Citebase Search: Autonomous Citation Database for e-Print Archives

Tim Brody <tdb01r@ecs.soton.ac.uk>

Intelligence, Agents, Multimedia Group
University of Southampton

Abstract

Citebase is a culmination of the Opcit Project and the Open Archives Initiative. The Opcit Project's aim to citation-link arXiv.org was coupled with the interoperability of the OAI to develop a cross-archive search engine with the ability to harvest, parse, and link research paper bibliographies. These citation links create a classic citation database which is used to generate citation analysis and navigation over the e-print literature. Citebase is now linked from arXiv.org, alongside SLAC/SPIRES, and is integrated with e-Prints.org repositories using Paracite.

Introduction

Citebase Search is a Web service that harvests research articles from online, e-print archives (both pre- and post- peer-reviewed articles). Citebase parses references from the full-text, citation-linking those references to the cited articles. This is the basis for a classical citation database.

Citebase was first announced in December 2001 [1], although it was not until Citebase was integrated in August 2002 with arXiv.org that any sizeable number of users started to use Citebase. As an add-on to arXiv.org, arXiv provides links from its abstract pages to the equivalent abstract page in Citebase. Citebase currently draws most of its traffic from users following these links, without them then going onto use Citebase's own search service (see Figure 1).

In the first section I will introduce the motivation behind Citebase, section 2 gives some of the structure of the system, section 3 introduces the service as seen by the user and section 4 gives an example of some of the data mining that can be performed on Citebase's database.

Top 30 of 110 Total URLs					
#	Hits		KBytes		URL
1	118220	34.68%	2414846	32.63%	/cgi-bin/citations
2	94899	27.84%	272849	3.69%	/cgi-bin/graph
3	38444	11.28%	401754	5.43%	/cgi-bin/search
4	25283	7.42%	12513	0.17%	/style.css
5	13662	4.01%	4009348	54.17%	/cgi-bin/oai2
6	2513	0.74%	25166	0.34%	http://citebase.eprints.org/usage/
7	1699	0.50%	5714	0.08%	/cgi-old/soap-proxy
8	623	0.18%	3821	0.05%	/help/
9	419	0.12%	321	0.00%	/help/context/graph/
10	293	0.09%	1779	0.02%	/help/coverage.php

Figure 1 Top 10 most hit pages in Citebase in September 2003. By a ratio of 3:1 Citebase receives more hits to its '/cgi-bin/citations' page (the abstract page linked from arXiv.org) to its search engine ('/cgi-bin/search'). See [2] for more usage statistics.

