

第5章 NRDF システムの運用・支援システムの開発

JCPRG（原子核反応データベース研究開発センター）は 80 年代半ばまでは、NRDF というデータベースの検索システムおよび検索結果のグラフ表示システムによる利用者サービスを行ってきた。その後、NRDF システムの機能・性能の向上のために、各種 NRDF ユーティリティプログラムやグラフデータ読み取りシステムなどが開発されてきた。これらについては第 4 章にまとめられている。

当センターは国際的な核データセンターネットワーク (NRDC) のメンバーとして活動している。その責任の 1 つが NRDF に登録した核データを NRDC の核データ交換書式である EXFOR に変換して IAEA の核データサービス部 (NDS) に送ることである。5-1（千葉正喜氏）ではこの変換プログラムの作成経緯について述べられている。

NRDF データベースの作成のためには、まず、論文に掲載されたグラフデータの読取と、論文からの書誌情報、実験条件、および実験データの採録・コーディングが行われる。前者については 5-2（鈴木隆介氏）で、後者については 5-4（大塚直彦氏）でそれぞれのシステム作成の経緯についてまとめられている。なお、これらの機能や工程に関しては、科研費の試験研究や研究成果公開費、事業費の時期にいくつかの試験研究の実績の積み重ねがあった^[1]ことを付記しておく。

次に、NRDF の維持と管理のためには、核データの書誌情報・実験条件とグラフデータのファイルをマージし、NRDF 文法との整合性を検査し、索引ファイルを更新して、1 つの論文を単位とする核データを NRDF データベースに登録する、という処理プロセスを行う。第 4 章の NRDF システムにおいては、データベース管理システム (DBMS) を含め、これらの処理の大半を担うモジュールが作成されていた^[2]。しかし、大型汎用計算機から UNIX を OS とする複数のワークステーションの利用に移行する 1990 年代後半になると、NRDF システムから、独自設計の DBMS の部分を一般的かつ多目的に使われている DBMS に置き換えることが緊急の課題であった。検索システムについては個別の報告があるが、NRDF システムの更新全体については時期的にはかなり遅くなってから報告が出ている^[3]。

NRDF の作成のプロセスでは、データ収集と入力を担当するメンバー、(旧) 管理運営委員会や(新) センター会議のメンバーの間で、各種の会議や共同作業の連絡、作業に関する相談などは電子メールで行われることが多い。これらを一括してアーカイブするシステム「Stock」については、執筆依頼者との相談の結果、5-5（鈴木隆介氏）に年報を再録することになった。

Web の機能を活用する NRDF データベースの検索・作図システムについては 5-3（升井洋志氏）で、新しい情報流通プラットフォーム IntelligentPad または最近の Webble World を利用して NRDF データベースの検索システムを作成する試みについては 5-6-1（千葉正喜氏）と 5-6-2（江幡修一郎氏）で述べられている。この 2 つの試みの間の時期には、前者の発展として CONTIP^[4, 5] という検索システムが作成・配布された。他にも VBL の研究員の活動として様々なシステムやツールが開発されているが、これらについては第 7 章 7-1 にまとめられている。

[1] 荷電粒子核反応データファイル ユーティリティ開発報告書（1985），風間裕，NRDF 年次報告 88，(1989) p.2；及び本章 5-2 に詳しい文献紹介がある

[2] 片山敏之，NRDF 年次報告 87，(1988) p.14；能登宏，*ibid.*, p.30

[3] 升井洋志，NRDF 年次報告 99，(2000) p.15

[4] 千葉正喜，NRDF 年次報告 95，(1996) p.2，NRDF 年次報告 96，(1997) p.10

[5] 升井洋志，大林由英，NRDF 年次報告 98，(1999) p.56

5-1 NRDF の EXFOR への変換

千葉 正喜（元札幌学院大学）

私が NRDF のデータを EXFOR に変換することになった直接のきっかけは、田中先生から 1981 年の夏も終わり秋になる頃だったと思うが、呼び出されて、「NRDF データを EXFOR に変換するプログラムを 3 か月間で開発すべし」と言われたことに始まる。そして「日本では研究的な仕事しか評価しないが、このような実務的な仕事が、国際的には評価される」と強調されたことを印象深く記憶している。そのとき、富樫さんが手掛けていた EXFOR の構造を調べ作成された資料が渡された。富樫さんは科研費で雇われて NRDF データベースシステムの開発とメンテナンスを担当していたと思うが、EXFOR への変換システムまではできないということで、科研費プロジェクトのメンバーに加えてもらっていた私にその仕事が回ってきたものと思っている。

3 か月という期限がきられたことで、EXFOR のシステムマニュアルと NRDF の開発報告書、それに富樫さん作成の資料を参考にしてシステムの設計に専念した^[1]。この課題に取り組む前に、大型計算機センター長だった大野公男先生から PLOT79 という作図プログラムのソースプログラムを収録した磁気テープが渡され、大型計算機センターのライブラリとして移植を済ませていた。このときの作業から、大きなプログラムを単一機能のサブルーチンモジュールの集合として作成する技法を学んでいたのは、幸いであった。

何をどのようにすればいいのだろうかなどと悩むことはなく、やればできるという自信があったこと思い出す。この年の年末・年始は絶好の時間だと思い、ひたすらプログラムのコーディングをした。

年が明けて、テストデータでデバッグし、プログラムが通るようになって、田中先生に報告し、チェックを受けた。そして、プログラムで変換して作成した EXFOR エントリーを IAEA の岡本さん宛てに TRANS E001 として送った。岡本さんからは、「私宛に送っていただいたので、スムーズに処理ができた」と感謝された。

これは後で知ったことであるが、E001 エントリーは同じ ID で、別に（多分手作業で）作成されたエントリーが既に IAEA に登録されていた。IAEA に対して日本からデータを送るという計画・試みが以前からあった、NRDF 作成と同じ頃またはそれ以前からの物ではないかと思われる。

EXFOR にはコード辞書のデータベースがあり、これが磁気テープで得られ、コードレベルの変換はこの辞書をベースとする NRDF-EXFOR コード対応辞書のデータベースを利用するように変換プログラムを改定した。

この改定で NRDF から EXFOR への変換プログラムは一応の完成と考えて、かなりの数の NRDF データを変換プログラムで EXFOR に変換した。この変換が（核反応データとして）物理的に正しく行われているか、EXFOR の規則に正しく適合しているかが問題であった。田中先生にチェックを受けましたが、この変換結果を持って IAEA に行き議論してもらうべしとなって、1983 年 8 月夏休み期間の 1 週間 IAEA に単身ではじめて行くことになった。英語もよくできないで、よく行ったものである。IAEA ではレムメル（H.D. Lemmel）さんが対応してくれ、変換結果を丁寧に見ていただいた。

この変換プログラムは NRDF のすべてのデータを EXFOR へ変換しようという考え方であったので、EXFOR に正しく変換できていない部分がかかなりあった。EXFOR では、EXFOR に定義されていないコードや物理量は、その定義を提案し承認を受けることになっている。NRDF には、EXFOR に定義されていない物理量や複数の項目を組み合わせることが必要な情報があり、これはプログラムで機械的に変換することができないことが判明した。そこでこの問題を解決するために、片山さんと一緒に取り組むことになり、NRDF に含まれる情報の核物理的意味レベルで EXFOR の情報との対応関係を分析し、EXFOR が求めているデータのみを抽出・加工して EXFOR に変換するという考え方に変更することで解決するという結論を得た。

この結論に従って NRDF-EXFOR 翻訳変換プログラム「NTX データコンパイラ」を改定し、論文にまとめて発表した^{[1], [2]}。私としては、これでこの仕事は一段落のつもりでいた。

それから 1 年かそれ以上過ぎたある日、理化学研究所の橋詰さんから連絡が入った。IAEA の会議で札幌グループは切るという話になっているとの切迫した電話を頂いた。田中先生にこのことをお伝えした。これを契機にして、NRDF のデータを変換プログラムで EXFOR に変換し、それが物理的に正しいか否かのチェックを受け、TRANS テープに編集して IAEA に送る仕事をするようになった。そして、IAEA の核データセンター会議とそのテクニカルミーティングにも、NRDF のメンバーとして参加してきた。Lemmel さんから NRDF から送るデータには誤りがないとの温かい評価を受けたことを記憶している。この会議への参加では、札幌学院大学からの研究支援の旅費の補助を頂いた。

IAEA に定常的にデータを送り、毎年核データセンター会議に参加するようになった頃、NRDF グループは IAEA では「Study Group 札幌」と呼ばれていたが、1992 年の核データセンター会議で Study Group ではなくちゃんとした呼び名にしてはという提案があった。どういう名前にしようかと言いよどんでいたら、マックレーン (V. McLane) さんから「JCPRG」にしたらという案がだされ、今日の IAEA における NRDF に対する呼び名になったのである^[3]。

また、チェクレエフ (F.E. Chukreev) さんからは日本に行く機会を作ってほしいとも言われましたが、わたしはそれを判断する立場ではなかったし、NRDF からの IAEA への参加者を交代すべきと考えていた。

NRDF という日本が独自に考案した核データのデータベースであったから、その EXFOR への変化を通して国際協力になったと思う。もし、それがなくて、はじめから EXFOR でデータ収集を行っていたら、このように継続してデータベース活動ができたであろうかと思う。真の国際協力の在り方が、この NRDF にあらわれているように思われる。

- [1] 千葉正喜, 片山敏之, 田中一, 「荷電粒子核反応データベースのデータコンパイラの開発」, 北海道大学大型計算機センター テクニカル・レポート No.5 (1982) 9-29
- [2] M. Chiba, T. Katayama, H. Tanaka, A database translator of nuclear reaction data for international data exchange, *Journal of Information Science* Vol.12, No.4 (1986) 153-165
- [3] 千葉正喜, 「資料: 1992 テクニカル NRDC 会議に参加して」, NRDF 年次報告 92 (1993) 92-100

5-2 JCPRG におけるデジタイザについて

鈴木 隆介（北海道大学）

はじめに

NRDF は、論文情報、実験情報そして実験データから構成されます。最初の二つの構成要素を論文から抜き出し、NRDF へ変換することが NRDF の狭義のコーディングです。辞書で定義されているコードを用いて論文の情報を NRDF へ変換、もしくはコメントとして入力する作業となります。もう一つの構成要素である実験データは、解析や研究で直接使われる肝の部分です。ここで大きな役割を果たしているが、デジタイザと呼ばれるシステムです。昔は、図 1 のような機器を指し、最近ではコンピュータ上で用いられる同様のソフトウェアを指します。私は NRDF の採録においてどれだけデジタイザが使われているかを詳細に調べたことはないのですが、文献[1]によると、NRDF のデータのうち 25% はデジタイズされたデータとあり、また、国際的なデータベースである EXFOR の採録では、文献[2]によると、デジタイザにより読み取られたデータは 40% に上るとあります。核データのデータベースの構築において、実験データは論文の著者から直接入手し、デジタイザの出番がないという状況が理想的です。現在は、JCPRG 設立当時と比べると実験家の協力により直接得られるケースが増えていますが、協力が得られない場合には、JCPRG の責任で論文のグラフから数値を読み取る必要があります。現在でもデジタイザは活躍しています。

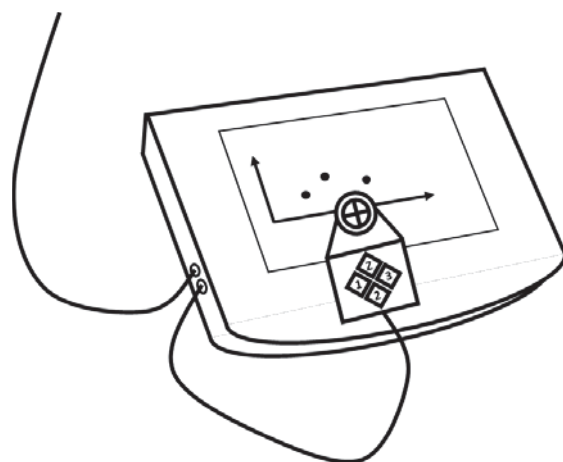


図 1: デジタイザ

JCPRG におけるデジタイザの歴史

JCPRG におけるデジタイザの詳細な歴史は、年次報告を見て頂くことになりますが、簡単にその歴史をまとめます。JCPRG では、当初 (1980 年) は大型機センターのデジタイザを用いていました [3]。1984 年以降、JCPRG において論文から数値を読み取るシステムを NRDF の採録活動と共に発展させてきました。1984 年に風間、田中により GRADIS、1989 年に GRADIS2 が開発され [4-6]、1991 年に岡部によりデジタイザのアップデートが行われました [7]。

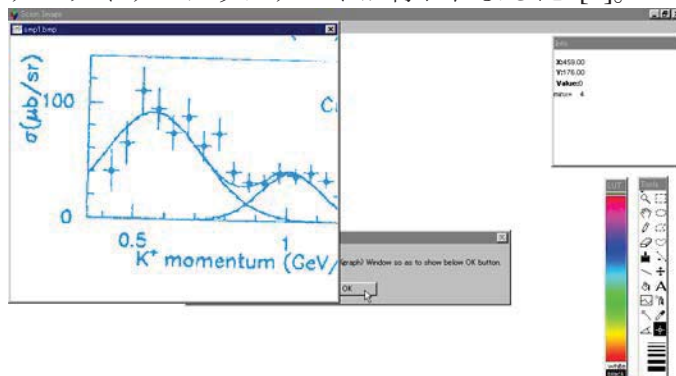


図 2: SyGRD のスクリーンショット [8]

