Exploring Biological Databases

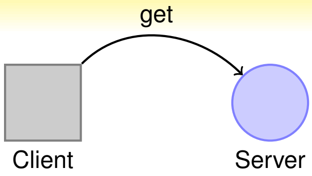
Holger Dinkel

EMBO Practical Course Computational analysis of protein-protein interactions: From sequences to networks

Web Services

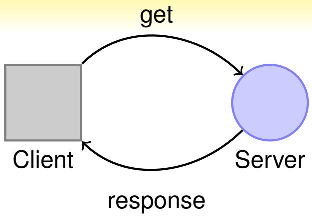
Client
Server

Web Services

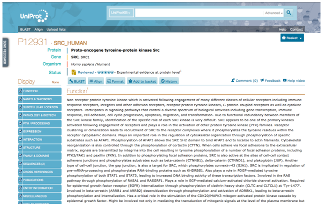


get: `http://www.uniprot.org/uniprot/P12931`

Web Services



get: `http://www.uniprot.org/uniprot/P12931`
response: **HTML**



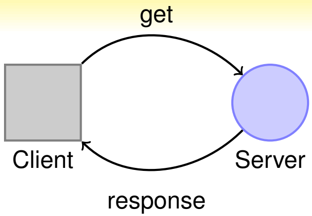
P12931 - SRC_HUMAN

Protein	**Proto-oncogene tyrosine-protein kinase Src**
Gene	**SRC**, SRC1
Organism	*Homo sapiens (Human)*
Status	Reviewed - ●●●●● - Experimental evidence at protein level

Function

Non-receptor protein tyrosine kinase which is activated following engagement of many different classes of cellular receptors including immune response receptors, integrins and other adhesion receptors, receptor protein tyrosine kinases, G protein-coupled receptors as well as cytokine receptors. Participates in signaling pathways that control a diverse spectrum of biological activities including gene transcription, immune response, cell adhesion, cell cycle progression, apoptosis, migration, and transformation. Due to functional redundancy between members of the SRC kinase family, identification of the specific role of each SRC kinase is very difficult. SRC appears to be one of the primary kinases activated following engagement of receptors and plays a role in the activation of other protein tyrosine kinase (PTK) families. Receptor clustering or dimerization leads to recruitment of SRC to the receptor complexes where it phosphorylates the tyrosine residues within the receptor cytoplasmic domains. Plays an important role in the regulation of cytoskeletal organization through phosphorylation of specific substrates such as AFAP1. Phosphorylation of AFAP1 allows the SRC SH2 domain to bind AFAP1 and to localize to actin filaments. Cytoskeletal reorganization is also controlled through the phosphorylation of cortactin (CTTN). When cells adhere via focal adhesions to the extracellular matrix, signals are transmitted by integrins into the cell resulting in tyrosine phosphorylation of a number of focal adhesion proteins, including PTK2/FAK1 and paxillin (PXN). In addition to phosphorylating focal adhesion proteins, SRC is also active at the sites of cell-cell contact adherens junctions and phosphorylates substrates such as beta-catenin (CTNNB1), delta-catenin (CTNND1), and plakoglobin (JUP). Another type of cell-cell junction, the gap junction, is also a target for SRC, which phosphorylates connexin-43 (GJA1). SRC is implicated in regulation of pre-mRNA-processing and phosphorylates RNA-binding proteins such as KHDRBS1. Also plays a role in PDGF-mediated tyrosine phosphorylation of both STAT1 and STAT3, leading to increased DNA binding activity of these transcription factors. Involved in the RAS pathway through phosphorylation of RASA1 and RASGRF1. Plays a role in EGF-mediated calcium-activated chloride channel activation. Required for epidermal growth factor receptor (EGFR) internalization through phosphorylation of clathrin heavy chain (CLTC and CLTCL1) at 'Tyr-1477'. Involved in beta-arrestin (ARRB1 and ARRB2) desensitization through phosphorylation and activation of ADRBK1, leading to beta-arrestin phosphorylation and internalization. Has a critical role in the stimulation of the CDK20/MAPK3 mitogen-activated protein kinase cascade by epidermal growth factor. Might be involved not only in mediating the transduction of mitogenic signals at the level of the plasma membrane but

Web Services



get: `http://www.uniprot.org/uniprot/P12931`**`.txt`**
response: **TEXT/TSV**

```
ID SRC_HUMAN Reviewed; 536 AA.
AC P12931; E1P5V4; Q76P87; Q86VB9; Q9H5A8;
DT 01-OCT-1989, integrated into UniProtKB/Swiss-Prot.
DT 23-JAN-2007, sequence version 3.
DT 03-SEP-2014, entry version 187.
DE RecName:  Full=Proto-oncogene tyrosine-protein kinase Src;
...
```

REST

A RESTful application

is an application that exposes its state and functionality as a set of resources that the clients can manipulate and conforms to a certain set of principles:

- All resources are uniquely addressable, usually through URIs; other addressing can also be used, though.
- All resources can be manipulated through a constrained set of well-known actions, usually CRUD (create, read, update, delete), represented most often through the HTTP's POST, GET, PUT and DELETE; it can be a different set or a subset though - for example, some implementations limit that set to read and modify only (GET and PUT) for example
- The data for all resources is transferred through any of a constrained number of well-known representations, usually HTML, XML or JSON;
- The communication between the client and the application is performed over a *stateless* protocol that allows for multiple layered intermediaries that can reroute and cache the requests and response packets transparently for the client and the application.

REST

A RESTful application

is an application that exposes its state and functionality as a set of resources that the clients can manipulate and conforms to a certain set of principles:

- All resources are uniquely addressable, usually through URIs; other addressing can also be used, though.
- All resources can be manipulated through a constrained set of well-known actions, usually CRUD (create, read, update, delete), represented most often through the HTTP's POST, GET, PUT and DELETE; it can be a different set or a subset though - for example, some implementations limit that set to read and modify only (GET and PUT) for example
- The data for all resources is transferred through any of a constrained number of well-known representations, usually HTML, XML or JSON;
- The communication between the client and the application is performed over a *stateless* protocol that allows for multiple layered intermediaries that can reroute and cache the requests and response packets transparently for the client and the application.

REST

A RESTful application

is an application that exposes its state and functionality as a set of resources that the clients can manipulate and conforms to a certain set of principles:

- All resources are uniquely addressable, usually through URIs; other addressing can also be used, though.
- All resources can be manipulated through a constrained set of well-known actions, usually CRUD (create, read, update, delete), represented most often through the HTTP's POST, GET, PUT and DELETE; it can be a different set or a subset though - for example, some implementations limit that set to read and modify only (GET and PUT) for example
- The data for all resources is transferred through any of a constrained number of well-known representations, usually HTML, XML or JSON;
- The communication between the client and the application is performed over a *stateless* protocol that allows for multiple layered intermediaries that can reroute and cache the requests and response packets transparently for the client and the application.

REST

A RESTful application

is an application that exposes its state and functionality as a set of resources that the clients can manipulate and conforms to a certain set of principles:

- All resources are uniquely addressable, usually through URIs; other addressing can also be used, though.
- All resources can be manipulated through a constrained set of well-known actions, usually CRUD (create, read, update, delete), represented most often through the HTTP's POST, GET, PUT and DELETE; it can be a different set or a subset though - for example, some implementations limit that set to read and modify only (GET and PUT) for example
- The data for all resources is transferred through any of a constrained number of well-known representations, usually HTML, XML or JSON;
- The communication between the client and the application is performed over a *stateless* protocol that allows for multiple layered intermediaries that can reroute and cache the requests and response packets transparently for the client and the application.

