USPTO PATENTS DATABASE CONSTRUCTION DOCUMENTATION

Graduate School of Library and Information Science, UIUC

08/15/2012

ACKNOWLEDGEMENT

The *Infrastructure and Planning for Informetric Analyses of Patents* project is funded and supported by Deere & Company. Throughout the process of finishing this project, two supervisors Prof. Vetle Torvik & Prof. Dave Dubin have offered invaluable guidance and advice, we also thank Derek D. Riddle for his help as a coordinator between Deere and us.

CONTACT INFORMATION

Vetle I. Torvik Assistant Professor
Graduate School of Library and Information Science, University of Illinois at Urbana-Champaign
501 E Daniel St, Room 221
Champaign, IL 61820
Email: vtorvik -at- illinois.edu
Site: http://people.lis.illinois.edu/~vtorvik/

Dave Dubin Research Associate Professor
Graduate School of Library and Information Science, University of Illinois at Urbana-Champaign
501 E. Daniel Street, Room 330
Champaign, IL 61820
Email: ddubin@illinois.edu
Site: http://people.lis.illinois.edu/~ddubin/

Qiyuan Liu (刘启元) M.S. Student
Graduate School of Library and Information Science, University of Illinois at Urbana-Champaign
Email: qliu14 -at- illinois.edu
Site: http://liuqiyuan.com

CONTENTS

INTRODUCTION

With the development of information processing technology, large amount of patent files granted by the governments are readily accessible nowadays. Exploring tools including databases and statistical algorithms like clustering are urgent to be implemented revealing the potential highly valuable information. The USPTO and Google make patent and trademark public data available to the public in bulk form so that the data can be used to load into databases or other analytical tools for research and analysis (USPTO, 2012).

This summer project aims to design and populate a database of patents granted by the USPTO from 1976 to now. This project obtains online patent bulk data automatically and parses the data by writing Python programs; then, all the processed data are populated into a well-designed MySQL database; special processing and database optimization is also finished with the testing of actual data. Two main types of patents, Patent Grants and Patent Application Publications, with five different formats including XML4.0-4.2, XML2.5, XML1.5-1.6, SGML 2.4, and APS are processed by two designed parsers Grants Parser and Publications Parser. In addition, Source Parser is created to parse the webpage to get downloadable links, PAIR (Patent Application Information Retrieval) Parser is designed to parse PAIR data crawled by Google from USPTO website, and Classifications Parser is produced to parse the Classification data. Automatic updating for new data is handled by an auto-updater application.

Up to August 15th 2012, 4,718,115 patent grants and 3,152,724 patent application publications have been successfully parsed and populated into the MySQL database. Detailed information about the dataset, processing methods, programs installation and execution, preliminary statistics of all the records hosted by the database, improvements and extensions, as well as special statements are also enclosed in this documentation.

DATASET

DATA STATISTICS

All the data are obtained directly from USPTO Bulk Downloads at Google Patent website[1]. By August 15th 2012, 858 Patent Grant Bibliographic Bulk Data packages with four formats (XML4.0-4.2, XML2.5, SGML 2.4, and APS), 596 Patent Publication Bibliographic Bulk Data packages with two formats (XML4.0-4.2 and XML1.5-1.6), 2,118,375 PAIR Data packages in an unified tab-delimited text format, and 1 U.S. Classification Text Attribute File have been downloaded and processed in this project.

Table 1. Statistics of the dataset

	Format	File Suffix	Time Range	# of Packages	Collection Size
Grants	XML4.0 - 4.2	.xml	JAN 2005 - DEC 2012	397	6G
	XML2.5	.xml	JAN 2001 - DEC 2004	157	
	SGML 2.4	.sgml	JAN 2001 - DEC 2001	52	
	APS	.dat or .txt	JAN 1976 - DEC 2001	252	
Publications	XML4.0 - 4.2	.xml	JAN 2005 - DEC 2012	397	12G
	XML1.5 - 1.6	.xml	MAR 2001 - DEC 2004	199	
PAIR	TSV ASCII	.txt		2,148,435	21T
Classification	TSV ASCII	.txt		1	
* *Appendix 1* Shows detailed information and examples about all the formats.					

FORMATS

For **patent grant** bibliographic data, there are four different formats totally: XML4.0-4.2, XML2.5, SGML 2.4, and APS. According to USPTO (2012), this data contains the bibliographic text of each patent grant issued weekly from January 1976 to present. The data set excludes images and/or drawings. The data set includes patent number, series code, application number, type of patent, filing date, title, issue date, inventor information, assignee name, foreign priority information, classification information, U.S. and foreign references, legal representative, Patent Cooperation Treaty (PCT) information, and abstract. Small differences also appear in one format with different versions. For example, data in 2001 have two formats: XML 2.5and SGML 2.4. Tiny differences between these two formats always throw errors during the parsing procedure. APS files are tab-delimited text files which list all the metadata in a certain sequence. There are two official documentations containing detailed information about the formats of XML[2] and APS[3].

[1] http://www.google.com/googlebooks/uspto-patents.html

[2] http://www.uspto.gov/products/PatentGrantXMLv42-Documentation.pdf

[3] http://storage.googleapis.com/patents/docs/PatentFullTextAPSDoc_GreenBook.pdf

For **patent application publication** data, their two formats XML 2.0-4.2 and XML 1.5-1.6 have a relative consistency. According to USPTO (2012), this data contains the bibliographic text of each patent application publication (non-provisional utility and plant) published weekly (excludes images/drawings) from 2001 to the current Calendar Year.

However, these xml files in both grants and application publications are not well-formed XML. Each of these files contains about 5,000 start/end tag combinations because of the concatenation of the individual documents, as well as one or more declaration lines. Data parsers created in this project need to reprocess these xml files and generate new well-formed xml content into the memory in order to facilitate subsequent processing.

For **PAIR** data, Google has begun crawling patent documents from the USPTO's public PAIR site (Google, 2012). Their crawl of new available documents operates continually and each of the files is packaged into a single zip file. This project accesses the PAIR data using 'http' although more than 2,000,000 patent application packages make the downloading and parsing very time-consuming. Address and Attorney/Agent, Application Data, Continuity Data, Foreign Priority, Image File Wrapper, Patent Term Adjustments / Extension History and Transaction History information are included in each of the file as far as they have. Overview of zip file contents is listed here[4]. This project also adopts The U.S. Classification Text Attribute File (CTAF) which contains data from the Manual of Classification excluding notes to obtain the U.S. patent classifications information (Google, 2012). Documentation about this classification file is downloaded here[5].