ここまでのデジタイザと呼ばれる図 1 のようなデバイスによって数値が読み取られていた時代です。その後の数値読み取りは PC のディスプレイ上で行われるようになりました。ソフトウェアとしてのデジタイザは、近江による SyGRD [8] が最初です (図 2)。

このシステムは、1998 年から 2005 年までの長い間採録に使われました。しかし、マクロプログラムとして開発されていたため、プログラムサイズに限界があり、ユーザの希望を反映させるには限界がありました。また、システムのインストールには、SyGRD だけではなく、画像ソフトウェアの設定も必要であり、採録者が各自の端末で使用することは、(著者が知る限りは) 行われていませんでした。そのため SyGRD は、基本的には JCPRG の共有 PC 上で利用され、数値読み取り作業をされている芦澤氏、吉田氏でなければ、デジタイズの作業は、敷居の高い作業でした。

その後、NRDF の採録に採用されたのが、新井により作成された GSYS [1] と呼ばれるシステムです (図 3(左図))。このシステムは Java アプリケーションなので、基本的にはどのプラットフォームでも使用することができ、数値読み取り作業の敷居が下がりました。

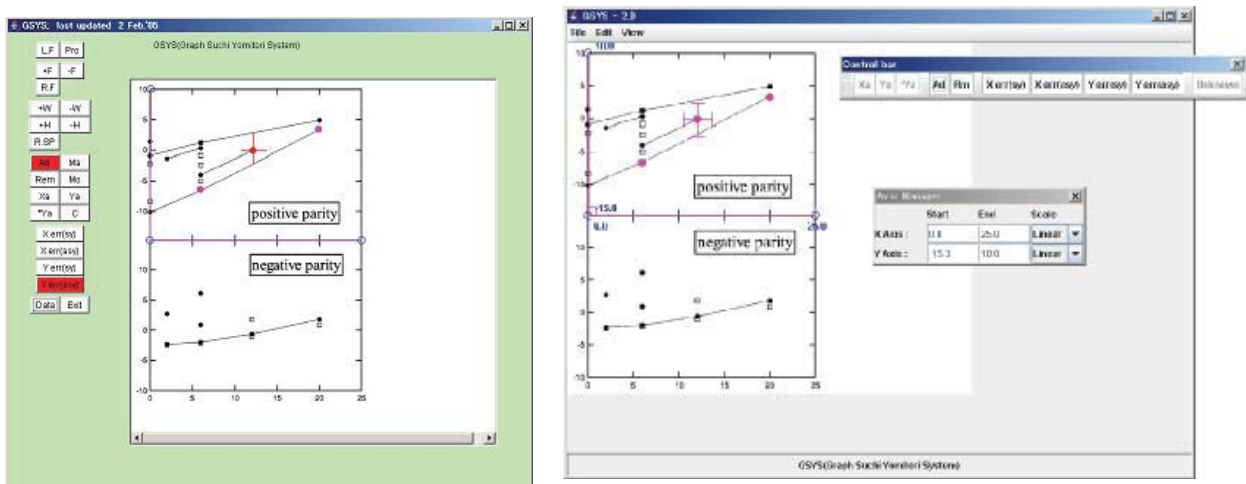


図 3: 初代 GSYS のスクリーンショット(左図)。GSYS2.0 のスクリーンショット(右図)。

この GSYS を作成した新井は最初の版 (狭義の GSYS) を作ったのちすぐに北大から異動になり、その後は、著者が中心となり開発が行われました[9-11]。GSYS の近年のバージョンは 2.X という番号が付けられていますが、これは著者が責任を持って開発した版ということを表しています。

GSYS の次期バージョンである GSYS2.0 は、2005 年 2 月から開発を開始し、2006 年 4 月に公開しました。この版では、フィードバックシステムと呼ばれる、数値データをデジタイザに読み込ませる機能が追加されました。この機能により作業を中断することや、一度読み取られたデータや一般的なデータの再利用、数値データの確認を可能にしました。また、図 3(右図) を見てわかるようにインタフェースを一新されました。作業領域を最大限に使えるようになったり、操作についての柔軟性も上がりました。今であれば当然ですが、ダウンロードしたソフトウェアをダブルクリックで起動できるようになったのもこのバージョンからです。

GSYS2.0 後の開発は伊藤が中心に行い、主に自動軸読み取り機能を実装したものが GSYS2.2 として 2006 年 12 月に公開されました[10]。その後しばらく伊藤が中心に開発が続きましたが、伊藤が留学、鈴木が北大病院へと異動し、GSYS2 の開発は一度中断しました。その後、鈴木が開発を再開し、自動点認識機能や Sarov のデジタイザが実装している部分的な拡大機能 (図 4 のスクリーンショット右下) の実装 (Glass window 機能)、ポイントの点の形の追加などの改良を行い、2010 年 4 月に GSYS2.4 として公開しました[11]。

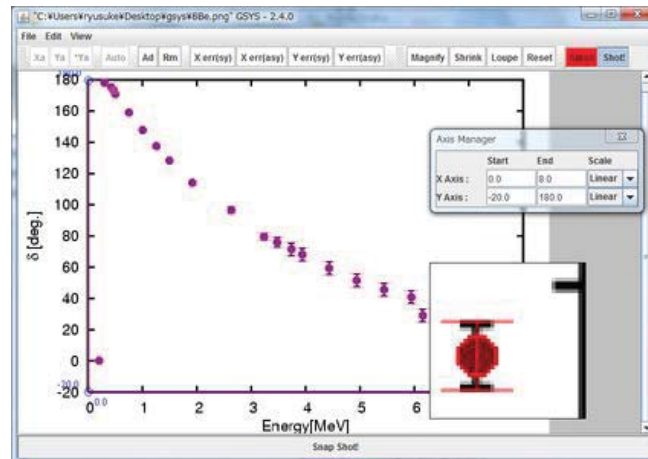


図 4：最新版の GSYS2 のスクリーンショット (GSYS2.4)

デジタイザの将来のアップデート

GSYS は、バージョン 2.4 が公開後、2.4.X とマイナーアップデートが続けられています。今後は、2.6 という新しいアップデートされたバージョンを公開予定です。この版の GSYS には、TODO 機能が実装されました。この機能は、実際の作業で使われていることを想定していますが、コードドリフタリングを目的としたものでもあります。自動点認識機能の改良、Glass window の拡張、ErrReDo と呼ばれる誤差データを全ポイントに適応する機能などが実装されています。その後のバージョンとしては、コードドリフタリングの 2 段階目として、よりオブジェクティブなコードベースへの移行とそれをベースとしたアクティブなシステムへの開発を予定しています。

国際的な位置づけ

この文書を執筆している時点では、核データに関連するデジタイザのソフトウェアは広く使われているものとして、3 つあります。Sarov の InpGraph[12]、中国の GDgraph[13]、そして日本の GSYS[14] です。2012 年にウィーンで開催された Benchmarking of Digitization Software というタイトルの Consultant Meeting [2] では、汎用で使いやすい GSYS、新しく開発され、精度が期待される GDgraph、歴史のある InpGraph という印象でした。この会議では国際的なものとしては、2 回目のデジタイザのベンチマーク試験が行われました[15]。この試験において、参加者の 13 人中 10 人が GSYS を使用していましたので、GSYS が広く使われているということがわかります（このベンチマーク試験参加者のうち日本人は 4 名でした）。現在では、デジタイザはソフトウェア形式で配布されているので、各システムを世界中の誰もが使用できる環境にあります。また、GSYS は核データの読み取り以外の目的でも使われることがあるようです。

数値読み取りシステムの精度について

これまでに、前述のベンチマーク試験を含めて国際的な数値読み取りシステムのベンチマーク試験が 2 回行われており、数値の質に関する認識が高まっています。これまでの結果では、適切に使われる限りでは、妥当な精度での読み取りが行われることが示されています。ただし、読み取り者によるミスが起きると、単純なミスでも読み取られた数値に大きな影響を与える可能性があります。グラフ座標の位置の指定のミスや、その座標の値の入力を間違えると、数値が系統的にずれたり、数値そのものが本来の値と比較して大きく外れてしまいます。ベンチマークテストの結果からも、

そのようなミスが起きる可能性が示されており、このようなミスは、核データの信頼にも影響するものと思いますので、極力減らすことが重要だと思います。これは個人の責任ではなく、グループとして対応すべき種のものと考えます。読み取りシステムを使う場合には、そのシステムの特性をグループで把握すること、使用者への教育、マニュアルの作成、データベースに登録する前の確認方法等をグループの責任で決めておく必要があります。

また誤差の要因として、読み取るグラフの質も影響します。画像をスキャンしたものであれば、解像度が影響するでしょうし、昔の論文ですと、手書きでグラフが書かれていましたので、これらをどのように分類して、精度として定量化していくことが今後の課題と考えています。また、読み取りにおいては画像の形式は、PDF のようなベクタ形式がベストと考えますが、数値の読み取りでは、png のようなラスタ形式の画像が使われています。

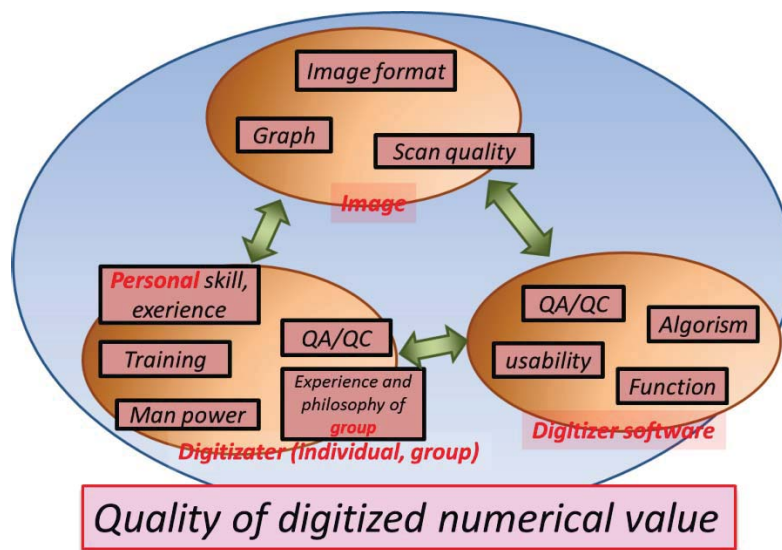


図 4：デジタイズにより読み取られた数値データの精度について

終わりに

私が関係させて頂いたデジタイザについて、これまでの歴史と最近のシステムについて書かせて頂きました。私は **GSYS** と呼ばれるシステムの開発を行いました、その期間中には、たくさんの方々にお世話になりました。その時々感謝の意については、これまでの年次報告書に書かせて頂きましたので、本書では割愛させて頂きますが、これまであまり名前が挙げられていなかった内藤健一さんについて少し触れさせて頂きたいと思います。**GSYS** が作成される前ですが、当時北大近辺に水曜日に安くなる回転寿司屋があり、水曜になると良く、「水曜寿司の日」といって当時ポスドクの内藤さん、新井さん、学生の著者でお寿司屋さんに行っていました。そこでは、当時興味があった **Java**、オブジェクト指向について内藤さんとお話をしたことを覚えています。内藤さんは当時から、「これからはアスペクト指向だよ！」と僕の理解を超えたお話しや、プログラミングの楽しさを語ってくれたことを覚えています、そういう議論を聞いて新井さんも **Java** に染まったのは...と僕は勝手に思っています。また、そういう議論ができた内藤さんがいたからこそ今でも楽しくプログラミングが続いたのではと考えています。

GSYS の開発は僕にとってはとても楽しいものでした。多くの利用者から直接、もしくは海外の方からはメールで感想を頂くことがありました。大変うれしく、かけがえのない経験をさせてもらいました。**GSYS** のコードに手を加えはじめた当初は、ちょうど長男が生まれた時でした。自分の股にちょこんと座らせながらプログラミングを行っていたことを覚えています。2013 年には、NRDC

の Consultant meeting に呼ばれました。この会議では、これまで名前だけだった sarov の方々や、今まさにデジタイザの発で花を咲かしている中国の開発者とも議論をすることができました。またお互いリスペクトしあう雰囲気でしたので、非常に良い関係を構築できたのだと思います。GSYS が各国の方々から評価され、今までかけた時間が無駄ではなかったということを感じられ大変幸せでした（本来は家庭に費やさねばならない時間を GSYS のために使っていたので、正直なところ、これまで後ろめたい気持ちがありました）。Marina Mikhaylyukova 氏と S. Dunaeva 氏が、しみじみと GSYS は使いやすいと会議の中で話されていたのが印象的でした。また、参加者からは論文化というお話ができましたのでぜひこれを実行したいと考えています。この会議に私が参加できたのは、IAEA の大塚氏のご尽力によるところが大きく、彼への感謝でこの文章を終えたいと思います。

参考文献

- [1] A. Minoguchi et al., GSYS: Development and usage of a software to read-in and digitize the graphical data, INDC(JPN)-194/U, (2005) 19-22
- [2] N. Otuka, V. Semkova (ed.), Meeting summary INDC(NDS)-0629, IAEA (2013) p3.
- [3] 「1, 核データ活動」荷電粒子核反応データファイル作成報告書 (1981)
- [4] 田中一, 風間裕「デジタイザによるグラフ読み取り変換システム (GRADIS)」荷電粒子核反応データファイルユーティリティ開発報告書 (1985) 1-2
- [5] 風間 裕「NRDF 入力用 グラフ・データ入力システム (GRADIS) 使用の手引き」荷電粒子核反応データファイルユーティリティ開発報告書 (1985) 3-14
- [6] 風間 裕「グラフ・データ読み取り変換システムの強化 (GRADIS Ver. 2)」荷電粒子核反応データファイル年次報告 No.2 (1988) 2-16
- [7] 岡部 成玄「デジタイザによるグラフ読み取り変換システムの更新」荷電粒子核反応データファイル年次報告 No.4 (1990) 2-8
- [8] H. Ohmi, Development and User's Manual of Graph Reading System with Customized Image Analysis Software, NRDF Annual Report 12, (1998) 2-20 (Japanese); Development, installation and user's manual of SyGRD (System of Graph Reading and numerical data Displaying with image analysis software), NRDF Annual Report 15, (2001) 50-85
- [9] R. Suzuki, GSYS2 Manual, NRDF Annual Report 19, (2005) 10-44
- [10] S. Ito and R. Suzuki, GSYS2.2 Manual, NRDF Annual Report 20, (2006) 39-76
- [11] R. Suzuki, GSYS2.4 Manual, NRDF Annual Report 20, (2010) 3-25
- [12] S. Taova, Development of EXFOR Digitizer Wizard – New Interface, INDC(NDS)-0629, IAEA (2013) p16.
- [13] GDGraph, Chen Guochang, Jin Yongli, Wang Jimin, Introduction of the Digitization Software GDgraph, INDC(NDS)-0629, IAEA (2013) p27.
- [14] R. Suzuki, Introduction, Design and Implementation of Digitization Software GSYS, INDC(NDS)-0629, IAEA (2013) p19.
- [15] V. Semkova, Benchmarking of Digitization Software (Memo CP-D/761): Results and Discussions INDC(NDS)-0629, IAEA (2013) p42.