REST Methods

Method	defines what you want to do (**GET**=retrieve, **POST**=create/update, **DELETE**=remove). We'll be using just GET requests which can be thought of as read-only access. POST/DELETE are used to modify data on a server.
URL	defines a path to a resource
Parameters	additional arguments, filters etc. usually in the form *parameter* = *value*; the first parameter is separated from the url by '?' while subsequent ones use '&'.

Example: searching for the term 'EMBO':

https://startpage.com/do/search?query=EMBO&with_language=lang_de

REST Methods

Method defines what you want to do (**GET**=retrieve, **POST**=create/update, **DELETE**=remove).
We'll be using just GET requests which can be thought of as read-only access. POST/DELETE are used to modify data on a server.

URL defines a path to a resource

Parameters additional arguments, filters etc. usually in the form *parameter* = *value*; the first parameter is separated from the url by '?' while subsequent ones use '&'.

Example: searching for the term 'EMBO':

https://**startpage.com/do/search**?query=EMBO&with_language=lang_de

REST Methods

Method	defines what you want to do (**GET**=retrieve, **POST**=create/update, **DELETE**=remove). We'll be using just GET requests which can be thought of as read-only access. POST/DELETE are used to modify data on a server.
URL	defines a path to a resource
Parameters	additional arguments, filters etc. usually in the form *parameter* = *value*; the first parameter is separated from the url by '?' while subsequent ones use '&'.

Example: searching for the term 'EMBO':

https://startpage.com/do/search**?query=EMBO&with_language=lang_de**

REST Methods

Method	defines what you want to do (**GET**=retrieve, **POST**=create/update, **DELETE**=remove). We'll be using just GET requests which can be thought of as read-only access. POST/DELETE are used to modify data on a server.
URL	defines a path to a resource
Parameters	additional arguments, filters etc. usually in the form *parameter* = *value*; the first parameter is separated from the url by '?' while subsequent ones use '&'.

Example: searching for the term 'EMBO':

https://startpage.com/do/search?query=EMBO&with_language=lang_de

Note:

For all these examples, any common browser can be used, however for proper 'programmatic' access tools such as 'curl' or 'wget' on the Linux/Mac commandline are much more efficient and can easily be incorporated into little scripts...

Benefits

Easy requests The data can be requested with simple HTTP requests and returned in a variety of programatic and bioinformatical relevant formats such as JSON, XML, YAML and FASTA.

Easy debugging Debugging can be done in any browser. While some might not call this real programming, it surely is the first step towards programmatically querying resources.

Reproducable You can write all your queries into a simple script and repeat the same query later. Even just saving the URL as a bookmark in your browser helps!

Powerful Any data can be made available via a REST service.

Bandwidth An API allows programmatic access to some information if one does not want to download the entire dataset.

Standards By using existing protocols and best-methods (HTTP), all the existing knowledge can be reused (Caching, Redirecting, ...).

Widespread More and more resource providers change from fat/heavy webservices to this lightweight system, for obvious reasons.

Benefits

Easy requests The data can be requested with simple HTTP requests and returned in a variety of programatic and bioinformatical relevant formats such as JSON, XML, YAML and FASTA.

Easy debugging Debugging can be done in any browser. While some might not call this real programming, it surely is the first step towards programmatically querying resources.

Reproducable You can write all your queries into a simple script and repeat the same query later. Even just saving the URL as a bookmark in your browser helps!

Powerful Any data can be made available via a REST service.

Bandwidth An API allows programmatic access to some information if one does not want to download the entire dataset.

Standards By using existing protocols and best-methods (HTTP), all the existing knowledge can be reused (Caching, Redirecting, ...).

Widespread More and more resource providers change from fat/heavy webservices to this lightweight system, for obvious reasons.

Benefits

Easy requests The data can be requested with simple HTTP requests and returned in a variety of programatic and bioinformatical relevant formats such as JSON, XML, YAML and FASTA.

Easy debugging Debugging can be done in any browser. While some might not call this real programming, it surely is the first step towards programmatically querying resources.

Reproducable You can write all your queries into a simple script and repeat the same query later. Even just saving the URL as a bookmark in your browser helps!

Powerful Any data can be made available via a REST service.

Bandwidth An API allows programmatic access to some information if one does not want to download the entire dataset.

Standards By using existing protocols and best-methods (HTTP), all the existing knowledge can be reused (Caching, Redirecting, ...).

Widespread More and more resource providers change from fat/heavy webservices to this lightweight system, for obvious reasons.

Benefits

Easy requests The data can be requested with simple HTTP requests and returned in a variety of programatic and bioinformatical relevant formats such as JSON, XML, YAML and FASTA.

Easy debugging Debugging can be done in any browser. While some might not call this real programming, it surely is the first step towards programmatically querying resources.

Reproducable You can write all your queries into a simple script and repeat the same query later. Even just saving the URL as a bookmark in your browser helps!

Powerful Any data can be made available via a REST service.

Bandwidth An API allows programmatic access to some information if one does not want to download the entire dataset.

Standards By using existing protocols and best-methods (HTTP), all the existing knowledge can be reused (Caching, Redirecting, ...).

Widespread More and more resource providers change from fat/heavy webservices to this lightweight system, for obvious reasons.

BENEFITS

Easy requests The data can be requested with simple HTTP requests and returned in a variety of programatic and bioinformatical relevant formats such as JSON, XML, YAML and FASTA.

Easy debugging Debugging can be done in any browser. While some might not call this real programming, it surely is the first step towards programmatically querying resources.

Reproducable You can write all your queries into a simple script and repeat the same query later. Even just saving the URL as a bookmark in your browser helps!

Powerful Any data can be made available via a REST service.

Bandwidth An API allows programmatic access to some information if one does not want to download the entire dataset.

Standards By using existing protocols and best-methods (HTTP), all the existing knowledge can be reused (Caching, Redirecting, ...).

Widespread More and more resource providers change from fat/heavy webservices to this lightweight system, for obvious reasons.

Benefits

Easy requests The data can be requested with simple HTTP requests and returned in a variety of programatic and bioinformatical relevant formats such as JSON, XML, YAML and FASTA.

Easy debugging Debugging can be done in any browser. While some might not call this real programming, it surely is the first step towards programmatically querying resources.

Reproducable You can write all your queries into a simple script and repeat the same query later. Even just saving the URL as a bookmark in your browser helps!

Powerful Any data can be made available via a REST service.

Bandwidth An API allows programmatic access to some information if one does not want to download the entire dataset.

Standards By using existing protocols and best-methods (HTTP), all the existing knowledge can be reused (Caching, Redirecting, ...).

Widespread More and more resource providers change from fat/heavy webservices to this lightweight system, for obvious reasons.

Benefits

Easy requests	The data can be requested with simple HTTP requests and returned in a variety of programatic and bioinformatical relevant formats such as JSON, XML, YAML and FASTA.
Easy debugging	Debugging can be done in any browser. While some might not call this real programming, it surely is the first step towards programmatically querying resources.
Reproducable	You can write all your queries into a simple script and repeat the same query later. Even just saving the URL as a bookmark in your browser helps!
Powerful	Any data can be made available via a REST service.
Bandwidth	An API allows programmatic access to some information if one does not want to download the entire dataset.
Standards	By using existing protocols and best-methods (HTTP), all the existing knowledge can be reused (Caching, Redirecting, ...).
Widespread	More and more resource providers change from fat/heavy webservices to this lightweight system, for obvious reasons.

