

265 Jahre
Universitätsbibliothek Braunschweig
65 Jahre Dietmar Brandes

Herausgegeben von
Beate Nagel

Braunschweig 2013

Veröffentlichungen der Universitätsbibliothek Braunschweig
- Hrsg. von Dietmar Brandes -
Heft 16

© Universitätsbibliothek Braunschweig
ISBN 978-3-927115-72-9

allegro-C: Eine Software im Epochenwandel

BERNHARD EVERSBERG

Will man sich auf dem weiten Feld der Datenverarbeitung die Entwicklungsstadien der letzten drei Jahrzehnte vergegenwärtigen, ist es hilfreich, mindestens vier Aspekte in den Blick zu nehmen:

- Hardware: Industrie, Systeme
- Software: Programmierung, Sprachen, Entwicklungssysteme
- Software: Konkrete Anwendungsfelder, z.B. Bibliothekswesen
- Bestimmte Anwendungssysteme, z.B. *allegro-C*

Am Beispiel des an der UB Braunschweig entstandenen Systems *allegro-C*¹, dessen Entwicklung 1980 begann und bis heute andauert, lassen sich die Trends, die Evo- und Revolutionen der 30 Jahre nachverfolgen, und dies wird hier versucht. Die Stationen und Etappen der *allegro*-Geschichte sind knapp aufgelistet in einer Chronik², die ihrerseits im Kern eine spezialisierte *allegro*-Datenbank ist. Für die Anfangszeit bis Ende 1994 existiert eine ausführliche Darstellung der Entwicklung von Torsten Ahlers³.

Wie konnte es dazu kommen?

Möglich wurde *allegro-C* anfangs durch den Wagemut des früheren Direktors, JOSEF DAUM, der neben dem Sinn für das Bewahren von

¹ Homepage für *allegro-C*: <http://www.allegro-c.de> (dort die gesamte aktuelle Dokumentation etc.)

² *allegro*-Chronik : <http://www.allegro-c.de/chronik/>,
Kurzfassung: <http://www.allegro-c.de/chronik/a15.htm>

³ Ahlers, Torsten: Das Projekt *allegro* : Entwicklung, Anwendungen, Einsatz in Netzen. - Braunschweig: Univ.-Bibliothek der TU, 1994. - 102 S. (Veröffentlichungen der Universitätsbibliothek Braunschweig ; 11). Zugl.: Köln, Fachhochsch. für Bibliotheks- und Dokumentationswesen, Hausarb., 1993
ISBN 3-927115-24-X. Online: <http://www.digibib.tu-bs.de/?docid=00028938>

Kulturgut das beherzte Betreten von Neuland nicht scheute. Nach seinem Eintritt in den Ruhestand war es dann der lange Atem seines Nachfolgers DIETMAR BRANDES: Er vereinbarte stets den Weitblick und den Sinn für Kontinuität, den eine Bibliothek braucht, mit der Tatkraft und der Umsicht des Managers, der seinen Betrieb arbeits- und zukunftsfähig halten will.

Nicht zuletzt ist dem Niedersächsische Ministerium für Wissenschaft und Kultur zu danken, weil es immer wieder mit Sondermitteln half, die Entwicklung in Gang zu halten, nachdem es im Jahre 1991 der Bibliothek (nicht der TU!) die Weiterentwicklung als „zusätzliche staatliche Aufgabe“ übertrug.

Auf befristeten Projektstellen haben dann etliche Programmierer (s. Anhang) mit sehr unterschiedlichen Hintergründen in der Entwicklung mitgeholfen und jeweils mit spezifischen Beiträgen neue Anstöße hineingebracht oder ganze Programme oder Funktionen neu konzipiert.

Entwicklungen profitieren nicht selten auch von Zufällen. Einer von vielen war in dieser Sache, daß die Firma Commodore kurz nach unserer Entscheidung, mit der sie selber natürlich gar nichts zu tun hatte, in Braunschweig eine Produktionsstätte mit einer kleinen Entwicklungsabteilung einrichtete, um hier genau das Rechnermodell für Europa zu produzieren, für das wir uns entschieden hatten. Kontakte waren schnell geknüpft, auch zu anderen Mikrocomputer-Enthusiasten der ersten Stunde in Braunschweig, und dies war nützlich in mancherlei technischen Details, die man allein nicht hätte lösen können.

Schon 1981 setzte aber auch Eigendynamik ein: Anwender außerhalb der UB Braunschweig trieben die Sache voran, weil sie in wachsender Zahl und in allen Sparten des Bibliothekswesens bis heute die Entwicklung begleiteten und immer wieder aus der Praxis heraus neue Anforderungen formulierten.

Stationen der Entwicklung, Megatrends in der Datenverarbeitung

Ab 1980 : Dezentralisierung

Als es anfang mit der Software *allegro-C*, im Jahre 1980, da kannte man nur Großrechner und Minicomputer. Sie alle hatten, im Vergleich zu heutigen Netbooks oder Smartphones, physisch riesige Ausmaße bei sehr bescheidener Leistung. Endnutzer bemerkten kaum einen Unterschied, weil man in jedem Fall ein „Datensichtgerät“ (auch „Terminal“ genannt) vor sich hatte und der Rechner im Maschinensaal eines Rechenzentrums stand, klimatisiert und nur wenigen direkt zugänglich. Die große Zeit der Rechenzentren mit ihrem Quasi-Monopol und „Herrschaftswissen“ ging aber dem Ende entgegen. Die strikte Zentralisierung von Rechenleistung, erzwungen allein schon durch die Kosten der Hardware, wurde durchbrochen, als die ersten Mikrocomputer kamen. Und die ersten, die man im Büro einsetzen konnte, gelangten 1980 auf den Markt, zu bestaunen erstmals auf der Hannover-Messe, und zwar in der Halle 1 im Centrum für Büro- und Informationstechnik (CeBIT, erst ab 1986 wurde daraus eine eigenständige Messe).

Zunächst auch als „Hobbycomputer“ bekannt und von den gestandenen Datenverarbeitern noch als Spielzeuge abgetan, haben in den folgenden Jahren die Mikros, bald schon Personal Computer genannt, eine Epoche der Dezentralisierung herbeigeführt. Aber erst mit dem ernsthaften Markteintritt von IBM (welche flugs den Terminus PC für sich okkupierte) im Jahre 1983 wurde der PC „hoffähig“ und zog in immer mehr Anwendungsfelder ein, ja ermöglichte erstmals einen Computereinsatz auf Gebieten, wo dies vorher viel zu teuer gewesen wäre.

Die ersten Anwendungen, aus denen sich ein Software-Massenmarkt entwickelte, waren Textverarbeitung - vorher auf den Großrechnern wenig komfortabel - und die als Sensation erlebte Tabellenkalkulation, im Bürobereich auch bald die Finanzbuchhaltung für kleinere Unternehmen. In technischen Bereichen kamen z.B. vielerlei Anwendungen der Automatisierung und Prozeßsteuerung auf, die heute längst Alltag sind.

In Bibliotheken stand obenan die Katalogdatenerfassung, die vor 1980 mit äußerst geringem Komfort, hoher Fehleranfälligkeit, geringer Produktivität und ohne Standardisierung mit offline-Geräten geleistet wurde: auf Lochkarten, Fernschreiber-Lochstreifen und schließlich auf Papierformularen, die mit speziellen Schreibmaschinen umständlich zu beschreiben waren: mit solchen Geräten wurden die ersten Daten für den Niedersächsischen Monographien-Nachweis (NMN) erfaßt. An der UB Braunschweig wünschte man sich aber mehr Komfort für effizienteres Arbeiten. Dafür kam die neue Geräteklasse, wie sich bald zeigte, gerade recht.



Abb. 1: Mikrocomputer CBM 8032



Abb. 2: Akustikmodem zur Datenübertragung

Ab Frühjahr 1981, nach wenigen Monaten Vorlaufzeit, wurden schon die ersten „Mikros“ vom Typ CBM 8032 regulär in der Datenerfassung eingesetzt⁴. Damit war die UB Braunschweig wohl die erste, die solche "Spielzeuge" im normalen Geschäftsgang verwendete, lange bevor sie dann die Kartei- und Aktenschränke als allgegenwärtige Büromöbel verdrängten, heute von keinem Schreibtisch mehr wegzudenken sind und die ICEs zu rollenden Arbeits- und Vergnügensstätten gemacht haben .

