Using the PAnnBuilder Package

Hong Li[‡]*

April 22, 2010

[‡]Key Lab of Systems Biology Shanghai Institutes for Biological Sciences Chinese Academy of Sciences, P. R. China

Contents

1	Overview									2
2	Getting Started									2
	2.1 Requirements			 						2
	2.2 Installation .			 						3
	2.3 Public Data So	ources		 						3
		ckages Produced								4
		es produced by P	•							4
	•	nnotation data p								5
3	Function Descrip	ion								13
	3.1 Getting URL a	nd Version		 						13
	0	riting Data								14
	ũ là chiến the second se	ing perl program								14
		data using R .	-							15
	-	Documents								16
	° -	Packages								16
4	Building Annotat	ion Data Pack	ages							17
5	Session Informati	on								19

*sysptm@gmail.com

1 Overview

In genomic era, genome-scale experiments and data analyses require genes to be annotated from different sources, an R R Development Core Team (2008) package AnnBuilder was developped for this purpose Zhang et al. (2003). In post genomic era, advances in proteomics highlight the urgence of understanding protein language Jensen (2006). However, relative to genes, AnnBuilder is limited in protein annotation due to the complexity of proteins caused by 3-D strucutre, alternative splicing, modification, dynamic location and other features. The package PAnnBuilder focuses on assembling proteomic annotation data, which should be very useful for proteomic data interpretation. It not only inherits the good features of AnnBuilder such as automatically parsing protein annotation data and building R packages from selected sources, but also emphasizes specific features of proteins. PAnnBuilder currently supports 16 databases involving diverse aspects of proteomics, such as protein primary data, protein domain/family, subcellular location, protein interaction, post-translational modifications, body fluid proteomics, homolog/ortholog groups and so on. Additionally, PAnnBuilder allows annotation to unknown proteins based on sequence similarities to other well-annotated proteins. To extend the use of PAnnBuilder, 54 standard R annotation packages are produced from main protein databases, which are freely available for download via biocLite.

2 Getting Started

2.1 Requirements

PAnnBuilder requires the support from the following items:

- 1. For PAnnBuilder $\geq 1.3.0$, R $\geq 2.7.0$ is needed for building SQLite-based package. Dependent R packages are needed to be installed: *methods*, *utils*, *RSQLite*, *Biobase* ($\geq 1.17.0$), *AnnotationDbi* ($\geq 1.3.12$). If you use the installation script "PAnnBuilder.R" to install PAnnBuilder ($\geq 1.3.0$), it will automatically check these dependent packages and install missing packages from CRAN or Bioconductor.
- 2. Rtools is needed for Windows user. The matched version of Rtools with R can be downloaded via http://www.murdoch-sutherland.com/Rtools/.
- 3. Perl is required to parse the rather large annotation source data files. It can be down-loaded from http://www.perl.com/download.csp.
- Program formatdb and BLASTP is needed for function crossBuilder, crossBuilder_DB, pSeqBuilder, pSeqBuilder_DB.
 BLAST can be downloaded from http://www.ncbi.nlm.nih.gov/BLAST/download.shtml.

Note: It is better to have enough space for the temporary directory. The path of the persession temporary directory can be acquired by:

- > tempdir()
- [1] "/tmp/RtmpY8WZBO"

2.2 Installation

The biocLite script is used to install PAnnBuilder from within R:

```
> source("http://bioconductor.org/biocLite.R")
> biocLite("PAnnBuilder")
```

> library(PAnnBuilder)

Note: Web Connection is needed to install PAnnBuilder and its depended packages.

2.3 Public Data Sources

PAnnBuilder parses proteomics annotation data from public sources and build R annotation packages. It also provides convenient functions to access these sources. For example, you can get all supported databases for "Homo sapiens" by:

- > library(PAnnBuilder)
- > getALLUrl("Homo sapiens")
- > getALLBuilt("Homo sapiens")

Detail description of all supported public data sources in PAnnBuilder are as follows:

- UniProt The data ftp://ftp.ebi.ac.uk/pub/databases/uniprot/knowledgebase will be used to map protein UniProt identifiers to diverse annotation available in UniProt database.
- **IPI** The data ftp://ftp.ebi.ac.uk/pub/databases/IPI/current will be used to map protein IPI identifiers to diverse annotation available in International Protein Index database.
- **RefSeq** The data ftp://ftp.ncbi.nih.gov/refseq will be used to map protein RefSeq identifiers to diverse annotation available in NCBI RefSeq database.
- Entrez Gene The data ftp://ftp.ncbi.nih.gov/gene/DATA will be used to annotate genes after the Entrez Gene identifiers have been obtained.
- Gene Ontology The data ftp://ftp.ebi.ac.uk/pub/databases/GO/goa/proteomes and http://archive.geneontology.org/latest-termdb will be used to obtain gene on-tology information.
- **KEGG** Some data at ftp://ftp.genome.jp/pub/kegg will be used to obtain pathway information.

- HomoloGene A data file provided by ftp://ftp.ncbi.nlm.nih.gov/pub/HomoloGene/ current/ will be used to extract mappings between GeneID/ProteinGI and Homolo-Gene ids.
- InParanoid The data http://inparanoid.sbc.su.se will be used to obtain ortholog protein groups between two organisms.
- Gene Interaction The data ftp://ftp.ncbi.nih.gov/gene/GeneRIF will be used to extract protein-protein interactions between Entrez GeneID or Protein RefSeq ids.
- IntAct The data ftp://ftp.ebi.ac.uk/pub/databases/intact/current/psimitab will be used to extract protein-protein interactions between UniProt protein accession numbers.
- **MPPI** The data http://mips.gsf.de/proj/ppi/data/mppi.gz will be used to extract protein-protein interactions between UniProt protein accession numbers.
- **3DID** The data http://gatealoy.pcb.ub.es/3did/download will be used to extract domaindomain interactions between Pfam domain identifiers.
- **DOMINE** The data http://domine.utdallas.edu will be used to extract domain-domain interactions between Pfam domain identifiers.
- DBSubLoc The data http://www.bioinfo.tsinghua.edu.cn/~guotao will be used to obtain subcellular localization for protein from SWISS-PROT and PIR database.
- BaCelLo The data http://gpcr2.biocomp.unibo.it/bacello will be used to map SwissProt eukaryotes protein identifiers to subcellular localization.
- **SCOP** The data http://scop.mrc-lmb.cam.ac.uk/scop will be used to map PDB structure identifiers to SCOP domain identifiers.
- **PeptideAtlas** The data http://www.peptideatlas.org/builds will be used to obtain experimentally identified peptides and their coordinates on chromosomes.
- **SysPTM** The data http://www.biosino.org/papers/SysPTM will be used to obtain protein post-translational modifications information.
- Sys-BodyFluid The data http://www.biosino.org/papers/Sys-BodyFluid will be used to map IPI protein identifiers to body fluids.

2.4 Annotation Packages Produced by PAnnBuilder

2.4.1 Packages produced by PAnnBuilder

PAnnBuilder has powerful ability to build R package for assembling proteome annotation. However, the process of building new package may be time-consuming because of the downloading and parsing of large data files. To make PAnnBuilder useful for any users, we have built many frequently used annotation packages in advance. These pre-built package can be downloaded via biocLite.

These packages are divided into two classes: environment-based packages built by "*Builder" functions (see Table ??); SQLite based packages built by "*Builder_DB" functions (see Table 1). They are widely used methods for building Bioconductor meta-data packages. Each SQLite-based annotation package (identified by a ".db" suffix in the package name) contains a number of AnnDbBimap objects in place of the environment objects found in the old-style environment-based annotation packages. In future, SQLite-based annotation package will replace environment-based packages.

The pre-built packages provide a quick start for R beginners. If one wants to analyze protein set in Human IPI database, the quickest way is to download and use SQLite based package "org.Hs.ipi.db". However, if the package one wants has not been built or a new-version database is released, new package should be built using functions in PAnnBuilder (See Section 4 for methods of building annotation packages).

2.4.2 Using annotation data package

SQLite-based *.db packages are capable of flexible data queries, reverse mapping, and data filtering. Vignette of "AnnotationDbi" detailedly described how to use SQLite based packages (http://www.bioconductor.org/packages/release/bioc/vignettes/AnnotationDbi/inst/doc/AnnotationDbi.pdf). Here package "org.Hs.ipi.db" produced by PAnnBuilder is illustrated as an example in the following code chuck.

- 1. Install and load annotation package. Package "org.Hs.ipi" is used as an example, other packages can be derived accordingly.
 - > biocLite("org.Hs.ipi.db")
 - > library(org.Hs.ipi.db)
- 2. Browse data in the package.

> `?`(org.Hs.ipiGENEID)

3. Convert the environment object into a "list" object, and get values by index or name.

```
> xx <- as.list(org.Hs.ipiGENEID)
> xx[!is.na(xx)][1:3]
$IPI00000001.2
[1] "6780"
$IPI00000005.1
[1] "4893"
```

Description	R Function	Source	Organism	Package
complete and canonical			Homo sapiens	org.Hs.ipi.db
annotaion for all proteins		IPI	Mus musculus	org.Mm.ipi.db
of a specific organism,			Rattus norvegicus	org.Rn.ipi.db
including protein			Homo sapiens	org.Hs.sp.db
description, Entrez gene	pBaseBuilder_DB	Swiss-Prot	Mus musculus	org.Mm.sp.db
identifier, KEGG pathway			Rattus norvegicus	org.Rn.sp.db
gene ontology, domain,	,		Homo sapiens	org.Hs.ref.db
and so on.		RefSeq	Mus musculus	org.Mm.ref.db
and so on.			Rattus norvegicus	org.Rn.ref.db
protein indentifier		Suring Duct ID	Homo sapiens	org.Hs.cross.db
-	crossBuilder_DB	RefSeq	I,Homo sapiens 'Mus musculus	org.Mm.cross.db
mapping		neiseq	Rattus norvegicus	org.Rn.cross.db
		IntAct		int.intact.db
protein-protein or	intBuilder_DB	NCBI Gene		int.geneint.db
domain-domain		MPPI		int.mppi.db
interaction data		3did		int.did.db
		Domine		int.domine.db
protein subcell location	subcellBuilder_DB	BaCelLo		sc.bacello.db
protein subcen location	subcendunder_DD	DBSubLoc		sc.dbsubloc.db
protein structure classifi-	scopBuilder_DB	SCOP		scop.db
cation		G DEL		
protein post-	$ptmBuilder_DB$	SysPTM		sysptm.db
translational modifi-				
cation		a	TT 1	
body fluid protein	bfBuilder_DB	Sys-	Homo sapiens	org.Hs.bf.db
		BodyFluid		
homolog protein group	HomoloGeneBuilder_			homolog.db
ortholog protein group	InParanoidBuilder_D	BInParanoid	Homo sapiens,	org.HsMm.ortholog.db
			Mus musculus	
peptides identified by	PeptideAtlasBuilder_	DBptideAtlas	Homo sapiens	org.Hs.pep.db
mass spectrometry				
gene ontology	GOABuilder_DB	GOA	Homo sapiens	org.Hs.goa.db
identifier and name	dNameBuilder_DB			dName.db

 Table 1: SQLite-based Annotation Packges Produced by PAnnBuilder.

