

---

# **HymenopteraMine Documentation**

***Release 1.0***

**Aditi Tayal, Deepak Unni, Colin Diesh, Chris Elsik, Darren Hagen**

**Apr 06, 2017**



---

## Contents

---

<b>1</b>	<b>Welcome to HymenopteraMine</b>	<b>3</b>
1.1	Overview of HymenopteraMine . . . . .	3
1.2	Searching in HymenopteraMine . . . . .	4
1.3	Report Page . . . . .	12
1.4	Lists . . . . .	17
1.5	API . . . . .	19
1.6	Data Sources . . . . .	20
1.7	BLAST . . . . .	20
<b>2</b>	<b>Indices and tables</b>	<b>21</b>



HymenopteraMine is an integrative resource for genomic data on Hymenoptera, including honeybees, ants, wasps, etc.

Link to main site: <http://hymenopteragenome.org/hymenopteramine>

Link to PDF version of this documentation: <https://media.readthedocs.org/pdf/hymenopteramine/stable/hymenopteramine.pdf>

Link to the available datasets in HymenopteraMine: <http://hymenopteragenome.org/hymenopteramine/dataCategories.do>

Contents:



# CHAPTER 1

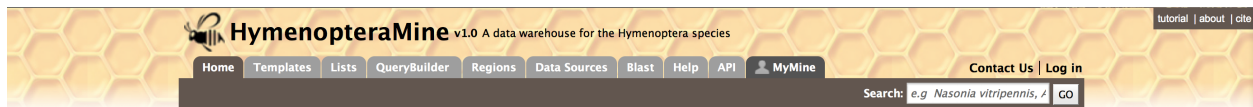
---

## Welcome to HymenopteraMine

---

HymenopteraMine is powered by InterMine and provides a user-friendly way to access genomic, proteomic, interaction and literature data. HymenopteraMine is a part of the Hymenoptera Genome Database.

This tutorial is aimed at giving users an introduction to the different parts of HymenopteraMine and how users can make the most of HymenopteraMine.



## Overview of HymenopteraMine

Before going into details, please go through this brief summary about the layout of HymenopteraMine.

*Home* – The home page for HymenopteraMine.

*Templates* – Shows the lists of templates that users can use based on the nature of their query.

*Lists* – Takes the user to a list page where they can upload their list of genes and perform enrichment analyses. Users, once logged in, can also save their lists for future use.

*QueryBuilder* – page where users can build custom queries by browsing the HymenopteraMine data model and customize their results. Users can also download the created query as XML.

*Regions* – a Genomic Region Search page where users can enter genomic coordinates and fetch features that fall within the interval. Users can also extend the interval to increase the range of search.

*Data sources* – a page where all the data sources, their links and the date of download is specified in a tabulated format.

*Blast* – a Blast page where users can BLAST their sequence(s) of interest with the Hymenoptera species reference genome, CDS sequences and protein sequences. *Help* – a page for guidance, when working through Hymenoptermine. *API* – a page where users can discover more about the InterMine API to programmatically access HymenopteraMine.

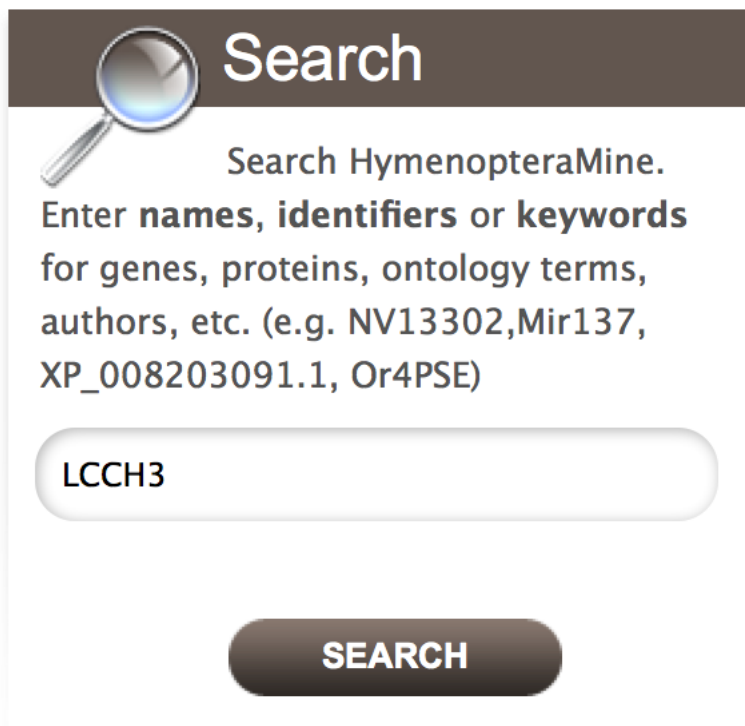
*MyMine* – Once users are logged in, MyMine serves as portal for accessing saved lists and saved templates. Users can also check their account details and manage their account using MyMine.

## Searching in HymenopteraMine

There are several different ways for users to query HymenopteraMine.

### ‘Quick-search’

The ‘Quick-search’ enables users to search keywords from any of the datasets on HymenopteraMine. The quick-search box is on the main page and on the upper-right hand corner of each page

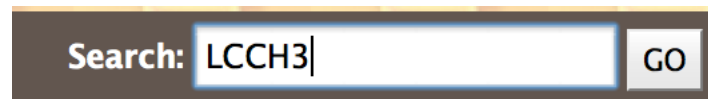


**Search**

Search HymenopteraMine.  
Enter **names, identifiers or keywords**  
for genes, proteins, ontology terms,  
authors, etc. (e.g. NV13302, Mir137,  
XP\_008203091.1, Or4PSE)

LCCH3

**SEARCH**



Search: LCCH3 | **GO**

Users can input their gene name, gene identifiers or keywords and search HymenopteraMine. Users can make use of wildcard character ‘\*’ to get all results that matches their query.

Lets consider gene ‘LCCH3’ as an example. Enter ‘LCCH3’ in the search box and click ‘Search’. The results page is tabulated and shows a summary about your query.



