

オンライン目録データの品質管理

—筑波大学附属図書館の場合—

栗山正光

1. はじめに

オンライン・データベースの急速な普及とともに、データの質に対する要求も高まっている。MEDLINE 中の誤りについていくつか報告がある¹⁾し、OCLC についても、誤りの種類の調査²⁾や誤りが発見された場合の報告についての研究³⁾がある。かつてはオンライン検索自体が目新しく、競合するデータベースもなかったために、データの質が問題にされることはあまりなかったが、現在ではオンライン・データベースは珍しいものではなくっており、利用者がデータの品質向上を求めて消費者運動を起こす時期に来ているとも言われる。⁴⁾

ここでは筑波大学附属図書館の洋書目録業務におけるデータの品質管理について御紹介したい。もとよりオンライン目録業務自体が試行錯誤の連続であり、到底まとまった結論を出すところまでは行かないが、何らかの形で御参考になれば幸いである。なお、以下の文中における意見、観点はすべて筆者個人のものであり、必ずしも筑波大学附属図書館のシステム構想に沿ったものではないことをお断わりしておく。

2. 目録データベースの構造とデータの作成方法

筑波大学附属図書館では Model 204 というデータベース管理システムを使って、TULIPS (Tsukuba University Library Information Processing System) と呼ばれる総合的なデータベース・システムの開発を進めてきた。⁵⁾ 目録業務の機械化もその一環である。TULIPS では、目録データは二つのレコードの組合せで表わされる。すなわち、書誌的な情報を持つ書誌レコードと、受入れられた図書一冊毎のデータを管理する受入

レコードとで、この二つは受入レコードが書誌レコードという外部キーを持つことによりリンクされ

表 1 書誌レコードの各項目

◎ 図書書誌コード	BCB
書名・著者表示*1	TIT
版表示	EDI
出版地	PLA
出版者	PUB
出版年	DPU
ページ、冊数	PAG
図版	ILL
大きさ	SIZ
付随資料	ACM
叢書注記*2	SER
一般注記	GEN
内容	CON
*D 目録データベースコード	CDC
* 書名トレーシング	TRT
*D 書名アクセスポイント*3	TTI
巻次トレーシング	TVO
D 著者トレーシング	TRA
*D 著者アクセスポイント*4	TAT
D 叢者名トレーシング	TRS
*D 叢者名アクセスポイント*5	TST
D 叢者巻次トレーシング	TSV
D 副叢書名トレーシング	TPB
D 副叢書巻次トレーシング	TBV
出版年種別コード	TPD
D 出版年トレーシング	TDP
D 言語コード	LAN
*D 分類番号	NDC

*1 本書名、その他の書名、並列書名、著者表示、巻次を I S B D の形式で編集している。

*2 叢書名、叢書巻次、副叢書名、副叢書巻次を I S B D の形式で編集している。

*3 書名トレーシングを単語の単位で切り出し、各各を個別のフィールド値にしている。

*4 著者トレーシングの姓名を別々に切り出し、各各を個別のフィールド値にしている。

団体著者の場合は単語で切り出している。

*5 叢書名トレーシング、副叢書名トレーシングを単語の単位で切り出し、各各を個別のフィールド値としている。

表 2 受入レコードのおもな項目
(目録関係)

◎ 図書ID番号	N(11)	BID
* 図書書誌コード	N(8)	BCB
配架番号	X(V12)	CAL
* 形態コード	X(V2)	FOC
* 配架場所コード	N(V2)	LOC
* 登録番号	X(V9)	BAN
東教大登録番号区分コード	X(1)	TID
* 学系コード	N(2)	INC
図書整理年月日	N(8)	EDP
* 図書発注コード	N(9)	BOC

◎：プライマリー・キー

*：キー項目

D：繰返し項目

N：数字

X：英数

V：可変長

()内の数は長さ

ている。書誌レコードおよび受入レコードの項目は、表1、表2のとおりである。

目録データは、大きく分けて、パンチ外注方式とオンライン方式の二つの方法で作成される。

パンチ外注方式は、原稿をもとに業者がパンチし、磁気テープの形で納入するものである。磁気テープのデータは、担当係である情報検索係によって Model 204 のファイルに取り込まれる。洋書係の仕事であるデータ原稿の作成方法については、すでに報告がある⁶⁾ので、そちらを御参照いただきたい。