⁴ EVERSBERG, B. Einsatzmöglichkeiten von Mikrocomputern in der Bibliotheksarbeit. In: Bestände in wissenschaftlichen Bibliotheken, 71. Deutscher Bibliothekartag in Regensburg 1981. - Frankfurt a.M.: KLOSTERMANN, 1982. - ISBN 3-465-01511-8. - S. 221-229. (ZfBB Sonderheft ; 34)

Die Programme wurden damals in Basic und Assemblersprache geschrieben, jeweils in Dialekten, die nur für den CBM galten. Ein plattformunabhängiges Programmieren war nicht möglich, weil die Industrie noch keine normierten Sprachen und Schnittstellen kannte. Die Mikros verstanden zwar alle Basic, aber jeder mit eigenen Besonderheiten, und Assembler war ohnehin nicht normierbar, weil eng verbunden mit den Eigenheiten des jeweiligen Mikroprozessors.

Erfasste Daten wurden gespeichert auf handelsüblichen Tonbandkassetten (!), die Dateien wöchentlich mittels Akustikmodem über Telefon zum Rechenzentrum der TU übertragen und von da weiter nach Göttingen. Dort wurden Korrekturausdrucke erstellt, per Bücherwagen nach Braunschweig geschafft, die Fehler dann online an dem einzigen Terminal korrigiert, das man ergattern konnte. Nach der Korrektur erfolgte in Göttingen der Ausdruck von „geköpften“ Karten, die wieder per Bücherwagen nach Braunschweig gelangten und endlich manuell in den alten Zettelkatalog einsortiert wurden. Das ging so bis Ende 1992, als der im Entstehen begriffene Niedersächsische Verbund das Pica-System einführen konnte. Mit der Eröffnung des Pica-OPAC am 2.1.1993 hatte Braunschweig deshalb schon einen Datenbestand von 13 Jahren vorzuweisen. Und die *allegro*-Software war um einige Funktionen erweitert worden, darunter ein parametrierbares Exportverfahren und eine Volltextsuche, obwohl die Datenmengen begrenzt waren, die man auf Disketten unterbringen konnte und auch Festplatten noch viel weniger Volumen hatten als heute ein Memory-Stick oder ein Mobiltelefon.

Ab 1985 : Industriestandards

Die noch nicht enorm umfangreiche *allegro*-Software wurde neu in der Sprache C geschrieben, als nach dem Markteintritt von IBM klar wurde, wohin sich die Industrie (mit Ausnahme der Firma Apple) bewegen würde: Hin zu offenen Hardwareschnittstellen sowie standardisierten Betriebssystemen und Programmiersprachen. MS-DOS war dann für viele Jahre die Kernsoftware von PCs, daneben UNIX (und später dessen populäre offene Variante Linux) für die etwas gehobene Klasse von mehrplatzfähigen Systemen. Zur Sprache der Wahl wurde C, das in beiden Welten funktionierte, wenn auch mit gewissen Unterschieden. Der Umgang mit C

war damals noch recht schwierig und das Testen zeitraubend (man könnte von einer "Entdeckung der Langsamkeit" reden), wogegen man vorher in der "Interpretersprache" Basic (stets sofort und direkt ausführbar) schon viel schneller und leichter entwickeln konnte. Aber fertige C-Programme waren dafür (und sind heute noch) konkurrenzlos schnell, C ist viel besser standardisiert und etabliert, und es ist für viel größere Systeme geeignet als Basic, welches heute für anspruchsvolle Anwendungen nicht mehr zum Einsatz kommt.

In Zusammenarbeit mit der Herzog August Bibliothek Wolfenbüttel wurden ab 1990 die ersten in C geschriebenen Module von *allegro* auch unter UNIX zum Laufen gebracht und ermöglichten dort das Erfassen erster Daten auf Siemens-Anlagen (mit der UNIX-Variante SINIX).



Abb. 3: Der erste IBM PC

Die PCs, zunächst stets Einzelplatzsysteme, wurden mit dem Entstehen von Fileserver-Betriebssystemen allmählich vernetzbar: das waren PC-Programme, die eine zentrale Datenhaltung und den Zugriff auf eine gemeinsame Platte ermöglichten. Software lief getrennt auf jedem einzelnen der angeschlossenen PCs, auf dem Server lagerten nur die gemeinsam genutzten Daten. Das war neu, denn UNIX-Anlagen waren damals immer noch Minicomputer, auf denen auch alle Programme zentral abliefen, und die Endgeräte waren weiterhin „dumme“ Terminals, nur geeignet zum Anzeigen von Daten und Meldungen und zum Eintippen von Befehlen.

Standardisierung war in der Industrie die Voraussetzung für rapides Wachstum des PC-Marktes, und mit der Vielfalt unterschiedlicher, inkompatibler Betriebssysteme und Sprachen war es deshalb schnell vorbei.

Für *allegro* konnte um 1988 eine neue Art des Zugriffs über frei konfigurierbare und ständig aktuelle alphabetische Register realisiert werden, als Magnetplatten mit mehr als 20 MB in den Handel kamen und rasch billiger wurden. Hinzu kamen die neuen Techniken der Vernetzung, die dem PC-Markt nochmals starken Auftrieb gaben, weil damit auch Mehrplatzfähigkeit auf preisgünstigen lokalen Netzen in den Bereich des Machbaren kam: eigene Rechenleistung am Arbeitsplatz wurde kombiniert mit dem Zugriff auf gemeinsam genutzte Daten.^{5,6} Damit begann eine neue Epoche:

Ab 1990 : Re-Zentralisierung

Schon vor der Pica-Einführung wurde im Jahre 1990 in Braunschweig ein *allegro*-OPAC, noch ohne Ausleihe, eröffnet, und zwar auf einem selbstgebauten lokalen Novell-Netz. Denn *allegro* konnte zu der Zeit bereits in solchen Netzen eine auf dem Server liegende Datenbasis auf allen angeschlossenen Plätzen zugleich anbieten.

⁵ EVERSBERG, B.: Bibliothek - Computer - Information - Hintergründe und Folgerungen einer Innovation. In: Kleincomputer in der Bibliotheksarbeit, Referate einer Fortbildung 1986. - Berlin: DBI, 1988. - ISBN 3-87068-873-4. - S. 17-36. (dbi-materialien ; 73)

⁶ EVERSBERG, B.: Allegro : A new database concept for Micros (MS-DOS and UNIX). - In: Developments in microcomputing, 12th International Essen Symposium 1989. - Essen: Ub, 1990. - ISBN 3-922602-13-4. - S. 128-144. (Veröffentlichungen der Universitätsbibliothek Essen ; 12)



Abb. 4: allegro-OPAC in der UB (Startbild der ersten CD-ROM zu V15)

Neben den UB-Katalog trat bald ein Institutskatalog, in dem erstmals Monographienbestände der TU-Institute zusammengefaßt wurden: In den Instituten wurden jeweils auf Einzelplatz-PCs vom Institutspersonal die bibliographischen Daten erfaßt und der UB übergeben. Weit über 300.000 Bände wurden bis heute auf diese Weise erfaßt und später in den gemeinsamen Pica-Katalog überführt, der nun als OPAC für den Gesamtbereich der TU dient. Ohne *allegro* wären wohl diese Bestände noch immer nicht so umfassend auffindbar, wie sie es heute sind.

Die Mehrplatzfähigkeit war entwickelt worden zusammen mit DR. PETER PFEIFFER, damals an der Herzog August Bibliothek Wolfenbüttel tätig und vor 1989 in Berlin an der Staatsbibliothek unter den Linden. Diese Kooperation begann - eine seltene Sache - schon vor der "Wende". PFEIFFER war UNIX-Spezialist und ihm ist der Einstieg in die UNIX-Welt zu verdanken, die heute, in Gestalt von Linux, als Serverplattform vielerorts im Einsatz ist. Die Netztechnik brachte also eine Re-Zentralisierung, die aber nur die gemeinsam zu nutzenden Daten betraf:

vorher getrennt arbeitende Einzelplatz-PCs konnten nun gleichzeitig auf einen zentral gehaltenen Datenbestand zugreifen.

Unentbehrlich wurde der Mehrplatzbetrieb für die Programmierung von Geschäftsfunktionsfunktionen, weil Erwerbung und Ausleihe nur in sehr kleinen Bibliotheken auf einem und demselben Endgerät ablaufen können. Sogar in einer OPL (One-person library) gibt es doch wenigstens zwei PCs, davon einen für den Publikumszugriff.

