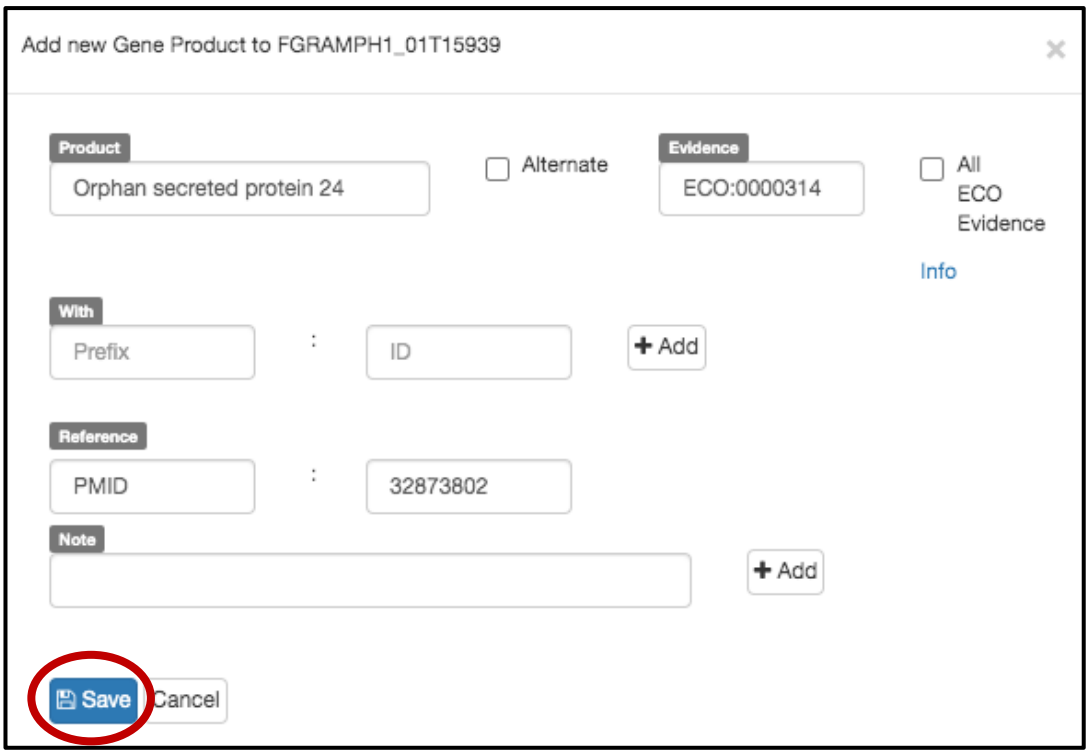


Adding a product description

To add a product description, choose the tab Gene Product (**1**) and click on **New** at the bottom of the editor window (**2**).

