電子学術書利用実験プロジェクトでの技術的成果と課題

入江（いりえ）　伸（しん）
（メディアセンター本部電子情報環境担当課長）

1　はじめに

本稿では，電子学術書利用実験プロジェクト（以下，本プロジェクト）における技術的な成果と課題について報告する。

本プロジェクトを通じてプラットホームとコンテンツを実際に開発しながら技術的な課題解決を模索してきたことにより，一般的な標準化や手法の確立というよりも，現実的な落としどころが見えてきたと感じている。取り組んできた技術的な課題は，スキャン仕様，テキスト化技術（OCR），電子書籍フォーマット，プラットホーム実装手法など多岐に渡るが，本稿ではまず，プロジェクトの基本設計，フローとその課題を整理した上で，主要な課題として電子書籍のスキャン仕様を中心に報告し，プラットホーム，ビュアーの実装における諸問題とコンテンツの流通という側面についても触れる。紙面の都合上，詳細なまとめについては，別のプロジェクト成果報告の中で行なっていきたい。

2　本プロジェクトの基本設計

2.1　システム設計

本プロジェクトを進めるにあたり，実験のための機能要件を定義し，そのための電子書籍フォーマットを設計した。この実験において，特に意識的に盛り込んだシステム機能を以下に示す。

① DRM（Digital Rights Management）機能

大学図書館での電子書籍貸出サービスのためのDRM 機能

・学内認証システム（keio.jp）を使った認証機能

・電子書籍ダウンロード型での貸出機能

②本プロジェクト独自の検索機能

・ダウンロードした電子書籍の柔軟な全文検索とヒット文字のハイライト

・サーバー側での全文検索と検索結果の「立ち読み」機能

③マルチデバイス

・iPad，Android，PC 上での動作保証

④学習支援機能

・教材としての配信，SNS との連携

2.2　電子書籍フォーマット

電子書籍フォーマットについては，AZW，.book，XMDF など多くのフォーマットが発表される一方で，HTML5, ePub 等の標準化に向けた議論も急速に進んでいる。本プロジェクトで採用した電子書籍フォーマットは，新しい標準フォーマットを提案するためではなく，汎用性のあるプラットホームを実現するために，HTML5＋XML 技術をベースにePub を参考にして設計したものである。また，本プロジェクトが対象とする書籍は比較的古い書籍が多いため，版面スキャン画像を前提とした電子書籍フォーマットに限定することで，電子化やシステム実装を容易にした。更に，本プロジェクト独自の検索機能の実現に当たっては，ヒットしたテキストの高輝度表示を可能にするべく，検索のためのテキストと文字座標を実装するための XML スキーマを設計した。版面画像に対し OCR 処理を行い，中間フォーマットとして透明テキスト付き PDF を作成した。そこから版面画像と版面座標付きの検索用のテキストを抽出し，開発した XML フォーマットを作成し，画像と XML をあわせて ZIP 圧縮を行い本プロジェクト用の電子書籍フォーマットとしている。このフォーマットは，ePub を意識して開発した経緯もあり，ePub へのコンバートが可能となっている。