5-3 Web 上での NRDF データ検索システム構築

升井 洋志（北見工業大学）

「インターネット上のデータを、Web ブラウザを用いて検索し可視化する。」2014 年の現在、これは非常に当たり前のことであり、PC のみならずスマートフォン等を用いて様々なデータの検索・利用を誰もが普通に行っている。しかし、インターネット初期の 1990 年代では、まだ発展段階にあったと言える。

JCPRG のデータ公開活動は、発足当初は北海道大学の大型計算機センター（現情報基盤センター）にあるコンピュータ上のデータベースを用いて行われており、他大学等のユーザにとって現在の検索サイトの様な自由なサービスとして利用することは難しかった。1995 年、Microsoft 社の Windows95 が発売されると同時に一般家庭への PC の普及が爆発的に広がり、インターネットインフラが整うことと合わせて、Web を用いたデータ検索・利用が身近なものとなった。この状況に合わせ、JCPRG においても Web におけるデータ検索ページを 1996 年に公開している^[1]（図 5.3.1. 参照）。このページにより、入射粒子やターゲット、著者といった項目について検索が可能となり、実験値のデータについてはグラフ化したものを表示する機能を備えていた。しかし、検索後に表示されたデータ本文は NRDF のコードそのものであり、データを利用するためには NRDF のデータ構造およびキーワードについての知識が必要であった。

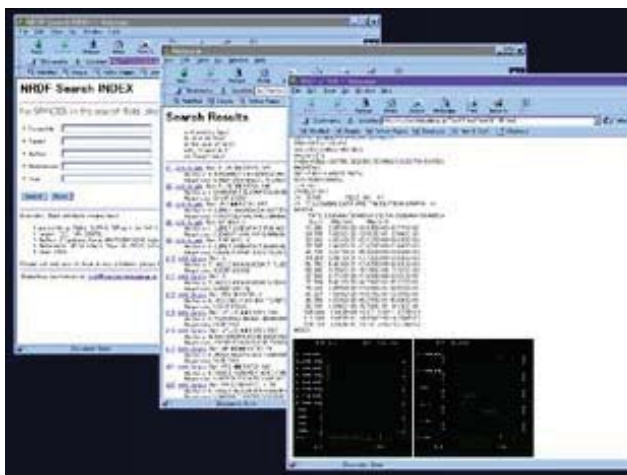


図 5.3.1. 初期のデータ公開ページ

そこで、NRDF のコードについて知識の無いユーザ（一般のユーザ）を対象に、より使いやすいページの開発を行うプロジェクトを 1998 年に開始した^[2]。当時指針としていた要件は以下の 3 つであった。

1. Web ブラウザによる「なじみの深い」インタフェースの実装（フォーム検索）
2. システム開発を容易にする構成（Perl によるシステム開発）
3. 利用形態のプラットフォームフリー化（Web ブラウザでの利用）

上記のうち、1. のインタフェースについては、Web ブラウザのフォームを用いた検索キーワード入力ページと、Yahoo! や Google（当時はまだ Yahoo! の方が主流であった）といった検索サイトの結果表示ページと同様の結果表示ページを構築した。当時、Web サーバを構築する上で主流となっていた言語は Perl で、CGI も多くは Perl によって作成されていたため、我々も Perl を用いてシステム開発を行った。思えば、今ほどセキュリティに対して要求が高くなかったため、サーバへの DDoS 攻撃対策やデータベースの脆弱性攻撃（SQL インジェクション等）対策に気を使う必要がなかったことも、システム開発の敷居を下げてもらえた要因であろう。

では次に、少し具体的に当時開発したシステム^[2]の解説を行う。検索システムはユーザの利便性向上の為、以下の機能を持たせた。

- NRDF の辞書と対応させることによる書誌情報、実験情報等の表示
- データダウンロード用のリンク作成
- 文献中のグラフデータの一覧表示
- 各グラフデータの個別表示と、対応する数値データの抽出

図 5.3.2. 検索項目入力ページ

インデックスファイルとのパターンマッチであるので、Author の検索時において例えば、「TANAKA」と入力した場合、「HATANAKA」も検索結果として表示される。実用性を向上させるには、厳密一致のオプションが必要であろう。

検索結果は一覧で表示され、検索キーワードがボールド書体でハイライトされるようになっている。この機能も現在の検索サイトでは標準のものである。検索結果には Title, Author, Reference, Reaction の 3 項目が表示され、検索の手がかりを与えている (図 5.3.3.)。各データの閲覧には表示される D 番号にリンクが作成されているので、それをクリックすればよい。現在の主流に従えば、JavaScript 等の機能を用いて、カーソルを合わせた際に詳しい内容のサムネイルを表示する機能を実装することも考えられる。

システムの実装として SQL を用いたデータ検索を行っていなかったため、検索はインデックスファイルを作成し、それに対する Perl によるマッチング処理により実現していた。このため、検索項目はそれほど多くなく、Projectile, Target, Author, Reference, Year, Energy range の 6 つであった (図 5.3.2.)。Reference は、検索時に NRDF のコードとの対応付けが行えるように、一覧へのリンクを用意した。

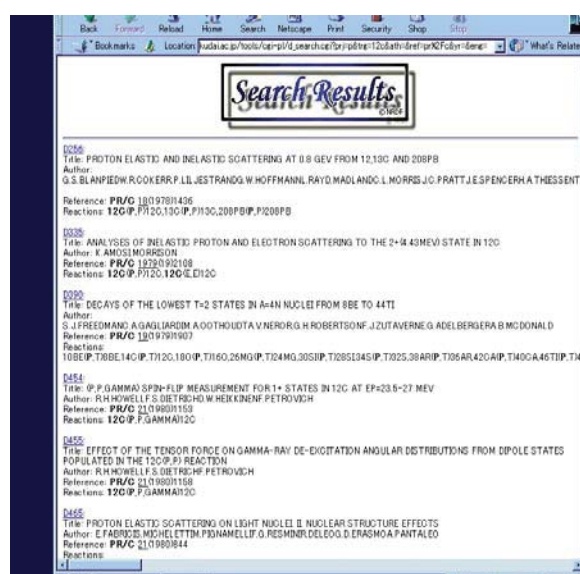


図 5.3.3. 検索結果

このシステムと 1996 年バージョンとの大きな違いは各データ(D 番号)を個別表示した際における表示機能の部分であり、本システムでは、書誌情報、実験情報、データ数値の 3 カテゴリーを分けて表示している (図 5.3.4.)。また、それぞれの項目には NRDF のコードそのままではなく、辞書と対応させてユーザが理解できるようにした内容が表示される。これにより NRDF のコードについての知識が無くても検索結果の意味を理解することが可能となっている。

論文を読んだ場合と同様の理解が得られるよう、Author においては著者と所属、実験数値においては図の番号と反応式が対応するようにした。

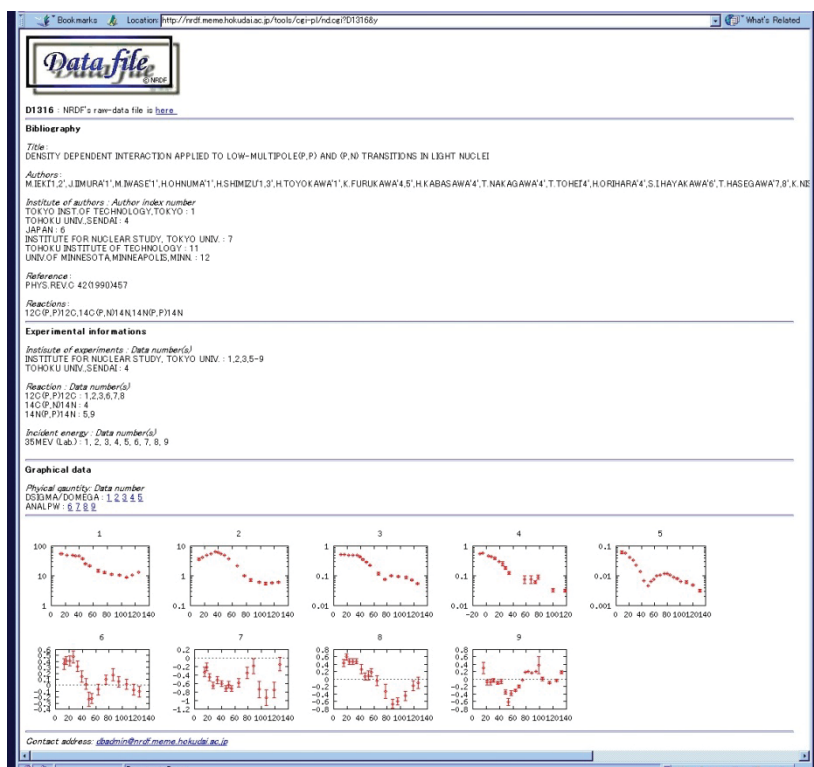


図 5.3.4. 検索結果別表示

ページの下の方には数値データがサムネイル表示される。これはサーバ上の Perl スクリプトによって Gnuplot を起動し、検索があった際に数値データのサムネイルを作成するものである。サーバの負担を軽減するため、一度作成したサムネイルは保存しておき、インデックスファイルに「作成済み」のフラグを立てるとい、いわゆるキャッシュ処理を行った。膨大なデータを扱う局面になった場合には、この辺りの加減はサーバの処理能力向上に伴いキャッシュせずにリクエスト毎に作成した方が適当である可能性もあるが、当時のサーバのハードウェアとしての処理能力、データの絶対量を考慮に入れると、キャッシュ方式の方が適当であっただろうと考えられる。

NRDF のデータファイルそのものが必要である場合は、ページ上部のリンクから生データを表示することを可能としている。これによりそのまま NRDF のデータを表示するのではなく、ユーザにリンクを辿らせることでワークション置き、取り付けにくさを防いだ。

数値データの図をクリックすると、さらに詳細な数値データのページが表示される (図 5.3.5.)。このページの図は縦軸および横軸をログスケール表示に変換することができ、さらに微分断面積のような通常ログスケールで表示されるものはデフォルトでそ

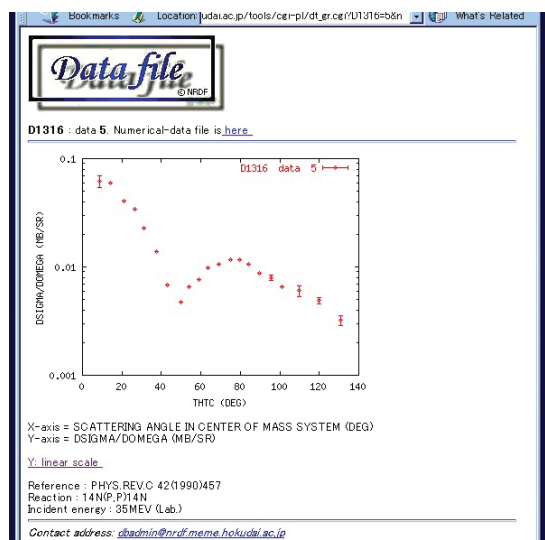


図 5.3.5. 数値データ表示

の表示となるようにした。

以上が 1998 年に開発を行ったデータ検索システムであり、1999 年の NRDF 年次報告書^[2]にまとめられている。2014 年までの 15 年の間に、コンピュータやネットワークを取り巻く環境は大きく変化してきた。それは単に処理能力の向上や回線速度の高速化という量的な部分だけでなく、仮想化やクラウドといった利用形態の質的な部分も大きく変化した。たとえば、現在のウェブサービスはクラウド上で行われることが通常で、サーバの物理的な位置や接続形態は本質ではなくなっている。