**HymenopteraMine** v1.0 A data warehouse for the Hymenoptera species

tutorial | about | cite

Home Templates Lists QueryBuilder Regions Data Sources Blast Help API MyMine Contact Us Log in

Search: e.g. *Nasonia vitripennis*, / GO

### Search our database by keyword


LCCH3

[Back to index](#) [Search \(with current restrictions\)](#) - or - [Search](#)

### Examples

- Search this entire website. Enter **identifiers, names or keywords** for genes, pathways, authors, ontology terms, etc. (e.g. *eve*, *embryo*, *zen*, *allele*)
- Use **OR** to search for either of two terms (e.g. *fly OR drosophila*) or quotation marks to search for phrases (e.g. *"dna binding"*).
- Boolean search syntax** is supported: e.g. *dros\** for partial matches or *fly AND NOT embryo* to exclude a term

### Search results 1 to 2 out of 2 for *LCCH3*

Category restricted to Gene 

0.758s

#### Categories

Category: *Gene*  
« show all

Hits by Organism

- A. mellifera: 1
- N. vitripennis: 1

Type	Details	Score
<input checked="" type="checkbox"/> Gene	<b>412740   GB45542   LCCH3</b> Source: RefSeq Status: Protein Coding Additional Identifiers: GB12078(OGSv1.0)GB45541GB45542(OGSv3.2) Length: 11679 Chromosome: Group9.1: 35438-47116 Location: A. mellifera Organism:	.....
<input checked="" type="checkbox"/> Gene	<b>100116310   NV14245   LCCH3</b> Source: RefSeq Status: Protein Coding Length: 11633 Chromosome: na: 883712-895344 Location: N. vitripennis Organism:	.....

Users can filter the results based on ‘Category’ or ‘Organism’. The score column in the result table indicates the similarity of your query to the result fetched by HymenopteraMine.

The results page can also be converted to a list. To enable this feature click on ‘Gene’ in ‘Hits by Category’, marked with a red box.

**HymenopteraMine** v1.0 A data warehouse for the Hymenoptera species

tutorial | about | cite

Home Templates Lists QueryBuilder Regions Data Sources Blast Help API MyMine Contact Us Log in

Search: e.g. *Nasonia vitripennis*, / GO

### Search our database by keyword

LCCH3

[Back to index](#) [Search](#)

### Examples

- Search this entire website. Enter **identifiers, names or keywords** for genes, pathways, authors, ontology terms, etc. (e.g. *eve*, *embryo*, *zen*, *allele*)
- Use **OR** to search for either of two terms (e.g. *fly OR drosophila*) or quotation marks to search for phrases (e.g. *"dna binding"*).
- Boolean search syntax** is supported: e.g. *dros\** for partial matches or *fly AND NOT embryo* to exclude a term

### Search results 1 to 6 out of 6 for *LCCH3*

0.696s

#### Categories

Hits by Category

- mRNA: 3
- Gene: 2
- Publication: 1

Hits by Organism

- N. vitripennis: 3
- A. mellifera: 2

Type	Details	Score
Gene	<b>412740   GB45542   LCCH3</b> Source: RefSeq Status: Protein Coding Additional Identifiers: GB12078(OGSv1.0)GB45541GB45542(OGSv3.2) Length: 11679 Chromosome: Group9.1: 35438-47116 Location: A. mellifera Organism:	.....
Gene	<b>100116310   NV14245   LCCH3</b> Source: RefSeq Status: Protein Coding Length: 11633 Chromosome: na: 883712-895344 Location: N. vitripennis Organism:	.....
mRNA	<b>NM_001077812.1</b> Length: 11679 Chromosome: Group9.1: 35438-47116 Location:	.....
mRNA	<b>XM_008205080.1</b> Length: 11633 Location:	.....

This will filter the results for the feature type ‘Gene’ and checkboxes will be available for users to select genes they would like to add to their list. Once the genes are selected, click on ‘CREATE LIST’. List features described later.

## Templates

Apart from the search box, users can make use of predefined templates by clicking on the ‘Templates’ tab.

The Templates page provides users with a list of templates users can choose based on the nature of their query.

**Gene → GO Terms**

Show genes annotated with a specified GO term. Optionally constrain the organism. Also various GO Evidence Codes(Inferred from Direct Assay (IDA),Inferred from Electronic Annotation (IEA),Inferred from Mutant Phenotype (IMP),Inferred from Sequence or structural Similarity (ISS),Non-traceable Author Statement (NAS),Inferred by Curator (IC)) can be used to refine the search

**Organism > Name**

=  ⓘ

**Ontology Term > Name**

CONTAINS

**GO Evidence Code > Code**

=  ⓘ

**Show Results** Edit Query

[web service URL](#) [Perl](#) [Python](#) [Ruby](#) [Java \[help\]](#) [export XML](#)

Users can also specify a filter for GO evidence code. Lets search for the GO term ‘protein’ with GO evidence code ‘IEA’. Click on ‘Show Results’.

Trail: Query

**Gene → GO Terms**

Show genes annotated with a specified GO term. Optionally constrain the organism. Also various GO Evidence Codes(Inferred from Direct Assay (IDA),Inferred from Electronic Annotation (IEA),Inferred from Mutant Phenotype (IMP),Inferred from Sequence or structural Similarity (ISS),Non-traceable Author Statement (NAS),Inferred by Curator (IC)) can be used to refine the search <sup>3</sup>

<sup>4</sup> [Manage Columns](#) [3 Filters](#) [Create / Add to List](#) [Get Code <sup>2</sup>](#) [Download](#)