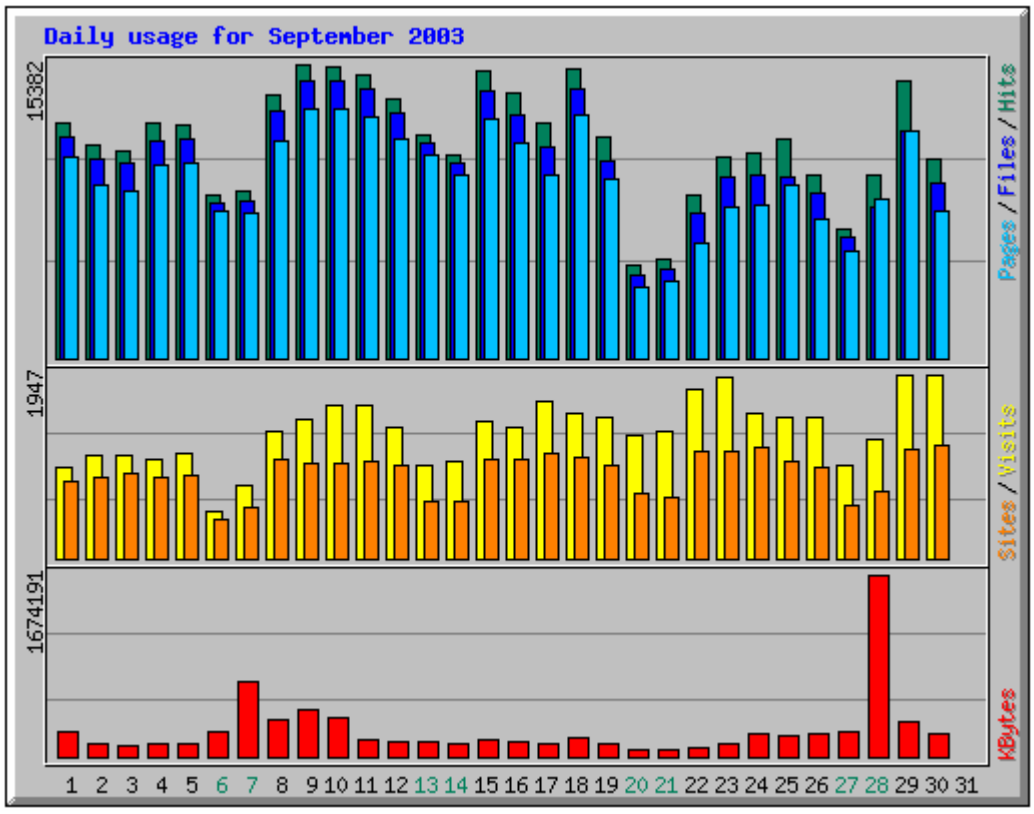

Figure 2 Usage statistics for Citebase Search in September 2003 (excludes most search engine crawlers, but includes downloads from the OAI interface). Usage of Citebase is steadily increasing, with approximately 1000 users visiting each day. Citebase accounts for around 8GB of outbound traffic each month.

Motivation Behind Building Citebase

Citebase started as a simple metadata search engine, harvesting Dublin Core data from arXiv.org using the Dienst protocol (OAI 0.9). Citebase was one of the first two OAI services. While ARC [3] has covered all OAI repositories and has extended

into building subject trees, Citebase has focused almost exclusively on arXiv.org and building a citation linking service for it.

In focusing on arXiv.org Citebase is still very much the product of the Open Citation Project (1999-2002) [4]. OpCit aimed to "attempt to hyperlink each of the over 200,000 papers in Los Alamos's unique online Physics Archive to every paper in the archive that it cites." This statement has had to be updated during the duration of the project, as arXiv's holdings have gone from 120,000 papers (when Citebase was first started) to over 250,000 now.

OpCit's approach to citation linking was to build a database of number triples (year/volume/starting page), and then when parsing a document the references could be linked by finding any series of three numbers and looking them up. This citation linking was done in-situ within a PDF version of an article by passing the article through a program that searched and parsed the text using a library inherited from the Open Journals Project. The first Citebase-with-references provided links to the full-text of the article from the metadata search engine (and - as the user retrieved the full-text - the OpCit linker highlighted linked citations, so the user could click a reference and move instantly to the cited article).

As Citebase developed the citation linking function was more tightly integrated, until the OpCit database was dropped in favour of using an SQL database. Citation linking was then done at the stage the article was harvested from the source archive, with the references and the identifier of the cited articles (if found) being stored. As well as avoiding the need to perform citation linking in real-time (allowing more complex citation matching to be performed, by using the number triple, authors, journal title and article title), this citation database allowed a citation navigation interface to be built and – when performing a search - to rank articles by their citation impact (the number of citations to an article).

Post-OpCit the motivation for Citebase is three-fold – as an ongoing service to physicists, as part of the broader e-prints strategy developed at the University of Southampton, and as a source for researching infometrics (data mining) the research literature.

Although no direct support has been required from the physics community, as the most active subject within the e-print community there is demand for a service like Citebase. arXiv.org demonstrates the circularity of demand - the more used a tool is, the better it becomes, the more it is used and so on. This is especially critical to information services – without information there is no service. There is considerable interest in self-archiving, performance measuring and Open Access. Considering the cost of current assessment exercises, it isn't unfeasible to imagine an assessment service being built (which collates and measures the output from authors), that has the side-benefit of being a citation search and navigation service.

Southampton has long had an interest in e-prints – both from a scholarly publishing interest (Prof. Stevan Harnad and Dr. Steve Hitchcock), and from a hypertext perspective (Dr. Les Carr). These projects have coalesced under the e-Prints.org banner. The GNU e-Prints software, developed from the software behind the Cogprints e-print archive, allows institutions or individuals to set up e-print archives. E-Print archives are collections of research articles deposited by their authors, usually for access by anyone on the Web. OpCit and Paracite have looked at building reference linking services for arXiv.org and GNU e-Prints respectively. The TARDIS project (part of the JISC e-Fair cluster) is studying how to get authors to self-archive (self- vs. 3rd party archiving of research materials). Citebase is a cross-archive search engine with citation ranking, navigation and analysis.