BENEFITS

Easy requests The data can be requested with simple HTTP requests and returned in a variety of programatic and bioinformatical relevant formats such as JSON, XML, YAML and FASTA.

Easy debugging Debugging can be done in any browser. While some might not call this real programming, it surely is the first step towards programmatically querying resources.

Reproducable You can write all your queries into a simple script and repeat the same query later. Even just saving the URL as a bookmark in your browser helps!

Powerful Any data can be made available via a REST service.

Bandwidth An API allows programmatic access to some information if one does not want to download the entire dataset.

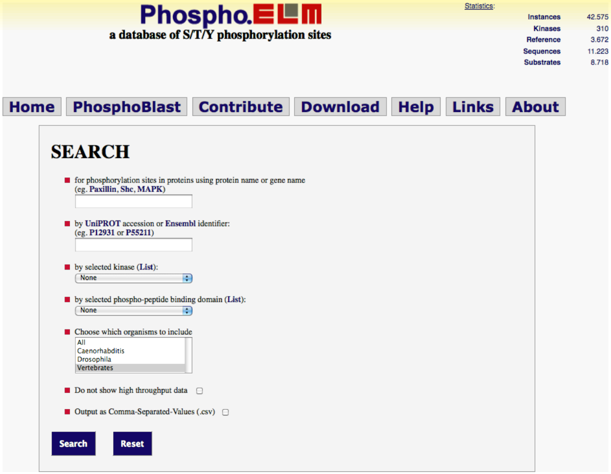
Standards By using existing protocols and best-methods (HTTP), all the existing knowledge can be reused (Caching, Redirecting, ...).

Widespread More and more resource providers change from fat/heavy webservices to this lightweight system, for obvious reasons.

Note:

Not meant to be a substitute for resources such as BioMART etc!

Example: Phospho.ELM


Phospho.ELM
a database of S/T/Y phosphorylation sites

Statistics:

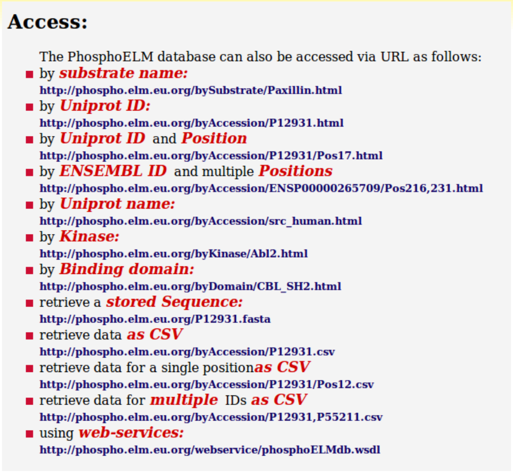
Instances	42.575
Kinases	310
Reference	3.672
Sequences	11.223
Substrates	8.718

SEARCH

- for phosphorylation sites in proteins using protein name or gene name (eg. Paxillin, Shc, MAPK)
- by UniPROT accession or Ensembl identifier: (eg. P12931 or P55211)
- by selected kinase (List): None
- by selected phospho-peptide binding domain (List): None
- Choose which organisms to include: All, Caenorhabditis, Drosophila, Vertebrates
- Do not show high throughput data ☐
- Output as Comma-Separated-Values (.csv) ☐

http://phospho.elm.eu.org/index.html

Example: Phospho.ELM



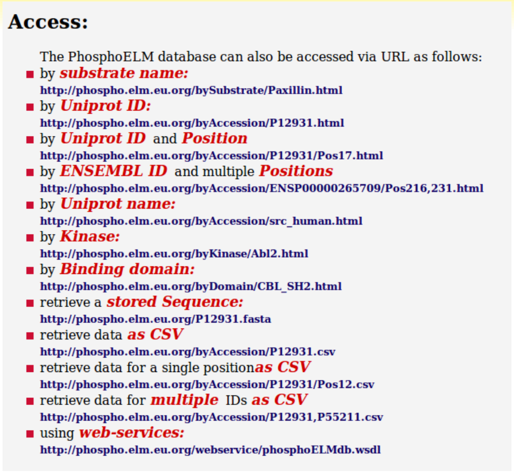
Access:

The PhosphoELM database can also be accessed via URL as follows:

- by ***substrate name:***
  http://phospho.elm.eu.org/bySubstrate/Paxillin.html
- by ***Uniprot ID:***
  http://phospho.elm.eu.org/byAccession/P12931.html
- by ***Uniprot ID*** and ***Position***
  http://phospho.elm.eu.org/byAccession/P12931/Pos17.html
- by ***ENSEMBL ID*** and multiple ***Positions***
  http://phospho.elm.eu.org/byAccession/ENSP00000265709/Pos216,231.html
- by ***Uniprot name:***
  http://phospho.elm.eu.org/byAccession/src_human.html
- by ***Kinase:***
  http://phospho.elm.eu.org/byKinase/Abl2.html
- by ***Binding domain:***
  http://phospho.elm.eu.org/byDomain/CBL_SH2.html
- retrieve a ***stored Sequence:***
  http://phospho.elm.eu.org/P12931.fasta
- retrieve data ***as CSV***
  http://phospho.elm.eu.org/byAccession/P12931.csv
- retrieve data for a single position***as CSV***
  http://phospho.elm.eu.org/byAccession/P12931/Pos12.csv
- retrieve data for ***multiple*** IDs ***as CSV***
  http://phospho.elm.eu.org/byAccession/P12931,P55211.csv
- using ***web-services:***
  http://phospho.elm.eu.org/webservice/phosphoELMdb.wsdl

http://phospho.elm.eu.org/byAccession/P55211.html

Example: Phospho.ELM



Access:

The PhosphoELM database can also be accessed via URL as follows:

- by ***substrate name:***
  http://phospho.elm.eu.org/bySubstrate/Paxillin.html
- by ***Uniprot ID:***
  http://phospho.elm.eu.org/byAccession/P12931.html
- by ***Uniprot ID*** and ***Position***
  http://phospho.elm.eu.org/byAccession/P12931/Pos17.html
- by ***ENSEMBL ID*** and multiple ***Positions***
  http://phospho.elm.eu.org/byAccession/ENSP00000265709/Pos216,231.html
- by ***Uniprot name:***
  http://phospho.elm.eu.org/byAccession/src_human.html
- by ***Kinase:***
  http://phospho.elm.eu.org/byKinase/Abl2.html
- by ***Binding domain:***
  http://phospho.elm.eu.org/byDomain/CBL_SH2.html
- retrieve a ***stored Sequence:***
  http://phospho.elm.eu.org/P12931.fasta
- retrieve data ***as CSV***
  http://phospho.elm.eu.org/byAccession/P12931.csv
- retrieve data for a single position***as CSV***
  http://phospho.elm.eu.org/byAccession/P12931/Pos12.csv
- retrieve data for ***multiple*** IDs ***as CSV***
  http://phospho.elm.eu.org/byAccession/P12931,P55211.csv
- using ***web-services:***
  http://phospho.elm.eu.org/webservice/phosphoELMdb.wsdl