It's worth mentioning that not all of the files hosted by Google or USPTO website are correct without any doubt. For example, APS format files named "pba.*.zip" in the year of 1996, entirely different from other data in the same format, could not be parsed successfully because of their first huge blank lines. Due to those quality issues of the original dataset, 100% of publications packages and 92% of grants packages are parsed and loaded successfully into the database at the end.

SPECIAL PROCESSING

- XML Formats

All the xml files in the dataset don't have root/end tags but have one or more declaration rows in each file. This project adopts *ElementTree module* to parse the xml structure, so Python scripts add the root/end tag and delete all the declaration lines firstly, and then read preprocessed xml content into the Element Tree in order to finish the metadata extraction.

[4] http://www.google.com/googlebooks/uspto-patents-pair.html

[5] http://storage.googleapis.com/patents/patent_classification_information/USManualofClassDocumentation.doc

- Patent Numbers

Patent number formats have been enclosed in the aforementioned two documentations (Footnote 2 page 77 & Footnote 3 page 18). Patent numbers in the XML format dataset have 8 positions and numerical parts are always with a leading zero. However, patent numbers in the APS format dataset have 9 positions with a check digit in the end position. Because correct patent numbers should have 7 positions which corresponds to the cited IDs in the reference place of each patent, patent numbers confusing has been fixed in the parsers. Detailed information about the patent number formats and fixing examples are listed in the following *Table 2*.

Table 2. Patent Number Fixing Examples

Patent Type	XML			APS		
	Positions	**Original String**	**Fixed String**	**Positions**	**Original String**	**Fixed String**
Design Patents	8	D0345678	D345678	9	D03456789	D345678
SIR Patents	8	H0345678	H345678	9	HD0456789	HD45678
				9	HP0456789	HP45678
				9	H03456789	H345678
Plant Patents	8	PP045678	PP45678	9	PP0456789	PP45678
Reissue Patents	8	RE045678	RE45678	9	RE0456789	RE45678
Utility Patents	8	02345678	2345678	9	023456789	2345678
Defensive Publications	None			9	T03456789	T345678

- Classes and Subclasses

It's not interesting to see two different formats in the same field of the dataset. U.S. classification information in the XML file and APS file has two different formats: Main class and further classes. Class and subclass information are not separated in the original dataset, so this project tried the following strategy (as shown in *Table 3*) to divide U.S. classification information into class and subclass. More information can be found in the Footnote 2 page 85.

Table 3. U.S. Classification Identifying Examples

Patent Type	**Classification Type**	**Original String (9 or 10)**	**Class (3 Positions)**	**Subclass (6 Positions)**
Design Patents	Main class	D_2860	D02	860000
	Further class	D2_860	D02	860000
Plant Patents	Main class	PLT_ _ 345	PLT	003450
	Further class	PLT123	PLT	123000
All Other Patents	Main class	_ _2 _ _ 3	002	003000
	Further class	2_ _ _ _ 3	002	003000
* The flag '_' means a space in the string. Sometimes there will be other unexpected strings having 10 positions with letter 'R' or others and they need to be processed specially.				

- Date Time

In order to facilitate subsequent SQL queries, there are many Date data type in the database. As MySQL uses 'YYYY-MM-DD' format to store date value, all the Date data extracted from the original dataset are transformed into this format. Patent grants dataset and patent application publications dataset have almost the same format with 'YYYY-MM-DD', however, PAIR dataset has the format of 'MM-DD-YY' which needs to be preprocessed before populating into the database.

- Person Names

Person name disambiguation is a real problem facing in or after this project. Tables ASSIGNEE, EXAMINER, INVENTOR, and CITATION in this database have person's names information. We kept the intact look of person names from the original dataset. For example, 'Henry D. Anstey ' and 'Henry Dennis Anstey' is the same person who is one of the top 20 inventors in Deere & Company, but he has two or more different name strings in the database. Person name disambiguation is considered to be addressed later.

- Special Characters

As some characters have a special meaning in XML or HTML, many special characters have been replaced with an entity reference like '&..;' in order to realize parser interpretation. This project only encoded and decoded several special punctuation characters to allow *ElementTree* parsing the XML file. More information can be found in Footnote 2 page 86.

METHOD

DATABASE CONSTRUCTION

MySQL, which is regarded as the most popular open source database, is selected to host all the final data. A balance between data populating ease and high retrieval efficiency has been achieved in the process of constructing a database. Because of the format issues, the database is divided into three main parts. As shown in *Figure 1*, the left part, which is for PAIR data, contains the basic information, address, attorney and agent, continuity data, foreign priority, patent term adjustments, patent term extension history, and transaction history of patent applications; the main part, which contains the basic information, application information, examiners, agents, assignees, inventors, international classification, national classification, publication citations, grant citations, foreign patent citations and non-patent citations of both patent grants and patent application publications, is the core of this project; the right middle table USCLASSIFICATION, which contains all the U.S. classifications information used in the classification system by USPTO, is an independent part of the database.

Foreign key constraints have been set in order to keep data consistent and allow cross-reference related data across tables, and indexes have also been created to find rows with specific column values much more quickly.