オンライン方式は、係員が端末に向かって、直接 Model 204 のファイルにアクセスし、レコー

ドを作成するものである。複本目録処理、オリジナル目録処理、MARC 目録処理の3方式がある。これらの処理の実際についても、やはり文献がある^{7)~8)}ので、ここでは触れない。オンライン目録処理では、新しく作られた書誌レコードはすべて一時ファイルにたくわえられる。書誌の本ファイルは、オンライン目録処理の際には、参照されるだけで更新されない。これは、本ファイルは貸出業務や蔵書検索に供しているため、更新の際必要なレコードの排他制御に失敗することが多いためである。また、システム・ダウン時にファイルが壊れるのを防ぐ意味もある。一時ファイルに蓄積されたレコードは、週一回、情報検索係によって本ファイルに移し変えられる。

MARC 目録処理の開始以来、オンラインによる処理数が急速に増えている。将来はすべての処理がオンラインでなされる予定である。

3. オンライン修正処理

書誌レコードに誤りが発見された時は、洋書係がオンラインで修正する。修正専用のプログラムが用意されているが、複本目録処理のプログラムでも修正ができるようになっている。これは、既存のデータを利用しようとして誤りに気がつくことが多いため、その場で直すことができるようにしたものである。表3に示したようなコマンドが使用される。

オンラインで修正すると言っても、本ファイル

表 3 修 正 コ マ ン ド

修正種別	コマンド	備 考
項目値のおきかえ	R□項目名□'文字列'	文字列に引用符(')を含む時は(')とする (以下同様)
項目の追加	I□項目名□'文字列'	
項目の削除	D□項目名	
文字列の修正	M□項目名□'文字列1□'文字列2'	項目値中の文字列1を文字列2におきかえる。 文字列2を省略した場合は文字列1が削除される。
項目の表示	V□項目名	項目名を省略した時は全項目を表示する。
項目値のコピー	RF□項目名1□項目名2	項目名1の値を項目名2にも設定する。 項目名1の値はそのままである。
項目値の移動	FC□項目名1□項目名2	項目名1の値を項目名2に設定する。 項目名1の値は削除される。
終了	Eあるいは/	

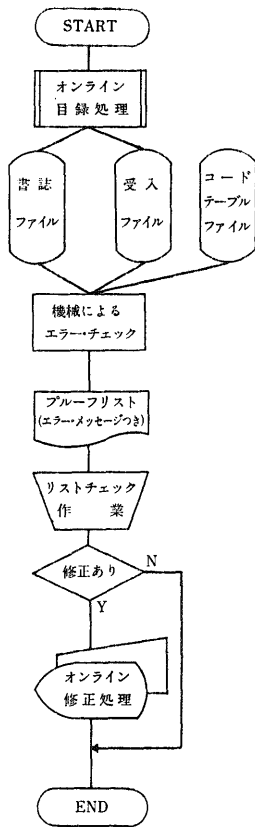


図 1 洋書目録データ校正作業流れ図

を直接修正するわけではない。目録処理で新しいレコードが作成されるのと同じ一時ファイルに、修正しようとするレコードがコピーされ、修正はこのコピー・レコードに対してなされるのである。そして、週一回の本ファイル更新の際に、新しい書誌レコードの追加と共に、修正レコードの置き換えが行なわれる。つまり修正時から本ファイル更新までの間は、本ファイルには修正内容が反映されないこととなる。目録業務では一時ファイル中のレコードも目録処理や修正作業に使うので、常に最新のデータを参照できるようになっている。一時ファイルにおける修正レコードと新しく作成されたレコードの区別は、書誌コードによってつけられる。修正レコードは本ファイルのレコードと同一の書誌コードを持っているが、新規作成レコードには9で始まる仮の書誌コードが与えられている。この仮の書誌コードは、本ファイルへの移行の際、恒久的な書誌コードに置き換えられる。オンライン修正処理プログラムでは、誤修正や