1998 年頃（つい最近まではこれが通常であったが）は、サーバ開発にあたっては物理サーバを用意し、そこに Linux をインストールしシステム構築を行う必要があった。また、稼働後はサーバがダウンしないように無停電電源装置（UPS）を整え、定期的なデータバックアップを行うことでサーバの安定運用を担保していた。ネットワークについても同様で、サーバの置いてある部屋まで確実にネットワークが到達している必要があり、非常に小さい原因、例えば部屋のハブが故障しているという状況だけで簡単にサーバへの到達性が失われる。考えてみれば、これは非常に「手作り」な状況であり、実際にサーバは私の机の上に置いてあり、ともすればなにかの拍子にサーバの電源コードが抜けてしまう可能性もあったはずである。もし現在の技術を用いてサーバを構築するのであれば、そういった心配ごとから開放されるように確実に仮想化サーバを用いているであろう。また、予算が許せばクラウドサービスを利用して、サーバの物理的な位置も問題とならないようにしているであろうと考えられる。

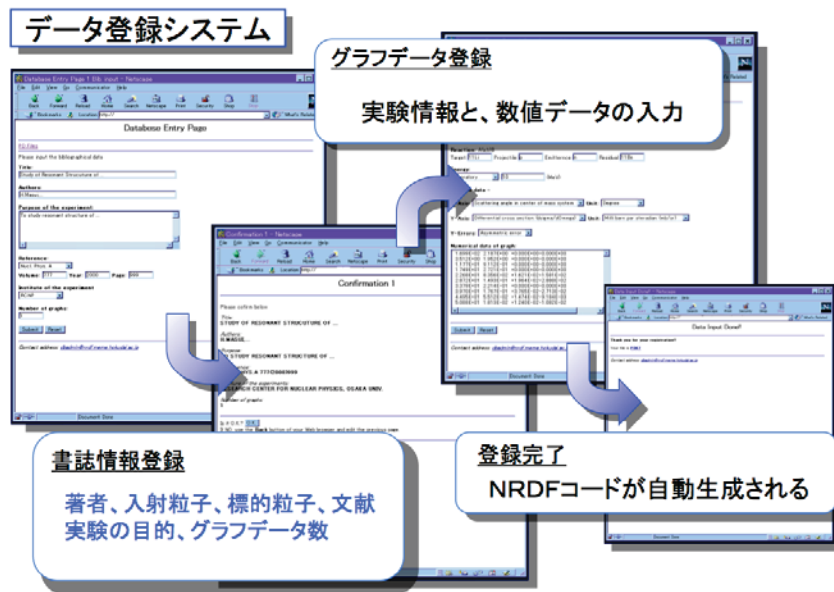


図 5.3.6.
試作の入力システム

本システム開発段階において、NRDF コードからユーザへの翻訳という流れを逆にたどれば、NRDF のデータ入力システムに発展可能であることがわかってきた。我々が当時示した試作のデータ入力システムは、NRDF の完全なコンパイルシステムではなく、開発していた検索システムに表示される部分のデータを入力することで NRDF のコードを作成するものであった。入力の流れは図 5.3.6.に示すように、まず書誌情報を入力し、次に実験情報およびグラフの数値データを入力するというものであった。検索システムに表示される項目のみを入力するので、入力したデータは他の NRDF データと全く同じフォーマットで検索システム上から閲覧可能となっている。

ただし当時はシステムの可能性を提唱したに過ぎず、またデータコンパイレーションとは単にデータを入力するだけでなく、データ管理の組織的な運用が必要であることから、このシステム自体は本運用されることは無かった。しかし、この考え方は後に大塚氏によってデータ入力システム「HENDEL」へと発展し、このHENDEL^[3]はNRDFおよびEXFORのデータ採録活動において非常に重要な役割を担っているシステムとなっている。

データの公開、認知度向上において、検索システムの果たす役割は非常に大きい。これは現在のインターネットの状況を見れば明らかで、閲覧の入り口は「検索」からである。逆に言えば、検索サイトに出てこないデータ（ページ）は、直接 URL を知っている人以外に対しては存在しないのと同じことであり、検索に出てこない状況を指して「村八分」ならぬ「Google 八分」なるスラングも存在する。

JCPRGの活動においても、検索ページの充実はデータ利用を促進することのみならずJCPRGの活動そのものの認知度向上に非常に重要なものであると考えられる。国内の実験データをコンパイルしているわけであるから、国内の実験データは論文を検索しなくてもJCPRGの検索ページに行けば数値データも含めて全て見ることができる、という状況が本来は理想のはずである。そういったことを実現するには、使いやすく、かつ網羅的にデータの整っているWebページというのは必要不可欠で、1999年から2014年の15年間でシステムや端末等様々なことが時代とともに変わったが、「ブラウザを用いてデータを閲覧する」という部分は不変であった。おそらく、今後もWebよりもさらに画期的なものが登場しない限りは「ブラウザによる文書閲覧」の部分は変化しないであろう。よって、今後もこのWeb検索ページは必要であり、重要な活動の一つ^[4]であると考えられる。

ただし、逆に言えば不変であるということは、活動対象としては重要でも研究対象としては重要性も新規性も無い、ということになる。大学で活動を続ける上においては、研究対象としてのNRDFの立ち位置がある程度必要で、最近の流行である「ビッグデータ」としてNRDFを捉えることも方法の一つかもしれない。それは単に収集した論文データを検索によって閲覧するだけでなく、データ全体から得られる研究者・研究機関・年代依存性等を分析することによる新たな知見の獲得があるとこの目的は満たされるのではないであろうか。

参考文献

- [1] 大西明,片山敏之, NRDF Annual Report No.10, (1996) pp.2-9.
- [2] 升井洋志, NRDF Annual Report No.13, (1999) pp.15-26.
- [3] 大塚直彦, NRDF Annual Report No.15, (2001) pp.12-49.
- [4] セルゲイ=コレノフ, 内藤謙一, NRDF Annual Report No.16, (2002) pp.39-50.

5-4 エディタ(HENDEL)の誕生

大塚 直彦（国際原子力機関）

誕生の経緯

核理論研究室の院生だった頃、先輩方が論文の数値データをよく分からない記号とともにパソコンに入力しているのがひどく面倒な作業に見え、核データのアルバイトはできるだけ避けるようにしてきました。年報の最後の構成員一覧によれば、私の名前が登場するのは 2000 年度のことのようです。これは私にとっては博士後期課程最後の年です。ちなみに 2000 年の 9 月末から 10 月にかけて IAEA で私の前任者にあたる Schwerer が来札してますが、当時、核データが自分の生業になるとは夢にも思っておらず、札幌滞在中の彼と会話を交わしたことは数えるほどでした。理由は簡単で、当時のわたしは英語をほとんど話すことができなかったからです。

その 2000 年度の終わり頃に VBL の研究員のポストが一つ空き、候補者として挙がっていた人が赴任しなくなった関係でそのポストが私に回ってきて、2001 年度の頭から私と核データのお付き合いがはじまりました。高エネルギー重イオン衝突のシミュレーションを学位論文の課題として行っていた関係で、当初大西さんから与えられたテーマはウェブを使ったオンラインの計算システムでした。具体的には、ウェブページから標的・入射粒子・入射エネルギーなどを入力すると、JAM という研究室の先輩の奈良さんが作成したコードが動き出し、計算終了後に反応のアニメーションとともに粒子のラピディティ分布が表示される、というシステムでした（現在も稼働している）。このシステムの作成に必要な Perl も CGI スクリプトも私はプログラミング経験がありませんでしたが、やはり VBL 研究員として NRDF のウェブ検索・作図・編集システムを開発された升井さんからこのあたりの技術のイロハを伝授いただきました。NRDF 運営委員会の議事録（2001 年 5 月 21 日）によれば、5 月のうちにこの開発は一段落したようです。

その 5 月の末、NRDC 会議を終えてウィーンから札幌に戻られた加藤さんが NRDF から EXFOR への変換率の低さをなげかれ、その変換率を改善して世界をあっと言わせることくらいどうしてできないのか、というようなことをいわれました。既に升井さんが NRDF のウェブエディタを試作されており、NRDF と EXFOR の出力を同時に行うエディタをウェブツールとして作成できないか、とは思っていたのですが、Perl や CGI を用いたツールが一つ完成したこのタイミングででた加藤さんのこの一言が、エディタ (HENDEL) 作成を開始する直接のきっかけとなりました。

開発の過程

HENDEL の開発過程で苦勞をしたことはいろいろとありますが、主には以下の 3 つの課題があったように思います：

- (1) 機能の実装に伴うプログラミング上の課題
- (2) NRDF/EXFOR 同時出力に必要な NRDF と EXFOR の辞書の整備
- (3) NRDF/EXFOR 同時出力に都合良い NRDF 出力の NRDF の規則との整合性

(1)は主には値（コード）を辞書から選択する部分についてです。HENDEL のコード入力プルダウンメニューからの選択が一般的ですが、選択対象となる値が 100 を超えるような項目（たとえば研究所名や雑誌名）をプルダウンメニューから選ぶのは現実的ではありません。そこで、別窓に候補を表示してそこでコードを選択すると親窓に選んだものが表示される、という機能を JavaScript を用いて実装することにしました。ところが類似の例題を掲載した JavaScript 関連の書籍等がなく、思い通りの動作を完全に実装するのにかなり時間を要しました。

(2)はある一つの値をエディタ上で選択して、それを NRDF や EXFOR に出力する際に、選択した値に対応した NRDF あるいは EXFOR のコードが存在しない、または一意に定まらない、という問題です。セクターを実装するにあたっては NRDF または EXFOR の辞書をそのまま HENDEL 上に表示し、他方のコードへはシステム中で変換がおこなわれるようにしました。例えば検出器に関して、HENDEL は EXFOR の辞書をセクターとして採録者に提示し、選ばれた検出器の NRDF のコードへの変換をシステムが行います。これを実装する上では EXFOR に検出器コードを追加する必要があります、Memo CP-D/E042（2004 年 4 月 26 日）はこの目的で作成されたものです。セクターとして NRDF ではなく EXFOR の辞書を表示することで、中性子入射反応の研究にしか用いられないような装置も選べるようになっていきます。一方、標的の化学形状のように NRDF にしか辞書が存在しないものに関しては、当然 NRDF の辞書がセクターとして表示され、EXFOR にはその展開形がフリーテキストとして出力されます。以上のような入出力を実装する上で、EXFOR の辞書へのコードの追加と平行して NRDF の辞書にもコードを追加する必要が生じました。当時、能登さんが辞書作業部会(NTX-WG)を毎週リードしてくださるようになったおかげで、月に一度の運営委員会を待たずに色々と議論ができたことに大いに救われました。

(3)はいわゆる「括り出し」に関する問題です。例えば炭素（単体標的）と水素（CH₂ 標的）に関する次の 4 つの表を採録することを考えます：

炭素弾性散乱角度微分断面積	¥¥DATA,1 (NRDF), Subentry 2 (EXFOR)
炭素弾性散乱偏極分解能角度分布	¥¥DATA,2 (NRDF), Subentry 3 (EXFOR)
水素弾性散乱角度微分断面積	¥¥DATA,3 (NRDF), Subentry 4 (EXFOR)
水素弾性散乱偏極分解能角度分布	¥¥DATA,4 (NRDF), Subentry 5 (EXFOR)

NRDF であればこの状況は

```
¥¥EXP,1[2;  
CHM=ELM;  
¥¥EXP,3[4;  
CHM=CH2;  
¥¥EXP,1,3;  
PHQ=ANGL-DSTRN;  
¥¥EXP,2,4;  
PHQ=ANALPW;
```

と採録しますが、EXFOR では全ての表に共通である場合を除いて、複数の表に共通な情報を括り出して記述することができません。開発の段階ではこの括り出しを実装したエディタのイメージがわからず、現在の HENDEL では全体に共通な情報(DATA0A, DATA0B)あるいは個々の表に個別の情報(DATA1, DATA2,...)の二種類の入出力が可能となっています。これにより NRDF の出力が

```
¥¥EXP,1;  
CHM=ELM;
```

PHQ=ANGL-DSTRN;
¥EXP,2;
CHM=ELM;
PHQ=ANALPW;
¥EXP,3;
CHM=CH2;
PHQ=ANGL-DSTRN;
¥EXP,4;
CHM=CH2;
PHQ=ANALPW;

と冗長となりますが、この点については2001年6月25日の運営委員会でご了解を頂いたようです。ただ、NRDF的な括り出し入力の方が明らかに編集の手間が少なくなるという場合もあり、この辺りについては次世代のエディタ開発で改めて検討する余地のある部分かも知れません。

稼働開始とその後

運営委員会の議事録をめくると、2001年10月1日の運営委員会では、私が「ウェブを用いたエディタの試作品が完成した。」と報告しており、また同年11月26日の運営委員会では「新規ファイル編集機能の部分については新しい試みとして了承する(βバージョン)」「既コーディングファイルの再編集機能の部分については、必要な情報が完全にエディタに反映されるかどうか確認を進める。」とあります。「再編集機能」とは過去に作成されたNRDFのファイルをHENDELが読み込み、必要な編集を施してNRDFとEXFORに出力することです。新規ファイル編集機能がNRDFとEXFORへの同時出力であるとすれば、この再編集機能は従来でいうところのNRDFからEXFORへの「変換」にあたるといえるかも知れません。年報No.15(2001)掲載のHENDELマニュアルによればD/E1735～D/E1759の論文が2001年度に初めてHENDELを用いて採録されたようです。現在のEXFORのマスターファイルのHISTORYレコードがHENDELによる採録日を記録しているのであれば、D/E1745(2001年10月19日 採録者：合川+大塚)とD/E1735(2001年10月20日 採録者：鈴木+大塚)が、HENDELを用いた採録による最初期の出力ということになりそうです。