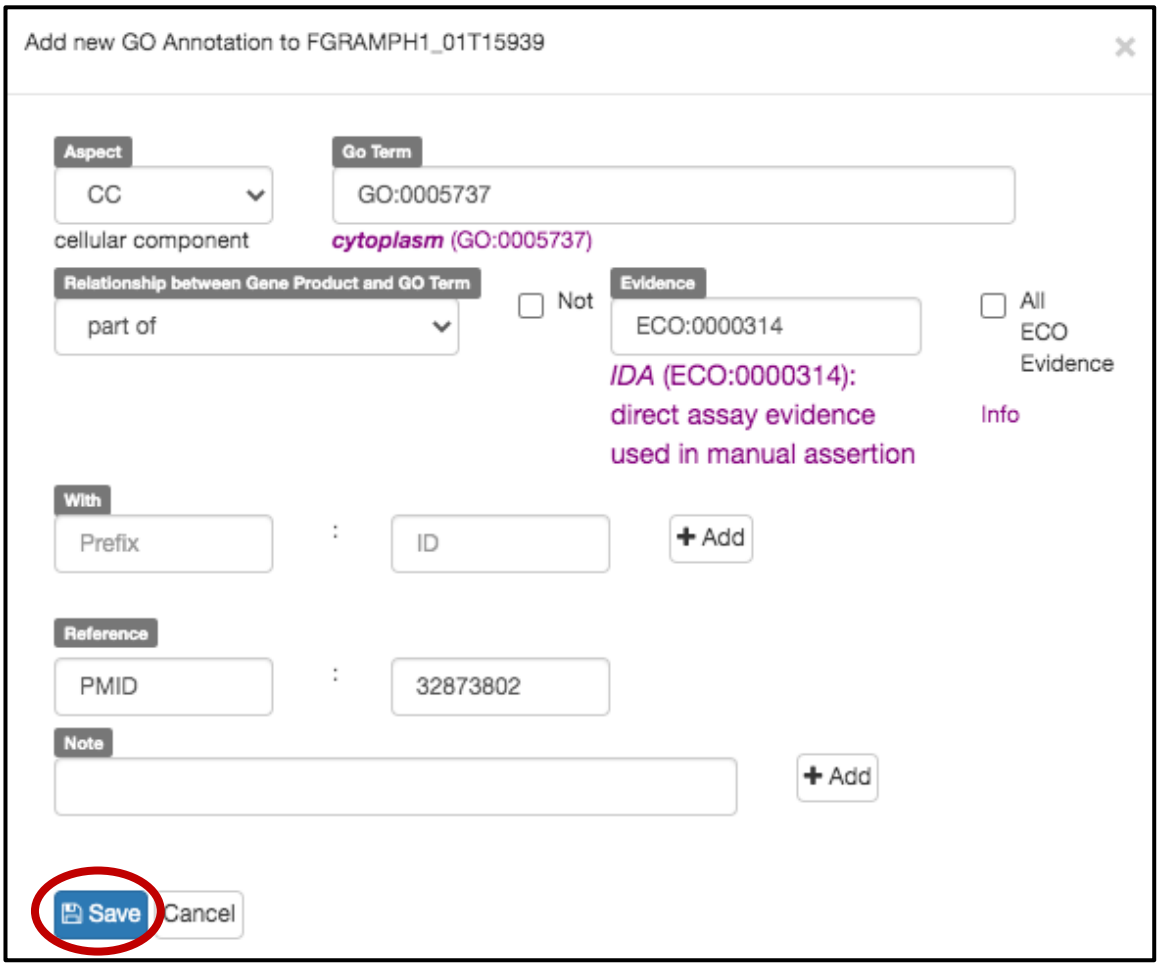


Fill in the fields for product, evidence and PMID. Click on Save. More information about evidence codes can be found here: http://geneontology.org/docs/guide-go-evidence-codes



Adding GO terms

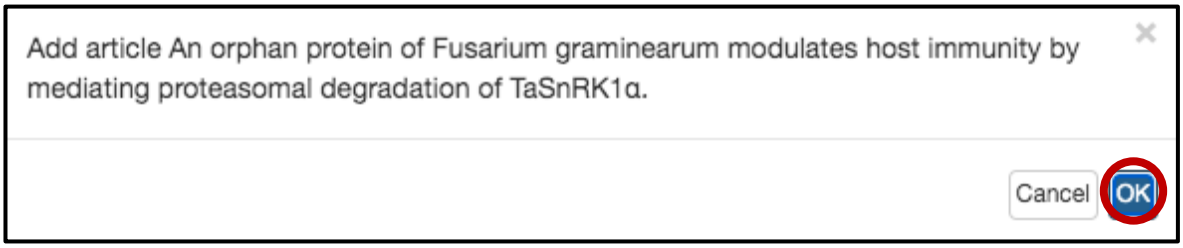
Choose the tab GO in the editor window, fill in the required fields and click on Save.



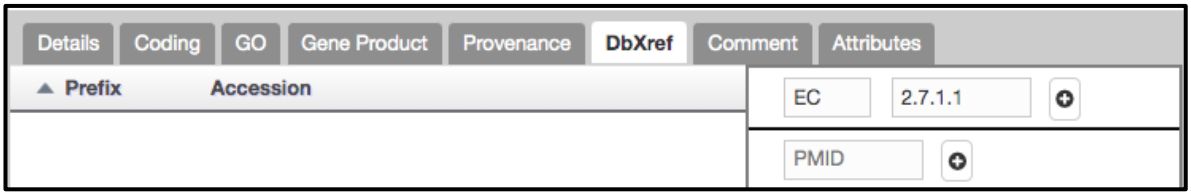
Adding a PubMed ID

Choose the Tab DbXref in the annotation editor window, add the PMID as shown in the screenshot. Click on the **+** sign.