Lastly, Citebase's database provides a rich source of data for data mining the research literature. We – at Southampton – are particularly interested in studying how free access effects the research literature (Do free-access articles receive more citations than toll-charged access?).

How Citebase Works

Citebase combines metadata harvested from e-print archives using the Open Archives Initiative Protocol for Metadata Harvesting (OAI-PMH) and references parsed from the full-text, harvested using bespoke interfaces.

Like most OAI services Citebase harvests Dublin Core metadata. For e-print articles this typically has the article title, authors (unstructured, but separated), abstract/comments (under the generic DC 'description'), locations (e.g. a URL), a citation (free-text under either DC's 'source' or 'identifier'), and some dates (e.g. dates of revision). The article title and abstract are put into an inverted index for free-text searching. The authors are stored with the lastname/firstnames separated (by splitting around a comma). The citation is parsed using the OpCit reference parser, which can normally extract the journal title (often abbreviated), publication year, volume/issue and pagination.

The three archives that Citebase harvests from (arXiv.org, biomedcentral (BMC), and cogprints) require different mechanisms to retrieve the full-text. BMC is the simplest, as full-texts can be retrieved in XML by requesting a different metadata format from BMC's OAI interface. As BMC articles are in already semantically marked-up (e.g. an author is contained with an 'author' tag), the structured references can be read directly and stored in the database.

Most arXiv.org articles are in Latex format (a language that is primarily a layout description, but some semantic information can be derived). This 'source' format is harvested by Citebase and is parsed for references, which are in turn parsed into structured references. If the source can not be parsed a PDF version is retrieved from arXiv (which will automatically convert source Latex and Postscript), converted to plain text, and then parsed.

The Dublin Core metadata from cogprints includes the URLs of the formats available for the full-text of the article. Citebase retrieves the formats it can handle best – PDF, HTML and plain-text – and attempts to parse out the reference section and individual references.

The code used to find the reference section and parse individual references is derived from the OpCit project and includes some features from Paracite [5]. When a Latex file is parsed the OpCit code uses the Latex 'bibitem' (and a few variants) to locate references, marking the start and end of each reference. When the Latex file is processed the text of the reference can then be extracted by finding the markings in the output. When parsing unmarked-up text (PDF, HTML etc.) the reference parsing code finds the reference section by looking for an appropriate title ('References', 'Bibliography'), then extracting the references from the following text using some predefined styles, e.g. square-bracket numbered references. Individual references are, again, parsed using predefined styles, e.g. author at the beginning of the string, a 4-digit number is likely to be a year, an abbreviated journal title, title in quotes and so on. This domain-specific code works fairly well, but means Citebase is limited in its ability to handle non-arXiv subjects (physics, maths). There is also a specific problem with compound references - where an author puts a reference to more than one item under a single reference number. Compound references are a sequence of references with no clear delimiter, which can lead to problems with the reference parsing code

finding elements it thinks are from one reference, but in fact refer to completely different articles (and even authors).