さて、HENDELが他センターにはじめて紹介されたのは、私のNRDCデビューともなった2002年のパリのNRDC会合でのことです。NRDCというのがいったいどういう人達の集まりかも知れぬまま、WP2002-31(=INDC(NDS)-434, p144)という作業文書を用意してパリにでかけて発表を行いました。NRDFからEXFORへの変換の歴史のような話をイントロとして延々としたために発表後に加藤さんが苦い顔をされた一方、北大のようにNRDFとEXFORを平行して行うセンターのやり方としてこれはなかなか良い、とVicki McLaneがいつてくれたことなどをよく覚えています。

HENDEL稼働開始後の(採録者の目から見ても分かりやすい)改善事項のうち、主なものとしては、たとえばNRDFとEXFORのそれぞれに設けられていた入力項目の統合という点があげられます。似たような内容をNRDFとEXFORのそれぞれについて繰り返し入力するというのは採録者にとって大きな負担であり、これをできるかぎり減らしたい、という意識がありました。先に挙げた3つの問題点の(2)はこの段階で少しずつ解決されたものです。例えば稼働当初のHENDELではNRDFとEXFORのために検出器の情報を独立して入力する必要がありましたが、例えば2004年になされた検出器のコードに関する提案CP-D/E042などを通じてNRDFとEXFORの入力項目が統合

されました。ちなみに、現在も NRDF と EXFOR のために個々に入力を要求される項目としては物理量とヘディングがあります。

入力項目の統合以外では、例えば EXFOR 出力時の EXFOR チェックコード CHEX の自動実行と結果表示が挙げられます。今でこそ CHEX をコマンドラインで実行させる人は少なくなりましたが、私が採録を開始したころは CHEX を使うためにウィーンにある NDS の VMS サーバにログインする必要があり、そのような中、コマンドラインなしにボタン一つで CHEX が実行できたのは目新しいことだったはずです。また、本稿執筆現在、JCPRG のサーバで稼働している NRDF のウェブ検索・作図システムはやはり現行の EXFOR/ENDF のウェブ検索・作図システムを改造したのですが、NRDF のマスターファイルの内容を解釈して MySQL データベースに落とし込むツールを開発する際に副産物として CHEN という NRDF のチェックコードを開発し、これを HENDEL に実装しました。

また、一般の採録者が知らない HENDEL の「裏機能」として、毎年の年報で報告される入力データの一覧の TeX 原稿を作成する機能があります。その年度に採録された論文の表題・著者などの一覧は HENDEL を用いれば簡単に出力ができるわけで、NRDF 年報 No.16(2002)以降、最新号（センター年報 No.2)に至るまで、HENDEL はこの用途にも活用されているようです。

サーロフのエディタ(“EXFOR Editor”)の誕生

先述した通り 2002 年のパリでの NRDC 会議で HENDEL は一定の評価を受けましたが、その後、HENDEL が他センターに普及する、という状態には今のところなっていません。Memo CP-D/433 (2005 年 5 月 2 日)によれば、この後 2005 年の 4 月 26～28 日の 4 センターの小会合の席で NDS に対して”NDS will contact JCPRG about availability of their EXFOR editor for all compilers, and possibility to place it on other sites (NNDC and NDS) as well.”というアクションがでたようですが、私に具体的な問い合わせはなく、このアクションが HENDEL の普及のきっかけになる、ということはありませんでした。一方、同年 10 月の NRDC 会合では NNDC が EXFOR のためのエディタの開発開始の提案を WP2005-4 という作業文書で、またサーロフ (Sarov)が”EXFOR editor”の開発に関する最初の報告を WP2005-2 という作業文書で行っています。HENDEL は NRDF と EXFOR の双方への出力に配慮したエディタであったため、それが必ずしも EXFOR のみへの情報の入力に最適化されていないという問題があり、そのこともあって NRDC の場で他センターに対して HENDEL を売り込む、ということを私は特に行いませんでした。そのこともあって、いまは NRDC では”EXFOR Editor”といえばサーロフのエディタを指すことが一般的です。

ただ、サーロフの”EXFOR Editor”には色々と問題点もあるようです。ひとつには操作性が良くないという点があげられ、これは開発者が EXFOR の採録の経験者ではないということから来ているように思われます。また、HENDEL では採録者ができるだけ出力側の規則にとらわれずに入力ができるよう配慮されていますが、サーロフの”EXFOR Editor”は必ずしもそうなっていません。例えば”EXFOR editor”では REACTION コードでの放出粒子を軽い粒子から重い粒子に採録者が順に並べる必要がありますが、HENDEL ではそのようなソートはシステムが勝手にやってくれることになっています。NRDC のリエゾンという現在の職務柄、このサーロフの”EXFOR Editor”にも NRDC の一定の立場を持たせる必要があります、これまでは彼らの”EXFOR Editor”はコーディング入門者に向いている、というような立場を取るように国際的な場ではしてきました。ただ、これまでインドで

行われた2度のワークショップでの参加者の反応を見ると必ずしもサーロフの”EXFOR Editor”の評判は良くなく、また提出されたエントリには HENDEL ではありえないようなミス（カラムの位置がおかしいなど）が頻繁に見受けられます。

今後の展望

昨年(2013 年)10 月のカザフスタンのアルマティで行われた第四回のアジア核データベースシンポジウムの後に、当地での滞在を数日延長して EXFOR の採録方法を当地の学生に教えました。そこでは思い切って HENDEL を採用してみました。先方で採録のとりまとめをしてくれている学生の方が優秀なことにも大変に救われているのですが、この滞在後5カ月ほど経った現在までに、アルマティのグループでは既に9編の論文の採録を HENDEL 上で行い、その殆どは既に他センターに送信されました。採録が一段落したと先方から一報があればできあがったエントリーに目を通す、ということを繰り返すのですが、ファイルを電子メールでやりとりしたりせず、エディタの入出力画面を共有して採録上の助言ができる、というのはかなり便利だな、と感じています。この点、スタンドアローンのソフトウェアを別々に使っていたのでは、何かトラブルがあってもどちらに原因があるのかはつきりせず、困ったことになりそうです。これまでインドでの隔年の EXFOR 採録ワークショップではサーロフの”EXFOR Editor”が採用されてきましたが、今年(2014 年)の12月頃に予定されている次回のインドのワークショップでは HENDEL を採用してみようか、とインドのセンター長(Saxena)が言ってくれています。

このようにサーロフのエディタには随分遅れをとりながらも、HENDEL は利用者を日本国外にも広げようとしており、この機会に色々と改良をしてみたいと思っています。これまでは IAEA での任期の問題もあり、HENDEL の開発のような案件のために一定の時間を確保することは無理がありました。ただ、これからまだしばらく IAEA に在職することになりましたので、たとえば毎年一定期間（1～2週間）札幌に滞在するような形で、HENDEL をまた少しずつ共同開発し発展させることも可能なのではないか、という気がしています。

5-5 荷電粒子核反応データ採録活動におけるメールアーカイブシステム "Stock"

鈴木 隆介（北海道大学）

Stock システムの作成した当時、著者は NRDF の採録者として活動しておりました。NRDF の採録と関わることで感じたのは、具体的な採録方法や、自分の担当していない論文の採録過程や採録で行われている議論等の情報を得ることが大変難しいということでした。Stock システムが作られる前の状況は、採録の情報は基本的に紙ベースで残されておりましたので、情報量やその情報へのアクセスにはやはり制限がありました。このような環境を改善し、著者と共に活動されていた他の方々の努力を可能な限り次世代にも残せるようにしたいというのが、stock システムという、採録の議論などを共有できるような仕組みを作りたいと考えた背景です。

2007 年当時も現在も情報を共有するシステムは数多く存在していますが、どのようなシステムを採用するにしても、問題となるのは、採用したシステムをどのように運用するかという点です。例えば誰かが取りまとめてまとめるという運用ですと、そのまとめるという作業をうまく継続できなければ、そのシステムが持つ情報は古くなり、信頼性が疑われることになります。また、プロジェクトそのものの印象も悪いものとなります。私としては、そのような誰かさんが存在しなくても採録の最新の情報が集まるような持続可能なシステムを作りたいと考えておりました。

当時の採録の現場においては、すでに採録に関する遣り取りがメールで行われていたので、採録の遣り取りのメールをタイトルで分類し収集する方法を採用することが考えられました。この方法であれば、採録が行われれば大まかな分類で必ず情報が集まり、採録情報を貯めるという目的のために採録者が採録以外に余分に行う作業は少なくすむためです。

このような状況で開発され、運用された Stock システムは、うまく機能したと考えています。多くの方が協力して頂いて情報が貯まりましたし、過去の採録状況や現在進んでいる採録状況を把握することができます。情報を得られることのメリットのほかにも、stock で調べても情報が残っていなければ、その情報については基本的には残されていないという判断ができる点にもメリットがあると考えています。

現在でも stock システムが現場で採用して頂けているようで、大変ありがたく感じています。本システムが今後も NRDF の採録の品質向上に少しでも役に立てば幸いです。

以下に荷電粒子核反応データファイル年次報告 No.21(2007)に記載された原稿を再録します。

荷電粒子核反応データ採録活動における
メールアーカイブシステム“Stock”

Mail Archive System “Stock”

in Charged-Particle Reaction Data File Compiling Activities

北海道大学大学院理学研究院¹

鈴木 隆介

SUZUKI Ryusuke

Faculty of Science, Hokkaido University

1 はじめに

この報告では最近新たに開発された「電子メールアーカイブシステム“Stock”」についてその概要と使用法について説明する。

原子核反応データ研究開発センター(JCPRG) では、20 年以上に亘り日本の加速器で生産された荷電粒子核反応データのNRDF 形式による採録活動を継続し、これまでに蓄積されたデータ量は3 万レコード、90MB にのぼる^[1]。NRDF の採録の現場においては、このような成果物であるコーディングファイルの他にすべての論文に対して紙媒体の資料がファイリングされ保存されている。紙媒体の資料とは、コーディングシートや採録の際に用いられた論文のことで、採録者によるコメントやメモ等も記載されている。

他方、最近の NRDF 採録作業はインターネット上で遂行されるのが標準的となっており、採録の際に発生する疑問点・問題点や確認事項について採録者間を中心になされる議論や連絡も電子メールによって遣り取りされることが一般的である。この報告で説明する Stock システムは、前述のような紙媒体ではなく、電子媒体上で展開されている採録の作業工程、採録について採録者間を中心になされる議論、および、採録の元となったデータなどの情報を系統的に保存し、採録作業過程全般に関わる重要な了解事項や議論点を JCPRG で共有し活用に供して行くことを目的としたシステムである。

第2 章では Stock システム作成の背景と動機、第3 章では Stock システムの利用方法について説明する。第4 章では Stock システムについての設計の詳細と管理法について説明する。第5 章ではまとめを述べる。

2 Stockシステム作成の背景と動機

どのようなデータベースであれ、比較的規模の大きなデータベースを構築するためには、データ

¹平成19 年12 月、北海道大学医学研究科放射線分野へ異動。平成20 年4 月より北海道大学病院分子追跡放射線医療寄附研究部門に所属。

ベースの作成、維持、管理に多くのエネルギーや時間、労力が費やされる。NRDF データベースは絶えず発展しつつある基礎科学の原子核実験データを対象としているため、データベースの構築がさらに難しく、千葉氏の言葉を借りれば^[2]、

一度データ収集のフォーマットを決めれば、その後のデータ収集は自動的に進むというものではない。新しい実験による新しいデータが次々と生産されている。NRDF フォーマットはそのことを考慮して考え出されたとはいえ、一貫性をもってデータベースを編成していくには「どこを保存しどこを変えていくか」の的確な判断と創造性が絶えず求められる。

と説明される。事実NRDF形式が開発され 20 年以上たった現在でもNRDF形式の採録方法や採録基準についての議論が辞書作業部会や管理運営委員会等の場で活発に行われている^[3]。NRDF形式は、千葉氏が述べているように、新しい実験やデータに対応する目的や、多種多様な利用への対応のために柔軟な設計がなされており、採録方法や採録基準については必ずしも厳密な基準がはじめから与えられているわけではない。NRDFの採録方法の代表的なコーディングマニュアル^[4, 5]や、最近の採録において利用される採録エディタHENDELのマニュアル^[6]においても、細かな採録基準は掲載されていない。NRDF の採録では、採録に直接関わる現場の作業者が一番最初に論文を理解し、判断を求められる立場にある。このため、作業者レベルでの採録方法の理解が大変重要であり、作業者全体としての共通認識が必要である。

これまで JCPRG で残されてきたコーディングファイルや紙媒体の資料だけでは、これまでにどのような理解や判断により採録が行われていたかを十分知ることにはできない。過去に行われた採録の見直しや再採録の際に、現在残されている資料だけでは、なぜそのようなコーディングが行われたかが不明瞭なコーディングファイルに出会うことがある。著者は当時から既に主流となっていた、採録にまつわるさまざまな議論を電子メールで遣り取りする点に目をつけ、そのメールを系統的に収集・保存することで、放っておくと埋もれてしまうその時々における採録に関する経験や知識をJCPRGで共有し、今後への情報提供を目的とする Stock システムを開発することにした。

Stock システムは、2004 年に開発され、現在 12,000 を越える数のメールが貯められている。また、JCPRG のメンバーからのメールだけではなく、論文の著者から直接 Stock システムに送られてくるメールも増えている。このシステムはコーディングファイルなどの紙媒体の資料同様に、NRDF データファイルそのものとしては表には出て来ないが、NRDF の採録活動を影で支えるインフラの一つとなっている。