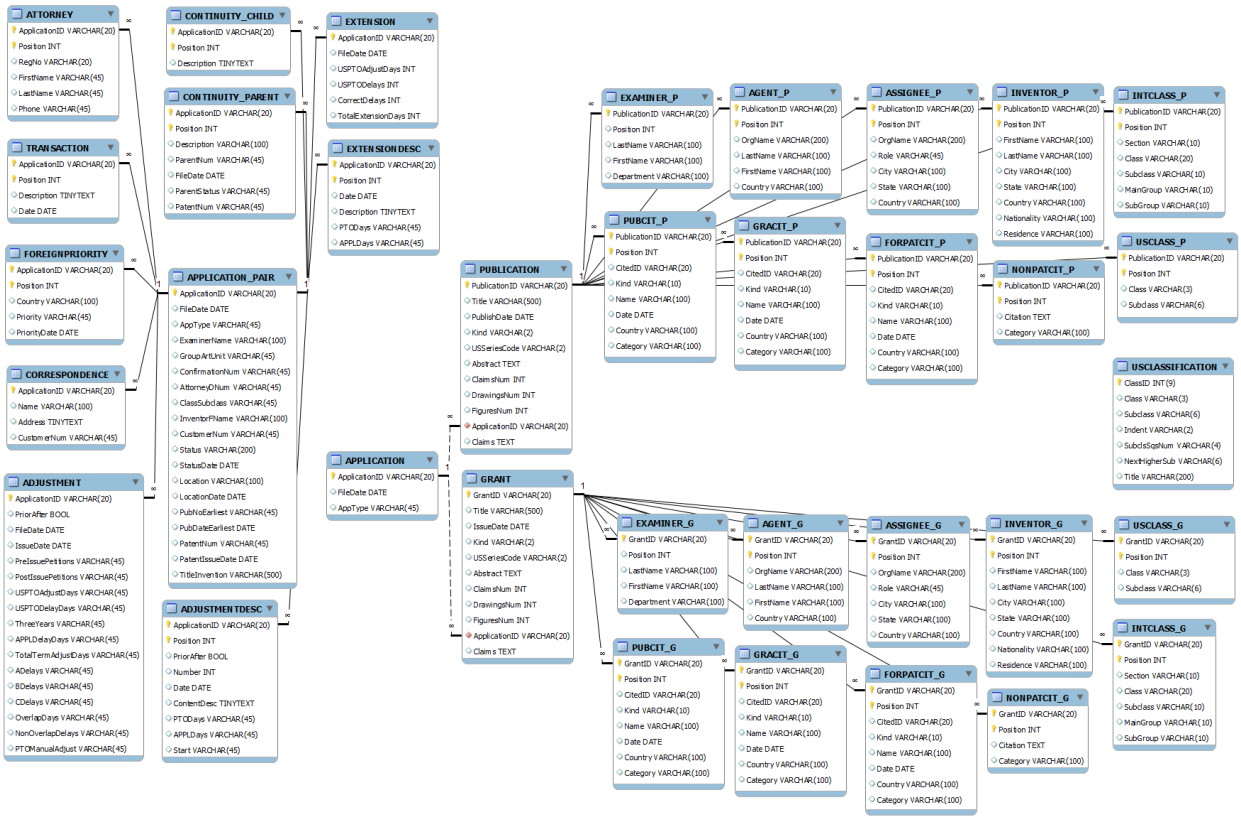


Figure 1. EER Diagram of the Database Schema

PROGRAMS DESIGN

Considering the portability and efficiency, the open source programing language Python has been chosen to build all the programs in this project. As shown in *Figure 2*, the Source Parser obtains all the links of accessible data from Google Patent website, then data are downloaded and their contained contents are processed as a list or string. These contents are transferred to specific data parsers including PAIR Parser, Grants Parser, Publications Parser, and Classifications Parser. Then, these four data parsers in the light gray box parse the contents into final tab-delimited format files (with the suffix '.csv') which are loaded into the database automatically. Note that data parsers may also delete the downloaded packages after generating the csv files in order to save space on the server. Log files (as seen in *Figure 3*) containing date time, file name, hyperlink of the package, data format, and processing status are generated at the same time. The auto-updater checks both the website and log files in a given period, and new packages are marked as unprocessed files which will be downloaded, parsed, and populated into the database automatically under the control of users in command line.