3　デジタル化のフローと課題の整理

前項で説明した基本設計をもとに，図 1 の電子化フローを作成した。個々の行程の概要を説明する。

a）スキャン

紙資料をスキャンして OCR 処理を行い透明テキスト化する。次の c 工程へは透明テキスト付き PDF が渡される。この a 行程の最後で，目次からのペー

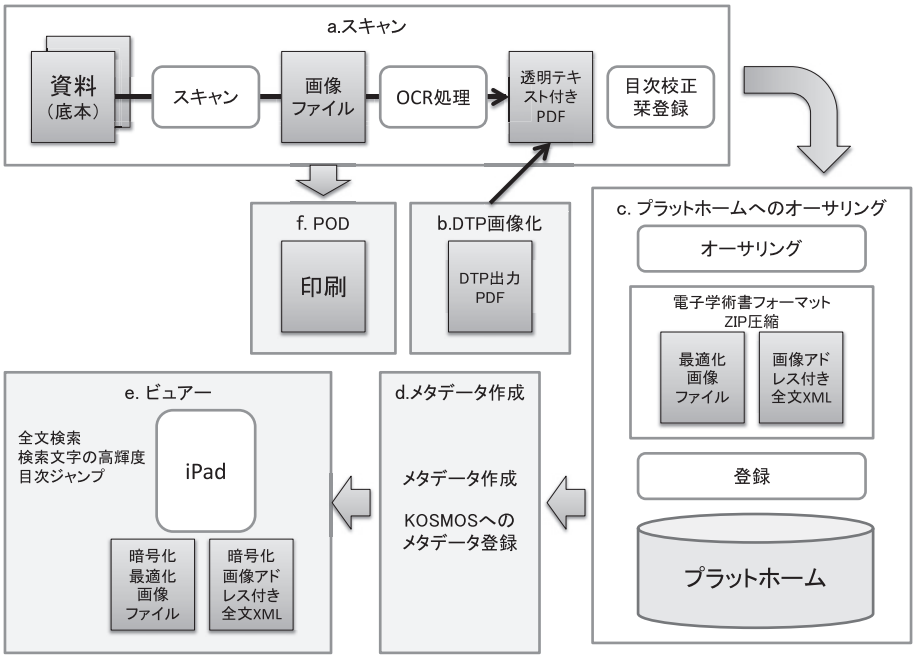

図 1.　デジタル化フロー

ジジャンプの精度向上のために，目次情報の校正とPDF の栞登録を行う。

課題：スキャン仕様，OCR 精度，電子化コスト，空白ページの扱い

b）DTP 版面の PDF 画像化

紙資料からでなく，Adobe InDesign のようなDTP ソフトの版面（協力出版社から提供されるもの）を用いる場合，その版面を画像化し PDF の表面に貼り付ける処理が必要となる。これによって，a の紙資料の透明テキスト付き PDF と同じ処理で，プラットホームへのオーサリングが可能となる。

課題：画像化仕様

c）プラットホームへのオーサリング

透明テキスト付き PDF をプロジェクト用の電子書籍フォーマット（開発担当の協力会社である京セラコミュニケーションシステム株式会社にちなんでKCCS フォーマットと呼ぶことにする）へオーサリングする。目次からのページジャンプのための目次データの XML 作成もこの段階で行なっている。KCCS フォーマットでは，最適化画像と画像アドレス付き全文テキスト XML ファイルを ZIP 形式の圧縮ファイルにしている。

課題：貸出のためのダウンロード用書籍と PC での表示用データの画像仕様，ePub 等の標準との互換性，目次ページ（論理ページ）と実ページの違い，PDF 透明テキスト抽出と XML 作成

d）メタデータ作成

プラットホームに搭載された電子学術書のメタデータを図書館蔵書検索システム（以下，KOSMOS）へ登録する。

課題：電子学術書の URI，分割配信の可能性

e）ビュアー

電子書籍をビュアーとなるデバイスにダウンロードして利用するため，Wi-Fi でのダウンロード時間が課題となった（詳細は後述）。また，ダウンロードの際，DRM 機能により書籍ファイルを暗号化し，ファイルに対して貸出期限を設定している。貸出期限を過ぎるとダウンロードした電子書籍は削除されるようになっている。加えて，学術書という特性から学習支援機能を開発し，メモや付箋，SNS 連携を実装している。