Showing 1 to 25 of 100 rows Rows per page: 25

DB Identifier	Symbol	GO Annotation Qualifier	Code Code	Code Id	Ontology Term Identifier	Ontology Annotations Id	Ontology Term Name	Secondary Identifier	Name	Organism
100577393	NO VALUE	NO VALUE	IEA	18000030	GO:0004714	18001316	transmembrane receptor protein tyrosine kinase activity	GB54477	NO VALUE	A. mellifera
100577393	NO VALUE	NO VALUE	IEA	18000030	GO:0004714	18001319	transmembrane receptor protein tyrosine kinase signaling pathway	GB54477	NO VALUE	A. mellifera
100577393	NO VALUE	NO VALUE	IEA	18000030	GO:0005524	18001316	transmembrane receptor protein tyrosine kinase activity	GB54477	NO VALUE	A. mellifera
100577393	NO VALUE	NO VALUE	IEA	18000030	GO:0005524	18001319	transmembrane receptor protein tyrosine kinase signaling pathway	GB54477	NO VALUE	A. mellifera
100577393	NO VALUE	NO VALUE	IEA	18000030	GO:0007169	18001316	transmembrane receptor protein tyrosine kinase activity	GB54477	NO VALUE	A. mellifera
100577393	NO VALUE	NO VALUE	IEA	18000030	GO:0007169	18001319	transmembrane receptor protein tyrosine kinase signaling pathway	GB54477	NO VALUE	A. mellifera
100577393	NO VALUE	NO VALUE	IEA	18000030	GO:0007275	18001316	transmembrane receptor protein tyrosine kinase activity	GB54477	NO VALUE	A. mellifera
100577393	NO VALUE	NO VALUE	IEA	18000030	GO:0007275	18001319	transmembrane receptor protein tyrosine kinase signaling pathway	GB54477	NO VALUE	A. mellifera
100577393	NO VALUE	NO VALUE	IEA	18000030	GO:0016021	18001316	transmembrane receptor protein tyrosine kinase activity	GB54477	NO VALUE	A. mellifera
100577393	NO VALUE	NO VALUE	IEA	18000030	GO:0016021	18001319	transmembrane receptor protein tyrosine kinase signaling pathway	GB54477	NO VALUE	A. mellifera
100577393	NO VALUE	NO VALUE	IEA	18000030	GO:0004714	18001316	transmembrane receptor protein tyrosine kinase activity	GB54477	NO VALUE	A. mellifera

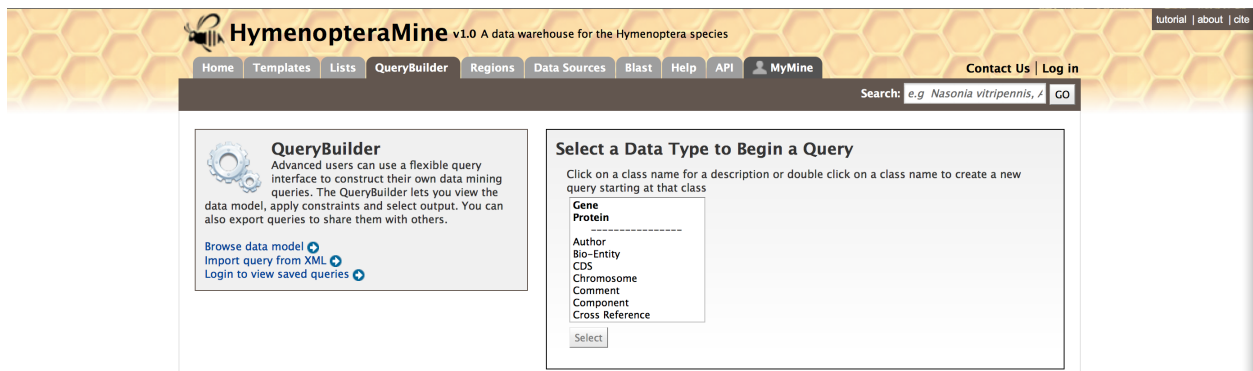
The result page shows all the genes having the Gene Ontology term ‘protein’ in their annotation. Users can create or add these genes to a list, by clicking on ‘Create/Add to List’ (Box 1) to perform further analyses. List function described later Users can get the code for the query in Perl, Python, Java, Ruby, JavaScript or XML by clicking on ‘Get Code’ (Box 2). Users can also download the search results, by clicking on ‘Download’ (Box 3), as tab-delimited, comma-separated values, XML or JSON. If the results are genomic features, which is true for the current example,

then users can download the results in GFF3 and BED format.

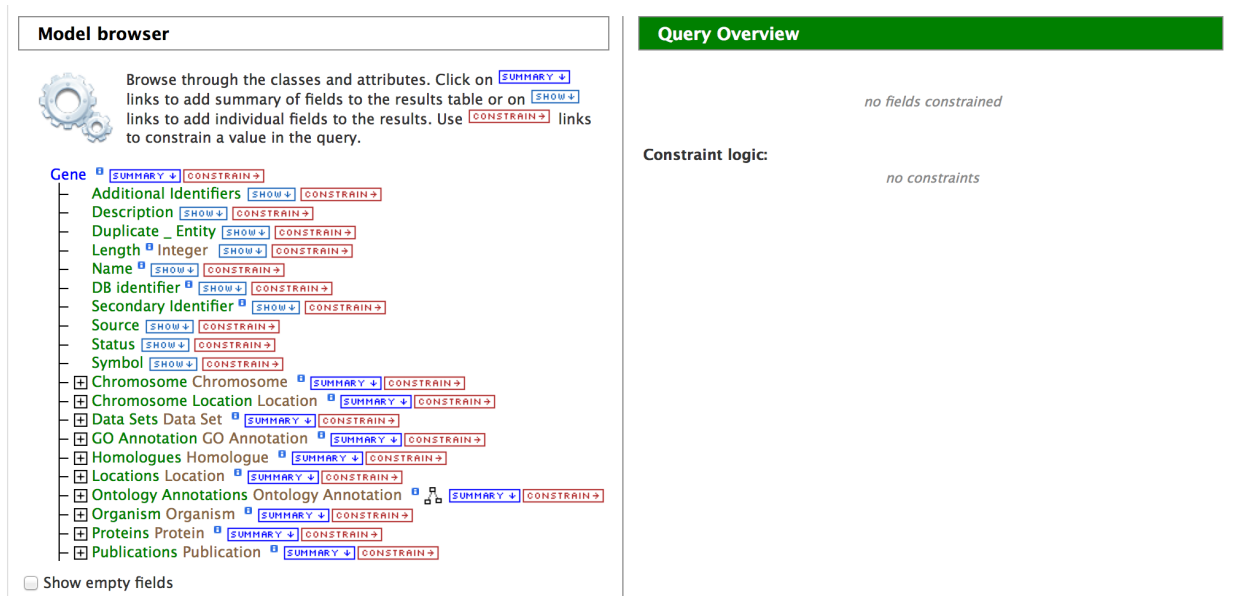
Users can also customize the layout of the result page by clicking on ‘Manage Columns’ (Box 4).