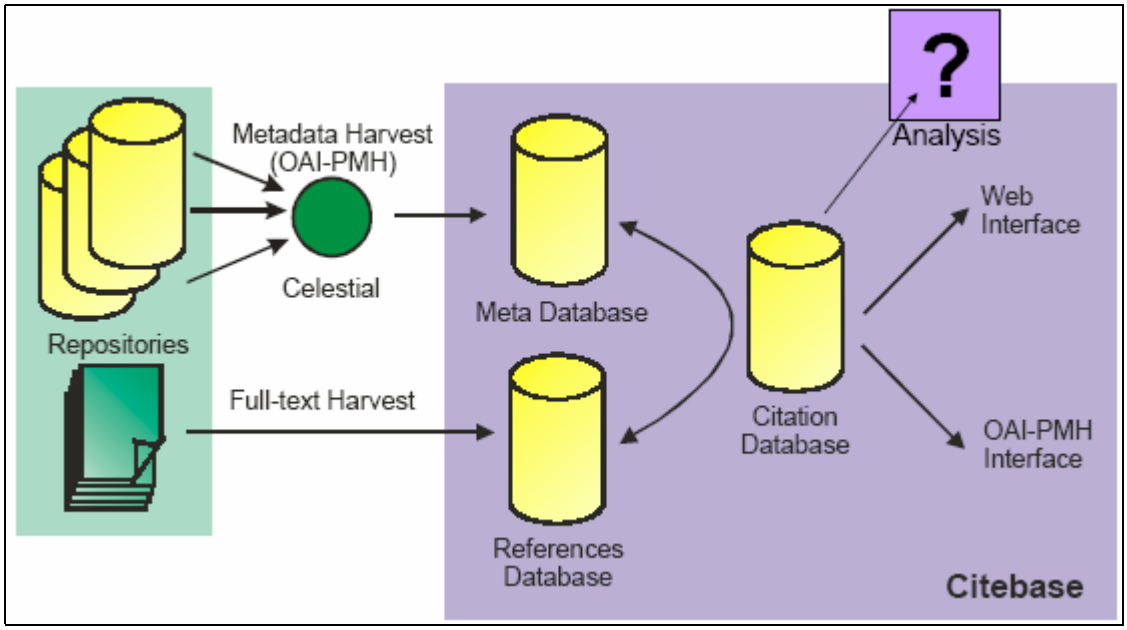

Figure 3 Data flow within Citebase. Metadata are harvested using the OAI-PMH from source repositories (using Celestial to cache). Full-text's are then retrieved for each metadata record using bespoke interfaces. The Full-text (from which references are parsed) and Metadata are used to build a citation database, which is the basis for a Web service, a separate OAI-PMH export, and data mining analysis.

The Web interface to Citebase – the user service – is a metadata search engine that provides links to abstract pages (Figure 4). The search engine allows searches to be made by author, title/abstract free-text, the journal title, and date of publication. Results can be returned in one of 6 rankings: citation impact of the article, citation impact of the article's authors, web downloads of the article, web downloads of the article's authors, date of creation, and date of last update. Citation impact and web downloads are the total number to an article, which biases older papers that have had longer to acquire citations and downloads respectively.

The Citebase abstract pages receive the most number of visits, as they are linked from the equivalent pages at arXiv.org. They allow a user to use citation navigation to find articles that are related to the current article, by both following referenced articles, articles that cite the current article, and co-cited articles. A link to the PDF full-text is provided, which passes the PDF through the OpCit linker, making links from references in the text to the cited article in Citebase (Figure 5).