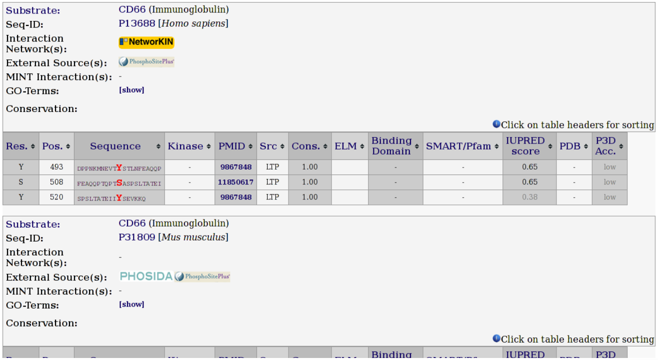
http://phospho.elm.eu.org/byAccession/P55211.csv

Example: Phospho.ELM

Query

http://phospho.elm.eu.org/bySubstrate/cd66.html

Output:



Substrate:	CD66 (Immunoglobulin)
Seq-ID:	P13688 [*Homo sapiens*]
Interaction Network(s):	NetworKIN
External Source(s):	PhosphoSitePlus
MINT Interaction(s):	-
GO-Terms:	[show]
Conservation:	

Click on table headers for sorting

Res.	Pos.	Sequence	Kinase	PMID	Src	Cons.	ELM	Binding Domain	SMART/Pfam	IUPRED score	PDB	P3D Acc.
Y	493	DPPNKMNEVT**Y**STLNFEAQQP	-	9867848	LTP	1.00		-	-	0.65	-	low
S	508	FEAQQPTQPT**S**ASPSLTATEI	-	11850617	LTP	1.00		-	-	0.65	-	low
Y	520	SPSLTATEII**Y**SEVKKQ	-	9867848	LTP	1.00		-	-	0.38	-	low

Substrate:	CD66 (Immunoglobulin)
Seq-ID:	P31809 [*Mus musculus*]
Interaction Network(s):	-
External Source(s):	PHOSIDA PhosphoSitePlus
MINT Interaction(s):	-
GO-Terms:	[show]
Conservation:	

Click on table headers for sorting

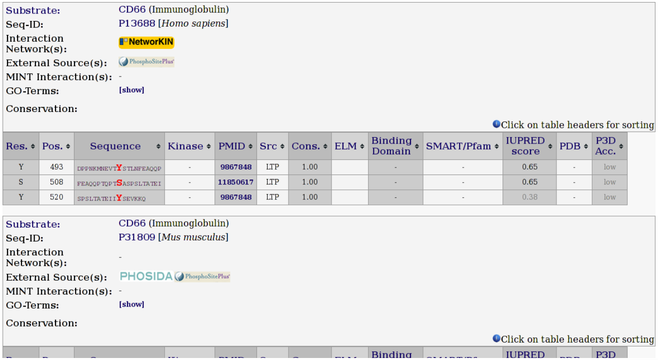
Example: Phospho.ELM

Query

http://phospho.elm.eu.org/**bySubstrate**/cd66.html

- **Query by Substrate name**
- Substrate name
- Output as HTML

Output:



Substrate:	CD66 (Immunoglobulin)
Seq-ID:	P13688 [*Homo sapiens*]
Interaction Network(s):	NetworKIN
External Source(s):	PhosphoSitePlus
MINT Interaction(s):	-
GO-Terms:	[show]
Conservation:	

Click on table headers for sorting

Res.	Pos.	Sequence	Kinase	PMID	Src	Cons.	ELM	Binding Domain	SMART/Pfam	IUPRED score	PDB	P3D Acc.
Y	493	DPPNKMNEVT**Y**STLNFEAQQP	-	9867848	LTP	1.00		-	-	0.65	-	low
S	508	FEAQQPTQPT**S**ASPSLTATEI	-	11850617	LTP	1.00		-	-	0.65	-	low
Y	520	SPSLTATEII**Y**SEVKKQ	-	9867848	LTP	1.00		-	-	0.38	-	low

Substrate:	CD66 (Immunoglobulin)
Seq-ID:	P31809 [*Mus musculus*]
Interaction Network(s):	-
External Source(s):	PHOSIDA PhosphoSitePlus
MINT Interaction(s):	-
GO-Terms:	[show]
Conservation:	

Click on table headers for sorting

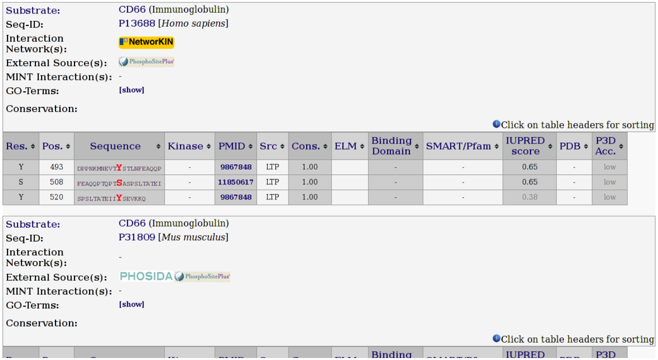
Example: Phospho.ELM

Query

http://phospho.elm.eu.org/bySubstrate/**cd66**.html

- Query by Substrate name
- **Substrate name**
- Output as HTML

Output:



Substrate:	CD66 (Immunoglobulin)
Seq-ID:	P13688 [*Homo sapiens*]
Interaction Network(s):	NetworKIN
External Source(s):	PhosphoSitePlus
MINT Interaction(s):	-
GO-Terms:	[show]
Conservation:	

Click on table headers for sorting

Res.	Pos.	Sequence	Kinase	PMID	Src	Cons.	ELM	Binding Domain	SMART/Pfam	IUPRED score	PDB	P3D Acc.
Y	493	DPPNKMNEVT**Y**STLNFEAQQP	-	**9867848**	LTP	1.00		-	-	0.65	-	low
S	508	FEAQQPTQPT**S**ASPSLTATEI	-	**11850617**	LTP	1.00		-	-	0.65	-	low
Y	520	SPSLTATEII**Y**SEVKKQ	-	**9867848**	LTP	1.00		-	-	0.38	-	low

Substrate:	CD66 (Immunoglobulin)
Seq-ID:	P31809 [*Mus musculus*]
Interaction Network(s):	-
External Source(s):	PHOSIDA PhosphoSitePlus
MINT Interaction(s):	-
GO-Terms:	[show]
Conservation:	

Click on table headers for sorting

Binding IUPRED P3D

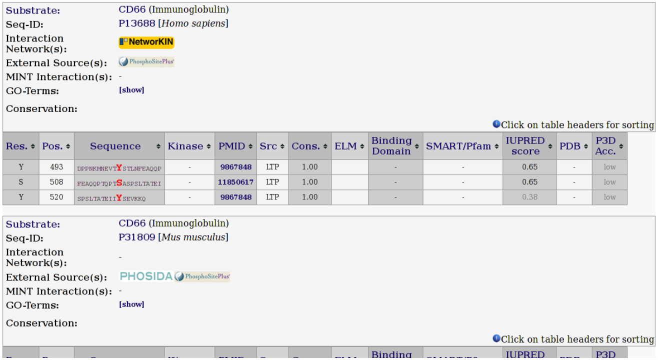
Example: Phospho.ELM

Query

http://phospho.elm.eu.org/bySubstrate/cd66.**html**

- Query by Substrate name
- Substrate name
- **Output as HTML**

Output:



Substrate:	CD66 (Immunoglobulin)
Seq-ID:	P13688 [*Homo sapiens*]
Interaction Network(s):	NetworKIN
External Source(s):	PhosphoSitePlus
MINT Interaction(s):	-
GO-Terms:	[show]
Conservation:	

Click on table headers for sorting

Res.	Pos.	Sequence	Kinase	PMID	Src	Cons.	ELM	Binding Domain	SMART/Pfam	IUPRED score	PDB	P3D Acc.
Y	493	DPPNKMNEVT**Y**STLNFEAQQP	-	9867848	LTP	1.00		-	-	0.65	-	low
S	508	FEAQQPTQPT**S**ASPSLTATEI	-	11850617	LTP	1.00		-	-	0.65	-	low
Y	520	SPSLTATEII**Y**SEVKKQ	-	9867848	LTP	1.00		-	-	0.38	-	low

Substrate:	CD66 (Immunoglobulin)
Seq-ID:	P31809 [*Mus musculus*]
Interaction Network(s):	-
External Source(s):	PHOSIDA PhosphoSitePlus
MINT Interaction(s):	-
GO-Terms:	[show]
Conservation:	

Click on table headers for sorting

EXAMPLE: PHOSPHO.ELM

Query

http://phospho.elm.eu.org/byAccession/P12931/Pos12,17.csv

Output:

```
Acc.; Res.; Pos.; Context; Kinase; PMID; Source; ConScore; ELM; Domain; SMART; IUPRED; PDB; P3D-
P12931; S; 12; SNKSKPKDASQRRRSLEPAE; none; 2136766; 1; 0.21; ; -; ; 0.9168; -; ;
P12931; S; 17; PKDASQRRRSLEPAENVHGA; none; 18088087; 2; 0.24; MOD_PKA_1; -; ; 0.8828; -; ;
P12931; S; 17; PKDASQRRRSLEPAENVHGA; none; 17192257; 2; 0.24; MOD_PKA_1; -; ; 0.8828; -; ;
P12931; S; 17; PKDASQRRRSLEPAENVHGA; none; 17081983; 2; 0.24; MOD_PKA_1; -; ; 0.8828; -; ;
P12931; S; 17; PKDASQRRRSLEPAENVHGA; PKA_group; 11804588; 1; 0.24; MOD_PKA_1; -; ; 0.8828; -; ;
...
```

EXAMPLE: PHOSPHO.ELM

Query

http://phospho.elm.eu.org/**byAccession**/P12931/Pos12,17.csv

- **query by Uniprot Accession**
- Protein Sequence Accession/ID
- Position / multiple Positions
- Output as CSV (character separated values)

Output:

```
Acc.; Res.; Pos.; Context; Kinase; PMID; Source; ConScore; ELM; Domain; SMART; IUPRED; PDB; P3D-
P12931; S; 12; SNKSKPKDASQRRRSLEPAE; none; 2136766; 1; 0.21; ; -; ; 0.9168; -; ;
P12931; S; 17; PKDASQRRRSLEPAENVHGA; none; 18088087; 2; 0.24; MOD_PKA_1; -; ; 0.8828; -; ;
P12931; S; 17; PKDASQRRRSLEPAENVHGA; none; 17192257; 2; 0.24; MOD_PKA_1; -; ; 0.8828; -; ;
P12931; S; 17; PKDASQRRRSLEPAENVHGA; none; 17081983; 2; 0.24; MOD_PKA_1; -; ; 0.8828; -; ;
P12931; S; 17; PKDASQRRRSLEPAENVHGA; PKA_group; 11804588; 1; 0.24; MOD_PKA_1; -; ; 0.8828; -; ;
...
```

Example: Phospho.ELM

Query

http://phospho.elm.eu.org/byAccession/**P12931**/Pos12,17.csv

- query by Uniprot Accession
- **Protein Sequence Accession/ID**
- Position / multiple Positions
- Output as CSV (character separated values)

Output:

```
Acc.; Res.; Pos.; Context; Kinase; PMID; Source; ConScore; ELM; Domain; SMART; IUPRED; PDB; P3D-
P12931; S; 12; SNKSKPKDASQRRRSLEPAE; none; 2136766; 1; 0.21; ; -; ; 0.9168; -; ;
P12931; S; 17; PKDASQRRRSLEPAENVHGA; none; 18088087; 2; 0.24; MOD_PKA_1; -; ; 0.8828; -; ;
P12931; S; 17; PKDASQRRRSLEPAENVHGA; none; 17192257; 2; 0.24; MOD_PKA_1; -; ; 0.8828; -; ;
P12931; S; 17; PKDASQRRRSLEPAENVHGA; none; 17081983; 2; 0.24; MOD_PKA_1; -; ; 0.8828; -; ;
P12931; S; 17; PKDASQRRRSLEPAENVHGA; PKA_group; 11804588; 1; 0.24; MOD_PKA_1; -; ; 0.8828; -; ;
...
```

EXAMPLE: PHOSPHO.ELM

Query

http://phospho.elm.eu.org/byAccession/P12931/**Pos12,17**.csv

- query by Uniprot Accession
- Protein Sequence Accession/ID
- **Position / multiple Positions**
- Output as CSV (character separated values)

Output:

```
Acc.; Res.; Pos.; Context; Kinase; PMID; Source; ConScore; ELM; Domain; SMART; IUPRED; PDB; P3D-
P12931; S; 12; SNKSKPKDASQRRRSLEPAE; none; 2136766; 1; 0.21; ; -; ; 0.9168; -; ;
P12931; S; 17; PKDASQRRRSLEPAENVHGA; none; 18088087; 2; 0.24; MOD_PKA_1; -; ; 0.8828; -; ;
P12931; S; 17; PKDASQRRRSLEPAENVHGA; none; 17192257; 2; 0.24; MOD_PKA_1; -; ; 0.8828; -; ;
P12931; S; 17; PKDASQRRRSLEPAENVHGA; none; 17081983; 2; 0.24; MOD_PKA_1; -; ; 0.8828; -; ;
P12931; S; 17; PKDASQRRRSLEPAENVHGA; PKA_group; 11804588; 1; 0.24; MOD_PKA_1; -; ; 0.8828; -; ;
...
```

EXAMPLE: PHOSPHO.ELM

Query

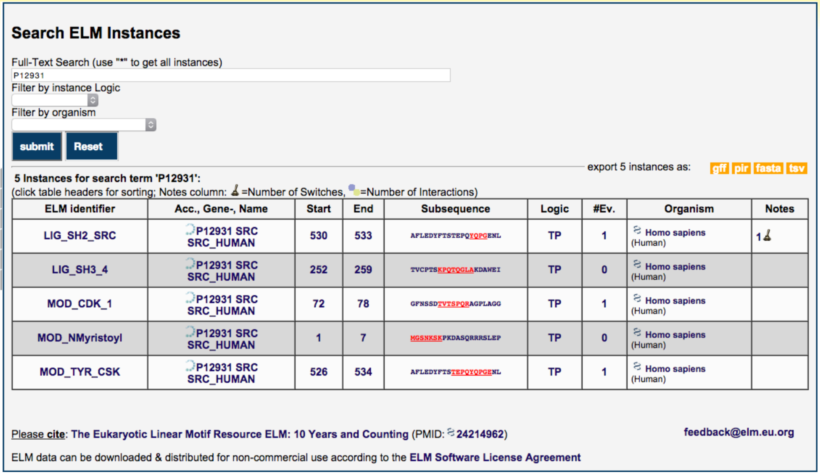
http://phospho.elm.eu.org/byAccession/P12931/Pos12,17.**csv**