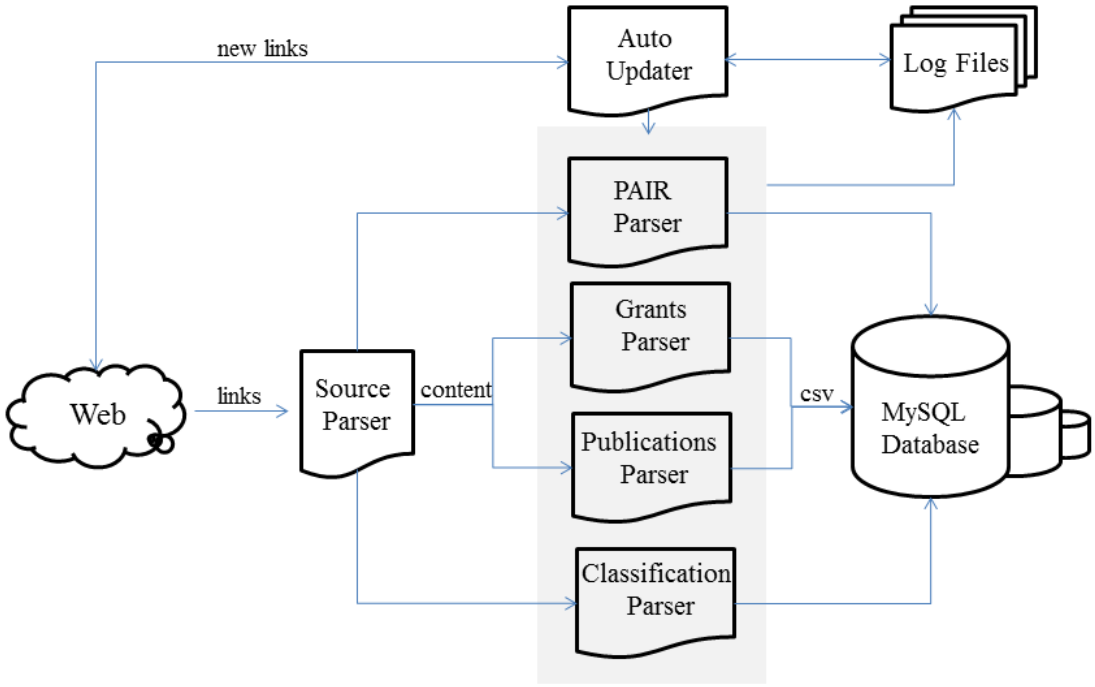


Figure 2. Flow chart of the programs execution procedure

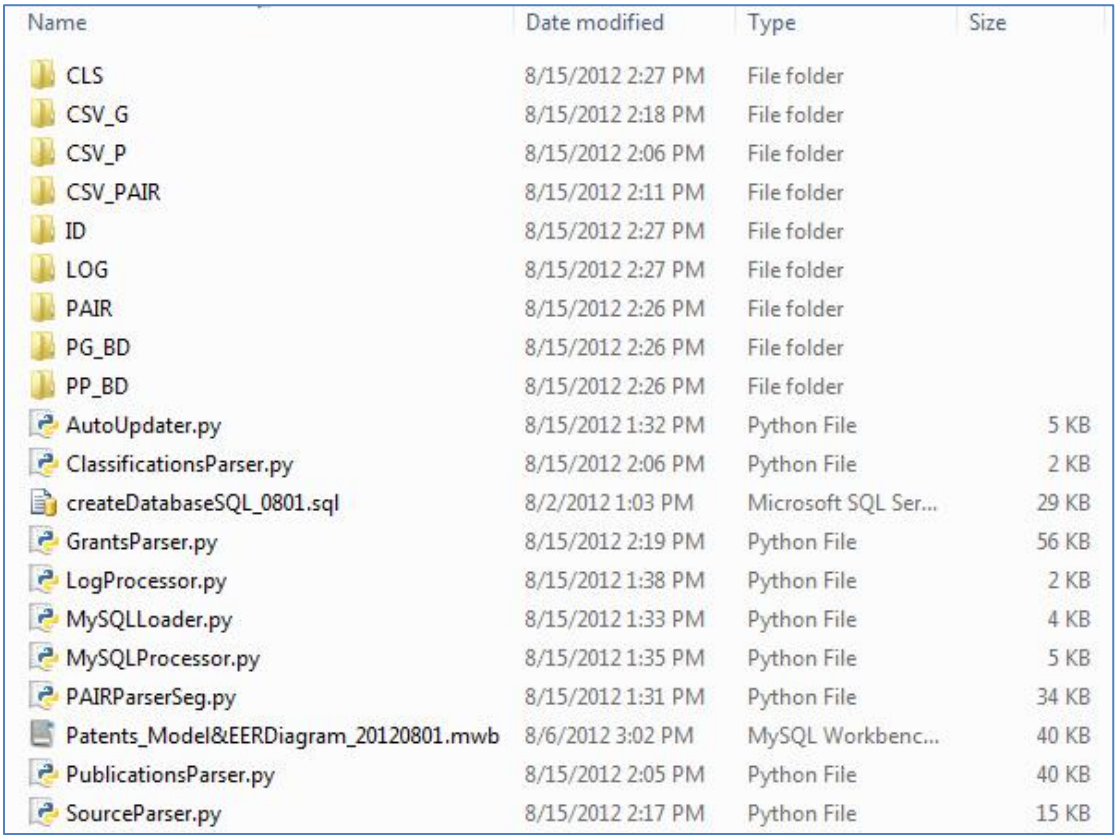
562	2012-08-07 08:27:33	ipgb20110524_wk21.zip	http://storage.googleapis.com/patents/grantbib/2011/ipgb20110524_wk21.zip	XML4	Processed
563	2012-08-07 08:29:09	ipgb20091110_wk45.zip	http://storage.googleapis.com/patents/grantbib/2009/ipgb20091110_wk45.zip	XML4	Processed
564	2012-08-07 08:35:40	ipgb20091117_wk46.zip	http://storage.googleapis.com/patents/grantbib/2009/ipgb20091117_wk46.zip	XML4	Processed
565	2012-08-07 08:37:13	ipgb20110531_wk22.zip	http://storage.googleapis.com/patents/grantbib/2011/ipgb20110531_wk22.zip	XML4	Processed
566	2012-08-07 08:42:09	ipgb20091124_wk47.zip	http://storage.googleapis.com/patents/grantbib/2009/ipgb20091124_wk47.zip	XML4	Processed
567	2012-08-07 08:47:00	ipgb20110607_wk23.zip	http://storage.googleapis.com/patents/grantbib/2011/ipgb20110607_wk23.zip	XML4	Processed
568	2012-08-07 08:48:32	ipgb20091201_wk48.zip	http://storage.googleapis.com/patents/grantbib/2009/ipgb20091201_wk48.zip	XML4	Processed
569	2012-08-07 08:52:13	pba19970610_wk23.zip	http://storage.googleapis.com/patents/grantbib/1997/pba19970610_wk23.zip	APS	Processed
570	2012-08-07 08:54:44	ipgb20091208_wk49.zip	http://storage.googleapis.com/patents/grantbib/2009/ipgb20091208_wk49.zip	XML4	Processed
571	2012-08-07 08:58:34	ipgb20110614_wk24.zip	http://storage.googleapis.com/patents/grantbib/2011/ipgb20110614_wk24.zip	XML4	Processed
572	2012-08-07 09:00:35	ipgb20091215_wk50.zip	http://storage.googleapis.com/patents/grantbib/2009/ipgb20091215_wk50.zip	XML4	Processed
573	2012-08-07 09:04:08	ipgb20091222_wk51.zip	http://storage.googleapis.com/patents/grantbib/2009/ipgb20091222_wk51.zip	XML4	Processed
574	2012-08-07 09:06:39	ipgb20110621_wk25.zip	http://storage.googleapis.com/patents/grantbib/2011/ipgb20110621_wk25.zip	XML4	Processed
575	2012-08-07 09:10:45	ipgb20091229_wk52.zip	http://storage.googleapis.com/patents/grantbib/2009/ipgb20091229_wk52.zip	XML4	Processed
576	2012-08-07 09:13:50	ipgb20080101_wk01.zip	http://storage.googleapis.com/patents/grantbib/2008/ipgb20080101_wk01.zip	XML4	Processed
577	2012-08-07 09:17:05	ipgb20080108_wk02.zip	http://storage.googleapis.com/patents/grantbib/2008/ipgb20080108_wk02.zip	XML4	Processed
578	2012-08-07 09:17:28	ipgb20110628_wk26.zip	http://storage.googleapis.com/patents/grantbib/2011/ipgb20110628_wk26.zip	XML4	Processed
579	2012-08-07 09:19:44	ipgb20080115_wk03.zip	http://storage.googleapis.com/patents/grantbib/2008/ipgb20080115_wk03.zip	XML4	Processed
580	2012-08-07 09:22:19	ipgb20080122_wk04.zip	http://storage.googleapis.com/patents/grantbib/2008/ipgb20080122_wk04.zip	XML4	Processed
581	2012-08-07 09:25:41	ipgb20110705_wk27.zip	http://storage.googleapis.com/patents/grantbib/2011/ipgb20110705_wk27.zip	XML4	Processed
582	2012-08-07 09:26:42	ipgb20080129_wk05.zip	http://storage.googleapis.com/patents/grantbib/2008/ipgb20080129_wk05.zip	XML4	Processed
583	2012-08-07 09:31:33	ipgb20080205_wk06.zip	http://storage.googleapis.com/patents/grantbib/2008/ipgb20080205_wk06.zip	XML4	Processed
584	2012-08-07 09:35:17	ipgb20080212_wk07.zip	http://storage.googleapis.com/patents/grantbib/2008/ipgb20080212_wk07.zip	XML4	Processed
585	2012-08-07 09:36:52	ipgb20110712_wk28.zip	http://storage.googleapis.com/patents/grantbib/2011/ipgb20110712_wk28.zip	XML4	Processed
586	2012-08-07 09:40:11	ipgb20080219_wk08.zip	http://storage.googleapis.com/patents/grantbib/2008/ipgb20080219_wk08.zip	XML4	Processed
587	2012-08-07 09:45:04	ipgb20080226_wk09.zip	http://storage.googleapis.com/patents/grantbib/2008/ipgb20080226_wk09.zip	XML4	Processed
588	2012-08-07 09:47:31	ipgb20110719_wk29.zip	http://storage.googleapis.com/patents/grantbib/2011/ipgb20110719_wk29.zip	XML4	Processed
589	2012-08-07 09:49:48	ipgb20080304_wk10.zip	http://storage.googleapis.com/patents/grantbib/2008/ipgb20080304_wk10.zip	XML4	Processed
590	2012-08-07 09:53:16	ipgb20110726_wk30.zip	http://storage.googleapis.com/patents/grantbib/2011/ipgb20110726_wk30.zip	XML4	Processed
591	2012-08-07 09:53:49	ipgb20080311_wk11.zip	http://storage.googleapis.com/patents/grantbib/2008/ipgb20080311_wk11.zip	XML4	Processed
592	2012-08-07 09:59:19	ipgb20080318_wk12.zip	http://storage.googleapis.com/patents/grantbib/2008/ipgb20080318_wk12.zip	XML4	Processed
593	2012-08-07 10:00:11	pba19970617_wk24.zip	http://storage.googleapis.com/patents/grantbib/1997/pba19970617_wk24.zip	APS	Processed
594	2012-08-07 10:04:24	ipgb20080325_wk13.zip	http://storage.googleapis.com/patents/grantbib/2008/ipgb20080325_wk13.zip	XML4	Processed

