Appendix D

Databases Used by CiteSeerX

There are two major databases used in the CiteSeerX project: *citeseerx* and *csx_ citegraph*.

D.1 The *citeseerx* database

In this section, we describe all the tables within the *citeseerx* database. This database stores all information about papers metadata.

1. *acknowledgmentContexts*. Contexts in which an acknowledgements was found inside a paper (Table D.4).
2. *acknowledgments*. Acknowledgements found on a paper (Table D.3).
3. *acknowledgments_ versionShadow*. Stores information indicating where the last version of the acknowledgement metadata came from (Header Parser, Inference, User Corrections). Each field stores the name of the source for the same field in the *acknowledgement* table.
4. *authors*. Information about each one of the authors for a given paper (Table D.2).
5. *authorsversionShadow*. Stores information indicating where the last version of the author metadata came from (Header Parser, Inference, User Correction). Each field stores the same of the source for the same field in the authors table.
6. *cannames*. Cannonical and disambiguated author names information (Table D.21).
7. *checksum*. Checksum data for each file type associated to a paper. A paper can have different file type representations like PDF, PS and so on. The same paper could also have different versions in a same file type; let's say the PDF

pre-print and camera ready versions of the paper (Table D.9).

8. *citationContexts*. Contexts in which a citation was found inside a paper (Table D.6).
9. *citations*. Citations found on a paper (Table D.5).
10. *citecharts*. Not documented.
11. *eTables*. Not documented.
12. *elinks*.External links for papers based on information from the paper, link_types, and and external metadata database (Table D.19).
13. *hubMap*. Associates URLs (pointing to a paper) with its page container if any (Table D.14).
14. *hubUrls*. URLs pointing to web pages which contains links to PDF, PS files (Table D.13).
15. *keywords*. Keywords found in the paper (Table D.7).
16. *keywords_versionShadow*. Checksum data for each file type associated to a paper. A paper can have different file type representations, like PDF, PS and so on. The same paper could also have different versions in a same file type.
17. *legacyIDMap*. Mappings between citeseer and citeseerx papers (CiteSeerX table only) (Table D.15).
18. *link_types*. Identifies external sources that can be linked for each paper (Table D.18).
19. *paperVersions*. Information about each one of the metadata corrections for a given paper (Table D.10).
20. *paper_keywords_noun*. Keyphrases extracted using Hung-Hsuan's NP Chunking Extractor (EKE [1, 3]) (Table D.8).
21. *paper_listing*. Papers logically removed from CiteSeerX website, i.e., they are not displayed and downloadable.
22. *papers*. Stores metadata and access information for each paper (Table D.1).
23. *papers_versionShadow*. Stores information indicating where the last version of the paper metadata came from (Header Parser, Inference, User Corrections). Each field stores the name of the source for the same field in the papers table.
24. *pdfTables*. Metadata and data from each one of the tables found in a paper (Table D.20).
25. *redirectpdf*. Not documented.

26. *redirecttemplates*. Not documented).
27. *tags*. Tags associated to a paper (Table D.17).
28. *textSources*. Used to setup a text source (right now only the banner; Table D.16).
29. *urls*. URL where a version of the paper was found. A paper can be found at different URLs. For example, a URL pointing to a PDF version of the paper, another one pointing to the PS version within the same web server, and/or different URLs pointing to the same PDF version but in different web servers one for each one of the paper's authors (Table D.12).
30. *userCorrections*. Information about a metadata correction made by a user (Table D.11).

Table D.1: Fields in the *papers* table. .

Field	Description
id	Unique id identifying each paper. System assigned using the DOI server.
version	Last valid version of paper metadata.
cluster	Identifier of the collection of similar papers/citations. Default value is 0, after the inference process is run.
public	Indicates if the paper is available or not
ncites	Number of citations found within the corpus for the given paper. (It is calculated by the inference process).
versionName	Name of the last valid version of paper metadata
crawlDate	When the paper was obtained
repositoryID	Repository identifier where all the files associated to the paper are located. The Repository identifier maps to a physical location in a file system.
ConversionTrace	All the tools (in order) used to get the text version of the paper.
selfCites	Number of self citations (This data is calculated by the inference process)
versionTime	When the last version of the paper metadata was created/updated

Table D.2: Fields in the *authors* table.