Programmerweiterungen für Ausleihe (aLF) und Erwerbung (ORDER) wurden 1993 eingeführt. Es handelte sich jeweils um erweiterte Spezialversionen des Katalogisierungs-Kernprogramms PRESTO (ein heute veraltetes DOS-Programm).

```

msdos.cp
_Datenbank: CAT DateI# 1 Satz# 38 U26.4 Laenge: 175, frei: 0
Muir, Kenneth: Signatur: 2588-1187
Shakespeare's Sonnets / Kenneth Muir.
London : Allen & Unwin, 1979. - 179 S.
ISBN 0-04-821942-0
Personenschlagwort: Shakespeare, William
Sachgruppen: an
(A) = Alternative Anzeige (z) = zum Register (letzter Zugriff)

Datei für neue Sätze: 1 <F10> : Globale Ersetzung ausführen
← = Register Hilfe: <F1>/<Shift+F1> EXIT: <F8>

```

Abb. 5: PRESTO-Titeldaten-Anzeige

```

MSD - PRESTO
1 : Namen von Personen 'allegro'-Demo
1 schwarze, wolfgang
31=>shakespeare, william
1 shakespeare, william #
46 shakespeare, william *
1 shakespeare, william - DRAMATISCHE WERKE
7 shakespeare, william - FESTSCHRIFT
1 shakespeare, william - WERKE
6 shakespeare, william / bibliographie
9 shakespeare, william / biographie
10 shakespeare, william / drama
1 shakespeare, william / henry iv
1 shakespeare, william / henry v
2 shakespeare, william / imagery
2 shakespeare, william / komodie
2 shakespeare, william / loves labours lost
4 shakespeare, william / lyrik
2 shakespeare, william / macbeth
1 shakespeare, william / monologe
1 shakespeare, william / naturwissenschaft
1 shakespeare, william / prosa
5 shakespeare, william / rezeption
1 shakespeare, william / richard iii

Andere Stelle? Suchwort eintippen. <Sh+F8> = erweit. Regist.
* = Satzanzzeige Hilfe: <F1>/<Shift+F1>

```

Abb. 6: PRESTO-Indexanzeige

Mitte der 90er Jahre gab es mehrere hundert Anwender in allen Sparten des Bibliothekswesens.^{7 8}

Nach der Wende war es so, daß die ostdeutschen Hochschulbibliotheken alle großzügig mit Hardware ausgestattet wurden, und unter anderem erhielten sie *allegro* als Software mit dazu. Viele bauten dann ihre ersten Online-Kataloge damit auf, z.T. auch eine automatisierte Erwerbung, bevor

⁷ EVERSBERG, B.: 13 years of *allegro* : questions, demands, users. - In: Technological university libraries in the nineties. Proc. of the 15th Biennial IATUL Conference, Hamburg-Harburg, 1993. - ISBN 951-22-2212-4. - S. 222-236. (IATUL Proceedings New Ser. 3(1994), ISSN 0966-4769)

⁸ Karte des Verbreitungsgebietes und Referenzliste von Katalogen, die im Netz erreichbar sind: <http://www.allegro-c.de/ac-dbs.htm>

sie sich später einem der Verbände anschließen konnten. Darunter waren Greifswald, Jena, Leipzig und Magdeburg.

Ab 1995 wurde der Leistungsumfang des *allegro*-Systems um wichtige Neuerungen vergrößert:

1. Mehrdateikonzept: Satzverknüpfungen aller Art werden möglich (z.B. für Normdaten!)

2. Kernfunktionen von C in eine Klassenbibliothek in C++ umgearbeitet. Dadurch können leichter neue Programme entwickelt werden. Zuerst entstanden auf dieser Grundlage Vorläufer des Windows-Hauptprogramms *a99* sowie ein Konsol- und Serverprogramm (heute *acon* genannt), u.a. für den Einsatz in Web-Schnittstellen. Schon ab 1996 konnte man damit *allegro*-Kataloge "ins Netz stellen".

Jahrtausendwende : Globalisierung

Betriebssysteme und Internet

Etwa ab 1996 begann ein neues Zeitalter, in dem wir uns heute wie selbstverständlich bewegen: Man konnte Server global „sichtbar“ machen im schnell expandierenden Internet. Zunächst war der PC dabei wenig mehr als ein Terminal, aber er konnte nun Daten von beliebig vielen, weltweit verteilten Servern abrufen. Standard-Betriebssystem auf PCs wurde Windows 3, später NT und schließlich Win'95/98, womit Microsoft neben IBM zum größten Softwarekonzern wurde.

Erste Methoden zur Präsentation von *allegro*-Datenbanken im Internet konnte man rasch entwickeln und dann schrittweise verbessern. Hunderte von *allegro*-Datenbanken gelangten ins Netz, wobei mehrere verschiedene „Schnittstellen“ zum Einsatz kommen, programmiert in Perl oder PHP auf der Serverseite, in HTML+JavaScript oder in Adobe-Flash auf der Clientseite.

Auf lokaler Ebene konnte man *allegro*-Datenbanken ab 2000 endlich mit einer vollwertigen Windows-Oberfläche benutzen. Diese prägt, über Jahre hinweg weiter verbessert, noch heute bei den meisten Anwendern das

„Gesicht“ des *allegro*-Systems⁹. Zugleich damit entstand eine Makro- und Skriptsprache namens FLEX¹⁰, mit der dann Vorgänge aller Art programmiert werden konnten, bis hin zur Nutzung von Web-Diensten. Die Funktionen für Ausleihe und Erwerbung, danach auch Zeitschriftenverwaltung, wurden damit neu entwickelt und lösten die alten DOS-Programme ab, d.h. das Windows-Hauptprogramm wurde zum Universalprogramm – anwenderseitig beliebig erweiterbar dank FLEX.

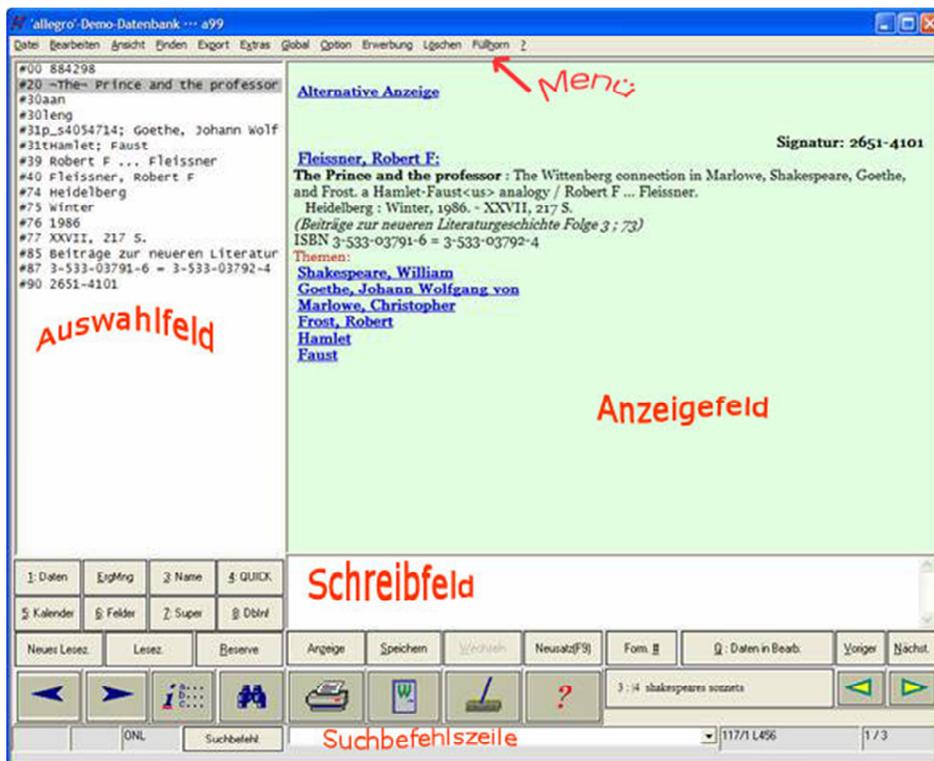


Abb. 6: a99, das Windows-Hauptprogramm ab 1999

⁹ Als Dokumentation, insbes. für das Konfigurieren und Parametrieren, gibt es das Systemhandbuch, und zwar ab Version 30 nur noch als PDF: <http://www.allegro-c.de/doku/handb30.pdf>, Aktualisierung 2013 geplant.

¹⁰ FLEX-Dokumentation: <http://www.allegro-c.de/flex/>

Programmiert wurde ab 1996 in einer IDE (Integrated Development Environment) namens Visual Studio von Microsoft, was die Arbeit mit den angewachsenen Quellprogrammen enorm erleichterte. Ähnliche Entwicklungsumgebungen, voran das System Eclipse (entstanden etwa 2001 bei IBM), kamen später heraus und sind heute die unentbehrlichen Werkzeuge vieler Softwareentwickler.

Die Zahl der *allegro*-Kataloge im Internet nahm ab 2000 ständig zu. Ein erster Spezialfall ist ab 1997 die Datenbank "Vegetationsökologie", die eine von Dietmar Brandes begründete Online-Bibliographie des Fachgebiets darstellt¹¹. Eine Anzahl von Fachkollegen nicht nur in Deutschland können ihre Veröffentlichungen per Browser selber einbringen, bis heute etwa 14.500 Beiträge. Nach diesem Vorbild erstellt auch eine Fachgemeinschaft von Wissenschaftshistorikern, die International Union of History and Philosophy of Science, eine gemeinsame Datenbasis ihrer Publikationen¹².