## Query Builder

We have provided templates suitable for different types of queries but if users need more fidelity in their search they can make use of the QueryBuilder. The possibilities of queries using the QueryBuilder are endless. You can format the output the way you want and constrain your queries to perform complex search operations.



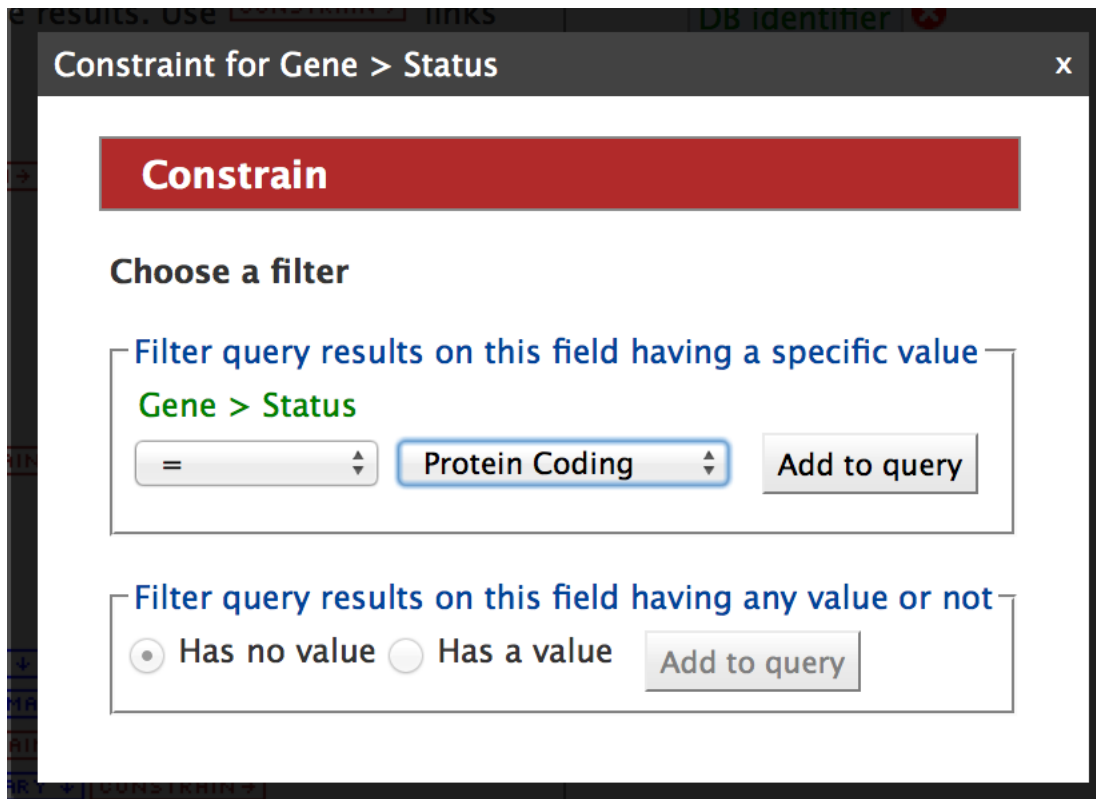
First lets select ‘Gene’ as a Data Type in the QueryBuilder. Then click on ‘Select’. This will take you to a Model browser where you can select the attributes for the feature class ‘Gene’.



Lets consider three scenarios for using the QueryBuilder,

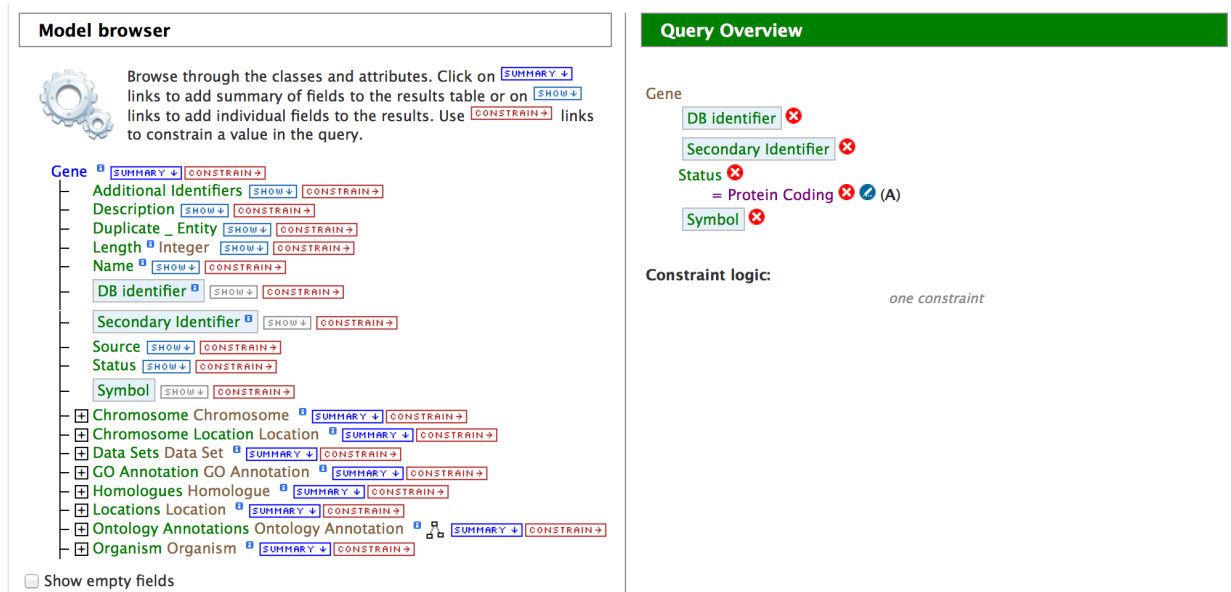
Querying for Protein Coding genes:

Click on ‘Show’ tab for the attributes ‘DB identifier’, ‘Secondary Identifier’, and ‘Symbol’. This indicates the QueryBuilder to show the DB identifier (Primary Identifier), Secondary Identifier and the Symbol for ‘Gene’. Then click on ‘Constrain’ tab for the attribute ‘Status’.



