Three Pillars of Historical Wisdom: Atomization, Data Building and Flexibility

Jean-Pierre Dedieu

Three Pillars of Historical Wisdom: Atomization, Data Building and Flexibility
On historical databases for research

Jean Pierre Dedieu
CNRS / FRAMESPA / IAO (ENS-Lyon)

What follows is based on experience. Since 1988 we have been developing a database, named Actoz, about actors involved in the government of the Spanish monarchy in the XVIIIth century. Some sixty researchers (French, Chilean, Portuguese, Spanish, German and Italian ones) took part in the project at some moment and many are still actively working in it. I was for my part in charge of the computing side of the undertaking. Under pressure of its users, Actoz, evolved from a rather simple, not inefficient, tool for the study of a limited set of appointments to administrative positions, into an embracing and powerful system, able to cope with almost any kind of historical information[1]. This development went along with a reflection on what a database should be, a reflection conducted on a piecemeal and pragmatical basis. Every time that a new development was needed, we thought on how to implement it. Once implemented and tested, once we were sure it worked in a practical way, we considered its meaning and implications as to what we could pompously call the general theory of databases[2]. The present text collects such conclusions[3]. We surpassed in that way, at least so we believe, the strictly technical and, in our view, all-too limited scope of many of the best manuals in existence on the topic of databases[4]. Some evidence of success was provided by the fact that various other research programs - around twenty of them by now[5] -

1 On Actoz, see Dedieu (Jean Pierre), "Fichoz 2011. Balance de una base de datos sobre la España moderna", *Homenaje a Juan Luis Castellanos*, Granada, 2012, in print. Fichoz was elaborated by a group of researchers who worked on the political structures of the Spanish monarchy known as the PAPE. The PAPE published, among other titles: Dedieu (Jean Pierre), Castellanos (Juan Luis), dir., *Réseaux, familles et pouvoirs dans le monde ibérique à la fin de l'Ancien Régime*, Paris, CNRS Editions, 1998, 267 p. and Dedieu (Jean Pierre), Castellanos (Juan Luis), M.V. López Cordón (María Victoria), *La pluma, la mitra y la espada. Estudios de historia institucional en la época moderna*, Madrid, Marcial Pons, 2000, 365 p. Fichoz was elaborated in the following research centers, to which I successively belonged: the Maison des Pays Ibériques of Bordeaux (1988-2004), and the LARHRA (Laboratoire de Reherche Historique Rhône Alpes) in Lyon (2005-2013). Both of them were sponsored by the CNRS and I was acting as an agent of the CNRS. Moreover, we got funds from the following programs: a) PICS 124 (Dedieu), with the Early Modern History department of the University of Sevilla (1990-1993); b) a "Europe" program of the CNRS, with the Early Modern History Department of the University of Granada (1991-1996); c) three franco-spanish "Actions intégrées" PICASSO (French and Spanish Ministeries for Universities), with the INSADE team (M. V. López Cordón, Department of Early Modern History of the Complutense University) (1993-1995 and 2007-2010) and with the Department of Early Modern History of the University of Grenade (Juan Luis Castellanos, 1996-1998). Presently, Actoz is developed and maintained by FRAMESPA, a CNRS research center located in Toulouse-Le Mirail University; and the IAO (Institut d'Asie Orientale), of the ENS-Lyon.

2 When basing the construction of the database on a permanent dialogue between practical engineering concepts and theoretical considerations, we proceeded in the same way as, for instance, the designers of Prospero, a successful package for a sociological analysis of arguing techniques which social actors handle in conflicts and debates (Chateauraynaud (Francis), *Prospero. Une technologie littéraire pour les sciences humaines*, Paris, CNRS Editions, 2003, 403 p., *passim*). Such reflective pragmaticism is universally considered as the only efficient way to create the kind of software which really meets the needs of its users.

3 For a shorter version, stripped of many theoretical considerations and oriented to the needs of Fichoz users in demand of a brief introduction, see: Dedieu (Jean Pierre), A brief introduction to historical databases, xxx.

4 Caldeira (Carlos Pampulim), *A arte da bases de dados. Com exemplos de aplicação para Oracle e SQL Server*, Lisboa, Edições Sílabo, 2011, 253 p.; Harvey (Charles), Press (Jon) et alt., *Databases in Historical Research: Theory, Methods and Applications*, Basingstoke, Macmillan, 1996, XVI + 352 p.; Pinol (Jean Luc), Zysberg (André), *Le métier d'historien avec un ordinateur*, Paris, Fernand Nathan, 1995; Cocaud (Martine), Cellier (Jacques), *Le traitement des données en histoire et sciences sociales*, Rennes, Presses Universitaires de Rennes, 2012, 554 p.

5 A short list would include: ACTOZ, on political and social features, personal networks and relationships in the Spanish Monarchy, from the end of the XVIIth to the end of the XIXth centuries. Language: Spanish. Main researchers currently involved: Jean Pierre Dedieu, Andoni Artola, Alvaro Chaparro, Francisco Andújar (Universidad de Almería), José María Imizcoz (Universidad del País Vasco), María Victoria López Cordón (Universidad

sought our help to create their own database on the most various matters. These new implementations gave birth to a family of databases, the Fichoz[6] family. All of them work along the same lines and are in many respects standardized, in such a way that almost all the items which compose one of them are identical to those which compose others; although we shall see that each of them comprises specific parts designed to meet the requirements of specific sources.

The first section of the present text describes the most important parameters which, in our view, determine the extent of the problem of building databases for historical research. A second section exposes the main concepts we put at play to solve that problem. We describe in a third part a set of technical solutions which allowed us to overcome practically some difficulties which, for a long time, were perceived as serious obstacles to the development of historical databases, such as the variability of names and the fuzziness of dates. A fourth section more specifically explains the general organization of data tables in Fichoz. Nevertheless, describing Fichoz is not the aim we are fundamentally pursuing there. This last part must be read as an example of how to implement the solutions previously suggested rather than a treatise on a specific database. We just intend to show how to make real the broader principles on which Fichoz is based. We try and keep a balance between theory and practice. In our view, this is a necessary condition in order to contribute some kind of solution to what we see as the greatest challenge facing by now our scientific community, namely the introduction of computing as a basic tool for historical studies.

The scope of this revolution could be regarded as debatable in the 80s of the last century, and was probably not fully perceived then, even by the most far-seeing supporters of the computer - by themselves a minority - which rallied around the journal *History and Computing*[7]. It cannot be

---

Complutense). In course of implementation on the Web. AIR, a general database on French aircraft industry, from the origins to present day (firms, planes, personnel, etc.). Manager: Jean Marc Olivier, Framespa, Toulouse. Language: French. ANCIENT HISTORY, a version designed to meet specific needs derived from the nature of the sources used by Ancient History. A project launched by Cyrille Courrier (ENS Lyon/EFR), with a view at processing Pompei political *scripta*, later developed with Bertrand Augier (PhD candidate), to process data on roman army officers of the 1st century. ARACHNE, a special implementation of Fichoz created to process tapestries, tapestry making and tapestry museography. Manager: Pascal Bertrand (Université de Bordeaux III, Histoire de l'art).; deputy managers: Stéphanie Trouvé and Elsa Karsallah. Language: French. FAR EAST, on Chinese early modern history and scientific relationships between China and Europe in early modern times. Manager: Catherine Jami (CNRS), with half a dozen of French and English colleagues. Languages: English and Chinese. CHINA, a database on modern Chinese history, developed at by the IAO center (ENS-Lyon), with various scores of Chines colleagues. Manager Christian Henriot, languages: English and Chinese. NAVIGOCORPUS, a database on shipping from the middle of the XVIIth to the middle of the XIXth century, giving a detailed account of every recorded travel made by any ship (places, cargoes, etc.). Managers: Silvia Marzagalli (University of Nice), Pierrick Pourchasse (University of Brest) and Jean Pierre Dedieu (CNRS). Language: English. , because of the complexity of the data, is probably the most interesting piece of work we ever realized. See: Dedieu (Jean Pierre), Marzagalli (Silvia), Pourchasse (Pierrick), Scheltens (Werner), "A technical introduction to Navigocorpus - A database for shipping information", *International Journal of Maritime History*, 2011, XXXIII/2, 12/2011, p. 241-262). POLITICS, on political actors in XIXth century Europe and America, focusing on the ideological side of the question: how were the liberal and the reactionary currents organized at international level? Managers: Jean Philippe Louis (University of Clermont Ferrand), and various PhD candidates or post doctoral students of France, Spain and Italy. Language: still to be decided. Probably multi-lingual. TUNISIE: a database dedicated to the study of colonial Tunisia; a private venture of Jean Pierre Dedieu. Language: French. WAQF: a specific implementation to store data on Islamic religious foundations in the whole of Muslim world. Manager: Randi Deguilhem and Mohammadreza Neyestani, with an international team which includes researchers from Japan, Turkey, Palestina, Tunisia, Algeria and various Gulf countries. The CHARLEVILLE project, in demographic history. Manager: François Joseph Riggiu and Carole Rathier (See: Rathier (Carole), Ruggiu (François Joseph), "La population de Charleville de la fin du XVIIe siècle à la fin du XIXe sièlce. Une enquête de démographie historique", *Histoire et Mesure*, 2013, XXVIII/2, p. 3-128).

6 Fichoz stands for FICHier OZanam, after the name of the historian who conceived the project which gave birth to the database. This name was given to the first implementation we created, the one we call now Actoz. It was later extended to the whole system, as new databases were being created in accordance to the same principles.

7 Due to travel delays, I missed by one day the foundation of the History and Computing Association in London, in

doubted nowadays that informatics change all the scales along which we used to evaluate our capacity of handling information, the scales on which were grounded in the XIXth century the basic guidelines for historical research, the rules which determined what could be and could not be received as scientifically valid in history. Change goes far deeper than data mining on the web or the automatized drawing of maps and charts, the kind of topics which seems currently to keep busy specialists of e-humanities[8]. To put it squarely, the introduction of computers in history plays the same part as the introduction of the telescope in astronomy or of the microscope in biology. Nothing can be the same after.

We must invent new ways of doing things which take into account so momentous a change. New ways do not mean throwing ancient tools overboard. Renouncing rigor, documentary critic and erudition is out of question. The rules of historical hermeneutics which our forefathers codified are still valid, and play a central part in our view of what historical computing might be, a point which we shall stress all along. Computers do not make historical research easier, nor do they provide laymen with a smooth access to science. We firmly believe that the rules established by our German predecessors at the end of the XIXth century hold true, exactly as they held true and never were renounced when the "Annales-school" history of Febvre, Bloch and Braudel expanded in a tremendous way the scope of historical research[9]. The question is how to define anew proceedings and procedures, how to invent a new way of managing information so as to make the best use of the versatility, of the fantastic volume of data, of the unprecedented capacities of collaboration computers put at our disposal, without loosing in terms of rigor and rightly-conducted interpretative capacity[10].

The problem lies in the media. Computing means rigidity. The basic operations of the machine are based on the endless repetition of identical sequences. On the contrary, the first and unconditional need of the researcher is flexibility, a continuous and close adaptation to ceaselessly changing sources, to unpredictable variations in formulations, wordings and concepts. To work correctly, computers need to be previously equipped with a structured description of the information they process. The kind of description that a researcher cannot provide beforehand, just because reaching it is precisely the aim of the research for which the computer is needed. The problem can be solved by injecting into basic data an artificial structure, by curtailing what information does not fit pre-established models. Such a solution may be considered satisfactory in an administrative world, among other reasons because curtailing data to make them fit a previously defined model in order to increase efficiency of processing routines is the essence of administration. As far as research is concerned, it is clearly inadequate. It would mean loosing the heart of the matter, the unpredictable

---

1986. I organized in Bordeaux the World Congress of the same in 1989 and published its proceedings (Dedieu (Jean Pierre), coord.: L'*ordinateur et le métier d'historien. Actes du Congrès de l'International Association for History and Computing*, Maison des Pays Ibériques, Talence, 1991, 250 p.). On the debate about the role of informatics which was then on the run, see: Genet (Jean Philippe), *Standardisation et échange des bases de données historiques*, Paris, CNRS, 1988, 380 p.

8 Trinkle (Dennis A.) (ed), *Writing, Teaching and Researching History in the Electronic Age: Historians and the Computer*, Armonk / New York, Sharpe, 1998, XIII + 267 p.; Bodenhamer (David J.), Corrigan (John), Harris (Trevor M.), ed., *The spatial humanities. GIS and the future of Humanities scholarship*, Bloomington and Indianapolis, Indiana University Press, 2010, 203 p.; Burdick (Anne), Drucker (Johanna), Lunefeld (Peter), Presner (Todd), Schapp (Jeffrey), *Digital_humanities*, Cambridge (Massachusetts), Massachusetts Institute of Technologie, 2012, 141 p. See also the collection of the *International Journal of Humanities and Arts Computing*, Edinburgh, Edinburgh UP, from 2007 on.

9 The most emblematic book of the Annales revolution, Bloch's *Rois thaumaturges*, is a good example of this kind of continuity (*Les rois thaumaturges. Etude sur le caractère surnaturel attribué aux personnes royales, particulièrement en France et en Angleterre*, 2ème ed., Armand Colin, Paris, 1961 [1923], VIII + 544 p.).

10 Dubucs (Jacques), "Digital Humanities. Foundations", Davidhazi (Peter), ed., *Exploring a Paradigm shift. New Publication Cultures in Humanities*, Amsterdam, Amsterdam University Press, forthcoming (philpapers.org/rec/DUBDHF, 30 June 2013), expresses similar views extended to humanities in general.

element from which discovery arises. We must find other ways of doing things.

I. A database, what for?

Historical research uses computing systems in three ways:

a) as a tool to access information.
b) as a tool for the making of databases.
c) as a tool for data analysis and modeling.

a) Accessing information

The first heading covers any device contrived to access data to be found on the Web or in any non-web-connected data deposit. Such tools are all-important for historians. In the last twenty years, they changed in depth our way of doing history. They include:

. Bibliographical tools, ranging from catalogs of the main public libraries[11] to databases of digitalized ancient texts[12], which make possible in question of minutes at a cost of some cents of Euros inquiries which, twenty years ago, meant traveling to foreign countries and visiting the ancient books section of local institutions. Some of them were charming places. But most of those who had to travel under such conditions nevertheless appreciate the change.

. Devices to access scientific papers, by means of specialized databases and portals. Various firms provide paying access to a wide range of scientific reviews[13]. Researchers more and more publish their conclusions (even provisional conclusions) on the Web, either privately or as part of official or semi-official ventures, on university portals, on personal pages, on specialized systems such as Dialnet in Spain or Hal-shs in France[14].

. Devices to access archive documents. This is one of the most interesting recent developments for the historian: a direct access to sources from his office desk[15]. We shall comment this point with some detail further.

. An access to databases of general knowledge, the role of which is similar to that of the printed encyclopedia and reference works we used before the computer era. Printed encyclopedia varied in many aspects, ranging from generic works for a broad audience (*Larousse encyclopedia*, *Encyclopedia Britannica* for instance) to specialized high level research tools, such as the l'*Encyclopédie de l'Islam* or Mac Millan's *Encyclopedia of Social Sciences*[16]. Databases of general knowledge to be found on the Web cover the same range.

11 Some instances: catalog of the Biblioteca Nacional de Madrid, http://catalogo.bne.es; catalog of the Bibliothèque Nationale de France, http://catalogue.bnf.fr, for instance. Many more can be easily be found. An on-line catalog is, by now, a standard status marker for any international, national and even local library.

12 Biblioteca digital hispánica, http://www.bne.es/fr/Catalogos/BibliotecaDigital and the Hemeroteca digital, for newspapers and reviews, http://hemerotecadigital.bne.es; the open library of hispanic texts of the Hathi Trust digital library, http://www.hathitrust.org; Gallica, the French database of digitalized ancient texts, gallica.bnf.fr/, etc.

13 Among the best known, JSTOR, for international journals, www.jstor.org; or CAIRN for French journals, www.cairn.info. Fees are nevertheless a frequent drawback. Such portals are usually accessed through public libraries which subscribe on behalf of their readers.

14 Dialnet, a Spanish database of historical papers and books on social sciences and humanities, which at first only provided bibliographical references and which more and more frequently publishes electronic full text versions, http://dialnet.unirioja.es/; Hal-shs, a French database of (mainly) full-text publications on social sciences and humanities, halshs.archives-ouvertes.fr. Spanish universities systematically put their PhDs on the Net. The European Community is strongly encouraging European journals to put their content free on the net on short delay.

15 Spain was a pioneer in this field. It published on-line the most consulted pieces of the Archivo de Indias so early as 1992. Later it created the "PARES, portal de archivos españoles" system which extended the experience to other deposits. Around 2005, the government had huge projects in mind. Electoral changes and economic crisis disgracefully stopped such plans, and Pares badly needs a serious overhaul (http://pares.mcu.es/). All the main State archives in the world and many private ones have programs to put on-line their most consulted pieces. Benefits are mutual: historians don't need to travel; documents are no longer moved and handled, and in that way better preserved.

We shall not dwell on the obvious interest of such tools[17]. We shall only observe that they provide a local knowledge[18] which could not be accessed before; and that they increase in that way the density of the contextual background against which researchers are able to set their data. So that Internet devices don't make research easier. They make it more complex as they oblige researchers to integrate more data than they did before; when rightly conducted, they also make it far better.

The Web, moreover, makes easier a material access to the stuff, but material access only. It does not solve associated cognitive problems, such as selecting relevant information among a wealth of references, understanding its meaning and evaluating its reliability. In some respects, it even makes things worse by eliminating external pointers which indicated the scientific level and possible uses of any piece of information, such as being stored in the reference books section of a specialized library or the mere fact that a scientific committee and / or a commercial publisher considered that it deserved publication. We are afraid that, as far as research is concerned, computers will never be able to provide efficient help on that point. Computers are able learn how to do complex tasks. But they learn by repeating successful past experience; while research means breaking new ground.

Formulating queries to find relevant data is made more difficult by the Web page setting, which breaks the informative continuity of the printed book. To find information, Net-users are now bound to imagine the words which describe the object in the data base. Passivity is the most expedient way to make a mess of a query on the Web. Users must be alert and creative. They must think of a way of formulating queries which will bring forth results by setting themselves in the author's shoes, trying to figure out the way he would himself word the question they are asking. They must accept perusing pages after pages of useless material to find at last the gem they are looking for. If it does not work one way, try another. Beware of overfeeding: web-users are prone to it. To fight it, put in practice old and long-tested techniques for bibliographical queries: don't try and get three scores of references; you'll never read them. Find the latest publication, see the works it quotes; find them, read endnotes, follow the string of mutual quotations, and read the items which play a central role in this network. You'll quickly know all you need to know on the topic your are interested in.

b) Analytical tools

So much for computers used as a source of information. Literature on the third heading, computers and computer programs as a tool for data analysis, is almost as abundant as that on this first topic. Experience showed that, up to now, the most useful analytical tools for historians are:

a) Data sheets (Excel and similar[19] are very efficient and rather user-friendly) and, if

16 Larousse (Pierre), dir., *Grand dictionnaire universel...*, Paris, Administration du "Grand dictionnaire universel", , 1865-1890, 17 vol. in-fol.; *Encyclopedia Britannica. A New Survey of Universal Knowledge*, Londres / Chicago / Toronto, Encyclopedia Britannica, 1959, 26 vol.; Sills (David L.), ed., *International Encyclopedia of the Social Sciences*, Mac Millan, 1968, 17 vol.; *Encyclopédie de l'Islam, nouvelle édition établie avec le concours des principaux orientalistes*, Leiden, E. J. Brill / Paris, G. P. Maisonneuve et Larose, 1960-2009.

17 And yet, we are tempted to. We are presently setting in their family context all Spanish XVIIIth Century captains-general. The same work has recently been done by Didier Ozanam, the man who indirectly gave birth to Fichoz and a kind of virtuoso of erudition; but who only handled books. We greatly enhance the quality of his work. The difference lies in the Web and the access it provides to an amount of literature Ozanam could not even dream of.

18 By local knowledge, we mean a detailed knowledge on such and such a topographical object (Argés, for instance, a tiny village in Spain, on which you might need information to understand a document). We also mean a specialized knowledge "in depth" of a determined topic. We found the contribution of the web especially striking when elaborating a general classification of commodities as part of research program (see note 5). Many products, denominations and technical processes which specialized encyclopedia ignore are there described by retired craftsmen who do not want such professional knowledge to die with them.

19 Excel® is a module of Office suite (http://office.microsoft.com/fr-fr/home-and-student/suite-microsoft-office). OpenOffice, a free-ware office tool, also provides an excellent service (www.openoffice.org/fr/).

necessary, specialized statistical packages (Orange canvas[20] and similar).

b) Mapping tools, and more generally tools for mathematical spatial analysis, so as to give data interpretation a spatial dimension[21].

c) Social networks[22] analysis tools. We personally use Pajek[23], which we found highly practical and flexible enough for historical research. But other good packages exist.

d) Linguistic analysis tools. This is a point which we shall later consider with some detail, given its bearing on conceptual issues.

At the end of this short presentation, we must emphatically insist on the fact that such tools do NOT produce conclusions. They just display data in another way, showing articulations of the same which direct observation of raw information could not clearly detect. It is up to the researcher to use their results, or discard them, and to integrate their contribution to his own conclusions. The mere publication of the results they provide without further elaboration is not science, because it explains nothing. It is just description. The main and most important analytical tool is and will ever be the researcher's brain. A point, by the way, which we'll have to take into account when designing a database.

c) The database: a link between raw information and analytical tools.

The so-called "digital humanities" show a huge interest for the two points we discussed till now. Curiously, most publications simply ignore our last heading, the database[24]. Databases come third in our exposition, but in fact are second in a logical order of things. The string of operations to be executed by any computer-based historical research can be summarized in this way:

Fig. I. Databases in the context of computer-based research

Source of information	==>	Database	==>	Analytical tools
Raw information			Structured information	

What do sources of information provide, in fact, be they computerized or not? Raw, global, information computers are unable to process.

Let us have a look at the first paragraph of the English Wikipedia entry on Charles Leclerc[25], a French general of the Revolution:

20 http://orange.biolab.si (freeware).

21 The absolute reference on that point is ArcGis (www.arcgis.com). It executes almostevery possible task. A high price and complexity are two serious drawbacks. I personally use Cartes&données, a French package for spatial analysis, the basic modules of which are free of charges (http://www.articque.com/). Electronic mapping, by the way, is a rather simple matter, once you get the relevant mapframes and identified all the points to be pictured on the map. Both requisites may be a fairly complex matters when historical documents are involved, but such intricacy does not derive from the computing side of the business.

22 "Social network" does not allude here to Facebook and similar, but to the study of relationships which actors maintain with one another and use as a tool for action. The main reference, in English, is: Wasserman (Stanley), Faust (Katherine), *Social network analysis: methods and applications*, Cambridge, Cambridge University Press, 1994, XXXI + 825 p.

23 De Nooy (Wouter), Mrvar (Andrej) et Batagelj (Vladimir), *Exploratory social analysis with Pajek*, Cambridge, Cambridge UP, 2005 [2002], 334 p., which apart from describing Pajek package, is an excellent introduction to network algebra. Pajek itself is free-ware, to be found at: **p**ajek.imfm.si/doku.php?id=download. Passing data from a database to Pajek means reformatting them and demands an interface program, rather easy to write, but nevertheless a work to be done by specialists.

24 Burdick (Anne) et alt., *Digital_humanities*, *op. cit*., absolutely silences such matters in spite of its claim to exhaustivity.

25 Wikimedia, *Wikipedia. L'Encyclopédie libre*, c. 2000 - 2009, http://fr.wikipedia.org, Leclerc, Charles, English version. Consulted on 20-03-2013.

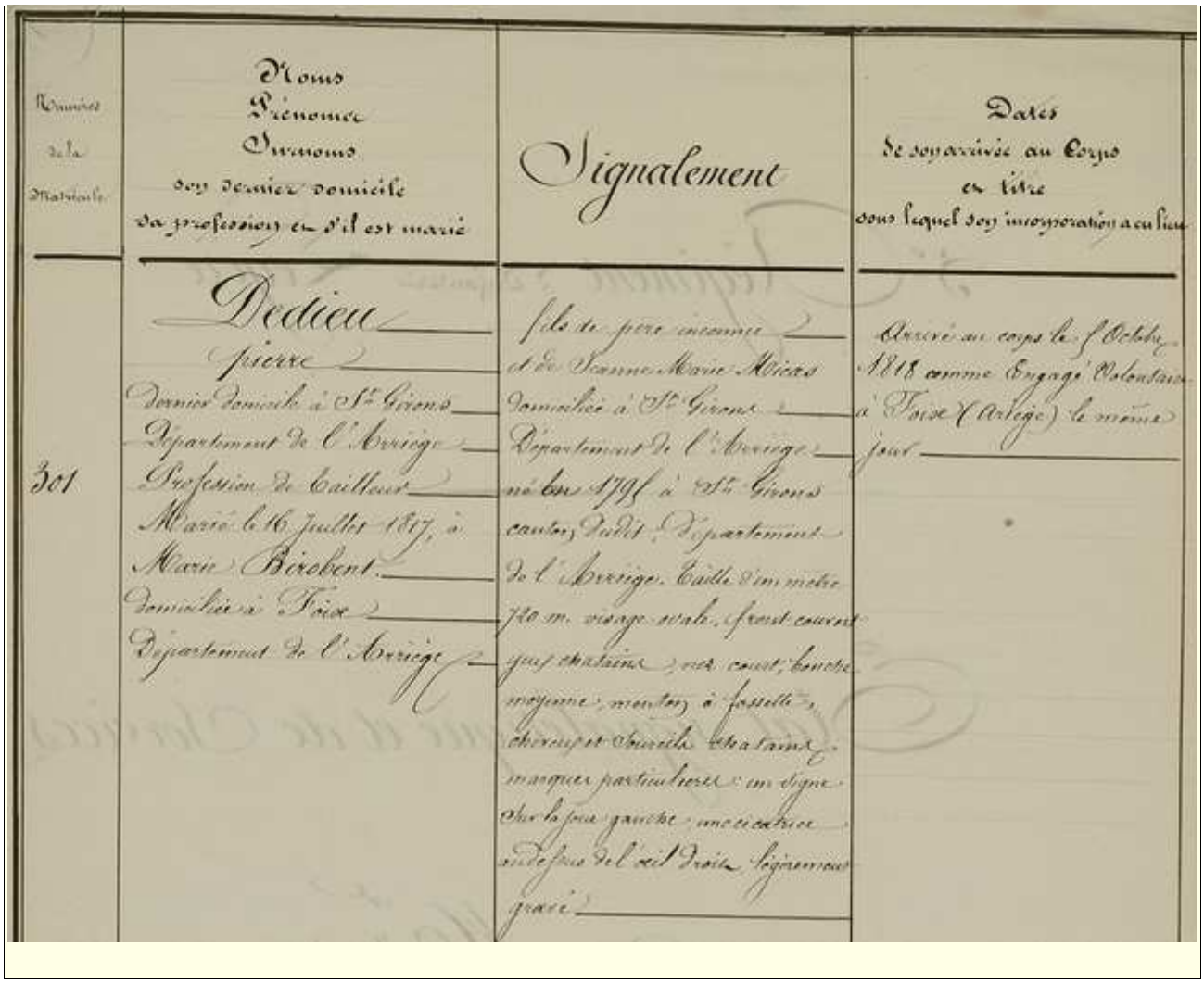
Example I

> "Charles Victoire Emmanuel Leclerc (17 March 1772, Pontoise – 2 November 1802) was a French Army general and husband to Pauline Bonaparte, sister to Napoleon Bonaparte.

Our second example is drawn from the Legion d'honneur files published on the Web by the French government. It is the first page of the service record sheet of Pierre Dedieu, a French soldier of the first half of XIXth century[26]:

Example II. Pierre Dedieu's service record sheet

26 http://www.culture.gouv.fr/LH/LH063/PG/FRDAFAN83_OL0685089V005.htm Cons. 20-03-2013. This Pierre Dedieu has nothing to do with the author of the present paper, except his birthplace.



Numéro de la Matricule	Noms Prénomes Surnoms des Dernier domicile de profession et s'il est marié	Signalement	Dates de son arrivée au Corps en titre sous lequel son inscription a eu lieu
301	Dedieu Pierre Dernier Domicile à St. Girons Département de l'Ariège Profession de Cultivateur Marié le 16 Juillet 1817, à Marie Birobent Domicilié à Foix Département de l'Ariège	fils de père inconnu et de Jeanne Marie Micas Domiciliée à St. Girons Département de l'Ariège né en 1795 à St. Girons canton, Distt. Département de l'Ariège. Taille 5 im mètre 760 m. visage ovale, front couvert, yeux chataine, nez court, bouche moyenne, menton à fossette, cheveux et sourcils chataine, marques particulières un signe sur la joue gauche une cicatrice au dessus de l'œil Droit, légèrement gravé	Arrivé au corps le 1 Octobre 1818 comme Engagé Volontaire à Foix (Ariège) le même jour

What can computerized analytical tools do with such documents? Strictly nothing. Let the machine find the birthdate of these persons, not too-hard a task. Apparently. The problem is to locate the data within the document. We know, in accordance to generally accepted typographical conventions, that in a biographical entry, the birthdate is the first item of the text between bracket which immediately follows the name, and in that way we find Leclerc's birthdate. As for Dedieu, we must read the record to find what we are looking for on the fifth line of the second column; and if you do not understand French, the worse for you!