Im Bibliothekswesen vollzieht sich ansonsten aber auf der Verbundebene eine immer stärkere Zentralisierung der Leistungen Katalogisierung und Fernleihe. Für deren Belange sind in der Tat die größten Datenbanken die besten. Die deutschen Verbunddatenbanken sollen nach Ansicht des Wissenschaftsrats in ihrer Zahl noch verringert werden, und die Verbünde selbst bringen zudem noch ihre Daten in den WorldCat von OCLC ein. Auf lokaler Ebene besteht deshalb ein Spektrum von völliger Autarkie bei sehr kleinen Einrichtungen, die keine Verbundanbindung realisieren können, bis hin zu Verbundteilnehmern, die lokal nur noch Endgeräte haben, aber keinen Server mehr, auch nicht für die eigentlich lokalen Funktionen der Ausleihe. Was Fremddatennutzung betrifft, können jedoch auch kleinste *allegro*-Anwender mittels Z39-Clientfunktion und anderen Prozeduren von den Leistungen der ganz großen profitieren und zudem auch aus ihrem Katalog heraus zum WorldCat und zu Google Booksearch verlinken. Von Verbundzentralisierung unberührt bleiben dagegen Spezialdatenbanken, wie sie an vielen Stellen gepflegt werden.

¹¹ Literaturdatenbank Vegetationsökologie Mitteleuropas,
Startseite <http://www.biblio.tu-braunschweig.de/vegetation/>

¹² WissTecMed*Lit : <http://lit.wisstecmed.de/>

Zeichencodierung

Im Zuge der Globalisierung wuchs die Bedeutung einheitlicher Zeichencodierungen und Datenformate. Der Uralt-Industriestandard ASCII verfügt über gerade einmal 96 Zeichen und 32 Steuer-codes und ging zurück auf die begrenzten technischen Möglichkeiten der 1960er Jahre. Ein Byte hat 8 Bit, und damit kann man $2^8 = 256$ verschiedene Codes realisieren. Für die 128 durch ASCII nicht normierten Codes entstanden zahlreiche Varianten für die Sonderzeichen einzelner Sprachgemeinschaften. Das erschwerte natürlich den Austausch von Daten, insbes. solchen, die Inhalte in vielen Sprachen aufnehmen sollten, wie eben gerade Bibliotheksdaten. Aber auch die Wirtschaft brauchte dringend eine Norm der Zeichencodierung, die rund um die Welt anwendbar war und einen unkomplizierten Austausch, ohne Umcodierung, zwischen allen Sprachumgebungen ermöglichte. Diese Norm heißt Unicode¹³ und kann inzwischen mehr als 90.000 Zeichen darstellen. Auf unterster Ebene gibt es zwar immer noch mehrere verschiedene Arten der Zeichencodierung, vor allem 16bit-Unicode und UTF-8, aber diese sind auf höherer Ebene äquivalent. Im Internet ist UTF-8 die globale Norm. Die 127 ASCII-Codes sind darin unverändert präsent und werden durch je ein Byte repräsentiert, die anderen Zeichen beanspruchen je bis zu 4 Bytes im Bereich der Codes 128 bis 255.

Auch die Bibliothekswelt hat heute den Wirrwarr der frühen Jahre überwunden und arbeitet mit UTF-8. Für *allegro* sieht es so aus, daß viele Bibliotheken noch intern eine alte 8-Bit-Codierung mit einem bibliotheksspezifischen Zeichensatz verwenden, der alle westeuropäischen und die meisten osteuropäischen lateinischen Buchstaben umfaßt. Diejenigen, die mehr brauchen, können mit UTF-8 arbeiten, wenngleich dies intern einige Komplikationen mit sich bringt, denn die Indexierung arbeitet nach wie vor mit 8 Bit, und das hat mit den Problemen der alphabetischen Ordnung in den Zugriffsregistern zu tun. Zwar gibt es Ordnungsverfahren für Unicode, jedoch kann man damit in mehrsprachigen Daten keine Ordnung realisieren, die den Gewohnheiten

¹³ Deutsche Einführung: <http://www.unicode.org/standard/translations/german.html>
allegro und Unicode: <http://www.allegro-c.de/unicode/>

aller Sprachgemeinschaften zugleich gerecht wird. *allegro* reduziert daher in der Indexierung die diakritischen Sonderzeichen auf ihre Grundbuchstaben. Bibliotheken mit asiatischen Beständen können jedoch ebenfalls mit *allegro* und mit Unicode arbeiten. Sinologische Bibliotheken verwendeten früher einen Code namens Big-5, nun können sie auf UTF-8 wechseln. An der Web-Oberfläche arbeiten heute jedoch alle *allegro*-Datenbanken mit Unicode: die jeweils nötige Umcodierung geschieht mittels geeigneter Tabellen zwischen Datenbank und Oberfläche vollautomatisch.

Datenstrukturen¹⁴

Die Grundeinheit beim Datenaustausch ist nicht das einzelne Zeichen, sondern der Datensatz. Datensätze sind zwar stets Zeichenfolgen, jedoch logisch gegliedert in eine Anzahl von Elementen, auch Datenfelder genannt. Auch auf dieser Ebene gab es in den frühen Zeiten viele sehr verschiedene Normen für die logische Struktur von Bibliotheksdaten: mehrere nationale Austauschformate, darunter diverse MARC-Varianten in der angelsächsischen Welt und MAB in der deutschen. Für das *allegro*-System wurde deshalb ein parametrierbares Umwandlungs-Hilfsprogramm entwickelt, das bis heute routinemäßig vielerlei Datenkonvertierungen ausführt.

Seit etwa 2002 laufen Bemühungen um eine globale Vereinheitlichung der bibliothekarischen Datenstruktur, zugleich auch um ein gemeinsames Regelwerk für die Inhalte von Katalogdatensätzen, früher „Titelaufnahmen“ genannt. Dieser Prozeß soll 2013 mit der Einführung neuer Katalogregeln namens RDA (Resource Description and Access¹⁵) einen großen Schritt vorankommen, wobei aber Ende 2011 die Library of

¹⁴ Übersicht zu bibliothekarischen Datenformaten und Katalogregeln:

<http://www.allegro-c.de/formate>

¹⁵ Information der LC, Zugriff auf die eigentlichen Regeltexte nur mit Lizenz :

<http://www.rda-jsc.org/rda.html>

Congress eine Initiative namens BIBFRAME¹⁶ eingeleitet hat, die nach längerer Übergangszeit zu einer Ablösung des MARC-Formats führen soll.

Damit will man vor allem die bislang geschlossene Welt der Bibliotheksdaten öffnen können für andere Fachgemeinschaften, die mit Kulturgut umgehen, wie Museen und Archive aller Art. Hier sind noch keine konkret anwendbaren neuen Strukturen vorhanden - vielleicht sind es zuviele komplexe Aufgaben gleichzeitig, die man sich vorgenommen hat. Der Übergang wird noch viel Zeit brauchen und umfangreiche Bemühungen in der Software, denn vom internen Datenformat hängen alle Prozeduren und Verfahren ab, die an irgendeiner Stelle mit Bibliotheksdaten umgehen.

Mehrheitlich verwenden *allegro*-Anwender bisher weder MARC noch MAB, sondern eine formal recht einfache Datensatzstruktur¹⁷, in die sich aber die Daten aus den wichtigen Quellen (Verbünde und Nationalbibliotheken) konvertieren lassen. Eine zwingende Notwendigkeit zu einem Umstieg wird sich deshalb in absehbarer Zeit noch nicht ergeben; was man vorerst braucht, sind sog. Importparameter für neu entstehende Austauschformate.

Für die Strukturierung der Austauschdatensätze kommt zunehmend XML¹⁸ zum Einsatz, das universelle und global verbreitete Mittel zur Darstellung von Satz- und Feldstrukturen. (Offiziell: "Originally designed to meet the challenges of large-scale electronic publishing, XML is also playing an increasingly important role in the exchange of a wide variety of data on the Web and elsewhere.")

Intern, also innerhalb einer Datenbank, wird dagegen auch anderswo noch kaum mit XML gearbeitet, obwohl es eine XML-Darstellung von MARC- wie auch MAB-Daten gibt.