2 IPI0000005.1 hsa04012 3 IPI0000005.1 hsa04062 4 IPI0000005.1 hsa04360 5 IPI0000005.1 hsa04370 6 IPI0000005.1 hsa04530 7 IPI0000005.1 hsa04540 8 IPI0000005.1 hsa04650 IPI0000005.1 hsa04660 9 10 IPI0000005.1 hsa04662 11 IPI0000005.1 hsa04664 12 IPI0000005.1 hsa04720 13 IPI0000005.1 hsa04722 14 IPI0000005.1 hsa04730 15 IPI0000005.1 hsa04810 16 IPI0000005.1 hsa04910 17 IPI0000005.1 hsa04912 18 IPI0000005.1 hsa04916 19 IPI0000005.1 hsa05200 20 IPI0000005.1 hsa05211 21 IPI0000005.1 hsa05213 22 IPI0000005.1 hsa05214 23 IPI0000005.1 hsa05215 24 IPI0000005.1 hsa05216 25 IPI0000005.1 hsa05218 26 IPI0000005.1 hsa05219 27 IPI0000005.1 hsa05220 28 IPI0000005.1 hsa05221 29 IPI0000005.1 hsa05223 30 IPI0000006.1 hsa04010 31 IPI0000006.1 hsa04012 32 IPI0000006.1 hsa04062

```
34 IPI0000006.1 hsa04360
35 IPI0000006.1 hsa04370
36 IPI0000006.1 hsa04510
37 IPI0000006.1 hsa04530
38 IPI0000006.1 hsa04540
39 IPI0000006.1 hsa04650
40 IPI0000006.1 hsa04660
41 IPI0000006.1 hsa04662
42 IPI0000006.1 hsa04664
43 IPI0000006.1 hsa04720
44 IPI0000006.1 hsa04722
45 IPI0000006.1 hsa04730
46 IPI0000006.1 hsa04810
47 IPI0000006.1 hsa04910
48 IPI0000006.1 hsa04912
49 IPI0000006.1 hsa04916
50 IPI0000006.1 hsa05200
51 IPI0000006.1 hsa05211
52 IPI0000006.1 hsa05213
53 IPI0000006.1 hsa05214
54 IPI0000006.1 hsa05215
55 IPI0000006.1 hsa05216
56 IPI0000006.1 hsa05218
57 IPI0000006.1 hsa05219
58 IPI0000006.1 hsa05220
59 IPI0000006.1 hsa05221
60 IPI0000006.1 hsa05223
61 IPI0000013.1 hsa04142
> tmp1 <- revmap(org.Hs.ipiPATH)</pre>
> class(tmp1)
[1] "AnnDbBimap"
attr(,"package")
[1] "AnnotationDbi"
> as.list(tmp1)[1]
$hsa00010
```

33 IPI0000006.1 hsa04144

```
[1] "IPI00003925.6" "IPI00005118.2" "IPI00006663.1" "IPI00009744.3"
[5] "IPI00009790.1" "IPI00015911.2" "IPI00016768.3" "IPI00018031.1"
[9] "IPI00018246.5" "IPI00019500.3" "IPI00021338.2" "IPI00024087.3"
```

[13]	"IPI00026663.2"	"IPI00027165.3"	"IPI00027497.5"	"IPI00028155.2"
[17]	"IPI00020003.2"		"IPI00027497.3	"IPI00103467.4"
[21]	"IPI00148061.3"	"IPI00166751.3"	"IPI00169383.3"	"IPI00184775.3"
[25]	"IPI00215979.3"	"IPI00216171.3"	"IPI00216792.2"	"IPI00216932.4"
[29]	"IPI00217872.3"	"IPI00217966.9"	"IPI00218407.6"	"IPI00218474.6"
[33]	"IPI00218570.6"	"IPI00218767.1"	"IPI00218768.1"	"IPI00218896.3"
[37]	"IPI00218899.6"	"IPI00219018.7"	"IPI00219217.3"	"IPI00219526.6"
[41]	"IPI00219568.4"	"IPI00219585.4"	"IPI00220271.3"	"IPI00220644.8"
[45]	"IPI00220663.3"	"IPI00220665.6"	"IPI00220667.3"	"IPI00221234.7"
[49]	"IPI00244083.4"	"IPI00248961.1"	"IPI00292698.4"	"IPI00292709.2"
[53]	"IPI00296183.7"	"IPI00299456.4"	"IPI00306044.3"	"IPI00306301.2"
[57]	"IPI00332371.9"	"IPI00333619.4"	"IPI00374975.2"	"IPI00382746.1"
[61]	"IPI00384116.3"	"IPI00384883.3"	"IPI00384967.3"	"IPI00385347.1"
[65]	"IPI00386733.2"	"IPI00394758.1"	"IPI00413730.4"	"IPI00418262.5"
[69]	"IPI00451401.3"	"IPI00465025.1"	"IPI00465028.7"	"IPI00465179.4"
[73]	"IPI00465248.5"	"IPI00465343.3"	"IPI00465439.5"	"IPI00473031.6"
[77]	"IPI00479186.7"	"IPI00479877.4"	"IPI00513830.1"	"IPI00549564.6"
[81]	"IPI00549725.6"	"IPI00549885.4"	"IPI00550364.8"	"IPI00550486.1"
[85]	"IPI00552290.3"	"IPI00552617.3"	"IPI00553155.1"	"IPI00554498.3"
[89]	"IPI00555601.1"	"IPI00555722.1"	"IPI00555728.1"	"IPI00556013.1"
[93]	"IPI00556284.2"	"IPI00604528.3"	"IPI00607708.3"	"IPI00639981.1"
[97]	"IPI00640568.1"	"IPI00640862.1"	"IPI00642546.1"	"IPI00642664.1"
[101]	"IPI00642732.2"	"IPI00642820.1"	"IPI00643196.1"	"IPI00643575.1"
[105]	"IPI00644994.1"	"IPI00645848.1"	"IPI00646468.1"	"IPI00647702.1"
[109]	"IPI00654709.1"	"IPI00743142.2"	"IPI00743713.3"	"IPI00746777.3"
[113]	"IPI00759806.1"	"IPI00784216.3"	"IPI00788640.1"	"IPI00788737.1"
[117]	"IPI00788836.1"	"IPI00788938.1"	"IPI00789081.1"	"IPI00789134.4"
[121]	"IPI00789171.1"	"IPI00789173.1"	"IPI00789301.1"	"IPI00790892.1"
[125]	"IPI00791170.1"		"IPI00791564.2"	"IPI00791666.1"
[129]	"IPI00792207.2"	"IPI00792375.1"	"IPI00792448.1"	"IPI00792655.1"
[133]		"IPI00793665.1"		
	"IPI00794605.1"			
[141]		"IPI00795622.2"		
[145]		"IPI00796333.1"		
[149]		"IPI00796852.1"		
[153]		"IPI00797580.1"		
[157]		"IPI00815950.1"		
[161]	"IPI00847989.3"		"IPI00872487.1"	
[165]		"IPI00902542.1"		
[169]	"IPI00908791.2"		"IPI00908927.1"	
[173]	"IPI00909158.1"			
[]//]	"IPI00909595.1"	"17100909694.1"	"17100909829.1"	"17100909949.1"

[1]	"IPI00003925.6"	"IPI00005118.2"	"IPI00006663.1"	"IPI00009744.3"
[5]	"IPI00009790.1"	"IPI00015911.2"	"IPI00016768.3"	"IPI00018031.1"
[9]	"IPI00018246.5"	"IPI00019500.3"	"IPI00021338.2"	"IPI00024087.3"
[13]	"IPI00026663.2"	"IPI00027165.3"	"IPI00027497.5"	"IPI00028155.2"
[17]	"IPI00060200.3"	"IPI00073772.5"	"IPI00102864.3"	"IPI00103467.4"
[21]	"IPI00148061.3"	"IPI00166751.3"	"IPI00169383.3"	"IPI00184775.3"
[25]	"IPI00215979.3"	"IPI00216171.3"	"IPI00216792.2"	"IPI00216932.4"
[29]	"IPI00217872.3"	"IPI00217966.9"	"IPI00218407.6"	"IPI00218474.6"
[33]	"IPI00218570.6"	"IPI00218767.1"	"IPI00218768.1"	"IPI00218896.3"
[37]	"IPI00218899.6"	"IPI00219018.7"	"IPI00219217.3"	"IPI00219526.6"
[41]	"IPI00219568.4"	"IPI00219585.4"	"IPI00220271.3"	"IPI00220644.8"
[45]	"IPI00220663.3"	"IPI00220665.6"	"IPI00220667.3"	"IPI00221234.7"
[49]	"IPI00244083.4"	"IPI00248961.1"	"IPI00292698.4"	"IPI00292709.2"
[53]	"IPI00296183.7"	"IPI00299456.4"	"IPI00306044.3"	"IPI00306301.2"
[57]	"IPI00332371.9"	"IPI00333619.4"	"IPI00374975.2"	"IPI00382746.1"
[61]	"IPI00384116.3"	"IPI00384883.3"	"IPI00384967.3"	"IPI00385347.1"
[65]	"IPI00386733.2"	"IPI00394758.1"	"IPI00413730.4"	"IPI00418262.5"
[69]	"IPI00451401.3"	"IPI00465025.1"	"IPI00465028.7"	"IPI00465179.4"
[73]	"IPI00465248.5"	"IPI00465343.3"	"IPI00465439.5"	"IPI00473031.6"

> tmp2[1]

\$hsa00010

[1] "list"

> class(tmp2)

> tmp2 <- reverseSplit(as.list(org.Hs.ipiPATH))</pre>

```
[181] "IPI00910420.1" "IPI00910642.1" "IPI00910754.1" "IPI00910781.1"
[185] "IPI00910974.1" "IPI00910979.1" "IPI00915933.1" "IPI00916206.1"
[189] "IPI00916818.1" "IPI00916990.1" "IPI00916994.1" "IPI00917139.1"
[193] "IPI00917193.1" "IPI00917237.1" "IPI00917473.1" "IPI00917841.1"
[197] "IPI00922697.1" "IPI00925520.1" "IPI00926110.1" "IPI00926319.1"
[201] "IPI00926810.1" "IPI00927039.1" "IPI00927177.1" "IPI00926319.1"
[205] "IPI00927598.1" "IPI00927949.1" "IPI00930416.1" "IPI00936002.1"
[209] "IPI00939286.1" "IPI00939339.1" "IPI00939637.1" "IPI00940003.1"
[213] "IPI00940201.1" "IPI00941093.1" "IPI00941338.1" "IPI00941899.1"
[217] "IPI00942944.1" "IPI00942961.1" "IPI00945309.1" "IPI00945466.1"
[221] "IPI00945625.1" "IPI00945694.1" "IPI00945766.2" "IPI00945873.1"
[225] "IPI00946018.1" "IPI00946160.1" "IPI00946173.1" "IPI00946252.1"
[223] "IPI00946018.1" "IPI00947319.1" "IPI00946812.1" "IPI00947277.1"
[233] "IPI00947129.1" "IPI00947319.1" "IPI00955815.1" "IPI00955977.1"
```

