Proceedings of the 9^{th} International Conference on Applied Informatics
Eger, Hungary, January 29–February 1, 2014. Vol. 2. pp. 29–39
doi: 10.14794/ICAI.9.2014.2.29

A different approach to Universal Decimal Classification in a Mechanized Retrieval System

Attila Piros

University of Debrecen
atilla.piros@gmail.com

Abstract

UDC is one of the world's foremost classification systems having been published in around forty languages and used in more than 120 countries worldwide. This prevalence is partially due to its having been invented more than one hundred years ago and its having been under continuous review since then in order to ensure that it constantly reflects the current status of the sciences and human knowledge. The other key to its popularity is its analitico-synthetic and faceted nature; in addition to attempting to collect and organize all branches of human knowledge in a hierarchical structure it provides a system of auxiliaries in order to create new concepts by joining existing ones. However, library software still cannot employ it perfectly. The three main solutions currently used to utilize UDC in information retrieval systems are: handling UDC notations as text strings, compiling a so-called KWOC-index of the parts of complex concepts and using an authority control function to manage the classification vocabulary. Neverthless, further improvements may result in a further increase in the effectiveness of bibliographic analysis and information retrieval by taking advantage of the characteristics of UDC. The first step to doing this is to implement a software interpreter to analyze UDC numbers. The goal of this paper is to present a program written for this purpose.

Keywords: Universal Decimal Classification, UDC, Library Classifications, Information Retrieval Systems, Information Retrieval

MSC: 68P20

1. Introduction

Universal Decimal Classification is one of the world's foremost library classification systems, being used in hundreds of thousands of library collections and their online catalogues, which try to adapt to the particularities of the system in order to retrieve contents based on their subjects.

Ágnes Hajdu Barat sets out the following in the introduction to her study about the possibilities of Hungarian OPACs [1]:

"What advantages has the UDC?

- Universal system. Standardization.
- Meaningful notation.
- Clarity and transparency.
- Rich network of relationships.
- Well-defined categories.
- Ability to describe special and general concepts with free movement between the different levels.
- Efficient retrieval, relevant hits.
- (...)
- The concept system is our common cultural heritage and value.
- Significant potential.
- Well-developed hierarchies, able to visualize information and conceptualize it independently within its structure [2].
- Last, but not least: UDC is nice. With a nice philosophy, a rich collection of concepts, clear principles, understandable structure, easy to survey, and logical relationship."

As she points out "there is marked interest in the UDC's potential to assist growing numbers of Internet users. The UDC can play a role of integration in knowledge organization." Her conclusion is that we should keep UDC in our retrieval systems instead of abandoning.

Questions like the above are raised because the information retrieval software systems currently used haven't reflected the significance of UDC numbers by using all the capabilities of the classification, although improvement in informatics and programming techniques make it possible to invent more effective UDC based retrieval methods.

In this paper I will try to outline the basics of solutions currently used to employ UDC in OPACs to search information, together with some of their weaknesses; afterwards I will provide a feasible alternative to these. Finally, as part of the alternate solution mentioned immediately above, I will demonstrate a software interpreter to analyze UDC numbers.

2. About the usage of Universal Decimal Classification in Information Retrieval Systems

In this section I will outline the original goals and nature of the classification itself and the basic techniques currently used in its employment as an indexing language in mechanized information retrieval systems.

2.1. About Universal Decimal Classification

Universal Decimal Classification was invented by two Belgian lawyers, Paul Otlet and Henry La Fontaine. After having investigated of a copy of Melville Dewey's Decimal Classification, they found it capable of being the basis of a universal bibliography and they organized the first International Conference on Bibliography, which assembled in Brussels in 1895, to consider their plan.

The conference created the International Institute of Bibliography (IIB) and the International Office of Bibliography (OIB) among others, to develop the Universal Bibliographic Repertory, a comprehensive bibliography of all information published worldwide [3].

After having obtained the permission to translate and expand the schedules of DDC, they also defined various relations between subjects with their signs and introduced terminal signs and tables of auxiliary numbers for common characteristics which can be used together with many subjects. They performed changes mentioned immediately above not only in order to create a classification for library use, but an international language based on numbers. Eventually they invented the world's first analytico-synthetic [4] classification system with some faceted characteristic too [5].