Figure 3. Log file sample of patent grants

INSTALLATION

As this project aims to provide researchers with open source Python scripts and database schemas, all the files (as shown in *Figure 4*) are downloadable and can be configured and installed easily. Five parsers including the Source Parser crawling downloadable hyperlinks from the website, Grants Parser paring and loading Patent Grant data, Publications Parser paring and loading Patent Application Publication data, PAIR Parser parsing and loading Patent Application Information Retrieval data, and Classification Parser populating the U.S. classification information into the database, are written to deal with the messy formats of the patents collection. A data auto-updater application is also created for checking auto-updates.

MySQLdb module (SOURCEFORGE, 2012), which is an thread-compatible interface to MySQL database server that provides the Python database API, has been used to load data into the database. In addition, we adopt multiprocessing module (Python Software Foundation, 2012) which allows the programmer to fully leverage multiple processors on the server to realize parallel execution and saving time. Instructions for the execution of all the programs are as following.



Name	Date modified	Type	Size
CLS	8/15/2012 2:27 PM	File folder	
CSV_G	8/15/2012 2:18 PM	File folder	
CSV_P	8/15/2012 2:06 PM	File folder	
CSV_PAIR	8/15/2012 2:11 PM	File folder	
ID	8/15/2012 2:27 PM	File folder	
LOG	8/15/2012 2:27 PM	File folder	
PAIR	8/15/2012 2:26 PM	File folder	
PG_BD	8/15/2012 2:26 PM	File folder	
PP_BD	8/15/2012 2:26 PM	File folder	
AutoUpdater.py	8/15/2012 1:32 PM	Python File	5 KB
ClassificationsParser.py	8/15/2012 2:06 PM	Python File	2 KB
createDatabaseSQL_0801.sql	8/2/2012 1:03 PM	Microsoft SQL Ser...	29 KB
GrantsParser.py	8/15/2012 2:19 PM	Python File	56 KB
LogProcessor.py	8/15/2012 1:38 PM	Python File	2 KB
MySQLLoader.py	8/15/2012 1:33 PM	Python File	4 KB
MySQLProcessor.py	8/15/2012 1:35 PM	Python File	5 KB
PAIRParserSeg.py	8/15/2012 1:31 PM	Python File	34 KB
Patents_Model&EERDiagram_20120801.mwb	8/6/2012 3:02 PM	MySQL Workbenc...	40 KB
PublicationsParser.py	8/15/2012 2:05 PM	Python File	40 KB
SourceParser.py	8/15/2012 2:17 PM	Python File	15 KB

Figure 4. Directory View of the Whole Project

CONFIGURATION

We highly recommend the Linux server to build the database and run all the following scripts in order to download, parse and populate all the data. Configuration information is listed below:

- OS: Linux/Unix, Windows, or Mac OS X
- Python Version: 2.6 download
- External Module: MySQLdb download windows
- Database: MySQL 5.0 or upper download windows

PROJECT FILES INVENTORY

All the project files are listed as in *Table 4*.

Table 4. Project files list & descriptions

Folder/File Path		Description
/CLS		Folder for Classifications
	/CLS/ctaf1204.txt	U.S. Manual of Classification File (CTAF, Classification Text Attribute File)
/CSV_G		CSV Folder for Grants Data
	/CSV_G/.*.csv ...	Any .csv files generated in the whole process
/CSV_P		CSV Folder for Publications Data
	/CSV_P/.*.csv ...	Any .csv files generated in the whole process
/CSV_PAIR		CSV Folder for PAIR Data
	/CSV_PAIR/.*.csv ...	Any .csv files generated in the whole process
/ID		Folder for Intermediate Data
	/ID/PAIRLinks	PAIR data links ranges file
/LOG		Log Folder
	/LOG/LOG_G	Log File of Grants
	/LOG/LOG_P	Log File of Publications
	/LOG/LOG_PAIR	Log File of PAIR Data
	/LOG/LOG_PAIR_ERROR	Log File of PAIR Data Errors
/PAIR		Original Data Folder for PAIR
	/PAIR/.*.zip ...	Any .zip files of PAIR downloaded in the whole process
/PG_BD		Original Data Folder for Grants
	/PG_BD/.*.zip ...	Any .zip files of Grants downloaded in the whole process
/PP_BD		Original Data Folder for Publications
	PP_BD/.*.zip ...	Any .zip files of Publications downloaded in the whole process
/AutoUpdater.py		Automatic updating application
/ClassificationParser.py		Data parser for classification files
/GrantsParser.py		Data parser for grant files
/LogProcessor.py		Log files processing class