修正もれを防ぐために、修正終了時に機械がチェックを行なう。チェック内容は、書名トレーシング、シリーズ名トレーシング、出版年トレーシング、巻号トレーシングといった項目の値が、書名・著者名・巻号表示、シリーズ表示、出版年、注記といった項目中に含まれているかどうかということや、ISBNのチェック・ディジット、LCナンバーの桁数などである。これらのチェックにひっかかった場合には、項目名が示され、再び修正ルーチンに入るかどうかを聞いてくる。機械的なチェックにかかったからと言って、必ずしもデータの誤りとは限らないので、ここで人間が終了のコマンドを入れれば、そのまま処理は終わる。また、この時、各トレーシングからのアクセスポイントの再作成を自動的に行ない、マニュアルによる修正の手間とミスをなくすようにしている。

4. エラー・データの検出 (自動チェックについて)

オンライン目録処理で作成された書誌データのチェックは、基本的には、リストを打ち出して担当者が目で見て誤りがないか確認するという作業になる。学術雑誌総合目録データベースの校正システムにおいても、さまざまなリストを用意して、人間の目によるチェックを行なうという方法をとっているようである。⁹⁾ 筑波大学附属図書館の洋書係では、毎朝、前日処理したレコードのリストを打ち出している。しかし、リストを全面的に点検するというのは非常に労力がかかり、しかも、必ずエラーが検出されるという保証はない。そこで、機械でできる範囲のチェックは機械でやろうということで、リスト打出しプログラムに機械的なチェックを組み込んで、これにひっかかったレコードにはエラー・メッセージがつくようにした。また、目録データとしては、一冊の本を表わす受入レコードとそれにリンクされた書誌レコードが単位となるわけだが、受入レコードはまた、発注コードによって発注レコードともリンクされている。この発注レコードの書名、著者名を書誌データと併記し、受入レコードと書誌レコードの結びつきが正しいものであるかどうか確認できるようにしてある。

機械的なチェックは、各項目値についてと、関連する項目値間の整合性について行なっている。

おもなチェック内容は次のようなものである。

まず、必須項目なのに落ちていないものがないか、繰返し項目ではないのに二重に値が入力されているものがないかを見る。さらに、受入レコードの各項目や言語コードなどはコードで入力するので、入力された値がコード表に存在するかどうかをチェックする。コード表はコードテーブル・ファイルに格納されており、情報検索係によって管理されている。NDCや出版年トレーシングは、数字何ケタというふうに決まっているので、これも値のチェックは容易である。NDCについては、使用されていないコードがある(180.2 など)ので、これについてもチェックしようとしたが、今のところ一部しかできていない。

文章型のデータを入力する項目にも、よく使われる語が決まっているものがある。版表示、さし絵表示などがそうで、これらについては、プログラム中の単語表にない語をエラーとして表示するようにした。ページ、大きさといった項目は、さらに語順もチェックしている。ページはやや例外が多いが、数字の次に p. とか cm. とかが来るといったふうに、決まったパターンであることが多いからである。出版地は頻出する地名を200件ほどプログラム中に入れてある。巻号トレーシング、シリーズ番号トレーシングなど数字であることが多い項目は、アルファベットの長い文字列が入っていた場合にエラー・メッセージが出される。付随資料はあることがまれなので、この項目に値が入っている時はすべてメッセージを出すようにした。ISBNはチェック・ディジットを確認しているが、本の記載自体が間違っていることもよくある。

項目値間の整合性のチェックとしては、オンライン修正プログラム中でも一部行なっていることだが、書名トレーシング、著者名トレーシング、巻号トレーシング、シリーズ名トレーシングといった項目の値が、書名・著者名・巻号表示、シリーズ表示、注記といった項目の中に存在するかどうかを見る。著者名トレーシングは姓を先にした形なので、‘,’で前後を逆にしている。トレーシングの文字列が表示部の文字列に含まれていない場合、どちらかのスペルミスである可能性が強いわけである。このチェックで、LC MARC のスペルミスが発見されたこともあった。ただし、