How could the computer manage it? In one of three ways:

a) We teach the machine all we know about the structure and conventional paging of documents, so as to make it able to read and fully understand them without human help. It is not an absurd goal. Hundreds of engineers are working on similar problems just now in view of industrial implementations, and they are progressing fast on this line. But they are still far away from a global solution and the partial answers they got so far, which rely heavily on the reproduction of previously successful solutions, work far better when the context can be predicted than when not. And as far as research is concerned, the context is essentially unpredictable. Understanding historical documents was recognized of old as so tricky a task, that historians elaborated an impressive set of rules to govern historical hermeneutics, a full understanding of which is (or was at least when I was a student) the main point in learning the job. With the aggravating circumstance that such rules are all you want them to be, except mechanical receipts to be blindly enforced. You must take them into account all of them together, as a whole, and determine what to do with them in function of the

actual context of the document you are studying, in a process which looks more like intuition than like rational inference. To put it bluntly, we don't have yet any tool to make the computer able to segment historical documents into historical data, and the possibility a creating one efficient and secure enough to provide hard data on which to build historical scientific conclusions is still debatable. The first way of making the computer work for us is closed and will probably remain so for some time.

b) The second way consists in letting the historian do the job, by splitting the document into as many homogeneous pieces of information as needed to make the computer's analytical tools work; and placing such pieces into a set of pigeon holes, each of them specialized and containing one kind, and only one kind, of information. Programming the computer to find one class of items or another is then a rather simple task. We shall discuss further that point, because we believe that it is by now the most efficient way of managing historical research databases. The main drawback is that by splitting a document, one destroys it. The information inside is preserved. The form is lost. Exactly the same as when one is eating nuts. One has to break the shell to get the fruit. If the shell does not matter, if the form of the document does not convey information, never mind it. If it does, this second way is impracticable.

c) There remains a third way, which we shall also consider further with some detail. It preserves the form of the document, be it text, graphics (a painting, an engraving, plans of a church or of a palace, an aerial view, etc.) or sounds (music, a recorded interview, for instance). It just inserts into the documents markers, or labels, or tags, the meaning of which the computer has been programmed to understand. Any segment contained between two given tags is marked as containing a class of information. We might decide, for instance that the expression:

<Bdb> 17 March 1772<Bde /> <Bpb>Pontoise<Bpe />

defines Leclerc's birthdate and birhplace. In such a way, we combine computer database efficiency with a strict preservation of the document. The main problem with this method is its highly cumbersome character. Intends were made to apply it full-scale to all kinds of historical data[27]. They crumbled under their own weight. Such a process is nevertheless necessary, we must insist on this point, when preserving the form of the document is in itself necessary. Tools for earmarking texts in such a way spectacularly improved in recent years[28]. For reasons independent from any circumstantial state of the art, reasons which I shall express further, I nevertheless believe that their use must be restricted to specific contexts.

The semantic web is a generalization of the tag strategy. It works in the way we just described, but changes arbitrary tags into unique resource identifiers (URI), which makes them accessible wherever they are located on any part of the Web. It also combines markers into ontologies, which describe not only their individual meaning, but also rules which make structured search possible. Retrieving data related to a province, for instance, may also mean, if such a rule is set, retrieving at the same time data on all the cities which belong to this province. These are highly useful features, which allow queries based on concepts and not only on ambiguous character strings; queries on dispersed sets of data the location of which is unknown to the user. But they do not change anything

27 See for instance Manfred Thaler's Kleio in the 80s of the last century (Thaller (Manfred), *Kleio. A data base system for historical research. Version 1.1.1, b-test Version*, Göttingen, Max-Planck-Institut für Geschichte, 1987, 127 p.).

28 The Text encoding initiative did a great job that way (http://www.tei-c.org/index.xml, consulted 20-03-2013). The HTML language belongs to this family. It is specialized in formal aspects of text typography, a field in which the short number of issues to be addressed makes possible universal conventions.

as to the cumbersome and verbose character, nor as to the other drawbacks of the proceeding[29].

Databases serve to store and retrieve information, the same as any other computerized tool used to handle data. But they also assume a task that no other data handling tool assumes: they transform information in such a way as to allow analytical tools to work on it. In other words, they transform information into data. In doing so, they must preserve the scientific quality of this information; and for that, they work under heavy constraints, a point which we shall now briefly discuss.

Transforming information into data is not a specific task of historical or scientific databases. Everyone, when using a computer, must tailor basic information into machine-readable homogeneous blocks. Exactly the same as when you write, you must graphically split your text into words and paragraphs; or when you draw an array you must assign a specific meaning to every cell of the same. When information has been gathered for administrative purposes, the task is fairly simple. Administrative processes in fact, with or without computers, are based on reducing information to smaller homogeneous blocks, which contain what is needed for administrative purposes, and nothing more. This is a point we must further stress to make clear, by contrast, the idiosyncracies of scientific information. French army knows me as:

> Male (field of the database: gender; value 1)
> Born 1948 (field: year; value 48)
> In August (field: month; value 08)
> In the department of Ariège (field: department; value 09)
> In a village called Prat-Bonrepaux (field: municipality within the department; value: 235)
> Registered as number 8 in the corresponding municipal roll (field: roll number; value 008)

That is: 1480809235008.

This was enough to call me to files if needed. The National Health Service, by the way, also uses to contact me through that same code. They know beforehand exactly what they need to know and do no look for more. The fact that I am a doctor in history, a respected (so I hope) researcher, the father of two lovely women and the happy grand-father of a couple of charming little brats does not matter. And justly so.

For social historians, it matters. They need to know all that, and a lot more. Creating beforehand a model, fitting the data into that model and cutting away what does not fit, is decidedly NOT a good way of doing history. Historians must preserve everything, even what they did not expect, above all what they did not expect, because unpredictability means further information. Creating a database for research purposes means elaborating not only on complex patterns, but most of all on unpredictable patterns. Researchers do not know what they need. An engineer does (better you do not experiment to much when building a bridge!). A researcher who knows beforehand where the solution lays and the paces he needs to reach it, has already achieved his goal. As a consequence, if needs cannot be defined beforehand, a scientific database cannot be oriented toward a definite goal, nor tailored to store information for this one goal only.

From there we conclude that flexibility and a capacity to cope with the most unpredictable situations, patterns and contents, is an essential feature of scientific databases. This demand for flexibility means that a correctly thought out database is by no means limited to the research it has been planned for. Being able to face diversity arising from changing working hypothesis elaborated by a same researcher is a fundamental characteristics of any efficient scientific database. But being

---

29 A list of good examples of semantic-web implementations should include ExaleadCloud (see http://fr.3ds.exalead.com/software/products/cloudview/cloudview-360-edition/), or, on a less ambitious scale, FAO's bibliographical databases (http://agris.fao.org/). Domingue (John), Fensel (Dieter), Hendlyer (James A.), ed., *Handbook of Semantic Web technologies,* Berlin / Heidelberg, Springer-Verlag, 2011, 2 vol., 1057 p., is an up-to-date study of the topic.

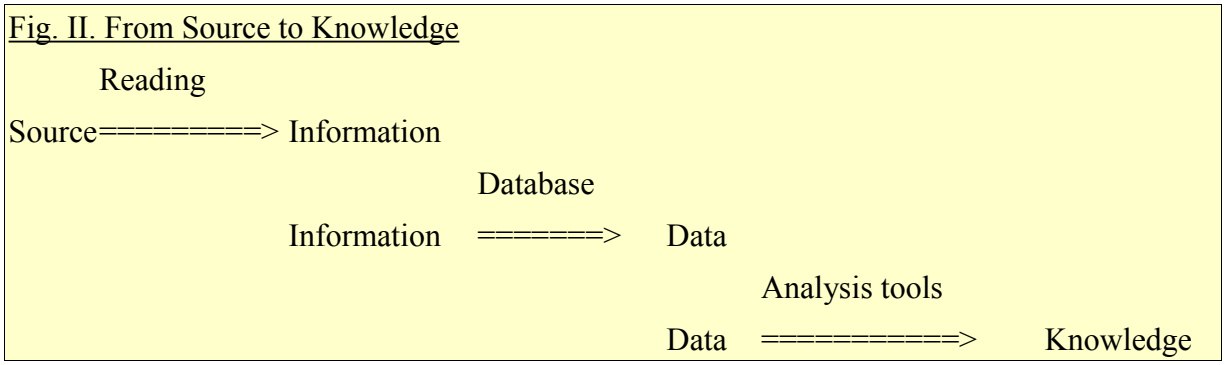
able to do so also means being able to face diversity arising from the fact that various researchers use the same data for various purposes, each one for a specific research. So that a well planned scientific database may and must potentially be used by various researchers. I shall comment this highly interesting observation in my conclusion.

*

* *

Summing up, sources do not provide knowledge. They provide raw material, from which knowledge can be produced. Sources and knowledge are both ends of a string of operations which transform the first into the second. Computerized databases, if computers are used in the process, insert themselves at some point of this string. The same would be true of any other electronic tool, such as data sheets, network analysis or statistical packages. The way in which electronic devices can and must be used is conditioned by the place they have been assigned to within this string of operations. We are now in condition to model this process in a somewhat more complete way as before:

Fig. II. From Source to Knowledge



Databases are the central part of a complex process. Their role goes far beyond a mere storage of information. They also break information into homogeneous blocks, all of them structured in the same way, each one equipped with a same set of descriptive labels to be found by the machine at a same known place, the content of which the computer can retrieve and pass to analytical packages, which in turn produce knowledge. Such blocks we call data. We call the process of breaking information into data atomization.

Atomization plays an central part in our vision of historical databases. It reduces the gap opened by the fact that the computer needs similar modules to work with, while historical information as well as users' expectations are characterized by variability and uncertainty. Atomization reduces information to blocks based on a perfectly uniform module, as computers need. When translating information into actions we create something similar to a mosaic, that is a fluid pattern composed of rigidly similar *tesserae*. By accumulating actions, that is small pieces of squared information, we draw complex patterns, or, better said, we let the computer draw complex patterns by arranging selected actions into a given order; an order which the researcher chooses in function of what his research's demands.

The kind of tools used to break information into data only matters marginally: any of the three ways we described here above achieves this goal. The kind of analytical tool set at the end of the information-to-data string does not matter either, be it a statistical package, a mapping-package, a word-processor or the user's mind. One point is all-important: the scale of the atomic data and the concept around which it must be arranged. Both must be chosen in such a way as to make possible the drawing of significant sketches without loosing any essential character of the original information. A question we will analyze with further detail in a second chapter, in the light of some specific characters of historical information.

d) Some disturbing characters of historical information

We already described the retrieval of structured pieces of information from any kind of documents as a difficult task. We must consider this point a little more in depth, to outline as best as we can some obstacles which a database specifically planned for historians must be able to suppress.

1) Identifying actors

Identifying actors, and assigning to the right actor the disconnected pieces of information which a variety of documents bring to light is one of the main problems which nowadays face historians. While history was fundamentally interested in kings, high aristocratic persons and ecclesiastical dignitaries, this was not really a pressing question, except for periods in which sources were so scant as to obscure this point also, namely, as far as Europe was concerned, Antiquity and Early Middle Ages. From late Middle Ages on, a handful of biographical dictionaries were all we needed to lift uncertainties[30]. Social history, as reshaped in the first half of the XXth century, enlarged our vision to other social circles, but did not really make the problem worse. Sources were in fact processed in an anonymous way, and entire books were published which practically mentioned no names. Individuals had no clearly assigned function, except that of being representatives of a group. Mac Farlane's study of parson Ralph Josselin's papers was hailed as a turning point towards a more personalized history, but in fact readers - may be Mac Farlane himself - were not really interested in Ralph Josselin's person, but in the material Josselin's diary contributed to the construction of the concept of Parson and of the concept of Family in XVIIth century England[31]. Moreover, when historians intended to reconstruct a social set in its entirety, as Mac Farlane himself later did[32], as Leroy Ladurie did[33], as I myself did[34], we dealt with some hundreds of actors, enclosed inside a limited chronological and geographical setting; a volume of information we could manage without the help of any tool, at least as far as identification was concerned. When I worked on Daimiel Moriscos, I handled data about around 500 persons. I could keep them all in mind fairly easily. The problem was the physical side of the question: getting, reading and putting again in alphabetical order dozens and dozens of cardboard cards, all that manually, was so time-consuming that I grew aware at that moment that I had to shift of the computer. But the problem was one of volume, not of identification. A couple of good research assistants would have done the job as well[35]. I needed the computer to handle blocks of information as blind boxes. Not to open the box and interpret its content.

Things changed when actors were considered as the heart of the matter, when they ceased to be illustrations of pervasive social forces, to become the makers of the same; when we saw them as agents who elaborated at every moment complex social artifacts and conventions, agents who competed to impose them on what looked like a competitive social market, and not simply puppets driven by collective conventions generated, outside human consciousness, by anonymous systemic forces. Moreover, computers broke the physical limits which hand-written cardboards imposed on

30 Each country had a standard one. In England, the *Who's who in History*, published by in Oxford by Basil and Blackwell was fairly used in the 1960s. In Spain the standard reference was Bleiberg (Germán), dir., *Diccionario de Historia de España*, Alianza, Madrid, 1979, 3 vol., with various reprints in quick succession.

31 Mac Farlane (Alan), ed., *The Diary of Ralph Josselin, 1616-1683*, Oxford University Press, London, 1976, 752 p. and the study attached to the same: Mac Farlane (Alan), *The Family Life of Ralph Josselin, a Seventeenth-Century Clergyman. An essay in Historical Anthropology*, Cambridge University Press, London 1970, 241 p.

32 Mac Farlane (Alan), *Reconstructing historical communities*, Cambridge University Press, London, 1977, 222 p.

33 Le Roy Ladurie (Emmanuel), *Montaillou, village occitan de 1294 à 1324*, Paris, NRF - Gallimard, 1975, 642 p.

34 Dedieu (Jean Pierre), "Les morisques de Daimiel et l'inquisition (1502-1612)", Cardaillac (Louis) (éd.), *Les morisques et leur temps*, Paris, CNRS, 1983, p. 496-522.

35 A persistent professional lore tells that Fernand Braudel and Pierre Chaunu freely used their respective wifes for the task... and that such a collaboration goes far at explaining the ground-breaking quality of their research.

the number of actors one could take into account at a same moment. We no longer could keep in mind who was who and what belonged to whom. Identifying actors became a problem.

We are now managing in Fichoz more than 150.000 persons. We are drawing information on every one of them not from one or two sources, as before, but in form of piecemeal fragments from a huge variety of different documents. In such documents, actors are mentioned in a great variety of contexts, and usually identified in a fragmentary and erratic manner. The expression "duke of Olivares", when mentioned around 1646, may refer to two persons: the famous Spanish Prime Minister, who died in 1645, or his son and successor to his title, although not to his political position. There are two counts of Luna in Spain, one in Castille, one in Aragon, and they are two absolutely different families; sources, of course, rarely indicate in an explicit way which of them is concerned. Father and son use to be named with the same quite unusual first names - families proudly cherish them as a rare social distinction -; they use to embrace the same careers, and sons use to cover the stages of this career somewhat faster than fathers did - fathers in fact use to favor and promote their sons. So that various people bearing a same name may fill the same position in rapid succession. If documentation is fragmentary and prevents any global view of the whole life course, confusions are unavoidable. Until new data, later to be discovered, allow correction.

The density of the information brought together by a huge database helps in fact to make this problem of identification less stringent than we figured at first. Fichoz users nevertheless spend much time moving bits of information from one actor to another; and displaying to the screen huge quantities of data to contrast dubious information by setting it on various backgrounds. All that means that one of the main requirements of a good historical database must be a capacity to change the attribution of any data from an actor to another; and a capacity to display jointly, with accurateness and agility, any given data and any elements of context. Such requirements have, as we shall see further, a fundamental impact on the choice of a model of database.

2) Data building: creating univocal and homogeneous data universes

Aggregating various sources always was a normal feature of historical research. Implicitly or explicitly a researcher always refers the information he is getting from a document to more information he got from other documents. Such a reference to the context is the basis of any scientific approach. This is the only way to determine the contribution of the current source to global knowledge and rightly to understand the biases which conditioned its elaboration.

In the pen and paper age, the various sources from which a researcher was drawing information combined themselves in the historian's mind. The historian himself read them, one by one; extracted them one by one, preserving what looked relevant and discarding the rest of it; combined notes extracted from various documents, formed provisional drafts which were the basis of the final writing of his work. The fact that he used elements of different kind and nature was no hindrance. A true researcher's mind was flexible enough to combine them all. He was able intuitively to manage, for instance, at the same time a picture, a text and a statistical array and to draw conclusions based on the three of them at the same time.

Such an ability for synthesis is still required. It is even more necessary than ever, as computers put at our disposal unheard-of volumes of data. But precisely because of the wealth of data provided, we need devices able to do for us as big a part of the job as possible if we want to preserve a capacity to supervise the process; that is to preserve part of our cognitive capacity for management, and not for a mere assimilation of new data. Helping us to do so is the function of the electronic analytical packages we mentioned before. The problem with them is that they are utterly unable to make a synthesis based on various classes of documents. They need homogeneous universes of data to work decently[36]. A good database must then create datasets equipped with two specific characters:

36 A fact which is not obvious for users with little training in informatics. Computers look sometimes as if the were

A. Univocity.

Every atom of data provided to the analytical package must be unique. No two different atoms of data must carry the same information. A same atom must carry all the information the analysis package needs to work on it. This means that the kind of database we aim at, must not put sources side by side and let the user choose, but extract from all the sources a unique piece of information with which analytical packages will be able to work. This unique piece of information must be built by the operator from various sources with the help of the rules of historical hermeneutics; and later transcribed to the database as a record.

Enforcing the rules of historical hermeneutics to achieve univocity is the user's job, not the database's, so that the task of extracting and interpreting data from the source is external to the database. Results, and only results of this process, are loaded to the same. Historical information, disgracefully, is conditioned by factors which blur limits between source and data. A piece of information is not usually given once for all by a unique document. It uses in fact to be progressively brought to light by a sequential reading of various sources. A same historical data uses to be progressively uncovered over time. The database must be planned in such a way as to make this progressive building of a same data possible.

Example III

Ship travels (Shipping databases). Port registers are the most important source. Some record arrivals, others departures. What kind of information does a departure register contain[37]? The fact that Ship N left port P1 on such a date, bound for port P2. On destination, that is at P2, an arrival register gives the reverse information: ship N arrived on such a date, coming from P1. Two different sources describe the same travel: P1-P2; but they are not brought to the researcher's attention at the same time. He reads first one of them, and may read the second one years later.

Which opens the way to three strategies. The first one consists in leaving apart incomplete information till we get a complete one.

Example IV

We load to the database the fact that the ship left P1, but we do not mention P2 till we get P2's entrance registers.

Such a strategy might possibly work for some sources which record past and wholly accomplished events of late modern history, a period when administrative processes tend to produce sets of documents able to stand alone by themselves; for most historical sources, like port registers, such a strategy would be utterly irrelevant. It would be clearly unacceptable for most early modern history documents, and worse still for medieval or ancient history.

Example V

Fichoz database on political actors of the Spanish Monarchy. At least three quarters of the information originally loaded to the database needed posterior completion in some way.

managing pictures, texts and statistics as a whole, extracting information from all of them at the same time. This is not so. At a deeper layer, they first translate all these elements to a common algebra. Once, and only once, this common ground has been built as a homogeneous space, do they handle the data to produce results.

37 For entrance registers, only reverse the demonstration.

A second strategy consists in loading every bit of information the first time the it comes at the researcher's notice. Then, once extra information on the same point has been found in another source, to change what needs to be changed in the relevant record to include new information, if any, brought by the second document. In that way, you progressively build your data without loosing any part of it.

The third strategy consists in fully loading to the database every layer of information provided by the source. This is a breach to the uniqueness principle, and as such, should be avoided. But in some cases, practical reasons makes it necessary. Port registers for instance. Identifying ships in such a source is a tricky business. Names (either of the ship's or of the captain's) vary too much from one register to another to provide a truly sound basis for identification. Neither does declared tonnage, which is often the result of a bargain between port officers and captain. Experience showed that journeys traveled and ports visited were among the best indicators we had. So that preserving destination or origins mentioned in the first document used to create the entry, even when better information is available from later uncovered ones, is a conservative measure in case the identification of the ship changes. Moreover, identification is so complex a process, that it needs putting the whole set of available information at the user's notice to work efficiently. So that in shipping databases, we chose to input all data given by the source. We later mark duplicated entries with a special marker, and we leave them apart when processing the whole with analytical tools.

Each strategy has implications as to the requirements which the data base must be able to meet. The second and the third ones are specially demanding from this point of view. The third strategy requires a capacity to display on the screen huge quantities of data which the eye must be able to embrace globally. This disposition practically rules out the use of thumbnails and makes necessary carefully thought out layouts. The second strategy is the most demanding. Apart from the requirements of the third one, it also means that all users working on the same database must simultaneously work on the same file. As soon a the database grows somewhat, most new data are in fact corrections and additions to existing ones. In such conditions, it is not even possible to think of the possibility of various files, loaded by various researchers, later to be merged into one and a same global warehouse. This is possible when implementing the third strategy, because knitting together the data brought by the various sources is done after loading them; but not when knitting must take place in the act of loading the data. Moreover, the second strategy means that before loading new data, operators must make sure that the same do not already exist in the database. This last point means that the database must be structured and equipped in such a way as to make possible fast, easy to formulate, and flexible queries. A point which hugely conditions the basic design of the same.

B. Homogeneity: a condition for global access in a context of diversity

All data, whatever be the source, must be organized in the same way so as to be accessed by the machine. So that we must structure data along homogeneous lines to allow computers to manage them all together, without loosing information in the process.

It is obviously impossible to create a specific model for every kind of documents: one for ancient books, another for archive sources, another for appointments to public office, another for wills, another for sales, another for firm balances, another for family relationships, another for people moving from one place to another, ... another for whatever you can imagine. This would mean multiplying files and tables in a way that would quickly make the system unworkable. We must find a concept which provides a common ground on which to base the input to the computer of data belonging to various of these classes, using a same set of fields, so as to limit as far as possible the variety of objects to be processed.

We described above three possible ways of transforming information into machine-readable data

(section Ic). The first process, that of letting the computer do the job by implementing a set of rules previously provided by the operator, we discarded as impracticable, in most cases, as far as historical documents are concerned, in the current state of computing techniques. The third one, that is marking segments of information by introducing tags into the text, would not resolve of question of uniqueness: a same information given by various sources would perforce be repeated. An piece of information composed of various items given by various sources which, combined together, would form a self-sufficient complete data, would be scattered among various parts of the document. The system would be able to display them all together, but no to combine them into a unique easy-to-handle computing object. The fact that all these pieces would be brought together on demand would make such a combination easy for a human mind and would be a huge improvement in relation to the previous pen and paper technology, which meant physically manipulating a number of cardboards. Our problem, nevertheless, is not to relieve human mind, but to allow the computer directly to access information. A set of rules could be given to the computer to allow such a combination; but in most cases, complexity would be so great as to make the process unworkable. This notwithstanding, as we saw before, the tag strategy is the only way to preserve the form of the document. Whenever preserving such an external layer of formal information is unessential, the only reasonable choice is what we describe as the second way, that is splitting manually information in identical segments, and feeding the computer with the same.

The question is: what kind of segments? On what basis should we split the information provided by the source? We must find the most basic criteria, the way which would make possible to account for as great a variety of documents as possible with the help of a same structure. The answer does not depend on any technical consideration, but on an analysis of what historical information is.

A combination of reflection and practical experience led us to two conclusions. First, that the most general possible way of structuring historical data consists in analyzing them in terms of actions, carried out by actors. We shall comment further this concept and expose in detail the meaning we assign to these words. For the time being, we'll just retain that action is the most general possible concept to describe historical events. Obviously, the first and main table of our database has to be based on the splitting of data provided by the source into actions. Each record of the database - each constitutive element of the table, to say it with other words - must be an action.

Second, that in spite of its pervasive character, the concept of action is unable to account for the whole of our documentation. Other dimensions exist which cannot be easily split into actions. Such dimensions must absolutely be taken into account, and cannot be left aside. Such data must be stored to other tables, organized in such a way as to account for the specific characters of the information they are based upon; tables which, the same as the action table, should be grounded on as basic and generic a representation of the concerned data as possible, so as to account for the greatest possible variety of cases by means of one table only. The final aim we pursue is, in fact, to limit as much as possible the number of existing tables without distorting the data. All such tables must be linked by specific links, so as to allow users to access from one of them data located in any other.

Example VI

Let us give an example taken from a project we are researching at the moment we write this paper, namely to what extend provincial military governors and vice-kings of the Spanish Monarchy belonged to a same restricted social group, characterized by a high level of endogamy. One the main indicators we manage is that of family relationships. For one part, we have a table of all appointments to the positions concerned. This table is an action table of the kind we just described. Every appointment is a record of the same. Every mentioned actor

is characterized by an identifier. On the other hand, we have a table of family relationships. This second table could not be structured on the basis of actions, for reasons we'll see further. It is composed of a succession of records, some of which describe births; others describe any kind of sexual partnerships, marriages included. Many of the actors featuring in the genealogical table also feature in the actions table. In both of them, they are characterized by the same identifier, which creates a link between both tables. We select, for instance, a military governor, captain-general or vice-king in the action table. We mark, in the genealogical table, all the members of his family up to, let us say, the fourth degree. Actor identifiers set in the genealogical table allow retrieving all actions (assignements to positions) in which actors selected from a genealogical point of view are involved. By selecting among these actions appointments to governorships and general captaincies we get a network of fourth-degree family relationships between such high-ranking officers. We did it by means of two different tables. It is obvious that to get this result, the computer had to be able to access to all the data it needed without any restriction or human intervention. And that the database had to be planned consequently.

Considering all that, we are in condition to design the main concepts on which to ground data atomization.

II. From source to knowledge: basic conceptual and computing options

a) Actions and actors

1) Actions

Actions are the basic units into which the operator splits information to transform it into data when using the second way of achieving atomization we described before. An action is a self-sufficient piece of information which answers five questions: who, what, where, when, with whom. If the answer to any of these questions changes, then we are facing another action. Some examples extracted from Actoz, our database on the Spanish political system of the XVIIIth century, will help making the concept clearer:

Example VII. Some actions

- [1] Antonio Adan Yarza [who] is regidor [what] of Bilbao [where] from April to August 1808 [when]
- [2] In 02/1820 [when]Antonio Arce Ovando [who] informs [what] his chiefs [with whom] of Rafael Riego's [with whom, bis] plans for an insurrection in February
- [3] On 07/07/1768 [when] Francisco Chacón Moya [who] makes Juan José Chacón Zabala [with whom] his heir [what].
- [4] On 07/07/1768 [when] Francisco Chacón Moya [who] charges to his own entail [with whom] the cost of the habit of Santiago [what] given to his son Juan José Chacón Zabala [with whom].
- [5] On 06/20/1660 [when] the ecclesiastical chapter of Santiago de Chile [who + where], in a report to the king, describes Pedro Pizarro Cajal [with whom] as "a man of great virtue and science" [what].
- [6] The Royal Press [who] of Madrid [where] publishes [what] Santiago Castro's [with whom] *Additions* [with whom] to Manuel Martinez's [with whom] *Judges' library* [with whom].

Reducing information to actions means breaking a same information piece into a set of elemental components. Each one of them is equipped with a set of dimensions, always the same, namely, who, what, when, where, with whom. Each dimension, being in itself a simple element, can be stored in one and same field. Taken together, these fields describe the action. Not all the fields need to be filled. Some of them may remain empty. Not all of them. [Who] and [What] are essential to define any action: if one of them is missing, there is no action at all. [When] is essential to historical data. [Where] does not always make sense (a nobility title, for instance, has no geographical location), and is often a circumstantial descriptive element only. [With whom] does not make sense when the actor is acting alone. Anyway, only a limited and thus foreseeable, set of fields must be used, and none must be created *ad hoc* to solve a momentary difficulty. So that these components have to be chosen at the lowest possible level of significance, that is the highest level of generality, to accommodate themselves to any kind of data. [Who], [what], [where], [when] and [with whom] are the most general possible descriptive dimensions for any conceivable action. That is the reason why we selected them for the task.

There remains a problem: the scale of the action. To tell the truth, the concept of actions, as we manage it, is not so different from the classical concept of event, as our founding fathers of the positivist school managed it. It may be seen as an event centered on the actor, provided that the concept of actor should be defined in a broad way so as embraces not only persons, but also artifacts, as we shall see in the next section. The same as an event, an action can always be divided

into smaller actions, which in turn can be divided into smaller sub-actions, and so on *ad infinitum*. When used as an argument against the possibility of achieving global history, this observation is in our view hopelessly erroneous. But it is a decisive argument against the dream of exhaustiveness. The answer is that the choice of the scale is up to the researcher and that it entirely depends a) on the purpose of his research; b) on the grain of the information which sources provide[38]. We shall go back to the problem in a further section.

2) Actors

An action is by necessity linked to at least one actor. The concept of actor is almost as central to our concept of databases as that of action. As we said before, we had to dovetail information to a strictly limited set of descriptive dimension. To process the material that sources are providing without creating new categories, we had no other solution but to expand the concept of actor. In such a way that, in Fichoz, actors include: a) individuals, such as the above-mentioned Francisco Chacón Moya and Antonio Adan Yarza (example VII/1); b) but also corporations, such as the chapter of Santiago de Chile (example VII/5), or legal entities, such as Chacon Moya's entail (example VII/4). c) It even includes artifacts deprived of legal personnality when they serve as intermediaries between other actors. For instance Castro's *Additions* and Martinez's *Judges' Library* (example VII/6). Martinez, Castro and the King's Press are actors, the first two ones individual actors, the third one a corporate actor; but actors are also the *Additions* and the *Judge's Library*, because they provide the real link between the other three. In the same way, in our database on shipping, we consider as actors ships which move from one port to another, and we treat them as such, absolutely in the same way as we do their captains and owners.

We were driven to such a decision by sheer technical necessity: it was the only way to model efficiently many actions we had to cope with, especially those related with cultural items which create around them dense clusters of relationships. We were puzzled by so unusual a move, which we made around 1995, till we grew aware that, practically on the same date, respected sociologists had reached the same conclusion, from another starting point[39]. Anyway, treating artifacts as actors gives our databases an unusual flexibility and makes far easier processing complex links. The system, among other properties, develops a capacity to grow and adapt itself to new situations without loosing its essential properties. Let us see an example:

Example VIII

Our last example, that of the *Judge's Library*, involves five actors. We could create five fields, one for each of them. But some actions may involve more actors still; the number of whom can by no means be predicted nor calculated beforehand. Such a situation would be a breach to the principle of uniformity. We resolve the problem by limiting to four the number of actors possibly mentioned in any action: a main actor, [who]; a secondary actor, [with whom]; two other actors, on whose behalf [who] and [with whom] are acting. Let us go back to the *Judge's Library*. The Royal Press is the main actor [who]. Castro, the author whose work they publish, is the [with whom] actor. The *Additions* are the actor on whose behalf Castro is acting. We link both sets by a relationship of "Publisher". There remain other two actors without situation: Martinez and the *Judge's Library*. To accommodate them, we create another action, in which Castro is the main [who] actor, the *Additions* are Castro's represented actor, Martinez the [with whom] actor, and the *Judge's Library* Martinez's representee. We link the block formed by Martinez and the *Judge's Library* on one side, Castro and his *Additions* on

38 Revel (Jacques), dir., *Jeux d'échelle. De la micro-analyse à l'expérience*, Paris, Gallimard, 1996, 248 p.

39 Latour (Bruno), *Reassembling the social. An introduction to actor-network-theory*, Oxford, Oxford University Press, 2005, p. 63-86.

the other, by a relationship of "continuator", reading from left to right that Martinez's (main with whom) *Judge's Library* (represented with whom) has for continuator Castro's (main who) *Additions* (represented who). We could have put it the other way, just changing the sign of the link between both elements: Castro's (main with whom) *Additions* (represented with whom) are a continuation of Martinez's (main who) *Judge's library* (represented who).