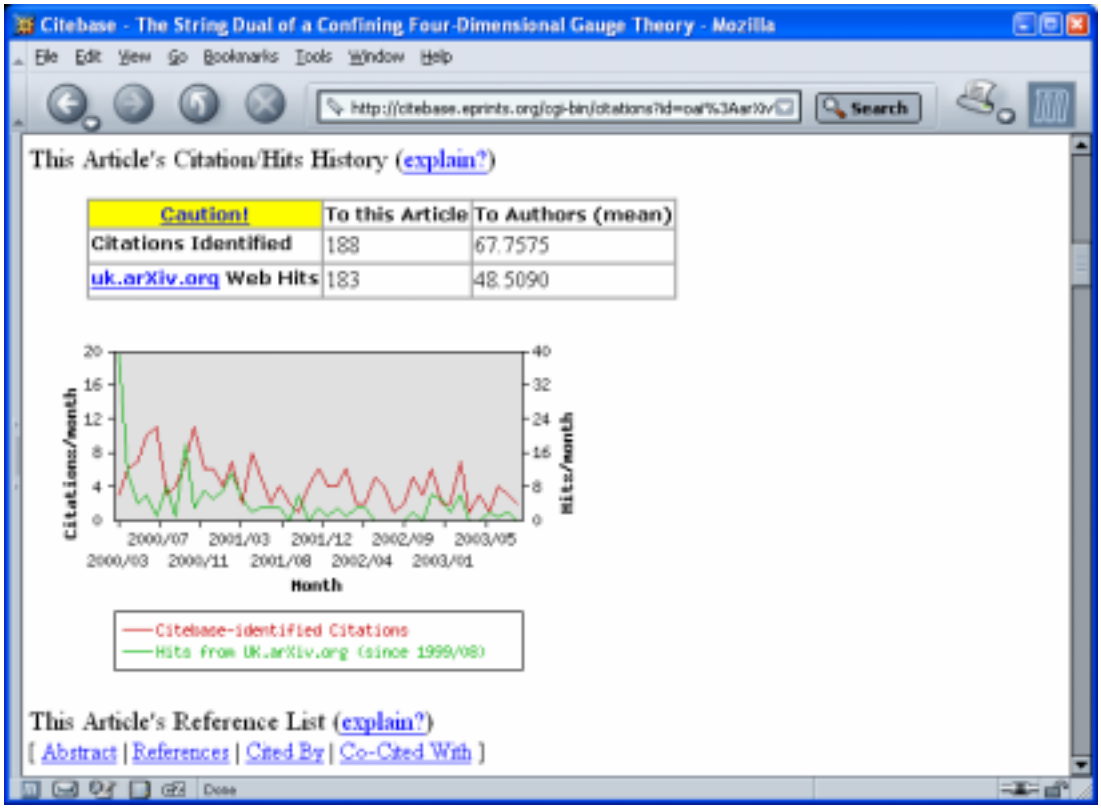

Figure 4 Citebase's abstract page. As well as the standard title, authors, abstract of the article this page also shows the citation data for the article. A summary section gives the number of citations to the article and the mean number of citations to its authors, similarly for Web downloads. A graph shows when the article has been downloaded and cited. Other sections show the references from the article, citations to the article ("Who has cited me?"), and co-cited articles ("Who has been cited alongside me?").

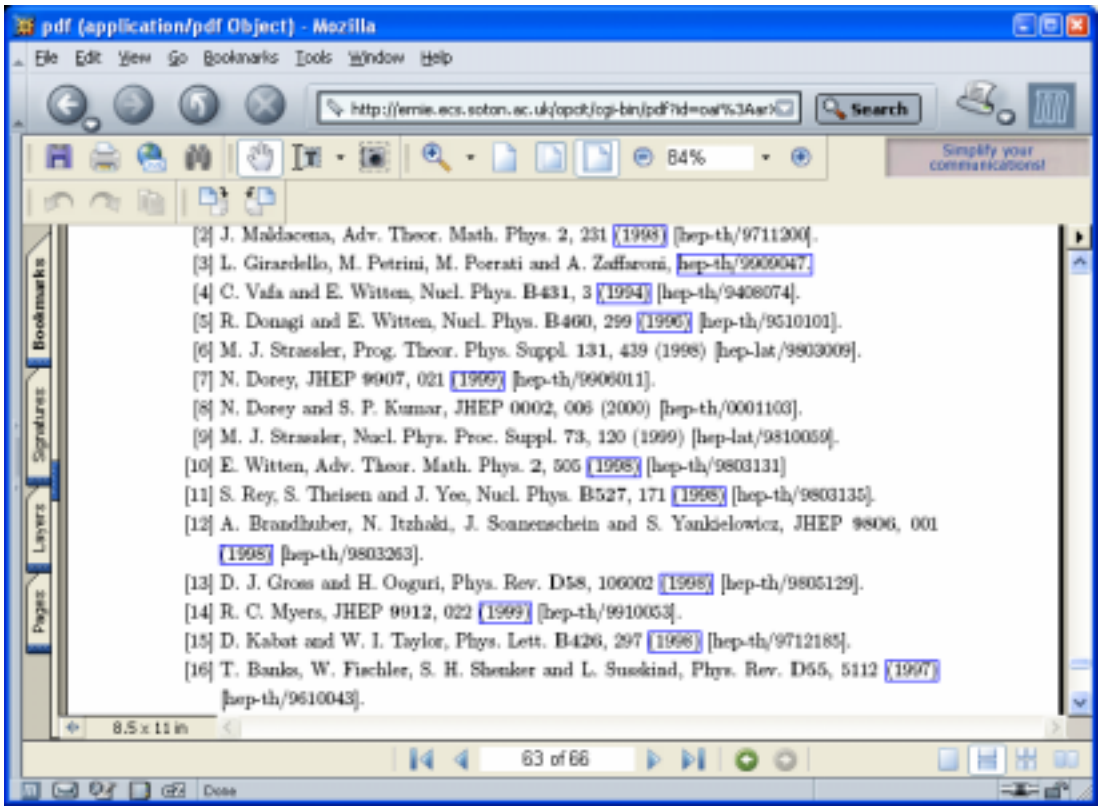

Figure 5 Reference-linked full-text PDF. Although the author has provided arXiv identifiers for his references, the Citebase reference linker has also provided links for the citation data (on the year of publication).