課題：システム実装手法，ダウンロード時間，学習機能（メモ・手書きメモ・付箋），SNS 連携

f）プリントオンデマンド（POD）

プリントオンデマンドのために 400dpi 超の PDF ファイルを別途作成する。

課題：分割印刷の可能性，価格設定，モノクロ画像の画質

4　書籍のスキャン仕様

前項で上げた課題のうち，ここではスキャンに関する技術面の概要を説明する。スキャン仕様について説明する前に，まずは書籍電子化の種類について簡単に説明する。

4.1　書籍電子化の種類

書籍の電子データには，一般的に①底本からスキャンした画像データ，②印刷を前提としたページ概念のある DTP データ，③リフロー型での利用を想定とした，ePub 等の電子書籍フォーマットへ変換するための DTP データの 3 種類がある。②と③は同じ DTP データであると思われているが，両者の違いは極めて大きい。②のデータがあれば③へのコンバートが容易になるわけではない。本プロジェクトでは，版面という概念を持つ①と②を対象として扱ってきているが，本稿では，主に①の「底本からスキャンした画像データ」について技術的な問題について報告する。

4.2　スキャン画像の仕様を決める要因

本プロジェクトでは，書籍の裁断が可能か否かによってスキャナーかデジタルカメラかを使い分けている。本稿では，裁断可能でスキャナーを使用した場合に絞って説明する。

ここで強調しておきたいのは，マスデジタイゼーションでは，目的を明確にして目標品質・コスト・作業フローを設計する必要があるということである。このため，仕様の設定は目的を明確にするところから始めなければならない。

本プロジェクトでのスキャン画像の利用目的は，

・OCR によるテキスト抽出と文字座標抽出

・ダウンコンバートによる PC，タブレット端末でのマルチデバイス表示（ダウンコンバートとは，デバイス用に最適な仕様へ解像度を下げて読みやすくすることで，文字だけのページはグレーからモノクロへ変換している。）

・カラーまたはモノクロでのオンデマンド印刷

の 3 点である。

これらを実現するための画像データの品質として以下の水準を設けている。

・版面の平面・歪み補正を行うが，汚れ除去等の自動処理は行わない。汚れがひどい場合は，手作業で除去する。

・OCR の精度は 99.6% を目標とする。

・カラーリングについては，カラーページはカラー，モノクロはグレースケールとする。

・解像度は原則 600dpi とする。

4.3　コスト

300 ページの書籍の電子化コストを 3,000 円，1 ページ 10 円以下と見積もる。この価格は，目次からのジャンプ機能の対応，オンデマンド印刷用と OCR の品質確保のための作業を含んで想定したものである。

あくまで筆者の現在の認識に基づくものであるが，書籍の電子化を請け負う大手業者に依頼してオンデマンドプリント用に出版物を電子化する場合，OCR 処理なしで 10,000 円程度かかる。一方，一般的な自炊代行業者では 300 ページの書籍の場合，OCR 付きで 300 円程度である。この違いは，ビジネスレベルで使用可能な品質保証があるか否かである。

また，経済産業省のコンテンツ緊急電子化事業でのコストは，OCR 処理を行わず，500 ページ以内で 12,000 円である。それに比べ，北米を中心に展開している Internet Archive では裁断しない電子化で，1 ページ当たり 0.1 ドルと言われている。この現況を考慮すれば，電子化の品質確保とコスト削減の両立のための研究開発を国内で活性化していく必要があろう。

4.4　OCR 識字率

OCR 識字率は 99.6% を目標とした。商用の OCR 用ソフトウエアはそれぞれに個性があり，版面の種類によって使い分けることになる。本プロジェクトで主に対象とする 1980 年代以降の日本語学術書では，文字だけの版面であれば目標に近い成績を出すことが可能であるが，以下のような課題がある。

a）縦書きと横書きが混在している場合など複雑な版組みを持つ版面は，その版組みが識別できずに誤動作を起こすことが多い。

b）図・表・数式・化学式など，文字コードに変換不可能なものは意味不明な記号文字列へ変換されてしまう。

c）古くて汚れのある書籍や旧漢字の書籍は，極端に精度が落ちてしまう。

a）については，学術書の特性を OCR ロジックへ組み込む，b）については，文字列へ変換できないものは変換しないよう読み飛ばす，などの工夫が可能であると考えている。c）については，できるだけ埃