The system is working like an algebra. It is in fact an algebra, with the same systemic functions, the same capacity to develop along given lines to cope with unforeseen cases and the same component of conventional arbitrariness as any algebra.

Actions and actors are not the only structuring concepts which a full-fledged historical database puts in play. They account for what actors did. They do not allow listing fixed characters of objects and actors, a necessary task when you describe an archaeological item, an estate for sale (the object of an action), or when you need a physical description of a person, or the physical characteristics of a book composed by an actor and edited by another. Neither do they allow a fully satisfactory processing of legal deeds, nor the characterization of places, nor that of sources, nor the mobilization of statistical data, or even of genealogical relationships, or even less the processing of the formal content of a text. We already mentioned many of these points and we shall specially insist further on some of them. Nevertheless, we consider the concept of action as the most fundamental of all structuring principles. First of all, because of the central part it plays in the nature of history itself. A historical narrative in fact, in our view, is first and foremost a set of actions carried over by actors. The extension of the concept of actors to items which were previously considered as objects stresses even more, if necessary, its structuring function. Second, because the process which led to the unveiling of the importance of the concept of action and actor is in itself a model to be transposed to sets of data which do not describe actions.

We already stressed the fact that the choice of the action as a fundamental structuring tool was not arbitrary. We chose it because it makes possible to split a continuous flow of information into identically structured segments which the computer could process without loosing content on the way. We used the term "atomization" to characterize the process. The concept of atomization must be extended to areas which the concept of action does not cover, some of which we listed above. We must identify, for each of them, a specific concept which makes possible a similar atomization, that is a reduction of the information contained in the universe we are processing into square similar bits. If something must survive of our concept of historical databases, it must be this idea of atomization, as the underlying concept on which all the rest has been built. A concept shaped to meet both the needs of the computer and those of historical data.

Action is a pervasive concept. Every historical database must take it into account. Atomizing historical information on the basis of actions is a rather easy matter. Extending atomization beyond the area covered by actions opens a margin of flexibility, verging on casuistic. Consequently, we shall not treat this point here, but reserve it for the chapter in which we describe Fichoz as it is, from a factual, non-conceptual point of view.

Before concluding on actions and actors, we must describe two necessary complements of the action strategy, which also provide insights on atomization in the broadest sense of the word.

3) Grouping actions

Example VII/6 provides an excellent introduction to the next problem. To preserve the principle of atomization, we had to split the relationships generated by the Martinez's*Judge's Library* into two action records. Quite a usual case, in fact, as far as our databases are concerned. This notwithstanding, we need to bring them together when needed as two parts of a same set. This task is performed by a "grouping record" stored into another table. In the present case, the grouping record is logically made of a bibliographical description of the book which links both actions. This

grouping record could be anything else than a book. The description of a legal writ (a will, a sale, any legal agreement, a trial, etc.) would link in the same way all actions and relationships generated by the same (Actor A sells an estate to Actor B; Actor C is witness of the sale; Actor D, as a lawyer, writes the deed; Actor E, a banker, makes a loan and takes a mortgage on the estate; Actor F, a relative of Actor A, approves the sale and renounces any right he might have to the estate, etc.). The description of a trial would, in the same way, link all the actions generated by the proceedings. The description of an historical event would bring together all recorded actions which, all of them, compose this event.

Example IX. The battle of Waterloo

A grouping record describing the battle of Waterloo would of course give a brief account of the same. It would also bring together the following action records:

- Napoléon [who] looses a battle [what] at Waterloo [where] on the 18th June 1815
- Wellington wins a battle at Waterlo on the 18th June 1815
- Marshal Ney commands a foolish charge of the French heavy cavalry at La Haye Sainte on the 18th June 1815
- Marshal Grouchy eats strawberries at Walhain on the 18th June 1815 (with a remark in the Remarks field to make clear that this story is probably a legend)

But also:

- Victor Hugo publishes a famous description of the battle of Waterloo in his *Misérables*, in Bruxelles, in 1862

The important point is not so much what the grouping record tells, although its content may be quite significant, but the action records it brings together. In that way, it generates an indirect link between actors, a link of which analytical tools will later take advantage to link a numerous set of actors who a) could not be brought together in any other practical way (let us think of how many links would be needed to make explicit on an individual basis the coincidence of 140.000 odds soldiers[40] on the same day, at the same place); b) maintain with one another only an accidental link which, expressed as a specific [with whom] record, would endow the personal relationship between the concerned actors with a substantial quality it does not in fact possess.

4) Describing actors

As we said before, actors, considered in the broader sense with which we endowed the concept, are characterized by a set of permanent characters which the database must account for.

Example X: permanent characters of actors

- An estate can be described by its location, geographical coordinates, neighboring estates, value, generated income, equipment (houses, shop, etc.)
- A person may be described by his physical particulars: the color of his eyes, of his hair, of his skin, his height, his weight, specific features
- A ship may be described by her class, her port, tonnage, her propelling system, etc.

The way in which sources express such characters may in some cases heavily depends on the context and demand a specific processing; but taken absolutely, they must be considered as permanently attached to the actor. So that they cannot be deemed actions, nor processed as such.

40 We only take into account members of Napoleon's ans Wellington's armies and leave aside Prussians.

Consequently, they must be stored in a specific table, linked by a same identifier to the actor they describe. This link makes possible to retrieve the described according to its permanent characters, as well as according to the actions it takes part in. Establishing beforehand a list of possible descriptive dimensions is an impossible task, given that research information is unforeseeable. So that this table must be structured in such a was as to leave the question open. The only possible solution is to make it of three fields: a first one holds the identifier of the actor to be described; a second one the descriptive dimension or predicate, a feature; a third one the value of this descriptive dimension. The table holds as many records as dimensions described for as many actors as needed, as showed in the following example:

Example XI: permanent characters of two actors (estates)

...				
Identfier:	00000001	Dimension:	Surface	Value: 15 a
Identifier:	00000001	Dimension:	Location	Value: Gradignan
Identifer:	00000001	Dimension:	Neighbour	Value: Gradignan town hall
Identifer:	00000001	Dimension:	Address	Value: Market street
Identifer:	00000001	Dimension:	Class	Value: Parking lot
Identifer:	00000002	Dimension:	Class	Value: Shop
Identifer:	00000002	Dimension:	Location	Value: Talence
...				

In such a way, we are able fully to describe any action by means of three tables: an action table, which atomizes stories into individual components; a description table, which atomizes the permanent characters of actors into specific dimensions, and links each of them to the concerned actor; a grouping table which pieces together on demand various actions which make a same narrative and were disaggregated from it by the atomization process. There remains a problem which we already alluded to, and which we must now consider more in depth.

5) Limits: stylistic information

Documents provide information. This information falls into two classes. The first one comprises deeds which actors carry on, evaluations which actors formulate on the situation they are living in, and behaviors which can be ascribed to individual and identifiable actors: to make it simple, factual stories the document explicitly tells, which can be extracted from the same and recreated independently from the document, without loosing anything substantial. Let us read an extract of the first chapter of the famous pamphlet *On Buonaparte and the Bourbons* written in 1814 by René de Chateaubriant:

Example XII. Chateaubriand's On Buonaparte

"It was therefore necessary to elect a chief who might be considered as the child of the revolution, a chief through whom the law corrupted in its source might serve to protect corruption and might even act in concert with it. Magistrates endued with integrity, constancy and courage, captains renowned alike for their probity and their talents had been excited and formed by our civil discords, but a power could not be tendered to them which their principles must have prevented them from accepting. The search was almost hopeless among Frenchmen for one whose temples would not shrink from the diadem of Louis XVI. A foreigner stepped forth and was successful. The views of Buonaparte were not openly professed his character was but gradually developed. Under the modest title of Consul he first

> accustomed independent minds to behold without alarm the power that they had granted. He conciliated true Frenchmen by proclaiming himself the restorer of order, laws and religion. The most perspicacious were deceived; the most prudent were over reached"[41].

From a strictly factual point of view, we learn: a) that Napoleon Bonaparte was made a consul; b) that Chateaubriand strongly disapproved of it, in 1814 at least, and considered him as a foreign usurper of the royal throne. This is the narrative we can extract from the document and transpose to any other contextual setting, independently of Chateaubriand's wording of the same.

It is obvious that such an analysis does not exhaust the content of that magnificent piece. Neither does it transmit its nerve, neither the wealth of arguments adduced to disqualify the Emperor and his family, all of them fundamental factors for any historical conclusions to be based on Chateaubriand's work. Capturing these fundamental features make necessary an internal stylistic and rhetorical analysis. Such a task can be executed by electronic means. All of them are based on introducing into the text tags, which mark segments of the same as rhetorical devices and assign them a special role in the global economy of the piece under analysis. The best existing tool we know by now, is probably the Prospero package[42], which makes possible to mark and select arguments and rhetorical figures and to build up progressively an argumentative model of the text, later to be compared with similar models built up from other texts. Not so sophisticated tools also exist, such as Atlas.ti®[43], or better still Nvivo®[44], a package largely used by sociologists for analyzing answers to inquests, and half a dozen more. They allow taking into account the rhetorical and argumentative organization of the document, which are aspects so replete of historical information as the bare facts we have been considering till now. For the moment, we did not integrate such techniques to our databases, although we plan to do so in a near future. We content ourselves with storing, when necessary, the text as such in a special sector of the database, to allow users to access the same, and to let them cope with these aspects by themselves. This means, for the time being, a serious limitation to the scope of our tool.

b) A mitigated relational model of database

This paragraph concerns technical issues. It has been written for readers with little knowledge of such questions. Those uninterested in database structure and computer technique may skip it. Nevertheless, we consider it all-important to understand the underlying assumptions of the kind of databases we recommend. The options we support there have, moreover, far reaching consequences as to the choice of the package to be used to support the database, and as to the general organization of files and tables.

It is clear from what we said till now that what is technically known as relational databases is the model which best suits our needs; that is a model based on tables, records, fields and links. Each table is composed of various records, themselves composed of fields. All the records of a same table are composed of the same set of fields. Records belonging to different tables (A and B) can be linked, with the only condition that one field of the first table (A) and another of the second one (B) have been declared as linking fields and that both linking fields contain the same value[45]. Any field of the linked table (B) can the be called starting from any record of the linking table (A). Example

---

41 Translation published in *The pamphleteer*, London, published by Abraham John Valpy, 1814, vol. III, n° V, p. 435-436 (available in Google Books, March 2012).

42 See n. 2.

43 See *www.atlasti.com* for more details.

44 See *www.nvivo xxx* for more details. Our experience with Nvivo on analyzing texts from the XVIIth to the XIXth century was highly satisfactory (see our presentation on xxx).

45 In fact, various fields of table A and various fields of table B may be declared as linking fields; in which case both linking sets must contain the same value. It is also possible to link two records belonging to the same table.

VI (see above) is based on such a link: when the identifier of the actor named in the genealogical table is the same as that of the actor named in the actions table, data belonging to this actor displayed in the genealogical table (a list of relatives, in the present case) can be set side by side with data concerning the same person contained in the action table (namely an appointment as vice-king or as captain general).

Nevertheless, it is important to know that you can manage a relational database in two rather different ways. A relational database in the classical sense of the word (RDBMS) creates a specific table for every class of information, and a record for every value taken by an item of the class. It aggregates a lot of small pieces of information around a master key which holds them together and makes them a unique set of as many characters[46]. Management is easy, redundancy is minimal: once a value has been loaded, the one computing item which holds this value will be used to characterize all objects equipped with this same value by means of a link established between the object and the item. Each information item can easily draw complementary information from others linked to the same master key: it is easy for instance to call all the actors born in such a place and given a bishopric at a later stage of their career, that is, in the system we use to describe actions, all those whose master key is linked to a [What] "Universities studies" and at the same time to a [What] "Bishopric". Queries in which information must be retrieved from various parts are, in such a way, easily formulated. Moreover, this technology is fairly simple, deters fantasy, provides a rigid frame easy to understand by beginners and commands discipline. For such reasons it is presently hugely favored by technicians. Figure II is a simplified example of how would look three actions, according to the criteria we exposed above to describe such actions. Readers will notice that in such a fully relational model, what we described as forming in Fichoz a unique action table, needs at least five tables to be recorded to the database (Fig. II). We shall base our demonstration of what is the alternative model (we call it: "Mitigated relational model") on this same example. In this model we store every action not as a set of disjunct elements, but as a unique entry equipped with the five characters which define it (Who, What, Where, When, With whom), a set which we might name "5W". Each record is an action. Each field of this record describes one of the dimensions which characterizes the action (Fig. III, further).

Each action is an independent and self-sufficient record. The elements composing the action are not brought together by external links. They stick to one another for being fields of the same record. An actor is described by the set of all actions in which he features as a "Who" or "With whom" actor. Each actor is characterized by an identifier, manually set by the operator while, or after, creating the action record. A query based on these identifiers selects sets of actions which describes the life course of the actor (green segments in Fig. III). A same action can be assigned to various actors, when various actors act together, just by mentioning their identifiers in the same record. Re-assigning an action to another actor is a simple task which does not affect any other piece of data: it just means changing an identifier. Users are free to create extra tables and to link action records to any entry of such tables, so as to group sets of actions to define"stories", for instance, or any other kind of grouping they think fit (See above. In Fig. III, red and blue lines feature such grouping links).

46 This model is described in the Wikipedia entry dedicated to "Relational databases" (en.wikipedia.org/Relational_databases; consulted 04-May-2013).

Fig. II. Full relational model

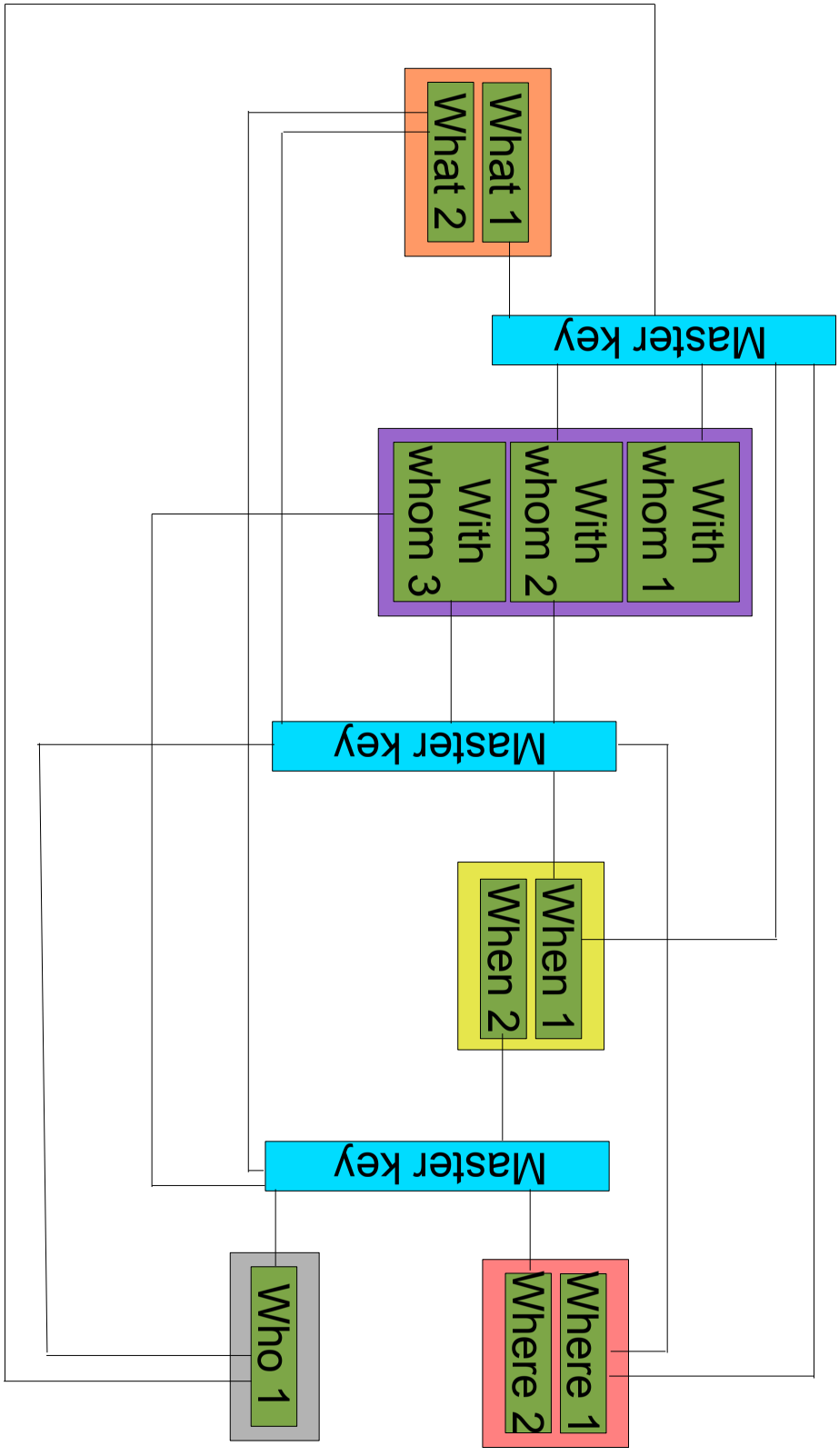
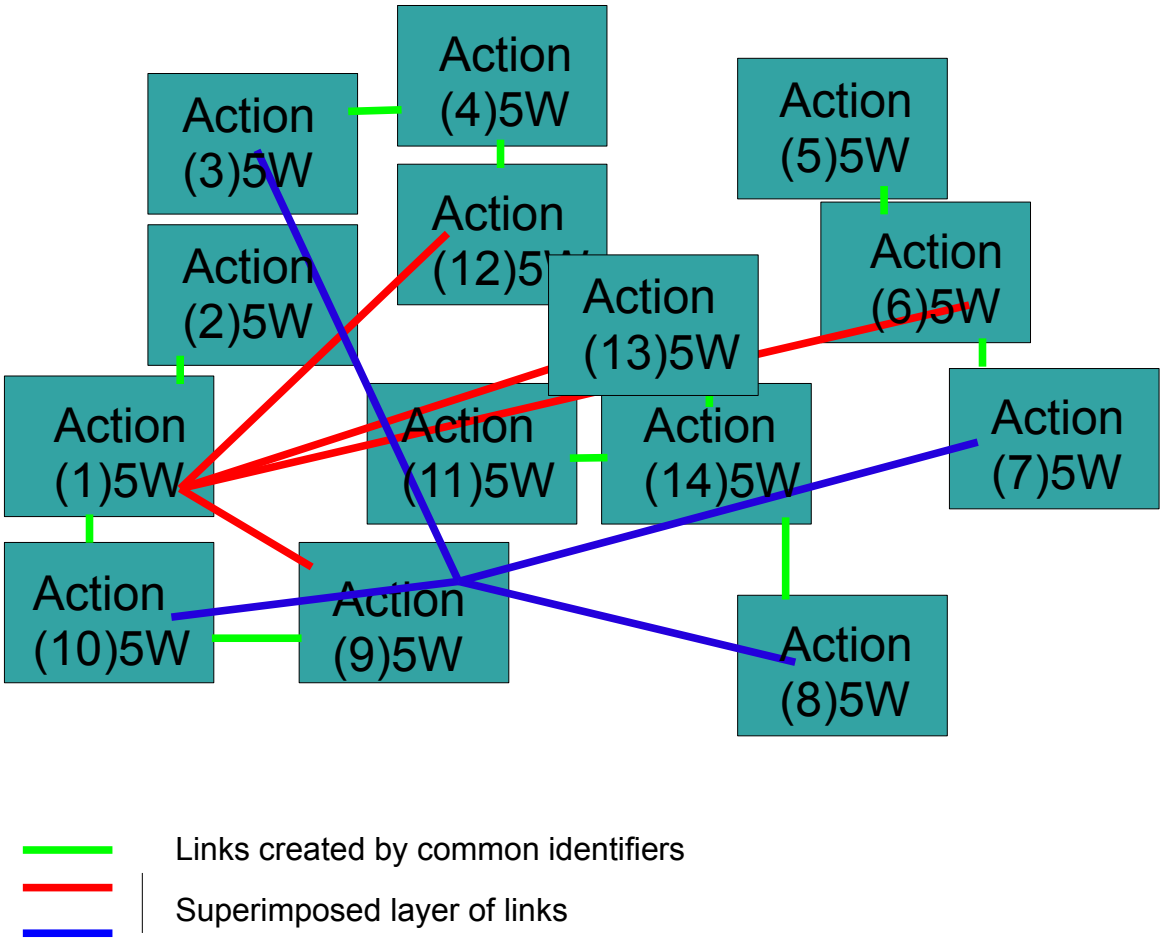


Fig. III. Mitigated relational model



In our databases, we systematically discard the full relational model. We consider it incompatible with the characters of historical information such as defined above. Two points make it, in our view, unviable. The first one is the fact that is does not accept redundancy. This point means that every time new data are loaded, all their characteristics must be clearly identified and their value localized, if existing, within the corresponding table to allow a link to be created with the relevant action or actor. Nevertheless, clear identification of characteristics at such an early stage goes contrary to the fact that historical data progressively build up at the pace that available documentation unveils them. Localizing previously used values would mean, moreover, that a huge set of queries might be made in various tables to make sure whether the value exists or not. This would mean slowing down data loading in such a way as to make the system unserviceable. Experiences such as that of the Symogih database, developed by a group of researchers of our own laboratory, showed conclusively that a fully relational model flatly does not work[47]. The second point is that the complexity of the resulting table structure makes transforming and reassigning existing data to different actors a tricky business.

47 On Symogih, see: Beretta (Francesco), Vernus (Pierre), "Le projet SyMoGIH et la modélisation de l'information", *Les Carnets du LARHRA*, 2012, 1 - p. 81-108. To say the truth, the authors do not seem conscious of this point. All those who tried Symogih nevertheless confirm such a diagnosis. Additional drawbacks, not so fundamental but quite crippling at a practical level, consist: a) in a tendency of the number of tables to grow in a disorderly way when facing unforeseen cases, a consequence of the rigidity of the system; b) the fact that queries are rather complex to formulate for untrained users, as in most applications based on a full-relational model; c) a real difficulty in reassigning facts and characters to other actors. We must also mention the generic danger notoriously generated by the use of master keys. Many errors derived from misuses of the same are part of the local humorous lore of every local computing community.

Such considerations overrule minor drawbacks of the mitigated relational system, such as redundancy[48]. A mitigated relational model, being based as it is on autonomous items, allows loading raw data as they come. It is not even necessary to identify the elements in play at that moment. Identifiers can be set later, when enough data have been loaded to generate a degree of contextualization sufficient to identify securely all relevant items. So we proceed when processing ship journeys in shipping databases - and we could not do it in any other way given the nature of the documentation.

We can then securely conclude that the characters of historical information demand a mitigated relational model, the only one which, by now, makes possible an efficient handling of such data: relational, because of the necessity to handle together various classes of data, each one organized in accordance with its own needs; mitigated to preserve a robust structure of nuclear actions, based on one computing item only, and as such easy to handle, to assign and to change without fear of loosing part of the information in the process.

d) The highest possible degree of user friendliness

We insisted on the fact that historical information demands a flexible handling of data, among other reasons because these data must be build up, in a process extended over time, which fundamentally consists in a progressive unveiling and piecemeal aggregation of tiny elements to a primitive structure; a process which makes necessary repeated access to the same piece of data. Practice stresses the practical importance of such an observation. It is difficult to describe in writing definite examples of this building process. Only direct observation of a researcher while inputting data would give a real idea of what is at play. Before loading any new item, the operator must first make sure that this item does not already feature in the database. In many cases - and every time more frequently as the database grows - he will find that the new information he uncovered only makes more complete already existing data, provides an end-date, or a more direct reference, an interesting circumstance - that a position in the army has been bought, for instance -, some further detail as to the nature of what happened - that an appointment was merely honorary, a fact not so easy to ascertain at first sight because recipients use to silence it; or that the Marquess of Cañada named by the source was not the marquess of Cañada the researcher believed he was, but the marquess of Cañada Ibáñez, quite different a person.

Example XIII.
This last case is specially interesting, because it is real. We first considered both actors as a same one. The documentation we handled then, named them Cañada, and nothing indicated that they were different persons. The facts we knew about them were perfectly compatible. We had a name, besides a title, but only for early events of their life course: in the late part of it, the actor was only referred to as Cañada. We detected the problem when we found that a marquess of Cañada, fully designated with name, surname and title, was acting after the date we supposed he died. This discrepancy could be seen only when embracing at one glance a set of some thirty chronologically ordered records, produced by a succession of half a dozen of tentative queries.

We said nothing till now of some classical problems posed by historical documents which have been considered for long as serious inconveniences for electronic data processing, such as the fuzzy character of dates, which the date format of most packages was unable to cope with. We solved them long ago in fact in quite a satisfactory way and they no longer are any drawback (see further). The

48 As long as computers had little RAM memory, redundancy was declared a taboo by the tribe of the Engineers. The taboo survives in some clans, although its function has long waned away. A moderate redundancy can even be beneficial as a protection against any accidental loss of data and to make data handling easier for human operators.

solutions we implemented nevertheless suppose a high degree of flexibility of the database system.

A capacity to provide this kind of flexibility must be a fundamental feature of any efficient historical data management system planned for scientific uses. In our view, great care must be taken of the following points:

a) To provide a capacity to display in a clear an manageable way huge volumes of data so as to allow the user's sight to embrace them as a whole. This point is fundamental to set data in context, an operation necessary to interpret them rightly. This means using large screens whenever possible; designing sets of layouts to display data at various scales, some of them broad sheets of numerous records, some others focusing on a more limited number of entries, others fully displaying a unique entry. Colors may also be a great help, as far as a same color code is used all over the database. The design of these layouts must be done in such a way as to avoid any visual recess which could hinder a free flow of the sight over them. A broken line, even a difference of one point in the alignment of two items, may have devastating consequences in that respect. Any unnecessary element must be carefully avoided, not to overload attention. Creating such visual tools means spending time and care on them. It also hugely conditions the choice of the package to be used as a basis for the implementation of the database. This package must make layout design easy and possess ample wysiwig and graphic capabilities.

b) To provide a capacity for easy queries. Inputting data means querying the database first to make sure they do not feature yet. Queried data are almost always partial and fuzzy. The package used as a basis for the implementation should make formulating such queries as light as possible a task. The structure of the database itself must be clear and simple enough to let users understand quickly where relevant matter lays - this condition, by the way, means reducing as far as possible the number of independent tables and points to a mitigated relational structure. The implementation must provide tools, programmed routines which make automatic the most usual queries as observed in practice- for instance, querying all records with the same content in such a field as the current one. And son on.

This side of the question is by no means trivial. May be it looks so when superficially contemplated. In fact, it is as important as any other factor. We showed in our first chapter that the database is not an isolated entity, but part of a string of data handling operations. The last element of the string is, in most case, the human eye. Taking into account its demands is so necessary a task as making the corresponding allowance for the requirement of any other analytical package.

e) Using well established technology

Creativity is necessary to build a database conveniently adapted to something so peculiar as historical information. Creativity, to what extent? Some researchers, when confronted with the specific needs of historical databases, were tempted with creating all-new packages to meet them. I was personally involved in two experiments of this kind. The first one was the above-mentioned Kleio venture of Manfred Thaler[49], the second one the Symogih venture, with a group of researchers of my own laboratory, the LARHRA[50]. Both were failures, in spite of the many resources which the importance of the goal they pursued attracted; in spite also of their many indisputable qualities. They simply collapsed because what they produced was unmanageable. From both attempts we may securely draw the conclusion that creating a new package is not the solution. For three reasons.

. A) Creating a package for a whole community and not to be reserved to a small group and limited tasks is a very demanding undertaking, simply out of range of a research laboratory.

49 See note 25.

50 See note 46.

Only a private firm is flexible enough for that, and even so success is a happy exception rather than a rule[51].

. B) Creating a package is so demanding a task that it shifts attention to the computer side of the question, when the problem is not the computer, but the way the historian approaches data. Data processing has long been hugely conditioned by the pencil and paper technology. So hugely as to forget that the classical approach mixed the basic rules of historical hermeneutics, those which were really essential to the process, with others, accidentally derived from technological limitations. A change in technology means rethinking our approach; preserving, and even enhancing, the basic requirements for coherency of our conclusions with the sources, with what we know of the historical context, and with common sense[52], three points on which the positivist school used to lay much stress; but throwing overboard other aspects derived from the limitations of hand-managed data handling. Elaborating a historical database is consequently NOT a task for engineers, but for historians[53].

. C) A third reason is that, as we insinuated above, a good scientific database is potentially eternal, or might be so, and in any case must be planned as if it were. Long term conservation of computerized data is in itself a highly complex and still uncertain business[54]. The existence of a large community of users is, for many reasons, a requisite for the long term survival of any computing entity. Our only chance lies in using in the simplest possible way the most widespread commercial packages; the only ones of which we can be reasonably sure that, the day they fade out of use, solutions will be provided to recover data and database structures built upon them.

Fichoz uses FileMaker, because it is widespread and under way of becoming a standard; because it is highly flexible, powerful and user-friendly. The fact that it massively uses graphical capabilities and allows easily to create new layouts for data display is specially appreciated. Fichoz table-structure, so as we implement it, is simple enough for an engineer to understand it in a question of minutes. Fichoz, nevertheless, is not dependent from any package. It must be seen, most of all, as a set of principle. Earlier versions worked on Texto. Changing from FileMaker to another package would mean that many features, which depend on FileMaker flexibility, would possibly be lost. But no basic one. The system would go on working.

---

51 I was also personally, though indirectly, involved in the Texto venture. Texto, as its name does not tell (a typical anti-commercial blunder) was a very advanced, for its times, database system, conceived by CNRS researchers of Lyon and practically without equivalent when it was launched in the 80s. Although a commercial society was created for its development, it was unable to marshal human and financial resources enough to adapt itself to relational databases and wysiwyg technologies when they became a standard at the beginning of the 90s.

52 Research has much to do with common sense. A good example of illuminating common sense applied to historical research is to be found in a monument of German positivist history, Delbrück (Hans), *Warfare in Antiquity. History of the art of war. Vol. I²*, Lincoln and London, University of Nebraska Press, 1990; 1975; 1920 [1901], eng. trans. of the third edition, 604 p.