Finally, the Universal Bibliographic Repertory was presented in 1900. The handbook for this, which contained the first edition of UDC was published in France between 1902 and 1907. Since then UDC has been under continuous review by the IIB and its successors (and by the UDC Consortium since 1992) [4].

Over the course of the years UDC has become one of the world's foremost classification systems having been published in different editions in more than 39 languages and used in at least 124 countries [6]. There are currently over 140,000 libraries employing UDC in Europe alone [7] and collections using it include the Scientific and Technical Information Institute of the Russian Academy of Sciences (VINITI RAS), the Hungarian National Union Catalogue (MOKKA) and The Network of Libraries and Information Centers in Switzerland (NEBIS) [8].

2.2. Mechanization of using UDC

As W. Boyd Rayward pointed out, the auxiliary signs and tables mentioned above resulted in a schizophrenic nature to the classification itself. As a bibliographic classification system its main purpose is to specify the subjects contained in documents by using its notation to relate the document to the classification as closely as possible; the capability to create compounds and reversal was provided to make this relation easier to conduct. However, the classification, schedules and auxiliary signs make deeper bibliographic analyses possible also. Then the notation's role changes, it becomes primary and the linguistic aspect comes to the fore. As mentioned above, this was the original goal of the inventors as well. Otlet believed that the notation of UDC can be used as a language, by using numbers as words and with auxiliaries expressing grammatical functions [9].

As the numbers mentioned above demonstrate, UDC is used in numerous library information retrieval systems. We can say that they mostly approach it from the bibliographic aspect and not from the linguistic one.

The need to use classifications in mechanized information retrieval systems was almost contemporaneous with the arrival of the first computers. UDC was most likely used in a mechanized retrieval system for the first time by E. G. Birsch [10]. KWOC (Keywords out of Context) form indexes were used for UDC numbers by Klaus Schneider and Karl-Heinz Koch in the 1960s. They compiled an index of parts of complex concepts by moving them out of context in order to retrieve documents with greater precision [11, pp. 36.].

The systems currently being used often store UDC numbers as simple text strings, but only some of these are able to compile a KWOC-index of them.

However, the predominant model of utilizing classification systems in modern information retrieval software is managing authority files; in addition to a UDC number, an authority record can contain links to its broader and narrower classes, other classes related to its equivalents in other classification or subject heading systems or searching terms in national languages for example [12]. The authority file provides a controlled vocabulary of UDC notations used in a retrieval system in a standard format (like the MARC 21 Format for Classification Data [13] or the UNIMARC Classification Format [14] for example) more or less capable of storing its inner structure. The concept of authority control was mainly invented in order to solve the problem of reusing the same authority entries and to help indexers and searchers to access subjects by using their descriptions in a national language instead of the artificial codes of the classification systems. In spite of the advantages mentioned immediately above, employing UDC authority files restricts deeper content analysis and the usage of synthesised UDC numbers; access to authority records is also restricted because the access points and relations between them are predefined manually, rather than being recognized by an automatic process.

In a pilot study conducted in 2004-2005, Aida Slavic examined the existence of the following functionalities in thirty Web OPACs (the numbers following the names of the functionalities are the percentages obtained as the result of the research) [15]:

1. automatic right truncation 66.7%

2. approximate matching (i.e. 'no zero result' option) 46.7%

3. availability of a Boolean search 10%

4. searching parts of a complex UDC number 23%

5. searching a UDC caption 36.7%

6. searching/browsing from an authority record choosing any related terms within the record. 16.7%

We can see that searching parts of a complex UDC number through an index is available in fewer than a quarter of the systems and using Boolean operands to join numbers to build more precise queries in only the tenth of them.

As the author of the article above points out, searching UDC captions and UDC using subject heading systems requires an authority file. Indexed searching of parts of a complex UDC number requires an authority file or at least a KWOC-index too.