[81] "IPI00549725.6" "IPI0055281.3" "IPI00553155.1" "IPI00554498.3" [85] "IPI0055601.1" "IPI0055572.1" "IPI00553155.1" "IPI00554498.3" [93] "IPI0055684.2" "IPI0064258.3" "IPI006425661.1" "IPI00642566.1" "IPI00642566.1" [97] "IPI00642588.1" "IPI00642586.1" "IPI00642566.1" "IPI0064257.1" [101] "IPI00642732.2" "IPI00642820.1" "IPI00642566.1" "IPI0064373.3" "IPI0064777.3" [103] "IPI00654709.1" "IPI00743142.2" "IPI00788640.1" "IPI007877.3" [113] "IPI00758966.1" "IPI00789081.1" "IPI00798737.1" [117] "IPI0078836.1" "IPI007989301.1" "IPI00790892.1" [122] "IPI0079207.2" "IPI0079275.1" "IPI0079265.1" [133] "IPI00794605.1" "IPI0079663.1" "IPI0079663.1" [141] "IPI0079623.1" "IPI00798331.1" "IPI0079673.1" [143] "IPI0079623.1" "IPI0079663.1" "IPI0079673.1" [144] "IPI0079623.1" "IPI0079663.1" "IPI0079673.1" [145] "IPI0079623.1" "IPI0079663.1	[77]	"IPI00479186.7"	"IPI00479877.4"	"IPI00513830.1"	"IPI00549564.6"
[85] "IPI00552290.3" "IPI00555617.3" "IPI00555155.1" "IPI005564498.3" [89] "IPI005556284.2" "IPI00642528.3" "IPI0067708.3" "IPI00653981.1" [97] "IPI00640568.1" "IPI00642528.3" "IPI00642546.1" "IPI00642564.1" [101] "IPI00642732.2" "IPI00642584.1" "IPI00642564.1" "IPI00642565.1" [105] "IPI00644994.1" "IPI00642583.1" "IPI006443196.1" "IPI00643777.3" [113] "IPI00758866.1" "IPI00784216.3" "IPI00788640.1" "IPI0078877.1" [117] "IPI00789171.1" "IPI0078173.1" "IPI00798081.1" "IPI00798083.1" "IPI00798084.1" [129] "IPI00791170.1" "IPI00793651.1" "IPI0079302.1" "IPI00795642.2" "IPI00795642.1" "IPI0079655.1" [133] "IPI0079616.1" "IPI00795622.2" "IPI0079633.1" "IPI0079772.1" "IPI0079633.1" "IPI0079772.1" [143] "IPI0079623.1" "IPI00796354.1" "IPI0079721.7" "IPI00796354.1" "IPI0079722.1" [143] "IPI0079623.1" "IPI0079633.1" "IPI0079723.1" "IPI0079723.1" "IPI0079723.1" "IPI0079723.1" "					
[89] "IPI00555601.1" "IPI00555722.1" "IPI0055728.1" "IPI00639981.1" [93] "IPI00640568.1" "IPI00640568.3" "IPI00642564.1" "IPI00642564.1" [101] "IPI00640568.1" "IPI0064232.2" "IPI00642364.1" "IPI00642573.2" [105] "IPI00644792.1" "IPI0064258.3" "IPI00642646.1" "IPI00642573.2" [109] "IPI00644792.1" "IPI0064258.3" "IPI00743713.3" "IPI00746777.3" [113] "IPI00758806.1" "IPI00789301.1" "IPI00789301.1" "IPI0078971.1" [117] "IPI0079171.1" "IPI0079123.1" "IPI007930301.1" "IPI00791266.1" [129] "IPI0079171.1" "IPI00792375.1" "IPI0079554.1" "IPI0079554.1" [133] "IPI00794605.1" "IPI00796633.1" "IPI0079563.1" "IPI0079563.1" [141] "IPI00795549.1" "IPI00796633.1" "IPI00796735.1" "IPI0079727.4" [143] "IPI0079572.1" "IPI009572.1" "IPI0084133.1" [144] "IPI0079682.1" "IPI0079682.1" "IPI0098256.1" "IPI009871.2" [155] "IPI00815793.1" "IPI00979632.1" "IPI0098256.1" <td></td> <td></td> <td></td> <td></td> <td></td>					
[93] "IPI00556284.2" "IPI00604528.3" "IPI0067708.3" "IPI00639981.1" [97] "IPI00642732.2" "IPI00642820.1" "IPI00642546.1" "IPI00642732.2" [105] "IPI00644732.2" "IPI0064588.1" "IPI006446468.1" "IPI00645709.1" [109] "IPI00654709.1" "IPI00743142.2" "IPI00743713.3" "IPI00788737.1" [113] "IPI00758806.1" "IPI00788938.1" "IPI0078804.1" "IPI00788737.1" [117] "IPI0078836.1" "IPI00789081.1" "IPI00788737.1" [117] "IPI00791170.1" "IPI00792375.1" "IPI00792484.1" "IPI00794508.1" [129] "IPI0079215.1" "IPI00795622.2" "IPI0079505.1" "IPI0079653.1" "IPI00796735.1" [131] "IPI0079616.1" "IPI00796833.1" "IPI0079720.7" "IPI0079683.1" "IPI0079721.7" [143] "IPI0079682.1" "IPI00798351.1" "IPI0079721.7" "IPI00797358.1" [144] "IPI0079682.1" "IPI00798351.1" "IPI0079727.4" "IPI00797580.1" "IPI00880764.1" "IPI0081578.1" [157] "IPI00815793.1" "IPI0098891.2" "IPI0098254.1" "IPI0098254.1"					
[97] "IPI00640568.1" "IPI00642820.1" "IPI00642840.1" [101] "IPI00642732.2" "IPI00648848.1" "IPI00643196.1" "IPI00643575.1" [105] "IPI00654709.1" "IPI0073142.2" "IPI00743713.3" "IPI00788777.3" [113] "IPI00788866.1" "IPI00788938.1" "IPI00788961.1" "IPI00788977.1" [117] "IPI00788866.1" "IPI00789973.1" "IPI00789071.1" "IPI00798973.1" "IPI00798961.1" "IPI00790892.1" [125] "IPI0079170.1" "IPI00792375.1" "IPI00799207.1" "IPI0079565.1" "IPI00795655.1" [133] "IPI00794065.1" "IPI00795663.1" "IPI00795655.1" "IPI00795655.1" [137] "IPI0079616.1" "IPI00795633.1" "IPI00795633.1" "IPI00796633.1" "IPI00796735.1" [141] "IPI0079623.1" "IPI00796852.1" "IPI0079683.1" "IPI0079683.1" "IPI0079683.1" "IPI00797738.1" [143] "IPI0079673.1" "IPI00796852.1" "IPI0079683.1" "IPI0079772.1" "IPI0079773.1" "IPI0079772.1" [149] "IPI0079673.1" "IPI0079773.1" "IPI0081793.1" "IPI0084143.3.1" [157]					
[101] "IPI00642732.2" "IPI00642820.1" "IPI0064396.1" "IPI00643765.1" [105] "IPI00644994.1" "IPI00743142.2" "IPI00743713.3" "IPI00746777.3" [113] "IPI00759806.1" "IPI0078216.3" "IPI0078840.1" "IPI00788737.1" [117] "IPI00788836.1" "IPI00788938.1" "IPI0078901.1" "IPI00789731.4" [125] "IPI0079170.1" "IPI00789173.1" "IPI00789301.1" "IPI00790892.1" [125] "IPI0079170.1" "IPI00792375.1" "IPI00799322.1" "IPI00794666.1" [129] "IPI0079207.2" "IPI0079465.1" "IPI00795075.1" "IPI00794508.1" [133] "IPI0079616.1" "IPI0079665.1" "IPI00796075.1" "IPI0079675.1" [141] "IPI0079616.1" "IPI0079633.1" "IPI0079633.1" "IPI0079633.1" "IPI0079635.1" [143] "IPI0079613.1" "IPI0079635.1" "IPI0079835.1" "IPI0079835.1" "IPI00797221.7" [153] "IPI0079633.1" "IPI0079835.1" "IPI0083064.1" "IPI0084433.1" [164] "IPI00873455.1" "IPI009822.1" "IPI0087922.1" "IPI00844133.1" [173]					
[105] "IPI00644994.1" "IPI00645848.1" "IPI00646468.1" "IPI00647702.1" [109] "IPI00759806.1" "IPI00743142.2" "IPI00743713.3" "IPI00788737.1" [113] "IPI00759806.1" "IPI0078838.1" "IPI00788640.1" "IPI00788737.1" [117] "IPI00788836.1" "IPI00788938.1" "IPI0078907.1" "IPI00789174.4" [121] "IPI0079170.1" "IPI0079173.1" "IPI00799064.2" "IPI007990792.1" [123] "IPI0079207.2" "IPI00792375.1" "IPI00799322.1" "IPI00794508.1" [133] "IPI00795465.1" "IPI0079557.3" "IPI0079557.3" "IPI0079557.3" [141] "IPI0079549.1" "IPI0079633.1" "IPI0079633.1" "IPI0079673.1" [142] "IPI0079616.1" "IPI0079633.1" "IPI0079633.1" "IPI0079673.1" [153] "IPI0079623.1" "IPI0079633.1" "IPI0079633.1" "IPI0079633.1" "IPI0079633.1" [154] "IPI0079633.1" "IPI0079633.1" "IPI0079633.1" "IPI0079633.1" "IPI0079633.1" "IPI0079633.1" [155] "IPI00374253.1" "IPI0039064.1" "IPI0039064.1" "IPI0039266.1" "I					
[109] "IPI00654709.1" "IPI00743142.2" "IPI00743713.3" "IPI00746777.3" [113] "IPI00759806.1" "IPI00784216.3" "IPI00788640.1" "IPI00788737.1" [117] "IPI00788836.1" "IPI00789301.1" "IPI00789134.4" [121] "IPI00788737.1" "IPI00789301.1" "IPI00799134.4" [122] "IPI00788717.1" "IPI00789301.1" "IPI0079892.1" [123] "IPI0079207.2" "IPI00792375.1" "IPI00793922.1" "IPI00794666.1" [133] "IPI00794605.1" "IPI00795075.1" "IPI00794508.1" [141] "IPI00796616.1" "IPI00795075.1" "IPI00796735.3" [141] "IPI00796632.1" "IPI00796633.1" "IPI00796735.1" [143] "IPI00796823.1" "IPI00797038.1" "IPI00798351.1" "IPI00797270.4" [153] "IPI00815793.1" "IPI00815766.1" "IPI0084133.1" [154] "IPI0084789.3" "IPI00872487.1" "IPI0084291.2" [155] "IPI00815793.1" "IPI0090325.1" "IPI00909325.1" "IPI00909326.1" [165] "IPI00910420.1" "IPI00910421.1" "IPI00910754.1" "IPI00910754.1"					
[113] "IPI00759806.1" "IPI00784216.3" "IPI00788640.1" "IPI00788737.1" [117] "IPI00788836.1" "IPI00788938.1" "IPI00789081.1" "IPI00789134.4" [121] "IPI00789171.1" "IPI00789173.1" "IPI00789081.1" "IPI00790892.1" [125] "IPI0079170.1" "IPI00792375.1" "IPI00792448.1" "IPI00792655.1" [133] "IPI00792715.1" "IPI00794605.1" "IPI0079567.3" "IPI0079557.3" [141] "IPI0079616.1" "IPI00796333.1" "IPI0079633.1" "IPI0079635.1" [143] "IPI0079616.1" "IPI00796852.1" "IPI00796833.1" "IPI00796772.1." [144] "IPI00796823.1" "IPI00796852.1" "IPI0079683.1" "IPI00796772.1." [145] "IPI00797270.4" "IPI00797880.1" "IPI00798351.1" "IPI00847886.1" [157] "IPI00815793.1" "IPI0098741.2" "IPI0083064.1" "IPI0083866.1" [165] "IPI00873455.1" "IPI0099052.1" "IPI00908791.2" "IPI00908791.2" "IPI00908791.2" "IPI00909325.1" "IPI0090966.1" [177] "IPI009055.1" "IPI00901642.1" "IPI0091074.1" "IPI0091074.1" <td></td> <td></td> <td></td> <td></td> <td></td>					
[117] "IPI00788836.1" "IPI00789081.1" "IPI00789134.4" [121] "IPI00789171.1" "IPI00789173.1" "IPI00789301.1" "IPI00790892.1" [125] "IPI00791170.1" "IPI00791428.2" "IPI00791564.2" "IPI00791666.1" [129] "IPI0079207.2" "IPI00792375.1" "IPI00792488.1" "IPI00794605.1" [137] "IPI00794605.1" "IPI0079567.2" "IPI00795075.1" "IPI00794057.3" [141] "IPI0079652.1" "IPI00796633.1" "IPI00796735.1" [143] "IPI00796823.1" "IPI00796633.1" "IPI00796735.1" [143] "IPI0079623.1" "IPI0079685.1" "IPI007970721.7" [153] "IPI0079623.1" "IPI0079685.1" "IPI007970721.7" [153] "IPI00797270.4" "IPI00797580.1" "IPI0083064.1" "IPI00847989.3" [164] "IPI00847989.3" "IPI00871353.1" "IPI0098226.1" "IPI00908791.2" [165] "IPI00908791.2" "IPI00909256.1" "IPI00909266.1" "IPI00909266.1" [177] "IPI009158.1" "IPI00909266.1" "IPI00909266.1" "IPI0091743.1" "IPI0091743.1" [18] <t< td=""><td></td><td>"IPI00759806.1"</td><td>"IPI00784216.3"</td><td>"IPI00788640.1"</td><td>"IPI00788737.1"</td></t<>		"IPI00759806.1"	"IPI00784216.3"	"IPI00788640.1"	"IPI00788737.1"
[121] "IPI00789171.1" "IPI00789173.1" "IPI00789301.1" "IPI00790892.1" [125] "IPI00791170.1" "IPI00791428.2" "IPI00791564.2" "IPI00791666.1" [129] "IPI0079207.2" "IPI0079375.1" "IPI00793922.1" "IPI00794605.1" [133] "IPI00794605.1" "IPI00793665.1" "IPI0079557.3" "IPI0079557.3" [141] "IPI00795549.1" "IPI00796333.1" "IPI00796633.1" "IPI00796735.1" [143] "IPI00796632.1" "IPI00796633.1" "IPI00796735.1" [144] "IPI00796632.1" "IPI00796633.1" "IPI00796735.1" [144] "IPI00797270.4" "IPI00797580.1" "IPI0083064.1" "IPI00815786.1" [157] "IPI00815733.1" "IPI0083064.1" "IPI00847389.3" "IPI0081555.1" "IPI00872487.1" "IPI0098386.1" [166] "IPI00873455.1" "IPI00909256.1" "IPI00909325.1" "IPI00909443.1" [177] "IPI0091042.1" "IPI00910642.1" "IPI00910754.1" "IPI00910781.1" [173] "IPI0091074.1" "IPI00910690.1" "IPI0091074.1" "IPI0091074.1" "IPI0091074.1" [174] "IPI0091					
[129] "IPI00792207.2" "IPI00792375.1" "IPI00792448.1" "IPI00792655.1" [133] "IPI00792715.1" "IPI00793665.1" "IPI00793922.1" "IPI00794508.1" [137] "IPI00794605.1" "IPI00794991.1" "IPI00795075.1" "IPI00794508.1" [141] "IPI00795549.1" "IPI00796622.2" "IPI00795914.1" "IPI00796111.1" [145] "IPI00796116.1" "IPI00796633.1" "IPI00796633.1" "IPI00796735.1" [149] "IPI00796823.1" "IPI00796852.1" "IPI00797038.1" "IPI00796735.1" [157] "IPI00815793.1" "IPI00797580.1" "IPI00830064.1" "IPI00815786.1" [157] "IPI00815793.1" "IPI00815950.1" "IPI00830064.1" "IPI00844133.1" [161] "IPI00847989.3" "IPI00871353.1" "IPI00872487.1" "IPI00872991.2" [165] "IPI00873455.1" "IPI00902562.1" "IPI00903226.1" "IPI00908386.1" [177] "IPI00909595.1" "IPI0090256.1" "IPI00909325.1" "IPI00909143.1" [173] "IPI00909595.1" "IPI00909256.1" "IPI00909325.1" "IPI00909499.1" [181] "IPI0091052.1" "IPI00909256.1" "IPI00909325.1" "IPI00909499.1" [183] "IPI0091052.1" "IPI00910642.1" "IPI00910754.1" "IPI0090940.1" [183] "IPI0091074.1" "IPI00910990.1" "IPI00910754.1" [193] "IPI00916181.1" "IPI00916990.1" "IPI00916994.1" "IPI0091743.1" [193] "IPI0092680.1" "IPI0092650.1" "IPI00926110.1" "IPI00926319.1" [193] "IPI00927598.1" "IPI00927394.1" "IPI00930416.1" "IPI00927398.1" [204] "IPI00927598.1" "IPI00927394.1" "IPI00939637.1" "IPI00946302.1" [205] "IPI00927598.1" "IPI00927949.1" "IPI00936637.1" "IPI00946		"IPI00789171.1"	"IPI00789173.1"	"IPI00789301.1"	"IPI00790892.1"