53 We say it exactly as Clemenceau, a French Prime Minister who won the First World War, used to say that war was too serious a business to trust it to soldiers. He never meant that soldiers were unnecessary, neither did he suggest that soldiers and the military requirements they expressed were not to be fully integrated to the final solution, but that militaria were a tool, no a goal in themselves.

54 For a good comprehensive study of this all-important issue, see: Banat-Berger (Françoise), Duplouy (Laurent), Huc (Claude), *L'archivage numérique à long terme: les débuts de la maturité?*, Paris, Documentation française, 2009, 286 p.

III. Resolving old problems

Before going further, it is necessary to explain how we solved some classical problems which for long impaired the advance of computer-based historical research. These are the processing of dates - a most basic concern in history -, of names - no less basic when actors are concerned - and the question of the wording of data.

a) Dates

1) The problem

Computers brilliantly handle present-day dates. Most database package are equipped with date format fields and the corresponding commands to handle chronological information. Such achievements are grounded on the fact that a totally normalized date system is today in use, based on a) a three-layers absolute reference grid, namely Christian era, plus Gregorian calendar and GMT time; b) an acute care for time which leads to a generalized dating of any event (even snapshots are dated, today!); c) the fact that only short range time spans are put under consideration. When all these three points hold true, computers work. When not, they fail in a spectacular way. The problem is that historical documents do not make true any of these three assertions.

Depending on the culture which produced it, historical information is based on different reference systems. Muslims for instance used (and in ceremonial practice still use) to count years after the hegira. In Europe various reference grids coexisted till the first half of the XXth century. Russia used the Julian calendar till the Communist Revolution, and Greece till 1923; Catholic countries passed to the Gregorian count in late 1582; Protestant countries at various dates: Britain for instance in 1752... except for fiscal purposes. Years did not begin on the same day in all countries, or even in all cities of a same country. Such problems have long been known, and norms to resolve them long imposed by academic institutions[55]. Any historical database must obviously implement such standard solutions. Namely: 1) all dates must be converted to the standard Gregorian system and only Gregorian dates must be used to sort and calculate; 2) the original wording of the the date must be preserved, and consequently every field of the database in which a standard date is stored must be mirrored in another field in which the original date features as it appears in the original document.

Difficulties arising from point (b) have also been resolved by XIXth century academic conventions. A further problem is generated by the fact that the proposed solution cannot so easily be transcribed into computing terms as the first one. Many historical document are undated. An undated document is simply useless for historical research, as its content cannot be set into the relevant context. The classical, and still wholly valid, norm is that when no date is provided, the historian must evaluate one. Such an assessment wholly depends in turn on the historian's ability; and rarely produces an absolute date, but almost always yields a relative date: the information can be dated "around" such a known event (historians invented a specific notation for that: *circa*, abbreviated in "c."), some time after or some time before another known date, somewhere between two dates. But computerized date formats simply do not handle relative dates. So that they cannot be used in historical databases. As the solution we suggest also covers the third point, we shall expose it only after propounding the terms of this last source of trouble.

The large time span considered by historians poses the problem of dates before Christ. They raise two difficulties. The first one is a question of notation. How to write them? Many unstandardized systems exist for that: "BC", minus sign, "bef. C.", etc. This part of the question is fairly easy to solve: you choose a way of writing them as a standard, and stick to it. The second point is far more tricky. After Christ, a higher year number means posteriority; before Christ, it means anteriority.

55 Mas Latrie (Louis, comte de), *Trésor de chronologie d'histoire et de géographie pour l'étude et l'emploi des documents du Moyen Age*, Torino, Bottega d'Erasmo [Librairie Victor Palmé], 1969 [1889], 2302 p.

Once more, standard date formats as powerless[56]. Solutions cannot easily be found. We had, moreover, to resolve such difficulties in a way which would impose as little strain as possible on the operator's cognitive capacities.

2) Wording dates

We obviously had to discard using database date formats. We chose to write dates as alphanumeric strings. For Gregorian dates and the conversion of non-Gregorian dates, we make use of alphanumerical strings of the following basic pattern:

yyyy=mm=dd

in which yyyy is the year (four digits), mm the month (two digits) and dd the day (two digits). The number of characters declared for each segment of the formula is compulsory. Various calculated routines assume correctness on this point.

If information referred to the month or the day is lacking, the value of the corresponding element is set to "00".

August the 8th of 1678 must be written as 1678=08=08
In 1677 must be written as 1677=00=00
In September 1677 must be written as 1677=09=00

(1) The marker "=" describes absolute dates.

January 15th 1765 must be written as 1765=01=15

(2) The marker "<" describes a terminus *ad quem*.

An event still current on january the 15th 1765, although initiated before must be written as 1765<01<15
An event still current in 1654, although initiated before, must be written as 1654<00<00

(3) The marker ">" describes a terminus *a quo*.

After September 18th 1654 must be written as 1654>09>18

(4) The marker "-" describes an absolute anteriority.

Before 1654 (and probably terminated in 1654) must be written as 1654-00-00

(5) The marker ":" describes an approximation (circa).

Around 1750 must be written as 1750:00:00

(6) The marker "==" describes an interval.

Between 1756 and 1759 must be written as 1756==1759

(7) The marker "++" describes a conventional interval (middle, first half, second third, third quarter of such a century; century)

Middle of the XVIIIth century must be written as 1745++1755
End of the XVIIIth c. as 1790++1800
First half of the XVIIIth c., as 1701++1749
Second third of the XVIIIth c. as 1733++1765

56 So powerless that some of the most current packages - namely Excel still at the beginning of the XXIst Century - do not process dates anterior to the early 1900s. We suppose it is not a problem for business. Few loans made before still need calculating compound interests. For historians, even for those interested in the most recent times, this is a decisive drawback.

(8) Dates before Christ are written exactly in the same way as those posterior to Christ, except that an hyphen is added in the first position.

> 40 BC must be written: -0040=00=00
> Second century BC must be written as -0199++-0100
> Second millennium BC must be written as -1999++-1000

Validation routines are provided to check that the structure of a given notation is valid. Notations are intuitive, an all-important point in our view. Experience shows that users learn them fast and readers understand them at first view once in context.

Dates may be used for two purposes in a historical database. The first one consists in ordering historical items in chronological order. The second one consists in calculating durations. At the price of imposing some restrictions on the data, we were long able to achieve these goals by using as such the above mentioned formulations. An alphabetical sorting of cases (1) (2) and (4), by far the most usual ones, also provides chronological sorting[57]. As for durations, isolating the first four elements of the field gives the year, and if years are enough to calculate durations, a simple subtraction is enough to get the result, when a result makes sense, that is when the separator is "=". Nevertheless, dates before Christ cannot be managed; neither do markers number (3), (5), (6) and (7) work for chronological sorting. While processing XVIIIth C. Spanish documents, such restriction could be accepted. When we extended our scope to Ancient history, and even to medieval history, they were clearly inadequate.

We then decided to preserve the original date fields as notation fields; and to mirror them into calculated fields which reformulated the original notation into a character string able to provide a universal basis for chronological sorting and for the calculation of durations based on years. The fact that the content of this field is absolutely incomprehensible for untrained users did not deter us, as this calculated field is hidden from the user's view as well as automatically fed and calculated[58]. When we need to calculate a duration on the basis of months or days, we reduce the date to the database date format in a fourth field, and we let the machine draw the result... whenever possible.

So much for dates. We chose to enlarge on this point to give some hints of what is the essence of our database philosophy. Making a database does not only mean implementing a set of techniques. It consists in a global approach of data. The implementation of technical pre-planned solutions is only part of it. The management of a database is mostly based on matching as best as possible contradictory requirements imposed by data, machines and users; avoiding *ad hoc* solutions, the validity of which would be restricted to the current case; but elaborating working processes which can later be extended to other situations and which, as far as possible, embody previously defined conventions.

b) Names of actors

Identifying persons is a highly complex task. We rarely understand how difficult it is. We are used to identify people in normal life. But we only do it under two sets of severe constraints. The first one is the small size of our social circle: being able to name half a thousand persons is a feat. Our main database on Spanish history holds data on at least 150.000 actors. This fact is enough to show that the question changes in nature. The second constraint is that we identify people only when they are embedded in a dense context of social relationships which create a continuous tissue of interlocking

57 To solve the problem posed by separator (3), which did not fit into the alphabetically ordered series, we used to set posterior dates not as dates after a known event, but as dates before the latest possible date for the event to which a had to be assigned.

58 A fundamental point consists in adding 7000 to the year string, which makes possible to order chronologically most dates before Christ. Another point consists in replacing the intuitive separators used for notation by letters chosen in such a way that their alphabetical order matches chronology.

links joining the actor to be identified to our own person. But the nature of historical data means that the historian usually cannot access, or can only very partially access, this network of identifying relationships, at least in the first moments of his encounter with the new actor. Moreover, this tissue of relationships is extremely complex. Only a long training allows human being to manage it efficiently, and at times even trained actors fail in normal life so complex situations can be. We were three of us, called Jean Pierre Dedieu, born the same year, all of us researchers, all of us from Toulouse and the surrounding region (France), all of us working in not so clear-cut fields: a geographer, a mathematician studying complexity, and myself. I know it because the central administration of the CNRS various times mistook me for the geographer, because booksellers in their catalogs assigned me books of mathematics and because, when the mathematician died, some friends hurriedly expressed their grief to other friends and relatives of mine, who phoned me to make sure I was still alive.

Trusting computers to identify people would then be unsound, to say it blandly. The only case I know in which things worked decently that way is the "Programme de recherche en démographie historique" (PRDH) of the University of Montréal, which achieved a full reconstruction of the history of French-Canadian population from the origins to mid-XXth century[59]; but it was done in ideal conditions, and even so a not wholly insignificant percentage of cases could not be identified by the computer.

Conclusions:

a) Identifying actors (giving the word the extension we gave it in previous chapters) must be reserved to human operators, and must NOT be done by the computer. We know it's time-consuming. But it is a price to be paid. Haste and research are rarely congruent.

b) Identification, at least in a first moment, can only be provisional. It will become more secure as new data aggregate around the kernel of what was first identified as a new independent actor. For that reason, such an identification must not be imbedded at too deep a level of the database structure. It must remain a peripheral data, which can be dispensed with without preventing the database to work.

Here does the structure of a database organized as a set of independent atomized records corresponding to as many actions, deploys its full potentiality. The identification of the actor is given by an identifier stored in a specific field, which does not generate any internal link with any other record. It can thus be changed without disordering in any way other data. As we suggested before, the actor does not exist properly in the database. The system builds him up when needed by gathering into a chronologically organized series all actions equipped with a same identifier. The system acquires in that way a high degree of fluidity, without renouncing its structuring capacities.

We do not let the computer carry on the task of identifying. This does not mean that it plays no part in the process. By no means. We stressed the complexities of identification, even for a trained human mind, the variety and the bulk of the information which the identifying party had to keep in mind to proceed rightly. We showed how difficult a task it was, even within our limited social world. Manual identification at the scale of a big database would simply be impossible. Only the computer makes it conceivable, by organizing and displaying data on demand, in question of seconds, in ways suggestive of similarities and ties, which might be interpreted by a human operator - not by the computer itself - as indicating identities[60].

Finally something must be said of names. It is clear enough that, although they are an important point in identification, they cannot be the only side of the question to be taken into account; and that

59 See the portal of the program: http://www.genealogie.umontreal.ca/fr/leprdh.htm (consulted 14 May 20103).

60 Such a process, in a specially complex context, is described in Dedieu (Jean Pierre), Marzagalli (Silvia), Pourchasse (Pierrick), Scheltens (Werner), "A technical introduction...", *art. cit*. n. 5.

the name in no way may be used as an identifier around which the computer would arrange the data belonging to the actor. The more so because the name is not a neutral label set upon a person, but an expression of the social value of the same. Names do not only identify. They describe. The name makes the actor a member of a social set. For those who know - and most people around me are supposed to know -, my name generates a set of relationships with other actors, independently of any claim from my part; a set of relationships of which I am hopelessly unable to get free, even if I wanted to[61]. So that names not only would be ambiguous marker - because of homonyms - but would also be unchangeable identifiers. A point which wholly disqualifies them for the task.

All these remarks point to a fundamental property of identifiers. Identifying is their function, and nothing else. They must be devoid of any meaning or function other than denoting identity, so as to be changed when needed without disorganizing other data. We saw, some pages ago, the way in which French army identifies me. This identifier includes a lot of extra information on my person. Those who elaborated it had no choice: technical limitations, in 1941, when they planned it, were such as to make impossible any other option. They were also in conditions to do it. They were managing a set of closed administrative information. We, researchers, are not. Once again, even at the risk of boring our readers, we must insist on the fact that versatility is the master word in planning historical databases; and that this quality can be attained only by juxtaposing strictly similar and strictly unidimensional elements into which the flow of events and the tissue of social relationships must be fragmented.

c) Codification vs original wording

1) The problem

Coding was a fundamental issue when I first got in touch with computers. In 1977 there were no personal computers. I used the CNRS mainframe in Orsay, one of the most powerful set in the country - in fact, only the Army had better hardware. Computers then had no screens, no keyboards, no programs, no disks - they came while I was working there; we used tapes, and only for long-term storage - and almost no memory. I processed 8000 cases tried by the Inquisition of Toledo using... 244K of RAM. We paid a fortune for each Ko/second and funding was as scarce as it is now. For the conclusive factor analysis which was the culmination of my research, we treated ourselves (me and Michel Demonet, the engineer I was working with) to ten times that amount - but we did it on a Saturday afternoon in mid-August, when nobody else used the computer and fares were lower. Younger researchers cannot imagine how strict were the constraints we were working under. I still remember how stunned were the engineers when the characteristics of the first Cray they got in Orsay were disclosed: 14M RAM! They could not believe it. I had to pack the content of each Toledo trial, with its 40 variables, into 80 characters. Of course, we could have used two punch cards for each trial. But induced complexity and, last but not least, problems posed by the the physical handling of so many punch cards, so prone to disarray and fatal bending, would have been such as to balance expected benefits. To say nothing of the financial side of the question which, anyway, barred such a possibility.

Readers must keep these facts in mind to understand the computing tribe's passionate relationship to

61 A spectacular consequence of this fact can be seen among Spanish nobility of the Old Regime. It is notorious that many aristocratic families got extinct and that the surviving ones accumulated titles and estates. But when a gentleman acted a legal deed concerning one of his many estates, he had to use the nobility title which went with the corresponding estate. In some cases, he even had to take the first name of the founder to possess the entail. So that he assumed, being physically a same person, various legal and social personalities; a fact that only names disclose; a fact which bars any possibility of standardizing names in the database. See: Dedieu (Jean Pierre), "Familles, majorats, réseaux de pouvoir. Estrémadure, XVe-XVIIIe siècle", Castellanos (Juan Luis), Dedieu (Jean Pierre), dir., *Réseaux, familles et pouvoirs dans le monde ibérique à la fin de l'Ancien Régime*, Paris, CNRS-Editions, 1998, p. 111-146.

coding, redundancy and "making it shorter". It is basically a consequence of a kind of pre-historical conditioning, exactly as our propensity to eat too much when we get food, because we don't know if we'll get more tomorrow. As the conditions which made necessary saving memory and disk space disappeared, we can now reassess the matter on sounder basis.

Coding saves space and adds meaning. The problem lies in the added meaning. Once more, we hit the limits which an imperfect knowledge and the fact that historical information only progressively unveils itself[62] force on historians. Coding as soon as we get the data means interpreting on an imperfect basis. Knowing what we know now of historical data, conclusions are easy to draw: information must be loaded to the database as it comes. Not just reproducing the document as it is, because a mere copy does not extract from the source all its content; but loading to the database data elicited from the document by a discreet use of the rules of historical hermeneutics, validated by the common agreement of the scientific community. And nothing more. We already raised this issue, but it is so important that we prefer stressing it once more, even at the risk of redundancy.

When processing the Toledo trials, we had to summarize offenses into a two positions coding string. We established the following series:

11 Judaism

12 Mahometism

13 Protestantism

15 Illuminism

16 Masons

17 Other formal heretics

...

31 Blasphemy

33 Scandalous propositions

34 Erroneous beliefs about sex

...

By doing so, we were introducing two biases into the original information. First, by forcing the matter into a limited number of classes, we were merging under a same name offenses which the document described as rather different. Judaism for instance, in the inquisitorial meaning, refers to baptized Christians who preserve Judaic beliefs. But we sheltered under a same term so diverse behaviors as wearing Jewish amulets, draining carefully the blood out of the flesh before cooking it, or praying in standing position, head-covered, facing a wall and swinging slightly forwards and backwards. Obviously, such descriptions may all of them refer to judaism. They also may point, each of them, to different beliefs. We assumed that, given that the inquisition chose to inform on them, they were indicative of judaism, but this was an assumption, not a fact.

The second bias does not relate to individual items, but to the whole of this classification. Not only did we delimit classes and force the information into them, but we also ordered such classes. We gathered into a first block a set of offenses, the coding string of which we decided would begin with a "1", then a second, a third and so on up to nine. By doing so, we implicitly assumed that judaism, mahometism, protestantism and masonry had, from an inquisitorial point of view, something in common which made them different from blasphemy, scandalous propositions or sexual beliefs a bit too discrepant from the teaching of the Church. It was a daring move. Assigning a phrase of the kind

62 On this point, see the first chapter of the present paper.

"It is not an article of faith that such a person is really the pope" to the scandalous propositions class or to formal heresy was in practice a matter of context for the inquisitors themselves. Being obliged to assume at first sight, as we did, without previous examination of the whole business, that it was the one of the other, was obviously risky.

Nevertheless, we had to do it, and hope for the best. In fact, in the context of our own research, the problem was largely mitigated by the fact that we were using inquisitorial sources, which reflected the inquisitors' mind, and that we were precisely investigating this opinion and the way it formed itself, rather than the content of the opinion in the defendant's view. But was it legitimate to recycle our data into another research, interested in the defendants' opinion? May be. Anyway not without an acute awareness of the problem and an adequate strategy to annul it.

2) Inputting and identifying data

With modern computers, space is no longer a problem. We suggest the following procedure which gave fairly good results in the many years of practice we accumulated working on various databases with data extracted from various cultural contexts:

1) Before loading any data to the database, make sure that this data does no already feature in the same. If it does, see if the source which you are managing brings some new information on the case. If so, add the new information to the existing record. If not, proceed to the next document. If the data does not feature, create a new record. Never skip any of these stages. Making systematically sure that the concerned data does not exist in the database before loading is boring, time-consuming and demands a high degree of self-denial. But trying to manage a database in which redundant data have been stored is still more time-consuming and boring. Remember that making a database does not mean only saving information, but creating manageable data.

2) The wording of the record must keep as close as possible to the wording of the best available document. The kind of database we use to work with usually demands that the various possible versions a a same action be combined together to form a unique entry. It is up to the operator to merge various information pieces in a way which preserves the integrity of the information without letting aside any piece of relevant significance. The operator must be competent enough to understand what really matters, what really is information and what is not. The fact that a councilor of Castille is appointed with honors and seniority but without a salary, is information, and minute details therein make all the difference. By contrast, the fact that he has been named by the king - a fact historians, especially those of ancient times but also modern genealogists use to underscore to make the actor look socially important - has no significance at all: all councilors of Castille were named by the king.

3) As for the physical transcription of the information, we personally do not reproduce out-of-date orthographic variations: they make queries more complicated and they do not mean anything substantial, except for philologists, and we are not making databases for philologists[63]. We long used to arrange graphically every part of the information so as to organize it in a way which allowed the computer to locate easily different parts of the same. We dropped this scheme when we grew aware that the same result could be obtained in a far more efficient way through different means which did not demand an intervention in the first stage of data loading, namely by using permanent coding strings, as we shall explain soon. Names may be orthographically normalized, at least for documents produced before the

63 We know enough paleography to be aware of the fact that reproducing minor graphic variations is in no way a promise of accuracy. One always reproduces what he reads. If the text has been misunderstood, its paleographical reproduction will be erroneous. So simple.

introduction of an efficient civil register service[64]. As far as original documents are concerned[65], nevertheless, all items used to name the person must be preserved, because they carry social information. We already mentioned this point. Louis de Rouvroy and the duke of Saint-Simon, for one part, Arthur Wellesley and the duke of Wellington for another, although a same body, are legally and socially different persons.

4) Set a date, and if none is provided, calculate one. We shall not dwell on this issue, the importance of which we already stressed. It is all-important. Let everything else fail, but not the date. So runs the teaching of our German founding fathers.

5) Identify actors. Onomastical variations make difficult retrieving all records referred to a same actor, an operation of vital importance, as we saw before. We treat identification as an extra descriptive dimension added to the record, a descriptive dimension which only points to the fact that all data items in which this dimension has a same value refer to a same individual. This dimension is brought to the data by an independent identifier, set in an independent field, at the side of the one which holds the name of the actor. A name field always goes with its identifier. The identifier of a same actor is obviously the same every time that this actor is mentioned, whatever be the wording of the name (Louis de Rouvroy and the duke of Saint Simon, obviously have a same identifier). The identifier is different whenever two different actors are mentioned, although their name be the same. A list of unused identifier is provided by a special dictionary, embedded into the system, to make make selection easier, as we shall explain in the next chapter.

6) Identify in the same way place names. They must also be equipped with identifiers, which make possible their location on the map[66].

Identification is a tricky, hard and time-consuming endeavor. It cannot always be done while imputing data. Identifying actors cannot be done while considering data as isolated items. They must be set in context. Only the consistency of the suggested identification with all other known elements referred to the actor makes possible to reach his identity under the disguise of the various denominations the sources use to name him. The same for place names. The same for the institutions through which the actor opens his way. The same for dates. For such fundamental tasks, the corpus of conventions for the hermeneutics of historical data codified by positivist historical science of the XIXth century, contributes guidelines and reference tools. They are a measuring rod against which operators must check the atomized data they have been elaborating.

Let us describe a real and average case taken from , a database on shipping[67]. It makes clear the kind of pragmatic approach needed to cope with the complexity of historical data. We record on paper every step we gave in the process. We stress in that way that a database is not an abstraction, but a tool designed to deal with practical cases, and must be planned so as to make possible and, as far as possible, easy, the kind of operations we are describing. We shall insist again on this point later.

We start from a list of actions. Each action records the fact that a ship a) named by the source, b) of a certain tonnage sometimes mentioned, sometimes silenced by the source, c) commanded by a captain usually named by the source, crossed a geographical point (usually, but not necessarily,

64 I never was able to find any difference between Giménez y Jiménez, or Gonzáles and González in XVIIIth Century practice. The same person is frequently named both ways in the same page.

65 You may perfectly drop this rule when using a published or secondary sources in which names have been reproduced by modern historians or genealogists.

66 Geographical points are in fact highly complex entities, which can be described from at least in three different points of view. A same geographical point receives up to three different identifiers, depending on which side of the concept the database is considering. See the entry "Point", in the global Help file for all our databases, to be found at: fm.tge-adonis.fr/Fichoz_help.fm12 (10-01-2014; see the Appendix to the present paper on how to access Fichoz).

67 On , see: Marzagalli (Silvia) "...", n. 5.

entered or left a port), on a certain date. The purpose is to identify those ships by setting a same ship-identifier to every record concerning the same ship. The information has been drawn from a variety of sources, some of them of difficult reading, and we suspect that different denominations in fact remit to a same actor. The syntax of the queries is that of FileMaker. The layouts we present in the examples quoted above are those of . The example is divided into as many steps as we gave to obtain the desired result

Example XIV. Identification of a ship in

(1) Query: Captain = "*ebert*" + sort on ship name. Results: more thant 100 records, refered to:
. Ship: "Aimable Elisabeth", captain: "Hebert J" or "Hebert Jacques", homeport Dieppe, 90 t
. Ship: "Aimable Marie", captain: "Guebert Jean B", 90 t
. Ship: "Aimable Reine", captain: "Hebert Jean Baptiste" or "Guebert Jean Baptiste", 30 tx
. Ship: "Aimable Rose", captain: "Hebert Ch" or "Hebert Christian", homeport: Dieppe, 40 t
. Ship: "Alexandre", captain: "Hebert Philippe in the first chronolocally sorted records; latter "Pierre Philippe" when registered in Rouen; "Hebert Pierre" when registered in Honfleur and Le Havre, 70 tx, usual route: Rouen / Le Havre
. Ship: unnamed, captain: "Hebert Germain", homeport: Brest, 400 tx
. Ship: "B... Aimé" / "Belle Aimée", captain: "Hebert Guillaume", "Hebert Gille", tonnage: 102, 104, 112 tx, usual route through Le Havre, Rouen, Honfleur
. Ship: "Benjamin" or "Binjamin", 49 / 69 tx, homeport: Dunkerque. Usual route through Cherbourg, Rouen, Bordeaux, Calais)
. Ship: "Bisquine", captain: "Hebert Guillaume", 112 tx, usual route: Honfleur, Rouen, Honfleur. Probably the same as the Belle Aimée
. Ship: "Couzeur", captain: "Trébert, Jean Joseph"
. Ship: "Victoire", captain: "Hébert, Jean Baptiste", homeport: San Valerie en Caux or Barfleur, 40 tx to 300 tx, usual route through Barfleur, Port Bail, Dieppe, with excursions.

From now on, we explore in depth this last case.

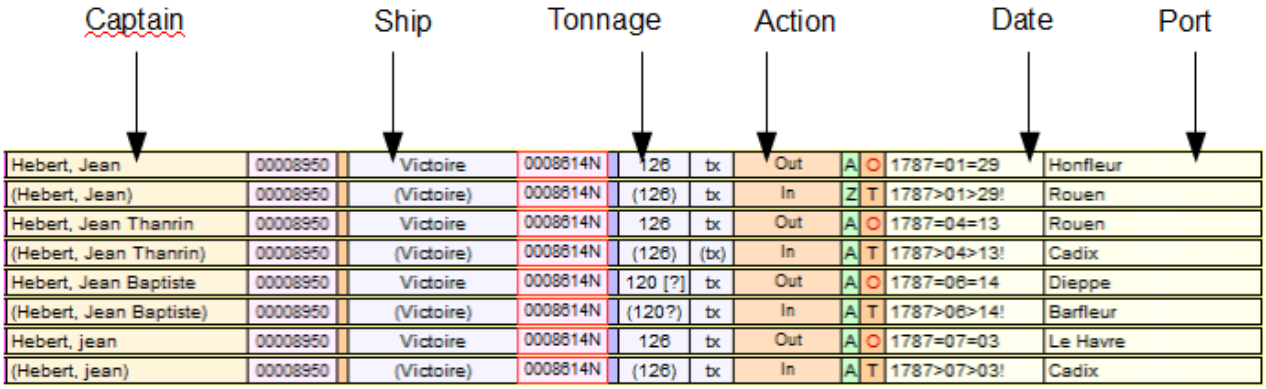
(2) Query: Captain = *ebert and ship =Victo*. Results:
Ship: "Victoire", captain: "Hebert Jean" or "Hebert Jean Baptiste", or "Hebert Jean Thanrin", usual route: Barfleur, Saint Valéry en Caux, Cherbourg, Dieppe; 27 records. We were tempted to identify all of them as a same ship. Nevertheless, declared tonnages range from 40 to 300 tx, with intermediate values of 120 and 126 tx, too wide a span to allow identification. We decided that they were different ships and we split the set, using tonnage as the discriminating criteria.

Then:

(3) Query: Captain = *ebert and ship = Victo* and tonnage = 12*. Result:
Ship: "Victoire", tonnage 120/126 tx, captain "Hebert Jean", "Hebert Jean Baptiste" or "Hebert Jean Thanrin". All of them we identified as a same ship and a same captain, giving the ship the identifier "0008614N" and the captain the identifier "00008950"; the identification of "Jean Thanrin" with "Jean Baptiste" was made stronger by the fact that all points of the route were interlocked and compatible, as the ship was repeatedly recorded as leaving one port bound to another, and later recorded again as entering this last one.

After step (3), the screen looked like that:

Fig IV. Screen capture after query (3) after setting Victoire's identifier (red borders)



Hebert, Jean	00008950	Victoire	0008614N	126	tx	Out	A	O	1787=01=29	Honfleur
(Hebert, Jean)	00008950	(Victoire)	0008614N	(126)	tx	In	Z	T	1787>01>29!	Rouen
Hebert, Jean Thanrin	00008950	Victoire	0008614N	126	tx	Out	A	O	1787=04=13	Rouen
(Hebert, Jean Thanrin)	00008950	(Victoire)	0008614N	(126)	(tx)	In	A	T	1787>04>13!	Cadix
Hebert, Jean Baptiste	00008950	Victoire	0008614N	120 [?]	tx	Out	A	O	1787=06=14	Dieppe
(Hebert, Jean Baptiste)	00008950	(Victoire)	0008614N	(120?)	tx	In	A	T	1787>06>14!	Barfleur
Hebert, jean	00008950	Victoire	0008614N	126	tx	Out	A	O	1787=07=03	Le Havre
(Hebert, jean)	00008950	(Victoire)	0008614N	(126)	tx	In	A	T	1787>07>03!	Cadix

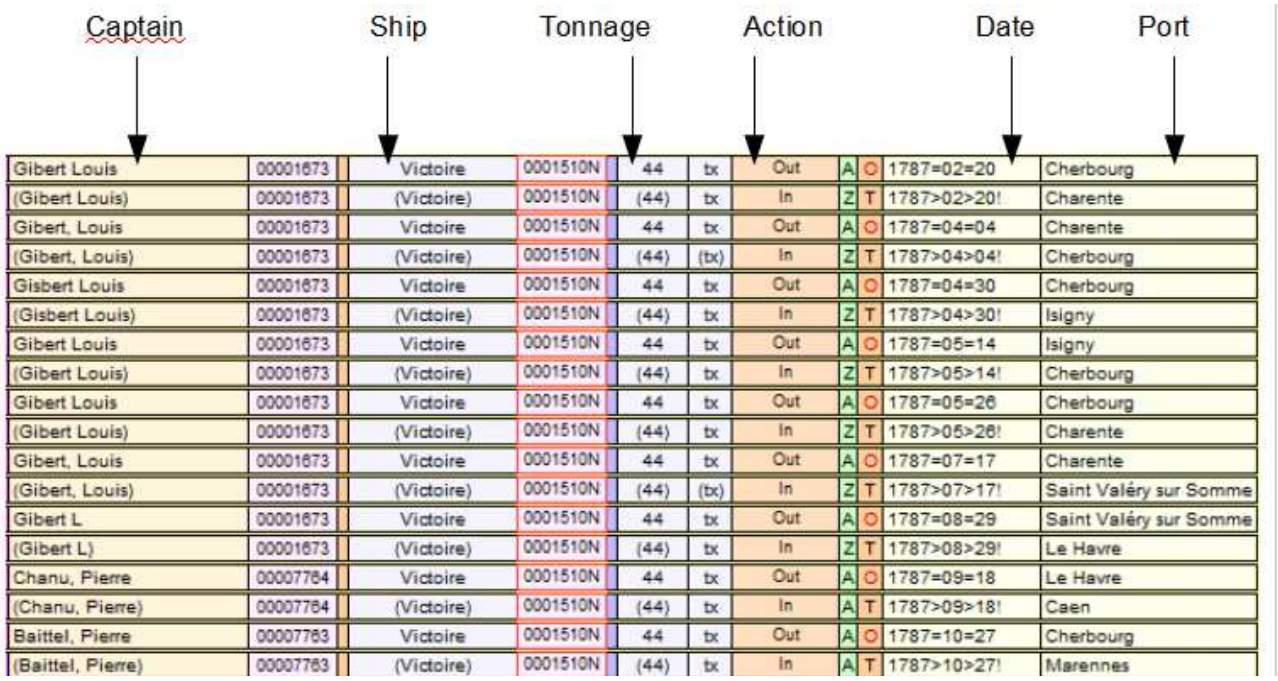
(4) Query: captain = *bert and ship = Vict* and tonnage = 40. Results:
Ship: "Victoire", captain "Hebert Jean", "Hebert Jean Baptiste" or "Hubert Jean", tonnage 40 tx, no interlocked, but compatible points. We made them a same ship, and gave the captain and the ship their own specific identifier.