XML als Internstruktur innerhalb *allegro* ist, soviel kann man sagen, nicht möglich. Es ist aber auch nicht notwendig, weil es möglich ist, XML-

¹⁶ Bibliographic Framework Transition Initiative (BIBFRAME), ab 2011 :

<http://www.loc.gov/marc/transition/>

¹⁷ Das sog. "konsolidierte Format": <http://www.allegro-c.de/doku/form2004>

¹⁸ Übersicht zu XML: <http://en.wikipedia.org/wiki/XML>

codierte Fremddaten in das interne Format zu wandeln und umgekehrt aus den eigenen Interndaten heraus XML-Daten nach außen zu geben. Beides ist mit *allegro* realisiert. Zusätzlich kann ein Anwender externe Hilfsmittel¹⁹ verwenden, die den Umgang mit XML-Daten erleichtern. Man muß allerdings sagen: In der Industrie und auch z.B. bei OCLC hat es nicht an Versuchen gefehlt, "native" XML-basierte Datenbanken zu entwickeln oder wenigstens praktikable XML-Datenmodelle, die besser wären als die alten MARC-Strukturen, bis heute jedoch ohne durchschlagenden Erfolg²⁰. (MARCXML ist anerkanntermaßen noch keine Lösung, weil es MARC nur 1:1 abbildet und mit erheblich mehr Aufwand verpackt. Es leistet keine modernere Strukturierung auf logischer Ebene.)

Ab 2005 : „Ubiquitierung“

Ab 2005 wird die Verteilung der Daten im Netz immer diffuser, man spricht auch von der „Cloud“ und meint damit, daß ein Endsystem die benötigten Daten dynamisch aus unterschiedlichsten und global verteilten Beständen heran-holen kann. Der Nutzer nimmt dabei nicht mehr gezielt, für jeweils eine „Sitzung“, den Kontakt mit einem bestimmten datenhaltenden System auf, sondern Zugriffe geschehen „transparent“ über die Browsersoftware und die jeweiligen Netz-Schnittstellen der Server. Es ist, m.a.W., dem Nutzer eines PC gar nicht mehr möglich, den Überblick über alle Quellen und deren reale Lagerorte zu gewinnen, aus denen während ganz normaler Arbeit tatsächlich Daten scheinbar wie aus Wolken auf ihn herabregnen: Die früher klare 1:1-Beziehung zwischen Client- und Serveranwendung ist aufgehoben. Sollen nun Bibliotheken Mitspieler in diesem globalen Konzert sein, brauchen sie Software, die ein Endnutzer nicht mehr gezielt und direkt ansprechen und dann verstehen muß, sondern die von anderer Software mittels standardisierter Zugänge abgefragt werden kann und Antworten in normierter Struktur zurückgibt: „Linked Data“ heißt hier ein Schlagwort, und gebraucht wird dazu ein „Application Programmer’s Interface“ (API). Vorläufer dieser Bewegung sind die Z39.50-Schnittstelle²¹ und das OAI-Protokoll²². Beides wurde für

¹⁹ Übersicht: <http://en.wikipedia.org/wiki/XML>

²⁰ Native XML-Datenbanken: http://en.wikipedia.org/wiki/XML_database

²¹ Z39.50 allegro-Schnittstelle: <http://www.allegro-c.de/z3950/>

allegro realisiert und ist jeweils konfigurierbar. Hinzu treten Abfragefunktionen, die in Fremdprogramme einbezogen werden können. Solchen Fremdprogrammen wird es ermöglicht, aus einem *allegro*-Datenbestand strukturierte Ergebnisdaten für definierte Zwecke zu extrahieren. Entscheidende Komponente auch hierfür ist die Skriptsprache FLEX⁸, die für *allegro* entwickelt wurde und die einerseits im Windows-System der internen Bibliotheksverwaltung zur Programmierung und Automatisierung von Vorgängen dient, andererseits Anfragen erledigen kann, die per Internet über den avanti-Server hereinkommen.

Möglich wurde mit FLEX aber auch die passive Nutzung von Webdiensten (internet services); die unter der Oberfläche Daten aus dem Internet bei Bedarf herbeiholen und einbeziehen in eine Anwendung. Es kann sich um so profane Dinge handeln wie den Dollarkurs, aber auch um Datensätze aus frei zugänglichen Datenbanken (GBV, DNB, Z39-Server, OAI-Server, ...), die dann so erscheinen, als hätte man sie gerade selber eingegeben – der Verkehr mit dem jeweiligen Server wird gar nicht sichtbar. Komfort und Schnelligkeit sind auf diese Weise kaum geringer als bei direkter Verbundteilnahme.

Ab 2010 : Mobilisierung

Um 2010 kommen dann mobile Endgeräte (sog „Smartphones“) hinzu, ferner „Tablets“, die nur aus einem Display bestehen, also keine physische Tastatur besitzen. Diese Geräte können für viele Aufgaben alle Ressourcen im Internet an jedem Ort mit Drahtlosnetz (WLAN) oder Mobilfunkverbindung nutzen. *allegro-C* konnte Schritt halten und bietet vielen Bibliotheken eine kostengünstige Möglichkeit, im Netz präsent zu sein und darin zeitgemäße Leistungen anzubieten. Gegen Ende 2012 gelang es, eine flexible und erweiterungsfähige Oberfläche für Mobilgeräte²³ mit HTML5 und JavaScript + jQuery zu erstellen.

²² *allegro* und OAI : <http://www.allegro-c.de/oai.htm>

²³ Ein Beispiel zur Demonstration von Mobilfunktionen:
<http://www.allegro-c.de/db/bolero/app.html>

Im Mainstream

So ist *allegro* im Mainstream der Web-Entwicklung angekommen, nachdem ein anderer Ansatz, genannt a30, mit der Flash-Software von Adobe auf zu geringe Akzeptanz stieß. Die Erfahrungen damit haben aber den Einstieg in die neuen Techniken und den Neuentwurf – Arbeitstitel a35 – wesentlich befördert. Externe Entwickler können damit nun ohne tiefere Kenntnisse der *allegro*-Interna mit den weitverbreiteten Standard-Techniken ihre Oberflächen gestalten und mit der Skriptsprache FLEX die Zugriffe zur *allegro*-Datenbank einrichten, wobei zahlreiche Standard-Skripte den Einstieg erleichtern und als Beispiele dienen können. Zugleich ist es nicht schwierig, Web-Anwendungen in allen gebräuchlichen Skriptsprachen und Entwicklungswerkzeugen zu schreiben, denn der Aufruf von FLEX-Skripten, um Datenbankinhalte heranzuholen und in definierter Form geliefert zu bekommen, ist in jedem Fall möglich.

Erst mit einer Anwendung, die am einzelnen Arbeitsplatz nichts als einen Browser braucht²⁴ (wie eben a35), wird es auch endlich möglich, Linux als Endnutzer-Plattform einzusetzen. Die nur unter Windows laufenden Clientprogramme a99 und alcarta für Katalogisierung, OPAC und Geschäftsgänge haben aufwendige graphische Oberflächen, die man nicht plattformunabhängig programmieren konnte. Aus solchen und anderen Gründen haben sich allenthalben auch für Anwendungssoftware anderer Art die „browserbasierten“ Lösungen etabliert, und dies gilt nun – mit HTML5 und JavaScript, wie erwähnt – auch für Mobilgeräte. Für populäre Massenanwendungen gibt es zwar Entwicklungssysteme für sog. „Apps“, diese sind aber systemspezifisch: sie arbeiten z.B. nur auf Apple-Geräten oder unter dem Android-Betriebssystem von Google. Der Aufwand für die Erstellung solcher Programme, mit Java und jeweils spezifischen Entwicklungswerkzeugen, wäre zu hoch und wenig zukunftssicher. Die Web-Anwendung a35 wurde deshalb so konzipiert, daß sie viele Eigenschaften und interne Strukturen an eine ebenfalls HTML5-basierte Mobilanwendung "vererben" kann. Zu den ersten Anwendern von a35 zählt die Universität Oxford, und zwar auch diesmal wieder, weil damit auch ostasiatische Originaldaten bearbeitet, verwaltet und sichtbar gemacht werden können.