- query by Uniprot Accession
- Protein Sequence Accession/ID
- Position / multiple Positions
- **Output as CSV (character separated values)**

Output:

```
Acc.; Res.; Pos.; Context; Kinase; PMID; Source; ConScore; ELM; Domain; SMART; IUPRED; PDB; P3D-
P12931; S; 12; SNKSKPKDASQRRRSLEPAE; none; 2136766; 1; 0.21; ; -; ; 0.9168; -; ;
P12931; S; 17; PKDASQRRRSLEPAENVHGA; none; 18088087; 2; 0.24; MOD_PKA_1; -; ; 0.8828; -; ;
P12931; S; 17; PKDASQRRRSLEPAENVHGA; none; 17192257; 2; 0.24; MOD_PKA_1; -; ; 0.8828; -; ;
P12931; S; 17; PKDASQRRRSLEPAENVHGA; none; 17081983; 2; 0.24; MOD_PKA_1; -; ; 0.8828; -; ;
P12931; S; 17; PKDASQRRRSLEPAENVHGA; PKA_group; 11804588; 1; 0.24; MOD_PKA_1; -; ; 0.8828; -; ;
...
```

EXAMPLE: ELM



Search ELM Instances

Full-Text Search (use "*" to get all instances)

P12931

Filter by instance Logic

Filter by organism

export 5 instances as: gff pir fasta tsv

5 Instances for search term 'P12931':
(click table headers for sorting; Notes column: =Number of Switches, =Number of Interactions)

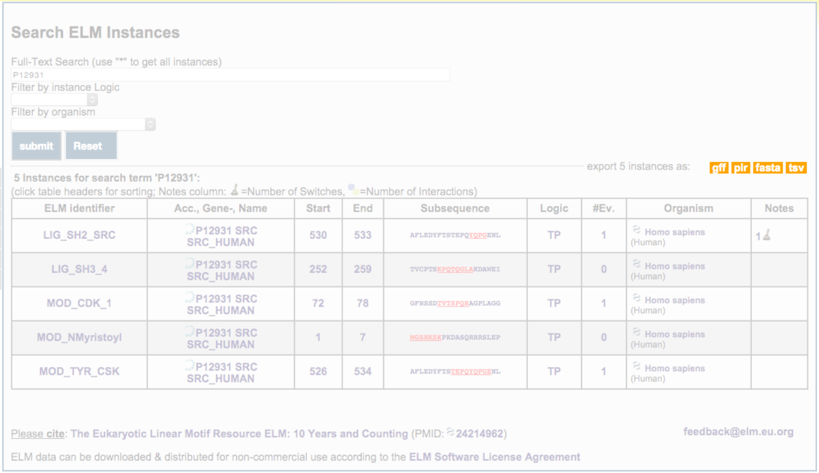
ELM identifier	Acc., Gene-, Name	Start	End	Subsequence	Logic	#Ev.	Organism	Notes
LIG_SH2_SRC	P12931 SRC SRC_HUMAN	530	533	AFLEDYFTSTEPQYQPGENL	TP	1	Homo sapiens (Human)	1
LIG_SH3_4	P12931 SRC SRC_HUMAN	252	259	TVCPTSKPQTQGLAKDAWEI	TP	0	Homo sapiens (Human)	
MOD_CDK_1	P12931 SRC SRC_HUMAN	72	78	GFNSSDTVTSPQRAGPLAGG	TP	1	Homo sapiens (Human)	
MOD_NMyristoyl	P12931 SRC SRC_HUMAN	1	7	MGSNKSKPKDASQRRRSLEP	TP	0	Homo sapiens (Human)	
MOD_TYR_CSK	P12931 SRC SRC_HUMAN	526	534	AFLEDYFTSTEPQYQPGENL	TP	1	Homo sapiens (Human)	

Please **cite**: **The Eukaryotic Linear Motif Resource ELM: 10 Years and Counting** (PMID: **24214962**)

feedback@elm.eu.org

ELM data can be downloaded & distributed for non-commercial use according to the **ELM Software License Agreement**

Example: ELM



Search ELM Instances

Full-Text Search (use "*" to get all instances)

P12931

Filter by instance Logic

Filter by organism

5 Instances for search term 'P12931':

(click table headers for sorting; Notes column: =Number of Switches, =Number of Interactions)

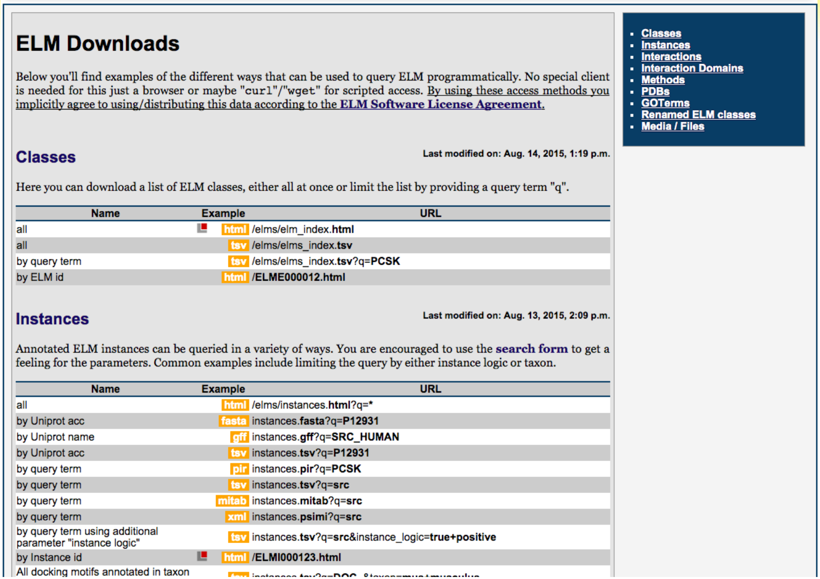
ELM identifier	Acc., Gene-, Name	Start	End	Subsequence	Logic	#Ev.	Organism	Notes
LIG_SH2_SRC	P12931 SRC SRC_HUMAN	530	533	AFLEDYFTSTEPQYQPGENL	TP	1	Homo sapiens (Human)	1
LIG_SH3_4	P12931 SRC SRC_HUMAN	252	259	TVCPTSKPQTQGLAKDAWEI	TP	0	Homo sapiens (Human)	
MOD_CDK_1	P12931 SRC SRC_HUMAN	72	78	GFNSSDTVTSPQRAGPLAGG	TP	1	Homo sapiens (Human)	
MOD_NMyristoyl	P12931 SRC SRC_HUMAN	1	7	MGSNKSKPKDASQRRRSLEP	TP	0	Homo sapiens (Human)	
MOD_TYR_CSK	P12931 SRC SRC_HUMAN	526	534	AFLEDYFTSTEPQYQPGENL	TP	1	Homo sapiens (Human)	

Please cite: The Eukaryotic Linear Motif Resource ELM: 10 Years and Counting (PMID: 24214962)

feedback@elm.eu.org

ELM data can be downloaded & distributed for non-commercial use according to the ELM Software License Agreement

Example: ELM



ELM Downloads

Below you'll find examples of the different ways that can be used to query ELM programmatically. No special client is needed for this just a browser or maybe "curl"/"wget" for scripted access. By using these access methods you implicitly agree to using/distributing this data according to the **ELM Software License Agreement.**

- Classes
- Instances
- Interactions
- Interaction Domains
- Methods
- PDBs
- GOTerms
- Renamed ELM classes
- Media / Files

Classes

Last modified on: Aug. 14, 2015, 1:19 p.m.