However, while searching the parts of a number is also possible if numbers are handled just as simple texts flexibilities in citation order (cf. UDC Summary Linked Data [16]) make it difficult. To avoid losing data the searcher should collect every possible citation order of parts of the number. Using wildcards is a useful tool in doing this as it raises the level of recall, which causes the level of precision to be lowered. Handling extensions and special auxiliaries (or facets) effectively are impossible in this way.

Using both KWOC-indexes and authority files primarily supports post-coordinated searches, which means that the user can create queries from UDC numbers using Boolean operands. Just as in the case of the method above, retrieving numbers hidden in extensions or searching for special auxiliaries, is not supported in this way, nor is handling subgrouping. The differentiation of concepts like 329.17:329.12 (relations of national and liberal movements) and 329.17'12 (national-liberal movements) is also impossible in this way.

The problem may derived from the fact that information regarding context was lost permanently during indexing; so we cannot utilize the information about which parts of a complex number join to each other, with which operands and in which order during the searching phase.

3. A feasible alternative

A feasible alternative to the solutions mentioned above is a system which analyzes the numbers and compounds during the indexing process, expands their inner structure and recognizes not only the parts of the numbers but information about how they join to each other too. Finally, it stores the result of the analyzing process on a database or as part of an authority record.

The method mentioned immediately above makes it possible to use complex UDC numbers as queries; the system can conduct the same analysis on them and compare its result with the stored information.

If the analysis is sophisticated enough, it can obtain all the knowledge about the number, which can constitute a satisfactory base for working more effective searching algorithms.

Because the generated representations not only contain information about the parts of the compounds but about the operands joining these to each other as well, it is possible to trace the changing of both the schedules and the rules. It also makes possible to expand the relationships between complex numbers, not only between a compound and the single numbers constituting it. In conclusion, merging different authority files would be possible too.

A relevant advantage of saving information regarding context is that it makes handling special auxiliaries (facets) and numbers hidden inside intervals possible.

A system as described above must have the following main components:

- The interpreter program to analyze UDC numbers and produce their representations
- The database or authority file to store the representations
- The searching software, which presents a result list after comparing the result of analyzing the query with the representations stored on the database instead of only using the relationships defined previously in the authority records.

The goal of this paper is to present the basic principles and a working implementation of a software interpreter capable of analyzing UDC numbers, in order to demonstrate that it is possible to process them to the degree of detail necessary for the retrieval of information using the method described above.

3.1. The basics concerning how the interpreter works

The rules governing the usage of UDC determine a formal language over an alphabet of decimal digits, dots and the characters of UDC metalanguage symbols. The interpreter is an automata which recognizes this formal language.

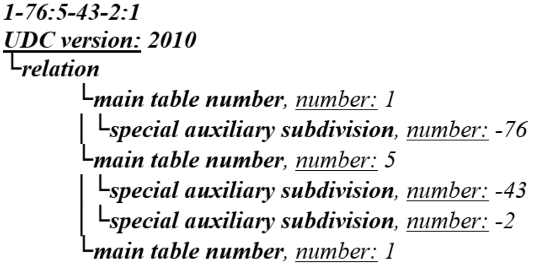
The input words are UDC numbers and the states of the automata describe the type and role of that part of the number to which the letter being processed belongs. While processing the number, the automata generates a tree which contains the parts of a number, based on the rules of precedence regarding auxiliary signs and joining auxiliary numbers. It also contains some additional information which may be necessary during a search. If the automata doesn't recognize the input word, then the given number is invalid or cannot be interpreted in the given UDC version. This means that the language recognized by the automata specified by a year is just the UDC edition current that year.

The inputs of the algorithm are the UDC number and the year of the UDC edition which was used to build it; the output will be the hierarchical representation of the number, or an error message which describes the problem if it cannot be interpreted in the given version. The software analyzes UDC numbers in a syntactic way using only the rules of UDC, which means that it doesn't need to store UDC numbers or any part of the tables. In the next phase of implementation we can fill the database with the generated representations and conduct searches using it.

3.2. Examples

The following examples represent the outputs of the software described above after receiving different UDC numbers as input. The goal is to demonstrate the hierarchical approach which renders it possible to describe and save the inner structure of UDC concepts.