²⁴ Dokumentation a35: <http://www.allegro-c.de/doku/a35/a35.pdf>

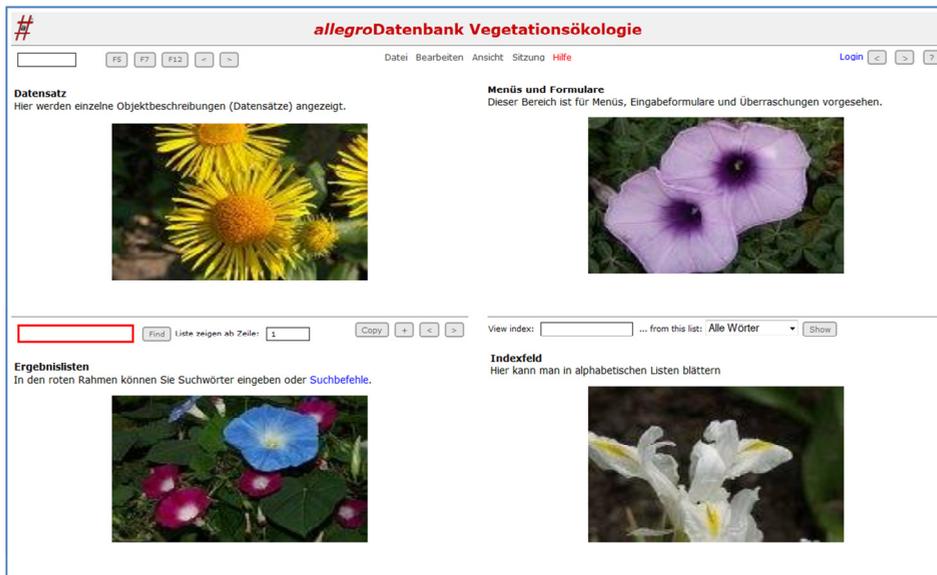


Abb. 7 : Zugriff per Browser ab 2013 mit a35 (Hier Testseite mit Pflanzenbildern)

2010-2014 : „Zurück zu den Quellen“

Im Zeitraum 2010 bis 2012 wurde der Kernbestand der Quellprogramme in C und C++ gründlich überarbeitet, dokumentiert und als Open-Source unter einer zeitgemäßen Versionsverwaltung (Subversion-Server) freigegeben²⁵. Nun sind endlich auch alle internen, sozusagen subkutanen Module plattformunabhängig, d.h. für UNIX/Linux mit dem GNU-System und für Windows mit Microsoft Visual-C++ gleichermaßen kompilierbar.

Damit sind drei wichtige Ziele erreicht:

1. Programmierer andernorts können darauf aufsetzen und nun auch auf Quellenebene eigene Funktionen entwickeln, was vorher allein in Braunschweig passieren konnte.
2. Anwender haben die Sicherheit, nicht mehr ausschließlich von „Braunschweig“ abhängig zu sein, immer in der Ungewißheit, ob

²⁵ Open-Source-Einführung: <http://www.allegro-c.de/os/>

und wie, und wie lange noch, es denn ohne Unterstützung von dort weitergehen könnte.

3. Die Quellen sind in der jetzt erreichten Form zukunftssicher, weil sie mit den gängigen, meistverbreiteten Compilersystemen übersetzt werden können.

Allerdings ist es so, daß *allegro* nie eine „black box“ gewesen ist, denn alles, was variabel sein sollte, war von Anfang an offen, auf Anwenderseite konfigurierbar und ab 2000, in der Windows-Version, im Funktionsumfang erweiterbar durch die Skriptsprache FLEX. Eben darin unterschied es sich immer von Systemen, die gezielt und mit festgelegten Strukturen und Funktionen für reine Bibliotheksanwendungen konzipiert waren. Die interne Datenstruktur und die Schnittstellen für Export und Import konnten ab 1985 frei verändert und in allen Einzelheiten den jeweiligen Erfordernissen angepaßt werden, und zwar auf einer höheren Ebene, nicht in der Programmiersprache C. Viele Anwender wurden damit in den Stand versetzt, eigene Strukturen, Indexierungen, Ausgabeprodukte und schließlich mit FLEX auch Prozesse zu erstellen, z.T. für Spezialsammlungen und Sonderprojekte. Freischaffende Supporter konnten auf der Basis *allegro-C* spezialisierte Anwendungen und neue Oberflächen erstellen und auch als Host für Internetzugänge agieren.

Die wohl größten Beispiele untypischer Anwendungen sind das Fernleih-Bestellverwaltungssystem Gefjon²⁶ (bes. für Subito-Lieferanten geeignet) und die HANS-Anwendung²⁷: ein komplexes System zur Erfassung und Verwaltung von Handschriften, Autographen und Nachlässen, entstanden ursprünglich in Hamburg, im Einsatz bei ca. 20 Bibliotheken.

Zu den Sonderfällen gehören auch eine Datenbankfassung des Katalogregelwerks RAK²⁸ und eine, in der die Lehrveranstaltungen der TU

²⁶ Dokumentation zu Gefjon: <http://www.allegro-c.de/gefjon/>

²⁷ HANS-Website an der SUB Hamburg:

<http://www.sub.uni-hamburg.de/bibliotheken/fuer-die-fachwelt/allegro-hans.html>

²⁸ Startseite: <http://www.biblio.tu-bs.de/db/rfk/>. In gleicher Weise könnte eine Datenbank des neuen Regelwerks RDA erstellt werden, jedoch sind die Regeltexte nicht frei zugänglich und zur Zeit auf deutsch noch gar nicht verfügbar.

Braunschweig ab 2005²⁹ verzeichnet sind (seit jenem Jahr gibt es kein gedrucktes Vorlesungsverzeichnis mehr).

Ein sozusagen exotisches Beispiel ist der Katalog ostasiatischer Bücher an der Universität Oxford³⁰. Als der europäische Verein sinologischer Bibliotheken (EASL³¹) um 1990 herausgefunden hatte, daß sich *allegro* für chinesische, japanische und koreanische Schrift eignete, führten einige es für ihre Kataloge ein, darunter Berlin, Heidelberg, Wien, München und eben Oxford. Bis heute existiert dieser Katalog, und zeitweise erfaßte auch die British Library in London ihre chinesischen Titel mit *allegro*. Die Bodleian Library hat noch eine weitere *allegro*-Datenbank im Netz: Die John Johnson Collection of Printed Ephemera³² ("...a fresh view of British history through primary, uninterpreted printed documents which, produced for short-term use, have survived by chance.").

Was im Open-Source-Projekt noch hinzukam, ist der Zugang zur maschinennäheren Ebene der Grundfunktionen, um auch dort ändernd und erweiternd eingreifen zu können. Sehr wichtig ist dies auch dann, wenn neue Betriebssystemversionen mit den Programmen nicht mehr klarkommen. Die Windows-Version 7(64bit) war schon so ein Fall. Die nunmehr offenen Quellen in C und C++ können mit den aktuellen Compilern von Microsoft (Visual C++ Versionen 11 und 12) übersetzt werden, und das wird voraussehbar noch längere Zeit so sein. In der UNIX/Linux-Welt ist alles mit dem dort gängigen offenen Standard (GNU-System) kompilierbar.

²⁹ Startseite Lehrveranstaltungen: <http://www.biblio.tu-bs.de/db/lv/>

³⁰ Startseite: <http://www.bodley.ox.ac.uk/dept/oriental/allegro.htm> (Zentralkatalog chinesischer Titel) und <http://www.niicat.eu/> (Zentralkatalog japanischer Titel). Dort findet man auch jeweils Dokumentation.

³¹ Homepage EASL: <http://www.easl.org>

³² Startseite JOHN JOHNSON: <http://www.bodleian.ox.ac.uk/johnson>

Vernetzte „Community“

In den ersten 10 Jahren fand die Kommunikation zwischen Braunschweig und den Anwendern konventionell über Post und Telefon statt, denn E-Mail war noch ein zu elitäres Medium.

1991 entstand dann ein E-Mail-Forum³³ mit heute weit über 100 Teilnehmern, in dem sich Endanwender und Entwickler sowie Supporter treffen, um Fragen zu stellen, zu diskutieren und Lösungen zu finden.

Vor 2003 wurden über den Fortgang der Entwicklung mit den gedruckten *allegro news*³⁴ verbreitet, danach nur noch - schneller und in kürzeren Abständen - per E-Mail (sog. *Verlautbarungen*, bis heute ca. 250).

Die Themen dieses Forums haben sich zuletzt in das Gebiet der eigentlichen Programmierung hinein erweitert, d.h. die Freigabe der Quellen zeigt hier bereits konkrete Ergebnisse. Ein separates Entwicklerforum hat sich noch nicht gebildet; nach aller Erfahrung ist es jedoch günstig, wenn die Entwickler im Anwenderforum verbleiben und dort stets auf die aktuellen Fragen der Praxis aufmerksam werden. Über viele Jahre hinweg sind aus diesem Forum die meisten Anregungen für Verbesserungen und neue Funktionen hervorgegangen.

Besonders aktiv in der Community sind mehrere frei- und nebenberufliche Supporter. Diese betreuen eine größere Zahl von Anwendern, denen Zeit und Knowhow fehlen, um die eigenen Vorstellungen umzusetzen. In vielen Fällen übernehmen Supporter auch das Hosting von Web-Datenbanken. Aus solchen Erfahrungen gingen schon viele anspruchsvolle und nützliche Vorschläge hervor.

In einer Reihe von Fortbildungen und Expertentreffen, nicht nur in Braunschweig, wurden Kenntnisse vermittelt, Erfahrungen ausgetauscht und Anregungen gewonnen für die Weiterentwicklung.