Click on the drop down list for Status and select 'Protein Coding'. Then click on 'Add to Query'.

The Model Browser should resemble the image below,



Now if you click on 'Show results' it would show all the genes that have status 'Protein Coding' as shown below:

Trail: [Query](#) > Results

[Manage Columns](#) [1 Filters](#)

[Create / Add to List](#) [Get Code](#) [Download](#)

Showing 1 to 25 of 150,674 rows Rows per page: 25

DB identifier	Secondary Identifier	Symbol
100037419	GB53114	Apd-3
100049551	GB45446	Burs
100101930	NV14617	Para
100101931	NV15292	Hb
100102554	NV10192	Gt
100104006	NV18258	nanos
100104126	NV15848	Eve
100107367	NV16623	NO VALUE
100110062	NV15145	NO VALUE
100110095	NV15146	Scr
100112394	NV12575	NO VALUE
100113483	NV19052	NO VALUE

ii. Querying for Protein Coding genes on a particular chromosome Users can customize this query by adding another constraint for Chromosome.

**Model browser**

Browse through the classes and attributes. Click on [SUMMARY](#) links to add summary of fields to the results table or on [SHOW](#) links to add individual fields to the results. Use [CONSTRAIN](#) links to constrain a value in the query.

- Gene
  - [SUMMARY](#) [CONSTRAIN](#)
  - Additional Identifiers [SHOW](#) [CONSTRAIN](#)
  - Description [SHOW](#) [CONSTRAIN](#)
  - Duplicate \_ Entity [SHOW](#) [CONSTRAIN](#)
  - Length Integer [SHOW](#) [CONSTRAIN](#)
  - Name [SHOW](#) [CONSTRAIN](#)
  - DB identifier [SHOW](#) [CONSTRAIN](#)
  - Secondary Identifier [SHOW](#) [CONSTRAIN](#)
  - Source [SHOW](#) [CONSTRAIN](#)
  - Status [SHOW](#) [CONSTRAIN](#)
  - Symbol [SHOW](#) [CONSTRAIN](#)
  - Chromosome Chromosome [SUMMARY](#) [CONSTRAIN](#) ←
  - Length Integer [SHOW](#) [CONSTRAIN](#) ←
  - DB identifier [SHOW](#) [CONSTRAIN](#)
  - Secondary Identifier [SHOW](#) [CONSTRAIN](#)
  - Data Sets Data Set [SUMMARY](#) [CONSTRAIN](#)
  - Located Features Location [SUMMARY](#) [CONSTRAIN](#)
  - Organism Organism [SUMMARY](#) [CONSTRAIN](#)
  - Sequence Sequence [SUMMARY](#) [CONSTRAIN](#)

☐ Show empty fields

**Query Overview**

Gene

- DB identifier ✖
- Secondary Identifier ✖
- Status ✖
- = Protein Coding ✖ (A)
- Symbol ✖

Constraint logic:

one constraint

While building the query in the QueryBuilder, expand the 'Chromosome' feature class and select the 'Constrain' tab for attribute 'DB Identifier'. Enter 'Group1.1' in the text box of the pop-up window and then click on 'Add to Query'. Now if you click on 'Show results' the query will result in all the genes that are of type 'Protein Coding' on Chromosome with DB Identifier 'Group1.1'.

Trail: [Query](#) > Results

Manage Columns

2 Filters

Create / Add to ListGet CodeDownload

Showing 1 to 25 of 107 rows

Rows per page: 25

DB identifier

Secondary Identifier

Symbol

100216325	NO VALUE	Emc
100577692	GB42180	NO VALUE
100578031	GB42172	NO VALUE
100578334	GB42177	NO VALUE
100578571	GB42182	NO VALUE
100578600	GB42183	NO VALUE
100578621	GB42184	NO VALUE
100642190	NO VALUE	NO VALUE
100642312	NO VALUE	NO VALUE
100642313	NO VALUE	NO VALUE

iii. Querying for Protein Coding genes on a particular chromosome and theirs exons Users can customize this further by configuring the query to show all the exons for each Gene.

Model browser

Length Integer

DB identifier

Symbol

Chromosome Chromosome

Chromosome Location Location

Data Sets Data Set

Exons Exon

Transcripts Transcript

Length Integer

DB identifier

Symbol

Chromosome Chromosome

Chromosome Location Location

Data Sets Data Set

Locations Location

Organism Organism

Sequence Ontology Term SO Term

Synonyms Synonym

Show empty fields

Query Overview

Gene

DB identifier

Secondary Identifier

Status

Symbol

Chromosome Chromosome

DB identifier

Transcripts Transcript collection

Exons Exon collection

Length

DB identifier

Constraint logic: A and B

A and B

Expand the 'Exon' feature class and select on 'DB identifier' and 'Length'. To make the output more visually pleasing, click on the blue square near Exon collection in the query overview and you will see a popup like below,

Switch Join Style Gene > Transcripts > Exons x

Constrain

Filter options

☐ Show only Transcripts if they have a Exon.
 ☒ Show all Transcripts and show Exons if they are present.

Add to query

Select 'Show all Genes and show Exons if they are present' and click on 'Add to query'.