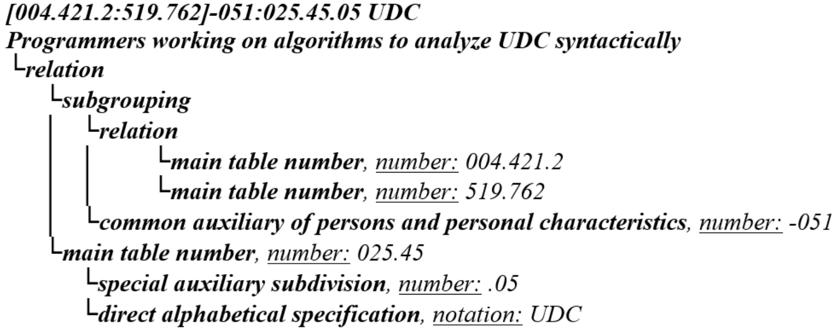
The first example was presented by Claudio Gnoli and Aida Slavic in their lecture on developing facets in UDC, to demonstrate the correct way of parsing special auxiliaries as a requirement for the computer software [17]. The interpreter solves this problem by keeping the structure of the whole compound by providing the output below:



```
1-76:5-43-2:1
UDC version: 2010
└relation
        └main table number, number: 1
        │ └special auxiliary subdivision, number: -76
        └main table number, number: 5
        │ └special auxiliary subdivision, number: -43
        │ └special auxiliary subdivision, number: -2
        └main table number, number: 1
```

The other examples were built using the Hungarian UDC edition published in 2005 [18].

The following example shows the handling of the special auxiliaries and the direct alphabetical specifications, just like the subgrouping. The first lines contain the number and its description, the tree is the result provided as the output of the software:



```
[004.421.2:519.762]-051:025.45.05 UDC
Programmers working on algorithms to analyze UDC syntactically
 └relation
     └subgrouping
     │   └relation
     │   │       └main table number, number: 004.421.2
     │   │       └main table number, number: 519.762
     │   └common auxiliary of persons and personal characteristics, number: -051
     └main table number, number: 025.45
         └special auxiliary subdivision, number: .05
         └direct alphabetical specification, notation: UDC
```

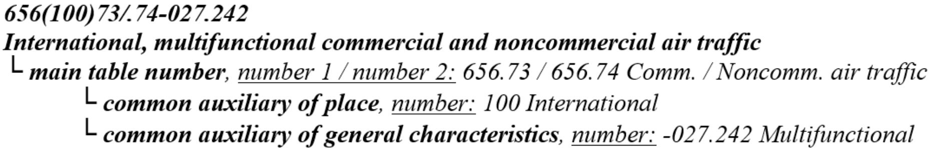
The relation marked with a colon means the connection between the two terms being joined by it. Subgrouping, marked with square brackets, is for determining the precedence of operands and auxiliaries.

In this example the number for ‘persons as agent’ describes mathematical algorithms for the syntactical study of systems of symbols, meaning the persons working on these kinds of algorithms. If it had been joined to the second part of the index, it would have meant the persons editing the classification itself. This reveals the importance of information regarding context too.

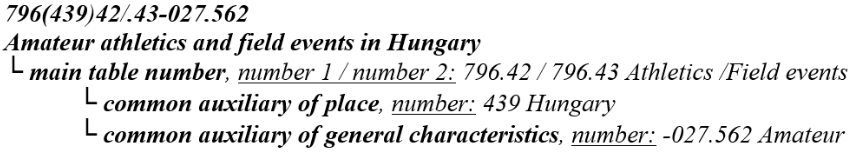
Below we can see two UDC numbers with different meanings but the same inner structure. These both contain an extension to describe an interval of main table numbers, an auxiliary of place (between brackets) and an auxiliary of general characteristics (prefixed with -0).

The same structure means that the automata will go through the same states while generating the results.