[133] "IPI00792715.1" "IPI00793665.1" "IPI00793922.1" "IPI00794508.1" [137] "IPI00794605.1" "IPI00794991.1" "IPI00795075.1" "IPI00795257.3" [141] "IPI00795549.1" "IPI00795622.2" "IPI00795075.1" "IPI00796111.1" [145] "IPI00796116.1" "IPI0079633.1" "IPI00796633.1" "IPI00796735.1" [149] "IPI00796823.1" "IPI00796852.1" "IPI00797038.1" "IPI00797270.4" [153] "IPI00797270.4" "IPI00797580.1" "IPI00830064.1" "IPI00815786.1" [157] "IPI00815793.1" "IPI00815950.1" "IPI00830064.1" "IPI00872991.2" [165] "IPI00873455.1" "IPI00902542.1" "IPI00903226.1" "IPI00909386.1" [169] "IPI0090955.1" "IPI009090256.1" "IPI009090925.1" "IPI009090925.1" [177] "IPI00910420.1" "IPI00910642.1" "IPI00910754.1" "IPI009090949.1" [181] "IPI00910974.1" "IPI00910974.1" "IPI00910974.1" "IPI0091690.1" [183] "IPI00916818.1" "IPI0091692.1" "IPI0091692.1" "IPI0091713.1" [193] "IPI00916131.1" <	[125]	"IPI00791170.1"	"IPI00791428.2"	"IPI00791564.2"	"IPI00791666.1"
[137]"IPI00794605.1""IPI00794991.1""IPI00795075.1""IPI00795257.3"[141]"IPI00795549.1""IPI00795622.2""IPI00795914.1""IPI00796111.1"[145]"IPI00796116.1""IPI0079633.1""IPI0079633.1""IPI00796735.1"[149]"IPI00796823.1""IPI00796852.1""IPI00797038.1""IPI00797221.7"[153]"IPI00797270.4""IPI00797880.1""IPI00798351.1""IPI00815786.1"[157]"IPI00815793.1""IPI00815950.1""IPI00830064.1""IPI00844133.1"[161]"IPI00847989.3""IPI00871353.1""IPI00903226.1""IPI009087991.2"[165]"IPI0098791.2""IPI00902542.1""IPI00908927.1""IPI00909158.1"[177]"IPI00908791.2""IPI00909056.1""IPI00909325.1""IPI00909949.1"[173]"IPI00909595.1""IPI00909056.1""IPI00909926.1""IPI00909949.1"[181]"IPI00910420.1""IPI00910642.1""IPI00910754.1""IPI00910781.1"[183]"IPI00910974.1""IPI00916990.1""IPI00916993.1""IPI00917139.1"[193]"IPI00917133.1""IPI00917237.1""IPI00926110.1""IPI00927398.1"[201]"IPI00926810.1""IPI0092709.1""IPI00930461.1""IPI00936002.1"[203]"IPI00927598.1""IPI0093339.1""IPI00930461.1""IPI00946000.1"[213]"IPI00927598.1""IPI0093939.1""IPI00941338.1""IPI00945860.1"[214]"IPI00942265.1""IPI00945694.1""IPI00945766.2""IPI00945873.1"[215] <t< td=""><td>[129]</td><td>"IPI00792207.2"</td><td>"IPI00792375.1"</td><td>"IPI00792448.1"</td><td>"IPI00792655.1"</td></t<>	[129]	"IPI00792207.2"	"IPI00792375.1"	"IPI00792448.1"	"IPI00792655.1"
[141]"IPI00795549.1""IPI00795622.2""IPI00795914.1""IPI00796111.1"[145]"IPI00796116.1""IPI00796333.1""IPI00796633.1""IPI00796735.1"[149]"IPI00796823.1""IPI00796852.1""IPI00797038.1""IPI00797221.7"[153]"IPI00797270.4""IPI00797580.1""IPI00798351.1""IPI00815786.1"[157]"IPI00815793.1""IPI00815950.1""IPI0083064.1""IPI0084133.1"[161]"IPI00847989.3""IPI00871353.1""IPI00872487.1""IPI00872991.2"[165]"IPI00873455.1""IPI00902542.1""IPI00903226.1""IPI00909143.1"[173]"IPI00908791.2""IPI00909256.1""IPI00909325.1""IPI00909143.1"[173]"IPI00909158.1""IPI00909266.1""IPI00909325.1""IPI00909949.1"[181]"IPI00910420.1""IPI00910642.1""IPI00910754.1""IPI00910781.1"[185]"IPI00910974.1""IPI00910979.1""IPI00915933.1""IPI00917139.1"[193]"IPI00916818.1""IPI00917237.1""IPI0091694.1""IPI00926319.1"[201]"IPI0092697.1""IPI00927398.1""IPI00926319.1"[203]"IPI00926810.1""IPI0093939.1""IPI00930637.1""IPI00940003.1"[204]"IPI00927598.1""IPI00939339.1""IPI00930637.1""IPI00940003.1"[205]"IPI00927598.1""IPI00939339.1""IPI00930637.1""IPI00940003.1"[206]"IPI00927598.1""IPI00939339.1""IPI00930637.1""IPI00940003.1"[213]"IPI00927598.1" <t< td=""><td>[133]</td><td>"IPI00792715.1"</td><td>"IPI00793665.1"</td><td>"IPI00793922.1"</td><td>"IPI00794508.1"</td></t<>	[133]	"IPI00792715.1"	"IPI00793665.1"	"IPI00793922.1"	"IPI00794508.1"
[145]"IPI00796116.1""IPI00796333.1""IPI00796633.1""IPI00796735.1"[149]"IPI00796823.1""IPI00796852.1""IPI00797038.1""IPI00797221.7"[153]"IPI00797270.4""IPI00797880.1""IPI00798351.1""IPI00815786.1"[157]"IPI00815793.1""IPI00815950.1""IPI00830064.1""IPI00847399.1"[161]"IPI00847989.3""IPI00871353.1""IPI00872487.1""IPI0098791.2"[165]"IPI0097455.1""IPI00902542.1""IPI00908927.1""IPI00909143.1"[173]"IPI00909158.1""IPI00909256.1""IPI00909325.1""IPI00909949.1"[174]"IPI00910420.1""IPI00910642.1""IPI00910754.1""IPI00910781.1"[185]"IPI00910974.1""IPI00916990.1""IPI00916994.1""IPI00917139.1"[193]"IPI00917193.1""IPI00925520.1""IPI00926110.1""IPI00926319.1"[201]"IPI00926810.1""IPI00927398.1""IPI00936602.1"[202]"IPI00927598.1""IPI009333.1""IPI00946319.1"[203]"IPI00927598.1""IPI0093333.1""IPI00927398.1"[204]"IPI00927598.1""IPI0093333.1""IPI00936602.1"[205]"IPI00927598.1""IPI0093333.1""IPI0093667.1"[206]"IPI009422697.1""IPI0093333.1""IPI0093667.1"[207]"IPI0093286.1""IPI00941033.1""IPI00945309.1"[208]"IPI009426810.1""IPI0094569.1""IPI00945309.1"[209]"IPI00942681.1""IPI0094569.1""IPI00945309.1" <tr<< td=""><td>[137]</td><td>"IPI00794605.1"</td><td>"IPI00794991.1"</td><td>"IPI00795075.1"</td><td>"IPI00795257.3"</td></tr<<>	[137]	"IPI00794605.1"	"IPI00794991.1"	"IPI00795075.1"	"IPI00795257.3"
[149] "IPI00796823.1" "IPI00796852.1" "IPI00797038.1" "IPI00797221.7" [153] "IPI00797270.4" "IPI00797580.1" "IPI00798351.1" "IPI00815786.1" [157] "IPI00815793.1" "IPI00815950.1" "IPI00830064.1" "IPI0084133.1" [161] "IPI00847989.3" "IPI00871353.1" "IPI00872487.1" "IPI00872991.2" [165] "IPI00873455.1" "IPI00902542.1" "IPI00903226.1" "IPI00909143.1" [169] "IPI00908791.2" "IPI00909256.1" "IPI00909325.1" "IPI00909560.1" [177] "IPI00909595.1" "IPI0090909256.1" "IPI009090925.1" "IPI009090925.1" [177] "IPI00910420.1" "IPI00910642.1" "IPI00910754.1" "IPI00910781.1" [183] "IPI00910974.1" "IPI00910979.1" "IPI0091753.1" "IPI009173.1" [183] "IPI00916818.1" "IPI00916990.1" "IPI0091743.1" "IPI0091733.1" "IPI0091743.1" [193] "IPI00917193.1" "IPI00917237.1" "IPI00926110.1" "IPI00926319.1" [201] "IPI00926810.1" "IPI00927598.1" "IPI00930339.1" "IPI00930416.1" "IPI00936002.1" <tr< td=""><td>[141]</td><td>"IPI00795549.1"</td><td>"IPI00795622.2"</td><td>"IPI00795914.1"</td><td>"IPI00796111.1"</td></tr<>	[141]	"IPI00795549.1"	"IPI00795622.2"	"IPI00795914.1"	"IPI00796111.1"
[153]"IPI00797270.4""IPI00797580.1""IPI00798351.1""IPI00815786.1"[157]"IPI00815793.1""IPI00815950.1""IPI00830064.1""IPI00844133.1"[161]"IPI00847989.3""IPI00871353.1""IPI00872487.1""IPI00872991.2"[165]"IPI00973455.1""IPI00902542.1""IPI00903226.1""IPI00908886.1"[169]"IPI00908791.2""IPI00908881.2""IPI00908927.1""IPI00909143.1"[173]"IPI00909158.1""IPI00909256.1""IPI00909325.1""IPI00909949.1"[177]"IPI0091092.5.1""IPI00909694.1""IPI00909829.1""IPI00910781.1"[181]"IPI00910974.1""IPI00910642.1""IPI00910754.1""IPI00910781.1"[185]"IPI00910974.1""IPI00916990.1""IPI00916993.1""IPI00916206.1"[189]"IPI00916818.1""IPI00916990.1""IPI00916994.1""IPI0091739.1"[193]"IPI00917193.1""IPI00917237.1""IPI00926319.1"[201]"IPI00926810.1""IPI0092739.1""IPI00926319.1"[201]"IPI00926810.1""IPI0092739.1""IPI00926319.1"[203]"IPI00927598.1""IPI00927949.1""IPI00930367.1""IPI00940003.1"[213]"IPI0094201.1""IPI00941093.1""IPI00941338.1""IPI00945873.1"[214]"IPI00945625.1""IPI00945694.1""IPI00945766.2""IPI00945873.1"[225]"IPI00946018.1""IPI00946404.1""IPI00946812.1""IPI00946252.1"[229]"IPI0094600.1""IPI00946404.1""IPI00946812.1"<	[145]	"IPI00796116.1"	"IPI00796333.1"	"IPI00796633.1"	"IPI00796735.1"
<pre>[157] "IPI00815793.1" "IPI00815950.1" "IPI00830064.1" "IPI00844133.1" [161] "IPI00847989.3" "IPI00871353.1" "IPI00872487.1" "IPI00872991.2" [165] "IPI00873455.1" "IPI00902542.1" "IPI00903226.1" "IPI00908386.1" [169] "IPI00908791.2" "IPI00908881.2" "IPI00908927.1" "IPI00909143.1" [173] "IPI00909158.1" "IPI0090256.1" "IPI00908927.1" "IPI00909143.1" [177] "IPI00909595.1" "IPI00909256.1" "IPI00909325.1" "IPI00909949.1" [181] "IPI00910420.1" "IPI00910642.1" "IPI00910754.1" "IPI00910781.1" [185] "IPI00910974.1" "IPI00910642.1" "IPI00915933.1" "IPI00916206.1" [189] "IPI00916818.1" "IPI00916990.1" "IPI00915933.1" "IPI00916206.1" [193] "IPI00916818.1" "IPI00916990.1" "IPI00916994.1" "IPI00917139.1" [193] "IPI00917193.1" "IPI00917237.1" "IPI00917473.1" "IPI00917841.1" [201] "IPI00926810.1" "IPI00925520.1" "IPI00926110.1" "IPI00926319.1" [201] "IPI00926810.1" "IPI00927039.1" "IPI00930416.1" "IPI00926319.1" [201] "IPI0092558.1" "IPI00927039.1" "IPI00930416.1" "IPI00946319.1" [201] "IPI0094201.1" "IPI0094509.1" "IPI0093637.1" "IPI00945309.1" [201] "IPI00945625.1" "IPI00945694.1" "IPI00945309.1" [202] "IPI00945625.1" "IPI00945694.1" "IPI00945309.1" [203] "IPI00945625.1" "IPI00945625.1" "IPI00945766.2" "IPI00945873.1" [204] "IPI00946400.1" "IPI00946404.1" "IPI00946812.1" "IPI00947127.1"</pre>	[149]	"IPI00796823.1"	"IPI00796852.1"	"IPI00797038.1"	"IPI00797221.7"
<pre>[161] "IPI00847989.3" "IPI00871353.1" "IPI00872487.1" "IPI00872991.2" [165] "IPI00873455.1" "IPI00902542.1" "IPI00903226.1" "IPI00908386.1" [169] "IPI00908791.2" "IPI00908881.2" "IPI00908927.1" "IPI00909143.1" [173] "IPI00909158.1" "IPI00909256.1" "IPI00909325.1" "IPI00909560.1" [177] "IPI00909595.1" "IPI00909694.1" "IPI00909829.1" "IPI00909949.1" [181] "IPI00910420.1" "IPI00910642.1" "IPI00910754.1" "IPI00910781.1" [185] "IPI00910974.1" "IPI00910642.1" "IPI00915933.1" "IPI00910781.1" [189] "IPI00916818.1" "IPI00916990.1" "IPI00916994.1" "IPI00916206.1" [193] "IPI00916818.1" "IPI00916990.1" "IPI00916994.1" "IPI00917139.1" [193] "IPI00917193.1" "IPI00917237.1" "IPI00917473.1" "IPI00917841.1" [201] "IPI00926810.1" "IPI00925520.1" "IPI00926110.1" "IPI00926319.1" [201] "IPI00926810.1" "IPI00927039.1" "IPI00927177.1" "IPI00927398.1" [205] "IPI00927598.1" "IPI00927039.1" "IPI00930416.1" "IPI00936002.1" [209] "IPI00939286.1" "IPI00939339.1" "IPI0093637.1" "IPI00940003.1" [213] "IPI00940201.1" "IPI00941093.1" "IPI00945309.1" "IPI00945866.1" [217] "IPI00945625.1" "IPI00945694.1" "IPI00945766.2" "IPI00945873.1" [225] "IPI00946018.1" "IPI00946400.1" "IPI00946173.1" "IPI009461727.1"</pre>	[153]	"IPI00797270.4"	"IPI00797580.1"	"IPI00798351.1"	"IPI00815786.1"