Now run the query and you should see the following results page,

Trail: [Query](#) > Results

[Manage Columns](#)
[2 Filters](#)
[Create / Add to List](#)
[Get Code](#)
[Download](#)

Showing 1 to 25 of 242 rows Rows per page: 25

DB identifier	Secondary Identifier	Symbol	Transcripts Exons
100216325	NO VALUE	Emc	1 Exons
100577692	GB42180	NO VALUE	3 Exons
100578031	GB42172	NO VALUE	6 Exons
100578334	GB42177	NO VALUE	3 Exons
100578571	GB42182	NO VALUE	6 Exons
100578600	GB42183	NO VALUE	4 Exons
100578621	GB42184	NO VALUE	3 Exons
100578621	GB42184	NO VALUE	2 Exons
100578621	GB42184	NO VALUE	3 Exons
100578621	GB42184	NO VALUE	3 Exons

There are 107 genes on Chromosome Group1.1 and the gene with NCBI GeneID 100578031 has 6 exons. Users can click on the '6 exons' to expand the table with additional rows describing the length and DB identifier for each of the 6 exons.

Trail: [Query](#) > Results[Manage Columns](#) [2 Filters](#)[Create / Add to List](#)[Get Code](#)[Download](#)

Showing 1 to 25 of 242 rows

Rows per page: 25

p. 1

<div>⬆ ⬇ ⬅ 🔍</div>	<div>⬆ ⬇ ⬅ 🔍</div>	<div>⬆ ⬇ ⬅ 🔍</div>	<div>⬆ ⬇ ⬅ 🔍</div>														
DB identifier	Secondary Identifier	Symbol	Transcripts <div>▾</div> <div>Exons</div>														
100216325	NO VALUE	Emc	<div>▮</div> 1 Exons														
100577692	GB42180	NO VALUE	<div>▮</div> 3 Exons														
100578031	GB42172	NO VALUE	<div>▮</div> 6 Exons														
			<table><tr><th>Length</th><th>DB identifier</th></tr><tr><td>562</td><td>exon1797</td></tr><tr><td>230</td><td>exon1798</td></tr><tr><td>173</td><td>exon1799</td></tr><tr><td>257</td><td>exon1800</td></tr><tr><td>395</td><td>exon1801</td></tr><tr><td>1801</td><td>exon1802</td></tr></table>	Length	DB identifier	562	exon1797	230	exon1798	173	exon1799	257	exon1800	395	exon1801	1801	exon1802
Length	DB identifier																
562	exon1797																
230	exon1798																
173	exon1799																
257	exon1800																
395	exon1801																
1801	exon1802																

## Report Page

Every query result has a report page and the layout of the report page depends on the data available for a given query. Continuing with the example of ‘Hb’, the report page for this gene is shown below

Gene : **LCCH3** *A. mellifera*DB Identifier [412740](#)Source [RefSeq](#)Additional Identifiers [GB12078\(OGSv1.0\)](#)[GB45541GB45542\(OGSv3.2\)](#)Secondary Identifier [GB45542](#)Status [Protein Coding](#)Description [ligand-gated chloride channel homolog 3](#)[SHARE](#)

Quick Links:

[Summary](#) [Genes](#) [Proteins](#) [Function](#) [Homology](#) [Other](#)

The report page provides a complete description for gene Hb. The header of the report page shows the DB identifier. The ‘status’ indicates the type of gene, in this case a protein coding gene. Other possible values are,

- Non Coding – for non coding RNAs
- Pseudogene – for pseudogenes
- Immunogenes – for protein coding genes that are annotated to be a part of the immune system
- Frameshift – for protein coding genes that have a frameshift in their coding frame
- miRNA – for micro RNAs
- rRNA – for ribosomal RNAs
- tRNA – for transfer RNAs

The contents of the report page are divided into categories based on the type of information provided,

### Summary:

Genome feature			
Region:	<a href="#">gene</a>	Length:	11679
Location:	<a href="#">Group9.1:35438–47116</a>		

Provides a summary about a gene such as length, chromosome location and strand information. Users can also get the complete FASTA sequence of the gene by clicking on the FASTA tab.



## Genes

Gene models – LCCH3 GB45542			
<a href="#">Gene models</a> <a href="#">Transcripts: 1</a> <a href="#">Exons: 9</a> <a href="#">CDSs: 1</a>			
Transcript	Exons		CDSs
LCCH3 NM_001077812.1 11679 <a href="#">FASTA...</a>	exon55	65	<a href="#">FASTA...</a>
	exon56	107	<a href="#">FASTA...</a>
	exon57	69	<a href="#">FASTA...</a>
	exon58	20	<a href="#">FASTA...</a>
	exon59	160	<a href="#">FASTA...</a>
	exon60	285	<a href="#">FASTA...</a>
	exon61	231	<a href="#">FASTA...</a>
	exon62	257	<a href="#">FASTA...</a>
	exon63	153	<a href="#">FASTA...</a>
			NM_001077812.1-CDS 1347 <a href="#">FASTA...</a>

This section provides information about the gene model. It displays all the transcripts and exons for a gene. The FASTA sequence of each transcript or exon can be downloaded by clicking on the FASTA tab. Users can also download the coding sequence for a transcript, when available, by clicking on the FASTA tab in the CDS column.

Gene --> Proteins. (1 rows)			
<a href="#">Manage Columns</a> <a href="#">1 Filters</a>		<a href="#">Create / Add to List</a> <a href="#">Get Code</a> <a href="#">Download</a>	
Showing 1 to 1 of 1 rows		Rows per page: 10 <a href="#">←</a> <a href="#">↶</a> <a href="#">↷</a> <a href="#">→</a> p. 1	
DB identifier	Symbol	Proteins » DB identifier	Proteins » Primary Accession
GB45542	LCCH3	A0A088A2R0_APIME	A0A088A2R0

This section provides information about the protein product of gene LLCH3. The comments section gives a brief description about the protein along with the UniProt accession.

## Function:

### Function

Gene Ontology	
<b>biological process</b>	
sequence-specific DNA binding RNA polymerase II transcription factor activity	IEA
regulation of development, heterochronic	IEA
<b>molecular function</b>	
sequence-specific DNA binding RNA polymerase II transcription factor activity	IEA
metal ion binding	IEA
nucleic acid binding	IEA
<b>cellular component</b>	
nucleus	IEA

This section provides Gene Ontology annotations for a gene. Annotations are divided into three categories,

- Biological process
- Molecular function
- Cellular Component

The GO terms are displayed along with the evidence code indicating how the annotations were derived.