(5) Query: captain = *bert and ship = Vict* and ship identifier empty. Results:
. Ship: "Victoire", captain: Aubert Pierre or Aubert Louis, 12 tx, homeport: Courseulles, usual route: Granville/Courseulles/La Hougue. Identified by us as an independant and same ship. Captain and ship were given specific identifiers.
. Ship "Victoire" or "Aimable Victoire", captain: "Ferey, Robert" or "Feray Robert", tonnage: 50 tx, identity confirmed by interlocked ports. Identified by us as an independent and same ship. Captain and ship were given specific indentifiers.
. Ship: "Victoire", captain: "Gibert L" or "Gibert Louis",tonnage: 44 tx. We decided a detailed exploration of this last casde, because all data were not, at first view, consistant.

(6) Query: Ship = "Victoire" and tonnage = "44 tx". Results:
. Ship: "Victoire", captain: "Gibert L" or "Gibert Louis", or "Gisbert Louis", tonnage: 44 tx., homeport sometimes metioned as Cherbourg.
. Ship: "Victoire", captain: "Chanu Pierre"
. Ship: "Victoire", captain "Baittel Pierre", homeport: Cherbourg.

The identity of name and tonnage, and most of all a highly interlocked set of geographical points, made possible the identification of these three entries as belonging to an identical and independent ship, which might then be given a same identifier, 0001510).

Fig. V. Final result



Gibert Louis	00001673	Victoire	0001510N	44	tx	Out	A	O	1787=02=20	Cherbourg
(Gibert Louis)	00001673	(Victoire)	0001510N	(44)	tx	In	Z	T	1787>02>20!	Charente
Gibert, Louis	00001673	Victoire	0001510N	44	tx	Out	A	O	1787=04=04	Charente
(Gibert, Louis)	00001673	(Victoire)	0001510N	(44)	(tx)	In	Z	T	1787>04>04!	Cherbourg
Gisbert Louis	00001673	Victoire	0001510N	44	tx	Out	A	O	1787=04=30	Cherbourg
(Gisbert Louis)	00001673	(Victoire)	0001510N	(44)	tx	In	Z	T	1787>04>30!	Isigny
Gibert Louis	00001673	Victoire	0001510N	44	tx	Out	A	O	1787=05=14	Isigny
(Gibert Louis)	00001673	(Victoire)	0001510N	(44)	tx	In	Z	T	1787>05>14!	Cherbourg
Gibert Louis	00001673	Victoire	0001510N	44	tx	Out	A	O	1787=05=26	Cherbourg
(Gibert Louis)	00001673	(Victoire)	0001510N	(44)	tx	In	Z	T	1787>05>26!	Charente
Gibert, Louis	00001673	Victoire	0001510N	44	tx	Out	A	O	1787=07=17	Charente
(Gibert, Louis)	00001673	(Victoire)	0001510N	(44)	(tx)	In	Z	T	1787>07>17!	Saint Valéry sur Somme
Gibert L	00001673	Victoire	0001510N	44	tx	Out	A	O	1787=08=29	Saint Valéry sur Somme
(Gibert L)	00001673	(Victoire)	0001510N	(44)	tx	In	Z	T	1787>08>29!	Le Havre
Chanu, Pierre	00007764	Victoire	0001510N	44	tx	Out	A	O	1787=09=18	Le Havre
(Chanu, Pierre)	00007764	(Victoire)	0001510N	(44)	tx	In	A	T	1787>09>18!	Caen
Baittel, Pierre	00007763	Victoire	0001510N	44	tx	Out	A	O	1787=10=27	Cherbourg
(Baittel, Pierre)	00007763	(Victoire)	0001510N	(44)	tx	In	A	T	1787>10>27!	Marennes

As we can see from this example, the operator must possess a keen sense of virtual possibilities. Finding matching cases is up to him. He must imagine, invent and create possible tracks of identification. At the same time, he must be careful not to break standards of consistency. This is decidedly not a job for beginners.

Our description of this process indirectly highlights, once more, the requirements which the database package and database structure must necessarily meet. We made a total of six complex queries, to resolve one set of identification. , at the moment we write this part of the present paper (end of September 2012), holds some 5.000 such sets. It means that between 40.000 and 50.000 queries will be necessary to identify men and ships. We equipped the database with triggers which make the most usual queries automatic. Our guess is that some 15.000 queries will nevertheless have to be manually written by users. User-friendliness of the package on that point is an absolute requisite. Which means that the database package must fully use the wisiwig and graphical facilities of the computer. A fact much to be taken into account when choosing it. All those which do not meet this demand must be discarded. This also means that the database implementation must provide as many automatized queries as needed to relieve operators of the highest possible number of manual queries. Even at the price of making the system more complex.

3) More markers and more descriptive dimensions

Once information has been atomised, once every bit of information has received a date and an identifier, and in the process has been transformed into data, we can still be enrich it in a variety of complementary ways, by adding to the data extra informative dimensions which bring to the user's attention underlying implicit information. In , for instance, we built up points into stages, that is pairs of geographical points linked by the fact that a ship went directly from one to another. From the 85.000 departures and arrivals which the database mentioned when we wrote the first version of this paper, we built 5.000 stages, that is sets of two points linked by at least one trip, many of them repeated by various ships which made a same route and journey. We built a special table with these stages. Each item of the new table, namely each stage, we equipped with its length in nautical miles. We characterized each of them as international, local, regional or inter-regional, and we used the

resulting dictionary to characterize ship routes and ports, according to the kind of stages they contributed to.

The same as we atomize journeys into points, we also atomize in cargoes into cargo items: we considered that the cargo item is different every time that the basic product, one of the qualities of the same mentioned by the source, or even the unit used to characterize its quantum changes, even if the source makes one entry out of what we considered various items.

Example XV:

Source: "35 bushels and 6 sacks of wheat, 260 bushels of barley" make three entries, stored into three records of the cargo table that is:
35 bushels of wheat
6 sacks of wheat
260 bushels of barley

Each cargo item is obviously characterized by the identifier of the point at which it has been described by the source from which we draw informations, and consequently by the identifier of the ship which carried it. We characterize it also with an identifier of commodity, that is a description of the product which makes the cargo item. An identifier which in turn remits to a commodity table, in which every commodity is characterized along three dimensions, namely the raw material it was based on, the industrial process used to produce it and the most usual uses it is subject to. All that makes possible complex classifications and calculations based on the commodity[68], which in turn can be combined with data on the ship, on the stage or with whatever information preserved in the database.

*

* *

We learn from such examples that building up data is an unlimited process which researchers can extend indefinitely depending on their needs. The kernel of the database must keep as close to the original information as possible. It must be atomized into actions, to be processed by the computer, but in a way which preserves untouched factual information, and the atomization must be carried out in accordance with universally accepted criteria of historical validity. To this nucleus a variety of layers of extra information may be added by successive users, to enrich it, to make it more global and embracing, easier to embed in complex and broad working hypothesis. These layers must be clearly distinguishable from basic information and data. Users must be able to change them if they deem it necessary without affecting the hard central core. A database must be build in a way which makes such an enrichment possible. It must be able to work and allow an unlimited range of complex queries from the moment in which the core has been filled in, without previously requiring that complex description processes be carried out. It must provide users with a set of dictionaries in which to store complementary characterizing elements.

4) Beyond identifiers: coding

Till now, we fundamentally described identifiers and identifying processes. We defined identifiers as unidimensional markers, which carry one information, and one information only, namely that all pieces of data equipped with the same identifier describe a same object and a same actor. We insisted on the fact that unidimensionality is an essential feature of any efficient identifier. We must now describe another class of markers, coding strings, which, contrary to identifiers, are characterized by the fact that they carry a huge amount of miscellaneous information. They also

68 Dedieu (Jean Pierre), Marzagalli (Silvia), "Tracking Trades in . The example of fish and cotton", communication à la European Social Science History Conference, Glasgow, avril 2012.

bring an identity to the item they are appended to, but an identity which, contrary to the identifier, does not stress their individuality, but the fact that they have in common the set of properties described by the coding string. We use two kinds of coding strings.

The first and more important one we call the "permanent coding string". It must be permanently embedded into each record, in a special field. It describes the current action in such a way as to make explicit all its relevant institutional[69] connexions. The best way of explaining its nature probably consists in commenting some cases.

Example XVI: Permanent coding of a nomination to a position of member of the Roman Congregation *De propaganda fide* (first years of the XIXth century), extracted from Actoz:

AAxxx-CGHxxD-EIxAAx-xx

The coding string is composed of four blocks of letters, separated by hyphens. Each one codes a descriptive dimension of the object. Each one is composed of a fixed number of signs, the position of which is significant. Empty positions are marked with an "x". Only upper case letters convey a meaning.

The first block defines the contextual universe. "AA" means Catholic Church. The three next positions remain empty. They would in other contexts point to the a state and to the legitimacy of the relevant government. Such concepts are irrelevant in the case of the Church.

The first letter of the second block marks that what follows concerns the Curia. The meaning of this position depends, obviously, of the previous block. In another contextual universe, this same letter set in this same position would have quite a different meaning. The next two letters indicate that, within the Curia, a Congregation ("G") is concerned, and that this congregation is that of the Expansion of the Faith (*De propaganda fide*, "H"). The next two positions remain empty: they should indicate the office within the Congregation, an irrelevant data in the present case. "D" marks the hierarchical position of the actor in the institution: all full-members of any institution, all over the database, have a "D" there.

The third block is indicative of the geographical location of the institution, if this information is relevant. "EIxAAx" codes Rome, as an Italian city belonging to the State of the Pope. The coding of this same city would be different at the end of the XIXth century when this State no longer existed and Rome had become the capital of the Kingdom of Italia.

The fourth and last block is used in the coding of official positions only. It marks the way in which the incumbent holds the position: full possession ("xx"), honorary, provisional, etc.

Roman congregations are institutions in the legal meaning. Our next example concerns a social institution:

Example XVII: permanent coding of a birth in Madrid (end of the XVIIIth Century) (from Actoz)

LVxxx-Nxxxxx-PGxAAx-xx

"LV", the universe, is that of the main vital events with articulate the life of everybody: birth, death, marriage and similar civil events (religious rites fall in the "A" class, "AA" for

69 We use the word "institution" in the meaning it has in economics, that is: rules which organize human interactions in such a way as to reduce individualistic or erratic behaviors; thus making behaviors more previsible" (Kasper (Wolfgang), Streit (Manfred E.), *Institutional Economics. Social order and public policy*, Cheltenham, The Locke Institute, 1999, p. 30). We embrace institutions in the legal meaning as well as social institutions which no legal definition embodies.

Catholics, "AM" for Muslims, "AP" for Jews, etc.). "N" is birth. "PGxAAx" is Madrid city, set into the institutional context of late XVIIIth Century:"P" for the Kingdom of Castile, "G" for Madrid y Real Palaces (a relevant territorial division at that moment), "AAx" for Madrid.

Example XVIII: a failed appointment of a member of the Council of Castile by the Austrian pretender to the Spanish throne (beginning of the XVIIIth Century) (from Actoz)

FFEAP-AKxxxD-CAxxxx-xU

. First section: first "F": political universe of the Old Regime; second "F": royal institutions; "EA": Spanish monarchy; "P": Austrian pretender (if the appointment had been made by the Bourbon king, whom mainstream historiography considers as the legitimate king, the corresponding value would have been "x").

. Second section: "A" Counsel; "K" Counsel of Castile; "D", full member, as in the first example.

. Third section: "CA" the capital of the monarchy; the position is not attached to any territorial district, but to the place where the government has its seat.

. Fourth section: "U" in the last position, appointed, but never took possession (the pretender lost the war and left Spain before the incumbent was able to take possession).

This is obviously a complex and elaborate way of putting things. Designing such an instrument was a rather long and complicated task. Till now, we fully carried it on it for Roman papal institutions and for the institutions of the Spanish monarchy in the Old regime only. Coding in such a way demands a full global view of all possible institutions which may be mentioned in the database, and a clear representation of the relationships they maintain. On the other side, permanent coding strings provide an extremely powerful tool to identify any institution, independently of the form which the record in which it features has been worded, and thus resolve a wide range of problems. A mere dictionary would provide a translation to any language. The kind of coding we are using does something more The hierarchical character of the string, from more global to more peculiar concepts as we read it from left to right, provides an immediate understanding of the position of the institution in context and makes possible an easy retrieval of related cases. The permanent coding string, moreover, reduces the action to its essentials. In such a way it allows retrieving purified information items. A query based on words is always contaminated not only by orthographic variations, but also by the fact that any text includes words which may also feature in other contexts. The most efficient way of retrieving through a word-query, in Actoz, appointments of counsellors of Castile would be to look for "Counsellor" and "Castile" in the Action_text field. It would display such appointments, but also strings such as "Counsellor of the Counsel of the Inquisition, assessor of the Counsel of Castille", which is not what we are looking for.

A permanent coding string has in database practice the same role as heavy artillery in warfare: crushingly efficient, but long to set in motion and lacking flexibility. We supplement it with what we call "on-the-way" coding strings. These are labels which users may easily stick to any previously selected set of data to mark them as belonging to a same *ad hoc* conceptual set. Such labels help to retrieve and handle marked data, and can be erased after use. We shall deal with them at length in the next chapter, in which we describe Actoz database, as an example of implementation of the principles we exposed so far.

5) Languages

Once the use of permanent and on-the-way coding strings has been explained, we are in condition to tackle the thorny issue of language. An issue which literature on databases rarely raises. In fact, it

only matters in social sciences and humanities databases. Other sciences naturally stick to English. The terms of the problem are, in our view, as follows:

1) A database as we conceive it, must not be planned for personal use. We know by experience that databases are fantastic tool for collective research. We shall explain why in our conclusion. This fact means that users may speak and write various languages and, a more embarrassing point, be native speakers of various languages.

2) A database cannot be divided into sections. The whole of the database contributes to the understanding of any specific part of it. No part can be left aside without affecting the rest of it. Queries must not be limited to specific parts of the database. Irrelevance to the research under way is the only acceptable limit. No technical consideration must prevent such an achievement.

3) Databases - even big databases - may fairly well draw information from sources written in one language only, or at least almost exclusively written in one language. Such is the case of our big and first database on Spanish political system in the XVIIIth Century, Actoz, from which we drew most of the experience on which we based the present paper. Others necessarily use multi-languages sources. A database on liberal and anti-revolutionary militancy in XIXth Century Europe must handle documents in Spanish, Portuguese, Italian, French, English, German at the very least; Greek, Polish, Dutch, Danish, Swedish and Russian would probably have to be taken into account; Arabic and Turkish also from mid-XIXth century on. This poses a problem of mutual understanding and necessarily reduces the access of any researcher to the data worded in languages he knows. Which contradicts point (2).

4) There are almost no technical limits today to the use of any language or alphabet in a database. Mixing various alphabets within a same field is even possible.

Fig. VI. An example of Chinese and Western characters in a same field (Source: FarEast database):

Rho, Giacomo	00000001			Presents		0009998L	Chong zhen li shu 崇禎曆書
Si ku quan shu 四庫全書	0000115L			Part		0009998L	Chong zhen li shu 崇禎曆書
Li Tianjing 李天經	00000005			Compiler-PIC		0009998L	Chong zhen li shu 崇禎曆書

5) Translating into another language information given by historical sources is a difficult task. It requires a perfect understanding in its original langage of the phrase to be translated, a no less perfect understanding of the equivalent vocabulary in the destination language and a capacity to establish an exact correspondence from the one to the other; that is a degree of linguistic and historical competence rarely to be found. In many cases no exact equivalent exist (try and find a French translation for Spanish "hidalgo"... "gentilhomme" does not work. English "gentleman" would be somewhat closer, but still imperfect). In any case, the most competent historians will be able to manage in such a way two of three languages, never more. And such a translation would require a careful pondering of various possibilities: we cannot imagine it could be done "on the way", as inputting translated data into a database would require.

6) Identifiers and coding strings provide universal and wholly exact representations of historical objects; representations independent from any linguistic capacity, except a general knowledge of the most basic conventions underlying all Western languages (reading from left to right and a knowledge of a basic set of Latin letters).

We suggest the following strategy, still to be tested in working conditions:

1) All texts which belong to the meta-structure of the database, tooltips, help files and similar parts of common use, might be written in English, which is the most common language used by the corporation and a reading knowledge of which is a common requirement for any scientific work.

2) The rest of the database should be written in the language of the source from which the information has been drawn; in cases in which various sources in various languages would be used to describe a same data, in the language on the best source.

3) Sets of identifiers and permanent coding strings should be implanted as soon as possible into the database so as to make it manageable to linguistically incompetent users.

4) An exception to the general use of local languages could be allowed for files containing information for common use, the knowledge of which would be necessary to understand fully the permanent coding strings, for instance institutional dictionaries such as the Diem file of Actoz (see further).

IV. Implementation: a full description of Fichoz database

The last part of this small treatise describes the Actoz database, not for its own sake, but as a practical example of how an efficient tool may look.

a) Core and periphery: a conceptual description

We must first introduce a fundamental concept which we are now in condition fully to understand, namely the difference between core and peripheral tables. We saw before (section I-d2) that, for one part, databases for research must provide uniformly structured and unambiguous data; but that for another part, the complexities, and the variety, of original sources from which we draw historical information make such a purpose almost impossible to achieve. Many classes of documents demand in fact specifically structured databases, to allow an efficient extraction of data.

Example XIX

A. Overlapping complex documents cannot be reduced to a set of independent unambiguous non-redundant actions without loosing much information on the way, or even without making impossible identifying actors, a necessary step to describe actions. We analyzed with some detail the case of port registers for shipping and the problem posed by the mention of a same journey in various documents (I-d2A). Population census and population lists, regularly repeated year after year in the same town, with a huge amount of overlapping redundant information, pose a similar problem. A detailed longitudinal analysis in their full complexity of all census referred to the same geographical entity, provides highly interesting insights on household structure and on the internal working of families which help understanding social behaviors. To make possible longitudinal studies of this kind, apart from transferring atomized data to the system, census must be preserved in a special database structure, close to that of the original document, and by no means compatible with the actions/actor model[70].

B. Special patterns. Some objects are socially processed in specific ways, which demand no less specific database patterns to be rightly accounted for. Tapestries, for instance, are based on a painting, which itself usually generates sketches, from which cartoons are extracted, forming together series which tell a narrative; some or all these cartoons are transformed into tapestries panels, some of them only once, others two, three or four times, generating in that way various sets of panels telling a same narrative in different ways. To account for such a complex set of relationships we must process cartoons, engravings, paintings and tapestry panels as if they were members of a same family related by filiation and brotherhood relationships, in a data table quite similar to the one we use for genealogies. Given that tapestries have hugely different properties than human actors, given also that tapestry making induces some specific rules, the tapestry genealogical table must be different from the normal genealogical table and specific to databases related to tapestries.

C. Sources containing stylistic information cannot be directly atomized into action (Section II-a4). They nevertheless must be stored somewhere and made accessible by markers indicative of their characters.

These reasons make necessary to distinguish three sets of tables or, better said, sets of tables organized in what we call subsystems:

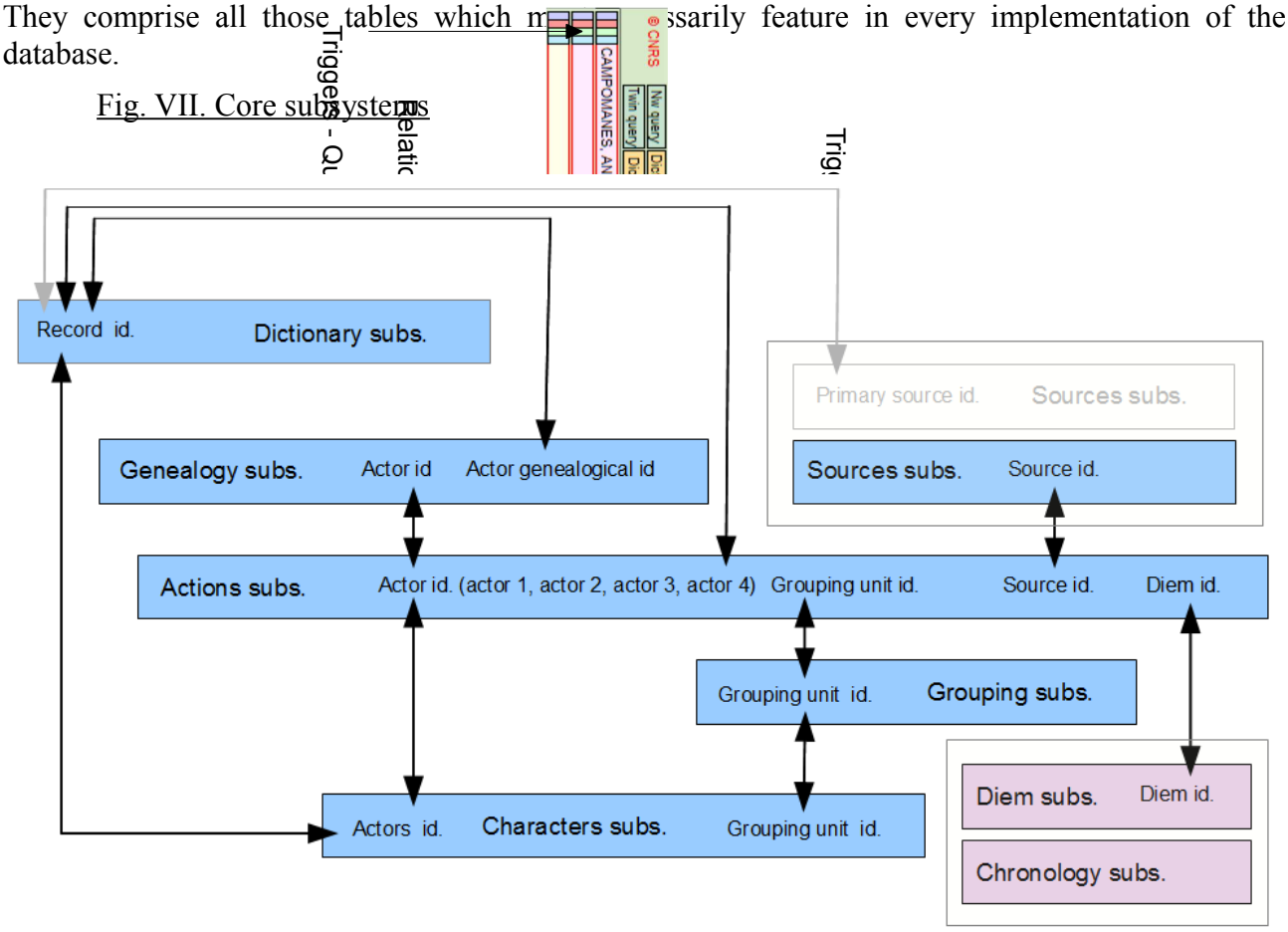
. Core subsystems
. Peripheral subsystems
. Trans-implementation subsystems.

70 See for instance the Charleville project, based on a database we personally planned: Rathier (Carole), Ruggiu (François Joseph), "La population ...", n. 5.

b) Core subsystems

They comprise all those tables which n ssarily feature in every implementation of the database.

Fig. VII. Core subsystems



Legend: subsystems set inside a same frame are stored in a same file. Except the Dictionary subsystem, all others are composed of a unique table.
Blue: central core subsystems
Purple: adjuncts

We distinguish among them central core su s (in blue in fig. VII) and adjuncts (purple in fig. VII). Central core elements are essential fo t working of the system.

1) The Actions subsystem

The main and more important subsystem ctions subsystem. We sufficiently described the concepts of action and actor before to be b account. It comprises one table only, composed of the eight fields we mentioned above (se 1 and II-a2) as essential to the action:

. [Who name] - [Who identifier]
. [Represented Who identifier]
. [With Whom name] - [Whith whom r]
. [Represented with whom identifier]
. [What]
. [Where]
. [Initial date]
. [Final date]

and, of course, a most essential field in databases: [Sources]. Moreover, a [Permanent

coding] field (see section III-c) and a [Record identifier] automatically set by the machine which numbers and identifies every record for internal and maintenance purposes.

The subsystem is equipped with various layouts, which allow displaying records - that is actions - in the most convenient way. The two most important ones can be seen hereunder (Fig. VIII and IX). The first one, the "Main" layout, displays as many actions as possible, to provide the contextual elements users need rightly to understand any action. Unessential fields are left aside, including [Represented Who] and [Represented With whom]. The second one, the "Expanded" layout, displays the same content complemented with extra non-essential but useful information, such as sources. It also displays (brown fields) descriptive elements of [Who] and [Whom], as given by the source in the context of the action[71].

These layouts, the same as all other Fichoz layouts, are equipped with a set of triggers which allow performing the most usual tasks in one click. This is a fundamental feature. We saw that when building up data, thousands of queries must be carried on. A gain of some seconds in each of them means hours and even days in the end. One easily imagine that screens packed with data displayed in so dense a way are not easy to manage. To make the task lighter, all layouts use colors, more exactly a same color code which marks, all over the database, with a same color, elements endowed of a same function.

The Action subsystem is linked to all other core subsystems, which contribute context data to enrich its content.

71 The description of the actor changes depending on the context. We found cases in which the age of a witness varied from 60 to 80 years, in question of days, depending on the topics on which he was giving evidence. We also remember an actor who was successively described as a silk merchant, a landowner and a person of independent means with an interval of three weeks, when successively making his will, buying an estate, and getting a position as a tax officer. Such variations are not errors or malfunctions, as many researchers believe, but the expression of various social personalities, a data which must be preserved for further analysis.