Field	Description
id	Unique id identifying for each author. System assigned auto increment.
cluster	The ID of the cluster for a disambiguated author. The identifier is assigned by the name disambiguation process.
name	Author names as extracted/corrected from the paper
afill	Affiliation as extracted/corrected from the paper
address	Author's addresses as extracted/corrected from the paper
email	Author's email addresses as extracted/corrected from the paper
ord	Location of the author in the list of author for the given paper

Table D.3: Fields in the *acknowledgments* table.

Field	Description
id	Unique id identifying for each acknowledgement. System assigned auto increment.
cluster	Identifier for similar/same acknowledgement
name	Entity mention (surface name)
ackType	Type of acknowledgement
paperid	The csxdoi of the paper where the acknowledgement is extracted from

Table D.4: Fields in the *acknowledgementContexts* table.

Field	Description
id	Unique id identifying for each acknowledgement. System assigned auto increment.
context	Raw text where the acknowledgement was found

Table D.5: Fields in the *citations* table.

Field	Description
id	Unique id identifying for each citation. System assigned auto increment.
cluster	Unique identifier that match the same citation in different papers to a canonical one; including the paper if present within the corpus. The id is assigned by the inference process.
raw	Citation text as found on the paper
self	Indicates if this citation is a self citation

Table D.6: Fields in the *citationContexts* table.

Field	Description
id	Unique id identifying for each citation context. System assigned auto increment.
context	Raw text where the citation is mentioned on the paper

Table D.7: Fields in the *keywords* table.

Field	Description
id	Unique id identifying for each keyword. System assigned auto increment.
keyword	The keyword

Table D.8: Fields in the *paper_keywords_noun* table.

Field	Description
id	Unique id identifying for each keyphrase. System assigned auto increment.
paper_id	The csxdoi of the paper from whose abstract this keyphrase is extracted from.
ngram	The keyphrase extracted using NP Chunking
count	The number of occurances the keyphrase appears.

Table D.9: Fields in the *checksum* table.

Field	Description
sha1	Digest of the paper calculated using the SHA1 algorithm
fileType	Type of the file. For instance, PDF, PS etc.

Table D.10: Fields in the *paperVersions* table.

Field	Description
id	Unique id identifying for each paper version. System asgined auto increment.
name	Assigned name for that version
version	Number of this version (1st, 2nd, and so on)
repositoryID	Repository identifier which maps to the physical location where the version information is stored
path	Rerlative path from the ROOT of the repository to the location of the version information
deprecated	Flag to indicate if the version has been deprecated or not
spam	Flag indicating if the versio is spam or not
time	When the version was created

Table D.11: Fields in the *userCorrections* table.

Field	Description
id	Unique id identifying for each user correction. System assigned auto increment.
version	The version number

Table D.12: Fields in the *urls* table.

Field	Description
id	Unique id identifying for each URL in this table. System assigned auto increment different from the id in the crawler database.
url	URL to the paper

Table D.13: Fields in the *hubUrls* table.

Field	Description
id	Unique id identifying for each item in this table. System assigned auto increment.
url	URL to a resource which contains links to PDF/PS documents
lastCrawl	Last time when this URL was crawled
repositoryID	Repository identifier which maps to the physical location where the version information is stored

Table D.14: Fields in the *hubMap* table.

Field	Description
id	Unique id identifying for each item in this table. System assigned auto increment.

Table D.15: Fields in the *legacyIDMap* table.

Field	Description
id	Unique id identifying for each item in this table. System assigned auto increment.

Table D.16: Fields in the *textSources* table.

Field	Description
name	Text source name
content	Content of the text source

Table D.17: Fields in the *tags* table.

Field	Description
id	Unique id identifying for each tag. System assigned auto increment
tag	The tag
count	Number of time this tag has been used

Table D.18: Fields in the *link_ types* table.

Field	Description
label	External source identifier
baseURL	Base URL to the external source

Table D.19: Fields in the *elinks* table.