## Homology:

### Homology

Homologues												
A. cephalotes	A. echinator	A. florea	B. impatiens	B. terrestris	C. floridanus	D. melanogaster	H. saltator	L. albipes	L. humile	N. vitripennis	P. barbatus	S. invicta
ACEP21748	AECH24563	Aflo05437	BIMP02380	Bter03851	CFLO21694			Lalb_09151	LH23929		PB11687	
ACEP25093	AECH11650	Aflo12324	BIMP11735	Bter11333	CFLO11493			Lalb_07369	LH23931		PB14979	SINV25217
ACEP14589	AECH27275	Aflo00900	BIMP10335	Bter11206	CFLO14467	FBgn0010240	HSAL17273	Lalb_12187	LH26068	Nasvi2EG008402	PB24104	SINV24648
ACEP20187	AECH12571	Aflo03129	BIMP13177	Bter00358	CFLO22754		HSAL19983	Lalb_11437	LH13052	Nasvi2EG003037	PB20249	SINV12985
ACEP18918	AECH16354	Aflo10066	BIMP13123	Bter00687			HSAL15497	Lalb_02192	LH13060	Nasvi2EG000014	PB17443	SINV17860
							HSAL21660	Lalb_11436	LH17589	Nasvi2EG009083	PB14976	SINV23390
									LH10304		PB10598	

The table below gives a detailed information about the homologue, the type of homologue and from which dataset the information was obtained.

## Others:

This section provides information such as,

Child features – lists all the features that are sub features of the current gene

Flanking regions – lists all the features flanking the current gene

Overlapping features – lists all the features that overlap with current gene

Publications – Publications related to the current gene

Genomic Regions Search

The Genomic Region Search is a tool to fetch features that are within a given set of genomic coordinates or to fetch features that are within a given number of bases flanking a given set of genomic coordinates.

The coordinates have to be of the format,

chromosome\_name:start..end

OR

chromosome\_name:start-end

OR

chromosome\_name start end

**HymenopteraMine** v1.0 A data warehouse for the Hymenoptera species

Home Templates Lists QueryBuilder Regions Data Sources Blast Help API MyMine Contact Us Log in

Search:  GO

### Overlap features search from a new list of Genomic Regions

Search for features that overlap a list of genome coordinates you enter or upload, e.g. scaffold00080:56297..58300

[Genome coordinates help](#)

1. Select Organism:  genome build: not available
2. ☒ Select Feature Types:
 

☒ CDS ☒ Gene ☒ Transcript  
☒ Exon ☒ mRNA ☒ tRNA
3. Type/Paste in genomic regions in ☒ base coordinate ☐ Interbase coordinate  
[\(click to see an example\)](#)

or Upload genomic regions from a .txt file...  
 No file selected.

4. Extend your regions at both sides:
 

1k

10k

100k

1M

10M

[Questions? Comments? Click here!](#)

Click on ‘click to see an example’ for a representative set of genomic coordinates.

Users can extend the regions on either side of the genomic coordinate using the slider or using the textbox.

Users can also select the type of coordinate system they would like to use: base coordinate or interbase coordinate.

Lets consider the example,

## Overlap features search from a new list of Genomic Regions

Search for features that overlap a list of genome coordinates you enter or upload, e.g. `Group1.15:15000..17000`

[Genome coordinates help](#)

1. Select Organism: 

A. mellifera

*genome build: not available*

2. ☒ Select Feature types:

☒ CDS ☒ Exon ☒ Gene

☒ miRNA ☒ mRNA ☒ Primary Transcript

☒ rRNA ☒ Transcript ☒ tRNA

3. Type/Paste in genomic regions in ☒ base coordinate ☐ interbase coordinate 

(click to see an example)

Group1.1:962622..972777  
Group1.15:15520..17553  
GroupUn999:2..939

or Upload genomic regions from a .txt file...

Choose File No file chosen

4. Extend your regions at both sides:

500

1k10k100k1M10M

Reset

Search

Click on the ‘click to see an example’ and extend the region search by 500 bp.

The result page will give a list of features that are present in each of the genomic interval provided in the input.

Export for all regions: 

TAB CSV GFF3 FASTA BED FASTA...

 or Create List by feature type: 

Gene

Go

Page size 10 << First < Prev | Next > Last >>

GENOME REGION	FEATURE	FEATURE TYPE	LOCATION
Group1.1:962122..973277 <div>FASTA...</div> Original input: Group1.1:962622..972777	551203 GB42145	Gene	Group1.1:962138..1134513
<div>TAB CSV GFF3 FASTA BED</div>	XM_006557368.1 XM_006557368.1	mRNA	Group1.1:962138..1134513
Create List by <div>CDS</div> <div>Go</div>	exon1961	Exon	Group1.1:962138..962737
	cds1625	CDS	Group1.1:962622..962737
	GB42147	Gene	Group1.1:962622..972777
	GB42147-RA	mRNA	Group1.1:962622..972777
	unknown identifier	CDS	Group1.1:962622..962737
	exon1960	Exon	Group1.1:962849..963219
	cds1624	CDS	Group1.1:962849..963219
	unknown identifier	CDS	Group1.1:962849..963219
	exon1959	Exon	Group1.1:963292..964064
	cds1623	CDS	Group1.1:963292..964064
	unknown identifier	CDS	Group1.1:963292..964064
	exon1958	Exon	Group1.1:964146..964370


The results can be exported as tab-delimited and comma-separated values. If the results have genomic features then they can be exported in GFF3 or BED format. Users can also export FASTA sequences of the features.

If users are interested in creating a list of particular features from the result page then they can filter based on feature type, shown in red box, and click on ‘Go’.

16

Chapter 1. Welcome to HymenopteraMine

## Lists



### Create a new list

Select the type of list to create and either enter in a list of identifiers or upload identifiers from a file. A search will be performed for all the identifiers in your list.

- Separate identifiers by a **comma, space, tab or new line**.
- Qualify any identifiers that contain whitespace with double quotes like so: "even skipped".