A small window will come up showing the title of the Pubmed Article. Click OK.



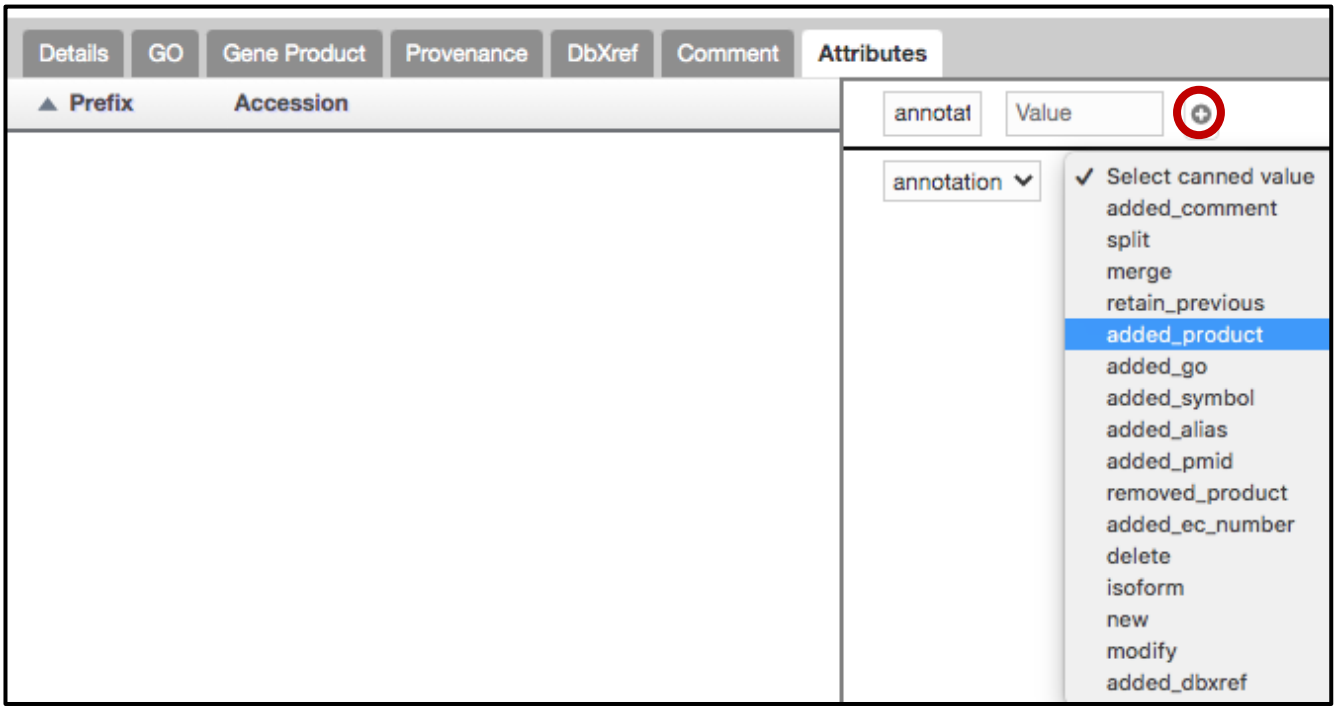
You can also add additional database identifiers (DbXref), i.e. EC numbers.