取りをした上でスキャンした画像を保存し，OCR の技術的な進化に期待するしかないだろう。

また，実際に OCR 処理してみたところ，モノクロでスキャンした画像より，カラーやグレーでスキャンした画像の方が高い識字率を示す結果となった。これは，OCR ソフトの側でもそのソフトにとって最適な画像加工が行われているからであり，スキャン画像と OCR ソフトとの間の整合性に調整の余地があると言える。スキャンから OCR までの一貫した工程の最適化を導くことが可能であると感じている。

また，本プロジェクトでパンチ入力によるテキスト作成ではなく OCR 処理を前提としているのは，メモ機能などのために，文字の版面上での座標を取得する必要があるためである。

OCR 識字率について補足すると，2010 年当時では日本語書籍をスキャンして電子書籍化していたのは，NetLibrary と Google Library Project，そして国立国会図書館の近代デジタルライブラリーぐらいであった。この内，全文検索機能をもつプラットホームである NetLibrary で採用していた OCR 識字率は 99.97% であった（たぶん，英語の OCR の識字率をそのまま採用していたのではないかと思っている）。この識字率は，日本語の場合，機械処理だけでは達成できないため，OCR 後の校正処理を行うことになり，電子化コストが 1 冊当たり 10 万円を超えてしまっていた。一方で本学が参加した Google Library Project では，できるだけ良い品質の画像を取得し，その時点で可能な OCR 処理を行い，識字率が悪くても校正作業は行っていない。識字率向上は OCR ソフトのレベルアップで行うという，校正処理で対応するのとは対極的なポリシーであった。

本プロジェクトでは，コストの問題から Google Library Project の考え方を採用し，OCR 処理後の校正は行わず，技術面の改善で OCR 処理の識字率を上げることを目標とした。

4.5 書籍を電子化する場合の注意点

本プロジェクトでの作業を通じて得られた，書籍電子化の際に注意すべき点を述べる。

a）空白ページの取り扱い

紙の書籍ではページ送りの調整のために空白ページを挿入する。本プロジェクトでは，電子化に当たって空白ページをそのまま差し込むかが議論になった。最初は，空白ページは無意味という判断で削除していたが，左右のページ合わせが変わってしまうこと，ページ数がずれてしまうなどの問題が顕在化したため，空白ページをそのまま挿入するよう方針を変更した。

b）目次のページ数と実ページ数の違い

ページ送りバーでのページ表示や，目次からのページジャンプを実装する際に，目次でのページ（論理ページ）と実ページの違いをどのように吸収して実装するか議論となった。現在は，論理ページの 1 ページと対応する実ページ数を指定してオーサリングしている。

4.6 フォーマットについて

印刷業界では，JPEG は非可逆圧縮のため敬遠され，TIFF の非圧縮フォーマットが採用されることが多い。しかし，JPEG の圧縮と操作の容易さ，それが可能にする大量の画像の保存は魅力的で，JPEG でも高圧縮にしなければ本プロジェクトの目的は達成できるものであった。それよりも，スキャン画像の加工・ダウンコンバート手法による劣化のほうが問題となることが多い。本プロジェクトでは，JPEG にはモノクロの圧縮が適用されないため，結果的には PNG ファイルを採用することにしたが，非圧縮フォーマットに固執する必要は全くないという結論が得られたのは有意義であった。

5 プラットホームおよびビュアーの実装とコンテンツの流通

5.1 プラットホーム，ビュアーの実装の諸問題

2.1 で触れたように，本プロジェクトでの利用端末はマルチデバイスを対象とした。開発の初期段階では，HTML5＋XML を基本アーキテクチャーとしてウェブブラウザのローカルストレージに電子書籍データを格納する仕組みを実装していた。しかし，評価の過程で多くの問題が発生し，当初の仕組みでは実用には耐えられないと判断し，マルチデバイスで同じ HTML5 を稼働させることを断念して，デバイスの種類毎のアプリケーション開発を行うことにした。以下にその経緯と問題点を説明する。