**Select Type:** Gene

**for Organism:** Any

**Type/Paste in identifiers** [\(click to see an example\)](#)

MI0014719, NR\_039003.1, 100526356,  
100120943, MI0014759, MI0014760,  
Mir137, Or4PSE, Mir7

**or Upload identifiers from a .txt file...** Choose File No file chosen

☐ **Match on case**

Reset Create List

Users can create a list of features. The input can either be gene IDs, transcript IDs, gene symbols, etc. The list tool tries to lookup the query throughout the database and will attempt to convert the identifiers to the type selected in the list 'Select Type' option.

Lets try the examples provided. Click on 'Click to see example' link and click on 'Create List'.

Identifier you provided	Matches <a href="#">?</a>							Action <a href="#">?</a>
	primary identifier	symbol	name	length	class	secondary identifier	organism short name	
Mir7	732510	Mir7		87	Gene		A. mellifera	<a href="#">Add</a>
	100526362	Mir7		87	Gene		N. vitripennis	<a href="#">Add</a>
Mir137	100315659	Mir137		100	Gene		A. mellifera	<a href="#">Add</a>
	100526322	Mir137		100	Gene		N. vitripennis	<a href="#">Add</a>

Summary [?](#)[Download summary](#)Direct Hits [?](#)

Identifier you provided	Match <a href="#">?</a>						
	primary identifier	symbol	name	length	class	secondary identifier	organism short name
Or4PSE	100463191	Or4PSE		1535	Gene	NV22749	N. vitripennis
100526356	100526356	Mir281		103	Gene		N. vitripennis
100120943	100120943	100120943		3726	Gene	NV13302	N. vitripennis

The list tool does a lookup of the identifiers and shows you the results. If there are any duplicates, users can decide to add the relevant entries individually. The summary section provides information regarding those identifiers that had a direct hit without any duplicates.

Click on the ones that have proper primary identifier and secondary identifier.

Users can download summary of the converted result.

Then click on ‘save a list of 3 genes’.

This will take users to the following page,

[Manage Columns](#)
[1 Filters](#)

[Create / Add to List](#)
[Get Code](#)
[Download](#)

Rows per page: 10

Showing 1 to 3 of 3 rows

DB identifier	Secondary Identifier	Symbol	Name	Length	Chromosome . Primary Identifier	Chromosome Location . Start	Chromosome Location . End	Organism . Short Name
100120943	NV13302	100120943	NO VALUE	3726	NW_001814904.1	2052084	2055809	N. vitripennis
100463191	NV22749	Or4PSE	NO VALUE	1535	NW_001815348.1	1731682	1733216	N. vitripennis
100526356	NO VALUE	Mir281	NO VALUE	103	Chr5_random049	692904	693006	N. vitripennis

**Orthologues**

A. cephalotes (1) A. mellifera (41)

**External Links**

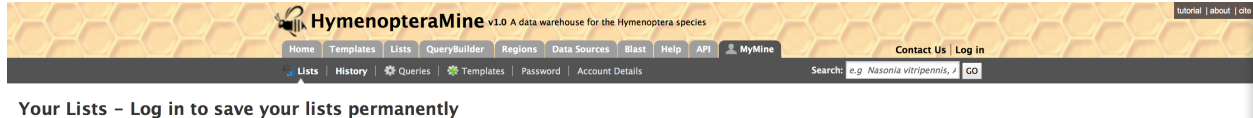
No external links.

This page provides users with widgets to perform analyses on gene lists that they have created.

## List widgets

1. Chromosome Distribution
2. Gene Ontology Enrichment

3. Protein Domain Enrichment
4. Publication Enrichment
5. Pathway Enrichment
6. Orthologues



MyMine serves as a portal for users to manage their lists, queries, templates and account details.

1. Lists – lists saved by the user.
2. History of queries by user – shows a list of most recent queries performed by the user.
3. Templates – Templates created by the user or existing templates that are marked as ‘favorite’ by the user.
4. Password change – change the password for the user’s account.
5. Account details – for updating user preferences.

## API

For users who would like to programmatically access HymenopteraMine, they can make use of the API.

**Perl Web Service Client**

The Perl web service client library makes it easy to run queries in HymenopteraMine directly from Perl programs. You can use these modules to construct any query you could run from web interface and fetch the results in a number of formats, including native Perl data structures and objects, and TSV/CSV strings.

Like all our code, it is open-source, coming licensed under the LGPL. For information on our API, visit our [wiki pages](#).

- Prerequisites**

You should install the [Perl webservice client library module](#) to get started. You can install it directly from CPAN (the comprehensive Perl archive network). This is a public repository of thousands of modules. Using CPAN will ensure you get the latest version, and that your dependencies are automatically managed for you.

To install the InterMine Perl client library type the following command into a shell:

```
> sudo cpan Webservice::InterMine
```
- Examples of using the downloaded script**

On each Template Query page and the QueryBuilder there is a link to get Perl code to run that particular query using the web service API. Just click the link, save the generated Perl script in a file and execute it. You can use the generated code as a starting point for your own programs. You can run the downloaded script by running the following command in a shell:

```
> perl path/to/downloaded/script.pl
```

If you get an error saying `Cannot find Webservice::InterMine in @INC`, or similar then see 'Prerequisites' above.

Feel free to edit the script – these are designed to be spring-boards to help you get where you want to. See the [Cookbook](#), which contains a set of short tutorial 'recipes' that demonstrate particular features of the Perl API, for ways to edit the scripts, and even extend their functionality. For a good reference to writing programs in Perl, see [here](#).

Perl, Python, Ruby and Java are the languages supported by the InterMine API.

For more information on the details of the API visit the InterMine Wiki

## Data Sources

Provides a description of the datasets that are integrated into HymenopteraMine along with their date of download, version or release, citations wherever applicable and any additional comments.

## BLAST

Users can perform BLAST against the Hymenoptera genomic, CDS or protein sequences using the BLAST page. Click [here](#) for a video tutorial, created by FlyMine, which highlights the features of InterMine.



## CHAPTER 2

---

### Indices and tables

---

- `genindex`
- `modindex`
- `search`