これで発見されたエラーも、現物で確認できない時は、修正できない場合がある。どちらの項目がスペルミスかわからないからである。

出版年、出版年種別コード、出版年トレーシングの3項目は密接に関連しており、出版年の記述がこうである時はあとの項目はこの値をとるというパターンが、ほぼ完全にフォローできる。また注記中に‘Reprint of’ という文字列のある時に、出版年種別コードは‘r’ となっているかといったチェックもしている。

このプログラムでは自動修正を行なうわけではないので、例外的なものが、正しいデータであるにもかかわらずひっかかるのには目をつぶって、かなり強引に各項目値の取り得る範囲を設定している。たとえば出版地などは、プログラム中に登録してある地名以外は全部エラーとして出るが、人間の判断で正しいものは見過ごせばいい。出版者などが誤って入っていたり、New york といったスペルミスがあったりするのが見つかっており、結構役に立っている。とは言え、例外が多い場合にはエラー・メッセージが頻出することとなり、本当にエラー・データであるために出ているエラー・メッセージを人間の方が見過ごすということになりかねない。機械的チェックで多くのエラーを見つけようとするほど、例外としてひっかかる正しいデータも多くなるようで、このジレンマの解決が今後の課題である。また、エラー・メッセージの出ているものだけ修正して事足りりという気になってしまうのも問題である。書名のスペルミスなど、今のところ機械的にチェックできないエラーで、注意深くリストを見ていけば発見できるものが確かにあるからである。将来は、辞書を内蔵した第五世代コンピュータがスペルミスまで検出してくれるようになるかもしれないが、この点については、二次資料がオリジナルのスペルミスまで検出しようとするのは筋違いではないかという皮肉な見方もある。¹⁰⁾

書誌データの管理と言えば、典拠ファイルが欠かせないようであるが、今のところ当館では典拠ファイルは使用していない。TULIPS の構想には入っており、将来の課題である。

コンピュータによるエラー・データ検出、さらには自動修正の技法は、これから十分検討されな

ければならない問題である。「人工頭脳論的アプローチ」¹¹⁾による研究の成果が待たれるところである。

5. 書誌統合¹²⁾

同一の書誌を表わす書誌レコードは一つだけというのが原則である。書誌データの重複は、ムダにデータベースのサイズを大きくするだけでなく、検索の効率をも著しく落とす。¹³⁾ 当館のシステムでは、一時ファイルに蓄積された書誌レコード(すなわち目録処理レコードと修正レコード)を本ファイルに移す際、あるいはパンチ外注データを本ファイルに取り込む際、既に同一の書誌レコードがあれば本ファイルに取り込まないようにチェックしている。これには書誌マスターファイルというものを使っている。書誌マスターファイルには、書誌レコードの全項目からブランクを除き、数個のフィールドに圧縮した形でレコードが入っている。一時ファイルのレコードはまずこの書誌マスターのレコードと比較され、重複レコードでないかどうか確認された上で、本ファイルに移されるのである。ただし、これは全項目が完全に一致しないと同一書誌とは認められないので、これだけで重複書誌レコードが生まれるのを防ぐわけにはいかない。重複チェックを厳しくして、新しい重複レコードを作らないようにすることが重要である。重複チェックからもれて、別々に目録処理がなされた場合、でき上った書誌データが完全に一致することは稀だからである。

実際、当館の書誌ファイルには多数の重複データが存在する。おもにオンライン複本目録処理開始以前にパンチ外注方式で作成されたデータである。この方式には以下のような欠点があるため、重複データが発生しやすい。①目録処理と目録データのデータベースへの入力(あるいはカード目録へのカード繰込み)の時期がずれるために、整理待ちあるいは発注中の本の中に複本があった場合にはチェックからもれる。同じ本に、違った購入請求者から、同じ時期に注文が出るのはよくあることである。②複本であることがわかっていてデータ原稿に同じものを使用しても、業者は一冊一冊について全項目同じことをパンチし直すのである。当然パンチミスによって細かい点が違ってしまふことが多い。③洋書係員がデータ原稿とな