[165] "IPI00873455.1""IPI00902542.1""IPI00903226.1""IPI00908386.1"[169] "IPI00908791.2""IPI00908881.2""IPI00908927.1""IPI00909143.1"[173] "IPI00909158.1""IPI00909256.1""IPI00909325.1""IPI00909560.1"[177] "IPI00909595.1""IPI00909694.1""IPI00909829.1""IPI00909949.1"[181] "IPI00910420.1""IPI00910642.1""IPI00910754.1""IPI00910781.1"[185] "IPI00910974.1""IPI00910979.1""IPI00915933.1""IPI00916206.1"[189] "IPI00916818.1""IPI00916990.1""IPI00916994.1""IPI00917139.1"[193] "IPI00917193.1""IPI00917237.1""IPI00917473.1""IPI00917841.1"[197] "IPI0092697.1""IPI00925520.1""IPI00926110.1""IPI00926319.1"[201] "IPI00926810.1""IPI00927039.1""IPI00927177.1""IPI00926319.1"[205] "IPI00927598.1""IPI00927949.1""IPI00930416.1""IPI00940003.1"[213] "IPI00940201.1""IPI00941093.1""IPI00945309.1""IPI00945466.1"[214] "IPI00945625.1""IPI00945694.1""IPI00945766.2""IPI00945873.1"[225] "IPI00946018.1""IPI00946160.1""IPI00946173.1""IPI00946252.1"[229] "IPI00946400.1""IPI00946404.1""IPI00946812.1""IPI009467127.1"	[157]	"IPI00815793.1"	"IPI00815950.1"	"IPI00830064.1"	"IPI00844133.1"
[169] "IPI00908791.2""IPI00908881.2""IPI00908927.1""IPI00909143.1"[173] "IPI00909158.1""IPI00909256.1""IPI00909325.1""IPI00909560.1"[177] "IPI00909595.1""IPI00909694.1""IPI00909829.1""IPI00909949.1"[181] "IPI00910420.1""IPI00910642.1""IPI00910754.1""IPI00910781.1"[185] "IPI00910974.1""IPI00910979.1""IPI00915933.1""IPI00916206.1"[189] "IPI00916818.1""IPI00916990.1""IPI00916994.1""IPI00917139.1"[193] "IPI00917193.1""IPI00917237.1""IPI00917473.1""IPI00917841.1"[197] "IPI0092697.1""IPI00925520.1""IPI00926110.1""IPI00926319.1"[201] "IPI00926810.1""IPI00927039.1""IPI00930416.1""IPI00927398.1"[205] "IPI00927598.1""IPI00927949.1""IPI00930416.1""IPI00940003.1"[213] "IPI00940201.1""IPI00941093.1""IPI00941338.1""IPI00941899.1"[217] "IPI00945625.1""IPI00945694.1""IPI00945766.2""IPI00945873.1"[225] "IPI00946018.1""IPI00946400.1""IPI009466173.1""IPI00946252.1"[229] "IPI00946400.1""IPI00946404.1""IPI00946812.1""IPI00947127.1"	[161]	"IPI00847989.3"	"IPI00871353.1"	"IPI00872487.1"	"IPI00872991.2"
[173] "IPI00909158.1" "IPI00909256.1" "IPI00909325.1" "IPI00909560.1"[177] "IPI00909595.1" "IPI00909694.1" "IPI00909829.1" "IPI00909949.1"[181] "IPI00910420.1" "IPI00910642.1" "IPI00910754.1" "IPI00910781.1"[185] "IPI00910974.1" "IPI00910979.1" "IPI00915933.1" "IPI00916206.1"[189] "IPI00916818.1" "IPI00916990.1" "IPI00916994.1" "IPI00917139.1"[193] "IPI00917193.1" "IPI00917237.1" "IPI00917473.1" "IPI00917841.1"[197] "IPI0092697.1" "IPI00925520.1" "IPI00926110.1" "IPI00926319.1"[201] "IPI00926810.1" "IPI00927039.1" "IPI00927177.1" "IPI00927398.1"[205] "IPI00927598.1" "IPI00927949.1" "IPI00930416.1" "IPI00936002.1"[209] "IPI00946810.1" "IPI00941093.1" "IPI00945309.1" "IPI00944899.1"[213] "IPI0094294.1" "IPI00942961.1" "IPI00945309.1"[221] "IPI00946018.1" "IPI00946404.1" "IPI00946173.1" "IPI00946252.1"	[165]	"IPI00873455.1"	"IPI00902542.1"	"IPI00903226.1"	"IPI00908386.1"
<pre>[177] "IPI00909595.1" "IPI00909694.1" "IPI00909829.1" "IPI00909949.1" [181] "IPI00910420.1" "IPI00910642.1" "IPI00910754.1" "IPI00910781.1" [185] "IPI00910974.1" "IPI00910979.1" "IPI00915933.1" "IPI00916206.1" [189] "IPI00916818.1" "IPI00916990.1" "IPI00916994.1" "IPI00917139.1" [193] "IPI00917193.1" "IPI00917237.1" "IPI00917473.1" "IPI00917841.1" [197] "IPI0092697.1" "IPI00925520.1" "IPI00926110.1" "IPI00926319.1" [201] "IPI00926810.1" "IPI00927039.1" "IPI00926110.1" "IPI00927398.1" [205] "IPI00927598.1" "IPI00927039.1" "IPI00930416.1" "IPI00927398.1" [209] "IPI00939286.1" "IPI00939339.1" "IPI00930637.1" "IPI00940003.1" [213] "IPI00940201.1" "IPI00941093.1" "IPI00941338.1" "IPI00941899.1" [217] "IPI00942494.1" "IPI00945694.1" "IPI00945766.2" "IPI00945873.1" [225] "IPI00946018.1" "IPI00946160.1" "IPI00946173.1" "IPI00946252.1" [229] "IPI00946400.1" "IPI00946404.1" "IPI00946812.1" "IPI00947127.1"</pre>	[169]	"IPI00908791.2"	"IPI00908881.2"	"IPI00908927.1"	"IPI00909143.1"
<pre>[181] "IPI00910420.1" "IPI00910642.1" "IPI00910754.1" "IPI00910781.1" [185] "IPI00910974.1" "IPI00910979.1" "IPI00915933.1" "IPI00916206.1" [189] "IPI00916818.1" "IPI00916990.1" "IPI00916994.1" "IPI00917139.1" [193] "IPI00917193.1" "IPI00917237.1" "IPI00917473.1" "IPI00917841.1" [197] "IPI00922697.1" "IPI00925520.1" "IPI00926110.1" "IPI00926319.1" [201] "IPI00926810.1" "IPI00927039.1" "IPI00927177.1" "IPI00927398.1" [205] "IPI00927598.1" "IPI00927949.1" "IPI00930416.1" "IPI00936002.1" [209] "IPI00939286.1" "IPI00939339.1" "IPI00939637.1" "IPI00940003.1" [213] "IPI00940201.1" "IPI00941093.1" "IPI00941338.1" "IPI00941899.1" [217] "IPI00942494.1" "IPI00945694.1" "IPI00945309.1" "IPI00945873.1" [225] "IPI00946018.1" "IPI00946160.1" "IPI00946812.1" "IPI00947127.1"</pre>	[173]	"IPI00909158.1"	"IPI00909256.1"	"IPI00909325.1"	"IPI00909560.1"
<pre>[185] "IPI00910974.1" "IPI00910979.1" "IPI00915933.1" "IPI00916206.1" [189] "IPI00916818.1" "IPI00916990.1" "IPI00916994.1" "IPI00917139.1" [193] "IPI00917193.1" "IPI00917237.1" "IPI00917473.1" "IPI00917841.1" [197] "IPI00922697.1" "IPI00925520.1" "IPI00926110.1" "IPI00926319.1" [201] "IPI00926810.1" "IPI00927039.1" "IPI00927177.1" "IPI00927398.1" [205] "IPI00927598.1" "IPI00927949.1" "IPI00930416.1" "IPI00936002.1" [209] "IPI00939286.1" "IPI00939339.1" "IPI00939637.1" "IPI00940003.1" [213] "IPI00940201.1" "IPI00941093.1" "IPI00941338.1" "IPI00941899.1" [217] "IPI00942494.1" "IPI00942961.1" "IPI00945309.1" "IPI00945466.1" [221] "IPI00945625.1" "IPI00945694.1" "IPI00945766.2" "IPI00945873.1" [225] "IPI00946018.1" "IPI00946160.1" "IPI00946812.1" "IPI00947127.1"</pre>	[177]	"IPI00909595.1"	"IPI00909694.1"	"IPI00909829.1"	"IPI00909949.1"
<pre>[189] "IPI00916818.1" "IPI00916990.1" "IPI00916994.1" "IPI00917139.1" [193] "IPI00917193.1" "IPI00917237.1" "IPI00917473.1" "IPI00917841.1" [197] "IPI00922697.1" "IPI00925520.1" "IPI00926110.1" "IPI00926319.1" [201] "IPI00926810.1" "IPI00927039.1" "IPI00927177.1" "IPI00927398.1" [205] "IPI00927598.1" "IPI00927949.1" "IPI00930416.1" "IPI00936002.1" [209] "IPI00939286.1" "IPI00939339.1" "IPI00930416.1" "IPI00940003.1" [213] "IPI00940201.1" "IPI00941093.1" "IPI00941338.1" "IPI00941899.1" [217] "IPI00942494.1" "IPI00942961.1" "IPI00945309.1" "IPI00945466.1" [221] "IPI00945625.1" "IPI00945694.1" "IPI00945766.2" "IPI00945873.1" [225] "IPI00946018.1" "IPI00946160.1" "IPI00946812.1" "IPI00947127.1"</pre>	[181]	"IPI00910420.1"	"IPI00910642.1"	"IPI00910754.1"	"IPI00910781.1"
<pre>[193] "IPI00917193.1" "IPI00917237.1" "IPI00917473.1" "IPI00917841.1" [197] "IPI00922697.1" "IPI00925520.1" "IPI00926110.1" "IPI00926319.1" [201] "IPI00926810.1" "IPI00927039.1" "IPI00927177.1" "IPI00927398.1" [205] "IPI00927598.1" "IPI00927949.1" "IPI00930416.1" "IPI00936002.1" [209] "IPI00939286.1" "IPI00939339.1" "IPI00939637.1" "IPI00940003.1" [213] "IPI00940201.1" "IPI00941093.1" "IPI00941338.1" "IPI00941899.1" [217] "IPI00942494.1" "IPI00942961.1" "IPI00945309.1" "IPI00945466.1" [221] "IPI00945625.1" "IPI00945694.1" "IPI00945766.2" "IPI00945873.1" [225] "IPI00946018.1" "IPI00946160.1" "IPI00946173.1" "IPI00946252.1"</pre>	[185]	"IPI00910974.1"	"IPI00910979.1"	"IPI00915933.1"	"IPI00916206.1"
<pre>[197] "IPI00922697.1" "IPI00925520.1" "IPI00926110.1" "IPI00926319.1" [201] "IPI00926810.1" "IPI00927039.1" "IPI00927177.1" "IPI00927398.1" [205] "IPI00927598.1" "IPI00927949.1" "IPI00930416.1" "IPI00936002.1" [209] "IPI00939286.1" "IPI00939339.1" "IPI00939637.1" "IPI00940003.1" [213] "IPI00940201.1" "IPI00941093.1" "IPI00941338.1" "IPI00941899.1" [217] "IPI00942494.1" "IPI00942961.1" "IPI00945309.1" "IPI00945466.1" [221] "IPI00945625.1" "IPI00945694.1" "IPI00945766.2" "IPI00945873.1" [225] "IPI00946018.1" "IPI00946160.1" "IPI00946173.1" "IPI00946252.1"</pre>	[189]	"IPI00916818.1"	"IPI00916990.1"	"IPI00916994.1"	"IPI00917139.1"
<pre>[201] "IPI00926810.1" "IPI00927039.1" "IPI00927177.1" "IPI00927398.1" [205] "IPI00927598.1" "IPI00927949.1" "IPI00930416.1" "IPI00936002.1" [209] "IPI00939286.1" "IPI00939339.1" "IPI00939637.1" "IPI00940003.1" [213] "IPI00940201.1" "IPI00941093.1" "IPI00941338.1" "IPI00941899.1" [217] "IPI00942494.1" "IPI00942961.1" "IPI00945309.1" "IPI00945466.1" [221] "IPI00945625.1" "IPI00945694.1" "IPI00945766.2" "IPI00945873.1" [225] "IPI00946018.1" "IPI00946160.1" "IPI00946173.1" "IPI00946252.1" [229] "IPI00946400.1" "IPI00946404.1" "IPI00946812.1" "IPI00947127.1"</pre>		"IPI00917193.1"	"IPI00917237.1"	"IPI00917473.1"	"IPI00917841.1"
<pre>[205] "IPI00927598.1" "IPI00927949.1" "IPI00930416.1" "IPI00936002.1" [209] "IPI00939286.1" "IPI00939339.1" "IPI00939637.1" "IPI00940003.1" [213] "IPI00940201.1" "IPI00941093.1" "IPI00941338.1" "IPI00941899.1" [217] "IPI00942494.1" "IPI00942961.1" "IPI00945309.1" "IPI00945466.1" [221] "IPI00945625.1" "IPI00945694.1" "IPI00945766.2" "IPI00945873.1" [225] "IPI00946018.1" "IPI00946160.1" "IPI00946173.1" "IPI00946252.1" [229] "IPI00946400.1" "IPI00946404.1" "IPI00946812.1" "IPI00947127.1"</pre>	[197]	"IPI00922697.1"	"IPI00925520.1"	"IPI00926110.1"	"IPI00926319.1"
<pre>[209] "IPI00939286.1" "IPI00939339.1" "IPI00939637.1" "IPI00940003.1" [213] "IPI00940201.1" "IPI00941093.1" "IPI00941338.1" "IPI00941899.1" [217] "IPI00942494.1" "IPI00942961.1" "IPI00945309.1" "IPI00945466.1" [221] "IPI00945625.1" "IPI00945694.1" "IPI00945766.2" "IPI00945873.1" [225] "IPI00946018.1" "IPI00946160.1" "IPI00946173.1" "IPI00946252.1" [229] "IPI00946400.1" "IPI00946404.1" "IPI00946812.1" "IPI00947127.1"</pre>	[201]	"IPI00926810.1"	"IPI00927039.1"	"IPI00927177.1"	"IPI00927398.1"
<pre>[213] "IPI00940201.1" "IPI00941093.1" "IPI00941338.1" "IPI00941899.1" [217] "IPI00942494.1" "IPI00942961.1" "IPI00945309.1" "IPI00945466.1" [221] "IPI00945625.1" "IPI00945694.1" "IPI00945766.2" "IPI00945873.1" [225] "IPI00946018.1" "IPI00946160.1" "IPI00946173.1" "IPI00946252.1" [229] "IPI00946400.1" "IPI00946404.1" "IPI00946812.1" "IPI00947127.1"</pre>	[205]	"IPI00927598.1"	"IPI00927949.1"	"IPI00930416.1"	"IPI00936002.1"
[217] "IPI00942494.1" "IPI00942961.1" "IPI00945309.1" "IPI00945466.1" [221] "IPI00945625.1" "IPI00945694.1" "IPI00945766.2" "IPI00945873.1" [225] "IPI00946018.1" "IPI00946160.1" "IPI00946173.1" "IPI00946252.1" [229] "IPI00946400.1" "IPI00946404.1" "IPI00946812.1" "IPI00947127.1"	[209]	"IPI00939286.1"	"IPI00939339.1"	"IPI00939637.1"	"IPI00940003.1"
[221] "IPI00945625.1" "IPI00945694.1" "IPI00945766.2" "IPI00945873.1" [225] "IPI00946018.1" "IPI00946160.1" "IPI00946173.1" "IPI00946252.1" [229] "IPI00946400.1" "IPI00946404.1" "IPI00946812.1" "IPI00947127.1"	[213]				
[225] "IPI00946018.1" "IPI00946160.1" "IPI00946173.1" "IPI00946252.1" [229] "IPI00946400.1" "IPI00946404.1" "IPI00946812.1" "IPI00947127.1"		"IPI00942494.1"	"IPI00942961.1"	"IPI00945309.1"	"IPI00945466.1"
[229] "IPI00946400.1" "IPI00946404.1" "IPI00946812.1" "IPI00947127.1"	[221]				
	[233]				
[237] "IPI00952964.1" "IPI00955788.1" "IPI00955815.1" "IPI00955977.1"	[237]	"IPI00952964.1"	"IPI00955788.1"	"IPI00955815.1"	"IPI00955977.1"