3 Stockシステムの利用

3.1 採録の現場

Stock システムでは、stock@jcprg.org のアドレスに送られてくる NRDF の採録に係るメールを対象とする。北海道大学原子核理論研究室では、stock というエイリアスが作成されているので、研究室のメールサーバを使う場合は stock 宛のメールも対象となる。メールを書く際には、タイトルに論文の D 番号(例えば“D1900”または“d1900”といった文字列、または EXFO の番号である“E1900”, “e1900”といった文字列)が入っていることが必要となるが、タイトルの中であればどこでもかまわない。もし 2 つの論文に関連する場合は、2 つの論文の D 番号が入っていることが必要

である。送られたデータは Stock システムに D 番号ごとに分類され保存される。

3.2 情報の閲覧

Stock システムに貯められたメールは Stock Viewer と呼ばれるシステムを使って閲覧することができる。Web ブラウザで <http://www.jcprg.org/stock/> にアクセスし、認証のためにユーザ名、パスワードを入力すると、図 1 のような画面が表示される。



図1: Stock Viewer 起動時の画面



図2: 論文に関連したメール一覧の表示(図1からd1801-d1900 を選択した画面)

画面構成は上から順にヘッダ部、ブックマーク部、メイン部となっている。ヘッダ部は、タイトル、stock view と呼ばれるフォーム、検索フォームから構成される。ブックマーク部は、採録に関連したページへのリンクから構成される。メイン部では、論文が D 番号で 100 ごとに分類される。メイン部の一番右側の列は、採録する論文とは直接関係のない特別な分類の一覧である。現在、ntx-wg, jcprg-sc, jcprg-ac, coder, coe, stock, gsys, nrhc, other の 9 個の分類項目がメール蓄積のために用いられている。tx-wg, jcprg-sc, jcprg-ac, coder, coe には JCPRG のメーリングリスト、stock, gsys には、それぞれ、Stock システム、グラフ数値読み取りシステム GSYS [7] についての議論内容が集められ、nrhc には、核反応データセンターネットワーク(Nuclear Reaction Data Center network) に関する情報が集められている。stock 宛のメールのタイトルにこれらの単語が入っていた場合は D 番号が記述されているメールと同様に分類される。タイトルにキーワードが入っておらず、分類ができないメールは、other に集められる。

メイン部から希望の論文が含まれている分類を選択すると、図 2 のような画面が表示される。もし、図 1 において“all” (本体部分、2 列目一番下) を選ぶと、Stock システムに貯めてあるすべての論文の一覧が表示される。

一覧からある論文(d****) をクリックするとその論文に関連したメールの一覧が図 3 のようにメイン部に表示される。ヘッダ部には、D 番号の他に論文へのリンク、採録者、チェック者、査読者の情報、進行状況等が表示される。この情報は、JCPRG が管理している表から作成されているので、常に最新の情報が表示される。ブックマーク部には、その論文に対する採録エディタ(HENDEL)、

NRDF, EXFOR のファイルへのリンクが追加される。

このメール一覧から、見たいメールを選択すると、図 4 のようにメールの本文が表示される。ヘッダ部にはメールのタイトルが表示され、ブックマーク部には現在表示されているメールの前後のメール、その論文に関するメール一覧、および、生メールへのリンクが追加される。

添付ファイルがある場合は、図 5 のようにメール本文の下に添付ファイルへのリンクが表示される。

現在では、800 近い論文に関するデータが Stock システムに保存されている。JCPRG は現在までに 1800 近くの論文を NRDF 形式に採録しているので、半分近くの論文に関する採録情報がこのシステムに保存されていることになる。



図3: メール一覧の表示(図2 からd1900 を選択した画面)

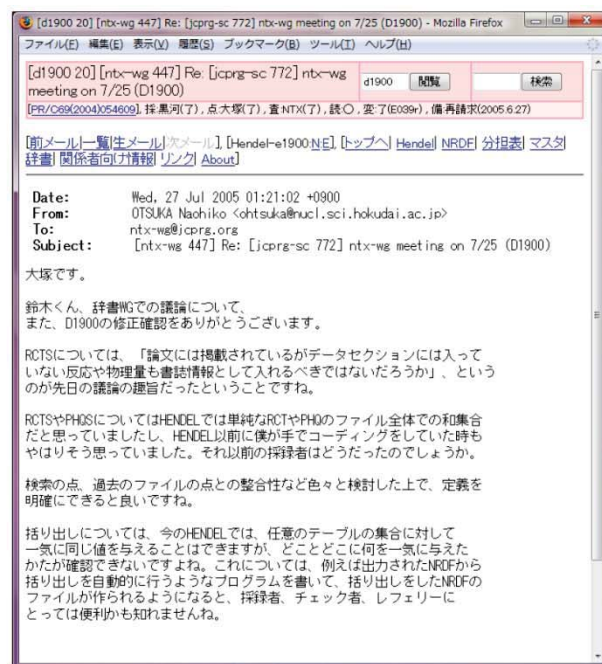


図4: メール本文の表示(図3 から[d1900 20]…を選択した画面)

3.3 stock viewおよびURL からの利用

ヘッダ部の右側にあるフォームを stock view と呼ぶ。このフォームに d1900 といった番号を入れ、**閲覧** ボタンを押すとその D 番号に対応するメール一覧が表示される。また、d1900 1 という形で入力すると、メール本文を表示できる(この例では d1900 の論文の 1 番目のメールが表示される)。

d1900_1-10 と入力すると、1 番目から 10 番目までのメールの一覧、d0001-d9999 と入力すると D 番号の一覧が表示される。

URL を直接入力してアクセスする場合は以下のように利用する。

【例1】 <http://www.jcpgr.org/stock/stview.cgi?d0001-d9999>

【例 2】 <http://www.jcpgr.org/stock/stview.cgi?d1900>

【例 3】 <http://www.jcpgr.org/stock/stview.cgi?d1900+1-10>

【例 4】 <http://www.jcpgr.org/stock/stview.cgi?d1900+1>

【例 1】では D 番号の一覧が表示される。【例 2】の例では、D1900 番の論文に関するメールの一覧が表示される。【例 3】では D1900 番の論文に関する 1 番目から 10 番目までのメール一覧が表示される。【例 4】では D1900 番の論文に関する 1 番目のメールの本文を見ることができる。



図

5: 添付ファイルの取扱いについて

図6: 全文検索システムNAMAZU による検索結果

3.4 情報の検索

Stockシステムでは貯められたメールの情報を検索することが可能である。Stockシステムでは NAMAZU (および、perl 版のpnamazuz) ^[8] と呼ばれる検索システムを採用している。図 6 が検索結果である。検索対象については、本文だけではなく、一度閲覧したあとの添付ファイルであれば、pdfの内容についても検索が可能である(次節 4.1 のtmpディレクトリ配下のファイルの展開を参照)。

4 Stockシステムの詳細と管理法

4.1 Stock システムの構成

Stock システムはサーバに届いたメールを振り分ける 150 行程度の perl スクリプト(stock.pl) および、システムに貯められたメールを閲覧する際に用いる Stock Viewer から構成される。Stock システムの概略図を図 7 に示す。

Stock Viewer は stview.cgi と呼ばれる perl スクリプトおよび system サブディレクトリ配下に置かれる設定ファイルから構成される。このプログラムの実体は 1,500 行程度の perl で書かれた CGI である。

メールが stock@jcprg.org に送られると、図 7 のように stock.pl スクリプトが、メールの Subject からどのディレクトリにメールを振り分けるかを判断し、そのディレクトリにメールを一つのファイルとして保存する。図 7 の例で言うともメールの件名に d1900 という文字列が入っているので、

stock.pl は、d1900 ディレクトリ配下にメールを保存する。d1900 ディレクトリにはすでに 20 通のメールが保存されているので、このメールは 21 という名前で保存される。

D 番号のつかない特別なサブディレクトリも同様に扱われる。

また、tmp 配下のファイルは、メールの添付ファイルの展開用に作成されるものである。検索時には、このディレクトリ配下のファイルを添付ファイルとして検索する。

逆にブラウザからの要求に対しては、stview.cgi (Stock Viewer の実体) が www サーバから呼び出され、要求に対応する。

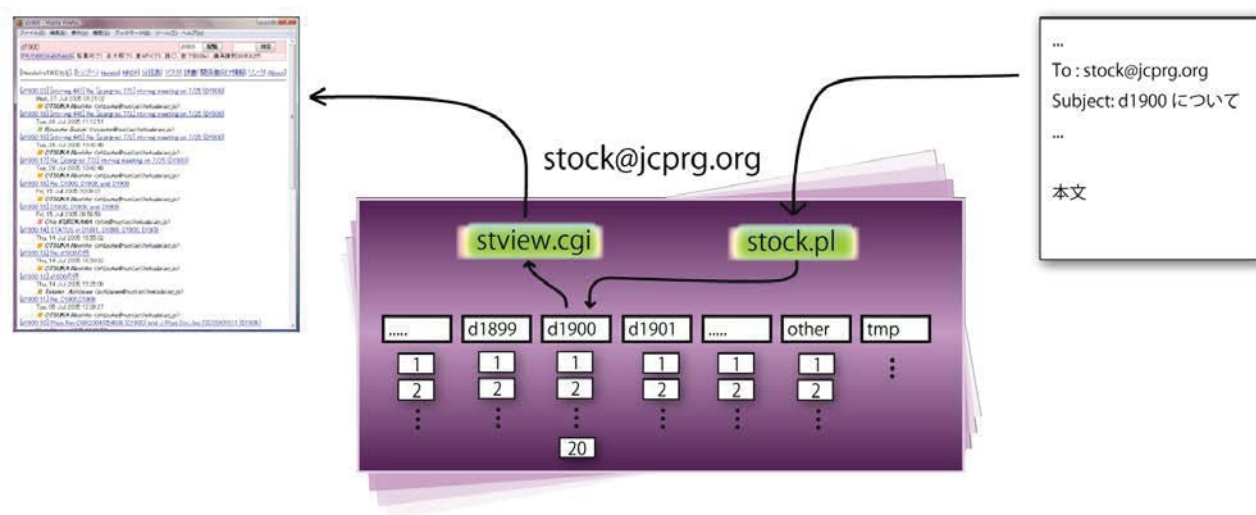


図7: Stockシステムの概略図

4.2 Stockシステムの管理

4.2.1 設定の変更

システムの設定は system サブフォルダ配下のファイルにより設定される。

メール送信者の色を変更

図 3 で表示されているメール送信者の色については、color user ファイルを書き換えることで変更ができる。

特別な分類の設定

Stock システムでは D 番号の他に特別な分類を作成することができる。Stock Viewer では、このような情報を list ファイルで扱う。ただし、タイトルから適切にメールが振り分けられるように stock.pl を設定する必要がある。

ブックマークの設定

ブックマーク部に表示されるリンク先は bookmark ファイルにより設定される。

メールに注釈を付ける

stview では、図 5(メール本文中のタイトル以下に表示) に示されるように、メールに注釈を付けることができる。comment ファイルにてコメントの追加、削除が可能である。

リンク

この報告では説明をしていないが、Stock Viewer では、ブックマーク部とは別に、採録に関連するリンク一覧のページがある。このリンク一覧は link ファイルにて設定される。

4.2.2 その他

Stock システムでは、検索システムとして、NAMAZU を採用している。NAMAZU を利用するには、検索用のインデックスファイルの作成が必要であり、現在では、一日一回 cron を用いて、インデックスファイルを更新している。

また、メールは MH 形式(1 メール 1 ファイル) で貯められているので、もしメールの分類が間違っていたり、必要のないファイルがあった場合には、MH のコマンドを使ってメールの管理を行うことができる。

5 おわりに

前章までで Stock システムの利用についての簡単な説明を終える。Stock システムの作成当時、著者は採録者として活動していたが、採録方法の情報や自分の担当する論文以外についての採録過程や議論などの情報を得ることが大変難しいと感じ、採録の議論などを共有できるような仕組みを作りたいと考えるようになった。情報収集システムを採用するプロジェクトは世の中に数多くあるが、情報が更新されなくなったシステムをかかえたプロジェクトはアクティビティの低下という負のイメージを持たれる危険性がある。Stock システムにおいては、採録作業全般に関する重要な了解事項や議論点、問題点等の情報(「採録情報」と呼ぼう)をあらためて新規に生成するのではなく、現在も継続的に進行している採録者及び採録者間で展開されている採録活動の実態そのものを、採録情報としてそのままアーカイブ化することによって、持続可能なシステムとして存立することを目指した。JCPRG からの指示はなかったと思うが、当時から採録の現場においては遣り取りがメールで行われていたので、当時行われていた採録の遣り取りのメールをタイトルで分類し収集する方法を考えた。この方法であれば、採録が行われれば大まかな分類で必ず情報が集まり、採録情報を貯めるという目的のために採録者が採録以外に余分に行う作業は少なくすむ。Stock システムの導入後、採録者にとって増えた作業はタイトルに D 番号を入れるということと、宛先に stock と書くことだけであった。著者は 2005 年後半ぐらいから、採録の直接の現場から離れることになったが、現在でも採録に関わるデータを集めることに成功している。集められたメールの数を単純に平日の日数で平均すると、1 日あたり約 10 通のメールを誰かがどこかで作成していることになり、これは JCPRG のアクティビティの高さを表していると考えられる。