Data Mining Citebase

Data mining is the process of finding patterns within a set of data. Citebase is a database of 250,000 articles, 6 million references (of which 1 million are linked to the full-text), and approximately 100,000 authors. There are approximately 2 million identifiable cited items (a combination of the 250,000 self-archived articles and the articles that they cite).

Figure 6 shows how authors are identified from the unstructured strings associated with articles (the archives that Citebase harvests from only store the name of the author, with no identifier or extra piece of information to cross-reference two names against). Some authors who share the same, or similar, name will be incorrectly joined together, while others will not be joined together because of different spellings (or, in a few cases, where the author has changed names e.g. by marriage).

Figure 7 and Figure 8 show some basic analysis of co-authorship within arXiv.org.

Papers	Mean Authors/Paper	Standard Deviation
256,116	2.6296	2.7223

Total Named Authors	By Family Name	By Family Name & First Initial	By Family Name & All Initials
673,484	72,894	109,462	129,931

Hawking

Hawking,S.

Hawking,S.W.

Figure 6 Authors identified by Citebase. Out of about 250,000 papers 670,000 authors were named (2.6 per paper). These names can be collapsed together (to try to find real-world authors) by a number of means: by lastname only (which results in 72,000 authors), lastname and first initial (110,000), and lastname and all initials (130,000).

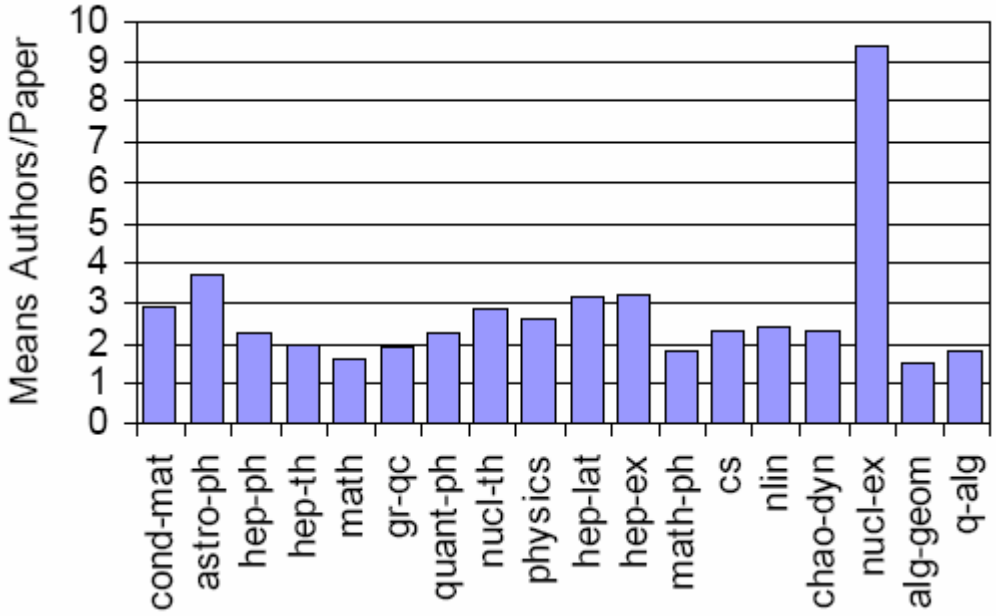

Figure 7 The number of authors per article by arXiv.org sub-field. Theoretical fields (e.g. hep-th, math) have fewer co-authors than experimental fields (nucl-ex, astro-ph). This is intuitive: articles on experimental results will have required more people to produce, than theoretical.