If the given UDC edition predates 1999, the software will display an error message because the auxiliaries of general characteristics were introduced in that year [19]; otherwise it will produce the following results (the descriptions of the parts of the compounds have been added only in order to make the results easier to understand):



```
656(100)73/.74-027.242
International, multifunctional commercial and noncommercial air traffic
└ main table number, number 1 / number 2: 656.73 / 656.74 Comm. / Noncomm. air traffic
        └ common auxiliary of place, number: 100 International
        └ common auxiliary of general characteristics, number: -027.242 Multifunctional
```

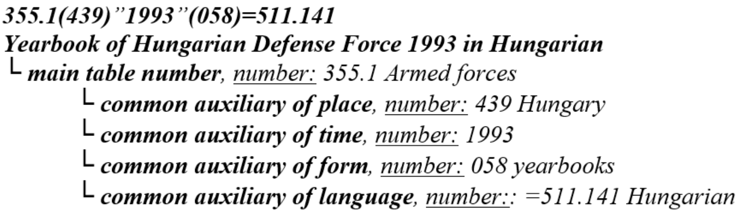


```
796(439)42/.43-027.562
Amateur athletics and field events in Hungary
└ main table number, number 1 / number 2: 796.42 / 796.43 Athletics /Field events
        └ common auxiliary of place, number: 439 Hungary
        └ common auxiliary of general characteristics, number: -027.562 Amateur
```

As was mentioned above, there exist a flexibility in the citation order of auxiliaries; it can be changed in order to list numbers containing the same auxiliary of place, form, language etc. in one place. For instance, the notations below differing from each other in the citation order have the same meaning and inner structure:

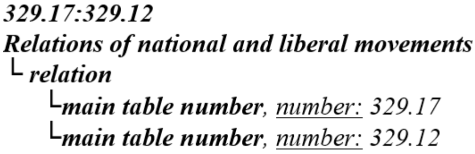
355.1(439)"1993"(058)=511.141
(439)355.1"1993"(058)=511.141
“1993"355.1(439)(058)=511.141
(058)355.1(439)"1993"=511.141
=511.141:(058)355.1(439) "1993"
355 (439)1"1993"(058)=511.141

The identical inner structures means that the parsing of the numbers above will always produce the same result:

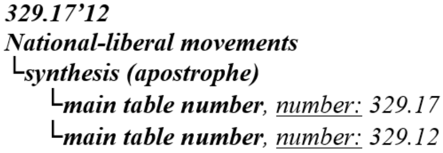


```
355.1(439)"1993"(058)=511.141
Yearbook of Hungarian Defense Force 1993 in Hungarian
└ main table number, number: 355.1 Armed forces
        └ common auxiliary of place, number: 439 Hungary
        └ common auxiliary of time, number: 1993
        └ common auxiliary of form, number: 058 yearbooks
        └ common auxiliary of language, number:: =511.141 Hungarian
```

The final example demonstrates the way of differentiating concepts mentioned previously.



329.17:329.12
Relations of national and liberal movements
└ ***relation***
  └***main table number***, *number: 329.17*
  └***main table number***, *number: 329.12*



329.17'12
National-liberal movements
└***synthesis (apostrophe)***
  └***main table number***, *number: 329.17*
  └***main table number***, *number: 329.12*

4. Conclusions

Following research conducted in the Seventies, in the main the two basic solutions mentioned above have been spread in OPACs to use UDC to retrieve contents, despite their incompleteness. In the late Nineties and the early part of this millennium, the concept of using authority files became the highly recommended way of utilizing classification systems in modern information retrieval software; this solution enhances the accuracy of indexing, cost efficiency of the system and user friendliness of the interface [12].

However, further improvements can result in more increase in the effectiveness of bibliographic analysis and retrieving information by taking advantage of analitico-synthetic and faceted characteristics (or even the linguistic nature) of UDC.

My goal was to support these improvements when I decided to publish the principles above and to implement the prototype of the software interpreter.

Acknowledgements. I would like to express my appreciation to Dr. Habil. István Boda for his valuable and constructive suggestions during the planning and development of this research work.

Advice given by Mr. Jonathan Wild has been of great help in writing this paper.

I would like to acknowledge the support provided by my family during my work too.

References

[1] HAJDU BARÁT, Á., Usability and Responsibility, Extensions & Corrections to UDC, Vol.28 (2006), 46-55.