現在では、大変ありがたいことにメールを書く際に stock 宛に送ることが採録者に受け入れられているように感じられるが、採録者には Stock システムにメールを残すことは意識せず、採録に集中していただくことを期待している。他方、採録関係者の参照に供したい、或は、何か残したいと思われるものがあれば、採録に関わるものは何でもよいのでとりあえずは stock 宛に送って頂きたい。単なる報告のメールでも採録の検証の際に助けとなると考えている。情報の価値は時代とともに変わるので、残せる情報は少しでも残しておけば将来の助けとなる可能性がある。NRDF の採録の品質向上に Stock システムが少しでも役に立てば幸いである。

謝辞

Stock システムを使っていたいただいた皆様に感謝しています。また、“stock”の名付け親であり、初代 stock.pl 実装者の合川正幸さんに感謝します。当時は海のものとも山のものともわからない stock のアイディアを実装してくださいました。

Stock システムを褒めてくださった方々に感謝します。特に辞書作業部会で能登先生が「おもしろい」と褒めてくださったこと、メール一覧で送信者に対し色を導入した際、黒河さんが褒めてくださったことはとてもよく覚えています。

最後に、このシステムに密接に関わって頂くことになった大塚直彦さん、吉田ひとみさんに感謝します。stock 宛のすべてのメール(12,000 以上ものメール!) は、JCPRG のサーバに送られると同時に彼らにも送られていました。現在でも Stock システムが使われているのは彼らのおかげです。

参考文献

- [1] 吉田ひとみ, 加藤幾芳「資料: 2006 年度入力データ」(荷電粒子核反応データファイル年次報告 2006 年第 20 巻[1996 年 3 月]) p.108.
- [2] 千葉 正喜「はじめに-若い力に期待-」(荷電粒子核反応データファイル年次報告 2002 年第 16 巻[2003 年 3 月]) p.1.
- [3] 管理運営委員会の議事録は、各年度の荷電粒子核反応データファイル年次報告参照。辞書作業部会の内容については、荷電粒子核反応データファイル年次報告 1999 年第 13 巻以降を参照。
- [4] 手塚洋一「NRDF コーディングマニュアル(ver.1.1)」(荷電粒子核反応データファイル年次報告 1995 年第 9 巻[1996 年 3 月]) p.71.
- [5] 向井重雄, 吉田ひとみ「コーディングに対する補足と注意(1995 年改訂版)」(荷電粒子核反応データファイル年次報告 1995 年第 9 巻[1996 年 3 月]) p.86.
- [6] 大塚直彦「ウェヴエディタ”HENDEL”を用いた核データ採録入門」(荷電粒子核反応データファイル年次報告 2001 年第 15 巻[2002 年 3 月]) p.12.
- [7] 新井好司、蓑口あゆみ、大塚直彦、内藤謙一「GSYS: グラフ数値化システムの開発とその利用法」(荷電粒子核反応データファイル年次報告 2004 年第 18 号[2005 年 3 月]) p. 78.伊藤慎他、鈴木隆介「グラフ数値読み取りシステム(Gsys2.2) 利用の手引き」(荷電粒子核反応データファイル年次報告 2006 年第 20 巻[2007 年 3 月]) p. 39.
- [8] 全文検索システム Namazu (<http://www.namazu.org/>)
全文検索システム Namazu の perl 版検索プログラム pnamazu
(<http://www01.tcpip.or.jp/~furukawa/pnamazu/>)

5-6 IntelligentPad の応用

5-6-1 インテリジェントパッドを応用した核反応データベース検索システムの開発

千葉 正喜（元札幌学院大学）

札幌学院大学に赴任して5年ほど経過した平成7年度に、北海道大学工学部田中譲教授の研究室に4月から1年間の研修の機会を頂いた。札幌学院大学の教員には、在職5年を経過すると1年間の研修制度があって、この制度により田中一社会情報学部長から学部としての最初の研修に出ることが許された。研修のテーマとして、「研究者データベースにおけるヒューマンインターフェースの研究」を設定したが、研修・研究の具体的内容は荷電粒子核反応データベース NRDF（Nuclear Reaction Data File）のデータを、（1）リレーショナルモデルに基づく標準的なデータベース管理システムで管理し、（2）IntelligentPad を応用した研究環境に適合したユーザインターフェースを開発することであった。

NRDFでは、データ言語と呼ばれる独自の規則（モデル）を用いて^[1]、荷電粒子核反応データを収集蓄積（コンパイル）している。この言語の基本的な考え方は、記述対象のオブジェクト、すなわち一遍の文献をいくつかの属性集合でとらえて、これらの属性とその値（一つまたは複数、または値の列）の対の集合としてオブジェクトを表現している。ここで重要な点は、どのような属性があるかがあらかじめすべて決まっているのではないということである。属性の値だけでなく、属性自身も時間的に追加や更新があってよいという点にあることである。このモデルでは、収録する文献の1件分のデータはデータストリームと呼ばれる。このデータストリームには、一つまたは複数の実験・測定データの数値テーブルが含まれる。

実験・測定データに論理的な単位を考えると、それは一つの数値テーブルとその数値の意味内容を特定する記述情報（属性集合とその各属性の値の対の集合）から構成される。NRDFでは、この実験・測定データの論理的単位はデータセットと呼ばれる。したがって、データストリームはデータセットの集合でもある。そして、複数のデータセット間では同じ記述を共有する部分が生じるが、その共通する記述はデータストリームを3種類のセクション（書誌情報「BIB」、実験条件「EXP」、データ「DATA」）を区別して記述し、それぞれのセクションのデータセット識別子のリストを付加して、データストリームにおける冗長した記述を排除するようになっている。

このような独自の構造を持つデータであったことから、その当時はデータベース管理システムも独自に開発されて^[2]、大型計算機センターのメインフレームコンピュータ上で検索利用に供されていた。

この間、蓄積されたデータ量の増加、書誌情報の索引に基づく典型的なデータテーブルの検索だけでなく、元素生成過程の研究への利用にみられるような、蓄積されたデータの分析やデータ相互

間の関連など多様なデータベースに対す要求が増大していた。

NRDF に対するこのような要求の実現、ユーザサービスの機能拡張には、独自開発のデータベース管理システム自身を改定し、新たな機能追加や保守を継続していくことは、とりわけ研究環境では困難であると考えられた。そこで、NRDF データベースの管理に一般的になっていたリレーショナルデータベース管理システムを利用するとともに、マルチメディア技術、オブジェクト指向モデリングなどのその当時の開発技術や最新のハードウェア・ソフトウェア技術を取り入れた新しい概念に基づく NRDF のユーザサービスの構築を目指した。

リレーショナルデータベースのデータ構造はリレーションと呼ばれるテーブルの集合になっているが、この構造をリレーショナルスキーマが定義している。

NRDF データをリレーショナルデータベースのテーブルにすると、Author や Title などの情報識別子をそのままテーブルの属性とすることができない。なぜなら格納するデータによって使われる識別子が固定していない、時間的にも変わりうるので、あらかじめ属性を定義できないし、既知の属性をすべて並べたとすると複数の属性の値が空 (Null) であるタプルをつくることになる。そこで、情報識別子自身も属性とする属性値対集合モデルで NRDF データをリレーションにマッピングすることにしたのである。記述対象のデータオブジェクト (収録文献に対応するデータ) にオブジェクト識別子を割り当てる。このようにして、同じオブジェクト識別子をもつすべての属性記号と値の組の集合で 1 件分のデータストリームを記述できるようにした。

このように表現した NRDF データベースの新しいユーザインターフェースは、IntelligentPad 技術を利用して開発した。IntelligentPad^[3]は、北海道大学工学部の田中譲教授の研究室で研究開発された新しいオブジェクト指向メディアシステムである。この IntelligentPad を用いて NRDF のユーザインターフェースを開発することで、多様な利用や機能の拡張に応えることを期待したのであった。

IntelligentPad システムでは、計算機上で取り扱うすべてのオブジェクトは用紙のイメージを持ったパッドと呼ばれるメディアで統一的に扱われる。数値や文字列、グラフなどのデータだけでなくアプリケーションプログラムもパッドで表現され、マウス操作で機能を発揮させる、個々の機能を持ったパッド (プリミティブパッドという) を貼り合わせて複雑な機能を合成すること、色や大きさ、配置などのレイアウトが自由に選べるなどの機能を持っていた。

プリミティブパッドの構造は MVC プログラミングアーキテクチャーに基づいており、新しい機能を持ったパッドを定義するには Model 部と View 部をプログラミングすればよく、Controller 部は標準化されている。パッドと他のパッドとはスロット結合と呼ばれる標準化された方法で結合され、子パッドと台紙となる親パッド間でデータ交換が行われパッドの状態を変えている。

このような機能を持つ IntelligentPad を用いて開発した NRDF の新しいユーザインターフェースは、Smalltalk 版 IntelligentPad にあらかじめ用意されていた汎用部品のパッドとこのシステムのために新たに開発したパッドで構成された。汎用データベース管理システムには NTT データの UniSQL を利用していた。

IntelligentPad を用いることで、新たに必要になる機能のみを開発し、すでに開発されている機能は標準のスロット結合で連携させて、そのまま再利用することができる。

その後 IntelligentPad は Windows 版が製品として作成され、NRDF のユーザインターフェースは

CONTIP^[4]としてさらにリメイク改良された。

最近、インターネット版IntelligentPadとでもいうべきWeble World^[5]が利用できる。これを利用した核反応データベースシステムを開発するならば、この分野の関係者が国際的にデータベース利用とツールの開発で資源の共有や再利用の場を提供することになるのではと期待している。

- [1] 荷電粒子核反応データファイル開発報告書（文部省科研費特定研究「情報システムの形成過程と学術情報の組織化」）核データグループ研究代表者 田中一、1979年3月
- [2] 核データファイル NRDF の開発（文部省科研費特定研究「広域大量情報の高次処理」）研究代表者 田中一、1976年3月
- [3] 田中譲「ミームメディア・アーキテクチャ IntelligentPad とその応用」、情報処理 38巻3号(1997) 222-231；長崎祥、田中譲「シンセティック・メディアシステム IntelligentPad」、コンピュータソフトウェア vol.11 no.1（1994）36-48
- [4] 升井洋志、大林由英「CONTIP(experimental system) User Manual」NRDF 年次報告 98（1999）56-74
- [5] 田中譲研究室「Web 上の知識メディアシステム Webble World システム」
<http://cow.meme.hokudai.ac.jp/WebbleWorldPortal/>

5-6-2 Webble World を用いた核データの利用

江幡 修一郎（北海道大学）

この項では Webble World を用いた核データの利用環境の整備を今後も推進する為に、その期待する利用価値と環境開発に必要な基本事項、40 年史編纂時に生じている問題点、今後の方向性について記している。40 年史に Webble World の様な技術的な内容が加わる事は年史編纂の趣旨を示している。センターの在り方や、技術継承の可否は、当時の状況に合わせて取捨されるべきで、過去の具体的な経験が選択肢を絞り、合理的なプロジェクトの推進を促すものだと思えるからである。また、非専門家が少ない障壁でアプリケーション開発を進められる様に、平易な言葉と説明を加えて記してある。

Webble World とは、北海道大学知識メディア・ラボラトリーが開発を続けてきた IntelligentPad の後継にあたる、アプリケーション開発ソフトウェアである。前項迄の記述にある様に、IntelligentPad では機能を Pad という単位の「部品」として扱い、Pad の組み合わせによって、新しい機能を構成し多様なアプリケーションを作成する事が可能なソフトウェアである。Webble World もこの知識メディアの発達過程の思想を踏襲しており、多くのユーザーに利用される事で自己拡大するソフトウェアとなっている。Webble World では Pad に相当するものを Webble と呼ぶ。

IntelligentPad では専用の環境（プラットフォーム）において Pad の編集や構成を行う。基本 Pad から合成した新しい Pad はユーザー端末に蓄積される。グループ内のネットワークを通じ Pad の複製、合成を繰り返し目的のアプリケーションに至る。Webble World と IntelligentPad の大きな違いは開発環境にある。Webble World は知識メディア・ラボラトリーの Web サーバー上にある Web アプリケーションであり、Webble 構成環境そのものである。ユーザー端末で行っていた、機能の合成を Web サーバー上で出来る為、蓄積される新しい機能(Webble)を Webble World を通して、多様なユーザーが端末環境に依存せず、共有、利用出来るものになった。

Webble World の利用環境は核反応データ利用者の多様性に対処し得るものである。日本の核データ利用者の主な目的は、学術的な利用が当初想定されていた。近年においては原子力関連の研究開発、重イオンビームの医療利用、中性子ビームによる材料イメージングなど、核反応データを積極的に社会利用する分野の拡大が進んでいる。例に挙げた分野に限ったとしても利用するデータの核種、反応、エネルギー環境、書式が異なっている。ユーザーの閲覧できるデータは同じであるが、出力に於いて利用目的の差が現われてくる。核データセンターの役割の一つに、ユーザーを考慮したデータ利用環境の提供が挙げられるが、多様化する分野の利用目的、加えて個人ユーザーの目的の一つ一つ対処する事は困難である。そこで、個々のユーザーに対応した出力装置をユーザー自身によって容易に調整出来る環境を提供出来れば、核データ利用環境は飛躍的に改善されると考えられる。Webble World 導入はこの利用環境の提供に全く適している。

まずは Webble World を核データ利用に応用する為の具体的な開発環境を述べる。Webble World は <http://cow.meme.hokudai.ac.jp/WebbleWorldPortal/> で利用する事が出来る(図 1)。現在は HTML5 を基盤とした開発(Webble World3.0)が進められている。但し、利用出来る機能は限定的である(2014/03/24 現在)。この項ではこれまでの Webble World(2.0)について述べる。