> Lkeys(org.Hs.ipiPATH)[1:3]

```
[1] "IPI00000005.1" "IPI00000006.1" "IPI00000013.1"
> Rkeys(org.Hs.ipiPATH)[1:3]
[1] "hsa00010" "hsa00020" "hsa00030"
> org.Hs.ipi_dbschema()
-- IPI_DB schema
CREATE TABLE basic (
  ipi_id CHAR(13) NOT NULL,
                                        -- IPI Protein Identifier
                                            -- IPI Acession Number in current vers
  ipi_ac VARCHAR(20) NOT NULL,
  len INTEGER NOT NULL,
 mw INTEGER NOT NULL,
                                          -- Protein Description
  de VARCHAR(255) NOT NULL,
  symbol VARCHAR(80) NOT NULL,
                                         -- Swiss-Prot Primary Acession Number
  sp_ac VARCHAR(20) NOT NULL,
                                          -- Swiss-Prot Protein Identifier
  sp_id VARCHAR(20) NOT NULL,
  ref_id VARCHAR(20) NOT NULL,
                                           -- RefSeq accession number
                                      -- NCBI Protein GI
  gi VARCHAR(10) NOT NULL,
 gene_id VARCHAR(10) NOT NULL
                                  -- Entrez Gene ID
 unigene_id VARCHAR(10) NOT NULL,
kegg id VARCHAR(20) NOT NULL.
                                              -- UniGene ID
                                        -- KEGG gene ID
 kegg_id VARCHAR(20) NOT NULL,
 FOREIGN KEY (ipi_id)
);
CREATE TABLE seq (
  ipi_id CHAR(13) NOT NULL,
                                         -- IPI Protein Identifier
  seq text NOT NULL,
                                 -- Protein Sequence
 FOREIGN KEY (ipi_id)
);
CREATE TABLE ipiac (
                                     -- IPI Protein Identifier
  ipi_id CHAR(13) NOT NULL,
                                         -- IPI Protein Acession Number
  ipi_acs CHAR(11) NOT NULL,
  FOREIGN KEY (ipi_id)
);
CREATE TABLE go (
                                      -- IPI Protein Identifier
  ipi_id CHAR(13) NOT NULL,
```

```
-- GO ID
  go_id CHAR(10) NOT NULL,
  evidence CHAR(3) NOT NULL,
                                                 -- GO evidence code
  ontology CHAR(2) NOT NULL,
                                                 -- GO ontology
  FOREIGN KEY (ipi_id)
);
CREATE TABLE path (
  ipi_id CHAR(13) NOT NULL,
                                           -- IPI Protein Identifier
  path_id CHAR(5) NOT NULL,
                                                 -- KEGG pathway short ID
  FOREIGN KEY (ipi_id)
};
CREATE TABLE pfam (
  ipi_id CHAR(13) NOT NULL,
                                          -- IPI Protein Identifier
  pfam_id CHAR(7) NULL,
                                                  -- Pfam ID
  FOREIGN KEY (ipi_id)
);
CREATE TABLE interpro (
  ipi_id CHAR(13) NOT NULL,
                                           -- IPI Protein Identifier
  interpro_id CHAR(9) NOT NULL,
                                                   -- InterPro ID
  FOREIGN KEY (ipi_id)
);
CREATE TABLE prosite (
  ipi_id CHAR(13) NOT NULL,
                                         -- IPI Protein Identifier
  prosite_id CHAR(7) NULL,
                                                   -- PROSITE ID
  FOREIGN KEY (ipi_id)
);
> selectSQL <- paste("SELECT ipi_id, de", "FROM basic", "WHERE de like '%histone%</p>
> tmp3 <- dbGetQuery(org.Hs.ipi_dbconn(), selectSQL)</pre>
> tmp3[1:3, ]
                                                                        de
         ipi_id
1 IPI00001830.1 HETEROCHROMATIN-SPECIFIC NONHISTONE PROTEIN (FRAGMENT).
2 IPI00002831.4
                             HISTONE DEACETYLASE COMPLEX SUBUNIT SAP30L.
3 IPI00002929.2 ISOFORM 3 OF HISTONE-LYSINE N-METHYLTRANSFERASE SUV39H2.
```