る複本のカードの誤りに気付いて、修正して新しい原稿とした時、既にデータベースに入っているレコードの方の修正を忘れやすい。

というわけで、重複書誌データが発見された場合、一つにまとめる作業が必要となる。この処理のことを書誌統合と呼んでいる。

オンライン書誌統合処理の手順は次のとおりである。まず書誌コードあるいは図書 I D 番号で残す方の書誌レコードを呼び出し、次に同様に削除する書誌レコードを呼び出す。呼び出されたレコードが本当に統合したいレコードかどうかの確認がすむと、機械が自動的に、削除される書誌レコードとリンクしている受入レコードの書誌コードを、残される方の書誌コードに変換する。そして削除される書誌レコードの書誌コードが、修正処理のところで触れた履歴ファイルに記録される。実際の書誌レコードの削除は、週に一度の書誌本ファイルの更新の際、この履歴ファイルの記録を参照しながら行なわれる。オンライン統合処理の時に更新されるのは受入ファイルと履歴ファイルだけであり、書誌ファイルは参照のみで更新がなされていないことに御注目いただきたい。

さらに、洋書係では、毎日目録処理をしたものについて、同一書誌がないかどうかをチェックするプログラムを作成した。これは、完全にデータが一致しなくても、同じではないかと思われる書誌レコードのセットを、違っている項目の指摘とともにプリント・アウトしたものである。これを担当者がチェックして、重複データだと判断すれば、前述のオンライン書誌統合処理をする。いわば後からの重複チェックである。これは思いの外効果的で、二人の係員が同じ書誌データを、数日置いて、それぞれオリジナル目録処理で作成していたという悲劇(?)が発見されたりもした。

書誌統合候補レコードの検出のアルゴリズムは次のようにした。まず書名アクセス・ポイント二つと著者名アクセス・ポイント一つが同じものを集める。こうして数を絞ってから、本書名の一致を見る。本書名から符号やブランクを除き、全体を大文字化する。この時の長さの差が0か1のものに限って、頭から一文字ずつ一致を見て行く。X字めで一致しなかったとすると、今度は後ろから逆に一文字ずつ一致を見て行く。するとY字めで一致しなかったとする。この時、 $X+Y-1$ が、

もとなる書名か比べられる書名かの、符号とブランクを取り去った後の長さと一致した時に、二つの書名は同じものと見なす。たとえば、“Library automation?”と“Library, Automaton”という書名を比較するとする。最初のタイトルは LIBRARYAUTOMATION となり、後の方は LIBRARYAUTOMATON と変換される。一字ずつ比較して行くと、 $X=15$ 、 $Y=3$ という値が得られる。 $X+Y-1=17$ となり、これは先の方の長さに等しい。ゆえにこの二つの書名は同じと判断される。後の方のタイトルが、たとえば LIBRARYAUTOMATION となっていたとしても、 $X=4$ 、 $Y=14$ で、やはり $X+Y-1=17$ となるので、同じタイトルとされる。こうして同じ書名と判断されたものの中から、もし版表示や巻号があれば、その中の数字の一致しないものは別書誌として除かれる。残りが書誌統合候補としてリスト・アップされるわけである。これとは別に、LC ナンバーや ISBN の一致するものがあれば、巻号が違わない限り、すべて書誌統合候補となる。なお、いわゆる分類の泣き別れを発見するため、版や巻号が違っていても、分類が異なる場合には、そのままリストに出るようにした。

重複書誌データの自動的な検出については、OCLC における Hickey と Rypka の研究が興味深い。これは、書誌データが同じかどうかの判定に、デジジョンテーブル（考慮条件一覧表）を使っている。たとえば書名の最初と出版年と LC ナンバーが正確に同じであれば出版地は一致しなくてもよいといったふうに、14の比較項目を、16のマッチングのパターンにわけて表にしている。¹⁴⁾

イリノイ大学の Williams と MacLaury の研究では、出版年とタイトルから何文字かを取り出して作ったキーで同一書誌を検出しようとする。この場合、タイトルの何文字めと何文字めを取り出せば最も効果的かを、綿密に調査している。このキーによって集められたものから、さらに、書名が同じかどうかを判定するわけだが、この時用いられる Harrison キーというのもおもしろい。これは1か0が71個並んだもので、何番めと何番めが1になるかは、書名からいくつもの文字列（3文字）を作ってそれを数値化して、71で割った余りで決められる。この Harrison キーを比較した時、違う箇所がいくつあるかによって、書名