（1）iPad の問題

初代 iPad では，HTML5 で実装したものは性能が出せなかったため，iPad 用にアプリを開発することにした。しかし，iOS はバージョンアップが早く，そ

の度にアプリの改修や再配布が必要となった。しかし，App Store でのアプリの配布に至らず，オフラインで人手により配布するしかなかったことは実験の大きな障害となった。また，2012 年の新 iPad では画像解像度が高くなったことで，これまで作成した画像では粗く見えてしまうようになってしまったため，画像の調整が必要になっている。

（2）PC の問題

HTML5 に対応しているウェブブラウザは Apple 社の Safari，Google 社の Chrome だけで，Microsoft 社の Internet Explorer は対応できていない。この制約は PC での実験を広げるための障害となった。また，HTML5 に対応していても，電子書籍を稼働させるためにはブラウザのデータベース容量のデフォルト設定値が少なく，デバイス毎に設定を変える必要もあった。

（3）Wi-Fi の問題

貸出利用の仕組みはダウンロード型とし，電子書籍データのファイルを貸出手続きの都度 1 冊分丸ごとダウンロードすることを前提とした。しかし，ファイルサイズが大きく，中には 500 ページを超える書籍もあり，特に iPad 等での Wi-Fi を使ったダウンロードに時間がかかるという問題が起こってしまった。今後は，オフラインでの利用とダウンロードの容易さ（書籍の分割等）の検討が課題である。

5.2　インターネット上のコンテンツとしての流通

（1）日本語電子書籍の URI

電子書籍へリンクする URI については，紙の書籍に ISBN が付与されている場合はそのまま ISBN を利用した。問題なのは ISBN が付与されていない書籍の URI である。国際的には，OCLC の ID や LCCN が一般的であるが，国内にはそういう国際番号が存在しない。これは極めて重要な課題であり，日本として早急に検討を進める必要がある。

（2）マイクロコンテンツとしての流通

今回の利用実験でも明らかなように，学術書が電子書籍になった時点で紙から開放されインターネット上のコンテンツとなる。その時点で 1 冊の電子学術書は分割されマイクロコンテンツ化し，インターネット上の多様な情報とのリンクを求められる。そうなった場合，図書館システム上ではどのように対応すればよいのだろうか。目録データは，紙として流通する単位を元に，図書館職員の労力と使用する図書館システムの性能がバランスする範囲で作成すればよかったが，今後，蔵書の電子書籍化が進んだ場合，メタデータとしての流通，マイクロコンテンツとしての流通に対応していく必要がある。

6　最後に

国立国会図書館のマスデジタイゼーションでは，近代デジタルライブラリーが拡張されたほか多くの成果を残している。その一つとして，著作者から許諾の得られた博士論文が公開され，各大学にもそのデータが配布された。これは一定の成果として評価できるものの，デジタイズの技術面では問題があったと考えている。大学に配布された博士論文画像は，コントラストがなく，PC 表示や印刷しても読みにくいものが多い。それを補正するにしても画像特性が不明のため，まとめて加工することは難しい。短期間のプロジェクトで時間が無かったことは理解できるが，スキャン時に画像問題の技術に関する議論が必要だったと強く感じている。

また，1990 年代の電子図書館プロジェクトでの高精細画像やデジタル・アーカイブにおける電子化と，インターネット上に展開するためのマスデジタイゼーションとの違いからも得るところが多いと考えているが，紙面の都合上別の機会に譲ることにしたい。

マスデジタイゼーションは，大規模に展開するため，製産工程の中で品質を作りこむ覚悟が必要である。工業製品の品質確保は日本のお家芸であるはずで，本プロジェクトで得られる様々な技術評価が，今後の日本語書籍のデジタイズに活かせればという思いで本プロジェクトを進めている。

謝辞

システム開発と技術的な挑戦は，本プロジェクトでシステム担当していただいた京セラコミュニケーションシステム株式会社（KCCS）の貢献によって進めることができた。この場を借りて深く感謝したい。