[2] HAJDU BARÁT, Á., Knowledge organization of the Universal Decimal Classification – new solutions, Extensions & Corrections to UDC, Vol.26 (2004), 7-12.

[3] RAYWARD, W. B., “International Institute of Sociological Bibliography”. Encyclopedia of Library History. New York: Garland Press,1994. 290-294.

[4] UDC CONSORTIUM, UDC History, `http://udcc.org/index.php/site/page?view=about_history` (accessed on April 16, 2014)

[5] GNOLI, C., Facets in UDC: a review of current situation, Extensions & Corrections to UDC, Vol. 33 (2011), 19-36., `http://arizona.openrepository.com/arizona/handle/10150/236511` (accessed on April 16, 2014)

[6] SLAVIC, A., Use of the Universal Decimal Classification: A World-Wide Survey, Journal of Documentation Vol. 64 (2008), iss. 2, 211-228.

[7] SLAVIC, A., UDC libraries in the world - 2012 study, Universal Decimal Classification Blog (blog), August 20, 2012., `http://universaldecimalclassification.blogspot.hu/2012/08/udc-libraries-in-world-2012-study.html` (accessed on April 3, 2014)

[8] UDC CONSORTIUM, Collections indexed by UDC, `http://udcc.org/index.php/site/page?view=collections` (accessed on April 16, 2014)

[9] RAYWARD, W. B., The UDC and FID - A Historical Perspective, The Library Quarterly, Vol. 37. (1967), iss. 3, 259-278.

[10] BHATTACHARYA, G, Vital role of depth classification in a system for document-finding: a trend report, Library Science with a slant to Documentation, Vol. 6 (1969), iss. 1, 52-70.

[11] RIGBY, MALCOLM. Computers and the UDC; A Decade of Progress 1963-1973. The Hague: International Federation for Documentation, 1974.

[12] SLAVIC, A., CORDEIRO, M. I., RIESTHUIS, G., Enhancement of UDC data for use and sharing in a networked environment, Paper based on the talk presented at The 31st Annual Conference of the German Classification Society on Data Analysis, Machine Learning, and Applications, March 7-9, 2007, Freiburg i. Br., Germany, `http://arizona.openrepository.com/arizona/handle/10150/106330` (accessed on April 16, 2014)

[13] LIBRARY OF CONGRESS NETWORK DEVELOPMENT AND MARC STANDARDS OFFICE, MARC 21 Format for Classification Data, `http://www.loc.gov/marc/classification/eccdhome.html` (accessed on April 16, 2014)

[14] INTERNATIONAL FEDERATION OF LIBRARY ASSOCIATIONS AND INSTITUTIONS, Concise UNIMARC Classification Format (20001031), `http://archive.ifla.org/VI/3/p1996-1/concise.htm` (accessed on April 16, 2014)

[15] SLAVIC, A., The level of exploitation of Universal Decimal Classification in library OPACs: a pilot study, Vjesnik bibliotekara Hrvatske, Vol. 49 (2006), iss. 3-4, 155-182. `http://arizona.openrepository.com/arizona/handle/10150/105346` (accessed on April 16, 2014)

[16] UDC CONSORTIUM, UDC Summary Linked Data: Common auxiliaries of place. Table 1e, `http://udcdata.info/001951` (accessed on April 16, 2014)

[17] GNOLI, C., SLAVIC, A., Developing facets in UDC for online retrieval, Presentation at the 8th NKOS Workshop, Corfu, October 1, 2009, `https://at-web1.comp.glam.ac.uk/pages/research/hypermedia/nkos/nkos2009/presentations/UDCFacets.ppt` (accessed on April 3, 2014)

[18] ORSZÁGOS SZÉCHÉNYI KÖNYVTÁR KÖNYVTÁRI INTÉZET, Egyetemes Tizedes Osztályozás [Universal Decimal Classification] : UDC Publ. No. P057, Budapest: Országos Széchényi Könyvtár Könyvtári Intézet, 2005

[19] UDC CONSORTIUM, Major changes to the UDC 1993-2013, `http://udcc.org/index.php/site/page?view=major_revisions` (accessed on April 3, 2014)