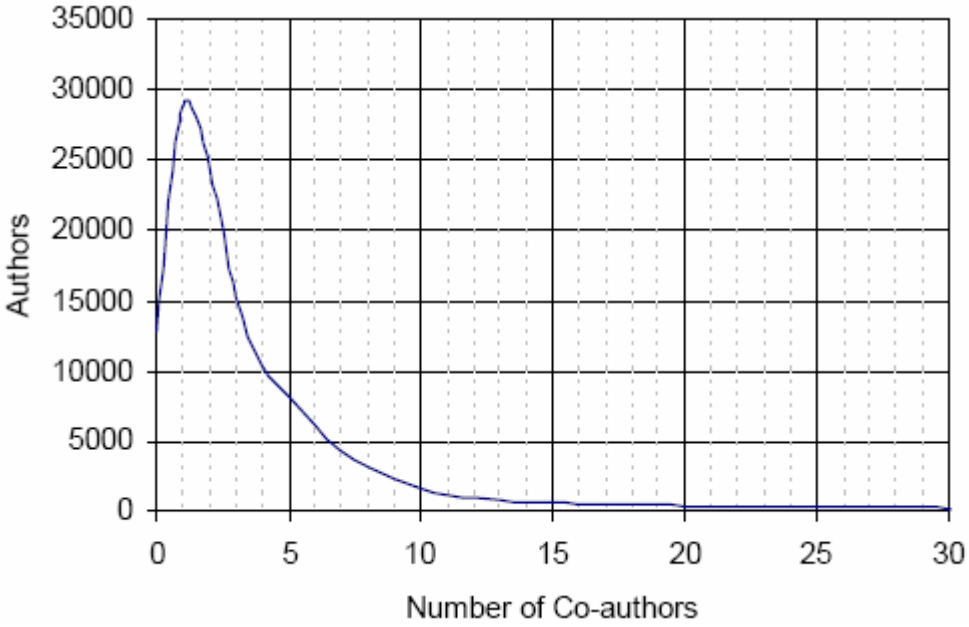

Figure 8 How many authors have individuals co-authored with? This distribution shows that most authors have co-authored with one other author (surprisingly few). If we are to look for communities of practise within arXiv.org by studying co-authorship, this distribution does not support this idea – if most authors are only linked to one other author, the connectivity in the co-authorship graph may not be very high.

Citation latency is the time between an article being deposited, and a subsequent article being deposited that cites it (Figure 9). It is the delay in the research cycle – from research being written, being read, and then built on and written about by others. Looking at citation latency over time in arXiv.org (Figure 10) we can see that the peak rate of citations has been reducing. This suggests that the effect of the arXiv.org archive – that provides near-instant access to research results – is to reduce the time delay in the research cycle, increasing productivity.

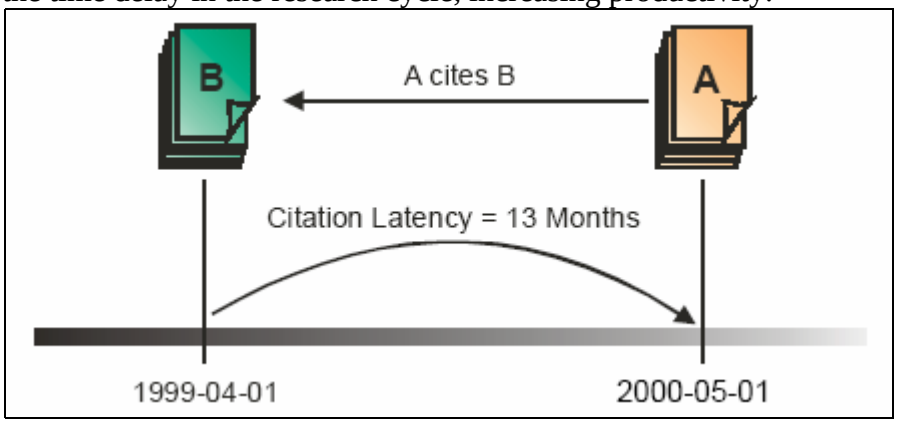

Figure 9 Citation latency. Article A cites article B with a citation latency of 13 months - the time between the two articles being deposited.