/MySQLLoader.py	MySQL loading application
/MySQLProcessor.py	MySQL processing class
/PAIRParser.py	Data parser for PAIR files
/PublicationsParser.py	Data parser for publication files
/SourceParser.py	Parser for webpages
/createDatabaseSQL_0801.sql	SQL scripts for Database creation
/Patents_Model&EERDiagram_20120804.mwb	Database schema file

PROGRAMS EXECUTION

1. Build the database

You need to confirm that you have installed MySQL 5.0 or upper in your server or current os. There are two ways to create your database.

- Download the database schema file *'/Patents_Model&EERDiagram_20120804.mwb'* and open it in the MySQL workbench, then use menu [Database]-[Forward Engineer] to create the database.

OR

- Download the SQL scripts file *'/createDatabaseSQL_0801.sql'* into a specific *filePath* and run it using the following command in your MySQL command line: "mysql > SOURCE *filePath"*

2. Run Python scripts

Your need to confirm that you have installed Python 2.6 and external Python module *MySQLdb* in your server or current os. There are several un-sequenced steps to run the scripts:

- Download the compressed project file *'**USPTOPatentsDatabaseConstrucation.zip**'* and uncompress it. Please keep the original location of all the folders (CLS, CSV_G,CSV_P,CSV_PAIR,ID, CSV_LOG, PAIR, PG_BD, PP_BD) with its contained files you have downloaded.

- Run ***GrantsParser.py*** to download, parse, and populate Patent Grants data into the database automatically. The parser gets all the patent grants downloadable hyperlinks through *SourceParser.py* which checks Google website. All the data will be downloaded firstly, and then their formats will be identified by their file names. This parser uses different format functions to obtain all the contained data in these packages. Then the parser extracts all the metadata and populates them into the database in a certain sequence.

- Run ***PublicationsParser.py*** to download, parse, and populate Patent Application Publications data into the database automatically. The patent application publications parser uses the same processing

strategy as *GrantsParser.py* which downloads zip packages from Google website and then parses them into the database.

- Run ***ClassificationParser.py*** to parse and populate Patent Classifications data stored in the folder of '/CLS' into the database automatically.

- Run ***PAIRParserSeg.py*** to download, parse and populate Patent Application Information Retrieval (PAIR) data into the database automatically. There are terabytes of PAIR data hosted in Google, so the consuming time depends on your network speed and server hardware. *PAIRParserSeg.py* gets all the PAIR data downloadable hyperlinks firstly, and then divides them into many segments. The parser creates ten processes processing 1000 packages at one time due to the network speed, and all the packages are deleted after the extraction and loading of the data in order to save space for your server.

- ***AutoUpdater.py*** is designed to check new updates and populate them into the database automatically. The parser obtains new unprocessed hyperlinks by comparing all the downloadable hyperlinks on Google website and the list of files processed successfully in the LOG file. Then new hyperlinks are transferred and processed by appropriate parsers. We highly recommend running this automatic updater per week because the USPTO data are updated once a week on Google (always on Tuesday).

OR

- Loading all the .csv files

We also provide .csv files to be loaded into the database much more easily. What you only need to do is download all the .csv files and run *'**MySQLLoader.py**'* to populate all the data into your database. Note that, after populating all the .csv files into the database, you need to download the existed log file (LOG_G, LOG_P, LOG_PAIR) in order to keep the accuracy of the automatic updating.

PRELIMINARY RESULTS

In basis of the current data collection up to August 15th 2012, *Table 5* shows the statistics of the database. 4,718,115 patent grant records from 1976 to now, 3,147,014 patent application publication records from 2001 to now, and 412,685patent application information retrieval records have been populated into the database.

Table 5. Statistics of the records hosted in the database

Patent	Table Name	# of Records	Description
Application	APPLICATION	4,744,735	Applications Basic Information
Grants	GRANT	4,718,115	Patent Grants Basic Information
	EXAMINER_G	4,708,191	Examiners of Grants
	AGENT_G	5,241,532	Agents of Grants
	ASSIGNEE_G	4,082,426	Assignees of Grants
	INVENTOR_G	10,485,014	Inventors of Grants
	USCLASS_G	17,825,566	U.S. Classes of Grants
	INTCLASS_G	5,829,071	International Classes of Grants
	GRACIT_G	57,307,092	Patent Grant Citations
	PUBCIT_G	6,632,208	Patent Application Publication Citations
	FORPATCIT_G	11,262,440	Foreign Patent Citations
	NONPATCIT_G	15,432,009	Non-patent Citations
Publications	PUBLICATION	3,152,724	Patent Application Publications Basic Info.
	EXAMINER_P	0	Examiners of Publications
	AGENT_P	760,729	Agents of Publications
	ASSIGNEE_P	1,394,315	Assignees of Publications
	INVENTOR_P	8,221,821	Inventors of Publications
	USCLASS_P	6,456,971	U.S. Classes of Publications
	INTCLASS_P	4,997,320	International Classes of Publications
	GRACIT_P	0	Patent Grant Citations
	PUBCIT_P	0	Patent Application Publication Citations
	FORPATCIT_P	0	Foreign Patent Citations
	NONPATCIT_P	0	Non-patent Citations
Classification	USCLASSIFICATION	168,838	U.S. Classification Information
PAIR	APPLICATION_PAIR	412,685	Application Data
	ATTORNEY	7,718,932	Attorney Data
	TRANSACTION	14,063,809	Transaction History Data
	FOREIGNPRIORITY	211,617	Foreign Priority Data
	CORRESPONDENCE	412,638	Correspondence Data
	ADJUSTMENT	218,248	Patent Term Adjustment Data
	ADJUSTMENTDESC	8,733,864	Patent Term Adjustment Descriptions
	CONTINUITY_CHILD	243,517	Child Continuity Data
	CONTINUITY_PARENT	358,481	Parent Continuity Data
	EXTENSION	9,676	Patent Term Extension History Data
	EXTENSIONDESC	273,731	Patent Term Extension History Descriptions

In addition, this documentation gives two charts (as shown in *Figure 5* and *Figure 6*) showing the frequency distribution of both patent grant records and patent application publication records in different years.