³³ Mailforum mit Archiv (durchsuchbar):

<http://sun250.biblio.etc.tu-bs.de/mailman/listinfo/allegro>

³⁴ Ausgaben 20-60 (1990-2002): <http://www.allegro-c.de/news.htm>

Unabhängigkeit von Fremdkomponenten

Ein zweiseitiges Schwert! Open-Source-Systeme setzen statt auf Unabhängigkeit immer mehr auf eine Einbindung oder Nachnutzung vieler fremder, ebenfalls offener Quellen, um bestimmte Teilaufgaben nicht selber wieder neu programmieren zu müssen. Dies gilt vor allem für komplexe Leistungen wie Rechtschreibprüfung, OCR-Texterkennung, Datenkonvertierung oder -umcodierung, besonders aber Datenbankfunktionen. Bis heute ist dies im *allegro*-System nicht der Fall, d.h. alle Komponenten sind selbstentwickelt, nur sehr kleine Teile wurden aus freien Quellen übernommen, aber vollständig integriert (z.B. für Aufgaben wie Sortierung, Suche mit sog. regulären Ausdrücken, Z39-Grundfunktionen oder Internet-Dateizugriffe). Ein Projekt wie *allegro* hätte man, so wie es über 30 Jahre gelaufen ist, nicht planen können, und ein heute zu beginnendes Projekt würde von ganz anderen Grundideen und Programmierparadigmen ausgehen. Unvermeidlich steht dies auch der Akzeptanz, wenn nicht gar dem Verständnis des *allegro*-Konzepts unter jüngeren Programmierern im Wege.

Dem deshalb natürlich z.T. erheblich höheren Entwicklungsaufwand – verteilt über 30 Jahre – stehen auch Vorteile gegenüber:

1. Unproblematische Installation: Endnutzer müssen neben *allegro* nichts anderes installieren oder separat konfigurieren.
2. Selbständigkeit: Stets unterliegen Fremdentwicklungen ihrem eigenen Versionswandel, den man u.U. nachvollziehen muß oder aber sich festlegen auf eine bestimmte Version, die man hernach selber bewahren und pflegen muß. Und hoffen, daß sie z.B. unter neuen Windows-Versionen noch läuft oder mit einem anderen Webserver. Falls nicht, kann es sehr schwierig werden.
3. Kompromißlosigkeit: Wenn eine Fremdkomponente nicht vollständig zu den eigenen Bestandteilen paßt, gibt es Anpassungsprobleme und letztlich oft Kompromisse oder Inkaufnahme von Nachteilen wie etwa Leistungseinbußen. Je mehr Komponenten, umso leichter kann es zu schwer aufzuklärenden

Wechselwirkungen kommen, umso leichter auch verliert man den Überblick.

4. Effizienz: Fremdkomponenten können i.d.R. mehr als man braucht, sie nehmen daher mehr Leistung und/oder Speicherplatz in Anspruch. Dies wird heute von Entwicklern nicht mehr als Nachteil gesehen, meist mit Hinweis auf sinkende Hardwarepreise. Doch es gibt viele Anwender, die froh sind, auf relativ alten PCs auch mit neuen Versionen immer noch flott arbeiten zu können.

allegro-C hat sich, anders gesagt, die Unabhängigkeit der frühen Jahre bewahrt, als es noch gar keine Fremdkomponenten gab. Auch die Datenbankfunktionen sind Eigenentwicklungen. Normalerweise werden Standard-Datenbanken (z.B. Oracle, Sybase oder das freie MySQL) verwendet, um Daten zu speichern. Das relationale Modell der tabellarischen Datenstrukturen hat aber für Bibliotheksdaten entscheidende Nachteile, die man nur durch umfangreiche Zusatzprogrammierung überwinden kann³⁵. In den 80er Jahren gab es noch keine genügend leistungsfähigen relationalen Systeme, schon gar nicht solche, die man hätte frei verwenden können. Inzwischen gibt es einen Trend zu Modellen, die nicht mehr auf dem relationalen Ansatz und SQL beruhen, die sog. NoSQL-Datenbanken³⁶. Darunter übrigens auch das BigTable-Konzept, auf dem das Retrieval von Google beruht. Von einer mit SQL vergleichbaren Standardisierung kann in dem Umfeld aber noch nicht die Rede sein.

Die datenbanktechnischen Eigenentwicklungen, in der Grundstruktur bis 1990 entstanden, erwiesen sich dann aber, mit späteren Verbesserungen, als genügend tragfähig bis heute. Die größte Datenbank – nur als Demo-Objekt und „Härtetest“ gedacht – besteht aus den Daten des Bayerischen Verbundes³⁷, die 2012 in einer für LinkedData-Konzepte gedachten Fassung freigegeben wurden: 24 Millionen Datensätze. Der *allegro-OPAC*

³⁵ Relationale Datenbanken im Vergleich zu *allegro*: <http://www.allegro-c.de/a-r.htm>

³⁶ Übersicht zu ca. 150 NoSQL-Datenbanken: <http://nosql-database.org/>

³⁷ Zugang BVB-Testkatalog: <http://www.allegro-c.de/db/a30/bvb.htm>

der UB Braunschweig³⁸, aus den Pica-Daten gespeist, ist dagegen mit 1.5 Mio. Titeln eher klein. In puncto Leistung brauchen somit *allegro*-Anwender noch auf einige Zeit nicht zu befürchten, ans Ende der Fahnenstange zu gelangen. Sollte es aber eng werden, wenn enorm große Datenmengen zu bewältigen sind, dann gibt es Wege, ein externes Suchsystem zu integrieren, ohne im Kern die Unabhängigkeit aufzugeben:

Solkraft

Seit einiger Zeit, wie oben schon bemerkt, werden Nachteile des lange dominierenden relationalen Modells stärker wahrgenommen. Insbesondere die großen Suchmaschinen, allen voran Google, arbeiten nicht mit SQL-Tabellen, weil diese kein genügend leistungsfähiges Retrieval von Textdaten ermöglichen. Neben den NoSQL-Datenbanken gibt es auch dedizierte Retrievalsysteme. Zu dieser Kategorie, in der es nicht um Datenverwaltung geht, sondern nur um schnelles und flexibles Retrieval, gehört Solr³⁹ (früher Lucene) von Apache. Immer mehr Systeme bauen diese Open-Source-Entwicklung in ihre Produkte ein, so z.B. auch MyCoRe⁴⁰ (für Dokumenten-Repositoryen), oder es werden zusätzliche Suchsysteme den klassischen bibliothekarischen Datenbanken an die Seite gestellt, so z.B. beim GBV. Dieser letztere Ansatz ist auch für *allegro* sinnvoll und bereits in zwei Formen realisiert worden: eine eigenständige („SolrPower“⁴¹) mit geringem Mehraufwand und eine aufwendige Lösung, die das (auf Solr gründende) Open-Source-Produkt VuFind⁴² nutzt. Die eigenen Suchfunktionen werden nicht aufgegeben und sind im Bereich der Bibliotheksverwaltung und Katalogisierung auch nicht entbehrlich, schon weil ein Solr-Index nicht völlig aktuell sein kann, aber Endnutzer erhalten dank Solr komfortable und attraktive neue Funktionen (Navigation,

³⁸ *allegro*-OPAC der UB Braunschweig mit a35:

<http://www.allegro-c.de/db/katalog/a35start.php>

³⁹ Startseite Solr : <http://lucene.apache.org/solr/>

⁴⁰ MyCoRe: <http://www.mycore.de>, damit arbeitet auch die Digitale Bibliothek der UB Braunschweig: <http://www.digibib.tu-braunschweig.de/> (s.a. Beitr. v. C. ELSNER in dieser Festschrift)

⁴¹ Beschreibung zu "SolrPower": <http://www.allegro-c.de/doku/solr.htm>

⁴² Vufind: <http://vufind.org/>

("... a library resource portal designed and developed for libraries by libraries.")

Facettensuche, Ranking etc.). Ferner kann man in VuFind das Suchen in anderen, externen Beständen mit den eigenen Daten kombinieren, wie es der GBV inzwischen auch unterstützt und an einigen Pica-Bibliotheken schon in Betrieb ist⁴³. Auf *allegro*-Seite braucht dazu nichts verändert zu werden. Alles bleibt auch ohne Zusätze funktionsfähig, Solr kann aber hilfreich hinzutreten und erhält seine zum Indexieren benötigten Daten in der geeigneten Form aus *allegro* per Export im MARC-Format. Mit Solr als Suchsystem sind dem Datenumfang kaum noch Grenzen gesetzt.

Verwendung in der Ausbildung

Bibliothekarische Ausbildungsstätten, z.B. Berlin, Köln und Leipzig, nutzen gern *allegro*, weil sich Studierende die Demo-Version schnell mal eben selber installieren können, um dann "hands on" das ganze Instrumentarium und Procedere der bibliothekarischen Datenverarbeitung aktiv erleben zu können.

Verbreitung des Produkts

Seit 1997 wird alljährlich die jeweils aktuelle Version auf eine CD, jetzt DVD, gebrannt und publiziert, zusammen mit einer *allegro*-Version des Katalogs der UB (z.Zt. 1.5 Mio Titel) und anderen Beispielen.