Here you can download a list of ELM classes, either all at once or limit the list by providing a query term "q".

Name	Example	URL
all	html	/elms/elm_index.**html**
all	tsv	/elms/elms_index.**tsv**
by query term	tsv	/elms/elms_index.**tsv**?q=**PCSK**
by ELM id	html	**/ELME000012.html**

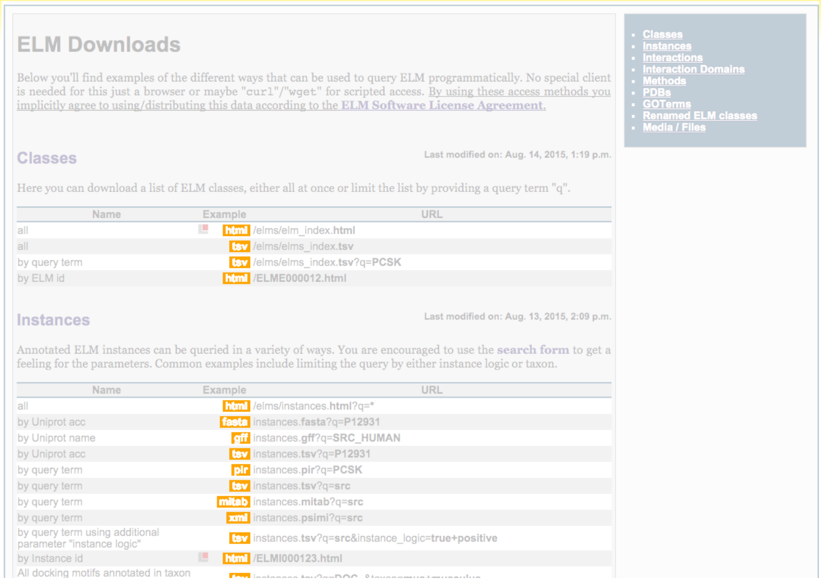
Instances

Last modified on: Aug. 13, 2015, 2:09 p.m.

Annotated ELM instances can be queried in a variety of ways. You are encouraged to use the **search form** to get a feeling for the parameters. Common examples include limiting the query by either instance logic or taxon.

Name	Example	URL
all	html	/elms/instances.**html**?q=*
by Uniprot acc	fasta	instances.**fasta**?q=**P12931**
by Uniprot name	gff	instances.**gff**?q=**SRC_HUMAN**
by Uniprot acc	tsv	instances.**tsv**?q=**P12931**
by query term	pir	instances.**pir**?q=**PCSK**
by query term	tsv	instances.**tsv**?q=**src**
by query term	mitab	instances.**mitab**?q=**src**
by query term	xml	instances.**psimi**?q=**src**
by query term using additional parameter "instance logic"	tsv	instances.**tsv**?q=**src**&instance_logic=**true+positive**
by Instance id	html	**/ELMI000123.html**
All docking motifs annotated in taxon	tsv	instances.**tsv**?q=**DOC_**&taxon=**mus+musculus**

Example: ELM



ELM Downloads

Below you'll find examples of the different ways that can be used to query ELM programmatically. No special client is needed for this just a browser or maybe "curl"/"wget" for scripted access. By using these access methods you implicitly agree to using/distributing this data according to the **ELM Software License Agreement.**

- Classes
- Instances
- Interactions
- Interaction Domains
- Methods
- PDBs
- GOTerms
- Renamed ELM classes
- Media / Files

Classes

Last modified on: Aug. 14, 2015, 1:19 p.m.

Here you can download a list of ELM classes, either all at once or limit the list by providing a query term "q".

Name	Example	URL
all	▨	html /elms/elm_index.**html**
all		tsv /elms/elms_index.**tsv**
by query term		tsv /elms/elms_index.**tsv**?q=**PCSK**
by ELM id		html **/ELME000012.html**

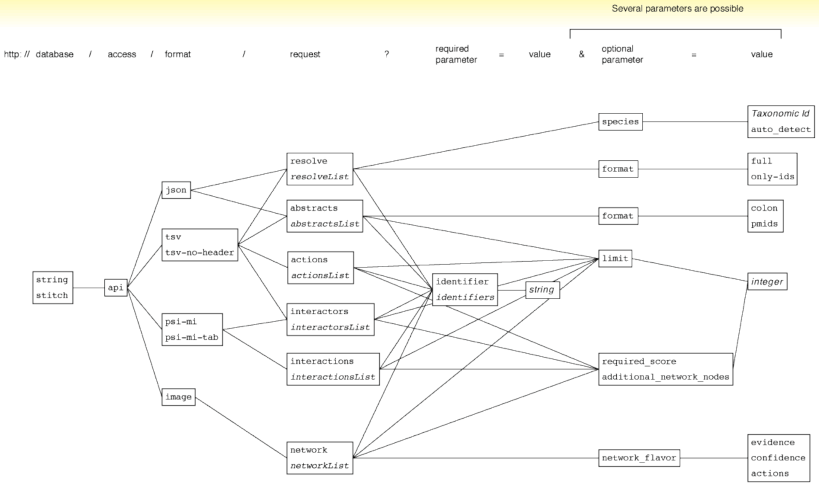
Instances

Last modified on: Aug. 13, 2015, 2:09 p.m.

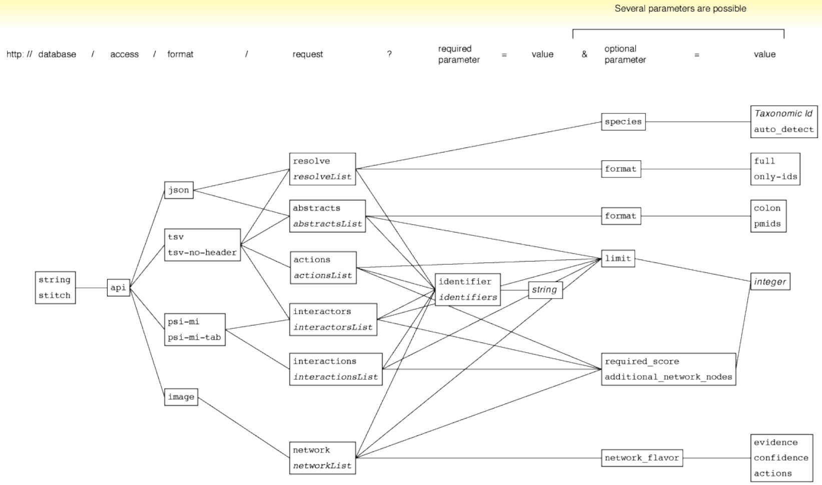
Annotated ELM instances can be queried in a variety of ways. You are encouraged to use the **search form** to get a feeling for the parameters. Common examples include limiting the query by either instance logic or taxon.