Webble WorldはMicrosoft社のSilverlightと呼ばれるWebアプリケーション開発フレームワークで記述される。SilverlightはWebブラウザ用のプラグインである。Webble Worldを利用する為には少なくともSilverlightのインストールが必要である。基本機能のWebbleはWebble Worldに幾つか用意されているが、十分に目的を満足するものではないので新しいWebbleの開発が必要になる。我々はWebble開発の環境にMicrosoft社のVisual Studioを利用した。この環境においてWebble開発支援のWebble



図 1 ブラウザに表示された Webble World

Modelパッケージが使用出来た事は採用理由の一つである。視覚操作性(GUI)アプリケーションの開発において、視覚オブジェクト(ウィンドウやボタンなど)を定義する部分(View)と入力された数値や真値から何らかの演算を行う部分(Model, Control)がある。Webble ModelではViewとControl部分をXAML (Extensible Application Markup Language)で、Model部分をC#でプログラミングしている。当時、核データの拡張と柔軟性の為、新しい書式を議論する中で、XML (Extensible Markup Language)で構成する案^[1]があったが、これはWebble開発を進めていた事が一つの要因だったと思われる。XAMLはViewだけ

でなくXMLで記述されるものをオブジェクトとして扱い、XML間のControlも記述するマークアップ言語の一つである。テキストBoxオブジェクト等に入力された値を利用して計算するModel部分はC#で別に書かれる。

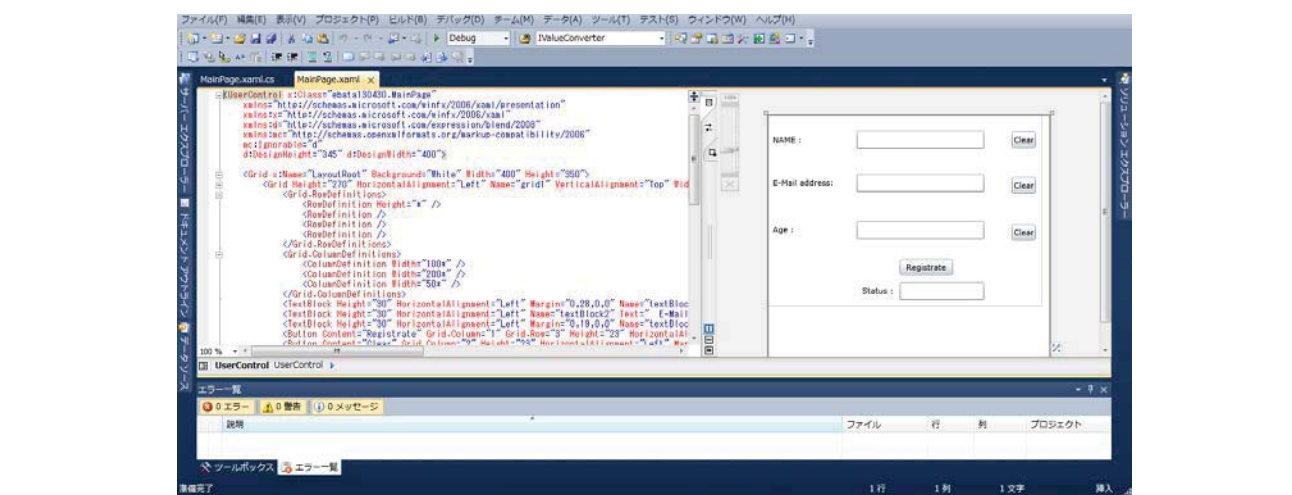


図 2 Visual studio 2010 の作業ウィンドウ XAML 部分が開かれている。左がソースパートで右がヴィジュアル再現パート

40 年史編纂時に問題、障害になっている点について述べる。先ずこれまでの核データ利用環境の開発について触れておく。IntelligentPad を利用した核データ利用環境の開発は少なくとも 1995 年から千葉氏によってその試作が進められていた^[2]。その後開発は、メンテナンスや維持、開発者の不足等の諸事情により、停滞してしまった。これを打破する為にIntelligentPadによるアプリケーションを外注する試みもあった様であるが、予算的な問題が生じ断念する事になった。2011 年度まで

この種の利用環境開発の年次報告はない。これ等の問題は開発環境が特殊であった事が最大の問題点であったと思われる。開発に特殊なスキルが必要であり、メンテナンスと維持する技術者の育成に労力が必要だった為である。この問題は現在に於いても同様に存在する。WebbleWorldの開発環境の基本はSilverlightである。比較的この環境は一般的に利用されるものであったが、プログラミングの知識の無い者にとっては、敷居の高いものであろう。2009年度から松本氏と椿原氏を中心に進められた核データフォーマットのXML化[1,3]はWebbleWorld開発が念頭にあったものと考えられるが、具体的な開発は2012年度の大木氏によるものが始まりである^[4]。大木氏の報告にある様に、Silverlightアプリケーションは基本的に外部サーバーにアクセスする事が出来ない。この対処法としてMicrosoftが提供しているWindows Communication Foundationサービスの導入が提案された。但し、このサービスの実装には十分な技術が必要な為、実用には至らなかった。この間XMLフォーマットの策定が進んでいた。

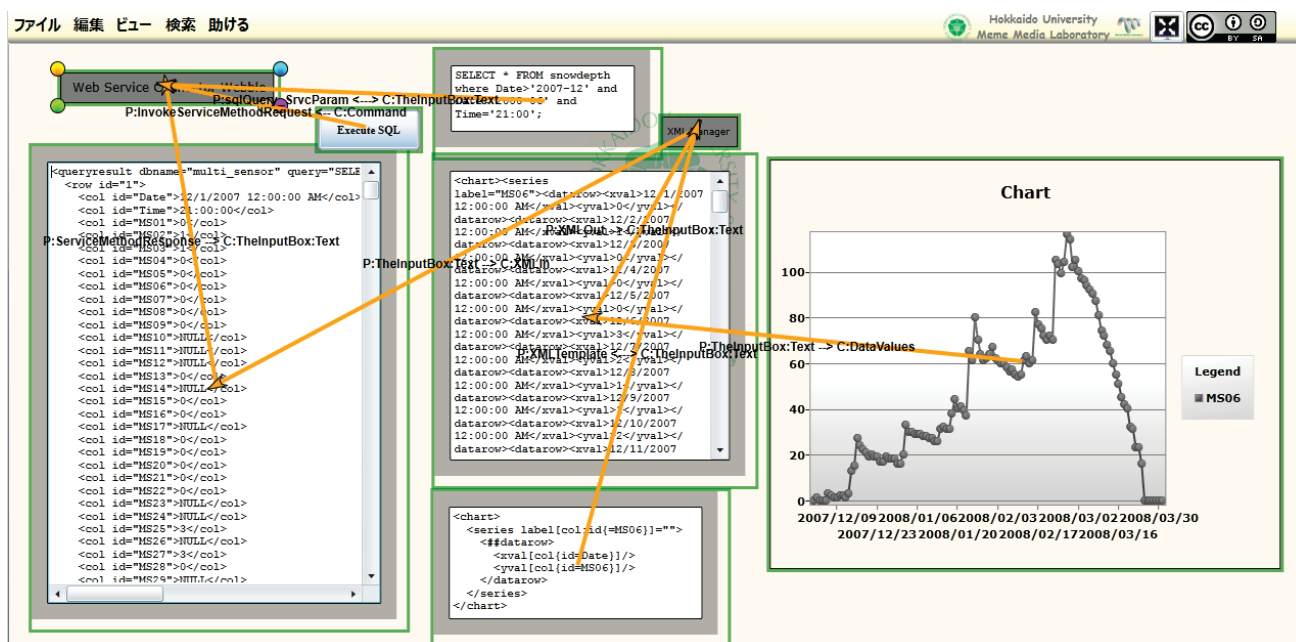


図3 データベースからデータを読み出し、自由フォーマット定義と変換を行う。変換したデータのテキスト書き出しとグラフ表示を行う。矢印はWebble間の親子関係を示す。

2013年度に著者がこの開発を担当する事になったが、同じ問題に対する事となった。これまでの開発過程と問題点の洗い出しの中でWebbleWorld上に有用なWebbleを発見した。図3はそのWebbleの内部構造を示している。この複合Webbleは8つのWebbleで構成されており、実行出来る事はデータベースにアクセスし、データの表示、任意フォーマットを定義し、その定義に従ってデータを変換、テキスト表示とグラフ表示までが行える。データベースへのアクセスはWebbleWorldサーバー上に限られていたが、我々の想定している基本機能をすべて備えている。著者はこのWebbleを基礎として開発技術の習得を図っていた。Webbleアプリケーションの開発に於いてもう一つ重要な問題はマニュアルである。操作と簡易開発のマニュアル^[5]は存在するが、年代が異なる為か、実際の機能とは異なる仕様も書かれている。特にSilverlightの知識を持たない事を想定しているので初学者にとっては非常に難しい。そこで著者は開発技術の習得と同時により平易なマニュアルの作成も行っていた。具体的に理解しにくい部分はWebble間をつなぐスロットの設定である。Webbleにはそれぞれ数値変数や論理変数、テキストを持っている。スロットを通じてやり取りを行い、入出力の関係(親子関係)を設定する。既存のマニュアルではWebbleの概観に関する変数についてスロットが付い

ているが、より高次の変数値の授受についてはスロットの詳細な設定が必要不可欠である。しかしながらWebbleWorld上での編集でその設定は容易ではない。この点はWebbleのソースファイルをユーザー端末にて編集する必要があると思われる。以上の問題点をまとめると、技術習得の過程が確立していない。マニュアルを読む為にSilverlightの知識が重要。データベースへのアクセスを可能にする技術はまた新たに習得が必要である事である。

今後の方向性について述べる。このプロジェクトの趣旨は継承していくべきものと著者は考えている。しかしながら、我々の現状の技術が必要なものまで到達する為には大きな労力を割かねばならない為、何らかの革新的な技術発展が無ければ、飛躍的な進展は困難である。この点については採用されている開発環境にも問題があると言わざるを得ない。2014年度現在 Silverlight を用いている Web アプリケーションは殆ど見る事が無い。それは技術習得に労力を割けない理由と繋がる。Web アプリケーション開発のスキルアップがスタッフの今後のキャリアに貢献していくのであれば、労力を割く理由になるが、今後衰退する可能性の高い技術を新たに身につけようという者はいない。開発環境は広く汎用的に利用されているものを採用すべきであり、開発者のキャリア形成も考慮に入れるべきである。この点については 2014 年度に入って改善されたと言ってよいだろう。知識メディア・ラボラトリーは開発環境を大きく変更し HTML5 を採用した。この技術は既に広く利用されており、発展性の高い技術である。まだマニュアルなどが発展途上ではあるが、WebbleWorld3.0 は再考すべきものであろう。また 40 年史編纂時において新しい採録用エディタの開発は Java で行われている。今後の Webble 開発に於いて、技術習得に良い環境である。HTML5 への移行は Web 技術開発において目下行われており、JavaScript (Java とは異なるが文法が似ている) のスキルが必要になる。

まとめると、WebbleWorld による核データ利用の環境開発は、核データベース研究開発センターの趣旨と合致しており、踏襲すべきである。これまでの障害となってきた Silverlight の開発環境は撤廃された為、今後の WebbleWorld の開発は HTML5 を念頭に進められていくべきである。その時必要になるスキルは JavaScript である。多様なユーザーに応えるべくアプリケーションを開発する為には、ユーザーの要望を集める事と周知する事が非常に重要である。アプリケーションの開発目的を明確化し、共有する事で複数の開発者で発展させる事が出来よう。また開発者のキャリアパスにおける開発の労力が見合うものであるか考慮する必要がある。具体的に進める事は、現在並行して行われている Java による新しい採録用エディタ開発は注視すべきである。このプロジェクトにおける、開発環境とマニュアルを汎用性の高いものとして残していく事が持続可能な開発に必須である。

参考文献

- [1] 松本 琢磨, 千葉 正喜, 加藤 幾芳「NRDF の XML 形式への変換(Transformation of NRDF to an XML Format)」, 荷電粒子核反応データファイル年次報告 No.23 p.2-9 (2009)
- [2] 千葉 正喜「IntelligentPadを用いた核反応データベース利用環境の試作(A newly designed platform for the NRDF with IntelligentPad)」, 荷電粒子核反応データファイル年次報告 No.9 p.2-56 (1995), 「IntelligentPadを用いた核反応データとその取り扱いツールの流通(Dissemination of Nuclear Reaction Data and Handling Tools for the NRDF by IntelligentPad)」, 荷電粒子核反応データファイル年次報告 No.10 p.10-17 (1996)
- [3] 椿原 康介, 松本 琢磨, 合川 正幸, 加藤 幾芳「XML ベースの新フォーマットの提案

(Suggestion of New Nuclear Reaction Database Format based on XML)」, 北海道大学原子核反応データベース研究開発センター年次報告 No.1 p.23-26 (2011)

- [4] 大木 平, 椿原 康介, 合川 正幸, 加藤 幾芳「Webble World を用いた新たな核データベース利用システムに向けて(Toward a new utilization system of nuclear databases with the Webble World)」, 北海道大学原子核反応データベース研究開発センター年次報告 No.2 p.23-31 (2012)

- [5] <http://cow.meme.hokudai.ac.jp/docs/WebbleWorldManual.pdf>