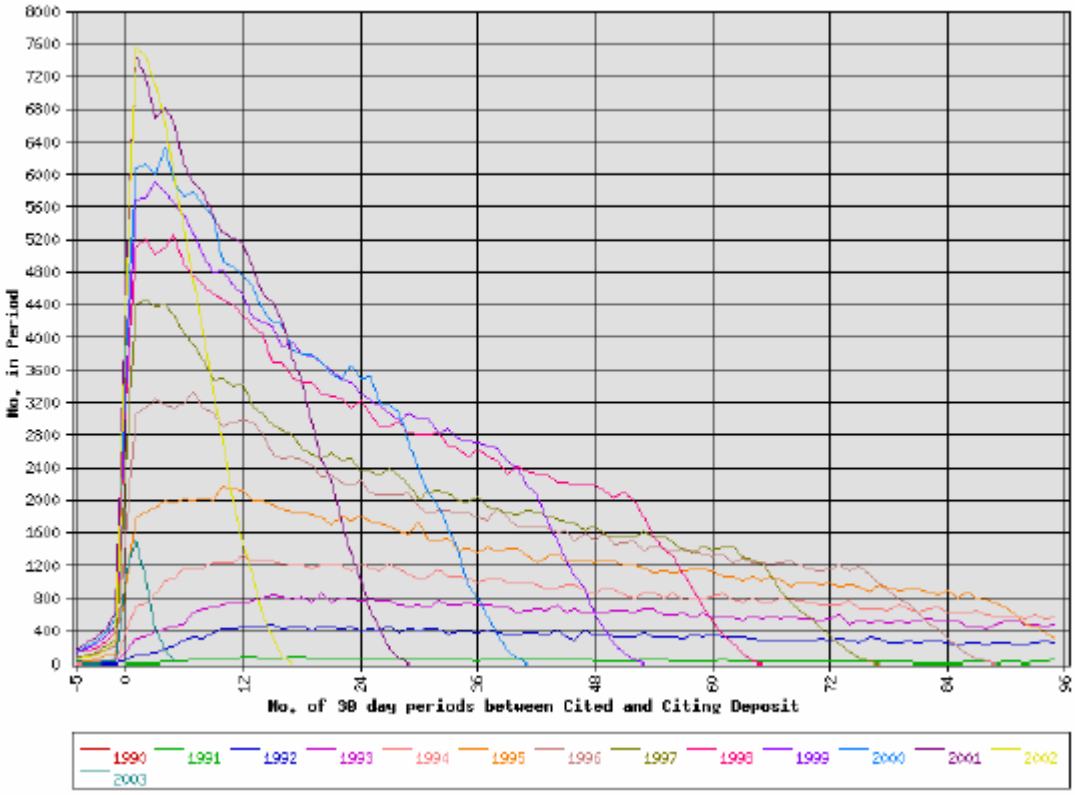

Figure 10 Changing citation latency over time in arXiv.org. More recent years have a peak of the number of citations (by citation latency) sooner after the article is deposited than in previous years. E.g. in 1995 the peak rate of citations were at roughly 12 months after an article was deposited, compared to 2000 where the peak is at 3-4 months.

Conclusion

Citebase Search continues to gain increased usage, in-line with increasing use of e-print archives. The challenge before Citebase – as a service – is to expand its ability to automatically parse and understand the information that it harvests. One strategy to improve Citebase's ability to understand the literature is to influence the communities that it serves. This could be reflected in building services for authors, that allow the author to correct their references, and in building more customisable services (e.g. Web resumes). A corrected bibliography for an author could be used to instantly provide a citation (and Web-) impact score for an author, to show who has cited their work, who they've worked with (co-authorship), and what fields they are working in.

For now, Citebase provides a rich source of data to analyse the physic's arXiv.org research literature. It is hoped that Citebase provides a useful service to the physic's community, and provides an incentive to other areas of research to provide open access to research literature.

Bibliography

[1] Peter Suber (2003) "Open Access Timeline" http://www.earlham.edu/~peters/fos/timeline.htm

[2] Tim Brody (2003) "Citebase Search Usage" http://citebase.eprints.org/usage/

[3] Xiaoming Liu, Michael Nelson (2001) “Arc – An OAI Service Provider for Digital Library Federation” D-Lib Magazine April 2001 v7 n4 http://www.dlib.org/dlib/april01/liu/04liu.html

[4] Steve Hitchcock, Donna Bergmark, Tim Brody, Christopher Gutteridge, Les Carr, Wendy Hall, Carl Lagoze, Stevan Harnad (2002) “Open Citation Linking: The Way Forward” D-Lib Magazine October 2002 v8 n10 http://www.dlib.org/dlib/october02/hitchcock/10hitchcock.html

[5] Mike Jewel (2003) “Paracite” http://paracite.eprints.org/developers/

[6] Tim Brody, Simon Kampa, Stevan Harnad, Les Carr, Steve Hitchcock (2003) “Digitometric Services for Open Archives Environments” Proceedings 7th European Conference, ECDL 2003 207pp *and references therein* http://eprints.ecs.soton.ac.uk/archive/00007503/