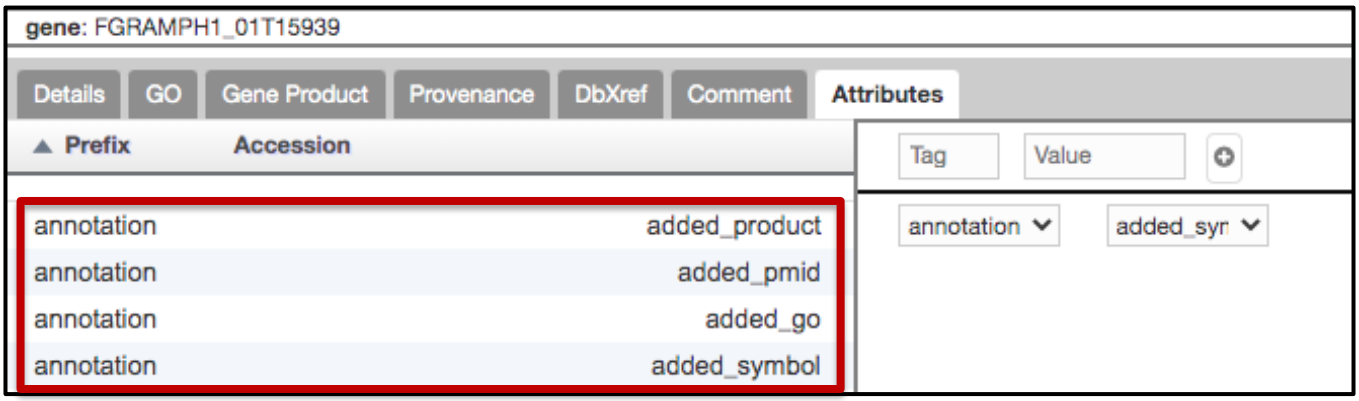


5) Finalising the functional annotation

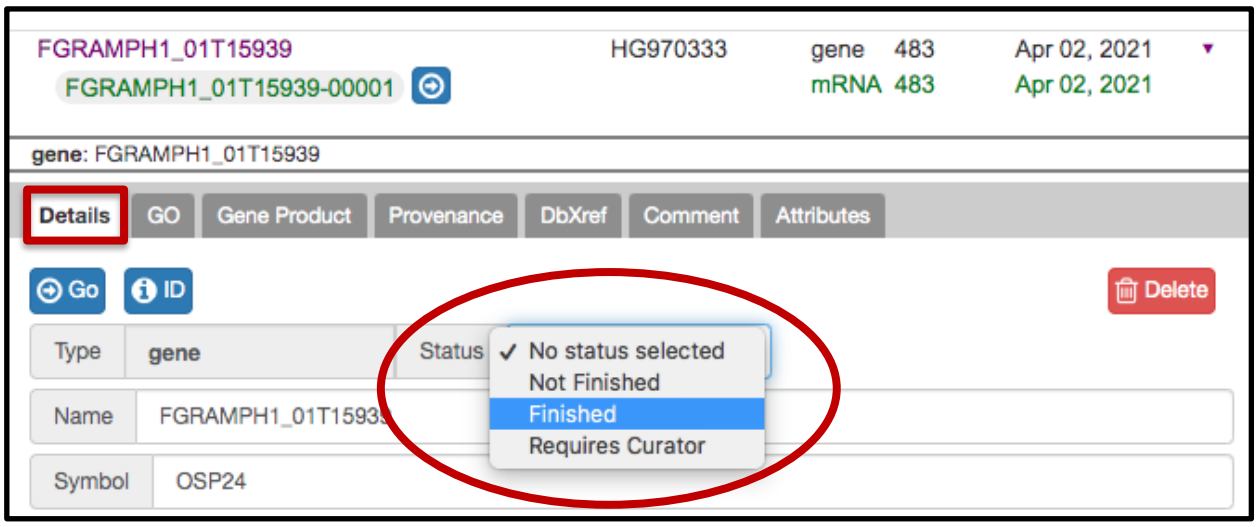
Go to the **Attributes** tab in the gene section, choose from the "Select canned tag" drop-down menu **annotation**.

From the "Select canned value" drop-down menu choose **added_product**. Repeat this and choose **added_symbol, added_go** and **added_pmid**. Finally click on the **+** sign.





In the last step, go back to the Details tab and add to the gene and mRNA the status **Finished**.



Done! For additional questions, please get in touch with the FungiDB help desk help@fungidb.org