Fig. VIII. Actions main layout

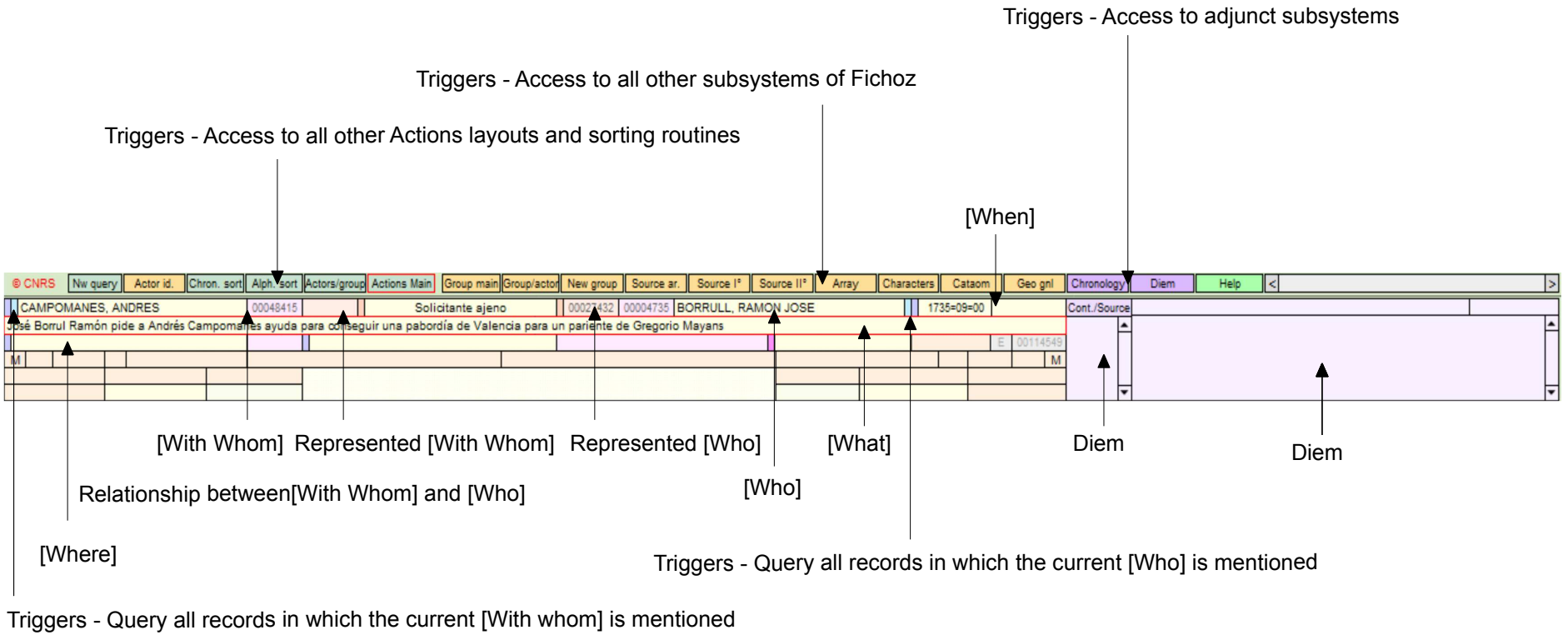
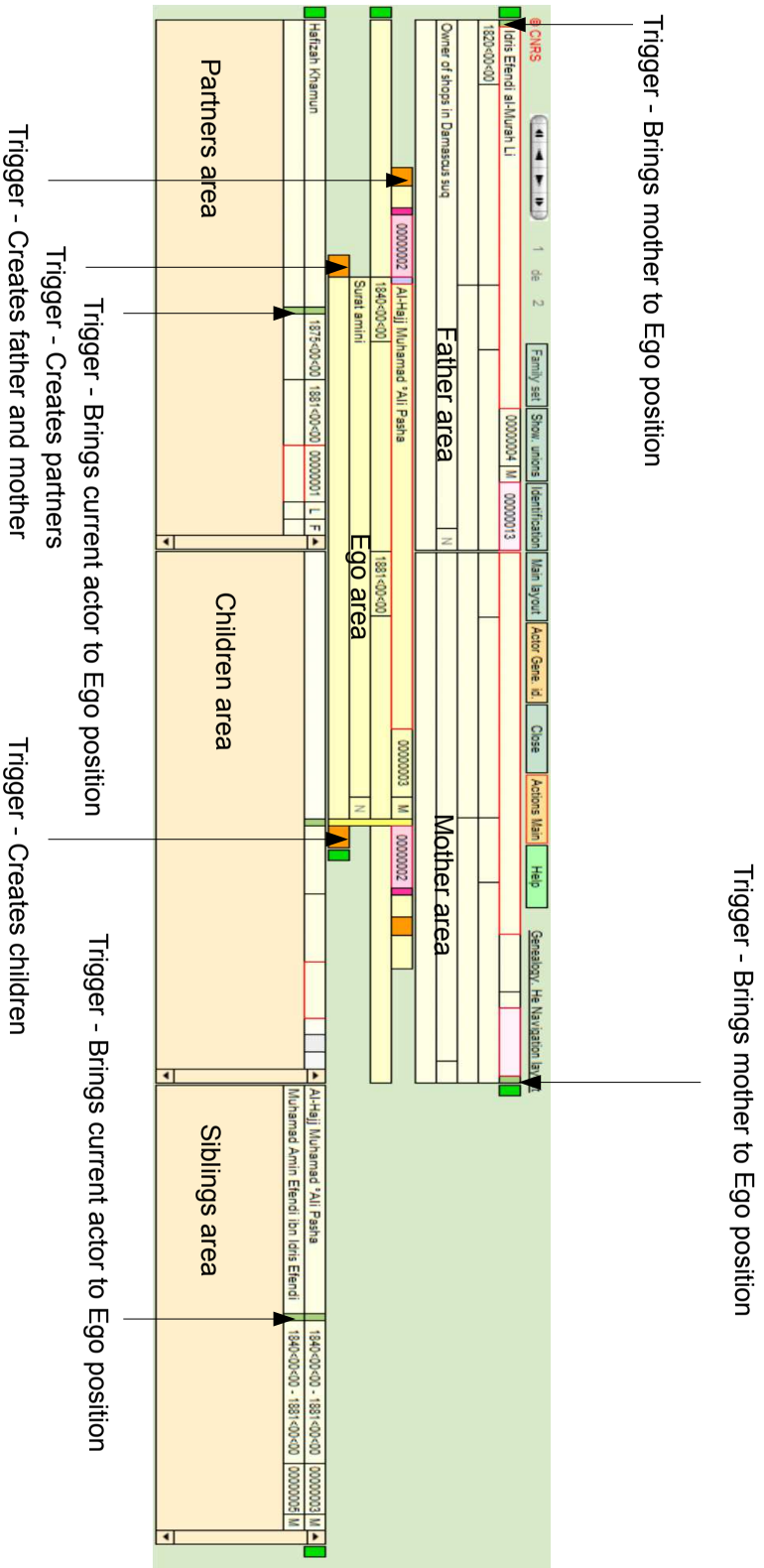


Fig. IX. Action expanded layout

2) The Genealogy subsystem

We already explained the function and main features of the genealogy subsystem (Example VI).

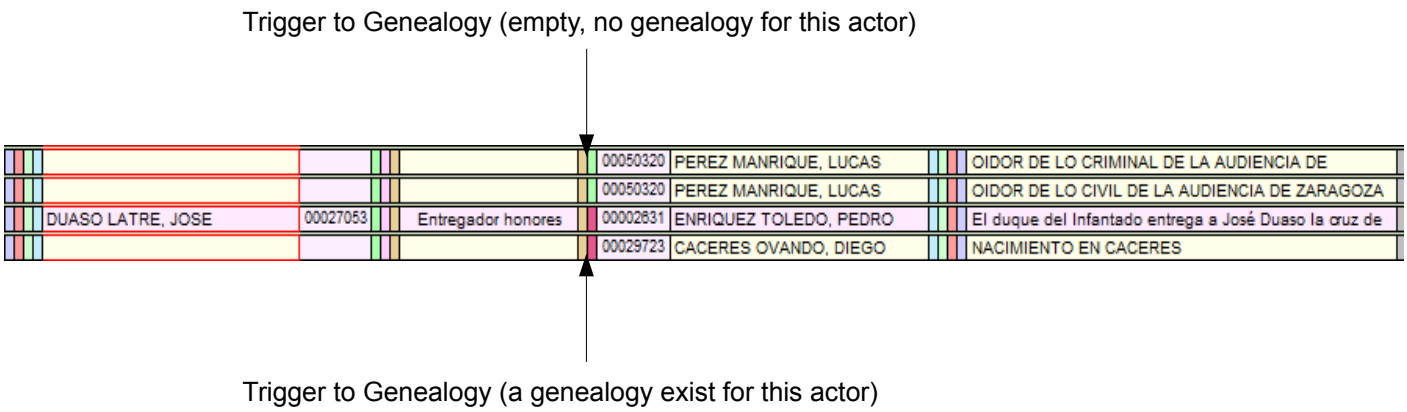
Fig. X. Genealogy main layout



Its main layout displays in the center data related to Ego, the actor on which the genealogy is currently centered (gender, name, birth date, birthplace, date and place of the death, specific identifier of the actor in the genealogical subsystem, identifier of the actor in the Action subsystem). Just above, on the left side, the same data referred to the father, on the right side, to the mother. On the lower part, a list of marital and sexual partners, a list of children, and a list of siblings (Ego included).

A little green trigger is set at the side of each name. It brings the corresponding actor to Ego position, and, obviously changes the display of all other sectors to this actor's father, mother, partners, children and siblings. The reddish field on both sides of Ego's name contains Ego's identifier in the Actions subsystem, if any. The deep red small trigger at the side of this identifier brings to the screen all Actions records which involve Ego. Conversely, all such Actions records include a similar red trigger, which displays Ego's genealogical data on the current Genealogy layout (Fig. XI).

Fig. XI. Trigger to Genealogy in Actions



As we mentioned before (Example VI) it is possible to mark Ego's relatives up to a given degree, and to pass the results to Actions. The Genealogy subsystem is obviously linked to the Actions subsystem.

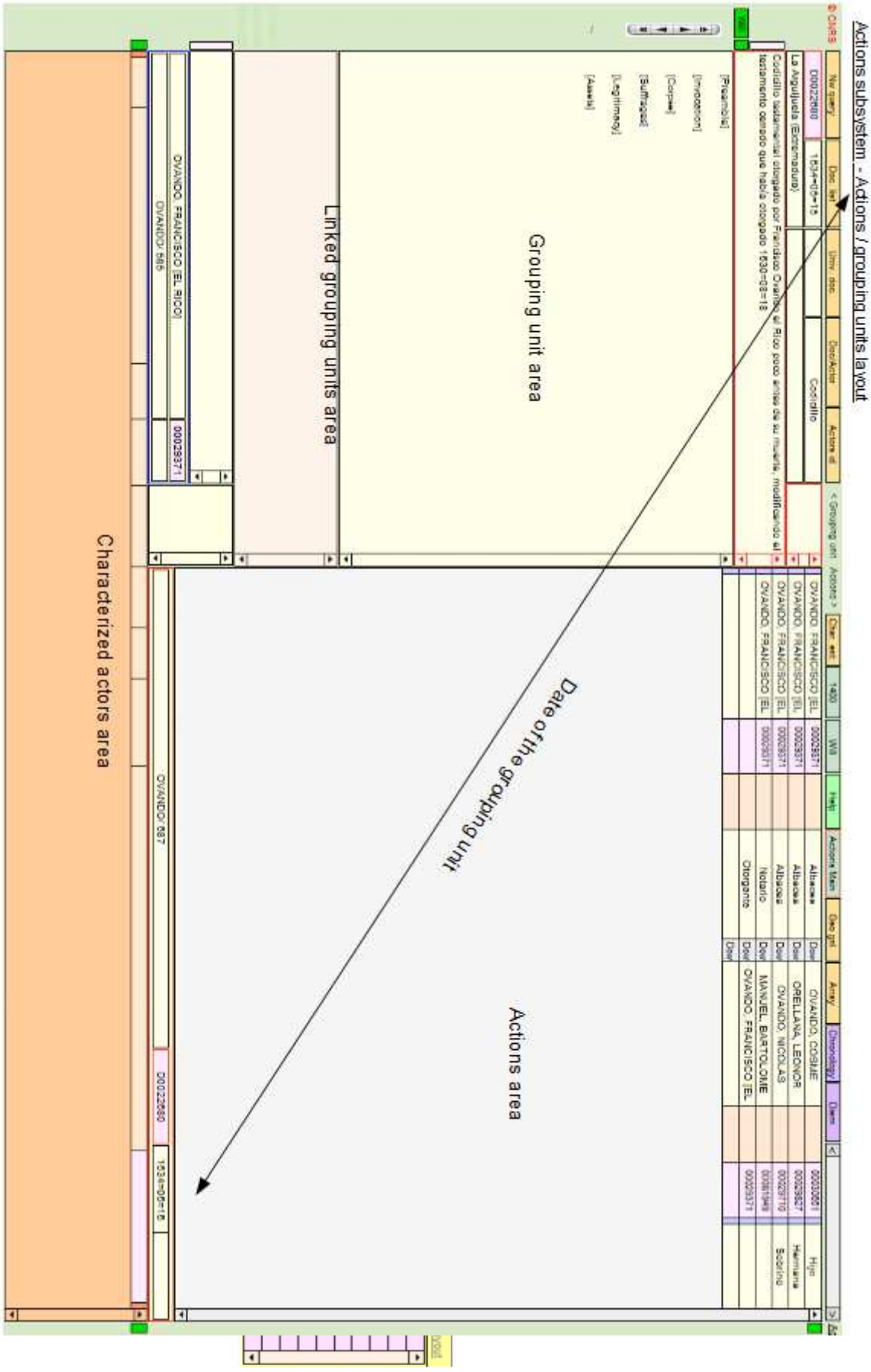
3) The Grouping subsystem

Each record of the Grouping subsystem (Section II-a3) stores a narrative which keeps together various actions. All these actions, brought together, tell the same narrative in a more sketchy way. The narrative stored in the Grouping entry may be a text, extracted from a source, telling what happened in such a place and such a date; a text written by the historian to account for a complex matter which actions alone would insufficiently describe; a legal document; or simply a void frame, which adds nothing to the sketch drawn by the actions, except that, for the mere fact of its existence, its knits them into an independent and identifiable unique object. Linked actions can obviously be accessed from the Grouping entry; and the Grouping entry from the Actions subsystem as well.

The Grouping entry displayed in fig. XII stores data about a will, the text of which we did not find, but the content of which we know through another source. The "Grouping unit area" is empty but could as well store the text of the document if we could find it. The layout displays a summary of the same just above. The Actions area lists all actions related to the will stored in the Actions subsystem.

Grouping is specially efficient at atomizing legal documents (wills, sales, powers, any kinds of contracts, birth and marriage certificates, etc.) into data.

Fig. XII. A Grouping entry and its linked actions



4) Characterization subsystem

An "Characterized actors area" features at the bottom of the Grouping entry display of Fig. XII. In this example, it is empty. If in use, it would display a list of actors mentioned in the narrative told by the Grouping unit. Data about such actors, of any kind (artifacts, corporations, individuals), would also feature in the Actions area. The Actions subsystem module is nevertheless unable to store a description of actors. This must be done in another subsystem, the Characterization subsystem, to which belongs this empty area.

The atom of the Characterization subsystem is not the actor, but the character assigned to the actor by the description extracted from the source: the subsystem holds as many records as characters mentioned.

Example XX. A black cat with a short tail

Describing "a black cat with a short tail" would generate four records:

. Nature: cat
. Color: black
. Tail: short
. Number: one

We may expand the description with a class character if we consider such an addition helpful:

. Class: animal

All entries related to a same actor are marked with its Actors identifier. A special entity linked to all of them keeps together all characters which describe a same actor and stores a full text description of the of the actor, if provided by the source. The most interesting feature of this process is probably that the number as well as the variety of possible characters assigned to any actor have no limits: the number, because each character is a record, and not a field, and as many records as wanted can be assigned to the actor; the variety, because no descriptive dimension is assigned beforehand by the system. Each descriptive record is in fact composed of two fields. The first one names the dimension the record is describing (in our example, Nature; Color; Tail; Class), and users are absolutely free to chose whatever they like. The second one gives a value to the current dimension (Cat; Black; Short; Animal), being users absolutely free to set the one they want to.

All these data can obviously be accessed, either from the Actions subsystem, to make certain of the characters of any actor involved in any action, or to select actions carried on by actors who possess such and such a character; either from the Grouping subsystem, for similar purposes. Fig. XIII (hereunder) displays the description of a painting.

5) The dictionary

The Dictionary is an essential, although almost wholly passive part of the system. The Dictionary table is linked to every other table in which actors happen to be mentioned. It is composed of a set of empty records, each one equipped with a serial record identifier. These record identifiers match all possible actors identifiers to be found in the database: some are composed of eight digits, like personal actors identifiers; some belong to the 000000C class, like corporate identifiers; some to the 0000000L class, like cultural items class; some to the 000000K class, which identifies all other artifacts, etc. When assigning an identifier to any actor in any subsystem, users switch to the Dictionary, copy an empty identifier of the relevant class and paste it to the relevant field of the relevant subsystem. By doing that, they automatically activate a marker in the corresponding record of the Dictionary which shows that the current record identifier is in use; and the name of the actor, as worded in the destination subsystem, appears in Dictionary along with the newly used identifier. The first function of the Dictionary is that of a tank of empty identifiers for actors.

Fig. XIII. Characters main layout

Its second function is that of coding. The database is structured by actions, the constitutive elements of which may be scattered over a variety of subsystems. Retrieving all actions assigned to a specific actor is an easy task. Retrieving all actions which match a defined set of conditions is also easy. In both cases the query can be answered by data contained in one record only, which makes possible an expedient selection of the same. It is far more difficult to select actors who match two or more conditions expressed in different records.

Example XXI. One record and multi records queries

One-record query:

[1] All actors who studied at Salamanca university

[2] All actors who got a bishopric in America

Two-records query

[3] All actors who studied at Salamanca University and got a bishopric in America

The Dictionary resolves the problem. Being linked to all records of the database in which an actor is mentioned, it can be accessed from any of those. In turn, all records of the database can be accessed from the Dictionary. We equipped the Dictionary entries with an extra field, in which to store markers. To select actors on the basis of multi-records criteria, we first select all actors who answer the first condition; we mark them all in the Dictionary's marker field; we then select all those who answer the second condition, and we add a second mark in the Dictionary's marker field. We repeat the operation as many times as conditions to meet. We then select all Dictionary entries the marker field of which contains the whole set of markers. This selection answers our original query. From the selected entries of the Dictionary we are able to access, if need be, all actions entries which match our query.

Example XXI. Multi-records query. A practical case

[Query]: All actors who studied at Salamanca University and got a bishopric in America

[1] Select all actors who studied at Salamanca university

[2] Set the marker "Salamanca" to all Dictionary entries which match any of the selected Action records.

The marker could be any string freely chosen by the operator. Marking is automatically done all over the set of selected entries by a special script.

[3] Select all actors who got a bishopric in America.

[4] Set the marker "America" to all Dictionary entries which match any of the selected Action records.

Some Dictionary markers are now: "xxx" (empty); some: "Salamanca xxx"; some: "America xxx"; some: "Salamanca America xxx", being "xxx" a meaningless marker used for technical purposes.

[5] Select all Dictionary records in which the value of the marker field is "Salamanca America xxx".

This is the result we were aiming at.

6) The Sources subsystem

Whatever school of historians you belong to, first comes the source. Each source must be delimited

in relation to others. A source provides contextual elements which give the information it carries its true meaning. In that sense, delimiting the source to which each information belongs is a fundamental task, and identifying sources is part of the basic training of the historian. This task is easy when the source is a legal document, or when it can be clearly characterized by formal criteria: even a beginner will recognize a will among the papers he is perusing; the same can be said, in most case, of a letter. Things may be more complex for many interesting - historically speaking - documents, such as trials. A same legal file may hold various embedded trials against a same person, some of them highly interrelated to one another - first degree trial and appeal for instance -, some of them far more loosely knitted, such as incidental affairs with slight bearing on the main one. If we define a source as a documentary unit the nature of which introduces interpretative constraints into the information it provides, each class of documents extant in the trial file should be considered as a different source: a writ of evidence is not a same source as a sentence or a writ of accusation, although they belong to the same archival file (see next paragraphs). Each one of these sources must be treated as an independent object. Conversely, the documentary system used by the researcher must provide a way to link all the pieces of information provided by a same source on an affair or topic so as to make them concur to the description of the same. It must also provide a way to link together various related sources which together contribute to the telling of a same narrative. An historical database must be able to process sources in such a way, when needed.

A source must be typified as belonging to a class. This typification implies a characterization of the information provided and introduces constraints as to its interpretation. A writ of evidence, for instance, means a specific orientation, in favor or against a determined party. Legal technicalities have a strong bearing on its content: for instance, before the XIXth century, legal writs tended not to mention female witnesses if enough male ones were available, given that female witnesses were considered as less valuable. A letter is a specific and highly complex object from a relational point of view. The sheer fact of sending it creates a relationship between sender and addressee. It creates or mentions other relationships between these two ones, either by transferring useful information from one to another, either by stressing mutual friendship and confidence. A letter also creates or conveys information about relationships between any of the correspondent and third persons, or between third persons only, independently of the correspondents.

Whole treaties have been written explaining how every one of the various kinds of possible sources recast information after its own needs and, by so doing, inject new information of its own as well as biases into the narrative it is supposed to tell. Identifying, mentioning and making clear to the user the class to which the source belongs is thus a fundamental step in database building. A description of the characteristics of each source and of the characters it transmits to the information it carries can be partly embedded into the information system. It will nevertheless be up to the researcher to complete such indications and to interpret the information correspondingly.

A source is, finally, characterized by its author. This is a fundamental point to catch its meaning. Various actors are liable to partake the authorship of a same document. A writ of evidence is the work of the notary as well as that of the witness. All such points must be carefully mentioned and made clear to the user, and the corresponding information embedded into the information system[72].

Summing up, a source is a knitted set of information, contributing all of it to a same narrative. The components of the source, which we call documents, have a same author and belong to a same documentary class. This documentary class injects into the information thus provided special characteristics and constraints which must be taken into account when interpreting contents. A source can be made of one or various documents. A writ of evidence, for instance, is composed of the sayings of various witnesses, each one being a document in itself. The document is the basis on

72 Which implies, by the way, that the name of the notary, and all available data about his person, must be as carefully recorded than the name of the witnesses, a point generally omitted in most historical studies.

which the operation of extracting information from the source, is based.

The Source subsystem stores all these data in one file. It provides, in a same table, two sets of layouts: one for archive sources; another for secondary sources (*vulgo*: bibliography). All entries of the Actions subsystem, of the Grouping subsystem, of the Genealogy subsystem and of the Characters subsystem are automatically linked to one or various records of the Sources subsystem by means of a short reference which, in the Source field of the linked entry, reproduces the content of a special "Short reference" field of the Sources file. A trigger allows users to get, from the linked entry, a longer description of the source.

Please, note that the peripheral subsystem "Primary sources" (see further) is also stored inside the Sources file, but that only some implementations activate it[73].

Sources layouts are typical bibliographical or archive reference layout, of the most usual kind.

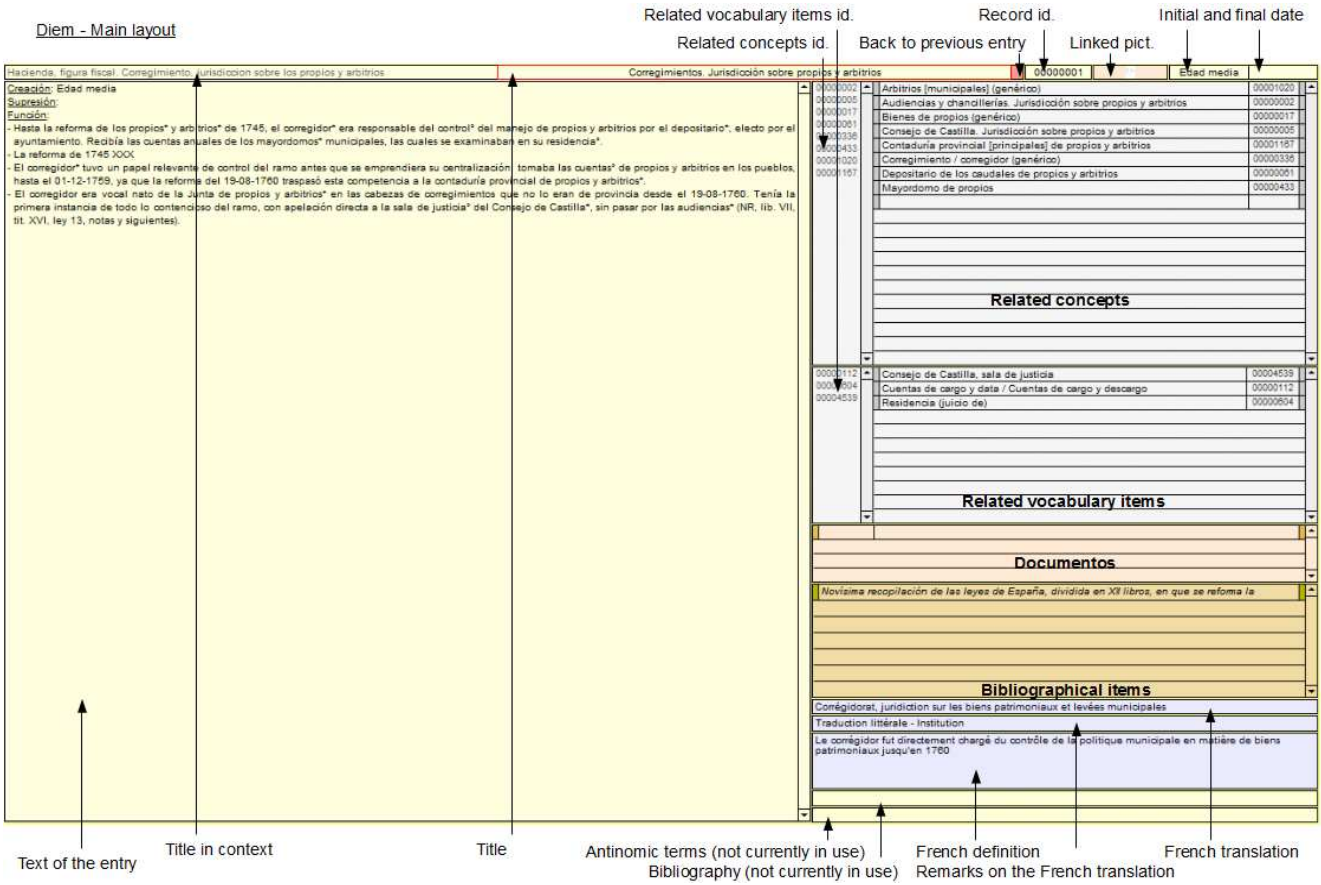
7) The Diem subsystem

The first six subsystems are central core parts of the system. Fichoz would not work efficiently, or even would not work at all, if any of them was lacking The Diem is not central core, and nothing would happen, from a mere computing point of view, if we suppressed it. We just would miss an important cognitive tool.

The database mentions a huge amount of events and institutions which only specialists know. We nevertheless stressed the importance of making the database a collective tool for research, which means that unspecialized researchers will necessarily have to use specialized information. The Diem bridges the gap.

73 This inclusion is a deliberate choice, given that many books may be indifferently used as primary or secondary sources and that storing them in the same file makes easier their processing in ambiguous cases.

Fig. XIV. Diem entry



The Diem (Dictionnaire des Institutions de l'Europe Moderne)[74] is in fact a kind of institutional dictionary. Institutional in the broader meaning of the word. Every concept, every legal entity mentioned in the database, must be described by a specific Diem entry: date of creation, date of suppression, function, insertion in the social and political context, main reforms, anteceding and succeeding institutions, should be the main heading of the description. Each entry can be linked to others, so as to provide combined descriptions of sets of institutions. Sources must be mentioned, and linked to the relevant entry. Documents (stored in the Grouping subsystem) can also be appended to the entry, as well as sketches.

Diem entries are progressively built and developed by users who specially know the problem each one is addressing. Each Actions entry can be linked to one or various Diem entry, providing in that way, starting from Actions, an easy access to usefull complementary information.

When fully developed, the Diem becomes a reference tool for a whole community.

From a technical point of view, the Diem is a table of a file which also stores the Chronology subsystem in another table.

74 Perhaps would it have been advisable to change the name in accordance with the geographical area covered by every implementation of the database. We decided not to, because: a) Diem (day, light, in latin) is a word we like; b) because changing the name of Fichoz files from one implementation to another would make more difficult an efficient management of the system. Let us forget the meaning of the acronym.

8) Chronology

Chronology is another non central core subsystem, which contributes knowledge, and as such we made part of every implementation. It is nevertheless quite unnecessary from a computing point of view.

Chronology stores dates of events, briefly described in each entry in the more neutral possible way. A special field stores a mention of the areas concerned.

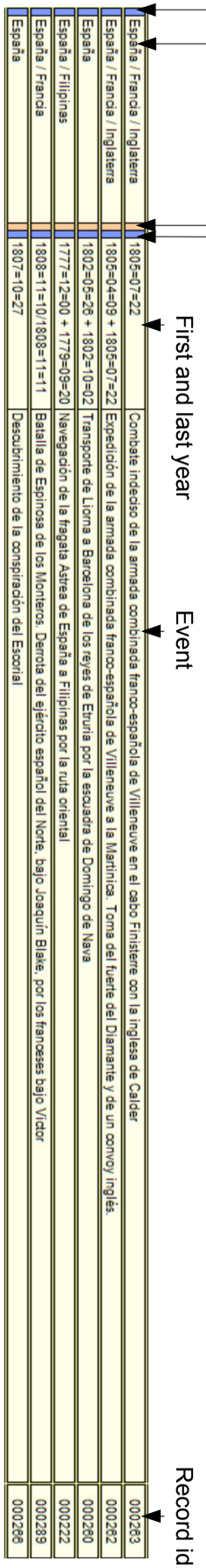
Chronology is helpful because many sources, specially non-administrative documents, use to date events not absolutely, but in relation with another event known to the actors; the exact date of which is not always easy to find, even with the modern resources of Internet. Each user must add to Chronology every time he determines such a date.

Chronology is a table of the Diem file, although it does not maintain any essential relationship with the Diem.

It has no link with any other table.

It can be opened from any layout by means of the "Chronology" purple trigger.

Fig. XIV. Chronology main layout



c) Peripheral systems

Some twenty implementations of Fichoz are presently running all over the world. They process data from the huge variety of sources. All these sources, whatever be their nature and structure, must converge towards the eight core subsystems we just described and must contribute data to the same. Experience nevertheless showed that many sources cannot be atomized and equipped with identifiers without undergoing heavy preprocessing. See for instance, section I-d2A, about shipping databases. Some others possess special characters for which an atomization based on actions and actors only imperfectly accounts (see for instance section II-A5 and Chateaubriand's text). The original information, in all such cases, must be preserved in special repositories and made accessible from the core when needed.

Depending on the nature of the sources to be processed in each implementation, we consequently add to Fichoz's core a variable set of *ad-hoc* subsystems, each one specialized in the preprocessing of a family of sources. They make easier the input of original data; they make far easier, and in some cases they simply make possible, atomization, identification of actors and the purge of repeated information. This being done, they pass purified data to the core and maintain with the records they helped to feed a permanent link.

It is obviously impossible to list all peripheral systems, the list of which is constantly changing. It is even less possible to describe them all in this paper. We'll just give a brief description of some of the most usual ones and refer interested readers to Fichoz's Help subsystem It can be accessed on the net (see Appendix) and contains a full description of every part of the system.

1) The Shipping set

As we said before(I-d2A), shipping implementations store and process data about shipping travels; fundamentally lists of points which a given ship was observed crossing on a given date, loaded with a given cargo, bound for a given destination, proceeding from a given port, paying a given tax amount (tax-gathering is a main purveyor of information), involving given individual and corporate actors in the proceeding. We already stressed the specific characters of the information which such sources provide and the steps which we had to take to turn it practical. We list here the main specific tools to be appended to the core system as a help for data collection:

To collect raw data, shipping aggregates three main tables to the central-core set:

. A Points table, in which all points mentioned by the source feature such as they appear in the original document;

. A Cargo table, in which every cargo item forms an entry and is linked to the point about which it has been mentioned.

. A Tax table, in which all taxes paid are mentioned, and in the same way linked to the point in which they were paid.

Actions and actors involved in all these processes are stored into the central-core Actors table, and also linked to the point in which they took place. Once raw data of the three extra tables have been tailored to a practical shape (elimination of duplicated information, coding, identifying, etc.), we still are uncertain of subsequent operations. It is most probable that in the end, we choose to import most of the purified point data to the Actions core-subsystem, but that we shall also preserve the three appended tables and use them in accordance to the needs of specific research operations.

The Cargo section of shipping poses an arduous problem. It mentions a huge variety of products, some of which are rather difficult to define. Moreover, it names them in a variety of languages. We consequently decided to add a specific Dictionary of commodities, in which we describe each of

them and provide the name they have in all languages used in the database, coordinating the matter around the English version of the name.

Understanding travels means understanding shipping conditions of the routes covered by the ships mentioned in the database. We created a special Dictionary of stages, in which each segment of a route is described in accordance to its shipping conditions[75]. Both tables do the part which the Diem table does in the central core, but limited to shipping implementations.