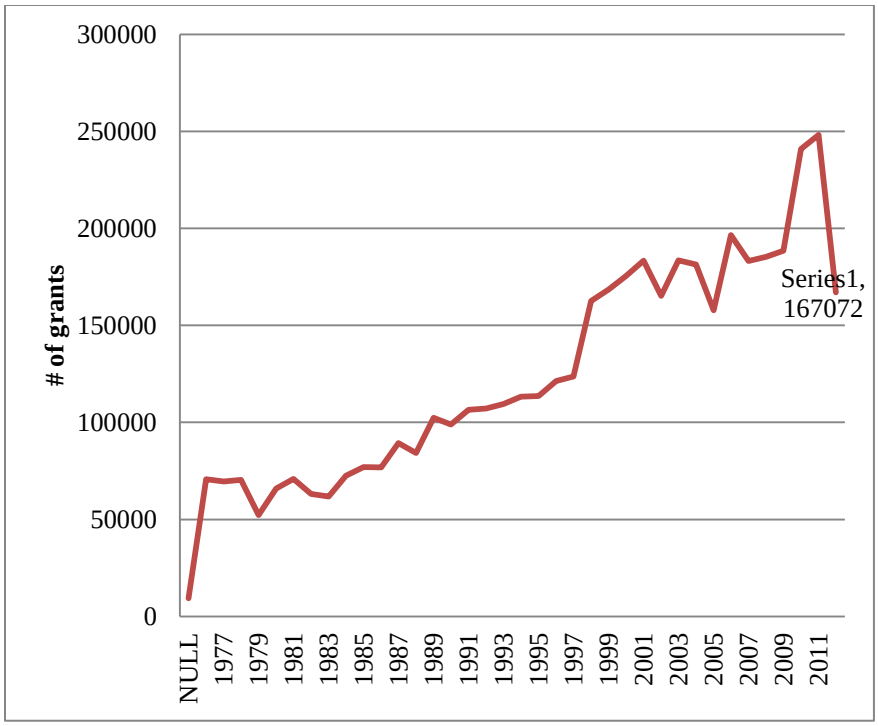


Figure 5. Distribution of Patent Grants Frequency

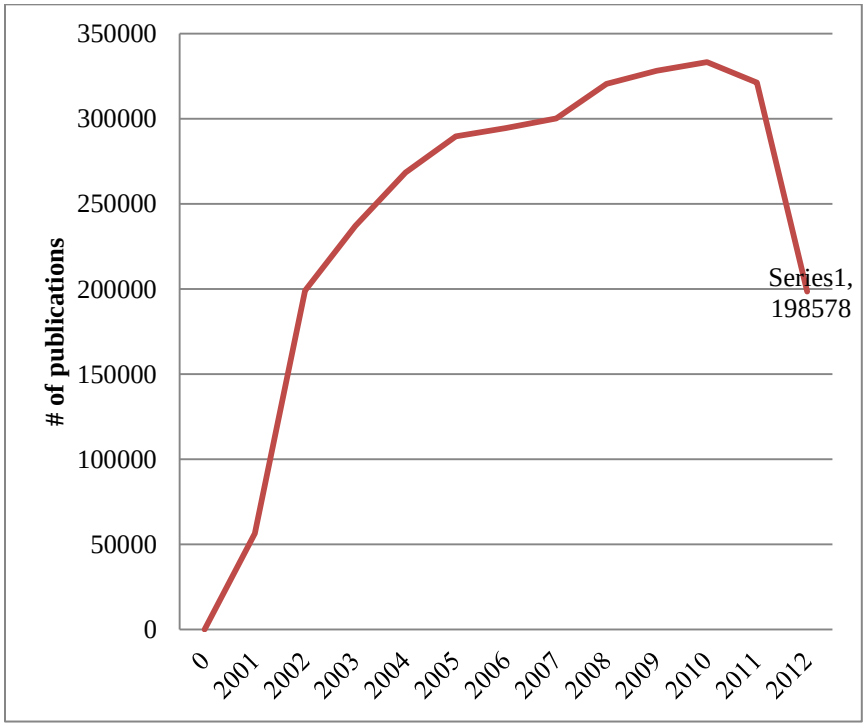


Figure 6. Distribution of Patent Application Publications Frequency

REFERENCES

Google. (2012, August 8th). USPTO Bulk Downloads: PAIR Data. Retrieved from http://www.google.com/googlebooks/uspto-patents-pair.html

Google. (2012, August 8th). USPTO Bulk Downloads: Patent Classification Information .Retrieved from http://www.google.com/googlebooks/uspto-patents-class.html

Python Software Foundation. (2012, July 31th). 16.6. multiprocessing — Process-based "threading" interface. Retrieved from http://docs.python.org/library/multiprocessing

SOURCEFORGE. (2012, July 1st). MySQL for Python. Retrieved from http://sourceforge.net/projects/mysql-python/

USPTO. (2012, August 1st). Electronic Data Products. Retrieved from http://www.uspto.gov/products/catalog/index.jsp

USPTO. (2012, August 8th). Patent Application Publication Data Products. Retrieved from http://www.uspto.gov/products/catalog/patent_applications.jsp

USPTO. (2012, August 8th). Patent Grant Data Products. Retrieved from http://www.uspto.gov/products/catalog/patent_grants.jsp

USPTO. (2012, July 31th). XML Resources. Retrieved from http://www.uspto.gov/products/xml-resources.jsp