Field	Description
label	Link type identifier
url	Complete URL pointing to information about the paper on an external source. The URL is built base on the base URL from *link_ types* and a process that match papers within the corpus with the external sources.

Table D.20: Fields in the *pdfTables* table.

Field	Description
id	Unique id identifying each item in this table. The id is based on the paperid (Note by Juan: should be an auto increment)
caption	Table caption text
content	Table content in HTML format. (Note by Juan: This should be stored in XML and probably in the file system or both the database and the file system.)
footNote	Table foot note
refText	Text snippets where the table is mentioned. (Note by Juan: This should be treated the same way citation and acknowledgeemnt context are handled)
pageNum	Page where the table was found within the paper

Table D.21: Fields in the *cannames* table.

Field	Description
id sid	Unique id identifying for each item in this table. System assigned auto increment
canname	Cannonical name for an author
ndocs	Number of papers whin the corpus for that author
ncites	Number of citations for the author within the corpus
url	Home page for the author
affil	Author's current affilication
address	Author's current address
email	Author's corrent email address
hindex	Author's H-index

Table D.22. Fields in the *csx_ citegraph.clusters* table.

Field	Description
id	Unique id identifying for each cluster in this table. System assigned auto increment
size	This is the number of ciations in the cluster (no incollection documents).
incollection observations	Flag to indicate if the reference is a paper within the corpus or just a citation
selfCites	Number of self-citations
updated	Last time the cluster was updated. For instance, adding a new observation

Table D.23. Fields in the *csx_ citegraph.keymap* table.

Field	Description
id	Unique id identifying for each item in this table. System assigned auto increment
ckey	The generated key

D.2 The *csx_ citegraph* Database

In this section, we describe tables in the *csx_ citegraph* database. This database is primiarly used for hosting the citation graph information (relationships of papers and citations), such as the clustering relation, and citation relation. Data in this database is maintained by the inference process.

1. *clusters* table. Stores the canonical reference for paper and/or citations (Table D.22). For definition of *clusters*, see 7.4.
2. *keymap* table. Stores several keys to allow citation/paper matching. Keys are generated using normalization over the title and authors anmes (Table D.23).
3. *citations* table. Stores all the citations for a given paper which have been assigned to a cluster (if a citation appears here it must also appear in *citeseerx.citations* table; see Table D.24).
4. *papers* table. Stores all the papers which have been assigned to a cluster (if a paper appears here it must also appear in *citeseerx.papers* table; see Table D.25).
5. *citegraph* table. Associates papers with citing and cited clusters (Table D.26). *infupdates* table.
6. *deletions* table. Stores deleted clusters. When a user performs a metadata correction, the cluster of this paper is checked. If the cluster does not contain other members, it is deleted and recorded in this table.
7. *IndexTime* table. Last time the indexer was run (Table D.29).

Table D.24. Fields in the *csx_ citegraph.citations* table.

Field	Description
id	Unique id identifying for each item in this table. System assigned auto increment
cluster	The cluster this citation belongs to

Table D.25. Fields in the *csx_ citegraph.papers* table.

Field	Description
id	Unique id identifying for each item in this table. System assigned auto increment
cluster	The cluster this paper belongs to

Table D.26. Fields in the *csx_ citegraph.citegraph* table.

Field	Description
id	Unique id identifying for each item in this table. System assigned auto increment
citing	IDs of clusters citing other clusters
cited	IDs of clusters being cited by the clusters in the “citing" field.
firstContext	The first context where the citation is found
selfCite	Flag which indicates if this one is a self citation

Table D.27. Fields in the *csx_ citegraph.infupdates* table.

Field	Description
id	Unique id identifying for each item in this table. System assigned auto increment
lastupdate	

Table D.28. Fields in the *csx_ citegraph.deletions* table.

Field	Description
id	Unique id identifying for each item in this table. System assigned auto increment
deleted	Timestamp this deletion occurred.

Table D.29. Fields in the *csx_ citegraph.IndexTime* table.

Field	Description
id	Unique id identifying for each item in this table. System assigned auto increment
param	Process name
lastupdate	Last time the procesds was run