3 Function Description

3.1 Getting URL and Version

To download data file from public database, the first step is getting its URL and release/version information. URLs of supported databases are stored in "data/sourceURLs.txt". Following functions are used to get the url:

- getSrcUrl return url according given database.
- getALLUrl return urls for all databases used in PAnnBuilder packages.
- getSrcBuilt return release/version according given database.
- getALLBuilt return release/version information for all databases used in PAnnBuilder packages.

3.2 Parsing and Writing Data

Parsing is a key step to convert original data file to R object. Sometimes R is directly used to parse and write data. But for large data file or complicated data format, perl is firstly employed to quickly process data, and then R function reads the result file into R objects.

3.2.1 Employing perl program to parse data

Segment of perl program is written into file in "inst/scripts". Name and function of these parser files are as follows:

- spParser parse protein data from SwissProt or TrEMBL
- ipiParser parse protein data from IPI
- refseqParser parse protein data from NCBI RefSeq
- equalParser find protein ID mapping with equal sequences
- mergeParser merge different ID mapping files
- mppiParser parse protein protein interaction data from MIPS
- paParser parse data from PeptideAtlas
- dbsublocParser parse data from DBSubLoc
- pfamNameParser parse domain id and name from Pfam
- blastParser filter the results of blast

Function fileMuncher and fileMuncher_DB perl file based on given parser file and additional input data file, then perform this perl program via R.