Name	Example	URL
all		html /elms/instances.**html**?q=*
by Uniprot acc		fasta instances.**fasta**?q=**P12931**
by Uniprot name		gff instances.**gff**?q=**SRC_HUMAN**
by Uniprot acc		tsv instances.**tsv**?q=**P12931**
by query term		pir instances.**pir**?q=**PCSK**
by query term		tsv instances.**tsv**?q=**src**
by query term		mitab instances.**mitab**?q=**src**
by query term		xml instances.**psimi**?q=**src**
by query term using additional parameter "instance logic"		tsv instances.**tsv**?q=**src**&instance_logic=**true+positive**
by Instance id	▨	html **/ELMI000123.html**
All docking motifs annotated in taxon		tsv instances.**tsv**?q=**DOC_**&taxon=**mus+musculus**

EXAMPLE: STRING / STITCH

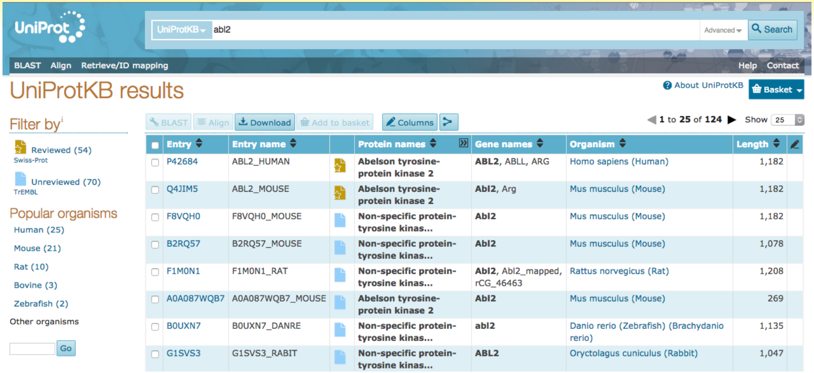


EXAMPLE: STRING / STITCH



```
http://string-db.org/api/psi-mi-tab/interactions?identifier=YOL086C&additional_network_nodes=2
```

Example: Uniprot



UniProtKB results

Filter by

Reviewed (54)
Swiss-Prot

Unreviewed (70)
TrEMBL

Popular organisms

- Human (25)
- Mouse (21)
- Rat (10)
- Bovine (3)
- Zebrafish (2)

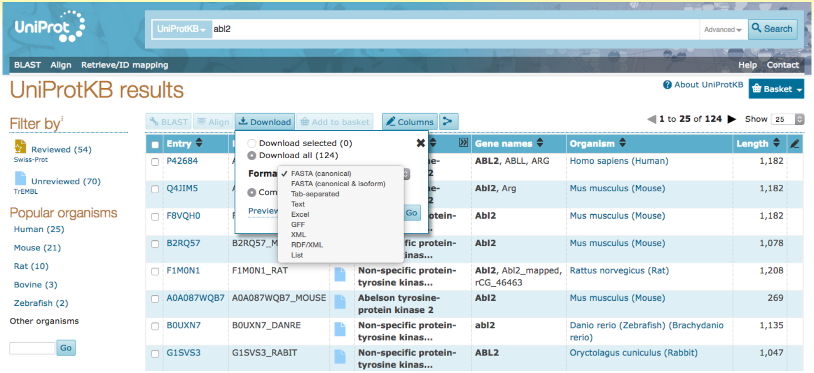
Other organisms

Go

1 to 25 of 124 Show 25

Entry	Entry name		Protein names	Gene names	Organism	Length
P42684	ABL2_HUMAN	★	Abelson tyrosine-protein kinase 2	ABL2, ABLL, ARG	Homo sapiens (Human)	1,182
Q4JIM5	ABL2_MOUSE	★	Abelson tyrosine-protein kinase 2	Abl2, Arg	Mus musculus (Mouse)	1,182
F8VQH0	F8VQH0_MOUSE	□	Non-specific protein-tyrosine kinas...	Abl2	Mus musculus (Mouse)	1,182
B2RQ57	B2RQ57_MOUSE	□	Non-specific protein-tyrosine kinas...	Abl2	Mus musculus (Mouse)	1,078
F1M0N1	F1M0N1_RAT	□	Non-specific protein-tyrosine kinas...	Abl2, Abl2_mapped, rCG_46463	Rattus norvegicus (Rat)	1,208
A0A087WQB7	A0A087WQB7_MOUSE	□	Abelson tyrosine-protein kinase 2	Abl2	Mus musculus (Mouse)	269
B0UXN7	B0UXN7_DANRE	□	Non-specific protein-tyrosine kinas...	abl2	Danio rerio (Zebrafish) (Brachydanio rerio)	1,135
G1SVS3	G1SVS3_RABIT	□	Non-specific protein-tyrosine kinas...	ABL2	Oryctolagus cuniculus (Rabbit)	1,047

EXAMPLE: UNIPROT



abl2

UniProtKB results

1 to 25 of 124

Filter by

Reviewed (54)
Swiss-Prot

Unreviewed (70)
TrEMBL

Popular organisms

Human (25)
Mouse (21)
Rat (10)
Bovine (3)
Zebrafish (2)

Other organisms

Download selected (0)
[✓ Download all (124)]

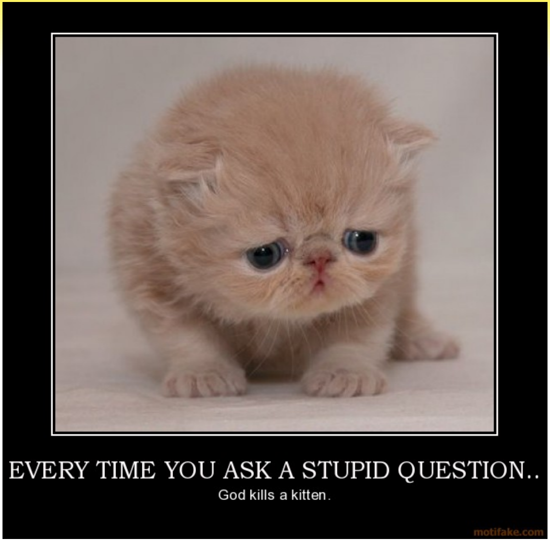
Format:
✓ FASTA (canonical)
FASTA (canonical & isoform)
Tab-separated
Text
Excel
GFF
XML
RDF/XML
List

Com

Preview

Entry				Gene names	Organism	Length
P42684	A		e- 2	ABL2, ABLL, ARG	Homo sapiens (Human)	1,182
Q4JIM5	A		e- 2	Abl2, Arg	Mus musculus (Mouse)	1,182
F8VQH0	F		otein-	Abl2	Mus musculus (Mouse)	1,182
B2RQ57	B2RQ57_M		fic protein- inas...	Abl2	Mus musculus (Mouse)	1,078
F1M0N1	F1M0N1_RAT		Non-specific protein-tyrosine kinas...	Abl2, Abl2_mapped, rCG_46463	Rattus norvegicus (Rat)	1,208
A0A087WQB7	A0A087WQB7_MOUSE		Abelson tyrosine-protein kinase 2	Abl2	Mus musculus (Mouse)	269
B0UXN7	B0UXN7_DANRE		Non-specific protein-tyrosine kinas...	abl2	Danio rerio (Zebrafish) (Brachydanio rerio)	1,135
G1SVS3	G1SVS3_RABIT		Non-specific protein-tyrosine kinas...	ABL2	Oryctolagus cuniculus (Rabbit)	1,047

Questions?


EVERY TIME YOU ASK A STUPID QUESTION..
God kills a kitten.
motifake.com