APPENDIX 1. FORMATS OF THE USPTO PATENTS BULK DATA

<table>
<tr><th>Patents Type</th><th>Format</th><th>Example</th></tr>
<tr><td rowspan="2">Grants</td><td>XML4.0 - 4.2</td><td><?xml version="1.0" encoding="UTF-8"?>
− <us-patent-grant lang="EN" dtd-version="v4.2 2006-08-23" file="USD0651786-20120110.XML" status="PRODUCTION" id="us-patent-grant" country="US" date-produced="20111227" date-publ="20120110" >
− <us-bibliographic-data-grant>
+ <publication-reference>
</publication-reference>
+ <application-reference appl-type="design" >
</application-reference>
<us-application-series-code>29</us-application-series-code>
+ <us-term-of-grant>
</us-term-of-grant>
+ <classification-locarno>
</classification-locarno>
+ <classification-national>
</classification-national>
<invention-title id="d2e53" >Burp cloth</invention-title>
+ <references-cited>
</references-cited>
<number-of-claims>1</number-of-claims>
<us-exemplary-claim>1</us-exemplary-claim>
+ <us-field-of-classification-search>
</us-field-of-classification-search>
+ <figures>
</figures>
+ <parties>
</parties>
+ <examiners>
</examiners>
</us-bibliographic-data-grant>
</us-patent-grant></td></tr>
<tr><td>XML2.5</td><td><?xml version="1.0" encoding="UTF-8"?>
− <PATDOC DTD="2.5" STATUS="Build 20030724" >
− <SDOBI>
+ <B100>
</B100>
+ <B200>
</B200>
+ <B300>
</B300>
+ <B300>
</B300>
+ <B300>
</B300>
+ <B300>
</B300>
+ <B300>
</B300>
+ <B300>
</B300>
+ <B400>
</B400>
+ <B500>
</B500>
+ <B600>
</B600>
+ <B700>
</B700>
</SDOBI>
+ <SDODE></td></tr>
</table>

		</SDODE> <SDOCL> </SDOCL> <SDODR ID="DRAWINGS" > </SDODR> </PATDOC>
	SGML 2.4	<?xml version="1.0" encoding="UTF-8"?> <PATDOC DTD="2.5" STATUS="Build 20020101" > <SDOBI> <B100> </B100> <B200> </B200> <B400> </B400> <B500> </B500> <B700> </B700> </SDOBI> <SDODE> </SDODE> <SDOCL> </SDOCL> <SDODR ID="DRAWINGS" > </SDODR> </PATDOC>
	APS	HHHHHT APS1 ISSUE - 990413 PATN WKU D04078853 SRC D APN 0882178 APT 4 ART 291 APD 19980518 TTL Letter-shaped pasta ISD 19990413 NCL 1 ECL 1 EXP Lucas; Susan J. NDR 1 NFG 6 TRM 14 INVT NAM Lupini; Paolo CTY San Paolo di Jesi CNT ITX ASSG NAM Datamarche S.A.S. Di Georgia Spaccapietra & C. CNT ITX COD 03 CLAS OCL D 1114 EDF 6 ICL 0101 FSC D 1 FSS 106;113;114;128;127 FSC 426 FSS 104 FSC D18 FSS 28 UREF PNO D297280 ISD 19880800 NAM Lucas et al.

<table>
<tr><td></td><td></td><td>XCL D 1106
UREF
PNO D376465
ISD 19961200
NAM Haro et al.
OCL D 1114
UREF
PNO 3081569
ISD 19630300
NAM Ownbey
XCL D18 28
LREP
FR2 Bloom; Leonard
DRWD
PAL FIG. 1 is a perspective view of letter-shaped pasta showing my new design.
PAL FIG. 2 is a top plan view thereof, the bottom plan view not shown and being a mirror image thereof.
PAL FIG. 3 is a front elevational view thereof.
PAL FIG. 4 is a rear elevational view thereof.
PAL FIG. 5 is a right side elevational view thereof; and,
PAL FIG. 6 is a left side elevational view thereof.
DCLM
PAL The ornamental design for letter-shaped pasta, as shown and described.</td></tr>
<tr><td rowspan="2">Publications</td><td>XML4.0 - 4.2</td><td><?xml version="1.0" encoding="UTF-8"?>
− <us-patent-application lang="EN" dtd-version="v4.1 2005-08-25" file="US20070011794A1-20070118.XML" status="PRODUCTION" id="us-patent-application" country="US" date-produced="20070103" date-publ="20070118" >
− <us-bibliographic-data-application lang="EN" country="US" >
+ <publication-reference>
</publication-reference>
+ <application-reference appl-type="utility" >
</application-reference>
<us-application-series-code>11</us-application-series-code>
+ <classifications-ipcr>
</classifications-ipcr>
+ <classification-national>
</classification-national>
<invention-title id="d0e102" >Assembled safety cap</invention-title>
+ <parties>
</parties>
</us-bibliographic-data-application>
+ <abstract id="abstract" >
</abstract>
</us-patent-application></td></tr>
<tr><td>XML1.5 - 1.6</td><td><?xml version="1.0" encoding="UTF-8"?>
− <patent-application-publication>
− <subdoc-bibliographic-information>
+ <document-id>
</document-id>
<publication-filing-type>new</publication-filing-type>
+ <domestic-filing-data>
</domestic-filing-data>
+ <technical-information>
</technical-information>
+ <continuity-data>
</continuity-data>
+ <inventors>
</inventors>
+ <correspondence-address>
</correspondence-address>
</subdoc-bibliographic-information>
+ <subdoc-abstract>
</subdoc-abstract>
</patent-application-publication></td></tr>
</table>

PAIR	TSV ASCII	Application Number 12/690,333 Filing or 371 (c) Date 01-20-2010 Application Type Utility Examiner Name HAYES, KRISTEN C Group Art Unit 3643 Confirmation Number 7625 Attorney Docket Number 0355.03 Class / Subclass 047/079 First Named Inventor Wen Hua He , Zhongshan, (CN) Customer Number - Status Patented Case Status Date 10-26-2011 Location ELECTRONIC Location Date - Earliest Publication No US 2011-0173884 A1 Earliest Publication Date 07-21-2011 Patent Number 8,056,284 Issue Date of Patent 11-15-2011 Title of Invention AUTO-IRRIGATING CASE
Classification	TSV ASCII	PLTFOR101 3 416FOR101New Guinea (PLT/318) PLTFOR102 2 417FOR102Petunia (PLT/356) 002000000 0 1 APPAREL 002001000 1 2000000MISCELLANEOUS 002455000 1 3000000GUARD OR PROTECTOR 002456000 2 4455000Body cover 002457000 3 5456000Hazardous material body cover 002458000 3 6456000Thermal body cover 002002110 3 7456000Astronauts body cover
* There are always some tiny differences among the same formats with different versions.		