3.2.2 Writing data using R

Besides using perl program, R functions also parse data from simple data file and store them as R environment objects or Bimap objects.

- createEmptyDPkg create an empty R packge at given directory.
- writeSQ write sequence data into R package.
- writeName parse mulitple data file, and write the mapping of id and name into R package. It employs writeGOName, writeKEGGName, writePFAMName, writeINTERPROName and writeTAXName to respectively write data from GO, KEGG, Pfam, InterPro and TAX.
- writeSCOPData parse structural classification of proteins from SCOP database.
- writeSubCellData parse data from protein subcellular location databases, and write into R package. It employs writeBACELLOData and writeDBSUBLOCData to respectively write data from BaCelLo and DBSubLoc.
- writeIntData parse data from protein-protein/domain-domain interaction databases, and write into R package. It employs writeGENEINTData, writeINTACTData, writeMP-PIData, write3DIDData and writeDOMINEData to respectively write data from NCBI gene interaction data file, EBI intact, MIPS interaction data, 3DID database and DOMINE database.
- writePtmData parse database involving protein post-translational modifications, and write into R package. It employs writeSYSPTMData to write data from SysPTM database.
- writeBfData write data involving body fulids proteomics into R package. It employs writeSYSB0DYFLUIDData to write data from Sys-BodyFluid database.
- writeGOAData write gene ontology terms from GOA database into R package.
- writeHomoloGeneData write homolog groups from NCBI HomoloGene into R package.
- writeInParanoidData write paralog groups from InParanoid into R package.
- writePeptideAtlasData write peptides identified by Mass Spectrometry from PeptideAtlas database into R package.
- writeMeta_DB write meta information about the annotation package into SQLite-based package.

- writeData_DB parse data from databases, and write data as tables into SQLite-based R package. It employs writeSPData_DB, writeIPIData_DB, writeREFSEQData_DB, writeGENEINTData_DB, writeINTACTData_DB, writeMPPIData_DB, write3DIDData_DB, writeSYSBODYFLUIDData_DB, writeSYSPTMData_DB, writeSCOP-Data_DB, writeBACELLOData_DB, writeDBSUBLOCData_DB, writeGOAData_DB, write-HomoloGeneData_DB, writeInParanoidData_DB, and writePeptideAtlasData_DB to respectively write data from Swiss-Prot, IPI, NCBI RefSeq database, and so on.
- writeName_DB parse mulitple data file, and write the mapping of id and name into SQLite-based R package. It employs writeGOName_DB, writeKEGGName_DB, writeP-FAMName_DB, writeINTERPROName_DB and writeTAXName_DB to respectively write data from GO, KEGG, Pfam, InterPro and TAX.
- createSeeds define a list of AnnDbBimap objects which indicates key and value of .
- createAnnObjs produce AnnDbBimap objects based on the definiation in create-Seeds.

3.3 Writing Help Documents

Help documents is an important part for new package. Diverse templates of help documents are stored in the "inst/templates" directory. When building new package, R functions use these templates to create "*.rd" help file in the "man" directory:

- getRepList return a list which will replace the symbols in template file.
- copyTemplates_DB implement similar function with copyTemplates, and is specially developped for SQLite-based annotation package.
- writeDescription_DB implement similar function with writeDescription, and is specially developped for SQLite-based annotation package.

3.4 Building Data Packages

Basic functions described above make it possible to build proteomic annotation data packages. Based on these, PAnnBuilder develops multiple sophisticated functions to assemble proteomic annotaion data. Each function is implemented by the "*Builder_DB" R function.

- pBaseBuilder_DB build annotation data packages for primary protein database such as SwissProt, TREMBL, IPI or NCBI RefSeq protein data.
- pSeqBuilder_DB build annotation data packages for query protein sequences based on sequence similarity.
- crossBuilder_DB build annotation data packages for protein id mapping in SwissProt, Trembl, IPI and NCBI Refseq databases.

- **subcellBuilder_DB** build annotation data packages for protein subcellular location from BaCelLo or DBSubLoc database.
- HomoloGeneBuilder_DB build annotation data packages for homolog protein group from NCBI HomoloGene database.
- InParanoidBuilder_DB build annotation data packages for ortholog protein group between two given organisms from InParanoid database.
- GOABuilder_DB build annotation data packages for mapping proteins of UniProt to Gene Ontolgy from GOA database.
- scopBuilder_DB build annotation data packages for Structural Classification of Proteins.
- intBuilder_DB build annotation data packages for protein-protein or domain-domain interaction from IntAct, MPPI, 3DID, DOMINE or NCBI Gene interaction database.
- PeptideAtlasBuilder_DB build annotation data packages for experimentally identified peptides from PeptideAtlas database.
- ptmBuilder_DB build annotation data packages for post-translational modifications from SysPTM database.
- **bfBuilder_DB** build annotation data packages for proteins in body fluids from Sys-BodyFluid database.
- dNameBuilder_DB build annotation data packages for mapping between entry ID and name from GO, KEGG, Pfam, InterPro and NCBI Taxonomy databases.

4 Building Annotation Data Packages

- 1. The first thing you need to do is setting basic parameters such as "pkgpath", "version", and "author".
 - > library(PAnnBuilder)
 > pkgPath <- tempdir()
 > version <- "1.0.0"
 > author <- list()
 > author[["authors"]] <- "Hong Li"
 > author[["maintainer"]] <- "Hong Li <sysptm@gmail.com>"
- 2. Then you can run diverse "*Builder_DB" functions to build packages by yourselves. pBaseBuilder_DB, subcellBuilder_DB, and pSeqBuilder_DB are taken as examples to build annotation packages.

pBaseBuilder_DB builds annotation data packages for proteins in three primary protein databases (SwissProt, IPI, RefSeq). It is a convenient way to obtain complete and canonical annotaion, including protein description, Entrez gene identifier, KEGG pathway, gene ontology, domain, coordinates on chromosomes and so on. For example, if you want to build annotation package for Mouse IPI database, you can use codes as follows:

```
> pBaseBuilder_DB(baseMapType = "ipi", organism = "Mus musculus",
```

```
+ prefix = "org.Mm.ipi", pkgPath = pkgPath, version = version,
```

+ author = author)

After running, a subdirectory called "org.Mm.ipi" will be produced in the path given by "pkgPath". This directory contains all data and files, which can be used to build R package by "R CMD build" command.

subcellBuilder_DB builds annotation data Package which provides protein subcellular location information.

>	<pre>subcellBuilder_DB(src</pre>	= "BaCelLo", pref	ix = "sc.bacello	", pkgPath,
+	version, author)			
>	dir(file.path(pkgPath,	"sc.bacello.db"))	
[1] "DESCRIPTION" "inst"	"man"	"NAMESPACE"	"R"

pSeqBuilder_DB uses blast to calculate sequence similarity between query proteins and subject proteins, then assign annotation for query protein according to existing annotation of its similar proteins. pSeqBuilder_DB is useful for proteins which have not well annotated. Following code chunk gives an example for annotation query proteins by pSeqBuilder_DB. Needed R packages *org.Hs.sp.db*, *org.Hs.ipi.db*, can be downloaded via biocLite.

```
> tmp = system.file("data", "query.example", package = "PAnnBuilder")
> tmp = readLines(tmp)
> tag = grep("^>", tmp)
> query <- sapply(1:(length(tag) - 1), function(x) {</pre>
+
      paste(tmp[(tag[x] + 1):(tag[x + 1] - 1)], collapse = "")
+ })
> query <- c(query, paste(tmp[(tag[length(tag)] + 1):length(tmp)],</pre>
      collapse = ""))
+
> names(query) = sub(">", "", tmp[tag])
> blast <- c("blastp", "10.0", "BLOSUM62", "0", "-1", "-1", "T",
      "F")
+
> names(blast) <- c("p", "e", "M", "W", "G", "E", "U", "F")
> match <- c(1e-05, 0.9, 0.9)
> names(match) <- c("e", "c", "i")</pre>
> if (!require("org.Hs.sp.db")) {
      biocLite("org.Hs.sp.db")
+
```

```
+ }
> if (!require("org.Hs.ipi.db")) {
+    biocLite("org.Hs.ipi.db")
+ }
> annPkgs = c("org.Hs.sp.db", "org.Hs.ipi.db")
> seqName = c("org.Hs.spSEQ", "org.Hs.ipiSEQ")
> pSeqBuilder_DB(query, annPkgs, seqName, blast, match, prefix = "test1",
+    pkgPath, version, author)
```

- 3. After the running of "*Builder_DB" function has been finished, a subdirectory named "pkgName" will be produced in given "pkgPath". Then the command "R CMD build" can be used to build source R package, and "R CMD –binary build" can be used to build binary R package for Windows.
- 4. Note:
 - Web connection is needed to download files from public databases, and Perl is needed to parse data files. Additionally, Rtools is needed for Windows user.
 - Users should be aware that downloading, parsing, and saving data may take a long time, in addition to requiring enough disk space to store temporary data files.
 - "R CMD build" and "R CMD -binary build" should be used in command line, not in R. Detailed document about how to create your own packages can be found in the book "Writing R Extensions" (http://cran.r-project.org/doc/manuals/R-exts.pdf). For Windows users, "R CMD build" needs you to have installed the files for building source packages (which is the default), as well as the Windows toolset (see the "R Installation and Administration" manual at http://cran.r-project.org/doc/manuals/R-admin.pdf).

5 Session Information

This vignette was generated using the following package versions:

```
R version 2.11.0 (2010-04-22)
x86_64-unknown-linux-gnu
locale:
[1] LC_CTYPE=en_US LC_NUMERIC=C LC_TIME=en_US
[4] LC_COLLATE=en_US LC_MONETARY=C LC_MESSAGES=en_US
[7] LC_PAPER=en_US LC_NAME=C LC_ADDRESS=C
[10] LC_TELEPHONE=C LC_MEASUREMENT=en_US LC_IDENTIFICATION=C
```

attached base packages:

[1] tools stats graphics grDevices utils datasets methods
[8] base
other attached packages:
[1] org.Hs.ipi.db_1.2.0 PAnnBuilder_1.12.0 AnnotationDbi_1.10.0
[4] Biobase_2.8.0 RSQLite_0.8-4 DBI_0.2-5

References

- O. N. Jensen. Interpreting the protein language using proteomics. *Nat Rev Mol Cell Biol*, 7 (6):391–403, 2006.
- R Development Core Team. R: A Language and Environment for Statistical Computing. R Foundation for Statistical Computing, Vienna, Austria, 2008. URL http: //www.R-project.org. ISBN 3-900051-07-0.
- J. Zhang, V. Carey, and R. Gentleman. An extensible application for assembling annotation for genomic data. *Bioinformatics*, 19(1):155–6, 2003.