2) The Census set

The Census set was planned in a first moment to pass to the computer data provided by the original lists of inhabitants made as a first step for demographic census. They record each inhabitant of every house of a given area, grouped by household. They usually mention their names, ages, office, gender and role in the household. They may be the result of national census, repeated at (usually) regular intervals; or they may be municipal lists of inhabitants, normally revised every year. We may aggregate to this class yearbooks, which are census of professionals, obviously not so exhaustive from a demographic point of view as demographic census - they only mention professional and say nothing of their families -, but with a great wealth of economic data. All these sources have in common the fact that they exhaustively describe the universe on which they are based; and the fact that they periodically repeat a same description of the same objects, thus making easy the detection of changes.

They are highly interesting on two heads:

. a) As a source to identify unknown actors: being exhaustive, they provide data on unimportant persons which other sources mention casually, without any detail, making them names without a content;

. b) As a unique source for longitudinal studies; which means following a same actor all along his life course, and considering the actor no as a fixed, dead entity, from which can only be extracted, fixed characters deprived of context, but as a person with a history, being the characters observed at a given moment time-dependent on previous actions and characters[76].

Both objects mean the management of huge amounts of data. Selection and sampling simply do not work, in case (a) because you don't know beforehand the data you'll need; in case (b) because sampling means breaking the chronological and social continuity which is precisely what researchers are looking for in such sources.

Massiveness raises two questions which prevent researchers from taking a full advantage of such documents: data input and the identification of actors. A collective use of databases resolves the first point. A simple management of identifiers, the setting of which is done manually by researchers, once for all - which on the long term means saving time - and backed by the whole power of the data base, resolves the second one.

All entries of all lists are loaded to the database, each as and independent record, equipped with all the data mentioned by the source. Sorting then brings together similar cases. Identical items being brought together, setting identifiers in all the dimensions described by current data (name, age,

75 Numerous books of nautical instructions published by a variety of national bureaus in the XIXth century provide a huge amount of information on that point, gathered fromt the point of view of sailboats.

76 For a good and early example of the possibilies of such sources, see the ground-breaking book of Pinol (Jean Luc), *Les mobilités de la grande ville : Lyon fin XIXe -première moitié du XXe siècle*, Paris, Presses de la Fondation. nationale de sciences politiques, 1991, 432 p. The importance of time-dependance was stressed by the final conclusion of the famous controversy on the the size of household, initiated by Laslett's work in the 70s of the last century, precisely based on census lists. See: Courgeau (E), Lelièvre (E), *Event History Analysis in Demography*, Oxford, Oxford University Press, 1992. For an up-to-the state of the are use of such sources, see the above mentioned papel of François-Joseph Ruggiu (note 5).

address, occupation, gender, role within the household, date, etc.) becomes easier. Researchers finally mark duplicated records (same data in two successive census) as redundant. The residue is a list of actions and changes of position of the actors, which can be transferred to the Actions subsystem. A permanent link makes possible to access from the Actions subsystem all the original stuff from which the action was elaborated by this refining process.

3) The Array set

Many sources provide quantitative data which cannot be efficiently processed except as arrays. Most historian presently use spreadsheet packages, such as Access or the equivalent OpenOffice. We already made clear that spreandsheets are fantastic calculation and analytical tools (see: I-b), but very poor storage instruments (see: III-a-2). What Fichoz needs is a tank where to store data. Spreadsheets obviously do not work.

So that to manage arrays, we created a specific FileMaker file, which in Fichoz we name "Array", which (a) describes stored data apart from storing them[77]; (b) allows storing as many arrays as necessary into the same file, thus resolving a serious problem of possible data mismanagement[78]; (c) allows accessing any cell in accordance to given criteria, from any other table of the database. We join this Array table to any implementation which needs to manage arrays such as election data, balance sheets of firms and the like and we link the relevant actions to the matching cell of the array.

d) Trans-implementation subsystems

Three subsystems are of so generally used that we cannot consider them as dependent of any specific implementation. The first one is the Help subsystem which, apart from a detailed description of every part of Fichoz, provides clues as to strategies to be implemented to input, process and explore data. The second one is the Geo_general subsystem, a gazetteer which provides the latitude and longitudes in decimal degrees of millions of places around the world, as well as many name variants, tools for an intuitive location of the same and a unique identifier for each one. The third subsystem, named "Geography", provides lists of places arranged in function of the various administrative districts they were part of in ancient and recent times.

1) Help

Help is a huge and complex file which describes every file, every table, every layout, a large number of scripts and routines, as well as many other processes, the idiosyncracies of every implementations and all special concepts on which Fichoz lays. Don't use it as a first approach to learn Fichoz basics. Manage it as a reference tool when you forget how to do something, how to write a date, or a class of names. Use it also to make concepts clear. If you have doubts about what actors are, for instance, Help will provide more detailed considerations than in the present paper.

Help entries are interrelated: each one is equipped with triggers which give an access to other related entries, either conceptually related ones or mere linguistic explanations. They look very much like Diem entries (see Section IV-b7, fig. XIV).

Users access Help from any part of Fichoz by activating the deep green "Help" trigger affixed to the Header of every layout. Each trigger activates the Help entry which describes the layout from which the query started. If that is not the entry you are looking for, activate <Ctrl 1> and formulate your query in the relevant field. When doing so, you will be able to switch back to the entry you started from by activating the red trigger set in the first line of the entry, exactly as in the Diem subsystem. If you have a question on how Help works, activate the Help green trigger from any Help entry.

77 The array file is composed of two tables: on for description; one for data proper.

78 The data table is composed of as many fields as columns exist in the broadest possible array. For each possible value, a specific layout is called which displays the relevant number of columns

Many sketches have been appended to Help entries. Use them freely!

All Fichoz implementations access a same and unique Help file, written in English, presently stored at the Humanum server of the CNRS (fm.tge-adonis.fr). You must be on-line and connected to Humanum to access, even if you are working with a local version of Fichoz.

2) Geo_general

Geo_general is also stored at Humanum and users must also be on-line to access it. This can be done from almost every Fichoz layouts which display places, by means of a "Geo. gnl" brown trigger usually extant in the header of the same. It is a monstrous gazetteer of more than 3,5M entries (January 2014), and probably more in a near future. It is based on NGA on-line data, which we reorganized to make them really manageable.

Each entity mentioned in the NGA source databases is a record of Geo_general. One of the most positive points of NGA databases and an appreciable asset when multi-lingual areas are concerned (most areas of the world are multi-lingual) is that it mentions various linguistic versions of a same name. Geo_general is able to display as a same visual bock all known versions of the name. It choses one of them as a standard denomination.

Each entry, that is each version of the name, that is each record, is given an identifier during the input. Those identifiers, all of them of eight positions, begin with a letter indicative of the area concerned: A0000015, for instance, is a point located in Europe or in the Mediterranean area (see Help for further details). The identifier of the standard denomination of the entity is what we call the "UHGS" [Universal Historical Geographical Identifier] of the same, and must be used in all Fichoz implementations to identify the place when mentioned in any Fichoz file.

Every Geo_general entry is also equipped with the coordinates of the entity, longitude and latitude, in decimal degrees, as given by the NGA[79]. This data can easily be retrieved from any Fichoz, through the UHGS identifier. In such a way as to make mapping an easy task.

Data can be easily changed in order to make localizations more precise, to add new points and new variants, and so on.

3) Geography

Under construction. Will be described in future versions of this paper. Makes the management of polygons as easy as that of points. This is specially interesting, because polygons are, among other interesting features a description of administrative districts, and object specially difficult to manage from a historical point of view due to its lability.

79 Which raises a slight problem. NGA data were elaborated at a time when computers could not easily process long strings of decimals; so that number are significant to the second decimal only. Which in practice means, in European countries, a margin of error of around one kilometer on every side in the situation of any point. For instance, Prat-Bonrepaux, the author's birthplace, goes mixed with Lacave, quite a different village, two kilometers away. Small scale maps are unaffected. Not the same large scale ones.

Concluding remarks

This introduction to historical research databases is grounded on the ones I personally created with the help of Spanish, Chilean, Italian, Belgian and French researchers, mainly as part of two huge research programs, one on the political actors of XVIIIth Century Spanish Monarchy[80], another on shipping movements of the XVIIth and XIXth century[81]. It was a long process, extended over more than twenty years. We did not plan beforehand to make an all-embracing system. We started (in 1988) from the need to computerize a paper file of appointments to positions of the Spanish royal administration, and we elaborated for this task the concept of atomization. As the result was good, the research program was progressively extended, and the system consequently developed. Many paths were explored and left aside. Concepts were elaborated to account for what functioned and to make possible transpositions to more cases than the one for which a correct solution had been elaborated. Every part of the system was created and tested in field-work situation, to answer the needs of and to be managed by operators who, for the most part, were absolutely devoid of any previous computing ability. The way Fichoz was elaborated accounts for the two main points which can be said on its behalf.

First all all, it works. At least twenty books and 200 papers have, till now (2014), been directly based on Fichoz data, many of which could not even have been contemplated without the capacity it provides to manage huge, complex and unplanned for sets of data. It works because it demonstrated a high degree of versatility to cope with unforeseen situations. This is the second point. Such a versatility derives from the fact that is is based on a reduced set of principles, which we exposed in the first part of this paper. This strong rooting in principles, and not in formal or technical details, enables the system to adapt itself to almost any kind of demands. On the present day, apart from the databases on early modern Spain and Navigocorpus, half a dozen other Fichoz implementations are working on subjects so different as can be the history of French aircraft industry, intellectual relationships between Europe and China in the XVIIIth century, or Muslim religious foundations. Fichoz was even able to process Roman inscriptions, a task it had absolutely not been planned for, with only minor changes, and some methods elaborated on that occasion were later imported to the system as a whole[82]. Even biologist and specialists of the physics of materials engineering have been interested in the global structure of the database to store detailed results of analysis: they have in common with the historian the problem to eliminate meaningless noise from their observations. Versatility, I insist, is a master concept in any tool for scientific research.

The technical side of the business is as important as the conceptual one. We had to decide a huge set of conventions on how to write data. Flexibility has its counterpart: complexity. To make the system manageable we had to program a lot and provide easy ready-for-use routines to execute the most usual tasks. Such routines are launched by the colored triggers we alluded to when describing some layouts. The way data have been stored is in itself independent of the package (namely FileMaker). The set of tools we created to manage them is not, and should probably have to be written anew if the underlying package was changed.

We mentioned at the proper moment that a basic function of a database consists in transforming information into data. We stressed that such an operation is specially tricky as far as historical data are concerned. It demands, among other requisites, setting provisional and imperfect information in context, to let the operator decide the correct interpretation of the same. We also alluded to the fact that the last stage of the analysis process is the researcher's mind and judgment. In most cases, final analysis does not require any specific package, but is done by perusing a set of displayed data and

80 See n. 1.

81 See n. 5.

82 See n. 5.

drawing conclusions from the same. Everything must be done to alleviate the burden set upon the researcher's eyes and mind when perusing huge amounts of data displayed to the screen. We had to create dozens of layouts to help users to find their way among huge sets of displayed graphic signs. This part of the job was the most time-consuming and not exactly the most gratifying. But it was as necessary as elaborating the concept of action or conceptualizing the Grouping susbystem, both of them moments of intense intellectual excitement. It was impossible to trust the operation to technicians. The result had to match the researcher's requirements in such a way that only researchers, and experienced researchers, could do it.

The same is true of inputting data to the database. Boring in most cases, time-consuming, always. And a job for skilled researchers. The more so because a good research database is necessarily complex. Not in itself. Just because the data it is processing are complex, and making them simple by discarding dubious and badly fitting elements does not work. First point. Moreover, the value of a database increases as its capacity to put data into proper context increases. Which means that the bigger the database, and consequently the more complex, the more efficient it is. Second point. Third point, we know by experience that the only way to bring together researchers working on different subjects, periods and areas consists in linking them to a same database. Scientific benefits are impressive. Last and fourth, Fichoz stores data in such a way as to make them directly available for any research program.

Conclusion: a database must be a collective venture. But to understand righly such an assertion, we must be aware of the meaning we give the vocable "database". Although a same word, it points to two quite different classes of entities:

> . The first class is the kind of "data-building" database we described here above. It is a tool. It is not, nor can it be, a fixed ready-to-use set of data. Any aggregation of new items changes the contextual setting of all others and make them different. Such lability is specially obvious when sources themselves provide fuzzy, partial information, or contemplate a same information from various point of view, as in the shipping databases. The same lability also exists in biographical datasets of the Fichoz kind, in which a new data piece, a new relationship, sometimes change the whole meaning of a biography.
>
> . The second class is a fixed and permanent set of ready-for-use data, never to be changed, which an author puts at the community's disposal for any use users thing fit[83]. Data storage, not data-building, is the point.

Both kinds are essential for research, and both must be in some way collective ventures. Both, nevertheless, demand quite different sets of management rules.

> . Stable data storage raises fundamentally questions of access. Procedures, file structures, even software kits can and, up to a point, must be made as uniform and as simple as possible to provide an efficient access to the broadest possible audience. Descriptive instruments can and must be elaborated and published. Technical considerations play in this universe a fundamental part. Provisional imperfect solutions to questions of access and storage, which would make the database not so good on some respects to maximize other factors, may be tolerable as far as they do not affect the data themselves.
>
> . Tools for data building raise fundamentally questions of cohesiveness between information and approach, questions arbitrated by the rules which govern the practice of a research community. Procedures, files structures, the choice of software tools are, and must remain,

83 By ready-for-use we obviously do not mean that users should stick to using the data without a critical assessment of the same; nor that they are not free to try re-arrangements of the same to extract funderlying information. All those who, among our readers, ever intended it, are probably aware of the strict limits imposed by the closed character of the data provided.

totally dependent of the absolute necessity to maintain such a cohesiveness. Complexity is not a problem, rather a quality as far a it is necessary to preserve cohesiveness. Imperfect provisional solutions, in this universe, are absolutely out of question, because they would by essence affect not only the quality, but the veracity of the data. The paper of the engineer must be here resolutely subordinate to that of the researcher. Public access, even reading-only access, must be limited to specialists: they alone master the set of hermeneutics rules which qualifies them rightly to understand the provisional, moving, unstable, oriented and incomplete character essential to the data provided.

The question is: how do we manage both sides of the question at a same time?

. As a first point, we shall remind that all permanent ready-for-use databases are themselves based on data-building databases. Statistical arrays which describe the demographic components of a population are the result of a complex process of elaboration of raw data. They mirror in some manner the state an elaboration process reached at a given moment. They are fixed partial concretions of essentially fluid processes of the kind we described in the bulk of the present paper. Different states reached at different stages of elaboration may be considered fit for publication for different purposes. As far as census are concerned, even raw data could be published - we mean the manuscripts forms filled by census agents when interviewing inhabitants -, because these forms are themselves administrative documents, built as a closed universe, and no really mere information (I-a, to I-c) from the point of view of census.

. The question then is no longer: to what extent can we and may we make a data-building database available to public use, but how to transform a data-building database into a fixed ready-for-use data provider? This is a point which we are not in condition to answer by now. The fact is that researcher have little experience in this field. Data-building databases are a rather new field for them. They practiced data building for a long time, of course. We could even say that it is one of the few operation all research fields have in common. But they used to do it not only without computers, but also on a mere private basis. This part of the job was private matter. Not even the most detailed research reports could give an exact idea of the wealth of minute decisions researchers make at every moment to shape their data one way or other. Computers make the matter of collective interest. Quite a new situation. For the moment, let us go and see. We shall try and write provisional guidelines once we get more experience.

. A last question, derived of this newly acquired character of data elaboration is that of the chronological extent to be given to the database. The ideal situation would be a unique huge database covering all periods from Mesopotamian antiquity to present day, from China to Greenland, from Patagonia to Cape North. Breaking history into separate parts, on a chronological or geographical basis, is in itself a fault. After all, Aristotle was probably the most important thinker of European XVIIIth Century. No technical reason makes such a dream impossible. A well planned database is perfectly able to manage so broad a range of data. Cognitive reasons nevertheless offset the possible benefits of such an undertaking. No human mind is able to dominate the variety of languages and the knowledge necessary not only to understand the data, but also to atomize them adequately. We nevertheless think possible the create collective databases centered on broad periods and a geographical areas, managed by groups of specialists from various fields. They pose new legal and organizational problems. To make such an endeavor possible is the challenge we are confronted with now.

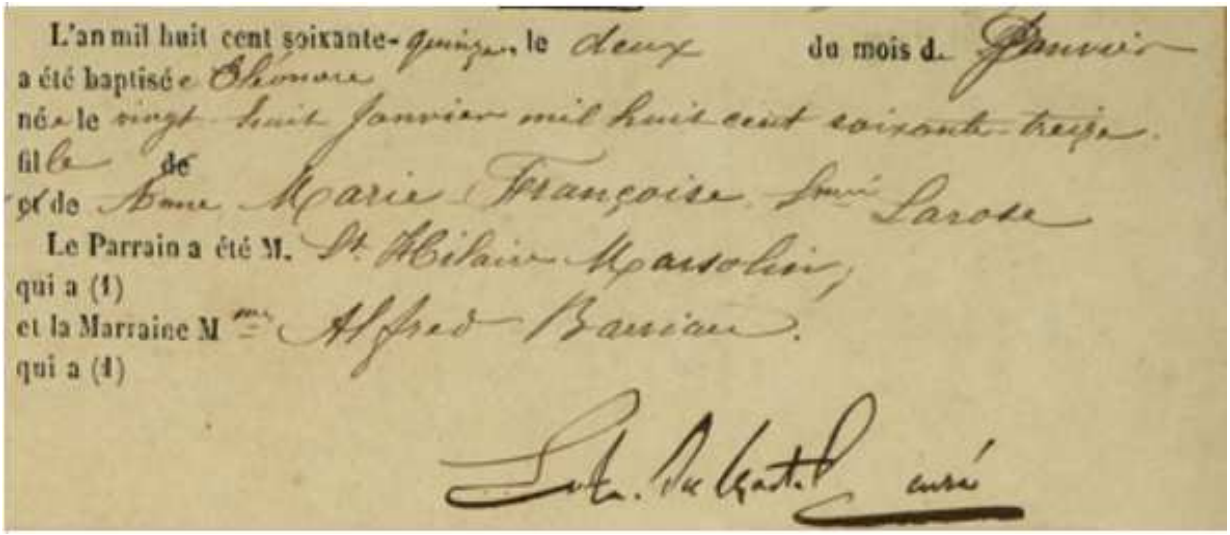
Appendix - The concept of relational tables

Fichoz processes data by means of related tables and conceptualizes them accordingly. The concept of related tables is not specifically ours. In its most general form, it was the foundation of database making since databases exist, even before the computer era. We did not think necessary to explain it in the body of this paper, because we thought all our readers would know it. On second though, given that our conception of data process is wholly based on it, we decided to add a brief explanation of the same as a help for unspecialized readers.

I. Flat databases: an avenue to nowhere

Let us imagine a database of baptisms celebrated in the year 1875 in the parish of La Trinité (French Martinique). Parish books, at that time, were fairly normalized by an efficient ecclesiastical organization, which imposed very strict criteria of administrative good practice on its members.

Doc. I. First entry


L'an mil huit cent soixante-quinze, le deux du mois de Janvier
a été baptisé Thomas
né le vingt huit janvier mil huit cent soixante treize.
fils de
et de Anne Marie Françoise dite Laroche
Le Parrain a été M. St Hilaire Marsolin,
qui a (1)
et la Marraine Mme Alfred Bauvian,
qui a (1)

Like a vast majority of our fellow researchers, we decide to use a spreadsheet. It is simple! Database packages are so complex, except when you use them as spreadsheet emulations... One baptism, one line. All data displayed on the same line belong to the same entry. Cells? How many? Let us have a look a the document: date, place (not mentioned, but implicit), name of the vicar, name of the child, name of the mother (no father; Tropics, you know...), name of the godfather and of the godmother. All right: seven cells:

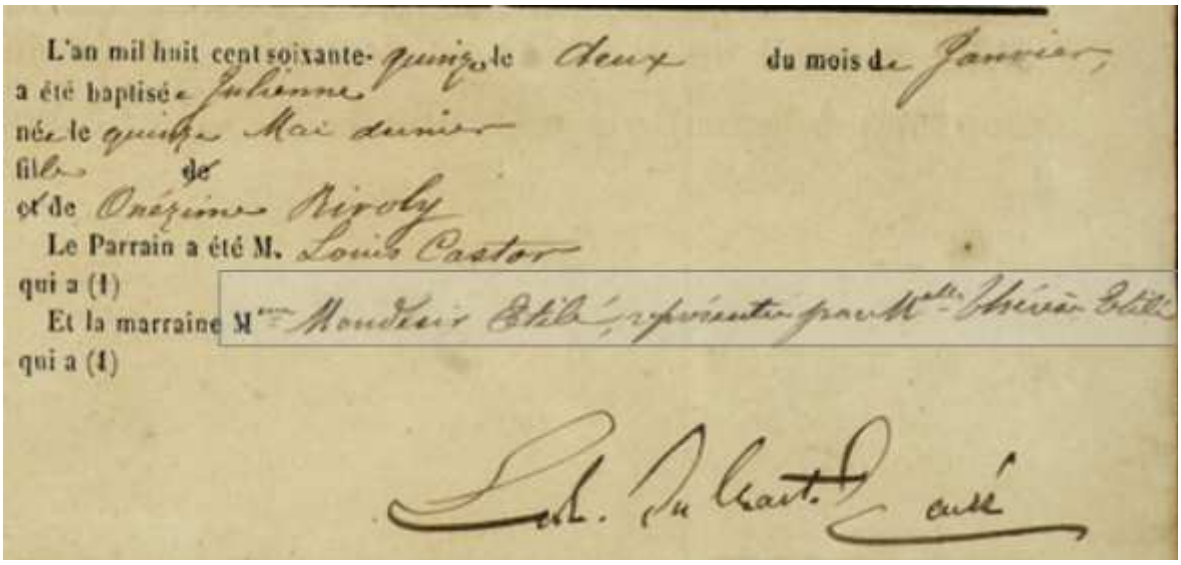
Fig. I. Spreadsheet, first version

A	B	C	D	E	F	G

A: date
B: place
C: child
D: vicar
E: mother
F: godfather
G: godmother

Let us proceed to the second entry:

Doc. II. Second entry


L'an mil huit cent soixante- quinze le deux du mois de Janvier,
a été baptisé Julienne
né le quinze Mai dernier
fille de
et de Onézime Riroly
Le Parrain a été M. Louis Castor
qui a (1)
Et la marraine M^me Monsieur Etté, représentée par Mlle Thérèse Etté
qui a (1)

By Jove! Our scheme no longer works! Godmother did not attend, and sent somebody else to act on her behalf (grey area)! Let us add a cell, for the delegate. What Godmother does, Godfather will also do sooner or later. Let us plan beforehand a field more for his possible representative.

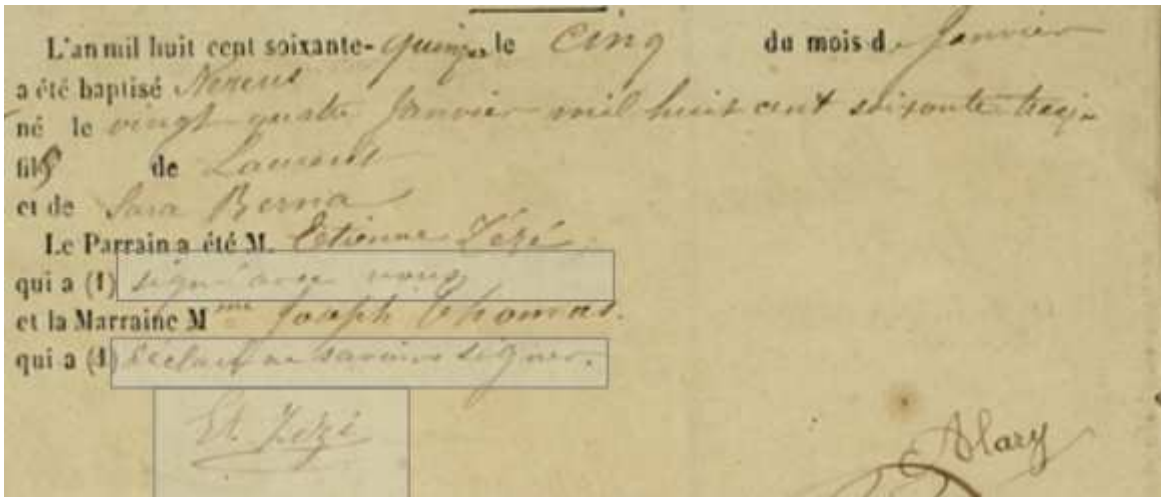
Fig. II. Spreadsheet, second version

A	B	C	D	E	F	H	G	I

The same as before, and:
H: Godfather's representative
I: Godmother's representative

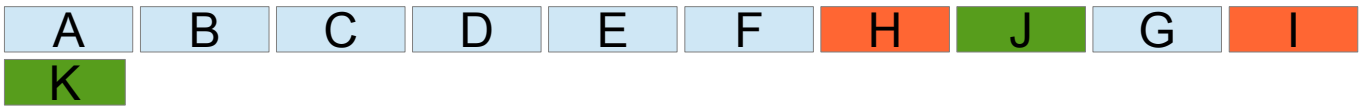
At that point, we bless the Church for setting a limit of two to the number of godparents. But why did they want them to sign the book (Document III, grey area)?

Document III. Third entry


L'an mil huit cent soixante- quinze le cinq du mois de Janvier
a été baptisé Renaud
né le vingt-quatre Janvier mil huit cent soixante-cinq
fils de Lauvin
et de Sara Berna
Le Parrain a été M. Etienne Lépi
qui a (1) signé à la mairie
et la Marraine M^me Joseph Thomas
qui a (1) déclaré ne savoir signer.

Let us leave aside the vicar's signature: no information conveyed, a vicar is necessarily literate. Godfather's and Godmother's signature do carry information. Two fields more.

Fig. III. Spreadsheet, third version



We are growing sort of worried. Still a hundred pages to go. What will the next surprise be? What if they give us the name of attendants, altar boys and singers? How many cells shall we need? [H], [J], [I] and [K] are rarely used, but we are obliged to preserve them even when empty. How cumbersome! Something's wrong. We are going to nowhere.

II. Related tables. A path to Heaven

Let us reverse our approach. Why should not we make each baptism a column, and arrange characters one under the other? Each one would be a record in the spreadsheet. We may create as many records (lines) as wanted, or as few of them when needed This would solve the problem.

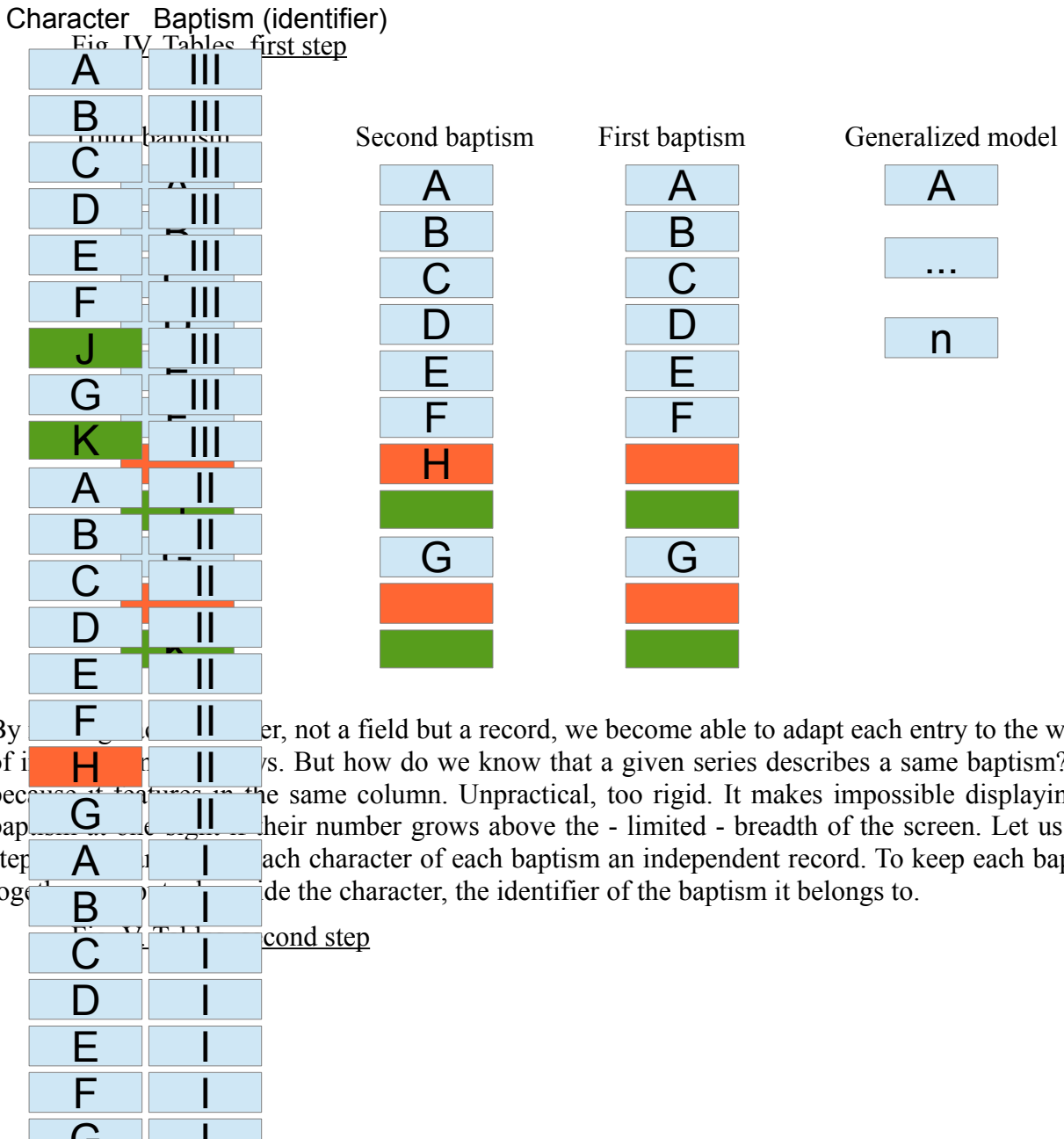


Fig. IV. Tables, first step

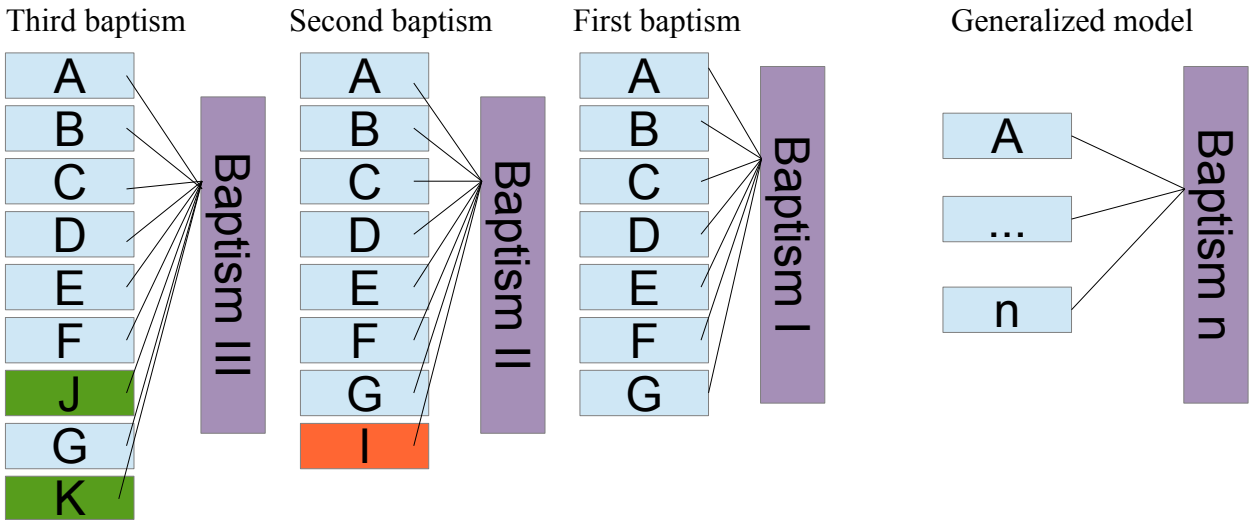
Fig. V. Tables, second step

By [illegible]er, not a field but a record, we become able to adapt each entry to the wealth of i[illegible]s. But how do we know that a given series describes a same baptism? Just bec[illegible]he same column. Unpractical, too rigid. It makes impossible displaying all bap[illegible]heir number grows above the - limited - breadth of the screen. Let us go a step [illegible]ach character of each baptism an independent record. To keep each baptism toge[illegible]de the character, the identifier of the baptism it belongs to.