が近いかがわかるわけである。スペルミスなどによるささいな違いは無視して、同じ書名（らしきもの）を拾い集めるためには、この違う箇所の数（Hamming distance と言う）をいくつかまで許容することにすればいい。詳細は文献を御参照いただきたい。^{15)~16)}

これらの研究では、出版年を、比較的誤りの少ない項目として、一次検索のキーとしているが、われわれの今までの経験から言えば、必ずしも誤りの少ない項目ではない。英米目録規則によれば、出版年はその版の最初の刷りの年であるが、後の刷りの年を出版年としているものが結構多い。また、コピーライトの年は、最新の年を取ることになっているが、見落しか誤解か、古いコピーライトの年を採用していたりする。¹⁷⁾

いずれにせよ、データにエラーがあることを見越して重複書誌データを自動的に検出するためのアルゴリズムについては、まだまだ研究の余地がありそうである。

それはそれとして、われわれが現在出している書誌統合候補リストは、機械的な重複の検出ということだけでなく、書誌データ内容のチェックの一環とも考えられる。というのも、統合候補として出されたレコードの片方が完璧ということはあまりなく、たいていは、ある部分はこちらの方がよく、あとはそちらの方がよいといったことになるからである。統合されたレコードは、統合前のレコードからそれぞれいい所を取り入れた、よりよい目録データとなるわけである。それだけに、人間の側のリスト解説作業もなかなか大変である。また、矛盾したデータが出てきて、現物を見ないと解決がつかないこともままある。著者名のスペルの違い、同じ版なのに出版年が違う、違う版なのに同じ出版年、などといった場合である。幸い当館では集中管理方式を取っていて、大多数の本は中央図書館に配架されているので、時間の許す限り現物を調査して正しいデータにするようにしている。

現在、この書誌統合候補リストを、全書誌レコードを対象に、少しずつ打ち出して、重複書誌データを統合中である。先に引用した Hickey と Rypka は、OCLC の重複レコードの率を7~9%と推定している¹⁸⁾が、わが洋書書誌ファイルにおいても、非常に大雑把な見積りではあるが、同

じくらしい割合の書誌レコードが削除されることになると思われる。

ところで、書誌統合を進める際には、同一書誌の判定基準が大きな問題となる。たとえばペーパーバックとハードカバーの違いである。ISBNは確かに別になっているが、中身は全く同じである。当館では、これは同一書誌としている。また、版が同じでも、刷りの年が重要である場合はこれを出版年中に記述して、他の刷りの年と区別するが、この場合の重要性の基準をどこに置くかもやっかいな問題である。¹⁹⁾ 出版社が引越して、刷りによって出版地のみ違うという場合もある。これは別書誌とせざるを得ない。同じ本をアメリカとイギリスでそれぞれ別の出版社が出すことも多い。

こうして見ると、本の内容には全くあるいはほとんど差がなくても、現行の目録規則やデータの構造では一つの書誌にできないものが多いことに気がつく。版や出版年や出版地、出版者の違いを目録データとして押えておくことが不可欠だとすれば、多巻物や逐次刊行物をセットとして一つの書誌で表わすように、版や出版年の違う本をグループとして一つの書誌で表わすような工夫が、せめてできないかと思われる。データベースの構造としては、親書誌レコードと子書誌レコードを作るといったことが考えられようが、これは目録規則、書誌記述の標準化、典拠コントロールといったさまざまな問題とかかわっているのだから、軽々しく結論を出すわけにも行かない。さしあたりは、一目録係員のほやきである。

6. その他のデータ・チェック

パンチ外注方式の場合、まずデータ原稿を洋書係員が点検する。そして、データが出来上がってModel 204のファイルに移された時点で、チェック・プログラムを実行し、エラー・データと思われるもののみをリストに出し、調査する。この時、データ原稿を見れば解決のつく誤りならいいが、現物を見て確かめなければならない場合は、すでに本は配架されてしまっているため面倒である。この点においても、整理の翌日にチェックを行なうため確認の容易なオンライン処理に比べて非効率的である。