Die aktuelle Version und alle ansonsten nötigen Dateien sind jederzeit vom FTP-Server zu beziehen⁴⁴.

Ausblick

Das Bibliothekswesen und damit das Feld der bibliothekarischen Datenverarbeitung ist im Jahre 2013 noch immer in heftiger Bewegung. Heute sind Bibliotheken Bestandteile einer umfassenden, ständig nach Verbesserungen drängenden, globalen Informations-Infrastruktur. Information über Bücher und deren Bereitstellung ist nicht mehr allein die Sache von Bibliotheken und Buchhandel, sondern diese erscheinen gelegentlich fast an den Rand gedrängt von ganz neuen Akteuren, die das Auffinden

⁴³ Übersicht: <http://www.gbv.de/wikis/cls/Vufind>

⁴⁴ zu finden unter <http://www.allegro-c.de/download.htm>

und Präsentieren schriftlicher Aufzeichnungen revolutioniert und den Handel damit globalisiert haben. Weil Information über Bücher und Aufsätze heute über viele Quellen online auffindbar ist, die Texte selbst in wachsenden Mengen ebenfalls, wird das Aufgabenfeld, das „Geschäftsmodell“ der Bibliotheken, immer mehr hinterfragt, und am meisten die Kataloge. Vom Bücherlagerhaus, das eine Bibliothek lange war, hat sie sich gewandelt zu einer vernetzten Stätte, die aufgezeichnetes Wissen – wo immer es liege - zugänglich macht. Dazu gehört das Lizenzieren von Verlagsproduktionen genauso wie das digitale Aufbereiten älterer Quellen, für die ein Bedarf besteht, die aber bislang nur physisch in Bibliotheken existieren und schwer zugänglich sind; im Extremfall Unikate, wie in den HANS-Anwendungen. Die Masse der noch nicht digitalisierten Quellen ist dabei noch lange nicht vernachlässigbar, und sie besteht keineswegs nur aus vergilbenden Altdokumenten. In jedem Fall braucht die Bibliothek ein System, das die verschiedensten Bestände integriert verwalten kann und die einzelnen Objekte auffinden hilft. Erwerbung, Fortsetzungsverwaltung und Ausleihe bleiben unentbehrlich, und damit auch die Software dafür, solange die Bibliothek noch erwirbt und ausleiht.

Das "Integrierte Bibliothekssystem" (engl. ILS = Integrated Library System), wie wir es seit langem kennen, existierte 1980 erst in Anfängen und nur auf Großrechnern. Heute ist es für die Softwareindustrie wohl kaum noch kommerziell interessant, andererseits existieren aber im Open-Source-Bereich neben *allegro* noch immer keine leistungsmäßig vergleichbaren kostengünstigen Lösungen, jedenfalls im deutschsprachigen Bereich. Alternativen zu *allegro*, mit andern Worten, sind noch immer rar.

Den Benutzerkatalog jedoch kann man aus dem ILS lösen. In jedem Fall muß er sich nun der Vernetzung stellen, wobei die Schnittstelle zum Publikum auch ein "Discovery-System" übernehmen kann, wie etwa VuFind. Das Bibliothekssystem braucht ferner Funktionen (s.o. Stichwort „Ubiquitierung“), mit denen von außen per Software abfragbar wird, ob ein gesuchtes Dokument vorhanden ist, sowie ob und wie man es erhalten kann. In dem Moment, da der eigene Bestand im Verbundkatalog oder WorldCat mit enthalten ist, werden jene viel größeren Kataloge zu wesentlich attraktiveren Quellen für den Endnutzer, zumindest im Normalfall. Nicht ganz so im Fall von Spezialsammlungen mit eigenen

Besonderheiten der Erschließung, und in diesem Bereich finden sich die meisten *allegro*-Anwender.

Auch Google Booksearch kann oft ein interessanter Startpunkt für den Endnutzer sein, weil über die Funktion „Finden in einer Bibliothek“ dann auch ein Weg via WorldCat in den eigenen Bibliotheks-katalog führen kann. Der Katalog im engeren Sinne wird an Bedeutung also abnehmen, die Funktionen der internen Katalogdatenbank - und damit ihre Software - als Schaltzentrale zum Bestand und zur Verfügbarkeit von Ressourcen sowie deren Erfassung bleiben aber wichtig⁴⁵.

Anhang : Mitwirkende Personen

Die Förderung durch das Ministerium, wie in der Einleitung erwähnt, hat viele Früchte getragen. Wir listen hier nur kurz die Entwickler und einige ihrer Beiträge auf, die im Laufe der Zeit beteiligt waren.

MATTHIAS EVERS (1991-)

Netzwerkbetrieb: Aufbau des ersten Novell-Netzes, u.a. für den ersten *allegro*-OPAC. Wurde dann zum Netz-Administrator der UB, vor allem für das Pica-System.

DR. PETER PFEIFFER (1991-1992)

UNIX-Experte, richtete C-Programme für die Kompilierung unter UNIX ein. Außerdem wichtige Beiträge zur Konstruktion der Module für Erwerbung und Ausleihe. Wechselte zur Herzog August Bibliothek Wolfenbüttel als DV-Leiter. Erarbeitete dort *allegro*-Implementierungen für die ersten Jahre der Datenerfassung für VD16 und VD17 ab 2001, die auch an der BSB München zum Einsatz kamen.

⁴⁵ EVERSBERG, B.: Zur Zukunft der Katalogisierung. Vortrag auf dem Österreichischen Bibliothekartag in Linz, 2004. <http://www.allegro-c.de/formate/zk.htm> (zuletzt aktualisiert 15.7.2008)

SIEGFRIED BUSEMANN (1991-1992)

Abgeordnet vom HBZ Köln zur Entwicklung einer NRW-Version, die u.a. in Münster, Köln und Bonn zum Einsatz kam. Wechselte dann zur GBV-Verbandzentrale nach Göttingen.

DIERK HÖPPNER (1992-2003)

Zunächst Fertigstellung des Erwerbungsmoduls ORDER. Entwicklung der ersten Web-Schnittstelle sowie des Serverprogramms *avanti* und der ersten Version des Konsolprogramms *acon*.

Entwickelte ferner das Bestellverwaltungs-System namens „Gefjon“ für Subito: ein in Python geschriebenes Skriptsystem mit einer speziell konfigurierten *allegro*-Datenbank für die Fernleihbestellungen. Nachnutzung an einigen anderen Bibliotheken. Wechselte zur TIB Hannover.

PETER HARTWIG (1992-1995)

Ausleih-Komponente *aLF*. Wechselte dann zur Büchereizentrale in Lüneburg und erarbeitete die *ÖB*-Version⁴⁶, die seitdem dort in Eigenregie (jedoch kompatibel) weiterentwickelt wurde.

HEINRICH ALLERS (1993) abgeordnet für einige Monate aus Oldenburg:

allegro-Lehrbuch mit Übungsmaterial (für das DOS-System)

Später war H. Allers *allegro*-Betreuer am Goethe-Institut für ca. 80 Auslandsbibliotheken)

CORD VELTKAMP (1996-1998)

DFG-Projekt: Entwickelte einen Z39.50-Server und verbesserte einige der C++-Programme sowie auch den *avanti*-Server und das Programm *acon*.

MARTIN BUTKUS (2003-2006)

Leistete Beiträge zu Verbesserungen an vielen Stellen; Reorganisation der Kombination *avanti+acon* zur technischen Verbesserung der Web-Anbindung. Richtete eine Versionsverwaltung ein (SVN). Entwickelte außerdem (ohne *allegro*) ein Online-Anmeldeverfahren für Pica-LBS sowie

⁴⁶ Büchereizentrale Niedersachsen, *allegro*-ÖB:
<http://www.bz-niedersachsen.de/allegro-oeb.html>

ein Verwaltungssystem für Semesterapparate. Ging dann in die Industrie als Entwickler.

CARSTEN ELSNER (2006-)

Viele Einzelbeiträge, bes. im Serverbetrieb, u.a. Umstellung der Versionsverwaltung auf Subversion. Dann Wechsel auf eine Dauerstelle mit Hauptaufgabe „Digitale Bibliothek“ (mit MyCoRe und Goobi)

ERIK NEUMANN (ab Dez. 2011)

Neuordnung und Modernisierung der *allegro*-Website und Überarbeitung von Dokumentationen, z.B. für FLEX; Web-Schnittstelle für Mobilgeräte.

In den frühen Jahren gab es sogar **Schülerpraktikanten**, die als Programmierer-Autodidakten einige brauchbare Beiträge leisteten. Sie hatten zu Hause den zum CBM 8032 kompatiblen Heimcomputer C64⁴⁷.

⁴⁷ Darstellung der Historie des C64: <http://oldcomputers.net/c64.html>