Far better for data loading and retrieving. We are even able to complete, if needed, a partial entry long after loading a first set of data. But as far as display is concerned, we must find a more efficient way. Something must hold together all records belonging to a same baptism, and take the place of the common belonging to a same line or column, or to the indefinite repetition a the same identifier in all entries which presently do the job in our flat model. Something external. In fact, a flat database model (fig. III and IV) is like writing all data about a same baptism on a same sheet of paper. The new model we are after (fig. V) is like writing each piece of information on an independent sheet. We gain in flexibility, we can easily classify again and again our data as we like to, we can easily make independent heaps of similar data and explore them independently from one another. But we need a kind of staple, a kind of fastener, to keep together all sheets which belong to a same affair. Not only a mark, which demands browsing all records to build the entry, but a mechanic process which does not need any calculation.

Database technology provides a tool of this kind: related tables[84]. Let us declare to the computer that all characters fields of our flat database belong to a specific "table", in which each of them is an independent record. Let us also declare that we create a second table in which each record mentions a specific baptism; and that every record of the second table is "linked" to a set of records of the first one.

Fig. VI. Tables, third step

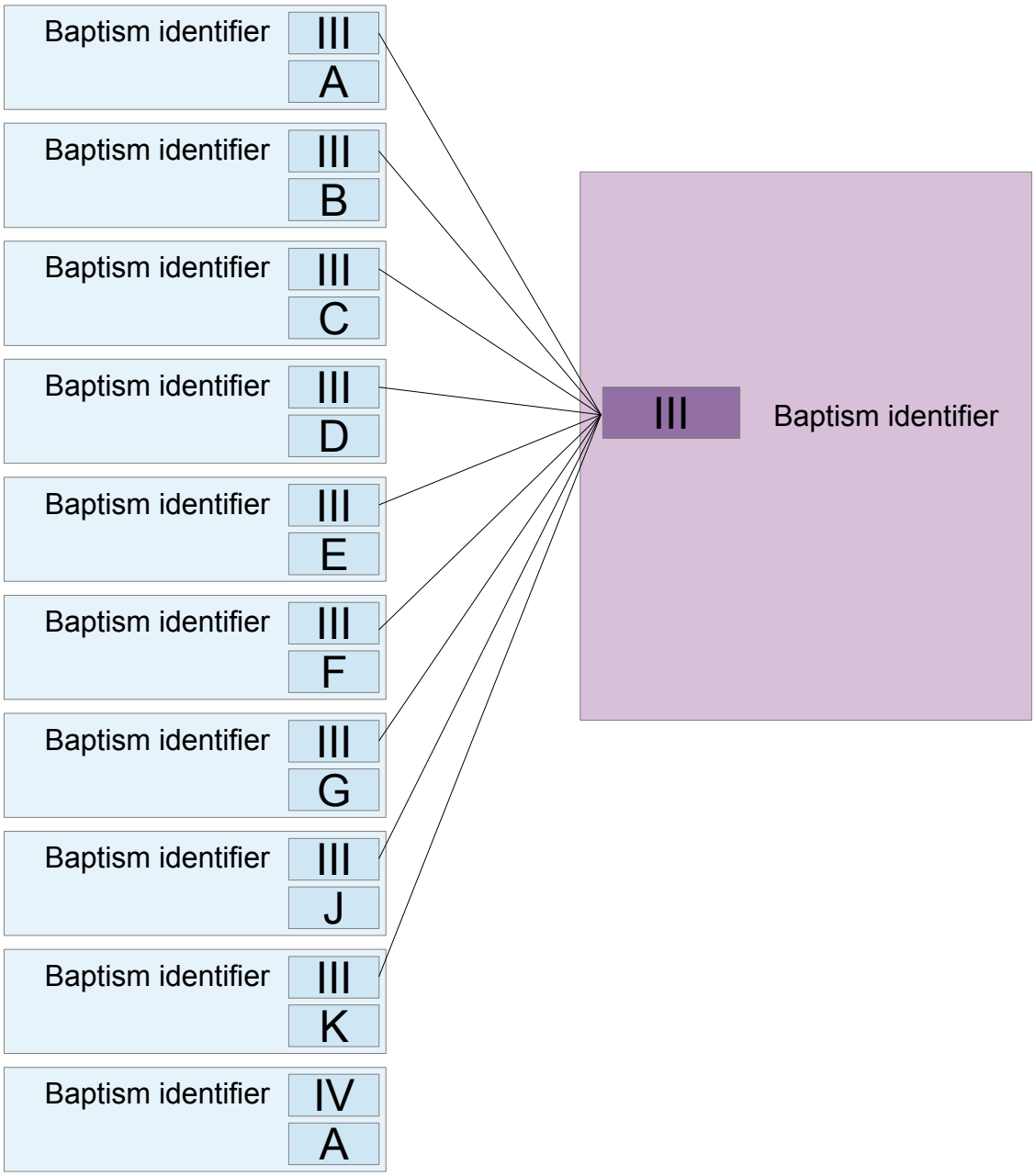


How do we create this link? It would be impractical to draw it manually to each record. In fact, the link sets itself automatically every time that a piece of data fulfills a condition which we declared once for all when planning the database. Let us have a more detailed look at the third entry (the most complex one), once processed in such a way (Fig. VI).

All records of the first table are now composed of two fields. One of them is the value of the data referred to the baptism, exactly as it was in the three previous steps. The other field is the identifier of the baptism. It can be anything you like, but it mus be formally identical on both sides. In the present case we decided it would be "III". Each record of the second table is composed of one field only, which also contains an identifier. We told the computer, when programming the database, that this field was a linking field which matched the "Baptism identifier field" of the first table. Consequently, every record of the second table is linked to all records of the first table if the value of the two linking fields is the same. All records of the first table which hold "III" in the baptism identifier field are linked to all records of the second table which hold the same value in their own database identifier field. As the second table is an image of the series of baptism, we create one and only one record in it for every entry of the book of baptisms. In such a way, all entries of the first table which describe a same baptism are linked to a same unique record of the second table. We can display them from this second table, which makes the paper of a fastener, keeping together loose sheets (i. e. records of the first table), each one of which mentions a specific character - or, in a more abstract terminology, a descriptive dimension - of a specific baptism (fig. VII).

84 At this point, we leave aside the spreadsheet and move to a real database package. Spreadsheets are able to emulate relational architecture, but in a rather clumsy fashion.

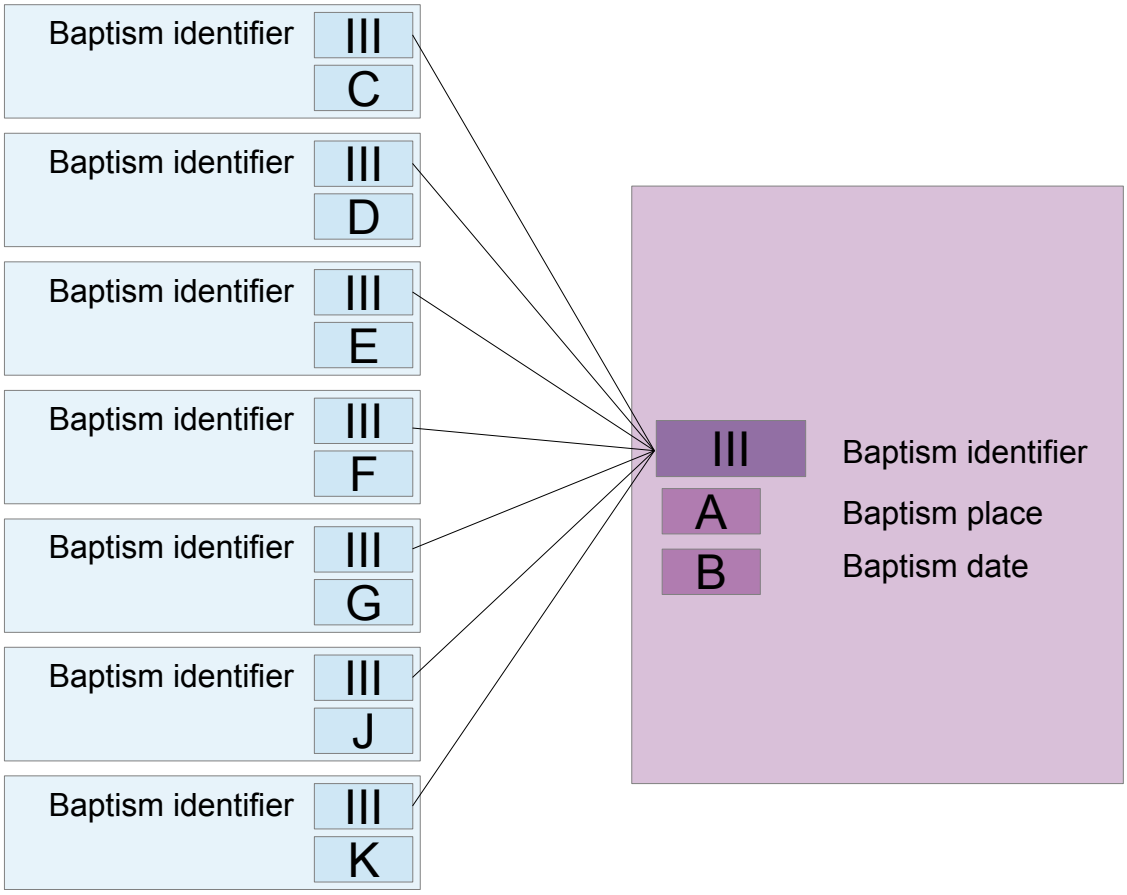
Fig. VII. Tables, fourth step



This model works. It nevertheless can be improved and made more simple. [A] and [B] describe the date and the place of the baptism. Every baptism entry necessarily mentions them. These data are always given, and always given once in every entry of the book. In other words, they have the same dimensions as the baptism itself. Being as permanent and as stable in their structure as the baptism itself, they can easily be transfered to the second table which denotes celebrations[85] (fig. VIII).

85 On the concept of dimension, see our conclusion to the present appendix.

Fig. VIII. Tables, fourth step



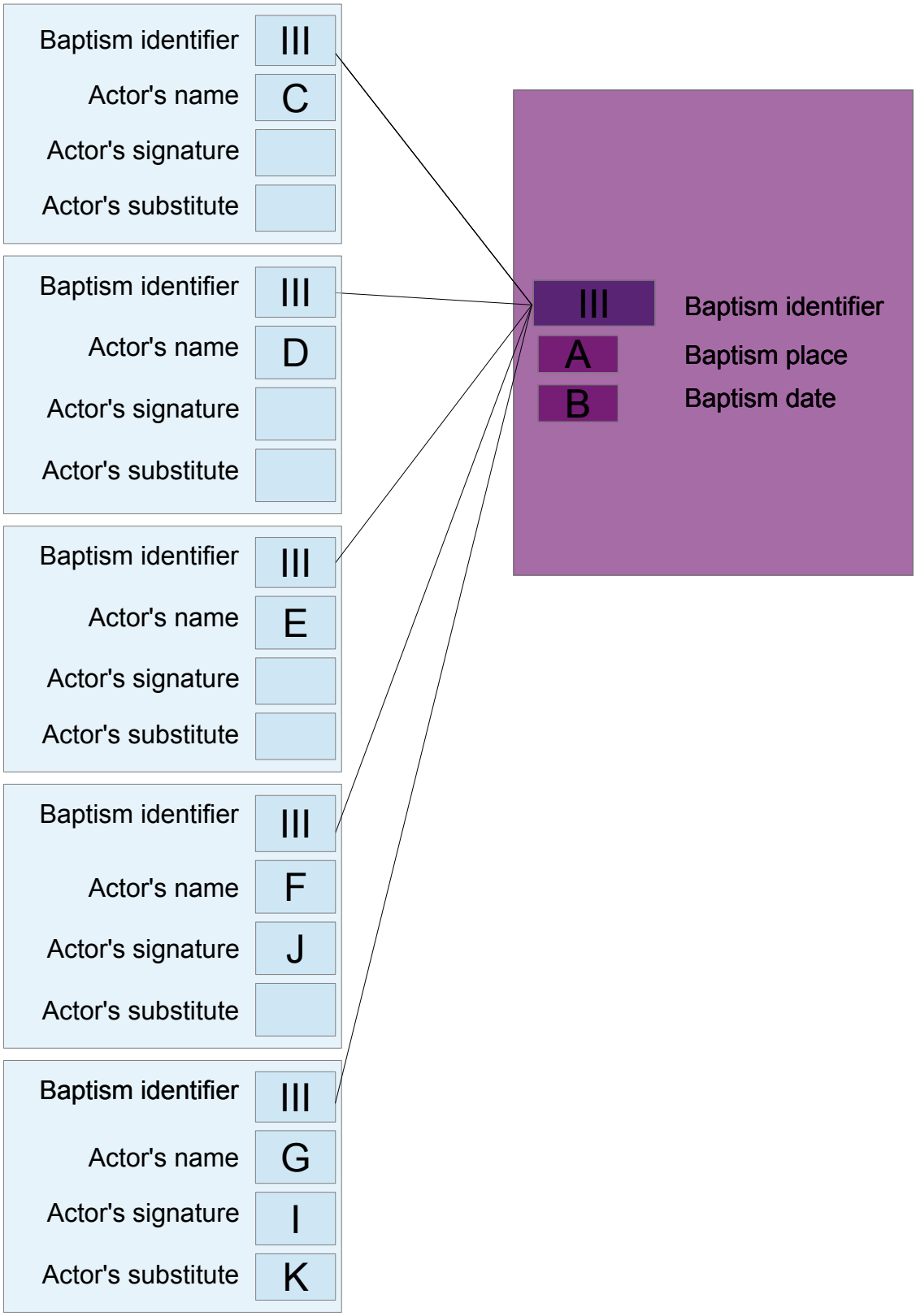
We had in the previous stage two computing blocks: table I (blue) and table II (purple). We still have the same two computing blocks, but they now shape two homogeneous information blocks: on one side (purple) information on the baptism *per se*; on the the other side (blue) information on the actors who take part in the baptism, a non-essential, circumstantial and variable data when seen from the point of view of the baptism itself. We are now in condition to name the first table (blue) the Actors table, and the second one (purple) the Baptisms table.

We can go a step further in our way towards simplification. [J] and [K] represent signing abilities of actors [F] and [G]. Every actor may potentially be described according to the quality of its signature. We may consider signatures as a descriptive dimension attached to any actor, and make it a field of every record which describes an actor. If we get information on this point, we store it to this field; if not, the field remains empty. [H] and [I] are representatives of [F] and [G]. Every actor may have a representative. Even the celebrant: as a general rule he must be the vicar in charge of the parish, but a vicar may name a delegate for any specific celebration[86]. The same as we created a field for the signature in every record in which an actor is mentioned, we are in condition to create another field for possible representatives[87]. We now have one record only for each actor. The original field around which each record has been built up holds the actor's name in every surviving record. We rename it to account for its new quality (fig. IX).

86 The same is true of marriage, the celebration of which the vicar may delegate. Even spouses may delegate their role to proxies without impairing the validity of the sacrament.

87 With the strict condition that each actor has no more than one representative, so that the dimension of the actor (1) would be the same the dimension of the representative (1). If an actor could be represented by various proxies, it would have been necessary to create a specific table for them and link the records of the same to the actors.

Fig IX. Tables, fifth step

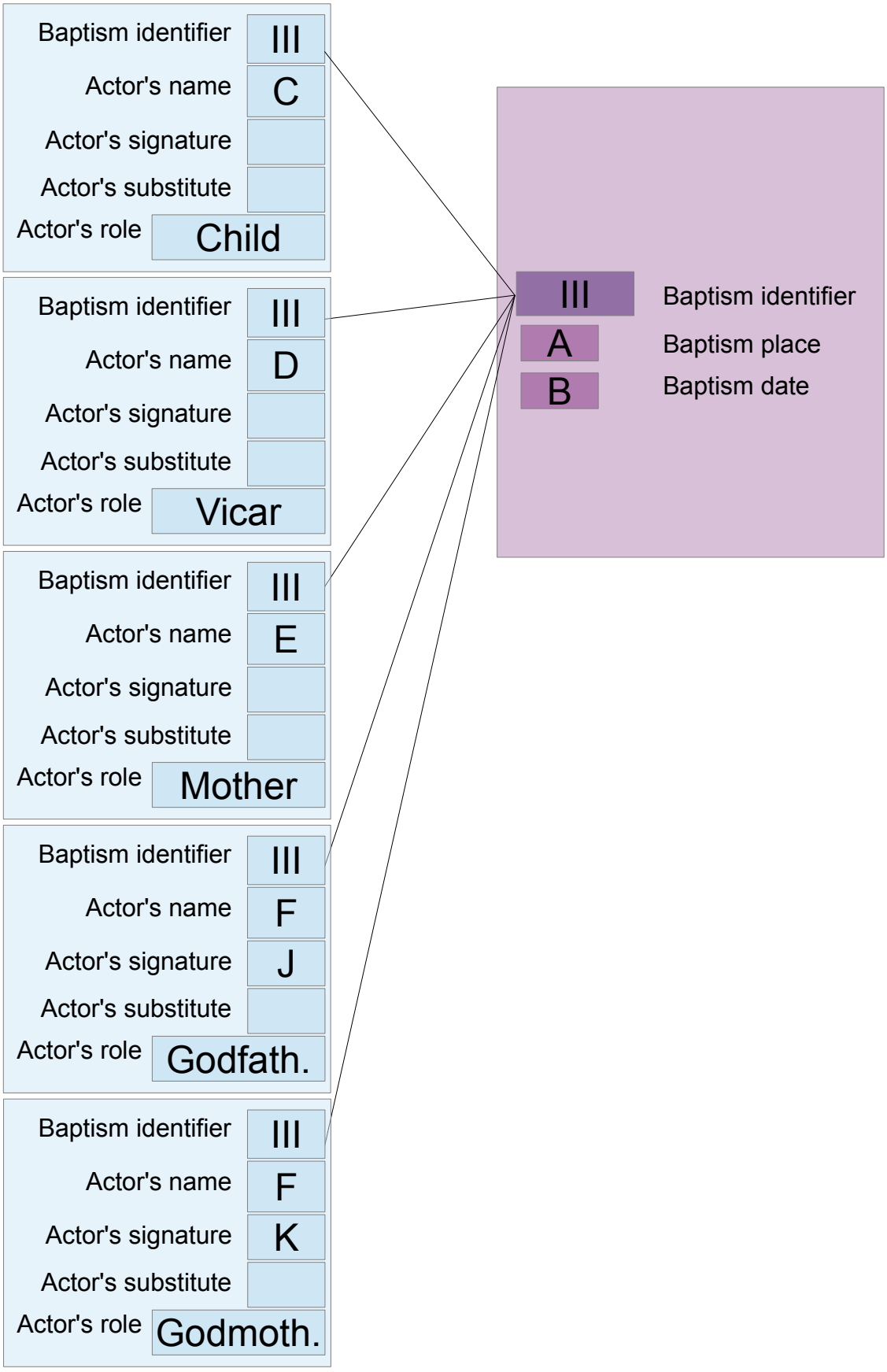


The same as we created a field for the signature in every record in which an actor is mentioned, we are now in condition to create another field where to store the name of possible substitutes and representatives[88]. We now have a record for each actor. The original field around which each record has been built up now holds the actor's name in every surviving record. We rename it to account for its new quality.

There remains a last problem. In the original flat table, the position of the field within the line indicated the paper of the mentioned actor. In its first version, the third position was the child, the fourth the vicar, the fifth the mother, the sixth the godfather and the sixth the godmother (fig. I). After representatives and signatures had been introduced, the godmother passed to the eighth position (fig. III), but positions still indicated roles. The relational model abolishes the concept of position. Nothing remains to define the role of the actor. We must perforce create an extra field, in every record of the Actors table, to make roles explicit (fig. X).

88 With the strict condition that each actor has no more than one representative, so that the dimension of the actor (1) would be the same the dimension of the representative (1). If an actor could be represented by various proxies, it would have been necessary to create a specific table for them and link the records of the same to the actors.

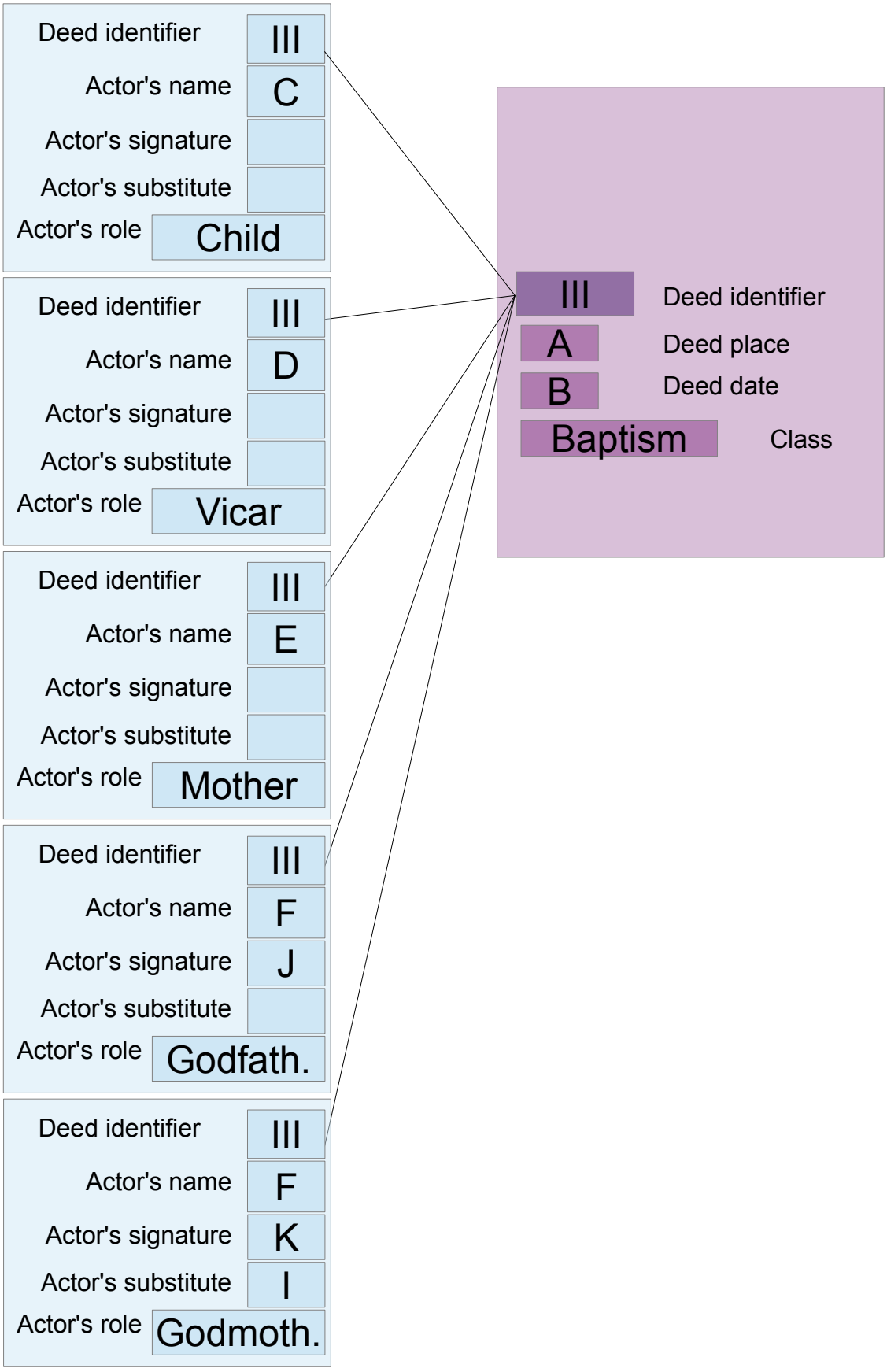
Fig. X. Tables, sixth step



III. Final generalization

This final result (Fig. IX) can easily be made general for any kind of legal deed registered by any institution. Just change the name of the field, and in lieu of Baptism, put Deed. You will be able to process any will, sale, power, testimony, marriage compact, death certificate, and the same, with this same tables and fields structure. Of course, you must change the name of the "Baptism" table also, to "Deed". By so doing, you loose the information the word "Baptism" conveys. You must consequently add a field to the "Deed" table to describe the kind of deed you are processing (fig. XI).

Fig. X. A global model for the process of legal deeds



Conclusive remarks

We began writing this appendix as a purely pedagogical contribution. On second thought, we see it as a perfect example of Fichoz methodology. You start from a factual problem. You refuse the easy option of leaving information out. You build an abstract model of the data you are processing, in accordance to the fundamental principles of data processing and computing. You implement a first solution, which you perfect step by step. The criteria of improvement are:

. Simplification: every step must reduce the number of components;

. Cohesiveness: every step must make the system more consistent;

. Globalization: every step must enlarge the system's spectrum of efficiency.

The system in its final state, must be able to process any piece of information belonging, not only to the same specific class, but to the same generic class as the original data from which you started. And by the way resolve a far broader range of problems than the ones you originally planned to settle.

The solution you reached is independent of the package you are using: it is not technical stuff; it is fundamentally data analysis, in the light of computer technology. In the era of flat databases (before 1990, more or less), the developments we describe in this paper could have been imagined, in spite of the fact that implementation would have been impossible. On the other hand, I believe that their general and abstract character would allow their transposition, with purely technical changes, to three-dimensional database technology which, possibly, will one day replace relational databases.

*

* *

The concept of mono-dimensionality of data underlies the process which we have been describing in this appendix. Positive results obtained in the end confirm its validity in the present case. We must now describe it with more detail and in a more abstract way to make it transferable to other contexts.

We analyze mono-dimensionality in the following way:

. The information sources provide can be described as a set of components related to one another within a hierarchical structure. The most general pattern is that of "subject / predicate": a subject (an actor, a thing, a place, etc.) is described by a set of predicates. For instance, actions, in our view, are predicates of actors. Actors are subjects.

. Within the limits of a database item (vulgo: record), predicates, as well as subjects, must be mono-dimensional: each predicate must affix to the each subject one, and only one quality. This is the crux of the matter.

If you study a group a middle-class households all of which own one or two cars, never more, you may create two classes for cars and store them into two fields belonging to the same record: "First car" and "Second car"; each of them as a separate predicate. Your classification will be consistent within the universe under consideration: "First" and "Second" will never have to store more than one item and no item will be left aside. The subject (the household) is mono-dimensional; each predicate is also mono-dimensional. Everything runs smoothly.

If one household, even one household only, owns a third car, you must create a "Third" class not to leave information out and to keep the predicate structure consistent with the data, that is to maintain each predicate and each subject mono-dimensional within the record. Creating

new mono-dimensional predicates as fields, as we did in the current example, is possible only if you know the maximum number of possible choices, not within the data universe you are studying, but, in rigor, within any data universe you plan to study. A conclusion unobtainable from factual observation, but only from theoretical considerations.

Another solution consists in creating as many database items (i. e. records) in the database as predicates to affix to the actor (First car: one record; Second car, another record; Nth car, Nth record). This is the solution we chose in Fichoz for actions affixed to actors. A very efficient and flexible solution in the present medium state of database techniques, as far as you keep under control the number of predicates processed in such a way.

. If you cannot preserve mono-dimensional coherency between subjects and predicate, either absolutely or for practical reasons (unwieldy layouts, etc.) you must make the set of predicates which breaks coherency one class, an erect this class to the rank of predicate. Being a unique entity, this class, seen from the point of view of the object, re-establishes consistency.

Forget "First", "Second" and "Third". Create a unique class "Car". The household (subject) owns cars (predicate).

. BUT this class is composed of various entities (First, Second, Third... Nth). To be processed, the class itself must be considered, from an internal point of view, as a subject, from which various predicates depend. To maintain mono-dimensional coherency, we must describe it either by means of various fields,

Car is described by "First", "Second", "Third"..."N"

or by means of various records, by far the most probable solution: we precisely had to create a class because of the failure of the fields solution at a higher level.

. The question is to manage such an ambiguous entity as the class, at a same time a subject, when considered from a certain point of view, and a predicate, when considered from another. Relational tables are the current technical answer.

Technical considerations do not force any unambiguous solution. They create constraints. But these constraints leave a large space open to users' choices and preferences. Other factors than computing technology must be taken into account, first of all feasibility and ergonomics, and consistency with the data. The design of classes, the central paper of which in the internal computing design we stressed, is of special relevancy: it must answer at the same time computing requisites and cognitive ones, it must match at the same time the needs of the computer and the nature of described phenomena. For this reason, data-building database design, let us insist once more on this point, is and must be of the researcher's responsibility.

Table des matières