修正データについては、本ファイルのデータと一時ファイルの修正後のコピー・データとを比較して、異なっている項目のみをプリントしたリストを出して点検している。この時も一応チェック・プログラムを通して、エラー・メッセージがつくようにしてある。ただし、修正件数が多い時は十分に人間の目によるチェックがなされていない。

他に、同じ種類のミスがいくつかあった(たとえば *Britanica* というスペルミスがいくつも見つかったというような)場合、同じミスを犯しているレコードを網羅することは容易なので、その場に応じてリストを作成し、修正作業を行なっている。

以上、コンピュータを利用した目録データのチェックと修正について述べてきたが、もとよりこうしたやり方でのデータの質の向上にも限界がある。この点、利用者による誤りの指摘は、データの品質改善にも、これをチェック体制なりプログラムなりに反映させてよりよい品質管理システムを作ることができるという意味においても、大きな力となる。冒頭で触れたデータベースの質の向上を求める消費者運動も、こうした意味あいでも、意義を持つのではないだろうか。

- 1) 奈良岡功 “Dialog/MEDLINE における 110,000 件に及ぶ発行年処理ミスについて” 『医学図書館』 31(1), 1984, pp.34-41. 他。
- 2) Ryans, Cynthia C. “A study of errors found in non-MARC cataloging in a machine-assisted system.” *Journal of library automation* 11(2), 1978. 6, pp.125-132.
- 3) Johnson, Judith J. and Josel Clair S. “Quality control and the OCLC data base: a report on error reporting.” *Library resources & technical services* 25(1), 1981, pp.40-47.
- 4) Norton, Nancy Prothro “‘Dirty data’—a call for quality control.” *Online* 5(1), 1981, pp.40-41.
- 5) 『筑波大学附属図書館情報処理システム, TULIPS, システム仕様 (その5)』桜村 (茨城県), 筑波大学図書館部学術情報課, 1984.
- 6) 青木公男 “筑波大学附属図書館における目録業務 (洋書の入力データ作成を中心として)” 『大学図書館研究』(21), 1982, pp.69-74.
- 7) 青木公男, 室橋 真 “筑波大学附属図書館における

- 目録業務（洋書のオンライン入力を中心として）『医学図書館』31(2), 1984, pp.145-152。
- 8) 『筑波大学附属図書館情報処理システム, TULIPS, システム仕様（その5）』
- 9) 根岸正光, 柴田正美 “学術雑誌総合目録データベース・オンライン/バッチ校正システムの開発”『大学図書館のシステム化』（根岸正光, 井上 如, 藤原鎮男共編）東京, 紀伊国屋書店, 1981, pp.472-502。
- 10) Gee, Ralph “Quality control in online databases.” *Aslib proceedings* 35(6/7), 1983, pp.239-248.
- 11) 根岸, 柴田, Op. cit.
- 12) 松井幸子 “書誌情報の共同利用ファイルの作成——重複文献同定の技法について——”『ドクメンテーション研究』27(4), 1977.4, pp.171-178。
- 13) Wanninger, Patricia Dwyer “Is the OCLC database too large? a study of the effect of duplicate records in the OCLC system.” *Library resources & technical services* 26(4), 1982, pp.353-361.
- 14) Hickey, Thomas B. and Rypka, David J. “Automatic detection of duplicate monographic records.” *Journal of library automation* 12(2), 1979, pp.125-142.
- 15) Williams, Martha E. and MacLaury, Keith D. “Automatic merging of monographic data bases—identification of duplicate records in multiple files: the IUCS scheme.” *Journal of library automation* 12(2), 1979, pp.156-168.
- 16) MacLaury, Keith D. “Automatic merging of monographic data bases—use of fixed-length keys derived from title strings.” *Journal of library automation* 12(2), 1979, pp.143-155.
- 17) Jones, Barbara and Kastner, Arno “Duplicate records in the bibliographic utilities: a historical review of the printing versus edition problem.” *Library resources & technical services* 27(2), 1983, pp.211-220.
- 18) Hickey and Rypka, Op. cit.
- 19) Jones and Kastner, Op. cit.
-
- <60. 4.23 受理 くりやま・まさみつ
筑波大学図書館部管理課洋書係>